ATT: Automatic Transcription & Translation with the Magic Leap 2

Alejandro Cuadron Lafuente, Elisa Martinez Abad, Ruben Schenk, Sophya Tsubin

ETH Zürich

{acuadron,emartine,rschenk,stsubin}@ethz.ch

Abstract

This report illustrates the process of building a Mixed Reality (MR) application to facilitate real-time multilingual communication using the Magic Leap 2 glasses. The application utilises the power of OpenAI's Whisper model for accurate transcription and translation, enabling users to overcome language barriers in an unobtrusive MR environment. Created with Unity, the application features a user-centred design optimised for simple and intuitive interaction, and employs a Flask API for reliable backend services. The project concludes with a detailed user study, employing the System Usability Scale to quantitatively measure user experience. Statistical analysis of the study's data suggests that prior experience with MR significantly enhances usability perceptions, while other demographic factors show no substantial impact. The study's findings are extended by user suggestions that formed the last iteration of the applications development. The successful development and validation of this application shows a great example of how MR can be used to create a new dimension to human-computer interaction and sets a baseline for future research in the field.

1. Introduction

In the merging field of MR, breaking down language barriers stands as an interesting and important challenge with great implications for global communications. Our project, based on the state-of-the-art Magic Leap 2 (ML2) glasses, has advanced this vision by building a real-time translation application for MR environments. By making use of OpenAI's Whisper model alongside a Flask API for backend tasks, we have created an application that not only transcribes spoken language but also translates it into English. This innovation was presented and validated during an interactive poster session, where we conducted a user study to evaluate the application's real-world efficacy.

2. Related Work

The intersection of MR technology with language translation is a relatively unexplored field within human-computer interaction, though some advancements have been made in the past years. Pioneering efforts by Toyama *et al*. [6] have demonstrated the potential of MR in enabling communication across different languages through a head-mounted display. Our project builds on this idea, extending it with the integration of speech recognition and translation systems. The notable development of Whisper by Radford *et al*. [4] has been instrumental to our application, providing a solid basis for our goals in real-time transcription and translation.

3. Methodology

3.1. Mixed Reality Hardware

Our project is based on the Magic Leap 2 glasses as the primary hardware platform. The ML2 glasses are powered by an AMD 7nm Quad-core Zen 2 processor and an AMD RDNA 2 GPU, complemented by other specialised computational units [1]. Given the significant processing power required for efficient operation of the Whisper model, the GPU capabilities of the ML2 glasses were an important aspect for this project. Although the ML2's GPU is robust, it didn't meet the requirements for optimal performance of our application. To address this, we utilised a Backend for Frontend (BFF) design pattern. This approach allowed us to leverage the power of an external, high-performance GPU with substantial dedicated memory, ensuring seamless operation of the application.

3.2. Application Development with Unity

For the development of the application's front end, we used Magic Leap's integration with Unity3D and Microsoft's Mixed Reality Toolkit (MRTK3). MRTK3 was chosen for its great amount of pre-implemented MR features in Unity, such as UI layouts and interactive buttons.

The structure of our Unity project is centred around two primary Unity scenes, supported by a collection of C#

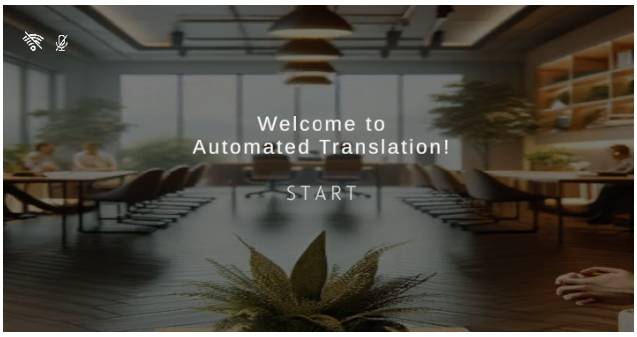

Figure 1. Application main menu

Figure 2. Translation screen

scripts that control the behaviour of the application. The initial scene serves as the main menu of the application (Figure 1) that transitions to the main translation screen (Figure 2) once the user interacts with the ”START” button. Both scenes feature icons at the screen corner, indicating the status of the connection to the translation service backend and audio recording permissions. The translation scene is designed to prominently display the translated text at the bottom of the screen for easy viewing.

The application's behaviour is managed by different scripts:

- **AppStart.cs**: Executes at the launch of the application, verifying the server connection via a GET request to the API's base URL and checking microphone permissions. The outcome of these checks is then represented by updating the corresponding status icons on the screen.
- **AudioHandler.cs**: Responsible for audio recording management. It initialises an ML2 audio capture object to capture microphone input and sends audio data to the translation API in intervals of 3 seconds for real-time translation.
- **TranslationAPI.cs**: Handles the process of sending POST requests to the `/translate` endpoint with the audio file in WAV format. It also manages the display of translations on the user interface once the data was successfully received.
- **UIManager.cs**: Controls scene transitions and the persistence of the icons across different scenes.
- **WavUtility.cs**: A utility script for converting Unity AudioClip data into WAV format for server transmission. In contrary to the rest of the scripts, this script was not developed by us but sourced from GitHub [3].

3.3. Backend Integration

For the backend of our MR application, we used a web application built on the Flask Python framework. Initially, we also considered using web sockets for communication with the ML2 glasses. However, we ultimately opted for a RESTful API approach due to the much simpler integration with the frontend. The final architecture of our system is illustrated in Figure 3.

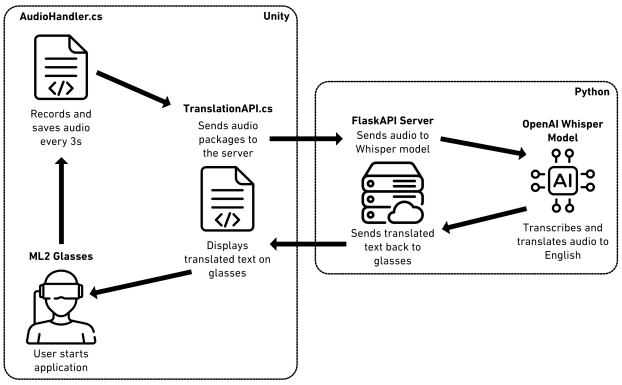

Figure 3. System overview.

The API consists of the following endpoints that process data from the glasses:

- **/supportedLanguages**: Provides a list of languages that are supported by OpenAI's Whisper model for translation and transcription.
- **/transcribe**: Delivers the inferred language and the transcription of a submitted audio snippet in text form.
- **/translate**: Offers both the inferred language and the English translation of an audio snippet, also in text form.

A crucial component of our backend is the integration of a modified, faster version of OpenAI's Whisper model [5]. The large language model is available in various sizes and can be executed on multiple platforms. During the development phase, we experimented with different model sizes

and tested their performance on two types of device settings: CPU and CUDA, with the latter proving to be much faster.

When assessing the different model sizes, we noted a trade-off between size and performance. While smaller models offer faster inference times, their accuracy in translations was unsatisfactory. Therefore, we shifted our focus to medium and large variations. The comparison of inference times between different model sizes and different sized audio snippets is shown in Figure 4.

Another important constraint we had to account for was the memory requirement of these models. OpenAI's default large model demands approximately 10GB of VRAM, exceeding our available resources. By switching to the faster variant of the Whisper model, we successfully reduced the VRAM requirement of the large model to 4GB, thus achieving a balance between translation accuracy and hardware limitations.

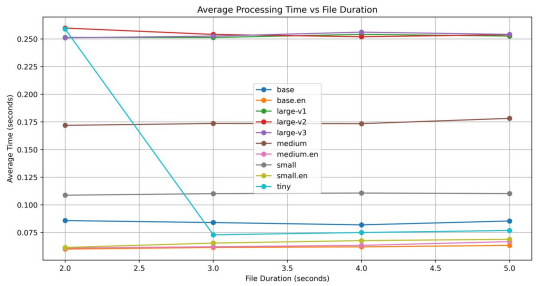

Figure 4. Evaluation of different-sized Whisper models and their effect on translation latency.

Given a 3-second audio snippet sampling rate, the fast large model running on a GPU achieved an average inference time of 0.25 seconds. This resulted in an overall latency of approximately 4-6 seconds, comprised of 3 seconds for recording the audio snippet, and 1-3 seconds for network latency and the processing of the audio.

4. User Study

4.1. Study Design

With our user study the goal was to evaluate the usability, the design, and the performance of our MR application. To achieve this, we utilised a combination of the System Usability Scale (SUS) [2] and additional custom questions to assess more specific aspects of our application, including noise perception, translation latency, text displaying times, and quality of translations. In total, the study included the 10 standard SUS questions, five custom questions, and four questions related to demographics. In detail, the additional questions included:

- **Custom Questions (Strongly Disagree 1 - 5 Strongly Agree):**
  - I found the environment to be noisy when I used the application.
  - I found the translation latency was high and prevented me from following the conversation in real time.
  - I thought that I had enough time to read the translation properly.
  - I thought the English translation was good enough for me to have a meaningful conversation.
  - I thought the translation from a non-English language I spoke was accurate.
- **Demographic Questions:**
  - Do you have prior experience with Mixed Reality? (Yes / No)
  - Are you a student of the Mixed Reality course? (Yes / No)
  - What is your age? (0-10 / 11-17 / 18-25 / 26-35 / 36-45 / 45-64 / 65+)
  - What is your gender? (Female / Male / Other)

4.2. Demographics

The study involved 21 participants with varying levels of experience with MR. A majority of the participants had prior experience with MR (n=16), and most were not students of the MR course (n=13). The age distribution was primarily between 26-35 years (n=11), with a smaller representation of ages 18-25 (n=9) and 36-45 (n=1). Gender distribution included more males (n=19) than females (n=2).

4.3. Procedure

Participants were invited to interact with the MR application during the poster presentation of our project. They received a brief orientation on using the ML2 glasses, including how to wear them, operate the controller, and navigate to start the ATT application. Once the application was initiated, members of the group engaged with the participant in Spanish and in Ukrainian for approximately one minute each to simulate a real-life conversation scenario in a multilingual setting. Subsequently, participants were asked to respond in a language other than English or Spanish, preferably their native language, for around thirty seconds. This multilingual exchange was designed to test the application's real-time translation capabilities within an environment of changing languages. Upon completion of the interaction, participants were asked to fill out the SUS questionnaire followed by additional custom questions to assess their experience.

4.4. Quantitative Assessment

The SUS scores averaged at 87.62 with a standard deviation of 8.35, indicating a high level of usability among participants. The analysis of the custom questions revealed that participants generally found the time the translation was displayed enough to read it (M=3.90, SD=0.83), however, latency was was perceived by some to be too high to follow the conversation in real time (M=2.96, SD=1.07). The quality of English translations was rated highly (M=4.48, SD=0.60), as was the accuracy of non-English translations (M=4.52, SD=0.81), even though the environment was found to be noisy by some participants (M=2.71, SD=1.27).

4.5. Analysis

To gain deeper insights into our user study data, we initially conducted cross-tabulations of SUS scores with demographic variables. This preliminary analysis, detailed in Table 1, aimed to identify potential influences of demographic factors on SUS scores. We observed that prior experience with MR seemed to play an important role in SUS scores, age and gender a less influential role, and the enrolment in the MR course seemed to make no difference in the perceived usability.

Demographic Variable	Average SUS Score
Prior Experience in Mixed Reality	
No	79.50
Yes	90.16
Student of the Mixed Reality Course	
No	87.69
Yes	87.50
Age Group	
18-25	88.06
26-35	86.36
36-45	97.50
Gender	
Female	80.00
Male	88.42

Table 1. Cross-tabulation of average SUS scores with demographic variables.

Building upon these observations, we employed statistical tests to validate these findings. An ANOVA test was conducted to assess the influence of age groups on SUS scores, revealing no significant differences, thus supporting our observation from the SUS cross-tabulations that age does not affect usability perception. Similarly, a point-biserial correlation test was applied to verify the impact of prior MR experience, resulting in a significant correlation that highlights its influence on usability ratings. In contrast,

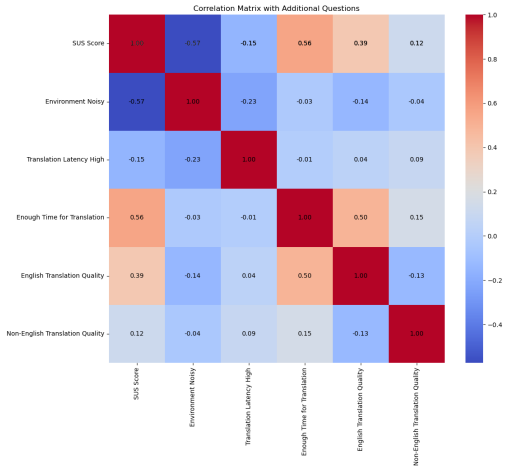

Figure 5. Correlation matrix illustrating the relationships between SUS scores and additional questionnaire items.

the same test applied to course enrolment showed no significant correlation. Additionally, a t-test for gender differences in SUS scores indicated no significant variance. The detailed test results are shown in Table 2.

Test	Statistic	P-value
Age Group	F(0.8225)	0.4552
MR Experience	r(0.5572)	0.0087
Course Enrolment	r(-0.0115)	0.9607
Gender	t(1.3882)	0.1811

Table 2. Summary of statistical test results.

Moreover, we also explored correlations between SUS scores and responses to the custom questions to understand how specific aspects of the application influenced user experience. This analysis was important to test whether factors like environmental noise, translation latency, and the quality of translations were determinants in the overall usability rating. The correlation matrix, depicted in Figure 5, revealed important patterns: a moderate negative correlation between SUS scores and perceived environmental noise (r = -0.57), suggesting that noise adversely affects usability ratings. In contrast, the quality of English translations showed a positive correlation with SUS scores (r = 0.39), underscoring its significance for user satisfaction. Additionally, the time allowed for reading translations was also positively correlated with usability (r = 0.56), indicating the importance of pacing in the user experience.

4.6. Qualitative Assessment

Alongside quantitative metrics through the questionnaire, we also collected qualitative feedback and comments

participants revealed during the testing of the application. The most common suggestions were ergonomic adjustments, such as repositioning the application's start button and the translated text to be less obtrusive within the user's field of view. Specifically, recommendations were made to place interactive elements further away in the virtual space and to lower the displayed translation away from the middle of the user's field of view. The efficacy of the translations feature was notably praised, even when testing less common languages like Romanian and Icelandic. Overall, the remarks emphasised the project's success and the engaging outcome of the experience.

5. Results and Discussion

5.1. Sampling Rates and Model Performance

In aiming for optimal performance, various sampling rates and OpenAI Whisper model sizes were evaluated for their impact on latency. Our investigations, as visually represented in Figure 4, showed that the size of sampled audio-snippets does not remarkably affect the translation latency. There was, however, a big difference in latency based on which Whisper model was used to transcribe and translate the audio-snippets. Notably, models with a latency greater than 0.17 seconds delivered accurate and reliable translations, which was a critical requirement for the application. These findings were important for balancing the applications responsiveness, accuracy, and computational efficiency.

5.2. User Feedback and System Usability

The quantitative and qualitative results from the user study provided an interesting and complete view of the system from a users perspective. User suggestions emphasised the need for ergonomic adjustments, which were implemented in the final iteration of the application. This included moving the interactive elements further back in virtual space and lowering the translated text, out of the direct sight of the user. The positive feedback and constructive comments, particularly regarding less common language translations, underscore the robustness of the translation feature and the overall success of the project.

5.3. Demographic Influence

Our data suggests that while prior MR experience influences usability perception, demographic factors such as age and gender have minimal impact. This is an encouraging sign that our MR application has the potential to cater to a broad user base, regardless of demographic diversity.

5.4. Future Improvements

The current application functionality can be further enhanced by adding spatial awareness to the UI. This feature would enable translations to appear adjacent to the speaker, particularly beneficial in multi-person conversations. Implementing this could be approached through various methods: utilising computer vision techniques to identify speaking individuals from video data, analysing incoming audio signals across multiple microphones to infer the direction of speech, or integrating a cloud service that informs the glasses of the precise locations of other participants in the conversation.

Another area of improvement is the support of translation to non-English languages. The current limitation arises from the OpenAI Whisper model's restriction to English translations. One possible workaround would be using an additional text-to-text model to translate the English translation into the target language, although this approach could lead to less accurate results. The best course of action would be to change the underlying model to one capable of direct translation into various languages.

Lastly, the introduction of a voice-to-voice translation feature would significantly benefit users who prefer or require auditory communication, such as those with visual impairments, differing educational backgrounds, or a preference for audio over visual information.

6. Conclusions

This project represents a significant contribution to the domain of MR-assisted multilingual communication. The development of an MR application that integrates real-time transcription and translation has been validated through a rigorous user study. The positive correlation between prior MR experience and usability scores reinforces the importance of user familiarity with MR technologies. However, the lack of demographic influence on usability ratings highlights the application's broad appeal. The qualitative feedback collected has been invaluable, leading to important improvements in the application's design and functionality. The results of this project set a baseline for future research and development in the MR field, opening new opportunities for human-computer interaction.

Acknowledgements

Our thanks go to Lukas Bernreiter from Magic Leap for his invaluable contributions to this project. His original idea for the project and ongoing support, including insightful ideas and answer to numerous questions about the ML2 glasses, were invaluable for our work.

We are also immensely grateful to Dr. Zuria Bauer for her assistance and prompt responses to our questions related to the course structure and the lectures. Additionally, her role in enabling us to present our project to academic guests offered us a unique opportunity to showcase our work and gain valuable insights.

References

[1] Magic Leap 2 techonology specifications. Available at https://www.magicleap.com/magic-leap-2. 1

[2] John Brooke. Sus-a quick and dirty usability scale. *Usability evaluation in industry*, 189:194, 1996. 3

[3] David Douglas. Wav Utility for Unity. Available at https://github.com/deadlyfingers/UnityWav. 2

[4] Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision. In *International Conference on Machine Learning*, pages 28492–28518. PMLR, 2023. 1

[5] SYSTRAN. Faster Whisper transcription with CTranslate2. Available at https://github.com/SYSTRAN/faster-whisper. 2

[6] Takumi Toyama, Daniel Sonntag, Andreas Dengel, Takahiro Matsuda, Masakazu Iwamura, and Koichi Kise. A mixed reality head-mounted text translation system using eye gaze input. In *Proceedings of the 19th international conference on Intelligent User Interfaces*, pages 329–334, 2014. 1