

EchoVision AI – Intelligent Assistive Technology for Visual Safety

Dr. B Meena Preethi¹, Lakshana K²

¹(Dept. of Software Systems & AIML, Sri Krishna Arts and Science College, Coimbatore, India

Email: meenapreethib@skasc.ac.in)

²(Dept. of Software Systems & AIML, Sri Krishna Arts and Science College, Coimbatore, India

Email: lakshanak23mss022@skasc.ac.in)

Abstract:

Visually impaired people often cannot tell whether the person at the door is a familiar visitor or a stranger, which affects both convenience and personal safety. This paper presents EchoVision AI, a native Android application that acts as a smart safety companion. Familiar people are registered after phone OTP verification, and a FaceNet TensorFlow Lite model converts each face into a 512-dimensional embedding. During monitoring, Google ML Kit detects faces in live camera frames, and the system compares each live embedding with the stored ones using cosine similarity (threshold 0.75, three consecutive results). The result is announced by Text-to-Speech as a registered name or an unknown person. Visitors are logged and shown on a dashboard. When an unknown person is currently detected, pressing Volume Down three times within three seconds sends an SMS with a Google Maps location link to the trusted contacts. All records are kept in a local Room database.

Keywords — Assistive Technology, Face Recognition, FaceNet, Google ML Kit, Android, Text-to-Speech, Emergency Alert

I. INTRODUCTION

Visually impaired people rely on sound, touch, and the help of others to understand their surroundings. Deciding whether to open the door is difficult because they cannot see who is standing there, and an unfamiliar visitor can be a safety risk. Normal doorbells and security cameras only show that someone is present; they do not tell a blind user who the person is.

Modern smartphones already contain a camera, speaker, GPS receiver, and SMS capability, so a safety application can be built without extra hardware. EchoVision AI uses these resources to detect a visitor's face, identify the person from registered faces, and speak the result aloud. It also keeps a visitor history, manages trusted contacts, and sends a location-based emergency SMS.

The main contributions of this work are: (1) a voice-first visitor identification system using ML Kit and FaceNet on the device, (2) a stable recognition strategy based on a similarity threshold and consecutive-result confirmation, (3) a privacy-conscious local storage design in which images stay on the phone and the database holds embeddings and file paths, (4) a visitor log and dashboard that include unknown visitors, and (5) a hands-free emergency trigger using the physical Volume Down button.

II. LITERATURE REVIEW

Face detection has progressed from classical cascade methods [2] to deep learning models. Detection only locates a face in an image; it does not say whose face it is. Face recognition methods such as FaceNet [1] map a face image to a compact embedding so that the similarity between two faces can be measured by distance or angle, and this allows

new people to be registered without retraining the model.

Mobile frameworks make these techniques practical on a phone. Google ML Kit offers on-device face detection [4], and TensorFlow Lite allows deep models such as FaceNet to run as lightweight inference on Android [7]. On-device processing avoids sending face images to a server, which supports privacy and reduces dependence on the network.

Assistive technology for visual impairment has mostly addressed navigation, text reading, and object description. The World Health Organization notes that many people live with vision impairment and that assistive products are not equally available [3]. Few systems combine visitor identification, voice announcement, visitor history, and an emergency alert in a single lightweight mobile application, and EchoVision AI is designed to fill this gap.

III. PROBLEM STATEMENT

A visually impaired person cannot visually verify who is nearby. Existing doorbells and security cameras indicate presence but typically do not announce a known person's identity, maintain a visitor history, detect unknown persons, and support emergency assistance together. Cloud-based face systems also raise privacy concerns and need continuous internet access. A system is therefore needed that identifies visitors, speaks the result, keeps personal data on the device, maintains a history, and allows a quick emergency alert with location.

A. Existing and Proposed System

TABLE I
COMPARISON OF EXISTING AND PROPOSED SYSTEMS

Capability	Typical Doorbell / Camera	EchoVision AI
Identity by voice	Not provided	Name announced via TTS
Known vs. unknown	Presence only	Embedding match
Visitor history	Not combined	Dashboard with photo path
Emergency alert	Not integrated	SMS with map link

IV. PROPOSED SOLUTION

EchoVision AI is a native Android application written in Kotlin. Its working flow is: splash screen, home screen, trusted person registration with OTP verification, face capture, embedding storage, monitoring, face detection, face recognition, voice announcement, visitor logging, and the emergency trigger with location-based SMS.

A. Technology Stack

TABLE II
TECHNOLOGIES USED IN ECHOVISION AI

Component	Purpose
Android Studio, Kotlin	Application development and logic
XML layouts, View Binding	Screen design and view access
CameraX	Camera preview and face capture
Google ML Kit	Face detection
FaceNet (TensorFlow Lite)	512-dimensional face embedding
Cosine similarity	Embedding comparison
Room (SQLite)	Local data storage
Firebase Authentication	Phone OTP verification
Android Text-to-Speech	Voice announcements
Android Location APIs	Device location
Google Maps URL	Location link in SMS
Android SMS APIs	Emergency SMS

B. User Interface and Navigation

The application opens with a splash screen and then the home screen. From the home screen the user can register a trusted person, start monitoring, view registered persons, manage trusted contacts, and open the visitor dashboard. Voice feedback and automatic monitoring reduce the need to read the screen.

C. Trusted Person Registration

A trusted person is a familiar individual, such as a family member or friend, whom the system should recognize. The user enters the name, phone number, and relationship, and then completes a phone OTP verification flow built with Firebase Authentication. After verification, the front camera captures the face using CameraX, with a retake option if the capture is poor. The face is processed by ML Kit and prepared for the FaceNet model, which produces the embedding. The details and the embedding are saved in the Room database.

D. Face Detection and Recognition

This is the core AI module. ML Kit and FaceNet serve different purposes: ML Kit finds where a face is in the frame, and FaceNet converts the face into numbers that represent its distinguishing features. The FaceNet TensorFlow Lite model takes a 160×160 pixel input and outputs a 512-dimensional embedding. The live embedding is compared with every stored embedding using cosine similarity, and a match is accepted only if the score reaches the configured threshold of 0.75.

To avoid false results from a single unstable frame, three consecutive matching results are required before a registered person's identity is confirmed. If no registered person satisfies the threshold, three consecutive unknown results are required before the visitor is treated as unknown. When no face is detected, the previous detection state is reset. The recognition settings are summarized in Table III.

TABLE III
RECOGNITION PARAMETERS

Parameter	Value
Model input size	160×160 pixels
Embedding size	512 floats (2048 bytes)
Similarity measure	Cosine similarity
Match threshold	0.75
Results to confirm match	3 consecutive
Results to confirm unknown	3 consecutive
No face detected	State reset
Repeated announcements	Cooldown mechanism

E. Voice Assistance

Because the target users are visually impaired, results are given as audio. After recognition, Android Text-to-Speech announces a message such as the person's name, or an unknown person alert. A cooldown mechanism prevents the same announcement from repeating on every frame while the person stays in front of the camera.

F. Trusted Contacts Management

Trusted contacts are the people who receive the emergency SMS. The user can add, view, edit, and delete a contact's name, phone number, and relationship. A trusted person and a trusted contact

are different: a trusted person is registered for face recognition and has an embedding stored, whereas a trusted contact is stored only for emergency messaging. The same individual may play both roles, but the two data types are kept separately.

G. Visitor Monitoring and Dashboard

While monitoring is active, the camera continuously supplies frames to the detection and recognition pipeline. Each confirmed visitor, recognized or unknown, is saved as a visitor log entry with the visitor name, visitor type, date, time, and photo path. The visitor dashboard shows these records under Today, This Week, and All tabs, and the user can delete individual records or clear the history.

H. Emergency Alert

The emergency feature is available while an unknown person is currently detected. The user presses the physical Volume Down button three times within three seconds, which avoids searching for an on-screen button. The application then obtains the device location, builds a Google Maps link from it, and sends an SMS to all saved trusted contacts. The message contains an EchoVision emergency alert notice, the unknown person detected message, and the location link. Long messages are sent as multipart SMS.

I. Database Design and Storage

Room, an abstraction layer over SQLite, stores the application data locally on the phone. Face images and visitor photos are saved in device storage, and the database keeps only the file paths. The facial embedding is stored as a serialized byte array ($512 \text{ values} \times 4 \text{ bytes} = 2048 \text{ bytes}$), so a registered face need not be reprocessed each time. The three tables are independent, with no foreign key relationships defined (Table IV).

TABLE IV
ROOM DATABASE TABLES

Table	Fields
registered_persons	id, name, phone, relationship, faceImagePath, faceEmbedding
trusted_contacts	id, name, phone, relationship
visitor_logs	id, visitorName, visitorType, date, time, photoPath

J. Permissions and Privacy

The application requests the permissions listed in Table V, and each feature works only when the related permission is granted. Registered person records, embeddings, trusted contacts, and visitor logs are held in the local Room database, and Firebase is used only for the OTP verification flow. The current design is therefore local-storage based and does not keep these records in a cloud database.

TABLE V
ANDROID PERMISSIONS

Permission	Needed For
Camera	Face registration and visitor monitoring
Location	Obtaining the device location
SMS	Sending the emergency message
Internet	OTP verification and network services

K. System Flow Diagram

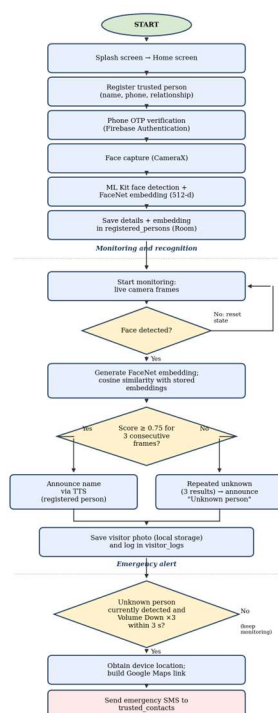


Fig. 1. System flow diagram of EchoVision AI

V. RESULTS AND DISCUSSION

Each implemented feature was checked against its designed behavior. Table VI lists the checks, and Table VII lists the recognition measurements recorded from test runs.

A. Functional Verification

TABLE VI
FUNCTIONAL VERIFICATION

Feature	Expected Behavior	Result
OTP verification	Registration continues only after verification	Pass
Face registration	Embedding saved with person details	Pass
Known person	Name announced after 3 matches	Pass
Unknown person	Unknown alert after 3 results	Pass
No face	Detection state resets	Pass
Visitor log	Entry saved, shown in tabs	Pass
Trusted contacts	Add, edit, delete work	Pass
Emergency SMS	Map link sent after trigger	Pass

B. Recognition Performance

TABLE VII
RECOGNITION PERFORMANCE

Metric	Value
Recognition accuracy	[90%]
Average response time	[10 seconds]
False accept rate	[0%]
False reject rate	[0%]

C. Discussion

The design shows that useful visitor identification can run on an ordinary smartphone. Separating detection (ML Kit) from recognition (FaceNet with cosine similarity) keeps each step simple, and the consecutive-result rule reduces the chance of a wrong announcement caused by one poor frame. Storing images locally and keeping only embeddings and paths in the database limits privacy risk, and the voice-first design removes the need to look at the screen. The emergency feature uses a physical button and the phone's own SMS service, so it needs no extra hardware.

D. Limitations

Recognition accuracy varies with lighting, camera quality, face angle, and image quality, and the 0.75 threshold cannot guarantee perfect recognition in every situation. Location accuracy depends on GPS availability and permissions, and SMS delivery depends on the mobile network and

SIM. The application does not call the police automatically and does not identify unknown people by name. Visitor classification currently depends only on registered or unknown status, and sending the unknown person's photo as an online link in the SMS is not yet implemented.

VI. CONCLUSION AND FUTURE WORK

This paper presented EchoVision AI, an Android safety companion for visually impaired people. It integrates trusted person registration with OTP verification, ML Kit face detection, FaceNet embeddings with cosine similarity recognition, voice announcements, trusted contact management, visitor history, and a location-based emergency SMS triggered by the Volume Down button. Records are kept in a local Room database, which supports privacy.

In future work, we plan to add context-aware visitor classification, upload the captured photo of an unknown person to secure storage and send a shareable link in the emergency SMS, improve recognition under different lighting and face angles, add filtering and reporting to the visitor history, and strengthen protection of face data and visitor photos.

REFERENCES

- [1] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in *Proc. IEEE CVPR*, 2015.
- [2] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. IEEE CVPR*, 2001.
- [3] World Health Organization, *World Report on Vision*, Geneva, 2019.
- [4] Google, "ML Kit face detection," *Google for Developers documentation*.
- [5] Google, "Room persistence library," *Android Developers documentation*.
- [6] Google, "Firebase Authentication: Phone number sign-in," *Firebase documentation*.
- [7] Google, "TensorFlow Lite," *TensorFlow documentation*.
- [8] Google, "CameraX overview," *Android Developers documentation*.