Boston University
Electrical & Computer Engineering

EC 464 Senior Design Project

Final Project Test Report

Portable Language Translator

By

Team 23

Portable Language Translator

Team Members

Yohan Kim yohank@bu.edu
Ryan Liao ryanliao@bu.edu
Cristian Palencia cris0211@bu.edu
Andrew Nguyen aynguyen@bu.edu

For the hardware setup we have a Raspberry Pi 5 that interfaces with all the relevant components. Specifically, it interfaces with the following: USB camera, USB speaker USB microphone and a touch display all of which are mounted throughout the 3D printed encasing. The Raspberry Pi will be connected to the user's phone hotspot using 1 of 3 UI tabs where a user can select their internet and insert their password through a virtual keyboard displayed on the touch display.

For the language translation, a dedicated UI tab allows users to configure settings for the language translation, such as, base language and voice gender before the translation starts. One device is used for the conversation. The users must wait for the translation and playback to fully finish before beginning the next phrase. The program captures audio from the USB microphone, splits it into chunks and uses voice activity detection software to determine if someone is speaking. Once it detects no speech, it sends the audio file to the google cloud for the transcription, translation, and finally text-to-speech. The audio file is sent back to the program which it plays from the system's speaker and to display the translated text on the translation UI tab.

For the ASL gesture detection and prediction, the external camera (USB 2.0 Camera) will record a live video stream and the script will initialize this video feed with OpenCV. First processing camera frames is done with Mediapipe's hand landmarks model to extract hand landmarks and then grouping these landmarks from the last 30 frames into a sequence. This sequence is fed into our LSTM (Long Short Term Memory) model that outputs the probability for each gesture. When a specific gesture's confidence is greater than the threshold value, that gesture will be then chosen as the detected gesture. An asynchronous thread handles this inference process, ensuring that consistent and confident predictions are added to a running sentence, which is eventually converted into speech for translation. Once the sentence is done playing, the script will then listen for the audio and save the transcription in a .txt file where it will be displayed in the UI. Then the script will go back to detecting ASL gestures and the cycle will restart.

Measurements and Data:

Language tests

Accuracy: We wanted to collect some data to demonstrate the accuracy and functionality of our language translation code. The data collected in Tables 1-6 below demonstrates different translation paths such as English - Korean and Korean - Spanish. For each row we notice: "Phrase Said" being the words spoken by the user, "Phrase Recorded" being the words picked up by the software, "Translated Phrase" being the final translation that is determined and outputted by the software. Under "Errors?" we write the degree to which the translation and transcription were incorrect. Transcriptions define whether the actual translation was wrong while the Translation then describes whether given a transcription we were able to correctly output it. As an additional note this is the same data as prototype test one and two since we should observe the same performance as long as the microphone on the Raspberry Pi is capable of picking up the intended phrase.

Phrase Said	Phrase Recorded	Translated Phrase	Errors
Hello, how are you?	hello how are you	안녕하세요. 어떻게 지내세요	Transcription: no errors Translation: no errors
Can I get a large coffee with two sugars and no milk, please?	can I get a large coffee with Two Sugars and no milk	Two Sugars와 우유 없이 큰 커피 한 잔을 마실 수 있나요?	Transcription: no errors Translation: no errors
Excuse me, how do I get to the nearest train station?	excuse me how do I get to the nearest train station	실례합니다. 가장 가까운 기차역까지 어떻게 가나요?	Transcription: no errors Translation: no errors

Table 1: English - Korean Translation

Phrase Said	Phrase Recorded	Translated Phrase	Errors
Do you have fresh strawberries or are they out of season?	you have fresh strawberries or are they out of season	¿Tienes fresas frescas o están fuera de temporada?	Transcription: missing word Translation: no errors
Can I try this jacket on in a medium size?	can I try this jacket on in a medium sized	¿Puedo probarme esta chaqueta en talla mediana?	Transcription: minor error Translation: no errors
What time are we meeting tomorrow for	what time are we meeting tomorrow for	¿A qué hora nos reuniremos mañana	Transcription: missing word

lunch at the new restaurant?	lunch at the new	para almorzar en el nuevo?	Translation: no errors

Table 2: English - Spanish Translation

Phrase Said	Phrase Recorded	Translated Phrase	Errors
¿Hola, tienes fresas frescas o están fuera de temporada?	Hola tienes fresas frescas o están fuera de temporada	안녕하세요. 신선한 딸기가 있나요? 아니면 제철이 아니나요?	Transcription: no errors Translation: no errors
¿Puedo probarme esta chaqueta en talla mediana?	puedo probarme esta chaqueta en talla mediana	이 재킷을 미디엄 사이즈로 입어봐도 될까요?	Transcription: no errors Translation: no errors
¿A qué hora nos reuniremos mañana para almorzar en el nuevo restaurante?	a qué hora nos reunimos mañana para almorzar en el nuevo restaurante	우리 내일 몇 시에 새 식당에서 점심 먹으러 만나요?	Transcription: no errors Translation: no errors

Table 3: Spanish - Korean Translation

Phrase Said	Phrase Recorded	Translated Phrase	Errors
안녕하세요. 어떻게 지내세요	안녕하세요 어떻게 지내세요	Hola, cómo estás	Transcription: no errors Translation: no errors
우유 없이 설탕 그게 큰 커피 주세요	우유 없이 설탕 그게 큰 커피 주세요	Me gustaría un café grande con dos azúcares y sin leche.	Transcription: no errors Translation: no errors
실례합니다. 가장 가까운 기차역까지 어떻게 가나요?	실례합니다 가장 가까운 기차역까지 어떻게 가나요	Disculpe, ¿cómo llego a la estación de tren más cercana?	Transcription: no errors Translation: no errors

Table 4: Korean - Spanish Translation

Phrase Said	Phrase Recorded	Translated Phrase	Errors
신천한 딸기 있나요 아니면 제철이 아니나요	신천한 딸기 있나요 아니면 제철이 아니나요	Are there any fresh strawberries or are they out of season?	Transcription: no errors Translation: no errors

이 재킷을 M 사이즈로 입어봐도 될까요	이 재킷을 M 사이즈로 입어봐도 될까요	Can I try that jacket in size M?	Transcription: no errors Translation: no errors
우리 내일 몇 시에 새 식당에서 점심 먹으러 만날 거야	우리 내일 몇 시에 새 식당에서 점심 먹으러 만날 거야	What time are we meeting for lunch at the new restaurant tomorrow	Transcription: no errors Translation: no errors

Table 5: Korean - English Translation

Phrase Said	Phrase Recorded	Translated Phrase	Errors
Hola, cómo estás	Hola cómo estás	Hello how are you	Transcription: no errors Translation: no errors
¿Puedo tener un café grande con dos azúcares y sin leche por favor?	puedo tener un café grande con dos azúcares y sin leche por favor	can I have a large coffee with two sugars and no milk please?	Transcription: no errors Translation: no errors
Disculpe, come alcanzo llegar al estacion de tren mas cercano?	Disculpe cómo alcanzo a llegar a la estación de tren más cercano	excuse me, how do I get to the nearest train station?	Transcription: minor errors Translation: no errors

Table 6: Spanish - English Translation

Latency: Referring to Table 7 below, we were able to record some measurements for the latency of our software for different sized phrases. For context, the data was collected using English - Spanish translations (English phrase translated into Spanish). In our collected data, latency is measured by the time the untranslated audio is sent to the cloud to when the translated audio playback starts. This was measured by the Python timer method to provide accurate times.

Specifically, as we move down the table we observe that we are testing with larger and larger phrases to test the correlation between different sized phrases and latency.

Since is the same data collected from Prototype Test Two as there have been no changes to improve or worsen latency.

Phrase	Word count	Latency (seconds)
Hello	1	0.34
Hello, how are you?	4	0.59
I would like to have a water please	8	0.64
Where is the nearest hospital? I am having a serious allergic reaction	12	0.83
I always enjoy quiet evenings at home, reading a good book with a cup of tea	16	1.19
Let’s plan a weekend getaway soon. It’s been too long since we all had some fun together outside at the beach	21	1.21
When we moved into the new neighborhood, we didn’t know anyone, but our neighbors were so welcoming. They even brought over cookies and invited us to their backyard barbecue. It’s been a great place to live ever since	38	1.90

Table 7: Latency Measurements - Data collected on the Raspberry Pi for latency for different sized phrases

ASL Test

Accuracy: With our ASL software, we wanted to test whether the model could accurately detect and predict signed gestures when a user attempts to gesture in ASL. Table 8 below demonstrates attempted gestures at different distances, with the outputs indicating whether each gesture was correctly detected.

Gesture	Distance from Camera	Correct?
Hello	.33 meters	True
Hello	.67 meters	True
Hello	1 meter	True
Bathroom	.33 meters	True
Bathroom	.67 meters	True
Bathroom	1 meter	False
Thank you	.33 meters	True
Thank you	.67 meters	True
Thank you	1 meter	True
Help	.33 meters	True
Help	.67 meters	False
Help	1 meter	True
Yes	.33 meters	True
Yes	.67 meters	False
Yes	1 meter	True

Table 8: ASL Gesture Translation Data - Table demonstrates how an intended gesture is detected and translated at different distances using our ASL gesture model.

Latency: We also wanted to record measurements for the latency or processing time for the time it takes for the model to process and play the gesture into the speaker. Latency is measured by the time the gesture is detected to when it is finished playing on the speaker. Table 9 demonstrates the difference gestures and then the corresponding latency in seconds.

Gesture	Latency (seconds)
Hello	2.50
Bathroom	2.65
Thank you	2.58
Help	2.19
Yes	2.38

Table 9: ASL Latency Measurements - Demonstrates how long it took to process a given gesture and play it on the speaker.

Microphone Test

Functional Distance: In order to test the functional distance of our new microphone, we used a similar testing method compared to last semester. Referencing Table 10, we were able to come up with a scoring guideline for any given audio recording. With this scoring guideline we were then able to test the microphone at different distances and at varying audio levels. Based on the audio recording we would then be able to give it a quality score between 0 and 4 (shown in Table 11). These are the same values as from Prototype Testing Two as there have been no changes to the microphone since then.

Quality Score	Meaning
0	Inaudible
1	Muffled and distant
2	Language can be understood
3	Clear audio
4	Perfect audio recording

Table 10: Quality Score Table - Ranking given to an audio recording based on overall quality.

Distance (m)	Whisper	Conversational Speaking	Yelling / Loud Speaking
0.25	3	4	4
0.50	2	4	4
0.75	2	4	4
1	1	3	4
1.25	1	3	4
1.50	0	2	3
1.75	0	1	3
2	0	0	3

Table 11: Microphone Performance - Performance of USB microphone at varying distances and audio levels.

Battery Life

We determined that at max performance the Pi would draw a maximum of 1.6A from an external power supply. With these considerations we purchased a 5000mAh battery that would at max performance would last us 3.125 hours.

Device Wearability / Portability

The dimensions of our device 139.3mm x 51.2mm x 111.5mm and includes shoulder straps such that a user can easily wear the device for translations.

Conclusions:

Language

The accuracy of the transcription and translation is the same as before. This shows that it is still highly accurate even when moved to the Raspberry Pi 5 and into a physical encasing. Even with the minimal errors, any ASL individual reading the transcribed text or other party speaking a different language should be able to understand the meaning of each phrase. The accuracy is also above our minimum expected percentage accuracy of 90%. More importantly is the latency is sufficiently within our range of less than 4 seconds when the software is now moved to the Raspberry Pi 5.

ASL

Over the course of the semester, we continued to build upon our initial LSTM model by significantly expanding our dataset. Compared to the 10 data points per gesture from the second prototype, we now have about 200 data points per gesture and as expected this has led to a significant increase in performance. The model's accuracy has now risen to 80%, showing improvement in our ASL gesture recognition capabilities. This reinforces our confidence in the model's potential and validates the importance of robust data collection. Although below our original expected accuracy we understand that with even more data points we could only expect to see this performance to continue increasing. Latency remains consistent, with most delays still attributed to the audio playback of translated gestures, rather than the recognition process itself, which continues to perform efficiently even as the model becomes more accurate.

Microphone

One of our requirements of the project is defined by how far someone can be from the device for full translation functionality. The requirement we defined was a distance of one meter away from the device. This requirement was already completed from our previous microphone from last semester. The change came from the desire to implement a more robust system that would be easier to implement into a physical encasing. Additionally, although we met the bare requirements from our microphone last semester we still noticed a lot of room for improvement. From the collected data we see that the microphone works well within our parameters.

Battery Life

Our minimum battery life requirement was 2 hours at max performance. With the currently implemented battery pack, we are able to get an additional hour of battery life. With this in place we can get full use of our device for extended periods of time while being able to recharge the battery pack whenever the device dies.

Device Wearability

Although slightly larger than originally planned, the addition of the straps provides the user with a lot of support to carry the device when needed. The straps provide significant shoulder support that makes carrying the device around very simple and easy.