K.H.U. Faculty of Engineering and Natural Sciences

**FENS 402 ENGINEERING DESIGN PROJECT**

**PROGRESS REPORT**

**WEARABLE SMART DEVICE FOR ​**

**VISUALLY IMPAIRED PEOPLE​**

Project Advisor: Assist. Prof. Dr. Arif Selçuk Öğrenci​

Project Team:[[1]](#footnote-2)\* Berke Kaya, Berkay Sarı, Hakan Çanakçı, Samet Kartlar

**1. System Overview**

The goal of this project is to create a system that will help those who are visually impaired in their daily lives. The system's IMU sensors allow it to track physical movements in real time and notify the appropriate individuals or units in the event of a potential issue. Additionally, the camera module uses a Bluetooth headset to deliver the user an auditory alert when it detects things in the environment. The wearable nature of this device makes it an alternative approach.

**metin, ekran görüntüsü, çizgi, diyagram içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.**

Figure 1 System Design

**Selected Hardware**

The system uses the following hardware components:

* **Raspberry Pi 5 (8GB):** Raspberry Pi 5 (8GB) was chosen because it is suitable for gathering and processing data in real time as well as communication with other components used. It is also used because of its high processing power and 8 GB RAM capacity as well as its wide GPIO support. It has sufficient hardware structure for simultaneous collection and processing of data from cameras and other sensors.
* **IMU Sensor (BNO055):** IMU Sensor is used to track the user's movement and collect data for activity recognition. It provides much more accurate data collection compared to other standard IMUs. Thus, it increases the accuracy of classifying the user's movements.
* **IMX219 Binocular Camera:** The IMX219 Binocular Camera was chosen for depth perception with the stereo imaging it provides. Thanks to the depth perception optimization performed by taking images simultaneously with the two cameras at different angles, it allows the distances to be calculated and the distance of objects to the camera to be determined precisely.
* **Raspberry Pi Camera Module 3:** By taking excellent pictures that work with the Raspberry Pi, it is utilized to identify items in the surroundings and serves as the system's visual data input.
* **Bluetooth Headphones:** The Bluetooth Headphones provide the user with audio feedback about obstacles and their distances detected by the camera in real time.

**2. Initial Components and Design Decisions**

Because it was readily available and reasonably priced, we used the MPU6050 as our IMU sensor in the first phase of the project. Ax, Ay, Az, Gx, Gy, and Gz are the three axes on which the sensor gives raw accelerometer and gyroscope data. Although this made it possible to gather preliminary data and identify some basic activities, we ran into a number of serious problems that compromised the dependability of our system. In order to obtain orientation variables such as pitch, roll, and yaw, the MPU6050 needed sophisticated sensor fusion algorithms and frequently generated noisy or unstable outputs. Furthermore, over time, the sensor experienced drift, necessitating extra calibration procedures to preserve accuracy.

metin, ekran görüntüsü, kablo, elektronik donanım içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

Figure 2 MPU6050 Output Data

**3. Component Replacement and Improvements**

At initial testing, the MPU6050 sensor was problematic, producing inconsistent or absent data. To overcome these constraints, we switched to the Adafruit Bosch BNO055 sensor, which includes a sensor fusion chip that offers preprocessed alignment data such as heading, roll, and pitch. This considerably increased measurement stability and consistency, reduced preprocessing requirements, and enabled more accurate real-time activity monitoring. This update enabled us to move forward with dependable feature extraction and model training for human activity recognition. Due to the Raspberry Pi's restricted availability in the early stages, preliminary sensor tests were carried out using an Arduino board. This allowed us to test sensor operation before fully integrating the Raspberry Pi.

**4. Progress on Sensor Integration and Data Collection**

We successfully linked the BNO055 sensor with the Raspberry Pi and tested real-time data collecting (heading, pitch, roll). A script has been created to collect labeled activity data over short time intervals, which will eventually be extended to longer sessions. The data collection procedure is ongoing. To ensure proper data separation, we decided to keep each activity in a different CSV file. This method streamlines the labeling process and assures uniform input formatting for the machine learning model.

Data Storage and Format:

* **File Formats:** Data collected from sensors is kept in CSV format. This format is easily compatible with libraries used for data processing and machine learning. In accordance with the measurement data, information such as activity status and date are saved in JSON format to make it more organized and readable.
* **Organization:** The acquired data is labeled based on the activity type and stored in various files. This enables clarity and distinction during model training.

We used a scenario-based labeling strategy, with each activity recorded in a distinct file for clarity and model training. The anticipated dataset will include at least one minute of data for each activity, such as walking, running, standing, sitting, lying down, and falling. A test subject will do these actions based on vocal orders, while the Raspberry Pi records sensor data in real time.

Although the software for this pipeline has been completed and successfully tested with brief 10-second trials, full-length data collection has not yet begun owing to the lack of a finalized wearable product body. The sensor will eventually be attached on the physical harness, which is still in the prototyping stage, to ensure constant and realistic sensor positioning during real-world motions. Once the body is finished, extensive trials will commence under controlled conditions to assure data quality and consistency.

**5. Audio Feedback Planning**

Although the original design focused on haptic feedback using vibration motors, we have broadened our plans to include audio-based feedback via Bluetooth headphones for more natural notifications. Using real-time image processing, items spotted in the user's path are classified as moving, stationary, or potentially moving.

Each category will play a pre-recorded audio message in WAV or MP3 format. For example:

* *“A person is approaching at 5 meters.”*
* *“A vehicle is moving at 8 meters.”*
* *“An obstacle is 3 meters ahead.”*

These messages will be generated dynamically by mixing preset words with real-time distance data obtained from the stereo vision module's depth analysis. The audio system attempts to boost user awareness without requiring visual focus, and it will run alongside the Raspberry Pi's object detection module.

**6. Machine Learning Model Preparation**

We chose the Random Forest classifier for the activity recognition challenge because of its robustness, low computational cost, and consistent performance on small- to medium-sized datasets. Random Forest is also more interpretable than deep learning models, making it ideal for edge devices with minimal computing capacity, such as the Raspberry Pi.  
  
Although deep neural networks like CNNs, LSTMs were initially investigated for their ability to learn complicated patterns, they were rejected during this phase due to their high computational and memory needs, lengthier training time, and reliance on huge labeled datasets. These factors make them less suitable for real-time applications on embedded devices that lack GPU acceleration.

We incorporated a model comparison analysis from earlier research (see Figure 1 for F1-scores across models and Figure 2 for the Random Forest Model's confusion matrix). The findings show that Random Forest competes with deep learning models in terms of categorization accuracy, particularly for basic human activities such as walking, lying down, and falling.

Once enough labeled data has been acquired using the BNO055 sensor, model training will commence. Our preprocessing pipeline consists of filtering, normalization, and feature extraction (mainly from orientation data: Heading, Pitch, and Roll). These characteristics were chosen due to their high association with body posture and movement dynamics.

metin, ekran görüntüsü, diyagram, sayı, numara içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

Figure 3 F1-score comparison of various models [1]

ekran görüntüsü, metin, diyagram, kare içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

Figure 4 Confusion matrix for Random Forest model [1]

**7. Activity Monitoring Progress and Model Selection**

Our system's activity monitoring part has successfully evolved from simple motion sensing to structured data collecting using the Bosch BNO055 sensor. This IMU delivers high-quality orientation data (heading, roll, and pitch), allowing us to distinguish between human behaviors like walking, lying down, and falling with more accuracy. For the classification model, we used the Random Forest approach. This option was made due to its minimal computing requirements, quick training time, and excellent performance with medium-sized, tabular datasets. Although more complex models such as deep neural networks were investigated, they were ultimately rejected due to their high processing requirements and Raspberry Pi's limitations as an edge device. RF also has the benefit of interpretability and resistance to overfitting, making it appropriate for noisy real-world IMU data. We have completed our data labeling technique by storing each action in individual CSV files, and we are actively expanding data collecting to provide a balanced dataset for model training and validation.

To start building and testing our activity monitoring algorithm, we built a simulated experimental setup with publicly available IMU datasets rather than gathering data ourselves. These datasets contained labeled sensor readings from activities such as walking, standing, and jogging. We processed the data by structuring it into consistent CSV files with six primary features: Ax, Ay, Az, Gx, Gy, and Gz. This enabled us to train and evaluate our Random Forest classifier without having to acquire real-time data in the early stages. The sample data is formatted as follows.

Table 1 Arranged Format for the Public Dataset [2]

metin, yazı tipi, siyah beyaz, ekran görüntüsü içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

To better understand the behavior of various sensor readings across activities, we plotted the distribution of each feature. The distribution plots revealed considerable differences across activities such as walking, standing, and running, especially in the accelerometer and gyroscope axes, justifying their inclusion in the categorization model.

metin, diyagram, öykü gelişim çizgisi; kumpas; grafiğini çıkarma, çizgi içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

Figure 5 Distribution Graphs of Features

We plotted the Random Forest model's feature significance scores to determine which sensor readings had the greatest impact on accurate predictions. The findings revealed that gyroscope and accelerometer readings along specific axes (such as Gz and Ax) were the most useful for discriminating between activities.

metin, ekran görüntüsü, diyagram, çizgi içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

Figure 6 Feature Importance

The confusion matrix represented a visual assessment of our model's performance. It demonstrated great accuracy in recognizing walking and standing activities, with negligible misclassification. This tested the model's reliability and demonstrated that the features and preprocessing processes were effective.

metin, ekran görüntüsü, diyagram, sayı, numara içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

Figure 7 Confusion Matrix of the Model

**8. Object Detection and Depth Estimation Progress**

Parallel to the activity monitoring work, we have made substantial progress in the image-based object detection module. Initially, the focus was on detecting general objects such as people, vehicles, and static obstacles using the Raspberry Pi Camera Module 3. However, to enhance environmental awareness, especially in navigation or mobility support scenarios, we began experimenting with binocular (stereo) vision. To calibrate the stereo camera system, we used a chessboard pattern and captured nearly 30 images from different tilt and angle positions. This allowed us to compute stereo calibration parameters and generate accurate disparity maps, which in turn provide reliable depth information. With this setup, we are able to categorize detected objects based on their distance and motion potential (e.g., “a person is approaching at 3 meters”). We plan to integrate this information with the real-time audio feedback system to notify the user via Bluetooth headphones. While object detection is currently rule-based and proximity-driven, future steps may include implementing lightweight models like YOLO-tiny for real-time detection on constrained hardware.

ağaç, ekran görüntüsü, dış mekan, gökyüzü içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

Figure 7: Semantic segmentation-based classification of objects based on movement potential.

In this phase of the project, semantic segmentation was implemented to enhance the spatial awareness of visually impaired users by providing detailed object-level understanding of the environment. Using an advanced version of the YOLOv11-based detection system, objects in the scene were not only detected but also segmented to highlight their occupied space within the frame. This approach provides a more comprehensive representation of the environment by covering the actual area of each object, rather than merely indicating its location with a bounding box.

As part of the movement-based classification, the segmented objects were color-coded according to their potential for movement: blue indicates stationary objects (e.g., parked vehicles), while red indicates objects with movement potential (e.g., pedestrians). This clear visual distinction allows the system to communicate environmental risks more effectively, especially when integrated with auditory feedback systems for visually impaired users.

As illustrated in Figure 7, multiple vehicles on the left and right side of the road are highlighted in blue, marking them as stationary, while pedestrians walking on the road are segmented in red, identifying them as potential moving obstacles. The enhanced visualization supports the development of a real-time assistive system that not only detects objects but also interprets and prioritizes them based on risk.

metin, kara taşıtı, taşıt, araç, tekerlek içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

Figure 8 Object Detection with YOLO

This project is an intelligent system developed to enhance environmental awareness for visually impaired individuals by detecting surrounding objects and classifying them based on their potential for movement. The system operates in real time, identifying objects in its environment and categorizing them according to whether they are likely to move or remain stationary. For object detection, the project utilizes the deep learning-based YOLOv11 (You Only Look Once) algorithm. Through this model, objects such as vehicles and pedestrians are successfully identified in the video stream and labeled with their positions.

The detected objects are divided into two main categories: those with potential for movement and stationary ones. Objects with movement potential are highlighted in red, while stationary objects are shown in blue. This classification aims to inform visually impaired users about objects that may pose a potential risk due to their mobility. The system is designed to be further enhanced with a voice notification mechanism, which would alert users about dynamic hazards in their immediate surroundings.

As shown in Figure 8, objects such as cars and pedestrians — considered to have movement potential — are marked with red bounding boxes and labeled as “car – Movement Potential.” This application is considered a foundational step toward a more comprehensive system that will contribute significantly to safer and more interactive mobility for visually impaired individuals.

To calculate a subject's distance from the camera using a stereo camera, we must first generate a depth map (disparity map). The disparity map shows the horizontal pixel difference (pixel shift) between matching pixel pairs in two stereo pictures taken from different angles of the same scene.

çizgi, diyagram içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

where Z=depth, f=focal length, B=baseline (camera separation), d=disparity.

The steps to create the disparity map are as follows:

1. 30 pairs of photos are captured, displaying the chessboard to the stereo camera from various angles and distances.   
2. The inner corners of the chessboard are identified in each frame, yielding the 2-D position of each corner in the image.  
3. Using the corner data, the focal length, optical center, and distortion coefficients for each camera are determined.  
4. The positional relationship between the two cameras (rotation R, base distance T) is identified.   
5. The photos are resampled so that the conjugate points are in the same horizontal row, and the search is limited to the x-axis.   
6. Each pixel's horizontal counterpart in the left and right images is looked for, and the pixel shift (disparity) is determined.

We did these steps in order. Figure 9, 10, 11 shows the steps required to create a disparity map. As can be seen in Figure 11, objects close to the camera are colored red because they have a higher disparity value, while objects far from the camera are colored blue because they have a lower disparity value.

iç mekan, duvar, kişi, şahıs, mobilya içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

Figure 9 Capturing Stereo Images of Chessboard

ekran görüntüsü, bilgisayar, multimedya, görüntüleme cihazı içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

Figure 10 The Inner Corners of Chessboard

metin, ekran görüntüsü, yazılım, multimedya yazılımı içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

Figure 11 Disparity Map

**9. Physical Body Design and Prototyping**

Development of the physical body is currently ongoing. We have designed an initial prototype using SolidWorks, which helps us visualize and refine the placement of components such as the Raspberry Pi, stereo camera module, and IMU sensor. To ensure the system is wearable and practical, we are designing a harness-like structure that can be comfortably mounted on the user's upper body.

tasarım, kutu, iç mekan içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

Figure 12 Product Body Design

However, the initial box design shown in the following figure, though comprehensive, turned out to be too large and bulky for practical use. After testing and evaluating ergonomics, we decided to redesign the system to prioritize comfort and usability. In the revised design, only the Raspberry Pi and the stereo camera will be enclosed in a compact case mounted on the front side of a pocketed vest. Additional equipment such as the power bank and sensor wiring will be distributed into the vest’s pockets to balance the load and ensure mobility. This approach improves both the ergonomic fit and day-to-day usability of the system.

tasarım, çizim içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.konteyner, kutu içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

Figure 13 New Design

**10. Challenges and Deviations**

• Power Management: We are currently evaluating whether the Raspberry Pi can operate all components efficiently with a 5V 3A power supply, especially under simultaneous sensor and image processing workloads.

• Hardware Availability: Due to the late arrival of the Raspberry Pi, we initially used an Arduino-based setup to test and validate the IMU sensor.

• Audio Feedback: Although audio feedback is included in our system design, its implementation was postponed to prioritize sensor integration and data collection.

• Physical Product Design: The absence of a finalized wearable body prevented us from beginning real-world data collection. Additionally, our limited experience in mechanical design contributed to delays in developing an ergonomic and functional prototype.

**11. Next Steps**

• Finalize and 3D-print the redesigned compact enclosure for the Raspberry Pi and camera module.

• Start full-scale data collection for all targeted activities (e.g., walking, sitting, falling) to build a balanced and labeled dataset.

• Begin training and validating the Random Forest model using the collected data.

• Integrate and test the stereo camera module for real-time object detection and depth estimation.

• Develop and implement the audio feedback system, dynamically triggered by object classification and spatial analysis.

**PROJECT CALENDAR**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***PROJECT TITLE: WEARABLE SMART DEVICE FOR VISUALLY IMPAIRED PEOPLE*** | | | | | | | | | | | | | | | | | | | | |
|  | 2025/1 | | | | 2025/2 | | | | 2025/3 | | | | 2025/4 | | | | 2025/5 | | | |
| 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 |
| **Work Packages** |  | | | | 100% | | | | 60%  90%  75% | | | | 65%  25%  45% | | | | 50%  0% | | | |
| **I. Phase I**  **a. Literature Review**  **b. Project Planning** |
|  |
| **II. Phase II**  **a. Obstacle Detection System Design**  **b. Activity Monitoring System Design** |
|  |
| **III. Phase III**  **a. Device Build-up**  **b. Hardware Integration** |
|  |
| **IV. Phase IV**  **a. Testing**  **b. Optimization**  **c. Validation** |
|  |
| **V. Phase V** |

**EXPENDITURES:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Work Packages** | | | |  |
|  | **I** | **II** | **III** | **IV** | **TOTAL** |
| **Equipment** | - | 9850 TL | - | - | 9850 TL |
| **Consumable goods** | - | - | - | - | 0 |
| **Publications/Software** | - | - | - | - | 0 |
| **Transportation** | - | - | - | - | 0 |
| **Services** | - | - | - | - | 0 |
| **TOTAL** | 0 | 9850 TL | 0 | 0 | 9850 TL |

**Equipment:**

Raspberry Pi 5 8GB with Active Cooling – 4.142,30 TL

IMX219-83 Stereo Camera - 2.901,64 TL

BNO055 IMU Sensor – 1.842 TL

Raspberry Pi 5 SD Card and HDMI Cable – 965,42 TL

**Consumable goods:**

**Publications/Software:**

**Transportation:**

**Services:**

**REFERENCES:**

1. N. G. Nia, E. Kaplanoglu, A. Nasab and H. Qin, "Human Activity Recognition Using Machine Learning Algorithms Based on IMU Data," 2023 5th International Conference on Bio-engineering for Smart Technologies (BioSMART), Paris, France, 2023, pp. 1-8, doi: 10.1109/BioSMART58455.2023.10162095.
2. Mohammad Malekzadeh, Richard G. Clegg, Andrea Cavallaro, Hamed Haddadi, Privacy and utility preserving sensor-data transformations, Pervasive and Mobile Computing, Volume 63, 2020, 101132, ISSN 1574-1192, https://doi.org/10.1016/j.pmcj.2020.101132.

1. \* (1) Electrical and Electronics Engineering Department [↑](#footnote-ref-2)