Autonomous Drone Assistance for Medical Emergencies

Liam Edmonds-Seng
101571206

Dang Khoa Le

103844421

Aarush Singh

103798065

Erfan Mangani

103218929

***Abstract*—Delays in emergency medicine in and out of the emergency room plague Emergency Medical Services (EMS). By applying drone delivery with and expanded telehealth service, the project aids in tackling overcrowding, slow triage times, and limitations in existing telemedicine. This is combined with an integrated webapp for communication and dispatch, integrating large language models (LLM) to provide better safeguards against human error in triage.**

Keywords—telehealth, drone, webapp, llm, medication.

I. Introduction

Our project aims to address the strain on emergency room (ER) capacity and triage time, by improving assessment earlier in the emergency response process, diverting non-critical conditions away from ER waiting rooms, and improving knowledge of conditions before they arrive at the ER. The constant flow of patients into a given emergency room can be overwhelming on the limited staff and resources available, with overcrowding being an issue that is both caused by and impacts the triage times of patients [1]. The severity of this is most visible in metropolitan areas, but the issues in rural areas are often found further down the line in ambulance response, with limited resources and wider areas that need to be covered by rural emergency services [2]. The project addresses these issues by expanding the capabilities of telephone triage with boarder means of assessment, and additional failsafe measures. These systems are an ever-expanding aspect of digital health systems that help improve the broader health system function but are becoming especially prevalent in the context of emergency medicine [3]. Systems such as the Victorian Virtual Emergency Department (VVED) are designed to alleviate the pressure on emergency rooms and ambulances by diverting traffic from non-life-threatening health conditions towards a telehealth approach [4][5].

The proposed design is an integrated system of drones capable of delivering simple medical equipment such as blood pressure monitors with the possibility of medication delivery, alongside a webapp for assisting in telehealth assessment. Although there are drone delivery systems on the market, the specific application as an extension of telehealth emergency services is yet to be explored, with further expansions upon the existing virtual emergency response systems being a critical step forwards for our emergency services. The project has some risks associated that have been addressed as the design process has progressed. The limitations around drone flight are addressable with json map integration, which is outlined later in the report. The risks associated with handling health data are addressed by only handling data given by the patient, with other data going through the normal telehealth systems.

II. Materials and Methods

A. Drone Technology

The drone utilised for this system was the Great Shark 330 PRO VTOL from Foxtech, [6] selected due to its robust design, payload capability of up to 10 kg, and recommended cruising speed of 79 km/h. Given it was necessary for the drone to be capable of delivering medication/equipment to a wide range of locations and environments, a VTOL solution was selected as it provided the system with the greatest operational range while not compromising the drones capabilities in dense, potentially, areas [7]. Given its flight time of 30-40 minutes, this means that the chosen solution can deliver to a maximum radius of 26 km from its base position when making a reciprocal flight and 52 km when making a no-return delivery. However, this range is contingent on flight path and weather conditions.

As for the payload, the drone offers both internal and external mounting points which can be used to haul containers which hold the relevant medical device or medication. Both internal and externally mounted storage compartments will feature adequate isolation to provide temporary temperature regulation for the duration of the drone flight.

Global Positioning System (GPS) and Inertial Measurement Unit (IMU) sensors facilitate part of the autonomous navigation and are managed through a Cube Orange Plus from CubePilot, running ArduPilot [8]. An RGB depth camera will be located beneath the drone, looking downwards. Using both the produced depth-cloud and RGB image to create an RGB-D image, a suitable landing spot can be established using a RGB-D image segmentation model [9]. A single laser mechanical lidar was placed on top of the drone to monitor the surroundings of the drone during all stages of the drone's flight to prevent the drone from sustaining collision or taking flight in unsafe environments.

To handle and facilitate the communication between the sensors and the Cube Orange, a Nvidia Jetson Nano will be connected to the cube via ethernet. To facilitate the automatic path planning and sensor data monitoring, the Jetson will run an instance of Robot Operating System (ROS) Noetic and will communicate to the cube via the MAVROS package provided by ArduPilot to send messages through the ethernet via UDP.

While the Cube Orange comes equipped with an internal IMU and GPS, a Here4, a professional High Precision Dual-band Real-Time Kinematic (RTK) navigation module [10], will also be externally mounted to provide greater navigational precision. The fusion of the GPS data from both the Cube and Here4 will be handled by the ArduPilot backend.

Due to budgetary constraints, as well as limitations in time and operating space, we were unable to create and test a physical implementation of our solution. However, since a ROS environment was employed to facilitate communication between the Cube and Jetson, a ROS Noetic environment was established to simulate flight capabilities, Jetson-to-Cube communication, and logging functions. The physics component of the simulation was managed by Gazebo. Flight plans were tested using simulated rural delivery scenarios to closely replicate realistic payload and routing conditions.

The communication between the system backend and the drone path planning involved a simple Application Programming Interface (API) that passed the latitude and longitude coordinates of the delivery location. The Jetson then received these coordinates and set them as waypoints within the Ardupilot backend, creating a direct path from the drone's take-off position to the specified delivery coordinates.

B. *Mobile Application*

The mobile application was developed using React Native [11], enabling seamless cross-platform compatibility for both Android and iOS. Emphasis was placed on performance responsiveness and usability, with user experience evaluated under emergency stress scenarios. The design supports intuitive bottom navigation, informative icons, and low-latency transitions between interfaces.

Key components of the application include:

- Login/Signup Screen: Implements secured credential verification using a FastAPI backend [12], with user-credentials stored in MongoDB (database) [13].
- Medical Chatbot: Integrated with a standalone Artificial Intelligence (AI) agent hosted on Hugging Face Space (https://huggingface.co/spaces/BinKhoaLe1812/Medical-Chatbot) - server [14], allowing users to query symptoms and conditions in real time.
- Medical Profile (MP): Supports dynamic updates and stores critical health information and emergency details. Medical documents (e.g., prescriptions or history reports) can be uploaded and transcribed using the Qwen2.5-VL 7B vision-language model [15].
- Emergency Screen: Enables immediate emergency communication. When activated, the user's voice is recorded and transcribed using OpenAI's Whisper-large-v3 [16], then relayed to backend services for triage processing.
- QR Wallet: Generates unique Quick-Response (QR) codes based on the user ID and user-token, used for drone delivery verification. Although face-ID and PIN-based authentication were not fully implemented due to time-constraints, this module is intended to prevent unauthorized access to delivered medication [17].

C. *Web-Service Triage Portal*

The triage portal was developed in React and deployed at: https://dispatch-portal-amber.vercel.app

It provides clinicians and first responders with centralized access to real-time emergency events and actionable insights. The interface features responsive design, quick routing, and dynamic alert handling.

Key components:

- Credential Module: Integrates secure authentication through backend FastAPI endpoints.
- Dashboard: Presents live and historical alerts. Facilitates drone dispatch or ambulance routing based on triage AI suggestions. Navigation among panels is supported via top-bar tabs.
- Emergency Logs: Archives active and resolved emergency cases, visualized through charts (e.g., line, pie charts). Each log displays patient data, profile embeddings, large language models (LLM) driven recommendations [18].
- Drone Status: Displays in-transit drones to the patient address (from MP), visualizing Global Positioning System (GPS) and dispatch conditions in real time.
- Communication Module: Bridges backend processing with nearby hospitals, transmitting patient profiles, emergency details (including address and emergency contact), and AI-driven triage plans.

D. *MIMIC-IV Dataset Integration*

The system is embedded with the MIMIC-IV version 3.1 dataset, which contains structured and de-identified clinical records for over 65,000 Intensive Care Unit (ICU) patients and more than 200,000 emergency department admissions. The dataset represents over 364,000 unique individuals and is derived from electronic health records maintained at the Beth Israel Deaconess Medical Center [19]. It includes patient demographics, medication records, procedure notes, and emergency triage data, forming the knowledge base for similarity search and AI triage decisions.

E. *Backend Architecture*

1) RAG Service (User Onboarding, MP Embedding)

Endpoint: *https://huggingface.co/spaces/BinKhoaLe1812/Medical_Profile*

Purpose: Accepts user medical data, performs semantic embedding, stores MP data in Facebook-AI similarity search (FAISS) index with Retrieval Augmented Generation (RAG) for fast query-answer (QA) searches.

Key Techniques:

- Sentence Embedding: Uses all-MiniLM-L6-v2 from sentence-transformers [20] to embed long-form user medical history into vector form for similarity search.
- Vector Indexing: FAISS IndexFlatL2 [21] is initialized per user or reused across sessions. Stored on disk per user ID, allowing flexible retrieval.
- MongoDB: Stores structured user profiles and corresponding FAISS file references. Ensures atomic updates when embedding changes.
- Profile Snapshot Sync: MongoDB record of profile is always updated with the latest input, ensuring frontend/mobile display consistency.

Additional Features:

- CR-powered medical document summaries (e.g., prescriptions, health reports) are extracted and summarized for embedding using Alibaba's Qwen2.5-VL-7B model.
- Performs user verification (username + password) before updating data.

2) Triage Service (Emergency Voice to Decision Support)

Endpoint: https://huggingface.co/spaces/BinKhoaLe1812/Triage_LLM

Purpose: Processes voice input and profile context to generate clinical triage recommendations.

Key Techniques:

- ASR (Voice Transcription): Whisper-large-v3 (transformers.pipeline) transcribes emergency speech to text with chunk-based streaming.
- Context Construction: Combines transcription + user summary into a structured FAISS query string.
- RAG Pipeline:
  - FAISS index (using MIMIC-IV guideline) is loaded from MongoDB GridFS, decompressed and used for semantic search.
  - Topmost similar QA pairs are retrieved from MIMIC-IV corpus to provide triage suggestions.
- LLM Integration: Gemini 2.5 Pro [22] is used to turn context into a structured-interpreted JSON output, including:
  - Key triage highlights
  - Treatment recommendations
  - Medication (if needed)
- Storage Optimization:
  - GridFS [13] stores binary FAISS blobs and QA map JSON with zlib compression, minimizing bandwidth + storage cost.
  - Uses in-memory caching to prevent redundant FAISS reloads per session.
- MongoDB Dual Cluster Setup: One for patient profiles, another for triage QA knowledge base.

3) *Containerization & Cloud Execution*

Docker-Ready: Each backend module is containerized with strict control over HF cache locations (HF_HOME, SENTENCE_TRANSFORMERS_HOME) to ensure Hugging Face dependencies work in rootless Docker environments [23].

Execution Port: Services run via unicorn on port 7860, mapped in Dockerfile.

III. Results

A. Drone Operations

Drone testing demonstrated successful navigation, maintaining precise GPS waypoints under varying conditions, confirming its reliability and suitability for medical delivery. The ROS environment

B. Mobile Application

Testing of the voice-triggered system consistently achieved high responsiveness (no failure), testing on edge cases (voice with strong-accent, disruptive, unclear) demonstrating effectiveness in emergency scenarios. User interaction tests validated the intuitive design and successful MP retrieval and continuous update.

C. Web-Service Triage Portal

The portal effectively processed 20 simulated emergency scenarios, showing rapid dispatch decision capabilities, smooth user interaction, and accurate retrieval of patient profiles with actionable-suggestion on LLM-driven and FAISS-based similarity searches.

D. Backend Architecture

The LLM-driven backend consistently produced contextually accurate medical recommendations aligned closely with professional clinical assessments. The integration of FAISS retrieval from embedded MIMIC-IV datasets ensured relevant and precise emergency responses.

IV. Discussion

A. Interpretation and Error Analysis

Overall system performance was robust; however, minor issues arose in voice recognition accuracy under significant ambient noise conditions, indicating potential for improved noise filtering in future iterations. FAISS occasionally faced challenges distinguishing closely related medical profiles, suggesting the need for enhanced embedding methodologies.

B. Regulatory Compliance

Compliance with CASA regulations and HIPAA data management standards [24] was maintained throughout system design and testing phases [1]. Given its medical intervention capabilities, the system is preliminarily classified as a Class II medical device, requiring formal regulatory assessments for market deployment.

C. Economic Viability

The system's financial viability was projected, accounting for initial investments ranging from AUD 500,000 to AUD 1,000,000. Costs encompassed software and drone hardware development, cloud infrastructure, regulatory compliance, and annual maintenance [5]. Reimbursement strategies through partnerships with healthcare providers and insurance entities were identified as potential financial pathways.

D. Recomendations for Future Development

Future system enhancements should include:

- Conducting extensive field trials with Ambulance Victoria for real-world validation.
- Obtaining comprehensive CASA approvals and refining airspace operation protocols.
- Optimizing speech recognition models to improve performance in high-noise environments.
- Completing detailed patent and licensing assessments to secure operational freedom.

V. Conclusion

The project combines drone delivery with a multi-layer telehealth system to relive pressure on existing emergency services. The integration of AI as an additional layer of the triage process ensures a layer of security against human errors with regard to drug interactions during tirage. These technologies combine to expand the digital health model further, allowing for more comprehensive health services overall.

References

[1] G. Savioli et al., "Emergency department overcrowding: Understanding the factors to find corresponding solutions," Journal of Personalized Medicine, vol. 12, no. 2, p. 279, Feb. 2022, doi: https://doi.org/10.3390/jpm12020279.

[2] A. Greaves, "Community Health Program", Victorian Auditor-General's Office, Jun. 2018. [Online] Available: https://www.audit.vic.gov.au/report/community-health-program?section=

[3] E. Vecellio, M. Raban, and J. Westbrook, "Secondary ambulance triage service models and outcomes: A review of the evidence," Australian Institute of Health Innovation, University of New South Wales. 2012. [Online] Available: https://www.mq.edu.au/__data/assets/pdf_file/0006/687021/Secondary-Ambulance-Triage-Literature-Review_final.pdf

[4] Ambulance Victoria, "Victorian Virtual Emergency Department (VVED)," Ambulance Victoria, 2024.

https://www.ambulance.vic.gov.au/community/victorian-virtual-emergency-department-vved/

[5] "WHO guideline: recommendations on digital interventions for health system strengthening," World Health Organisation, Geneva, 2019. [Online] Available: https://iris.who.int/bitstream/handle/10665/311977/WHO-RHR-19.8-eng.pdf?sequence=1

[6] FOXTECH, "FOXTECH Great Shark 330 PRO VTOL," foxtechfpv. [Online]. Available: https://www.foxtechfpv.com/foxtech-great-shark-330-pro-vtol.html?srsltid=AfmBOoqC4pqr0F7HlGRMyMeHmARIRNKYQ42DMOO2rYQ8-qZNTZc6QpPC

[7] A. Vidović, I. Štimac, T. Mihetec, and S. Patrlj, "Application of Drones in Urban Areas," Transportation Research Procedia, vol. 81, pp. 84–97, 2024, doi: 10.1016/j.trpro.2024.11.010.

[8] ArduPilot, "VTOL configuration and logging," 2023. [Online]. Available: https://ardupilot.org.

[9] B. Xu, R. Hou, T. Ren, and G. Wu, "RGB-D Video Object Segmentation via Enhanced Multi-store Feature Memory," in Proceedings of the 2024 International Conference on Multimedia Retrieval, May 2024, pp. 1016–1024. doi: 10.1145/3652583.3658036.

[10] CubePilot, "Here 4 Manual." [Online]. Available: https://docs.cubepilot.org/user-guides/here-4/here-4-manual#overview

[11] React Native, "Cross-platform mobile development framework," 2023. [Online]. Available: https://reactnative.dev.

[12] FastAPI, "FastAPI," fastapi.tiangolo.com, 2023. https://fastapi.tiangolo.com/

[13] MongoDB Atlas, "Cloud database service," 2023. [Online]. Available: https://www.mongodb.com/cloud/atlas.

[14] "Spaces - Hugging Face," huggingface.co. https://huggingface.co/spaces

[15] QwenLM, "GitHub - QwenLM/Qwen2.5-VL: Qwen2.5-VL is the multimodal large language model series developed by Qwen team, Alibaba Cloud.," GitHub, 2024. https://github.com/QwenLM/Qwen2.5-VL

[16] OpenAI, "Whisper: Robust Speech Recognition via Large-Scale Pretraining," 2023. [Online]. Available: https://openai.com/whisper.

[17] H. Jin et al., "QR Code Security in Healthcare Systems: A New Approach to Privacy and Authentication," IEEE Access, vol. 9, pp. 78345-78357, 2021.

[18] K. Lee et al., "AI-driven triage systems in emergency healthcare," J. Med. Syst., vol. 46, no. 3, pp. 1–12, 2022.

[19] A. Johnson et al., "MIMIC-IV." PhysioNet. doi: 10.13026/07HJ-2A80

[20] Hugging Face, "sentence-transformers/all-MiniLM-L6-v2 · Hugging Face," huggingface.co. https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2

[21] Y. Han et al., "FAISS and Retrieval-Augmented Generation in Large Language Models for Medical Decision Support," IEEE Access, vol. 10, pp. 127981-127995, 2022.

[22] "Gemini 2.5 Pro," Google DeepMind, 2025. https://deepmind.google/models/gemini/pro/

[23] Docker, "Enterprise Application Container Platform | Docker," Docker, 2024. https://www.docker.com/

[24] Civil Aviation Safety Authority (CASA), "Civil Aviation Safety Regulation 1998," Canberra, ACT, Australia, 2023. [Online]. Available: https://www.casa.gov.au. J.