Live Subtitles using Augmented Reality

B.E. (CS) PROJECT REPORT

by

Ifrah Ishtiaq

Department of Computer and Information Systems Engineering

**NED University of Engineering
& Technology, Karachi-75270**

B.E. (CS) PROJECT REPORT

Project Group:

Ifrah Ishtiaq	CS-17132
Ameema Arif	CS-17122
Mahrukh Khan	CS-17003
Syeda Sara Akif	CS-17002

BATCH: 2017

Project Advisor(s):

Ms. Fakhra Aftab (Internal Advisor)

September 2021

Department of Computer and Information Systems Engineering

NED University of Engineering
& Technology, Karachi-75270

ABSTRACT

"Live Subtitles" is an approach towards minimizing communication hurdles, which provides real-time speech to text conversion, allowing the users to view and read the speaker's speech in live conversations. This project is implemented as an android app to ease users in their daily lives. Real-Time text display allows the speech along with Urdu translation to be overlaid on screen of the user's mobile. This technique of live subtitling is the simplest and an efficient way to allow people share real time content without any difficulty.

ACKNOWLEDGEMENTS

First of all, we are grateful to Almighty Allah for giving us the strength, knowledge and understanding to complete and deliver this project. Secondly, we are thankful to our internal supervisor Ms. Fakhra Aftab for her immense support, time and guidance during the whole course of completion of this project. We would also acknowledge all our teachers who taught and gave us the required skills which we were able to implement in our project. We are also thankful to our departmental staff and university staff, who assisted us during our stay at the university.

Table of Contents

S.No **PageNo**

CHAPTER 1

Introduction

1.1 Project Background

Human communication is an essential part of our daily life through which people share ideas, information, opinions, feelings, and experiences with one another. Communication is fundamental to the existence and survival of individuals, groups, societies, and nations. For communicating, language is the most important tool. It plays an integral role to help people build relationship with others. But language can sometimes act as a barrier between the communication processes. When two people cannot understand each other's language, then it becomes difficult for them to communicate. Other times when language is not an issue, people who are deaf or have hearing deficiency finds difficulty in communication as well.

With technological advancements, it is now easy to overcome all these communication barriers. From language translations to speech captioning, technology provides inclusion to all.

1.2 Problem Statement

In today's world, many people are facing communication barriers where they can't understand the speaker speech due to unfamiliarity with the language or they have a hard time in understanding the speaker's speech due to loss of hearing. People often face the language issue when they travel to a place where a different language is being spoken. They continuously need a translator for effective communication. Also, the number of people having loss of hearing is increasing to a great number. According to World Health Organization, Nearly 2.5 billion people worldwide ー or 1 in 4 people ー will be living with some degree of hearing loss by 2050 [1]. This situation calls for solution that deals with not only the language translation issue but also provide an effective way of communication for people with hearing deficiency.

1.3 About the Project

Live Subtitles not only helps people with all kinds of disabilities and experiences such as auditory processing disorders, dyslexia and other intellectual disabilities but also support people with different language preferences in real time. Live Subtitles increases the retention and the accessibility of the presentation as it allows the users to soak up every word from the speaker, engage with him and his content with confidence. Live captions provide reinforcement so they never miss a word, enabling them to fully absorb the content and feel engaged in the subject matter. With live captioning, people do not have to try to perform lip reading, straining to follow and trying to grasp the speech at the same time. Such a process is an exhaustive mental load, and live subtitles relieve that burden and allow complete focus and participations. It offers great support by making access of speaker's content easier for people; without having them to take their eyes off of the speaker as the video plays a vital role to help people understand and comprehend the information being provided. Hence, here the need becomes clearly visible to make real time videos available to the people having auditory problems and even more for the people to remove the gaps of their native language and also for having to miss a point while hearing. This can be best done by the use of subtitles in the real-time scenario. Live captions can be delivered to anybody; people attending an in-person lecture or a formal meeting, people having casual chatting or asking for directions on the way.

1.4 Related Work

There are many mobile and web applications to provide language translations and subtitles which are available for use but very few products provides Real-Time translation and captioning. Following are some of them:

- *Google's Captioning on Glass:* A team of researchers from the Georgia Institute of Technology developed speech-to-text software for Google's wearable technology. Using Glass and an Android-based smartphone, the app converts speech to text and displays it on the Glass heads-up display. The app, called Captioning on Glass, is free and is available at MyGlass [2].

  However, Glass is a small, lightweight wearable computer with a transparent display for hands-free work [3]. Google's Glass (Smart Glasses) is a high-tech eyewear with a cost of $1,500 which first emerged for beta-testing in 2013 but the hype was shortlived. After killing Google Glass in 2015, the search giant brought them back in 2017 but only for limited industrial uses [4]. Though it is still available online from second-hand merchants and it can still download its final update released in 2019 [5].

- *National Theatre’s Smart Caption Glasses:* is one such example which received a lot of attention when it was launched for its use in theatres. It makes use of AR to fully integrate captions into the artistic elements of performances. Location markers are used to ensure that the captions are always visible from any seat. When wearing the glasses, users will see a pre-loaded transcript of the dialogue and descriptions of the sound from a performance displayed on the lenses of the glasses [6]. It costs around $1,200 a pair [7].

- *Epson Moverio Smart Glasses:* An Israeli tech startup GalaPro, working with the Schubert Organization, is testing out its smartglasses, aimed at theatergoers who have a hearing impairment. The glasses, which uses Epson’s Moverio smart frames, allows the user to view captions in real-time within the lenses – similar to the GalaPro app technology. The company actually wanted to add another live text option beyond looking down at the GalaPro app on the phone. With the new smartglasses, the subtitles are meant to sit on the edge of the stage in the viewer’s eye [4]. However, users struggled to keep the wide-framed glasses in place throughout the show. They range in price from $700 to $1,200 depending on the Epson model. The live captions through the glasses rely on a pre-loaded script from the show as well as GalaPro’s voice-recognition software [8]. In general, experts, analyzing the means of help for people with severe hearing problems, came to the conclusion that this tool is useful [9].

- *AR Captioning* is an AR app that approaches to real-time captioning for people who are deaf and hard of hearing. Unlike traditional captioning, which uses an external, fixed display (e.g., laptop or large screen), this approach allows users to manipulate the shape, number and placement of captions in 3D space which not only helps people with hearing deficiency but is also used by students in universities while attending lectures to get a better understanding [10].

- *SyncWords* is an online platform for providing automated subtitling, translations, dubbing and transcriptions for both real-time and pre-recorded content. It uses AI, Machine Learning and web-based tools for providing the solutions. It is a pay-per-use service which starts for free and then $0.60 per minute [11].

- *VoCaption* is a live automated captioning and subtitling solution which is built around Artificial Intelligence and delivers real time speech-to-text processing. It provides broadcasters with accurate, reliable, real-time and cost-savings live captions [12].

- *Live Subtitles App* is one of Shravan Apps by Oswald Labs. It offers “Live Subtitles” android application and is an Augmented Reality app which displays real-time conversational subtitles on your smart phone screen as someone is speaking with the use of real-time speech recognition. With this app, users with deafness can read what people are saying to understand them better. Though this provides great help to people but this app requires pre-registering and still not available to be used free by everyone. Live Subtitles is an Indian product yet to hit market. The date for release is not announced [13].

- *Live Caption App* is a mobile application available to Apple Users only with free trial and then subscription for $2.99/month. It allows the user to caption while speaking. The languages available are Spanish, French, Japanese and Sanskrit. It uses voice recognition software through which your spoken words appear live on your device. It enables people with hearing loss talk to anyone face-to-face with real time speech-to-text conversion [14].

- *Live Transcribe* by Google is a mobile application developed by Google for the Android operating system. It uses Google’s automatic speech recognition and sound detection technology to provide users with free, real-time transcriptions of their conversations and sends notifications based on their surrounding sounds at home. The notifications feature aware the user with important or urgent situations at home, such as a fire alarm or doorbell ringing, so that the user can respond quickly [15].

- *Live Transcribe* is a mobile application for Apple users and available on Apple play store only. It provides live captioning for deaf and hard of hearing along with transcriptions in 50+ languages. It is free to install from the App store but contains built-in App purchases for real-time transcriptions [16].

- *gotalk.to on Twitter* offers Live subtitles for video gatherings using Chrome's 'Live Captions' feature for both desktop and mobile and as of now accessible in English, the feature will do its best to show what is being said in real time. Video chat can be done right in your browser on desktop and mobile, both on Android and iOS. No apps and no sign up is required. Share your screen, record gatherings, stream live, and more [17].

1.5 Goals and Objectives

This android application is designed to facilitate and enhance the communication process between people specifically for the ones having hearing deficiency. It provides the idea of inclusion to communicate, so that no one feels excluded in their daily real-life activity by:

- Converting speaker's speech as subtitles on your application screen.
- Along with providing the Urdu translation of speech as subtitles.

CHAPTER 2

Speech Recognition and Translation

2.1 Speech Recognition

Speech Recognition is the process of conversion of speech into readable text by a machine or a program. This program ability to identify words in an audio signal and recognize them to produce the output in a written format is also called Automatic Speech Recognition (ASR) or simply speech-to-text (STT).

2.2 Working of Speech Recognition

Speech Recognition begins with transformation of audio - a physical sound, which is an analog signal into an electrical signal with the use of microphone. This signal yields digital data through an analog-to-digital converter. There are several models, methods and algorithms which can transcribe audio, once it is digitized, into text.

2.2.1 Conventional ASR Models

- Hidden Markov models
- Dynamic time warping (DTW)-based speech recognition
- Neural networks

The above models can be generalized simply as Conventional ASR and End-to-end ASR, on the basis of difference of decoding. Most Conventional ASR models consist of acoustic, pronunciation or lexicon and language model components, each of which is trained separately. This training involves curating pronunciation lexicon and defining phoneme sets for any specific language which not only requires expert knowledge but also is a time-exhaustive approach.

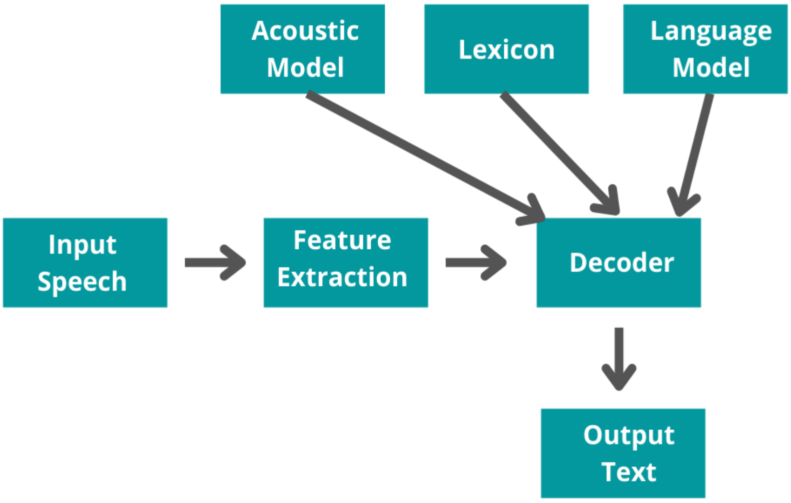


Figure 2.1: Conventional ASR Pipeline Figure

2.2.2 End-to-end ASR Models

On the other hand, End-to-end ASR model directly maps a sequence of input acoustic features into a sequence of grapheme or words. This model is trained to improve criteria related to the final performance evaluation metric of interest, which is Word Error Rate (WER). It is clear that end-to-end speech recognition significantly simplifies the complexity of traditional speech recognition models, by avoiding the necessity to manually label information as it can automatically learn language and pronunciation information, on its own. There are two key structures for end-to-end speech recognition: Attention Model and CTC.

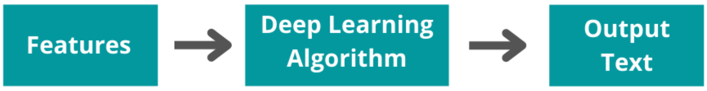


Figure 2.2: End-to-end ASR Pipeline Figure

However, there is no need for worry as a programmer or a coder due to numerous speech recognition services being offered for use easily through APIs and libraries.

Moreover, to prevent the speech recognizing system from wasting time in analyzing useless portions of audio which do not contain any speech, Voice activity detectors (VADs) are used.

2.3 Approaches and Types of ASR

There are three main approaches when it comes to Automatic Speech Recognition. These are listed below:

- **Synchronous Recognition (REST and gRPC)** sends audio data to the Speech-to-Text API, accomplishes recognition on that data and returns results after the audio has been processed completely. The duration of synchronous recognition requests for audio data is limited to 1 minute or less.
- **Asynchronous Recognition (REST and gRPC)** sends audio data to the Speech-to-Text API and initiates a Long Running Operation. Recognition results are periodically polled by means of this operation. The duration of asynchronous recognition requests for audio data is up to 480 minutes.
- **Streaming Recognition (gRPC only)** achieves recognition on audio data specified within a gRPC bi-directional stream. Streaming recognition requests are designed for real-time practices, such as capturing live audio from a microphone. Streaming recognition delivers interim results while audio is being captured, providing the result which starts to appear whilst a user is still speaking.

The umbrella of voice-activated technology is huge and contains a variety of different kinds of systems that fall under it. One common type of recognition systems is listed below:

Speaker dependent systems: For this type of system, the users may be required to assist the training of the system by reading sequences of words, phrases and sentences to it that is, the system is dependent on a speaker to be functional.

Speaker independent systems: This type of recognition system does not need any training before its use as it is already capable of comprehending most user’s speeches as per definition given by [18].

2.4 Performance Evaluation of ASR

2.4.1 Accuracy and its Improvement

Accuracy: The STT reliability is mainly founded on its correctness that is, Accuracy rate—an average number of errors in the text obtained from recognized speech. It is calculated as the percentage of errors for every 100 words and is called as Word Error Rate (WER). Technically, accuracy is the exact inverse of WER; the higher the WER, the lesser is the Accuracy of the System.

The WER according to a May 2020 benchmark was found to be 81.01%, 83.12% and 84.46% for Microsoft, AWS and Google respectively. Like Rev, devoted provider Temi also placed better reliability at 13.9% WER or 86.1% accuracy. In other words, there is an expected of 15 to 25 errors for every 100 words transcribed through the most prominent speech-to-text engines available today as given in [19].

Though, ASR captions are helpful only when WER < 30% according to study [20].

The simple mathematical formula for WER is given as:

$$\textit{Word Error Rate} = \frac{(\boldsymbol{Substitutions + Insertions + Deletions})}{\boldsymbol{Number\ of\ total\ words\ spoken}}$$

Where, a substitution is when a word gets replaced (“noose” may be transcribed as “snooze”), an insertion is when an unsaid word gets added(“SAT” may become “essay tea”) and deletion is when a word is missed completely (“move it around” becomes “move around”).

Accuracy Improvement: There are two ways to improve accuracy rates for automatic transcriptions – training the AI engine and reducing interference. You can train the AI to accurately interpret the specific accent, inflexion, and voice modulation commonly used by your agents by feeding the engine pre-recorded audio files. You could also work on reducing interference in the calling vicinity, by using superior quality microphones, keeping ambient noise levels to a minimum, and eliminating sudden interruptions. Finally, you could specially train the AI to accurately transcribe industry-specific terminology that could be commonly used by your agents, but may not be so commonplace enough for the engine to pick up correctly the very first time around.

2.4.2 Latency and its Improvement

Latency: Although the STT reliability is certainly indicated most appropriately through WER, one more factor also plays an important role; that is— Latency. Transcription latency is defined as the number of seconds taken by the speech-to-text engine in conversion of raw audio into readable written text.

There is a trade-off between latency and computation cost. Hence, it depends on the system requirements to opt for a solution which takes a longer latency period but guarantees better accuracy or a one which has a shorter latency period but also lower accuracy.

Latency Improvement: One of the main techniques that stand out when it comes to the improvement of latency is Prefetching. Latency becomes even more crucial as a performance metric for streaming speech recognition systems. Prefetching is the technique where responses are fetched sooner based on the partial recognition results. Particularly, prefetching is done by utilizing partial recognition results for subsequent processing for instance, receiving assistant server responses or second-pass rescoring before the recognition outcome is finalized. If there is a match between partial and the final recognition results, the earlier fetched response is delivered immediately to the user. This technique successfully speeds up the recognition process by avoiding the execution latency that typically occurs when recognition is completed. Even for a single query, Prefetching can be triggered multiple times but doing so would result in multiple rounds of downstream processing and greater computation cost. Hence, it is desirable to fetch the result earlier but meanwhile restricting the number of pre-fetches [21].

2.5 Translation and Real Time Translation

2.5.1 Translation

Translation is the process of conversion of meaning of linguistic discourse, that is— words either in speech or in text form, from one language to another. This activity transfers the linguistic entities of one language into their equivalents of another language. Translation is an act through which the content of a text is transferred from the source language in to the target language. [22] The language which is being translated from is called the source language (SL) and the language which is being translated into is called the target language (TL).

2.5.2 Real Time Translation

As the name suggests, Real time translation (RTT) technology is fundamentally a tech-driven solution that directly perform translation of content from one language to another.

2.6 Need for Translation:

As internet grew, it connected all parts of the world having around 7000 languages in use as per today. In today’s globally connected era, the need to comprehend one another has become possibly more important than ever. Half of the internet's content is in English but only 20% of the global population has any English skills whatsoever [23]. Such low percentage shows how important translation becomes. In 2016, Google estimated that it translates over 100 billion words a day and has over 500 million users [24]. This figure only continued to increase as The Google Translate app now has more than 1 billion active monthly users of which 95% are from outside the U.S. and more than 140 billion words are translated daily [25]. Meanwhile, more than half of the 2.5 billion people on Facebook post in a language other than English. Facebook’s AI on the social network itself along with on Messenger and Instagram indicates more than 6 billion translations a day [26].

Moreover, Translation is of critical importance in high stakes political, legal, financial and health-related exchanges. However, the performance of AI-fused machine translation methods can’t possibly match or substitute the costly, skillful human transcribers, though they may also count on translating engines from time to time too. This dependence results into big business; the cost of the business-to-business segment of translation is estimated to be a $23 billion annual market, known by a report in [27].

2.7 Background of RTT

The doors of opportunity first opened up for Real Time Translation when online language translation, Yahoo! Babel Fish, came to its ending point in 1997. Undoubtedly, translation of language has made continued progress ever since and has reached to the existing world where Artificial Intelligence has added to its significance making RTT more important and efficient than ever— so much that users can access and use this technology by themselves anywhere and anytime.

Conventional translation technology first listens to words being spoken, converts it into text, and then translates it into the destination language. This process observed change with the advent of Artificial Intelligence (AI) and a big revolution occurred with the usage of Deep Neural Network (DNN) in 2016. The new translation technology enabled translation engines to comprehend the meaning of a complete sentence, enhancing fluency unlike the conventional engines which were restricted to breaking up sentences into chunks resulting in destruction of intent and meaning from them. This is an important breakthrough as many obstacles are found during the speech of a person including accent, environmental noise, uttering speed and audible disturbances such as “uh” and “um”. Moreover, detection of sarcasm and cultural expressions proves to be a big problem. Hence, translation is incomplete if the semantic meaning or the conveyed message is not being upheld as a result of the operation.

“Translation is typically a literal interpretation of what’s there as opposed to the meaning and the context” [28].

2.8 Working of RTT

To perform Real Time Translation, a translation system listens to the audio signal, and immediately attempts to identify the language of the speech as well as the content which is being said. Waveforms of sound are analyzed in order to distinguish parts of the speech that seem to correspond to translations as it builds. The translating system then attempts to perform translation of its heard speech of source language into the normal speech or transcription of the destination language.

This is attained with the help of a combination of several machine intelligence tools and technologies; sounds are recognized and identified by advanced patter matching software, “long-term dependencies” and predictions of words being uttered are identified by neural networks and deep learning, all the information is processed by encoders, databases contain common words, meanings and information obtained from the learnings of previous analyses of millions of documents. Typically, these translation systems depend on cloud-based analysis for their performance resulting in a short lag between utterance and translation. This latency is expected to decrease with the networks and AI becoming faster.

This complex interaction of technologies to produce translated outcome is already capable of generating around 85% accuracy with latency of two to five seconds. With the continued growth of networks and AI, both fluency and speed may mark improvement.

To understand the need for RTT due to being different from conventional translating engines becomes clear through a response by a Google spokesperson, stating that a part of the Google Translating voice application was observed to be incompatible for listening and translation of longer time period tasks such as discussion at a classroom lecture or a video of a lecture, a story from a grandparent, a conference, etc. This presents the need for real-time translation technology in translating during live occasions which is expected to be both quick and efficient.

2.9 Future of RTT:

Technological advances are taking place at a rapid pace in the field of real time translation. Google plans on including the newest transcription feature on Android devices and also aims to bring it to the platform of iOS. The feature will be a “transcribe” option in the app once the user updates the device which will be stopped or restarted by tapping the mic icon. The fresh feature hit the market very soon and be available to all users in the world. The supported languages initially will be French, English, German, Russian, Hindi, Spanish, Thai, and Portuguese; meaning that the user will be able to translate any one of those languages spoken audibly and into any preferred language.

With the dawn of the internet and technology, translation has been enhanced to a higher level of ease and flexibility than the era of just the human translation. At present, Google Translate and Bing lead the setting of online translation, encompassing translation tools with the ability to work with millions of global languages. The time is near when the real-time language translation will be available on all channels and across numerous mediums, including social media communications as well as real-life discussions.

CHAPTER 3

Face Tracking and Subtitles Display in Real Time

3.1 Face Detection and Tracking

3.1.1 Face Detection

Face detection is the process of extracting and identifying human faces from digital images using Artificial Intelligence. With all the advancements in technology, Face Detection can also be done in real time videos.

3.1.2 Working of Face Detection

Machine learning algorithms are used to extract human faces from a larger image usually containing non-face objects such as different objects, buildings, and various other body parts. Facial detection algorithms usually begin by seeking out human eyes, which are one of the easiest facial features to detect. Next, the algorithm might try to find the mouth, nose, eyebrows, and iris. After identifying these facial features, and the algorithm concludes that it has extracted a face, it then goes through additional tests to confirm that it is, indeed, a face [29].

3.1.3 Face Tracking

Face Tracking Technology detects and tracks the presence of a human face in a digital video frame [30].

3.1.4 Working of Face Tracking

Artificial Intelligence algorithms are used detect faces in a video stream after which the Face Tracking software follows that face around within that video stream in real time.

3.1.5 Firebase ML Kit

ML Kit is a mobile SDK that brings Google's machine learning expertise to Android and iOS apps in a powerful yet easy-to-use package and enables app developers to have advanced machine learning capabilities into their with just a few lines of code.

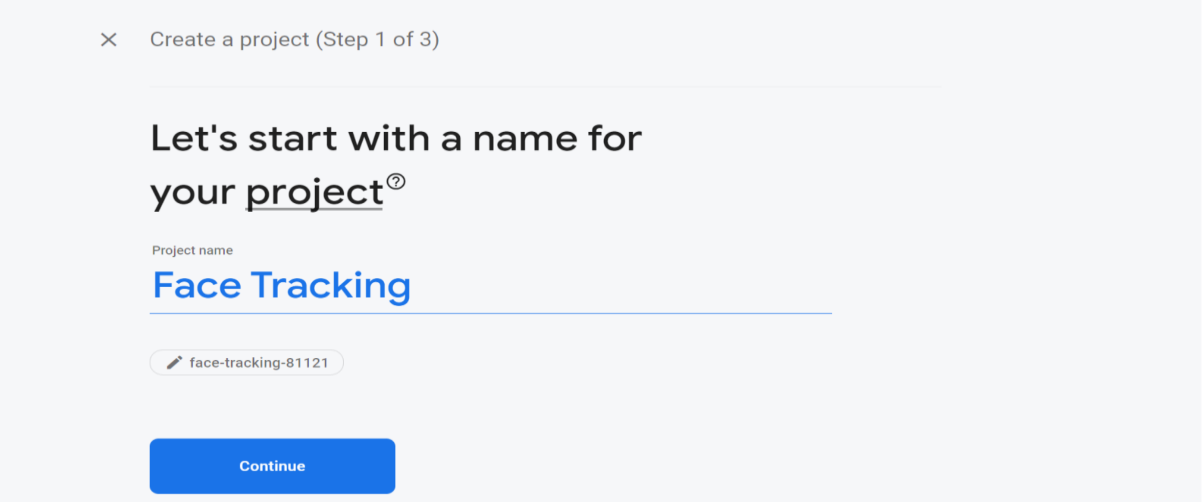
3.1.6 Firebase Face Detection

ML Kit's face detection API allows face detecting, its facial features and contours. Since ML Kit can perform face detection in real time, it can be used in many applications your requiring real-time accessibility.

3.1.7 Setting Android Project with Firebase ML Kit

1. Creating a new project on Fire Base with Google account

.



Create a project (Step 1 of 3)

Let's start with a name for your project

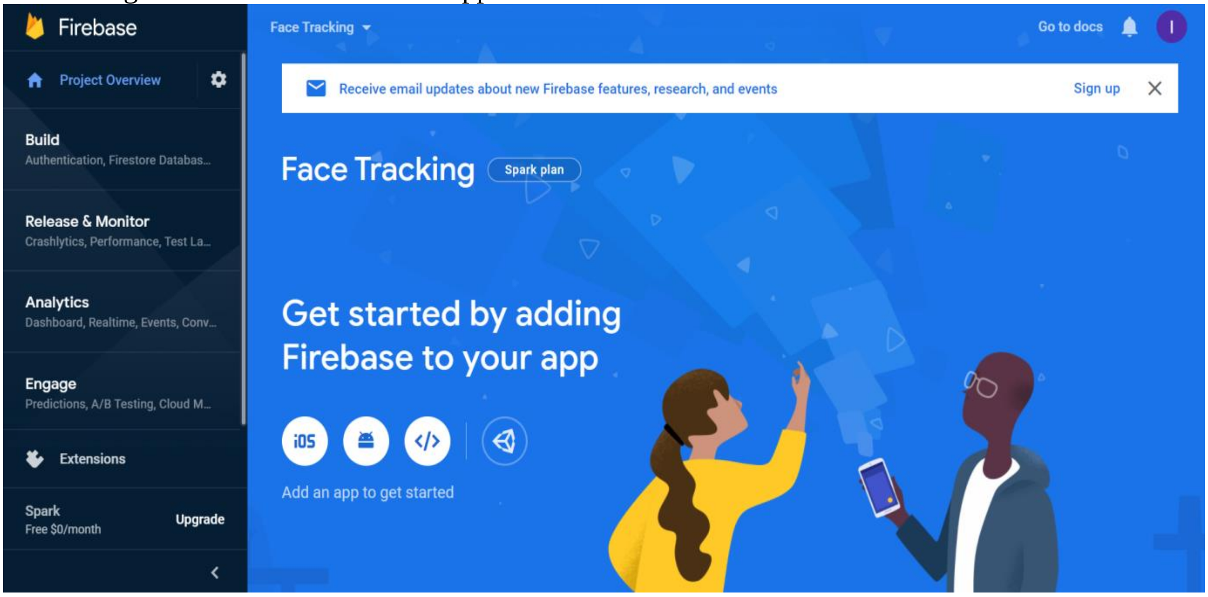
Project name

Face Tracking

face-tracking-81121

Figure 3.1

2. Selecting Android Icon for android application



Receive email updates about new Firebase features, research, and events

Face Tracking

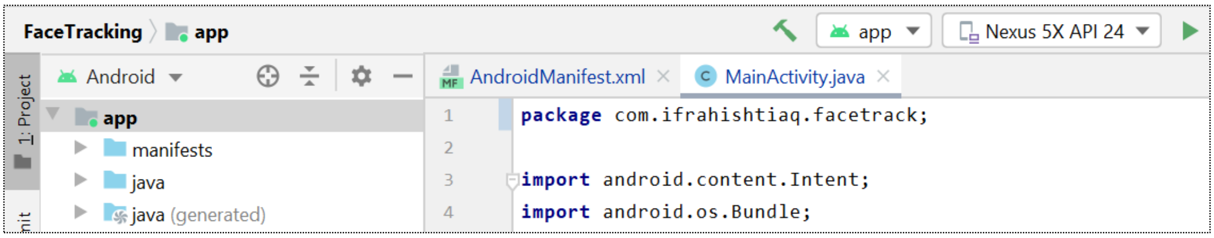
Get started by adding Firebase to your app

Add an app to get started

Firebase
Project Overview
Build
Authentication, Firestore Databas...
Release & Monitor
Crashlytics, Performance, Test La...
Analytics
Dashboard, Realtime, Events, Conv...
Engage
Predictions, A/B Testing, Cloud M...
Extensions
Spark
Free $0/month
Upgrade
Face Tracking
Go to docs
Sign up
Spark plan
iOS

Figure 3.2

3. Copying the package name from our Android Project

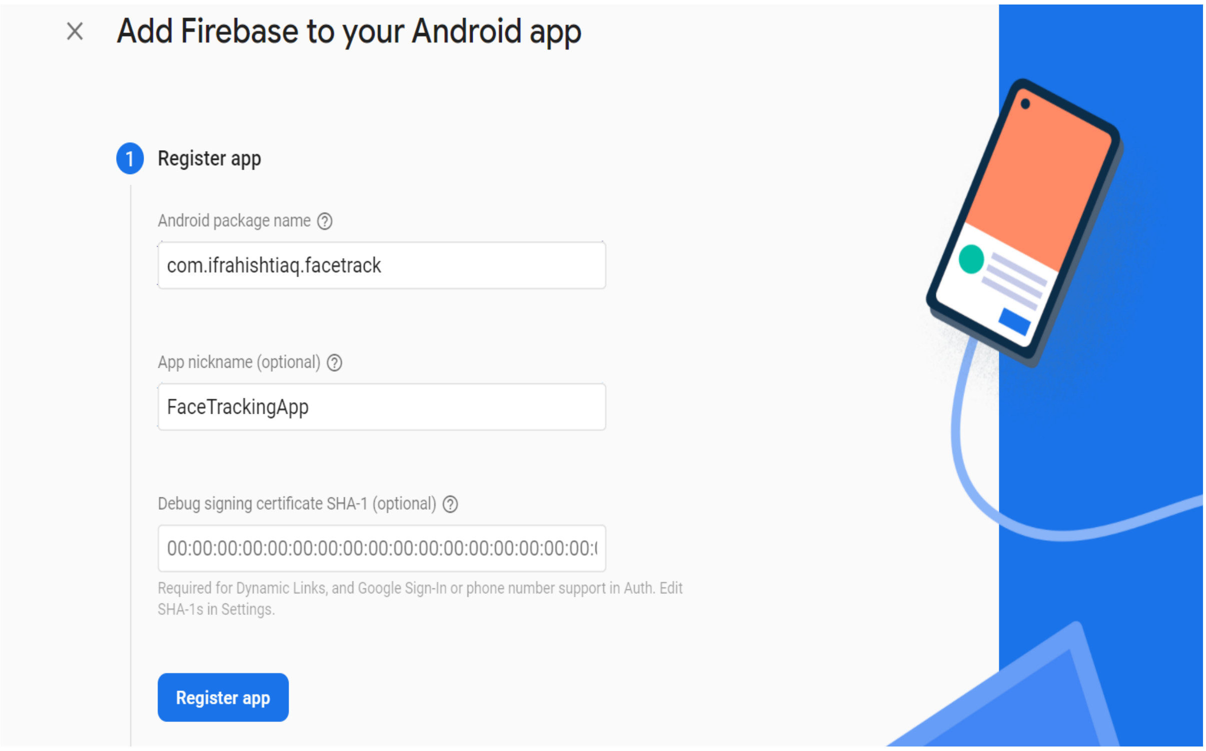


```
package com.ifrahishtiaq.facetrack;

import android.content.Intent;
import android.os.Bundle;
```

Figure 3.3

4. Pasting the copied package name below



Add Firebase to your Android app

1 Register app

Android package name
com.ifrahishtiaq.facetrack

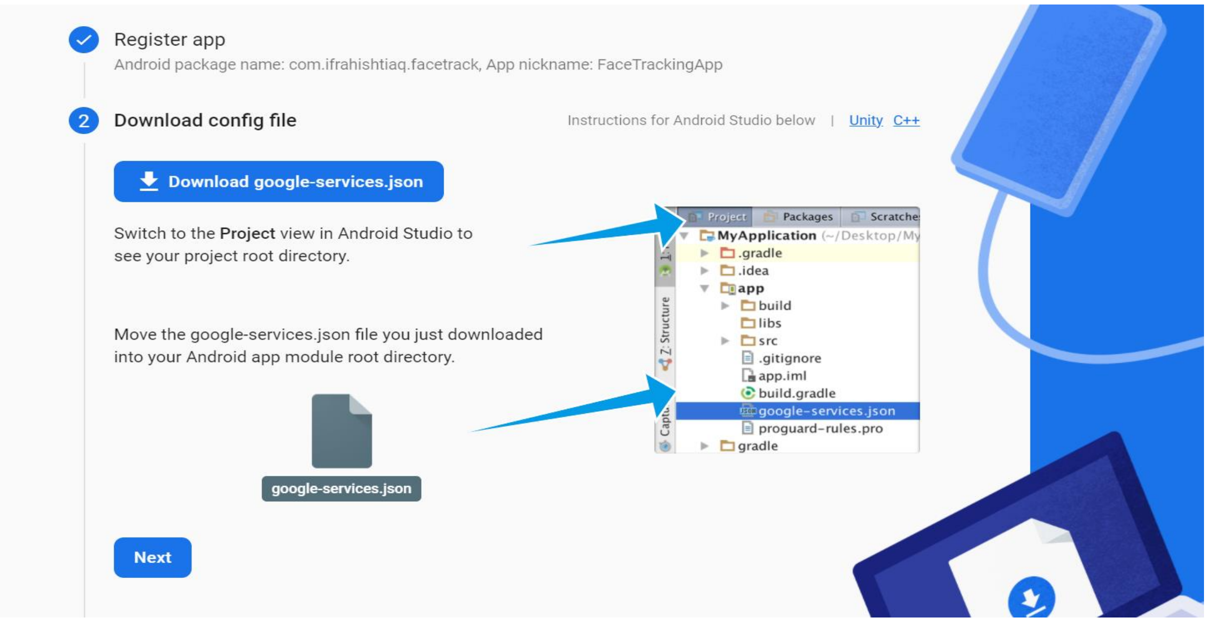
App nickname (optional)
FaceTrackingApp

Debug signing certificate SHA-1 (optional)
00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:0

Required for Dynamic Links, and Google Sign-In or phone number support in Auth. Edit SHA-1s in Settings.

Figure 3.4

5. Downloading the config. file



Register app

Android package name: com.ifrahishtiaq.facetrack, App nickname: FaceTrackingApp

2 Download config file

Instructions for Android Studio below | Unity C++

Download google-services.json

Switch to the **Project** view in Android Studio to see your project root directory.

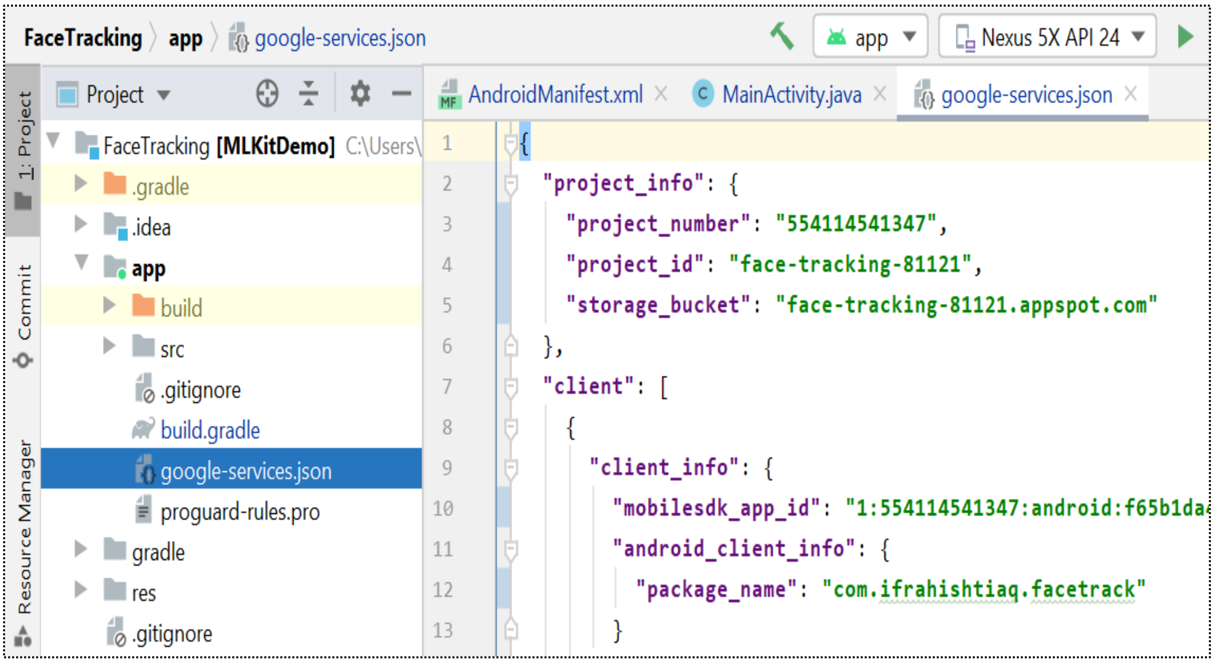
Move the google-services.json file you just downloaded into your Android app module root directory.

google-services.json

Next

Figure 3.5

6. Pasting the downloaded file in the app folder



```
{
  "project_info": {
    "project_number": "554114541347",
    "project_id": "face-tracking-81121",
    "storage_bucket": "face-tracking-81121.appspot.com"
  },
  "client": [
    {
      "client_info": {
        "mobilesdk_app_id": "1:554114541347:android:f65b1da
        "android_client_info": {
          "package_name": "com.ifrahishtiaq.facetrack"
        }
```

Figure 3.6

3.2 Subtitles Display in Real Time

Research and practical work on generating Live Subtitles is being carried out from a long time but still there is a lot of room for improvement, accuracy and innovation in this domain. Speech recognition technique is being used for live subtitling since 2005, or may be before that. Live subtitling technology is still not very popular world-wide as it is hard to develop. However, Countries like U.S and U.K are using this technology efficiently from a long time.

Live subtitles are introduced as a ceaseless progression of text as individuals talk. Live subtitles usually are not very accurate as there is not enough time for correction between speech recognition and its conversion into subtitles. Live subtitles will ordinarily slack the sound by a few seconds because of the inborn postponement in translating, encoding, and displaying the captions. Live captions also has some typographic blunders or mishearing of the verbally expressed words, with no time available for correction before transmission.

Since, live subtitles are not very accurate there is still a lot of improvement needed in this domain. Some technologies can be developed that can understand different accents and provide efficient hearing to reduce typographic blunders. The lag between the speech and display of subtitles is still a big challenge that open doors to a wide research domain which can devise some efficient solution for this. Besides these challenges live subtitling is still a very useful technique which can make the life of common people easier but still it is not used widely. Through our App Live Subtitles we want to take this technology to the common people, so that they can use it in an easy and convenient way.

3.2.1 Fundamentals to Live Subtitles

Subtitling is the need of this modern era. There’s a common saying that more screens means more subtitles. That is said because now everyone tries to increase the reach of their content and to make something understandable internationally language barrier should be removed and this job is done efficiently through subtitling.

3.2.1.1 Open Captions

Open captions are part of the video and cannot be turned off and on [1]. Means these captions/subtitles will be displayed throughout the video as they are recorded as part of the video. Since the main objective of live subtitles is to make communication easier for people with hearing impairment and reduce the language barrier so open captions will fulfill these requirements.

3.2.1.2 Subtitles in Same language

Usually referred to as SLS(Same Language Subtitles). By same language we mean that the subtitles for a video would be generated in the language used in the video. This type of subtitles are useful for people with hearing impairments as they don't have to be dependent on the sign language translation and can easily read what someone is saying. This also keeps them connected to people who doesn't know sign language. However it cannot be said that they are only useful for people with hearing impairment, because many times due to the difference of accent and pronunciation people are unable to understand what somebody has said although they can hear them perfectly.

3.2.1.3 Translated Subtitles

Translated subtitles mean that the subtitles would be generated in the language selected by the user, regardless of the language used in the video. This type of subtitling is very useful for increasing the reach of content and to reduce the language barrier. In the past, videos that were meant to be reached by people internationally had to be dubbed into different languages in order to make them understandable internationally. Dubbing is a time taking process as first it requires content translation from a proficient speaker of both the languages (The language used in the video and the language in which it is to be translated), then it has to be dubbed in that language again by some proficient speakers of that language. So this process needs a lot of resources and time. On the other hand, subtitling only needs translation of the content and it's placement in the video. Nowadays, such artificially intelligent softwares are available which only needs the content and they place it in the video by themselves and some software can even provide automatic translation of that content.

3.2.1.4 Subtitling Constraints

Subtitling constraints mean the limitations in the process of subtitling. Translated subtitles are not usually the exact translation of the content spoken in the video. Translated subtitles usually follow word to word translation technique. There can be some phrases, narrations, proverbs or poetry spoken in the video which when translated for subtitling doesn’t provides the exact meaning of it, however it hints at the overall meaning of it. This usually happens because nowadays people mostly make use of language translator softwares which employs word to word language translation.

3.2.2 Approaches

In this section we will discuss some approaches that are used for speech to text conversion and then putting that text as subtitles on the video.

3.2.3.1 Using Android Studio Only

Doing everything in android studio using Java alone and not using any external module or software.

Result: Couldn't find any function in Java to extract audio from time to time to convert it to text and put it on the screen as subtitles.

3.2.3.2 Using Python with Java

Making video using Java in android studio and passing that video to Python function as a parameter to do the further processing. This is done by using Chaquopy which is a platform to execute Python scripts in android studio.

Result: Couldn't find any function in Java to pass frames one by one to Python function for processing.

3.2.3..3 Using Python Only

When the button is clicked calling Python function to do everything from capturing video, extracting audio, converting speech to text to displaying that text in the form of subtitles.

Result: This approach build successfully but App crashes on run time.

3.2.3.4 Using SRT files in Python

Write the recognized audio in an SRT (SubRip Text) file time to time and read that file to put subtitles on the video, using Python. Note that SRT files are subtitles files that contain Block no. , starting and ending time for the subtitle to be displayed on the screen and the subtitle to be displayed within that time.

Result: Reading and writing the SRT file simultaneously causes a lot of delay and runtime errors.

3.2.3.5 Using OpenCV in Python and Java

Make video in android studio using OpenCV module and pass that video as a parameter to a Python function to display subtitles using OpenCV. Note that we are distributing the tasks between Java and Python because there are some functions of OpenCV that are available in Python but not in Java.

Result: App keeps crashing when we start recording video.

3.2.3.6 Using Python and OpenCV in Java

Making video in android studio using OpenCV get the recognized audio from python function using speech recognition module and put the text returned from python function on the screen using OpenCV putText function in android studio.

Result: App keeps crashing when we start recording video.

3.2.3.7 Using OpenCV in Java

Making video in android studio using OpenCV module and putting a hardcoded text on it using OpenCV putText function.

Result: App working fine and displaying the overlayed text on the screen when we start recording video.

3.2.3.8 Using OpenCV in Python

Making video using OpenCV in Python and recognizing audio using speech recognition module and putting that recognized text on screen using OpenCV.

Result: Things working fine but slow.

3.2.3.9 Using Java Along with Some Other Tools

Making video in android studio using OpenCV module then using Speech Recognizer class of android studio to recognize speech from the audio of the real-time video. Using Firebase Language identifier to identify the source language of the video. Then converting the speech to text and displaying the text on the screen using the Speech Recognizer class. The translation is done using Firebase ML Kit. The recognized speech is translated if needed by the user and then displayed on the screen.

Result: This method make the app work in the proposed way and everything is done efficiently.

CHAPTER 4

Methodology

4.1 UML Diagrams

The UML (Unified Modeling Language) diagram is used to visually represent a system along with mentioning its main actors, roles, actions, classes with the purpose to understand about the design of the system.

4.1.1 Use Case Diagram

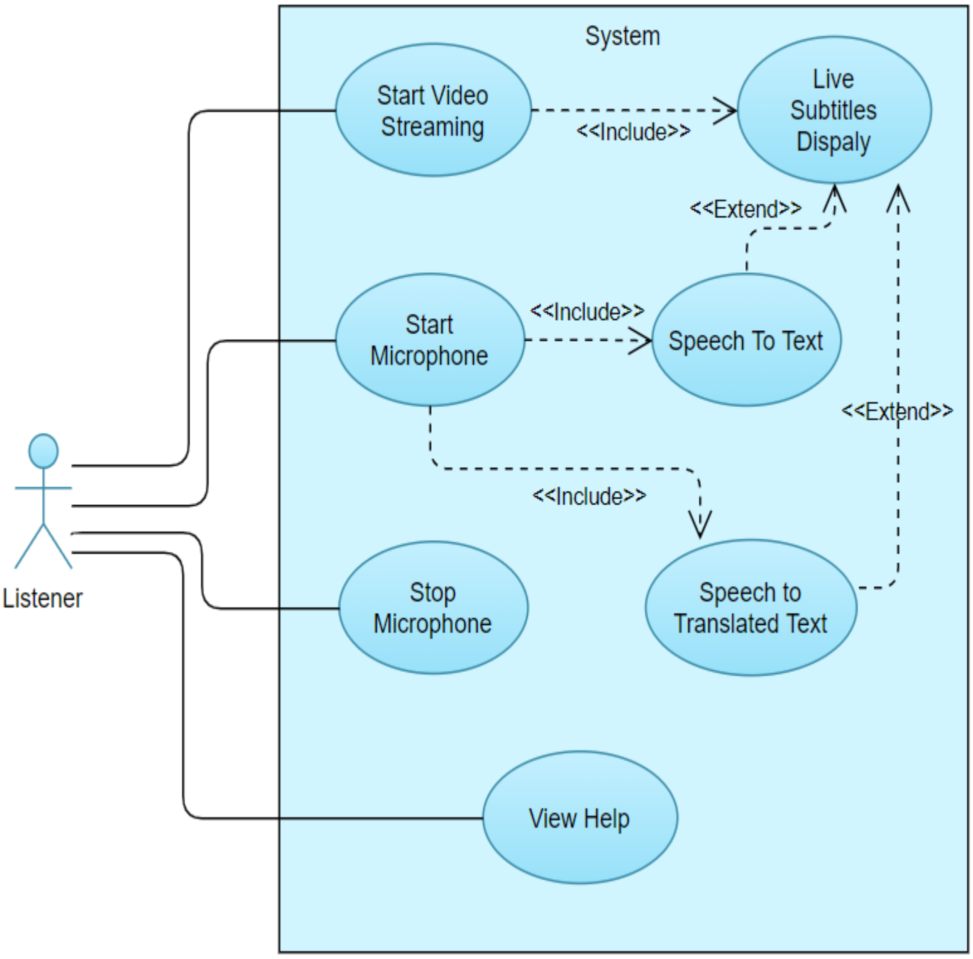


Figure 4.1 Use Case Diagram

4.1.2 Sequence Diagram

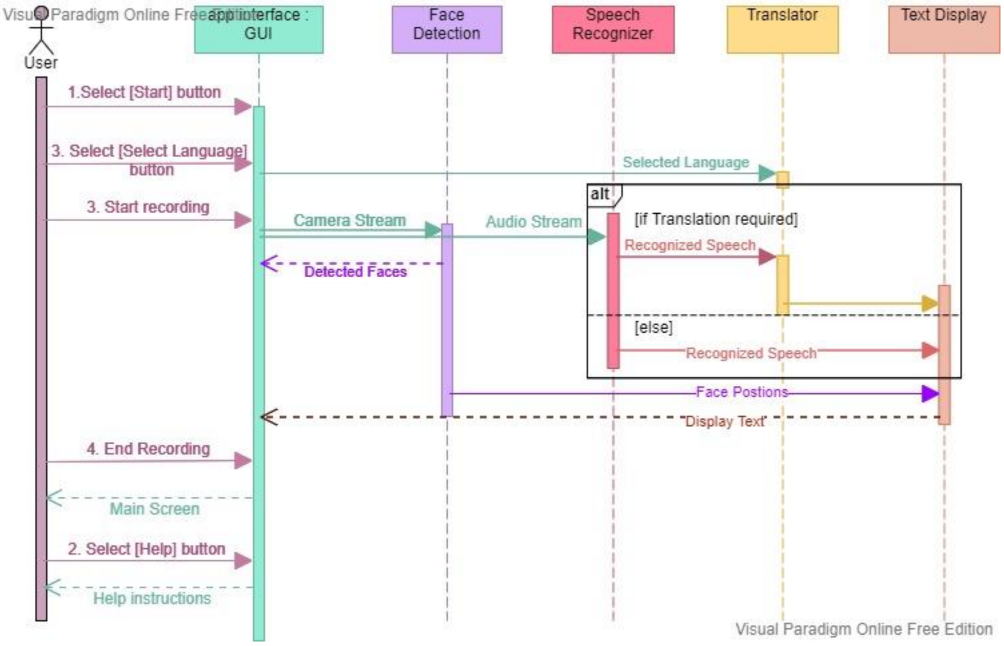


Figure 4.2 Sequence Diagram

4.1.3 State Transition Diagram

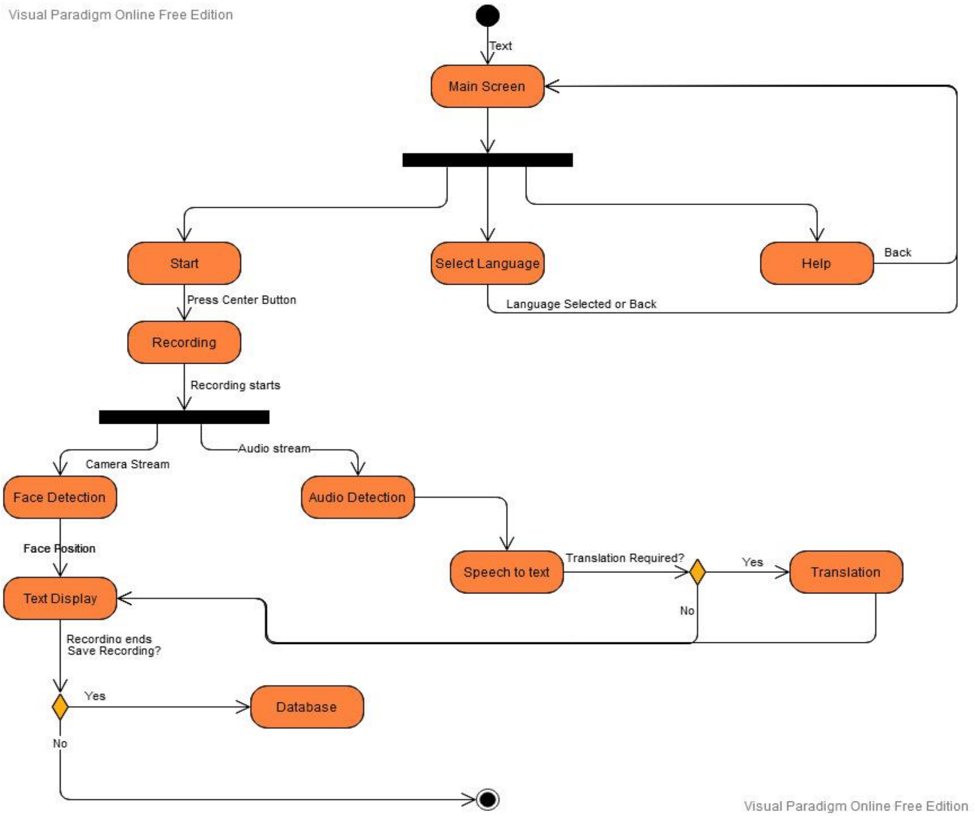


Figure 4.3 State Transition Diagram

4.2 Project Development

The main aim to develop the app Live Subtitles is to make the communication process more feasible. Our first goal is to make everyday conversation easy for people with hearing impairment by putting subtitles on real-time so that they can read what someone is saying and do not depend on hearing aids. Our second goal is to reduce the communication barrier by displaying translated subtitles in user selected language on real-time videos. In order to achieve these goals we have developed the following functions.

4.2.1 Making Real-time Videos

This app provides the ability to record real-time videos feasibly. This feature has been developed using OpenCV android sdk version 3.4.12. There is no fixed length for recording videos that means user can record videos of any length. The video can be made using front camera as well as rare camera as per user's choice.

4.2.2 Speech Recognition

This app records audio along with video and recognize speech in that audio. This feature is developed using the Speech Recognizer class provided in android studio. This speech recognizer class is popular for developing speech to text feature. This feature can tolerate ample noise but the speech recognition would be affected in case of severe noise. The clarity of speech also plays an important role in speech recognition.

4.2.3 Speech to Text

This app converts the recognized speech to text. This is also done by using speech recognition class. This text is further processed to achieve the remaining goals.

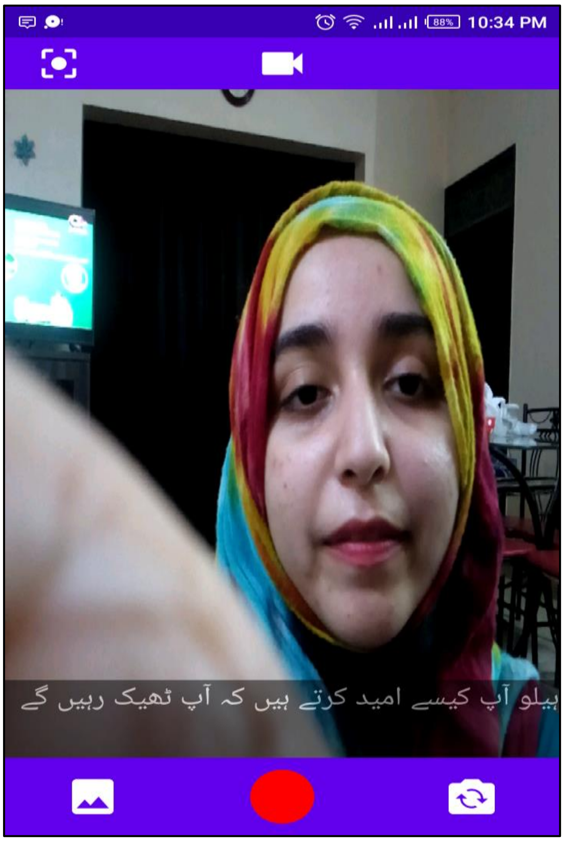
4.2.4 Text Translation

This app also provides the service to translate the speech spoken in the real-time video. Firebase ML Kit is used to translate the text recognized from the speech. First the source language of the real-time video is detected that is the language spoken by the speakers in the video using firebase language identifier. Then the recognized text is translated into target language that is the language selected by the user. This feature will be used when the source language and target language are different.

4.2.5 Subtitles Display

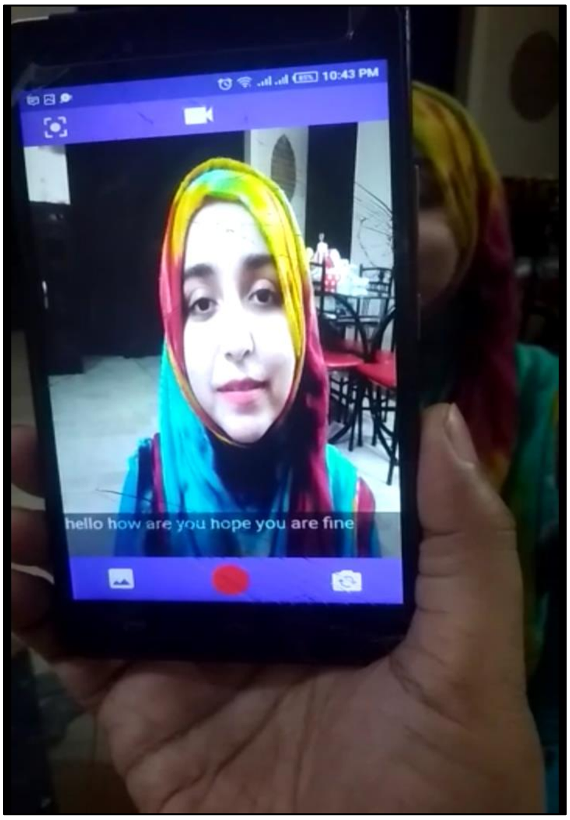
The app displays subtitles of the recognized speech from the real-time video that is converted into text and then displayed as subtitles. The subtitles can either be in the same language or in the translated language. It depends on the user.

These are the achievements that we have achieved so far. All the features are integrated in the app Live Subtitles which were proposed. Below are some screenshots taken from the video with subtitles:



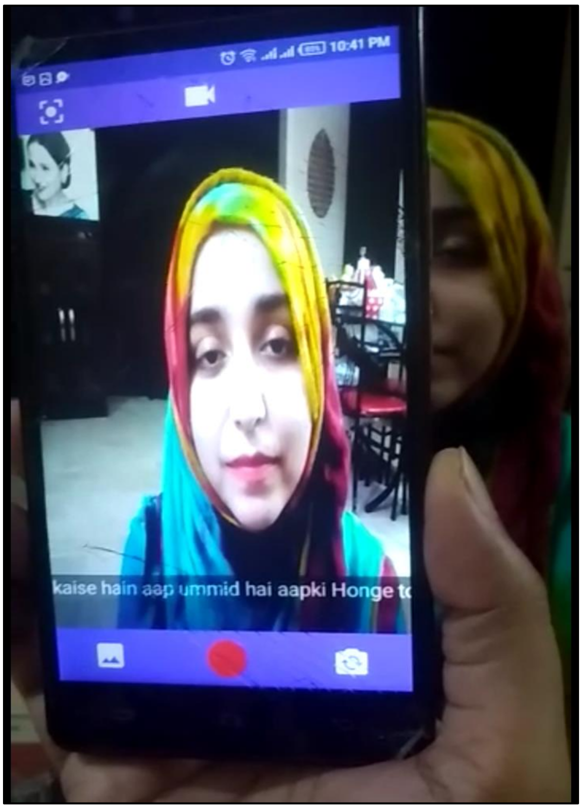
ہیلو آپ کیسے امید کرتے ہیں کہ آپ ٹھیک رہیں گے

Figure 4.4 With translated subtitles in Urdu



hello how are you hope you are fine

Figure 4.5 With Subtitles in English



kaise hain aap ummid hai aapki Honge tc

Figure 4.6 With translated Subtitles in English

4.2.6 Minimum Requirements

- Android version 4.1 Jelly Bean API 16 or above
- 88 MB Storage Space or more
- Internet
- Microphone
- Front Camera
- Rare Camera

4.3 App Activities

The front-end of this app is built on Android Studio 4.1.2. Android Studio is one of the most popular platform used to build android applications, as it is open-source and free to use. We kept the front-end of our application simple in order to make it user friendly. We have used the Android version 4.1 Jelly Bean API 16 so that our app can run on approximately 99.8% of the devices. Since, everyone doesn’t have the latest android version because some people don’t find it useful to upgrade their android version quickly or some people don’t know much about android versions. So, building an app using the latest android version can reduce its reachability.

An Android activity is one screen of the Android app's user interface. In that way an Android activity is very similar to windows in a desktop application. An Android app may contain one or more activities, meaning one or more screens. Every activity also have an XML file which can be used to design the activity and a Java file which is used to define the functionality. The Android app starts by showing the main activity, and from there the app may make it possible to open additional activities [2]. Every activity in an android app goes through some states during its life in the app this process is called Activity Life Cycle and it is illustrated in Fig. 4.1

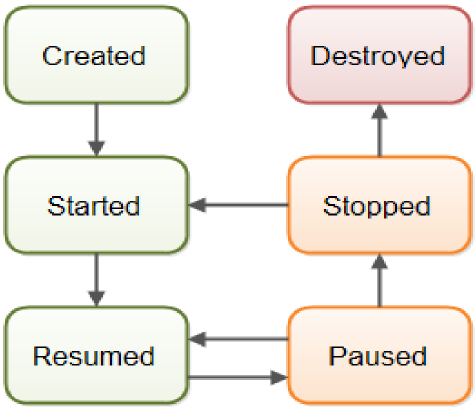


Figure 4.7 Android Activity Life Cycle

Our Android app is consists of 5 activities namely:

- Main Activity
- Camera Activity
- Gallery Activity
- Settings Activity
- Help Activity

4.3.1 Main Activity

Main Activity is the one which will appear right after when user opens the application. Its function is to provide a link to other activities.

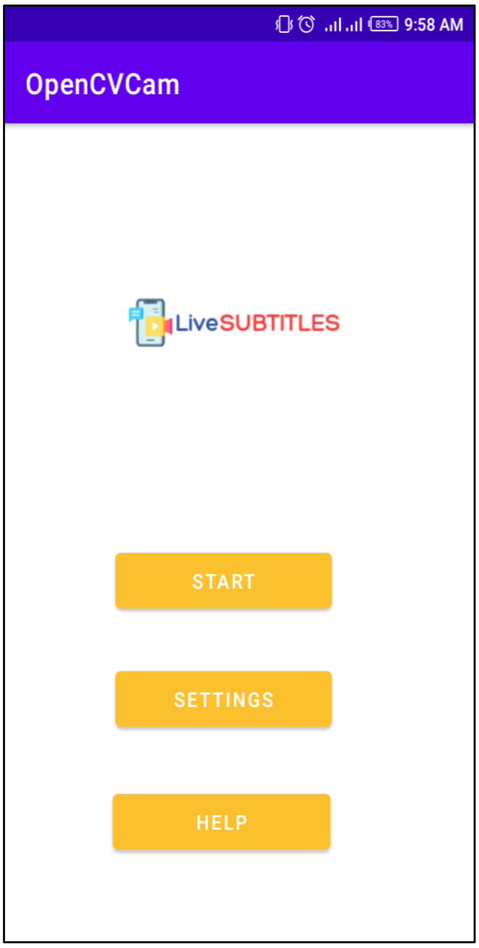


Fig. 4.8 Main Activity

- **Front-End** In Fig. 4.2 we can see the main activity. It has three buttons which can take the user to other activities. This activity is designed using a Relative layout within a Frame layout which has three Buttons and one ImageView for displaying the app logo. These things can either be coded or can be designed by using drag and drop functionality of Android Studio.
- **Back-End**

It has a simple functionality to open the camera, settings or help activity when their respective button is clicked. Each of these buttons is first defined using findViewById function and then has an onClick method to take user to that specific activity every time when the button is clicked.

4.3.2 Camera Activity

When the Start button is clicked Camera Activity is launched. It provides many functions and the basic functionality offered by the app is provided in this activity.

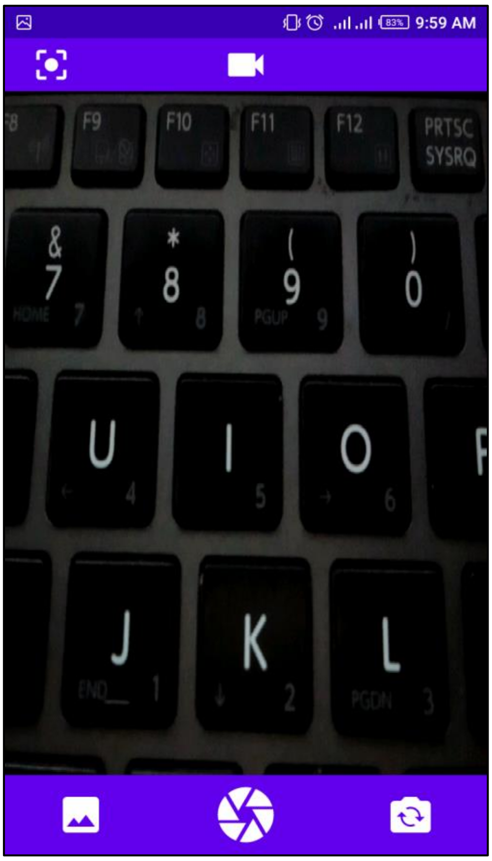


Fig. 4.9 Camera Activity

- **Front-End**

We can see in Fig. 3.3 that there are some icons on the top and bottom bar of this activity, these icons provide different functionalities. Camera activity is designed using Frame layout which is consist of a JavaCameraView and two linear layouts one for top bar and one for bottom bar. Top linear layout contains two ImageView’s one for video camera button and the other for change image resolution button. Bottom linear layout contains three ImageView’s one to view gallery, one for camera button and one for flip camera button. The media is played in JavaCameraView.

- **Back-End**

The camera function is implemented the OpenCV module android-sdk version 3.4.13 so first its status is checked whether it is connected or not. Then all required permissions are checked like permission to use camera, microphone, storage etc. Then the flip camera button is defined and its method is implemented to swap camera between back and front camera whenever this button is pressed. Method for gallery button is implemented to open gallery activity. By default Picture mode is set that means when the camera activity opens picture mode is enabled so user can take pictures from either back or front camera. This is an additional feature provided in the app. Video mode is enabled when the user will click on the Video camera button. In video mode the camera button will be replaced by a circle button.

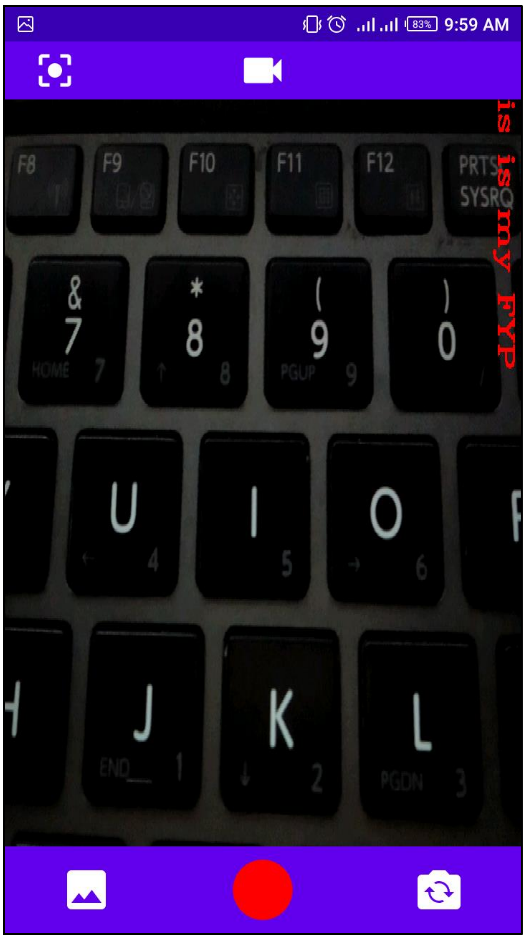


Fig. 4.10 Capturing Video

As it can be seen in Fig. 4.4 that when the user start capturing video the circle turns red and when the user stops capturing video the circle button turns white. All this is implemented in the onTouch method of video camera button. Functionalities to save either picture or video in user’s external storage are implemented in the take picture button onTouch method.

Then the following methods are implemented:

- onResume( )
- onPause( )
- onDestroy( )
- onCameraViewStarted( )
- onCameraViewStopped( )

The speech recognizer class is also used in this Java file to recognize speech, convert it into text and display that text as subtitles over the video. The text that can be seen in Fig. 3.4 on the screen is overlaid on the screen using the ImgProc.putText() function of OpenCV in the onCameraFrame method. Now we have changed the method and displaying subtitles using TextView. The text is achieved by recognizing speech and converting it into text and displayed on the textView using setText() function of the TextView. The translation of the subtitles is also done in this file using Firebase ML Kit. Firebase language identifier is used to detect the source language of the real-time video. The source language is then used to translate the text in target language that is the language selected by the user. The processed text is then displayed as subtitles on the real-time video. Some of the approaches to do speech to text conversion and displaying that text as subtitles on the video will be discussed in next section.

4.3.3 Gallery Activity

This activity will appear when user clicks the gallery icon placed on the bottom left corner of the camera activity

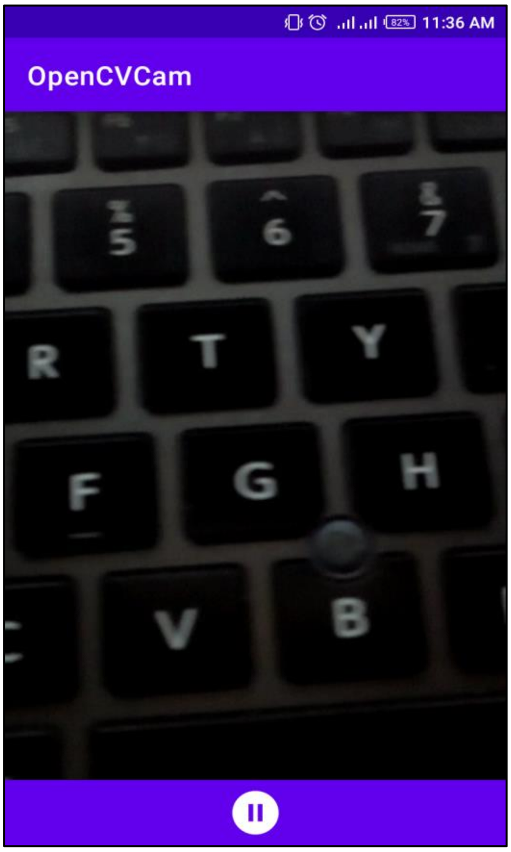


Fig. 4.11 Gallery Activity

- **Front-End**

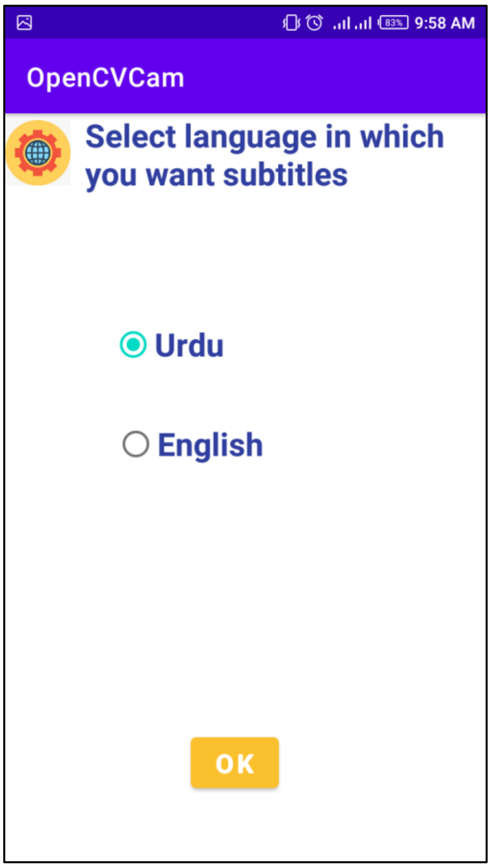
This is a simple activity which shows captured media in the order of oldest to latest. For video on the bottom it has a bar which has an icon in the center which keep changing into Play, Pause or Replay buttons according to the situation.

- **Back-End**

It fetches media files from the user’s external storage using the createFileArray method and allows user to swipe media file right and left with viewPager class.

4.3.4 Settings Activity

This activity will appear when user clicks the settings button on the main activity. This activity let the user to choose the language in which he wants to view the subtitles.



Select language in which you want subtitles

◉ Urdu

○ English

Fig. 4.12 Settings Activity

- **Front-End**

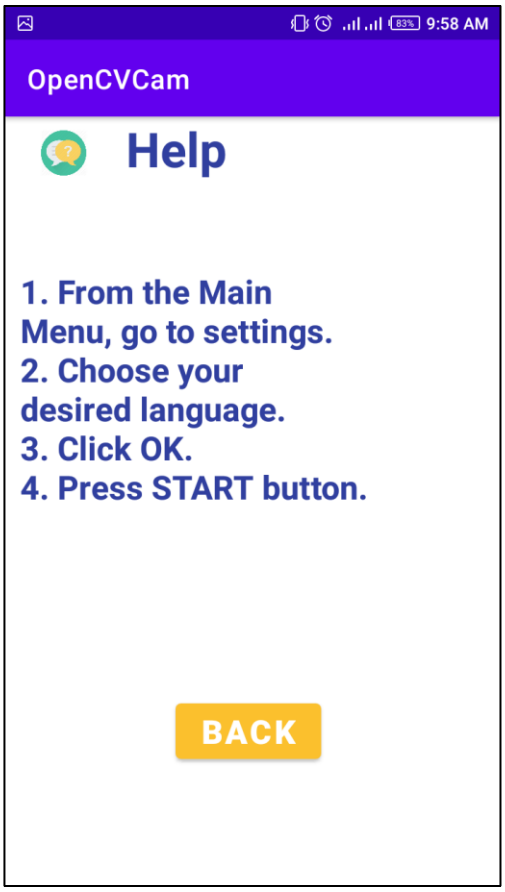
Activity in Fig. 3.6 is designed using a Relative layout. It has an ImageView to display the settings symbol, a TextView for description, a button and a RadioGroup. The RadioGroup has two RadioButton's one for English and one for Urdu, only one radio button can be selected at once.

- **Back-End**

It includes the method to pass the input from radio buttons to the camera activity's speech recognition class. It also includes the onClick method for Ok button to take the user back to the main activity every time when it is clicked.

4.3.5 Help Activity

This activity will appear when user clicks the help button on the main activity. This activity will provide user some guidelines on how to use this app.



Help

1. From the Main Menu, go to settings.
2. Choose your desired language.
3. Click OK.
4. Press START button.

Fig. 4.13 Help Activity

- **Front-End**

This activity is designed using Relative layout which is consist of an ImageView for displaying help symbol, two TextView's one for heading and one for guidelines and a button to take the user back to the main activity.

4.4 Working

When user opens the app, main screen is displayed with three buttons. User can read the guidelines on how to use the app by clicking on the Help button and can go back to the main screen by clicking on the Back button. User can select preferred language by clicking on the Settings button. User can check the radio button of the required language and then click on Ok button to save the settings.

When user clicks on the Start button camera screen is displayed. On this screen user can take picture by clicking on the middle button located on the bottom bar. User can record video by first clicking the Camcorder button on the middle of the top bar and then clicking on the Circle button on the bottom bar. The circle button will turn red when video recording is started and turns white again when the user stops recording.

When user starts recording the app will start recording audio too and keep converting the audio into text by first detecting the source language of the video using Firebase Language Identifier then comparing the source language to the language option chosen by the user in order to know whether there is need of translation or not. If no translation is needed that means the language chosen by user is same as of the source language, then the recognized audio will be converted into text using Speech Recognizer class of Android Studio. If there is need of translation then the recognized speech is first translated and then converted into text using the Firebase ML Kit. The converted text is then displayed on the screen.

User can watch the recorded videos by clicking on the Gallery button located at the bottom left of the screen. The videos are in the order of oldest to newest video.

4.5 App Deployment

While developing an Android app, you would usually run it on a physical device or an emulator. If you want to share it with someone for their feedback, you would share an APK that can easily be installed on any Android device.

4.5.1 Extracting APK file from Android Studio

1. In the Android menu, go to **Build > Build Bundle(s) / APK (s) > Build APK(s)**.
2. Android Studio will start building the APK for you. Once done, a pop-up on the bottom right will notify you of its completion. Click the ‘**locate**’ button in this dialog.
3. The ‘locate’ button should open File Explorer with the debug folder open that contains a file called “app-debug.apk”.
4. That’s it. Rename this file and share!

4.5.2 Deploying this file on Android Devices

The generated APK can only be used for testing. For deployment, you should generate a signed APK for which the process is a little more complicated.

- You need to enable the “**Allow unknown sources**” option in the settings of the device where you want to install the APK.

We deployed our App on a few different android devices;

Device No.	Device Name	Device Model	Device Version	Display Size	Status (Successful/Unsuccessful)
1.	Nokia 3	TA-1032	Android 9.0 (Pie)	5 inches	Successful
2.	Infinix HOT 4	Infinix X557	Android 7.0 (Nougat)	5.5 inches	Successful
3.	Huawei Y5 Prime	DRA-LX2	Android 8.1.0 (Oreo)	5.45 inches	Successful
4.	Lenovo K5 Play	-	Android 8.0 (Oreo)	5.7 inches	Successful

CHAPTER 5

Testing and Results

5.1 Application Testing

S.No	Test	Input	Result	Discussion
1	Passing the Preferred language Option through Radio buttons.	Check 1 radio button from two, either English or Urdu.	When we start taking video the subtitles shown is in the exactly same language.	We gave input through both radio buttons one by one to ensure they both are working fine.
2	Checking the flip camera option.	Click on the flip camera icon in the bottom right corner.	The flip camera icon is working fine.	We try to make videos using both front and rare camera to ensure the functioning of flip camera.
3	Checking picture taking feature.	Click on the shutter icon in the center bottom bar.	Picture taking feature is working fine.	When shutter is clicked it turns gray indicating that the picture has been taken
4	Checking video feature.	Click on the camcorder icon on the top bar to enter video mode then click on the circle icon to start recording.	The video feature is working properly.	First we enter the video mode and on clicking the circle icon it turns red indicating that the video is being recorded. When the circle button is clicked again the video stops recording and it turns white.
5	Checking the saved videos and photos.	Click the gallery icon in the bottom left corner.	All pictures and videos has been saved.	All the pictures and video can be view in the app gallery in the order of oldest to latest.

6	Checking English Subtitles.	First go to settings and set English as preferred language. Then start making video in English.	English subtitles are Displayed properly.	Since the preferred language is English and video is also in English, so subtitles are displayed without translation.
7	Checking Urdu Subtitles.	First go to settings and set Urdu as preferred language. Then start making video in Urdu.	Urdu subtitles are Displayed properly.	Since the preferred language is Urdu and video is also in Urdu, so subtitles are displayed without translation.
8	Checking Translated English Subtitles.	First go to settings and set English as preferred language. Then start making video in Urdu	English subtitles are Displayed but not properly translated as they are displaying Urdu in Roman.	Since the preferred language is English and video is in Urdu, so subtitles are displayed after translation.
9	Checking Translated Urdu Subtitles.	First go to settings and set Urdu as preferred language. Then start making video in English.	Urdu subtitles are Displayed properly.	Since the preferred language is Urdu and video is in English, so subtitles are displayed after translation.

5.2 Performance Testing

S.No.	**Performance Metric**	**Value**	**Comments**
1-	App Crashes	1-2%	App Crashes one time when it is used first time after installation if all the permission are not granted.
2-	API Latency	1-2 seconds response time	One API is used in this app for translation. So the average response time between some speech and its translation is almost 1 to 2 seconds.
3-	Session Length	Depends on user	Session length is the time

			between app open and close. The session length of this app depends solely on the user.
4-	Cost	Minimum 15k to 20k	This is the cost of mobile that is used for running the app. The app itself is free of cost.
5-	Stability	97 – 98 %	It means the percentage of sessions that are crash free.
6-	App Launch Time	1 – 2 seconds	App usually takes 1 to 2 seconds to launch even when there are already 9 to 10 app processes running on the device.
7-	User Interface Response Time	Less than 150mseconds	It usually takes less than 150ms to respond to user input.
8-	Battery Consumption	10 to 15% per hour on average.	May vary from device to device.
9-	CPU Usage	0 - 5mAh	The app is not using extensive CPU power.
10-	Memory Usage	0.5 MB memory per hour on average	Memory used by the app per hour.
11-	Storage	120 MB	It is the storage space used by the App. This may increase when data related to the app is increased.

CHAPTER 6

Future Enhancement and Conclusion

6.1 Future Enhancement

Although all the proposed features has been added into the app but there is always room for further improvement. Since, most probably this is the first android application to provide real-time subtitles with the option of translation in user selected language, there are a lot of things that can be added or improved in this application. This app can be considered as a prototype for future developments. Since this app possesses some great newly introduced features, so the research conducted in this domain can be proved very helpful for other developers. Subtitles are the need of this era since everyone want their products to be globally recognized and in order to increase understanding of the product among people subtitles are used widely. In this chapter we will discuss some enhancements that can be made in this application to make the user experience better.

6.1.1 Reduce delays

Since we are doing subtitling on real-time videos, initially there’s a delay of 2 to 3 seconds between the speech recognition and text display on the screen. This delay can be reduced by using some more efficient speech to text converter tools. Note that this project has been made by using tools and platforms that are open source and easily available so that this application can be used freely by people for learning purposes. There are some tools that are more efficient but they are not free.

6.1.2 Save Videos Along with Subtitles

Up till now we cannot save the videos along with subtitles so that the user can access them later too. So this feature can be introduced save the videos along with subtitles so that they can be reused anytime. When saving the videos with subtitles the user can be allowed to correct the mistakes in the recognized text and translations.

6.1.3 Adding more languages

Up till now the app only supports English and Urdu Language. More languages can be added so that the app can support subtitles and translated subtitles in more languages. This can increase the use of this app widely.

6.1.4 Face detection and Speech Detection

The current version of the app does not support face detection feature. Face detection feature can be added in to the app along with speech detection so that the user may know from whom the recognized speech is coming from. Additionally we can provide subtitles in different color for different speakers so that if there are more than 1 speaker in the video his face can be highlighted with a rectangular box in the same color in the recognized speech from this user is displayed as subtitles.

6.1.5 Using Machine Learning

As we know that machine learning plays a great role in developing artificial intelligent applications. So Machine learning can be accommodated in this app to make it more efficient.

The machine learning model can be trained with videos and subtitles such that it can learn that how a word can be spoken in many different ways, how speakers generally says a word, the pronunciation of different names and so on. This can help to recognize, convert and translate speech more efficiently.

6.2 Conclusion

The first part of the project is some basic research on speech recognition, speech to text conversion, translation and displaying of subtitles. The research is based on some products and techniques related to these domains. This part also includes the need and usefulness of this project. The next part is software development of this project. This includes how this project is made and which tools and platforms are used for this purpose. In this part each and every practical aspect of this app is discussed in detail. This part also includes some practical research or we can say different hit and trial approaches that are used to achieve the final outcome. This research can play the role of do’s and don’ts for further development in this domain. Later this project is tested and the results have been noted in accordance with these tests. The last part of the project is based on the discussion of some future enhancements that can made in this app to make it more useful and efficient.

REFERENCES

[1] World Health Organization, "WHO: 1 in 4 people projected to have hearing problems by 2050," WHO, Geneva, 2 March 2021.

[2] S. Gaudin, "Computer World," 8 October 2014. [Online]. Available: https://www.computerworld.com/article/2822820/i-understand-you-now-theres-a-google-glass-app-for-hard-of-hearing-users.html.

[3] "Glass," [Online]. Available: https://www.google.com/glass/start/.

[4] N. Vega, "NewYork Post," 24 January 2020. [Online]. Available: https://nypost.com/2020/01/24/new-moverio-smart-glasses-could-help-deaf-theatergoers/.

[5] J. Gvora, "ScreenRant," 23 November 2020. [Online]. Available: https://screenrant.com/google-glass-smart-glasses-what-happened-explained/.

[6] "National Theatre," [Online]. Available: https://www.nationaltheatre.org.uk/your-visit/access/caption-glasses.

[7] C. Kelsall, "American Theatre," 23 January 2020. [Online]. Available: https://www.americantheatre.org/2020/01/23/with-smart-caption-glasses-the-eyes-have-it/.

[8] C. Huston, "Broadway News," 28 January 2020. [Online]. Available: https://broadwaynews.com/2020/01/28/galapro-tests-out-smart-glasses-with-live-captions-on-broadway/.

[9] P. Mehar, "InceptiveMind," 2020 February 3. [Online]. Available: https://www.inceptivemind.com/epson-moverio-smart-glasses-live-captions-deaf-theatergoers/11660/).

[10] A. Canary, "Rev," 16 October 2020. [Online]. Available: https://www.rev.com/blog/the-future-of-captions-augmented-reality].

[11] "SyncWords," [Online]. Available: https://www.syncwords.com/company/about.

[12] "BroadStream Solutions," [Online]. Available: https://broadstream.com/vocaption-live/#languages.

[13] "Oswald Labs," [Online]. Available: https://oswaldlabs.com/platform/shravan/apps/live-subtitles/.

[14] "Live Caption," [Online]. Available: http://www.livecaptionapp.com/.

[15] D. Copithorne, "Hearing Tracker," 7 February 2019. [Online]. Available: https://www.hearingtracker.com/news/google-live-transcribe-app.

[16] "App Store," [Online]. Available: https://apps.apple.com/us/app/live-transcribe/id1471473738.

[17] "gotalk.to," [Online]. Available: https://gotalk.to/.

[18] Fifth Generation Computer Corporation, "Speaker Independent Connected Speech Recognition," 11 Novmber 2013. [Online]. Available: https://web.archive.org/web/20131111101228/http://www.fifthgen.com/speaker-independent-connected-s-r.htm.

[19] F. Filippidou and L. Moussiades, "A Benchmarking of IBM, Google and Wit Automatic Speech Recognition Systems," in *IFIP Advances in Information and Communication Technology,* May 2020.

[20] Y. Gaur, W. S. Lasecki, F. Metze and J. P. Bigham, "The Effects of Automatic Speech Recognition Quality on Human Transcription Latency," in *13th International Web for All Conference (W4A '16),* New York, USA, April 2016.

[21] S.-Y. Chang, B. Li, D. Rybach, Y. He, W. Li, T. Sainath and T. Strohman, "Low Latency Speech Recognition using End-to-End Prefetching," Interspeech 2020, Shanghai, 2020.

[22] J. F. FOSTER, "UNIVERSITY COMMENTARY," *Higher Education Quarterly,* vol. 12, no.

2, pp. 189-193, February 1958.

[23] B. Turovsky, Interviewee, *The A.I. (R)Evolution.* [Interview]. 8 March 2021.

[24] B. Turovsky, "Ten years of Google Translate," Google, 28 April 2016. [Online]. Available: https://blog.google/products/translate/ten-years-of-google-translate/.

[25] J. Pitman, "Google Translate: One billion installs, one billion stories," Google, 28 April 2021. [Online]. Available: https://blog.google/products/translate/one-billion-installs/.

[26] P. Guzman and D. Husa, "Expanding automatic machine translation to more languages," Facebook Engineering Blog, 11 September 2018. [Online]. Available: https://engineering.fb.com/2018/09/11/ml-applications/expanding-automatic-machine-translation-to-more-languages/.

[27] F. Faes, "Slator 2021 Language Industry Market Report," Slator, 12 May 2021.

[28] R. Thomas, "Watson Anywhere: The Future," IBM THINK Blog, 21 October 2019. [Online]. Available: https://www.ibm.com/blogs/think/2019/10/watson-anywhere-the-future/.

[29] C. Bernstein, "face detection," TechTarget, [Online]. Available: https://searchenterpriseai.techtarget.com/definition/face-detection.

[30] "Everything About Face Tracking," sightcorp, [Online]. Available: https://sightcorp.com/knowledge-base/face-tracking/.