Capstone Design Abstracts

Spring 2026

Stringless Guitar

Team Members: Olivia Peters Van Aalst, Justin Bosland, Laurel Brown, Defne Ulusoy, Jack Yu

Advisor: Professor Tom Consi

Abstract

Traditional acoustic guitars present physical challenges to beginners due to the strain of pressing metal strings while learning to play chords. This barrier to entry can deter individuals looking to learn a new skill and limits consistent practice, a critical element for skill development. This project's objective was to develop a stringless guitar that is less physically demanding to play, thus making it a more enjoyable and approachable experience.

The guitar system features five long flexible potentiometers, with each representing a string, a force-sensitive resistor (FSR) along the guitar neck to receive player inputs, and a strum bar to trigger audio output. A switch is also used to toggle between single string playing and chord playing. The position of a finger press is determined from the potentiometers and converted into chords, while the FSR ensures the press strength is great enough to produce an appropriate sound on a real guitar. The sensor inputs are processed through an Arduino Mega 2560 microcontroller and transmitted to a second microcontroller, a Teensy 4.0, where they are translated into audio output. Sound is produced through an audio shield with a headphone jack. The Arduino takes eight total inputs from the guitar: five analog inputs from the potentiometers along the neck, one analog from the FSR, one analog from the strum bar, and one digital from the switch. The Arduino packages the potentiometer and FSR readings together into MIDI note standard format with a start bit and checksum included. When the strum bar is strummed, the package is active and sent via the transmission port to the Teensy. This allows the Teensy to convert MIDI notes to frequency using the audio shield DAC through the I2S audio sample digital data stream protocol which efficiently compresses audio data. The guitar is fully operational without the use of an external computer program, allowing users to play on the move if they choose. A rechargeable battery powers the hardware, providing 12 hours of playing time, and the headphone output allows practice without additional amplification or disturbing others.

Testing of the guitar demonstrated accurate detection of finger positions and strum actions, with minimal lag time between the inputs and the audio output, providing a responsive playing experience. The system's portable and self-contained design enabled use in many different environments. With future expansion, incorporating the MIDI standard protocol, our device could be used as MIDI controller when paired with existing educational tools, which provide detailed lessons and feedback to improve the guitar learning process.

Ambulance Inventory Management System

Team Members: Ryan Woo, Dhruv Dingari, Thomas King, Anastasia Lindquist, Neirit Mahabub, Ethan Ostroff

Advisor: Professor Canek Fuentes

Abstract

The team's topic of choice for their senior capstone project is an Ambulance Inventory Management System. Most services place the sole responsibility of making sure the ambulance is well-stocked on the EMS clinician, who often as a result of either intentional or unintentional negligence, may fail even realize an area of an ambulance is understocked. This poses the risk of responding to an emergency without vital equipment that may be necessary for patient care. The team's system aims to solve these problems by creating an easily integrable and interfaceable system powered by load cell and GPS data processing that takes the onus of inventory management from the EMS clinician, ensuring that emergency medical service ambulances are consistently stocked and fully equipped for every call.

The mechanism of measurement for the Ambulance Inventory Management System is the load cell, which functions to detect variation in force up to a one-thousandth of a gram. Each rectangular cabinet in the back of an ambulance contains two load cells, upon which a baseplate is mounted that acts as the bottom surface of the cabinet. The load cells are connected to HX711's, which amplify the small differential analog signals from each load cell. The HX711's send information to an ESP32, which reads the values from both devices and processes them together to get a single force value. This value is sent to an MCP2515, which acts as a combined CAN controller and receiver board, and broadcasts the messages it receives to a Raspberry Pi5 computer. The Raspberry Pi5 reads the data from the CAN bus and compiles the force data from the different sensors into a database that serves as the foundation for the GUI. This GUI is displayed on a touchscreen panel that is mounted on the left-side interior wall of the ambulance patient compartment and displays all relevant cabinet metrics including which items are in stock, which items are out of stock, and how many of each item are currently in each cabinet. The system is conditionally powered based on information from a GPS module and has been designed to only stay activate when the ambulance location is at either a hospital or an ambulance base, so as to minimize distracting the EMS clinician while they are actively delivering patient care.

For testing purposes, the team built a minimal functional system with a total of six load cells and the requisite data processing hardware to simulate the system applied to three separate ambulance cabinets. During testing the team worked closely with university EMS teaching faculty, and administrative representatives from Cataldo Ambulance Service Inc., who supplied the team with real EMS equipment to use for testing and allowed the team to set up the system inside of a real ambulance to ultimately verify full, contextual functionality. Feedback from university EMS faculty and Cataldo Ambulance representatives during testing emphasized a greater focus on supply chain integration, such as automatically sending a request to order a new piece of equipment as soon as one is used. These features lie beyond the project's current scope but would be easily integrable with a few software modifications.

Dalmatians

Team Members: Rebecca Dozortsev, Matthew Baker, Tyler Von Pervieux, Kaitlyn Hu, Augustine Nguyen, Jason Yang

Advisor: Professor Bahram Shafai

Abstract

Our team working on Dalmatians has created a set of three voice-controlled multi-agent robotic cars with a speech pathology application. The goal is to aid kids in mastering full sentences, grammatical tools, and difficult sounds. For example, a child can ask the robots to make a circle, and all three will move in a circular motion. Our program includes various shapes, the ability to individually address robots, and driving features that require commands using prepositions, pronouns, and plurals. Ultimately, by making speech pathology more engaging, these robots help motivate children to keep learning while providing feedback that encourages improvement.

The software implemented on the cars can be broken down into three categories: voice processing, spatial mapping, and communication between cars. The voice processing uses an AI model called Vosk to integrate speech-to-command functionality, converting spoken audio into text in real time while working offline for privacy and reliability. The code outputs commands to the prompt for troubleshooting and also drives the feedback system displayed on an LCD screen. For spatial mapping, the goal is to model the physical world digitally using sensor data. Each Dalmatian uses an RGB camera to estimate position and pose, along with feature extraction for depth, while a binary occupancy grid stores obstacles. A front-facing ultrasonic sensor provides rough depth sensing and emergency detection. Finally, we use the MQTT for communication between cars, enabling devices to publish messages to a central broker that distributes them to subscribed robots for coordinated, bidirectional communication.

We also implement key hardware components. Using the SunFounder robotic car kit, we utilize a Raspberry Pi to communicate with the cars. Additionally, we designed a PCB board to regulate power distribution to different elements of the car, including the LCD screen. This PCB is centered around a buck converter that uses supporting components to efficiently distribute power.

For testing and validation, we assess each subsystem independently and then as a fully integrated multi-agent system. This includes measuring command accuracy and latency, evaluating mapping and localization error, and testing communication reliability over MQTT. The PCB is validated for voltage stability, efficiency, and thermal performance. Finally, system-level tests focus on coordinated multi-robot tasks and include children interacting with the cars to ensure effectiveness in real-world use.

Heat Seekers (Wildfire Detection System)

Team Members: Jani Passas, Dani McLaughlin, Fausta Fenner, Spandana Machavarapu, Abhiram Pulavarthi, Sarah Ninan

Advisor: Professor Tom Consi

Abstract

Wildfires pose a growing threat to ecosystems, communities, and infrastructure, driven by climate change and increasingly dry environmental conditions. The Heat Seeker project proposes to combine ground sensor modules with drone support to create a low cost, energy efficient, and scalable wildfire detection system.Each sensor module is mounted to a tree and collects environmental data using a thermal camera alongside gas, temperature, and humidity sensors. To increase detection sensitivity, the modules are equipped with fans to direct airflow towards the sensors. Additionally, the thermal camera is actuated by a servo enabling it to capture four frames which create a panoramic image. The sensor suite interfaces with an STM32 development board which is integrated as a daughter board onto a custom designed PCB. The modules package and transmit sensor data wirelessly to a base station using LoRa mesh (Long Range) communication protocol.

The modules are powered by a custom built 3s2p battery pack, sized to provide sufficient power. The system also employs a solar panel so the batteries remain charged. A charge controller integrated circuit (IC) regulates input power for safe battery charging, implementing maximum power point tracking (MPPT) via a digital potentiometer during solar operation. A battery management IC monitors individual cells passively balances cells, while fault and temperature detection across both ICs provide automatic shutdown. Power distribution includes 5V and 3.3V DC-DC converters which supply the STM32 development board and peripherals. All voltage rails feature EFuse protection (over/under voltage, current limiting and monitoring) and ADC voltage monitoring for comprehensive system safeguarding. To maximize energy generated from the solar panel, the system employs a single-axis solar tracking system. A high-torque worm-gear motor rotates the panel to synchronize with the earth's rotation with the sun. This ensures a sufficient state of charge on the battery pack.

Once data is transmitted to a base station, readings are stored in a MongoDB database and the frontend UI displays all the collected data. Modules are shown on a map, and users can click into specific modules to see live readings. A dashboard view also allows users to quickly see modules that have been flagged as detecting a fire, a temperature and humidity progression throughout the day based on readings, and average humidity and temperature in the area. A confidence score between 1 to 5 is calculated using a fuzzy logic approach, which combines maximum temperature readings from the thermal camera, as well as gas readings and humidity values. During stage two of detection, the drone flies to the sensor pod with the highest confidence score, given that it exceeds the threshold of 3, and streams RGB images back to the base station, which are fed into a Convolutional Neural Network trained to detect forest fire.

We tested each part of our system individually and then performed a full system test beside a bonfire to determine the accuracy of our sensors in detecting a wildfire. We also tested in a field to fly the drone and determine the range of our LoRa transmissions. This testing showed that our hardware and software work together to create a wildfire detection system.

CTHULHU: Cancer-Targeting Hyperspectral Ultraprecise LASER Handling Unit

Team Members: Hazel Bershatsky, James Brennan, Alexi Carlone, Matt Geisel, Iris Kaucher, Julia Keadey

Advisor: Professor Canek Fuentes

Faculty Support: Gunar Schirner, Bryan Spring, Kai Zhang

Abstract

Our project was developed for and with two Northeastern labs: Spring Lab, under the College of Science and the Embedded Systems Lab (ESL), under the College of Engineering. These labs aim to develop a biomedical imaging system to identify cancer in the human brain with cell-level resolution. The CTHULHU team has designed and built a system to drive a 32MHz bio-imaging laser and interpret the received data from its pulses. Existing solutions for high-bandwidth biomedical imaging either involve inferior data processing equipment or significantly higher costs; CTHULHU accomplishes more than was previously possible at a much lower price point (and with significantly less complexity, requiring only an off-the-shelf FPGA development board and one PCB).

At a high level, the architecture of our solution has three main parts: electronics, FPGA design, and software. Control flow starts from the software, where the user selects an imaging pattern and other parameters. This information flows to the FPGA, which calculates the required inputs to move the laser to specific positions in the field of view of the sample. At each position, the laser pulses (in 32 different subfrequencies) bounce off the desired cell and hit a silicon photomultiplier array (SiPM), which converts them into electrical signals. The electrical signals are handled within our own custom 6-layer PCB, designed to handle the processing and digitization of these raw signals. Once acquired, those signals are amplified by an array of fully differential amplifiers (FDAs) before being digitized by an ADC and fed back to the FPGA. The FPGA receives 32 channels' worth of data from each sample, performs further processing, and dispatches the data back to the host PC over a PCI-Express bus. There, the firmware receives the data and converts it into a format that can be rendered by a UI, which was initially designed by the ESL and modified to be compatible with this project.

Initially, this system was envisioned to be used specifically in the field of brain cancer imaging and treatment: existing systems suffer from issues of low data resolution and possible side effects of treatment, while a laser-based system can accomplish both imaging and treatment tasks with much greater precision and safety. However, over the course of the project's development, it became apparent that CTHULHU can be applied to a wide range of biomedical applications, allowing for an increased channel count for a cheaper price than what is otherwise available. In addition, the accessible design of the hardware means that this project's work can easily be adapted to suit a wide range of potential experiments, especially when compared to the existing costly, closed-source solutions.

Helping Hand: An Assistive Robotic Platform with Vision Control

Team Members: Mark Yang, Huaqing Wang, Riley Intihar, Krish Gupta, Nicole Contreras, Henry Gao

Advisor: Professor Bahram Shafai

Abstract

Millions of individuals worldwide suffer from profound motor impairments, with quadriplegia alone accounting for nearly 60% of all spinal cord injuries. For this population, the total loss of upper limb mobility creates a devastating barrier to independence, rendering even the most basic activities of daily living impossible without constant assistance. To address this critical loss of autonomy, this project introduces a vision-controlled robotic platform designed to restore agency through autonomous object retrieval. By integrating a motorized robotic arm with a custom eye-tracking interface and voice command module, the system translates a user's gaze or spoken input into precise control signals, bridging the gap between human intent and physical action.

To achieve the design goal, three components were designed. A glass frame was 3D printed and fitted with two CSI cameras, one facing outward toward the environment and one directed toward the user. The back camera tracks visible AprilTags in real time. The front-facing camera performs pupil-based gaze estimation with predefined regions using image processing. The software utilizes polynomial fitting based on recorded data from the calibration process to map the pupil locations to target gaze coordinates. Once the software detects 3 blinking motions in 5 seconds, it will send a ROS message to communicate with the robotic arm.

The project uses open-source ROS2 packages to control a Roarm-M3 Pro robotic arm. An end-effector-mounted camera localizes the target AprilTag, using TF trees to translate the pose into the world coordinate frame. Visual servoing then guides the arm to grasp and deliver the object.

As the system is intended to be portable and mounted to a wheelchair, the robotic arm and Jetson Orin Nano require a portable power source. To address this, a custom three-stage power conversion and delivery board was developed. This board charges a 12V 9Ah SLA battery and provides up to 12V, 6A of output. It accepts 120V AC mains, which passes through an EMI stage and rectification to 160-170V DC, before being stepped down to 14.4V via a custom DCM flyback converter. A buck-boost stage ensures a stable 12V output regardless of the battery's charge state, supplying both the arm and the computer. The system supports two input methods — eye-tracking with blink detection and voice commands — making it accessible to a broader range of users.

Testing showed the system is functional, successfully detecting pupils, responding to blinks, and retrieving book-shaped objects. However, it lacked precision and consistency, with pupil detection occasionally drifting. This limitation likely stems from the hardware; while conventional cameras were used, industry-standard infrared sensors would offer better precision across various lighting conditions. Additionally, limited calibration data occasionally caused gaze fluctuations despite the user focusing on a target.

Robotic Acoustic Identification for Disaster Emergency Response (RAIDER)

Team Members: Ritika Gurjar, Anna Valades, Casey Goyette, Ester Vicente, Grace Watson, and Madison O'Neil

Advisor: Professor Tom Consi

Abstract

The RAIDER team has built a proof-of-concept swarm robotics system that uses ground propagated vibration localization to detect trapped individuals, helping first responders locate survivors without entering hazardous environments. A fleet of small autonomous robots capable of localizing vibrations and voices could accelerate identification, and reduce the risk to human rescuers. Due to time and budget constraints and to allow for repeatable tests, the system was tested on a flat 4x8 ft drywall board, rather than mimicking collapsed terrain.

The system consists of three robot cars deployed on the board. Each robot is a four-wheeled chassis driven by an ESP32 microcontroller and equipped with an Inertial Measurement Unit (IMU) for detecting the vibration signals. The vibration source is a thumper placed on the board, driven by an amplifier and a MATLAB script to mimic a knock from a trapped person. The positions of the robots on the board are measured using the DW1000 Ultra Wideband chip (UWB), located on each microcontroller. Four additional modules are placed around the board to act as anchors to measure the position data.

The accelerometer and position data are sent from the IMUs and UWBs to the central computer. The UWB modules and the central computer are on the same Wifi network, and a Raspberry Pi is used as a broker for a Message Queuing Telemetry Transport (MQTT) server. A MATLAB script running on the central computer matches the vibration events across the robots, computes the Time Difference of Arrival (TDOA) values for each of the robot pairs, and feeds those measurements into a Levenberg-Marquardt nonlinear least-squares optimizer to triangulate the positions on the board. The estimated position and true thumper location are plotted on a map of the testing environment, allowing direct visual assessment of localization accuracy. The robots are constrained to bounding boxes on the drywall and move randomly to a new position within their box to generate several data points.

The system was validated through a series of controlled experiments with the thumper placed at known positions across the board and robots arranged in triangular formations to ensure geometric diversity in TDOA measurements. Results demonstrate accurate localization of the thumper. This project establishes a functional proof-of-concept for swarm-based seismic localization and lays the groundwork for future work including multi-source detection and deployment on more complex terrain.

RepSense

Team Members: Alex Wise, Aaron Xu, Connor McEleney, Maria Clara Porto, Thomas Glassman, Zachary Klusky

Advisor: Professor Canek Fuentes

Abstract

RepSense is a platform to support feedback for wellness and fitness activities through a range of sensors that inform and empower its user. RepSense utilizes a combination of gyroscope, accelerometer, and electromyography (EMG) data to provide a user with guidance when identifying improper form and misalignment as well as muscle imbalance. This feedback can prevent injury due to poor form and overexertion, respectively. By analyzing the correlation between form and muscle fatigue, the user can take advantage of real-time feedback to plan better, prevent injury, and lift smarter.

The RepSense hardware design accounts for a three printed circuit board split between two Barbell Sensing Units (BSU) and a Wearable Processing Unit (WPU). A BSU board is placed on each side of the barbell to measure accurate position, velocity and acceleration data to inform the user's form based on triangulation performed at a later stage. The BSU is housed within an enclosure that clamps to the end of a barbell. These boards forward the data acquired over Bluetooth Low Energy to the WPU. The WPU is housed in a similar enclosure featuring an adjustable, elastic band for personal fitting to the user's arm. To capture EMG data, two electrodes must be placed and wired from each muscle to the board, along with a reference pad.

A mobile application functions as a channel to visualize, interact, and collect the data that comes from the product. By establishing a persistent BLE connection with the WPU, the user can record active lifting sessions. Data is initially filtered on chip prior to being forwarding to the frontend. Upon receiving data, the frontend posts the data to the backend using an api. The backend runs a machine learning algorithm that performs continuous analysis of exercises to provide immediate feedback regarding the correlation between muscle fatigue and deterioration of form.

EMG and IMU data was tested and validated separately in order to assess individual components' accuracy. The initial EMG data was collected using the Cyton Biosensing Board provided by OpenBCI. The data collected aligned with our initial hypothesis where muscle activation will increase in tandem with reps, and starting point muscle activation increased throughout the workout. IMU data was pulled initially from Kaggle datasets to understand the mechanics behind barbell exercises. Kaggle data was then compared to that collected on the devkit and eventually using the BSU.

By combining wearable sensors, barbell-mounted hardware, and an AI-based software interface, RepSense empowers athletes and coaches with actionable insights to optimize performance and reduce injury risk.

Dognosis

Team Members: Francesca Cambareri, Brian Chung, Serena Lin, Cameron Moloney, John Roche, Gregg Vignaux

Advisor: Professor Bahram Shafai

Abstract

The Dognosis team has designed and built a health tracking harness for pet dogs. The Dognosis harness has functions akin to commercial fitness trackers for humans and is kitted with the ability to track a dog's heart rate, temperature, activity, and step count. Dog ownership spans across numerous cultures and countries, with over 65 million US households owning pet dogs and approximately 33% of homes worldwide having at least 1 dog. Being able to have constant monitoring of these vitals could be game changing for dog owners whose busy lifestyles would normally make it difficult to keep such a close watch of these stats. In addition, as dogs cannot easily communicate discomfort, and most have infrequent vet visits, monitoring this information and flagging potential health threats could greatly benefit the health and comfort of a pet, as well as ease the nerves of its owners.

To use the Dognosis harness, the owner secures the harness using the plastic clips and adjusts the straps so that it fits snugly and safely. Once powered on, embedded sensors start collecting heart rate, activity, and temperature data which are transmitted to the Dognosis web platform in real time. On the Status page, the dog's current heart rate, step count, and temperature are shown, as well as a graph of this data over time. Additional insights can also be found on the Flags and Incidents page, where events are automatically logged when readings exceed predefined thresholds or unusual patterns are detected over time. This event-based view helps owners distinguish between normal activity and potentially concerning episodes that may require veterinarian attention. This page allows for greater detail to be seen surrounding these incidents, with time as well as all sensor data shown for the 15 minutes before and after the incident occurred. The platform is designed to provide live visibility as well as historical context, allowing owners to track changes in their dog's condition over time. The Dog Profile page on the GUI allows for an increased aspect of customization for not just the appearance of the site, but for the insights as well. It has been found that heart rate, activity, and temperature averages can all vary based on a dog's breed, age, and size, and therefore by taking in this information we are able to account for this and adjust thresholds to be more accurate to the specific dog.

For testing and validation, the system was evaluated through controlled and real-world scenarios. Sensors were integrated into the harness and tested on device to ensure reliable data collection and transmission. Functional testing verified accurate sensor readings, stable communication with the web platform, and correct event flagging under simulated abnormal conditions. Additional trials were conducted to assess system performance during typical dog activity, including walking and resting, to validate consistency and robustness. Results demonstrated that the system could reliably capture and visualize real-time data while successfully identifying anomalous events. Overall, the Dognosis system provides a low-cost, portable, and scalable solution for continuous canine health monitoring. By combining embedded sensing, real-time data processing, and an intuitive user interface, the platform enables early detection of potential health issues and empowers owners with actionable insights into their pet's well-being.

GestAir: Gesture Operated Drone

Team Members: Adora Chen, Liam McCarthy, Olivia Lazar, Michael Veteri, Nicholas Sirota, Rachel Haynes

Advisor: Professor Tom Consi

Abstract

Unmanned aerial vehicles (UAVs), commonly known as drones, are widely used for adaptable aerial surveillance in research and safety applications, including environmental monitoring and search-and-rescue operations. However, conventional control methods, such as a handheld transmitter with joysticks, have steep learning curves and occupy the operator's hands, which are often needed for other tasks on the ground. GestAir presents the design and implementation of a gesture-based control system that enables a more intuitive, hands-free method of piloting a quadcopter drone through a wearable glove interface.

The system is composed of three primary components: a sensor-equipped glove, a base station, and a custom-built quadcopter. The glove integrates two inertial measurement units (IMUs), one mounted on the back of the hand and the other on the upper forearm, to capture hand orientation relative to the arm. A set of fingertip electrodes detects finger contacts, with the thumb powered and the remaining fingers acting as passive contacts, each mapped to different controls. When the thumb contacts the index finger, deviation from a calibrated neutral hand position is mapped to the drone's roll, pitch, and yaw inputs; when it contacts the middle finger, it is mapped to the drone's throttle (altitude). Contact with the ring finger is currently unused but can be mapped to control a mounted camera's orientation or other auxiliary functions. When no finger contact is detected, the drone hovers in the air, allowing the operator's hand to remain free for other tasks. Contact with the pinky finger toggles a menu mode, where finger contacts map to accept, up, down, and back for configuration control via a liquid crystal display (LCD) on the lower forearm.

The glove's onboard ESP32-C6 microcontroller continuously reads accelerometer, gyroscope, and magnetometer data from the two IMUs to compute a relative orientation estimate with minimal drift. This estimate is processed through configuration-based thresholding and scaling, while analog electrode readings determine the active control mode, mapping the resulting motion to the appropriate drone control inputs and producing normalized commands. These commands are transmitted over WiFi to a base station, where a multi-threaded application converts them into MAVLink attitude setpoints using the MAVSDK library. The resulting command packets are relayed to the drone's PixHawk 2.4.8 flight controller over WiFi via an ESP8266 module connected through a Universal Asynchronous Receiver-Transmitter (UART) interface. The flight controller interprets the MAVLink messages and generates corresponding pulse width modulation (PWM) signals to regulate motor speeds, enabling responsive and stable flight control.

PosturePerfect

Team Members: Victor She, Maxene Aguinaldo, Paul Wisnaskas, Sriya Lanka, Abdu Assaf, Kyle Andrews

Advisor: Professor Canek Fuentes

Abstract

The PosturePerfect team has designed and built a system for real-time posture monitoring that can help users maintain healthy sitting habits while working or studying. Poor posture is a widespread issue in modern society, particularly among students and office workers who spend a long period of time sitting at desks. Prolonged sitting and improper spinal alignment can lead to neck pain, back pain, and other musculoskeletal issues over time. Current solutions in the market such as ergonomic chairs or posture braces can be expensive, uncomfortable, or ineffective at actively correcting posture. PosturePerfect proposes to address these issues using non-invasive chair cover system that monitors sitting posture and provides real-time feedback to help users monitor and correct poor posture before it develops into a long-term health issue.

PosturePerfect comes in the form of a padded seat mat with sensors embedded into a layer of foam. The seat bottom contains four load cells housed in specialized 3D printed clamping containers. The load cells measure the weight of the user in four separate regions of the seat. The differential voltage from each load cell is fed into the HX711, a combined amplifier and 24-bit analog to digital converter designed for load cell integration. The digital weight data output from the HX711 is processed using an Arduino Mega microcontroller. The resulting user data sent via Bluetooth through an HM-10, which utilizes Bluetooth BLE 4.0, also connected to the Arduino Mega. On the user interface side, we have an electron desktop application to connect with the Bluetooth module and handle all the data parsing and posture logic. It uses Electron with React for the frontend and Node.js paired with MongoDB for backend logic. The application reads the users' pressure on the seat every second and stores the average pressure over a minute to be used in the logic for determining bad or good posture.

In its current form, PosturePerfect users can detect the weight of the user in four separate sections, which is used to determine the posture of the user by calculating the ratio of weight sensed by one load sensor compared to others. Notifications for the users calculated posture is sent to their laptop and they can see a real time heatmap of their pressure on the seat along with a graph of their posture over the past hour, day or week. Sensor noise from the load cells and amplifier are mostly negligible. Current iterations only exist for the seat bottom and lacks back support.

CourtIQ: Intelligent Basketball Training Device

Team Members: William Ferros, Max Langweil, Niki Manolis, Jordan Monsen, Alex Nikolov, Sonia Sheth

Advisor: Professor Bahram Shafai

Abstract

The CourtIQ team has designed and developed a smart basketball feeding system that leverages computer vision (CV) to track player movement and dynamically control ball delivery. The device is intended for basketball players seeking effective solo shooting training in both indoor and outdoor environments with access to power outlets. Many players lack access to consistent training partners or coaches, limiting the efficiency and repetition of independent practice. Without a coach, they are unable to recreate game-realistic practice drills to most effectively train. CourtIQ addresses this challenge by creating a dynamic, responsive training partner for ultimate solo skill development.

To use CourtIQ, at the start of a session, the user inputs their height via a user interface and initiates the system. When the player runs onto the court, the onboard CV algorithm then detects and tracks the player in real time, continuously analyzing their position on the court. As the player moves, the system adjusts its orientation using a servo motor to maintain alignment with the player. When the player signals that they are ready for a ball, by stopping and extending their hands out in front of them, the system recognizes this position, calculates the player's distance from the device, and dynamically adjusts motor speed to shoot the basketball with appropriate force and trajectory to the player. To ensure player safety, the system incorporates safeguards such as preventing ball launch when the player is too close to the device. At the end of a session, the player can stop the system via the user interface and view performance metrics, including the total number of shots taken, session length, and a bird's-eye visualization of shot locations across the court.

For testing and validation, the CourtIQ system was evaluated on both indoor and outdoor courts near Northeastern University's campus, demonstrating reliable tracking and consistent ball delivery across different environments. The camera and CV tracking were tested for players of different heights and at various angles and distances to ensure the ready position could be accurately identified. Due to budget constraints limiting choices, the power supply and servo motors had difficulties handling multiple consecutive shots without hitting voltage or torque limits. The rotating platform was tested across the entire 180 degree range of rotation to ensure ball delivery at every point on the court. By reducing downtime between shots and enabling continuous, adaptive practice, CourtIQ improves training efficiency. This system provides a practical and intelligent solution for athletes aiming to enhance their shooting performance through independent training.

Smart Walker

Team Members: Justin Bonapace, Owen Chun, Diego Klish, Nicholas Oskiper, Surya Ravindran, Adith Vettom

Advisor: Professor Tom Consi

Abstract

Our team designed and developed a smart walker system intended to improve the safety, awareness, and independence of individuals who rely on walkers for mobility. Traditional walkers provide limited feedback and no active safety response, making obstacle avoidance, fall detection, and health monitoring difficult in real-world environments. To address these limitations, our project integrates multiple sensing, control, and interface subsystems into a single smart walker platform. The system includes a handlebar subsystem for heart rate monitoring, grip force sensing, user input, and haptic feedback; a vision subsystem using LiDAR and ultrasonic sensors for front and side obstacle detection; a fall-detection subsystem using a magnetic safety strap and onboard IMU; a braking and drive subsystem for emergency response and assisted control; and a software interface for displaying walker data and warnings. By combining these subsystems into one cohesive design, the smart walker is able to provide both immediate feedback to the user and system-level responses to potentially unsafe conditions.

The smart walker is powered by a centralized power board that distributes regulated 12 V, 5 V, and 3.3 V supplies to the embedded components. Obstacle information gathered from the LiDAR and ultrasonic sensors is processed to generate vibrational alerts in the handlebar subsystem based on the side of the walker where the obstacle is detected. This allows the user to receive intuitive directional feedback without relying solely on visual cues. The fall-detection subsystem monitors strap status and walker orientation, and when a fall event is confirmed, the system can latch an emergency state that issues alerts and interfaces with the braking hardware. In the backend, data collected from these subsystems is transmitted to a server, where it can be accessed by an emergency contact or used for monitoring purposes. This combination of onboard sensing, embedded control, and remote data access allows the walker to function not only as a mobility aid, but also as a connected safety device.

Results showed that the sensing and feedback subsystems operated as intended in standalone testing, including reliable obstacle readings, repeatable fall-state detection, successful motor-driver communication, and functional handlebar sensing and vibration outputs. With each individual component complete, the next steps are to fully integrate each subsystem for proper handling, power distribution, and data transferring. This project demonstrates that a modular smart walker can combine health monitoring, obstacle awareness, fall detection, and active safety features into a practical assistive mobility system.

Athletic Form Assessor (AFA)

Team Members: Chamroeunrath Chiv, Luis Rodriguez, Erin Choudhari, Nora Schmitt, Aniket Sarkar, William Masterman

Advisor: Professor Canek Fuentes

Abstract

Strength training is a critical component of maintaining a healthy lifestyle, yet it carries a significant risk of injury when performed with improper form. Over 25 million Americans used free weights in 2023 alone, and a 10-year analysis of emergency department visits found over 15,000 resistance training-related injury cases among individuals aged 14-35. Research further identifies improper technique as one of the leading factors to these injuries. The same research concludes that proper training regarding set-up and lift execution is essential to minimizing injuries. Current solutions for personal form assessment fall short in several ways. Hiring a personal trainer is the most effective way to determine your athletic form, but the quality of trainers also varies significantly and can be prohibitively expensive and inaccessible for the average gym-goer. Camera-based form assessors require the user to set up and film themselves, which in a busy environment can raise privacy concerns and create an uncomfortable experience. Relying on gym mirrors also requires the users to have prior knowledge of what the correct form looks like. For example, a flat-bench bench press would require the user to be lying down, making it impossible to look at mirrors.

Our solution, the Athletic Form Assessor, aims to provide users with an affordable, portable, and simple solution, without the invasion of the users' and their surrounding privacy. Rather than using cameras, our system relies entirely on wearable sensors to detect movement, meaning users can train without filming themselves or requiring a dedicated space. The system delivers both rep detection and rep quality based on the duration of the rep, as well as the range of motion for the rep. This will provide users with meaningful feedback tailored to the user's physical features. The Athletic Form Assessor consists of three wearable sensor modules, each built around an Arduino Nano 33 BLE, which captures motion data during exercise. The collected sensor data will be fed into LIMU-BERT, an AI model trained and selected for its ability to extract generalizable motion features from IMU sensor data with minimal labelled training samples. The backend, built in Python, processes the sensor data and runs the LIMU-BERT model, passing the results to a React-based web application frontend where users can view their rep count, form quality scores, and feedback in real time.

Overall, the Athletic Form Assessor fills the gap left by expensive, privacy-invasive, and inaccessible form assessment tools. By combining lightweight wearable hardware with an AI-powered backend and a portable web application, the system is able to offer a solution for anyone looking to improve their physical fitness through strength training.

GloVR: A VR Hand Rehabilitation System

Team Members: Page Patterson, Catherine Menten, Jackymora Isa, Spencer Sweeney, Tobechukwu Aniagboso

Advisor: Professor Bahram Shafai

Abstract

Hand rehabilitation often requires patients to perform repetitive therapeutic exercises consistently; however, traditional approaches can feel monotonous, are difficult to monitor remotely, and lack meaningful real-time feedback, leading to reduced patient engagement and limited visibility into recovery progress. To address these challenges, we developed GLOVR, an integrated rehabilitation system combining wearable sensing hardware, a virtual reality (VR) therapy environment with guided exercises, and a web-based analytics platform. The core problem we aimed to solve was the absence of an engaging, data-driven rehabilitation system that simultaneously motivates patients and provides clinicians with quantitative, longitudinal insight into patient progress. Our solution is a closed-loop pipeline that captures hand and wrist movement data, delivers it through adaptive and immersive VR exercises, and stores it in a centralized platform for analysis and long-term tracking.

The system operates through a multi-layer pipeline connecting hardware, VR, and web components. A sensor-equipped glove captures real-time hand motion from the user as they complete their exercises. Force sensors, flex sensors, and IMUs are positioned along the inside of the glove to track movement of the wrist, arm, finger flexion, and grip strength of each finger. A custom PCB processes this data and transmits via WIFI to a patient/doctor website and the VR environment, where it is mapped onto three adaptive therapy exercises. In Pinch Control, patients pinch color-coded balls that travel toward the thumb, each corresponding to a specific finger; the exercise tracks individual finger pressure, pinch speed, and timing accuracy. In Grab Control, patients extend their arm and close their fist to catch birds flying across their field of view, with range of motion, catch rate, and grab speed recorded. In Wrist Flexion and Extension, patients tilt a closed fist upward to roll a ball in a shuffleboard-style game, with wrist range of motion and score tracked over time. All three exercises auto-adjust in difficulty based on prior session performance to progressively challenge the patient. Before beginning exercises, patients are guided through a hand-stretching warm-up led by an animated humanoid instructor, who also provides spoken tutorials when new exercises are introduced. Session data is transmitted to a web application serving as both a patient dashboard displaying progress charts and a clinician portal for monitoring performance trends, managing sessions, and evaluating recovery metrics such as grip strength, finger flexion, wrist range of motion, and session consistency.

System evaluation validated the end-to-end software pipeline including data transmission, session management, role-based access control, and performance visualization. Results demonstrate that rehabilitation data can be effectively integrated into a unified platform supporting both patient engagement and therapist oversight. Limitations include reliance on an external backend process for full session orchestration, limited real-world clinical testing, and a glove prototype that incorporates printed circuit boards and external wiring which may reduce patient comfort and portability. Future work will focus on improving deployment reliability, expanding analytical capabilities, adding further therapy exercises, and redesigning the glove into a more compact and user-friendly wearable to improve adoption in real-world rehabilitation settings.

Autonomous Indoor Delivery Robot Using ArUco-Based Localization

Team Members: Renke Cui, Hansen Cao, Matthew Shen, William Whelan

Advisor: Professor Tom Consi

Abstract

The Capstone Team has designed and built an autonomous indoor delivery robot capable of transporting small packages within structured environments such as laboratories, offices, and hospitals. Repetitive point-to-point delivery tasks in these settings place unnecessary burden on personnel, and existing solutions either rely on costly infrastructure or require significant environmental preparation. The proposed system addresses this gap by offering a self-contained, vision-guided mobile platform that can be deployed with minimal setup.

To perform a delivery, the robot is assigned a target destination identified by a unique ArUco marker placed at the delivery location. The robot carries a compact camera module that continuously captures video feed, which is processed in real time using OpenCV-based marker detection running on an onboard single-board computer. As the robot scans its environment, it identifies the target marker and determines the relative bearing and distance to the destination. Navigation decisions are made based on this visual input and relayed to an motor expansion board, which independently controls all four mecanum wheels. The mecanum drivetrain allows the robot to translate laterally and rotate in place, enabling precise maneuvering within confined indoor spaces without requiring wide turning radii. In addition to vision, a LiDAR sensor mounted on the platform performs continuous 360-degree scanning to detect obstacles in the robot's path, allowing it to avoid obstructions in real time.

For testing and validation, each individual component has been tested and confirmed to work as desired. We are currently in the process of integrating the components together. Our integration testing plan begins with testing robot mobility using the mecanum wheels. The next step is to test the LiDAR sensor on the robot; a successful integration would mean the robot can traverse a hallway while using the LiDAR to detect and avoid obstacles. Following that, the camera sensors will be tested for ArUco marker detection and matching. A successful integration would mean the robot can successfully scan a marker and identify the corresponding package drop-off point. The final step is to test the dump truck mechanism to ensure the robot can reliably drop off packages at the designated location. After these integration tests, the final test is a complete procedure test, ensuring the robot completes the whole delivery process as desired.

Search and Rescue Operation System (SAROS)

Team Members: Laanu Adeboye, Shyma Alshari, Daniel Costanzo, Matthew Egg, Evan Moncheck, Amanda Teixeria, Ted Wilson

Advisor: Professor Bahram Shafai

Abstract

The SAROS team has designed and built a system to improve the accessibility and efficiency of search and rescue missions by using a drone to detect cellular signals. Such a system has a wide range of applications including detection leading to rescues in ski resorts, forests, natural disaster zones (Red Cross), and military bases. Since people tend to always have their phones on their person, being able to detect the position of a person's phone in an unfamiliar environment can be crucial to rescue efforts. One problem this solution addresses is the reliance of helicopters to be deployed to locate missing people. SAROS' design proposes using a drone, or even a fleet of drones, to be sent out and autonomously or manually search for cellular signals, mark the position of where the signal was found, and alert rescuers.

To use SAROS, the person in need of help would either need to be carrying a powered iPhone, or a radio transmitting a constant-frequency tone (a further revision of this project might include a specially designed low-power radio for this purpose). Apple-brand mobile phones that are not dead transmit Bluetooth Low Energy signals in order to update their maintained location (Find My software) which can be detected by any Bluetooth receiver and companion computer. Furthermore, radio transmissions can be received by a software defined radio on board a drone which is configured to search for the frequency which the user's radio is operating at, so as to communicate any located signal power back to ground station. A companion computer is the orchestrator of this location process, where both the Bluetooth receiver and software-defined-radio communicate their latest findings, as the companion computer routes them back to ground station over the flight controller as telemetry. There is a laptop base station which receives the location where the signal is detected from the drone on an ArduPilot graphical user interface (GUI). The GUI can be used to create flexible user-defined search patterns for the drone to follow in order to search for signals as well as plan routes for the drone to autonomously follow. As the drone flies over a person, within a tested range of about 15 meters, the radio transmission of the person in need can be located and mapped back at ground station through heatmaps. The Bluetooth transmissions from their iPhone will be found, transmitted, and used in order to estimate a radius where the drone suggests the transmission from said iPhone took place.

For testing and validation, the SAROS team went to a park around Northeastern campus with the detector which was attached to the drone, and mobile devices. After verifying that the drone could be flown through the UI, the team went on to check if iPhone signals could be detected by the detector. The results indicated that the method of location was valid and the detector could locate the mobile device or transmitter radio with the drone being controlled by the GUI reliably. It is the belief of the SAROS team that this application could be further specialized and extended for use with other technologies, like RECCO, to aid in more search and rescue efforts around the globe.