GMSJ | Innovations Open Access

Utilizing a Voice-Guided Mixed Reality and Artificial Intelligence Based Platform for Difficult Intubation Simulation in Austere Teaching Environments

Boseong Kim [1] | Anant Shukla, MD [2,3] | Tuna Ozturk [1] | Helen Cui [1] | Jack Bolitho [1] | Timothy Tregubov, MA[1] | Harman Gill, MD [2,4] | Scott Lancaster, PhD [2] | Matthew Roginski, MD MPH [2]

[1]Digital Applied Learning and Innovation (DALI) Labs, Dartmouth College, Hanover, New Hampshire, USA
[2]Departments of Emergency Medicine and Critical Care Medicine, Dartmouth Hitchcock Medical Center, Lebanon, New Hampshire, USA
[3]Department of Emergency Medicine and Orthopedic Surgery, Evans Army Community Hospital, Fort Carson, Colorado, USA
[4]Departments of Emergency Medicine and Critical Care Medicine, NYU Langone Health, Long Island, New York, USA

Correspondence: Anant Shukla (shuklaa100@gmail.com)

Received: 3 November 2025 | **Accepted**: 12 March 2026

Funding: This research was supported by a Pilot Research Grant 250-4151 from the Hitchcock Foundation in addition to an Augmented Reality in Healthcare Innovation Grant from the DALI lab at the Magnuson Center at Dartmouth College. The associated testing protocol was review and approved by the Dartmouth Hitchcock Institutional Review Board.

Keywords: airway simulation | augmented reality | medical simulation | artificial intelligence in medical simulation | bloody airway simulation

Abstract

Background: Intubating a blood-contaminated airway is a high acuity, low occurrence procedure. In resource-limited environments, a high-fidelity bloody airway simulation is restricted by cost, equipment, and a lack of expert feedback. To bridge this gap, we developed a high-fidelity artificial intelligence (AI) and augmented reality (AR) informed simulation tool. Inside an AR headset, an overlay is generated over a standard airway manikin. The user then proceeds through a bloody airway scenario with integrated instructional feedback videos.

Methods: We utilized a Unity AR application integrated into an AR Headset that interacts with a laryngoscope and removable pressure, light, and proximity sensors inside a manikin. The sensors and laryngoscope are connected to a Raspberry Pi that transmits sensor and video data via WebSocket to the headset. The headset application also processes audio via Meta's Wit.AI service for transcription and intent recognition. The application handles simulation logic, sensor / video data handling, and the AR guidance display. A decision tree algorithm enables user interaction and feedback.

Results: Both non-medical users at the 2024 DALI exhibition and individuals with emergency medicine backgrounds at the 2025 Society for Academic Emergency Medicine Innovation exhibit reported that the device was educational for difficult airway management. However, challenges have arisen in terms of stable connectivity with the Raspberry Pi under certain internet network conditions.

Conclusions: Utilizing readily available AI and AR technologies, a simulation device can be developed to teach difficult airway management in resource-limited areas, providing both experience and constructive feedback from experts.

ABBREVIATIONS

- **API –** Application Programming Interface; a system that allows different software programs to communicate with each other.
- **AR** – Augmented Reality; technology that overlays digital content onto the real world, typically viewed through a headset or mobile device.
- **DAG** – Directed Acyclic Graph; a data structure representing a sequence of steps or dependencies where tasks must be completed in order without circular dependencies.
- **JPEG** – A compressed image file format used for efficient storage and transmission of photos and video frames.
- **JSON** – JavaScript Object Notation; a text format for storing and transmitting structured data between applications.
- **LLM –** Large Language Model; A type of Artificial Intelligence trained on vast amounts of data to understand, summarize, and generate human language (e.g., ChatGPT).
- **NLP –** Natural Language Processing; computer technology that interprets and understands human speech or text.
- **OpenCV** – Open Computer Vision; a software library for processing and analyzing video and image data in real-time.

TERMS & DEFINITIONS

- **Unity –** A game engine used for developing 3D/AR/VR applications
- **Raspberry Pi** – A low-cost, single-board computer often used for embedded systems and sensor integration
- **WebSocket** – A communication protocol that enables real-time, bidirectional data exchange between devices over a network connection
- **Wit.AI –** A voice recognition service provided by Meta that transcribes speech to text and extracts user intent
- **Coroutine** – A programming function that can pause execution and resume later, allowing other code to run concurrently
- **Gpiozero –** A Python library that simplifies programming of sensors and hardware connected to Raspberry Pi's input/output pins
- **Thread** – A sequence of programmed instructions that can run independently, allowing multiple operations to occur simultaneously
- **Texture** – In computer graphics, a digital 2D image applied to the surface of a 3D object

1 | INTRODUCTION

Frontline medical providers are frequently required to perform an emergency intubation to save a patient's life, such as during a heart attack or other episodes of respiratory distress. Intubations, particularly when in austere settings, are high-stress procedures that can have life-threatening consequences for the patient if performed incorrectly. To that end, the proficiency of this skill among emergency medical providers cannot be overstated [1]. Research has shown that independent intubation practice with manikin feedback on objective measures such as dental pressure can facilitate mastery and enhance procedural education for learners in emergency medicine residencies, anesthesia residencies, and critical care fellowships [2]. Key to any simulation-based education is how closely the clinical scenario mirrors reality. Research has rapidly developed in the areas of both AR and AI to increase the fidelity of the manikin and the simulations [3]. Using sensors embedded in the manikin, users can be evaluated on their performance, given feedback, and aided on identifying anatomic structures [4,5]. A review of the literature shows that integrating these novel technologies into education improves both experience and training quality [6,7]. Difficult airway simulation tools have been attempting to fill this innovation gap, and challenges remain in integrating AR/AI into difficult airway management education, particularly in austere environments [8,9]. This technical paper shares the lessons learned and our approach to translating the advances in structural and software components of AR and AI into next-generation simulation innovations.

2 | METHODS

The main purpose of this study is to demonstrate the efficacy of the integration of a physical model with sensors and voice commands into a real-time program. This study focuses on the simulated intubation of a bloody airway for training and assessment. The system is meant to simulate a resuscitation bay with scoring functionality that allows participants to determine whether their intubation is successful. This study used a standard airway manikin with a GlideScope laryngoscope.

The AR Airways intubation simulator employs a distributed architecture comprising three primary subsystems: (1) a sensor-implanted medical manikin with embedded computing, (2) an augmented reality headset running the simulation engine, and (3) network infrastructure facilitating communication between components.

2.1 Implementation

2.1.1 Raspberry Pi Implementation

The medical manikin is equipped with a Raspberry Pi 5 that interfaces with multiple input devices:

- A USB-connected video laryngoscope for real-time airway visualization
- Two infrared break-beam sensors positioned in the trachea and esophagus for detecting tube placement
- A magnetic reed switch sensor mounted on the epiglottic fold for monitoring laryngoscope blade positioning

The Raspberry Pi functions as a wireless access point, exposing two WebSocket servers on ports 8765 (video streaming) and 8766 (sensor data). This architecture enables low-latency, bidirectional communication between the manikin and the AR headset. Due to the Raspberry Pi's limited networking capabilities, an Ethernet connection was established with an external laptop. The laptop serves as a gateway for internet-bound requests, enabling access to cloud-based services while maintaining network isolation for the primary simulation components. This hybrid approach maintains low-latency local communication for real-time sensor and video data. Simultaneously, it leverages the laptop's superior network connectivity for cloud API calls (Figure 1).

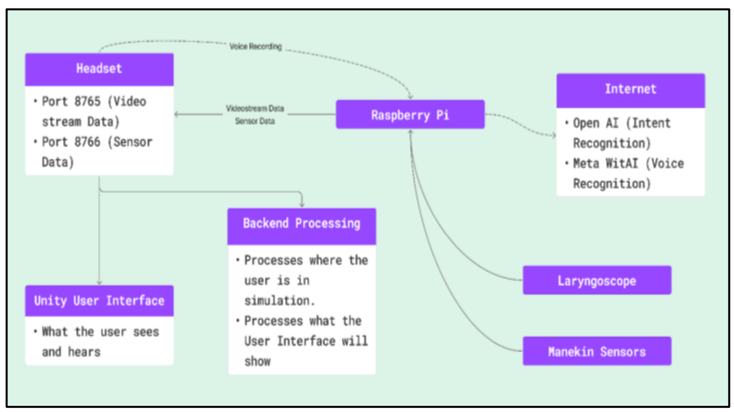

Figure 1. Raspberry Pi Integration

2.1.2 Meta Quest

The Meta Quest 3 headset hosts the Unity-based simulation engine, which manages:

- Real-time rendering of augmented reality overlays and user interfaces
- State management through a dependency graph that represents the intubation procedure workflow
- Integration with speech recognition (Meta Wit.AI) and natural language processing (OpenAI) services
- Sensor data interpretation and video stream processing
- Performance assessment and feedback generation

2.1.3 Manikin Sensor Integration

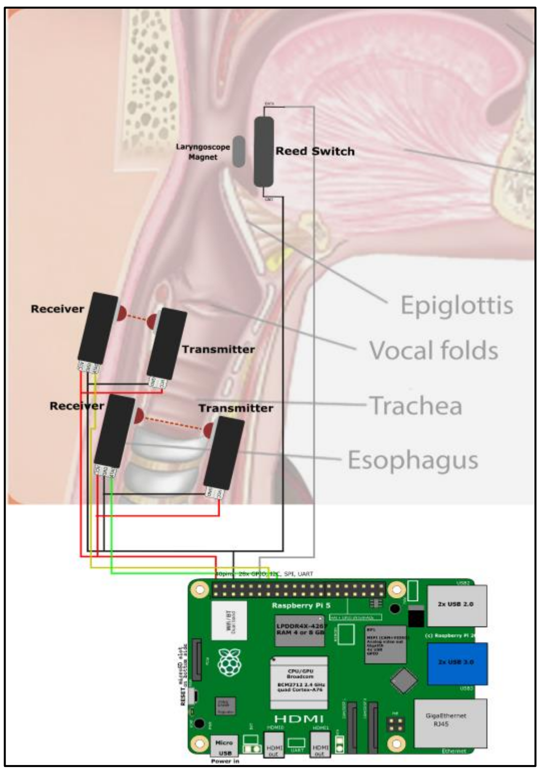

Figure 2. Sensor Placement

Strategically placed sensors re embedded in the manikin to monitor and evaluate the accuracy of airway-opening maneuvers (Figure 2). A magnetic reed-switch sensor is positioned beneath the simulated tissue layer to detect the correct placement of the laryngoscope. The tip of the laryngoscope contains an adjustable magnet array which enables modulation of the system's positional tolerance and difficulty level. The silicone thickness and material composition of the manikin were selected to optimize both anatomical realism and sensor responsiveness.

To ensure procedural accuracy, two infrared break-beam sensors are embedded near the entrances of the trachea and esophagus. These sensors monitor the insertion path and depth of the airway tube. Correct placement through the trachea confirms successful maneuvering, while activation of the esophageal sensor signals incorrect insertion. Their positioning and orientation are carefully calibrated to reflect anatomical constraints and provide real-time feedback. The break-beam design ensures that only a physical obstruction such as an inserted airway tube can interrupt the beam and trigger a response.

2.1.4 Software Architecture

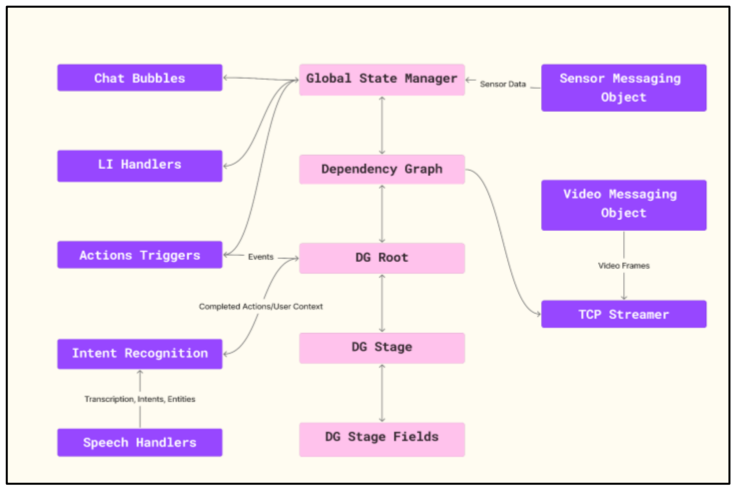

Figure 3. Software Architectural Flow of the Software in AR Airways.

The Unity application architecture consists of several key modules:

- **GlobalStateManager**: The primary orchestrator, initializing the simulation and managing inter-component communication.
- **DependencyGraph**: Constructs and maintains the procedure directed acyclic graph (DAG) using the "dependencies.json" configuration file.
- **VideoMessagingObject and TCPStreamer**: Handles WebSocket communication for laryngoscope video streaming, decoding JPEG frames and rendering them to Unity textures.

- **SensorMessagingObject**: Processes asynchronous sensor events from the manikin and triggers appropriate state transitions.
- **ActionTriggers**: Executes stage-specific callbacks and manages user interface (UI) updates based on simulation events.

2.1.4a. Dependency Graph

The AR Airways simulator employs a DAG to model the intubation procedure workflow (Figure 3). Each node represents a distinct procedural stage ("team prep," "equipment prep," "medication administration," etc.), with edges defining dependencies between stages. This architecture enforces clinical best practices by preventing users from attempting advanced procedures before completing prerequisites. Additionally, it mirrors reality by ensuring that users cannot repeat stages that have already been completed.

Within each stage, individual actions are classified into four priority levels:

- **Pass**: Mandatory actions required for stage completion
- **OK**: Recommended actions that improve performance metrics
- **Best**: Optimal actions demonstrating mastery
- **Critical**: Actions that trigger immediate failure if performed incorrectly

The system implements a stage-locking mechanism: once a user initiates actions in a downstream stage, all prerequisite stages become immutable, preventing retroactive modifications that could compromise patient safety.

2.1.4b. Natural Language Processing Pipeline

To advance through the dependency graph, user verbal cues are processed through the Natural Language Processing (NLP) Pipeline. Whether a verbal or a sensor-activated cue is necessary depends on the stage within the simulation. The specific-verbal cues align with industry best practices.

The system processes user utterances through a multi-stage NLP pipeline comprising of three stages:

- **Speech Recognition**: Meta's Wit.AI service transcribes audio input and performs preliminary intent/entity extraction based on pre-trained medical terminology models.
- **Intent Refinement**: OpenAI's GPT model receives the transcription, current simulation state, and available actions to perform context-aware intent recognition.
- **Action Mapping**: The refined intents are processed by the DGRoot controller, which validates actions against the current graph state and updates the simulation accordingly.

Speech Handling is implemented in the SpeechHandler component that serves as the entry point for voice interaction, managing speech capture and preprocessing. The component implements push-to-talk voice activation through Unity's Input System. A microphone animator provides visual feedback that indicates recording state.

Upon receiving a Wit.AI response, the handler extracts transcription text. Subsequently, it filters intents/entities by applying a confidence threshold of 0.8 to reduce false positives. Finally, it constructs a structured JSON payload containing raw transcription text, high-confidence intents with names and scores, and high-confidence entities with types and scores.

The intent refinement is implemented in the IntentRecognition component that processes user utterances through OpenAI's GPT-4.1 model. This serves as the second stage of the NLP pipeline after Wit.AI's preliminary transcription and entity extraction. The component implements a stateless architecture where each utterance is processed independently with explicit context in three parts: **Dynamic Intent Mapping**, which includes available actions based on current simulation state, **Completed Actions**, which includes previously satisfied intents to prevent duplicates, and **Wit.AI Preprocessing**, which provides initial intent/entity classifications with confidence scores.

This approach eliminates the need for conversational memory while maintaining contextual awareness for accurate intent resolution. This approach also allows for other benefits such as overlapping intent handling and structured output. First, overlapping intent handling recognizes when single utterances satisfy multiple intents (e.g., "I need two nurses" satisfies both "one nurse" and "multiple nurses" requirements), reflecting real medical scenarios with graduated performance levels. Then the structured output enforces JSON schema validation to guarantee consistent responses:

```
{
  "nurse_response": "string (≤70 characters)",
  "intents": ["matched_intent_1", "matched_intent_2"]
}
```

The 70-character limit ensures concise feedback that does not obstruct the AR view.

The component dynamically constructs GPT prompts that contain the current intent map, completed actions, and Wit.AI preprocessing data. It processes responses asynchronously through Unity coroutines and includes comprehensive error handling for network and parsing failures. The GlobalStateManager updates ChatBubble with the AI responses and advances the DependencyGraph with completed actions. This loose coupling enables independent testing and modification of the NLP pipeline without affecting core simulation logic.

2.1.4c. UI Handlers

The AR Airways system employs specialized interface handler components. GlobalStateManager invoke these handlers through method calls, creating a unidirectional data flow where the simulation state drives UI updates. The key components are:

- **VitalsPanelController**
- **ActionListController**
- **SuccessPanelController/FailedPanelController**
- **ChatBubble**

VitalsPanelController displays real-time patient vital signs (blood pressure, heart rate, SpO_2) with visual alerts for critical values. The dependency graph trigger updates when stage transitions affect patient status. In assisted mode, ActionListController renders available actions for the current simulation state. It queries the dependency graph to display the next possible actions, helping guide students through the procedure. The Success and Failure Panel controllers evaluate performance metrics including completion percentage by action type (pass/ok/best), time-based performance scores, and stage-specific feedback. Finally, ChatBubble manages the UI for AI assistant conversations, displaying user utterances and system responses with appropriate visual formatting.

This modular approach enables independent UI component updates without affecting core simulation logic, facilitating rapid iteration on visual feedback mechanisms based on user testing outcomes.

2.1.4d. Video Streaming

The video streaming subsystem enables real-time laryngoscope visualization through a two-component architecture that handles network communication and rendering separately. The VideoMessagingObject manages WebSocket communication with the Raspberry Pi server on port 8765. It implements **Asynchronous Connection Management** that establishes WebSocket connection with automatic retry logic for network resilience; **Thread-Safe Frame Queuing** using ConcurrentQueue<byte[]> to buffer incoming JPEG frames from the background WebSocket thread; and **Unity Thread Marshaling** that dispatches WebSocket messages on the main thread to comply with Unity's threading constraints.

The Raspberry Pi's video streaming service uses OpenCV to capture and encode JPEG frames from the laryngoscope and sends them to the component on the headset. The TCPStreamer is a rendering component that processes queued frames and manages visual display. It converts JPEG byte arrays to Unity Texture2D objects at a 360x180-pixel resolution. A Raw Image UI element is updated each Unity update cycle with the latest decoded frame, providing real-time display. A blood effect overlay implemented using a material shader simulates blood obstruction during the "obtaining a view" stage.

The separation between network handling and rendering ensures smooth video playback despite network jitter. The current implementation's direct dependency on the DependencyGraph object for blood effect state represents a coupling issue that could be resolved through an event-based architecture, improving modularity for future enhancements.

2.1.4e. Sensor Integration

The SensorMessagingObject component communicates with the Raspberry Pi via WebSocket connection on port 8766 and provides real-time sensor event handling. The component listens for three event types:

- **"epi"**: Magnetic sensor detecting correct laryngoscope blade positioning

- **"eso"**: Infrared beam break in esophagus (failure condition)
- **"trach"**: Infrared beam break in trachea (successful intubation)

Each sensor event is immediately routed to the corresponding handler in GlobalStateManager for validation and state updates. The Raspberry Pi implementation uses the gpiozero library's event-driven architecture, where sensor state changes trigger immediate WebSocket message transmission. This event-based approach ensures minimal latency between physical sensor activation and simulation response, which is critical for maintaining immersion during the intubation procedure. The component includes automatic reconnection logic and proper cleanup on application termination, ensuring reliable sensor communication throughout extended training sessions.

3 | RESULTS

Initial evaluations demonstrated that users of the device, both non-medical personnel at the 2024 DALI exhibition and individuals with Emergency Medicine backgrounds at the 2025 Society for Academic Emergency Medicine (SAEM) Innovation exhibit found the device educational in terms of difficult airway management. However, challenges with stable connectivity emerged under different network conditions.

Selected user comments from SAEM attendees-including medical students, emergency medicine residents, and domestic and international clinicians-include:

“I loved the feedback videos telling me what I needed to do to improve as a proceduralist.”

“I always feel nervous before intubations but by having the high-fidelity simulation tool I can practice this high acuity low occurrence procedures and feel ready for it in real life.”

“A lot of simulation can be boring and mundane, but this is exciting and engaging. The WiFi issues are a little fickle, but I can imagine once the connectivity issues are fixed, this is almost like playing a video game.”

“I was sweating, the scenario seemed so real with the audio and visuals. I’m glad that the feedback videos are integrated into the difficult airway algorithm to tell me how I can improve my technique.”

The system includes a prompt mode toggle that displays example phrases for available actions, serving as a training aid for novice users. When enabled, the ActionListController queries the dependency graph for the next five available actions and displays their example utterances (e.g., "May I have a nurse?") in the AR interface. This mode helps instructors guide students through the procedure while maintaining voice-based interaction as the primary control mechanism.

4 | DISCUSSION

Future work will assess the efficacy of this prompt mode as a pedagogical tool for intubation training. Especially for high acuity, low occurrence events, simulation in EMS training is crucial for skill development and retention (10, 11). Paramedic students in rural New Hampshire will be provided with a survey on simulation intubation prior to attempting the novel simulation device. The survey will determine the level of comfort that new airway operators have with managing a bloody airway. Military studies have shown that ascertaining the realism of the device will impart critical information about its long-term educational value in medical simulation (12-14). The 12-stage intubation procedure follows established emergency medicine protocols where stage dependencies reflect critical safety requirements. For example, "medication administration" depends on "procedure preparation" to ensure that proper drug selection and dosing plans are established before administering paralytics that render the patient unable to breathe independently. Similarly, "obtaining a view" requires both "medication administration" and "patient hypotensive" stages to be complete, as attempting laryngoscopy before adequate sedation risks patient injury, while unmanaged hypotension during the procedure can lead to cardiac arrest. These dependencies encode best practices that prevent life-threatening errors during emergency airway management. Throughout the simulation, users can fail and receive feedback videos depending on their stage in the dependency graph and whether incorrect actions were performed.

In conclusion, we developed a high fidelity, transportable difficult airway simulator that trainees may use on standard airway manikins that is novel in 3 primary aspects: (1) We simulate the management of a blood-contaminated airway using a headset that projects the scenario in AR onto a standard manikin. The manikin is equipped with mobile sensors, providing immersive visual and tactile feedback in a highly portable setup. (2) We have programmed into the simulation the current societal guidelines regarding procedural techniques to teach the four phases of a successful difficult intubation: preparation, environment management, procedure and post-procedure care. (3) Lastly, integrated feedback videos automatically play within the headset at each phase if pre-set criteria are not met, providing real-time guidance and reinforcement. Our formal testing is on-going, but reception at two academic conferences has been positive. While many large medical centers have access to high fidelity simulation, our pilot bring this vital education to austere environments, and the initial testing of our device with rural paramedics received uniformly positive reviews for the educational experience.

4.1 Future Work and Limitations

Current limitations include dependency on consistent network connectivity for NLP services and the requirement for users to use push-to-talk for voice commands. The system's reliance on cloud-based LLMs also introduces latency (typically 1-2 seconds) that could be reduced through model optimization or local deployment.

Further limitations include the absence of a control group. However, as this is a pilot device and with the field of AI and AR rapidly evolving it's very challenging to find comparative simulation devices.

ACKNOWLEDGEMENTS

We acknowledge the assistance of countless of Dartmouth students who rotated through the DALI lab since 2022 when this project was first started by Dr. Shukla, who learned, grew and provided their valuable insight towards this vision and helped overcome multiple hurdles and challenges.

SOURCES OF FUNDING

This research was supported by a Pilot Research Grant 250-4151 from the Hitchcock Foundation in addition to an Augmented Reality in Healthcare Innovation Grant from the DALI Lab at the Magnuson Center at Dartmouth College. The associated testing protocol was reviewed and approved by the Dartmouth Hitchcock Institutional Review Board.

AUTHORSHIP

BK wrote the first draft of the article with contributions from AS and was the lead computer scientist. BK, TO, HC, JB helped design the device over many semesters of work at the DALI lab under the mentorship of TT. TT provided continued mentorship and funding for the project. AS was the principal investigator who led and helped design the device, obtained grant funding, and led the project for many years. HG and MR both provided mentorship to AS in regard to the clinical implications. HG is currently leading the project. SL has assisted in developing the testing regimens. All reviewed the publication and provided valuable insight into the publication.

CONFLICT OF INTEREST DISCLOSURE

(MAJ) Anant Shukla, MD FS CAQSM: The views expressed represent those of the authors and do not represent the views of the US Army Medical Center of Excellence, US Army, the Department of War or the US Government.
No other authors claim any other possibly conflicts of interest.

REFERENCES

1. Wang, H. E., Balasubramani, G. K., Cook, L. J., Lave, J. R., & Yealy, D. M. (2010). Out-of-hospital endotracheal intubation experience and patient outcomes. *Annals of Emergency Medicine*, *55*(6), 527–537. https://doi.org/10.1016/j.annemergmed.2009.12.016
2. Patel, S. H., Rudolf, F., Schwartz, K., Gabriel, R. A., Hastings, R. H., Daniel, M., & Suresh, P. J. (2024). Enhancing laryngoscopy mastery: The impact of autonomous practice with feedback-providing simulators. *A&A Practice*, *18*(7), e01825. https://doi.org/10.1213/XAA.0000000000001825
3. Fei, G. T., Guo, A. W., Booth, T., Kiss, E., Hallac, R. R., Anderson, A., & Wang, Z. J. (2024). Development of an augmented reality system for tracheal intubation guidance of airway management. *Proceedings of SPIE, The International Society for Optical Engineering*, *12818*, 1281808. https://doi.org/10.1117/12.3005990
4. Steffensen, T. L., Bartnes, B., Fuglstad, M. L., Auflem, M., & Steinert, M. (2023). Playing the pipes: Acoustic sensing and machine learning for performance feedback during endotracheal intubation

simulation. *Frontiers in Robotics and AI*, *10*, 1218174. https://doi.org/10.3389/frobt.2023.1218174
5. Miller, M. E., Witte, D., Lina, I., Walsh, J., Rameau, A., & Bhatti, N. I. (2025). Development of machine learning copilot to assist novices in learning flexible laryngoscopy. *The Laryngoscope*, *135*(3), 1046–1053. https://doi.org/10.1002/lary.31812
6. Dias, P. L., Greenberg, R. G., Goldberg, R. N., Fisher, K., & Tanaka, D. T. (2021). Augmented reality-assisted video laryngoscopy and simulated neonatal intubations: A pilot study. *Pediatrics*, *147*(3), e2020027228. https://doi.org/10.1542/peds.2020-027228
7. Zhao, S., Xiao, X., Wang, Q., Zhang, X., Li, W., Soghier, L., & Hahn, J. (2020, November). An intelligent augmented reality training framework for neonatal endotracheal intubation. *2020 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, 672–681. https://doi.org/10.1109/ISMAR50242.2020.00092
8. Jain, S., Barua Chowdhury, B. D., Mosier, J. M., Subbian, V., Hughes, K., & Son, Y. J. (2025). Design and development of an integrated virtual reality (VR)-based training system for difficult airway management. *IEEE Journal of Translational Engineering in Health and Medicine*, *13*, 49–60. https://doi.org/10.1109/JTEHM.2024.3490989
9. Duffy, C. C., Bass, G. A., Yi, W., Rouhi, A., Kaplan, L. J., & O'Sullivan, E. (2024). Teaching airway management using virtual reality: A scoping review. *Anesthesia & Analgesia*, *138*(4), 782–793. https://doi.org/10.1213/ANE.0000000000006733
10. Green, A., & Hug, M. (2023). Simulation training and skill assessment in EMS. In *StatPearls*. StatPearls Publishing. https://www.ncbi.nlm.nih.gov/books/NBK570595/
11. Davis, L., Ha, Y., Frolich, S., Martin, G., Meyer, C., Pettitt, B., Norfleet, J., Lin, K. C., & Rolland, J. P. (2002). Augmented reality and training for airway management procedures. *Studies in Health Technology and Informatics*, *85*, 121–126.
12. Petersen, C., Rush, S. C., Gallo, I., Dalere, B., Staak, B., Moore, L., Kerr, W., Chandler, M., & Smith, W. (2017). Optimization of simulation and moulage in military-related medical training. *Journal of Special Operations Medicine*, *17*(3), 74–80.
13. Stone, R. J., Guest, R., Mahoney, P., Mluduch, M., & Carr, K. (2017). A 'mixed reality' simulator concept for future medical emergency response team training. *BMJ Military Health*, *163*(4), 280–287. https://doi.org/10.1136/jramc-2017-000783
14. Rabotin, A., Glick, Y., Gelman, R., Ketko, I., Taran, B., Fink, N., & Furer, A. (2023). Practicing emergency medicine in the metaverse: A novel mixed reality casualty care training platform. *Surgical Innovation*, *30*(5), 586–594. https://doi.org/10.1177/15533506231174092