8^{th} Semester Electronic Systems

Data-Driven Predictive Control for Robust Walking Assistance of Exoskeletons

P8 Technological Project: Systems of Systems

Department of Electronic Systems
Aalborg University
Project Group 859
Group participant: Qianrun He
Supervisors: Shaoping Bai and Ming Shen
Project Period: ES8 semester

AALBORG UNIVERSITET
STUDENTERRAPPORT

Title page

8^{th} **Semester Electronic Systems**
Department of Electronic Systems
Aalborg University
9220 Aalborg

Title:
Data-Driven Predictive Control for Robust Walking Assistance of Exoskeletons

Project:
P8 Technological project:
Systems of Systems

Project Period:
ES8 semester

Project Group:
859

Group participants:
Qianrun He

Supervisor:
Shaoping Bai,Ming Shen

Pages: 27

Abstract
This project investigates environment perception for an exoskeleton using a HoloLens 2 RGB camera together with synchronized IMU and FSR measurements. The perception task is formulated as a three-class classification problem: walking up stairs, walking down stairs and walking on flat ground. A MobileNetV3-Large visual backbone was first pretrained on a binary StairNet stair/flat task and then transferred to project-specific HoloLens 2 recordings, reaching 96.72% test accuracy on the RGB task. A synchronized RGB+IMU+FSR dataset was built at 3 FPS, where each image is paired with nearest sensor samples and short-window sensor statistics. The strongest RGB+sensor configurations reached 95.90% mean test accuracy over three seeds, while RGB-only reached 95.36%. Video testing showed stable stair-ascent predictions, but transition frames remained difficult. The results indicate that the current system is primarily visually driven, with IMU and FSR providing complementary but limited improvements.

Preface

This report is written by project group 859 from the 8^{th} semester of Electronic Systems at Aalborg University. We extend our gratitude to our supervisors and to the university for their guidance, support and academic environment throughout the project period.

The following chapters of this report provide a detailed description of how to design an exoskeleton walking-assistance system that combines visual perception with wearable sensor information. The report covers the problem analysis, system delimitations, implementation choices and experimental evaluation of an RGB, IMU and FSR based environment-perception pipeline.

Aalborg University, May 28, 2026

Qianrun He

qhe24@student.aau.dk

Table of contents

Preface **iii**

1 Introduction **1**

2 Problem Analysis **2**
2.1 Sensor Introduction . . . 2
2.2 Vision-Based Environment Perception . . . 3
2.2.1 Dataset Considerations . . . 5
2.3 Summary . . . 7

3 Project Scope **8**
3.1 Vision–Sensor System for Exoskeleton-Assisted Walking . . . 8
3.1.1 Microsoft HoloLens 2 . . . 9
3.1.2 IMU–FSR Sensor Strap . . . 10
3.2 Dataset Limitations and Model Selection . . . 10
3.3 Problem Statement . . . 11
3.3.1 Objectives . . . 11

4 Implementation **13**
4.1 System Overview . . . 13
4.2 RGB Dataset Generation . . . 13
4.2.1 StairNet Pretraining Subset . . . 13
4.2.2 Project RGB Dataset . . . 14
4.3 Sensor Data and RGB Synchronization . . . 14
4.3.1 Measured Sensor Signals . . . 15
4.3.2 Synchronization Method . . . 16
4.4 Model Architecture . . . 17
4.4.1 RGB Model . . . 17
4.4.2 Fusion Model . . . 17

5 Tests **18**
5.1 Test Strategy . . . 18
5.2 StairNet Pretraining Test . . . 18
5.3 Transfer-Learning Test . . . 19
5.4 Ablation Test . . . 20
5.5 Video Test . . . 22

6 Discussion **24**
6.1 Interpretation of Results . . . 24
6.2 Dataset Limitations . . . 24

6.3 Sensor Fusion Limitations 25
6.4 What Has Not Been Claimed 25

7 Conclusion **26**

Bibliography **27**

Chapter 1

Introduction

Lower-limb exoskeletons need information about both the surrounding environment and the user's current movement before they can provide assistance that feels timely and safe. A controller that only reacts after a foot has already contacted a stair is late for many assistive use cases. A perception module should therefore estimate the upcoming walking context from vision and wearable sensors before this information is needed by the control layer.

This project implements and evaluates an environment perception pipeline for an exoskeleton scenario. The implemented system uses the HoloLens 2 as an RGB acquisition device and combines the visual stream with synchronized inertial measurement unit (IMU) and force sensitive resistor (FSR) measurements. The current perception task is limited to three labels:

- **Up**: the user is walking up stairs.
- **Down**: the user is walking down stairs.
- **Flat**: the user is walking on flat ground.

The output of the implemented model is a confidence vector over these labels. In this report, confidence refers to the model's class score after softmax normalization. It should not yet be interpreted as a calibrated probability for safety-critical control.

The project combines two data sources. First, a 20,000 image subset was sampled from the StairNet dataset and used as a binary stair/flat pretraining task. Second, project-specific HoloLens recordings were converted into RGB frame datasets and synchronized with IMU and FSR logs. The transfer-learned RGB model is evaluated on the project-specific three-class task, and the final synchronized training manifest contains 596 sampled frames from 14 synchronized sessions, while the full synchronization manifest contains 6060 frame-level rows before 3 FPS sampling.

The main contribution of the work is a reproducible pipeline that:

- exports HoloLens 2 personal video recordings into standard video and frame metadata;
- constructs RGB-only and synchronized RGB+IMU+FSR datasets from the same recordings;
- trains MobileNetV3-based RGB and fusion classifiers for the three-class environment task;
- evaluates StairNet pretraining, HoloLens transfer learning and continuous-video prediction behavior;
- compares RGB-only, RGB+IMU, RGB+FSR, RGB+IMU+FSR and sensor-only ablations over multiple random seeds.

The report is intentionally written around what has been implemented and measured in the project folder. Details that are not documented in the available files, such as final exoskeleton controller integration or user-study protocol, are treated as future work rather than assumed results.

Chapter 2

Problem Analysis

The purpose of this chapter is to understand sensor-based and vision-based solutions for exoskeleton perception. The chapter therefore introduces the IMU and FSR sensors used to describe the user's motion and contact state, reviews visual methods for recognizing walking environments before the user reaches them, and uses this analysis to motivate a practical solution for the exoskeleton environment perception problem.

2.1 Sensor Introduction

An inertial measurement unit (IMU) is a compact motion sensor that typically combines a three-axis accelerometer and a three-axis gyroscope, and in some systems also a magnetometer. MEMS-based IMUs are widely used because they are small, low cost and able to sample motion at high rates [1]. The accelerometer measures specific force along the sensor axes. When the sensor is static or moving slowly, this measurement is dominated by gravity and can therefore be used to estimate inclination. The gyroscope measures angular velocity, which can be integrated over time to estimate orientation changes. This integration is accurate over short intervals, but bias and noise cause drift over longer periods, so IMU systems commonly use sensor fusion to combine accelerometer, gyroscope and optional magnetometer information [1]. In this project, the IMU stream gives information about body motion through gravity, angular velocity, quaternion orientation, linear acceleration and Euler-angle channels.

A force sensitive resistor (FSR) is a thin resistive force sensor whose resistance decreases when force is applied to the active surface. A typical FSR is built from flexible polymer layers: one layer contains interdigitated conductive traces, while the opposing layer contains force-sensitive ink. When pressure brings the layers into contact, the ink shorts the traces with a resistance related to the applied force [2]. The response is non-linear and can saturate at high loads, so an FSR is better suited for detecting contact, relative pressure changes and gait events than for precise force measurement. In wearable locomotion systems, FSRs are often placed under or around the foot to detect heel strike, toe-off and pressure distribution, while IMU or gyroscope signals describe segment motion. Negi et al. used FSR and IMU-related signals for gait phase detection on level ground, ramps and stairs, showing that these sensors are relevant for real-time terrain-dependent gait analysis [3]. In this project, the FSR stream contains eight pressure channels, which can provide contact information complementary to the visual input.

Figure 2.1: Physical IMU–FSR sensor strap used in the project.

The key advantage of IMU and FSR sensing is that these measurements are directly connected to the user's current motion and foot contact. Their limitation is that they mostly describe what the user is already doing or has just contacted. For stair assistance, this means that IMU and FSR measurements are valuable for disambiguating motion and stance, but they should be combined with a forward-looking perception source when the goal is to anticipate the environment.

2.2 Vision-Based Environment Perception

Vision-based perception is attractive for exoskeletons because it can identify stairs, ramps or level ground before the foot interacts with the terrain. This is important for high-level control: mechanical, inertial and neuromuscular sensors can infer locomotion state, but visual sensing can reduce the delay by observing the upcoming environment in advance [4; 5]. The same principle appears in the terrain classification analysis from the previous report: camera-centric methods can provide information from a distance, while contact or vibration methods mainly become informative after the system has entered the terrain.

For lower-limb exoskeletons and prostheses, vision methods can be divided into two main groups. The first group performs environment classification, where each image or short image sequence is assigned to classes such as level ground, stair ascent, stair descent or transition states. The second group performs detection or geometric estimation, where the algorithm locates stair lines, planes or step dimensions. Classification is sufficient when the control module needs a high-level environment belief, while geometric estimation is useful when step height, width or ramp slope must directly parameterize a gait trajectory.

Table 2.1: Comparison of visual environment perception methods relevant to exoskeletons and wearable locomotion.

Study	**Method and input**	**Task**	**Reported performance**	**Advantages and limitations**
Krausz and related RGB-D geometry methods, summarized by Laschowski et al. [4]	Hand-crafted RGB-D or geometric feature extraction	Stair recognition and parameter extraction	Up to 98.8% accuracy in one stair-recognition example, but evaluated on only five images with about 8 s/frame	Provides geometric information such as stair structure. Strongly limited by small evaluation sets, hand-tuned thresholds and slow processing.
Diaz et al., summarized by Laschowski et al. [4]	SURF features with bag-of-words classification	Six-class walking environment classification	86% image classification accuracy	Simpler than deep learning and easier to run on limited hardware. Requires manual feature design and gives lower accuracy than later CNN methods.
Massalin et al., summarized by Laschowski et al. [4]	Depth-image features with cubic-kernel SVM	Five-locomotion-mode classification	94.1% accuracy and about 14.9 ms/image	Depth features help separate terrain shapes and can be fast. Requires hand-engineered features and depends on depth sensing, which is less robust outdoors and at longer range.
Laschowski et al. [4]	CNN benchmark on ExoNet RGB images	Twelve-class environment classification	EfficientNetB0: 73.2% accuracy. MobileNetV2: 72.9% accuracy and best NetScore of 76.2. VGG16: fastest TPU inference at 1.4 ms/image	Large and diverse wearable-camera benchmark. Shows that lightweight architectures are important, but twelve-class classification is harder than a focused stair/flat task.
Kurbis et al. [5]	StairNet models using egocentric RGB images, including lightweight CNN and temporal models	Stair and level-ground recognition for human-robot locomotion	Over 515,000 labeled images. Models reached up to 98.8% accuracy and up to 2.8 ms inference on mobile GPU/NPU hardware	Directly aligned with exoskeleton and prosthesis control. Focused mainly on stair recognition, so additional classes or sensors are needed for broader environment understanding.

Study	**Method and input**	**Task**	**Reported performance**	**Advantages and limitations**
Wang et al. [6]	End-to-end monocular CNN for stair-line detection	Pixel-level stair-line detection	81.49% accuracy, 81.91% recall and 12.48 ms runtime; lightweight version over 300 FPS	Very fast and avoids a two-stage detector-plus-line-extractor pipeline. It detects stair structure rather than directly giving user-centered classes such as up, down and flat.
Wang et al. [7]	StairNetV2 with RGB-D inputs and selective feature fusion	Stair detection and geometric estimation	Day/night accuracy of 92.88% / 90.78% for the full model; stair geometry RMSE within 15 mm ascending and 25 mm descending	Robust to weak visual cues because depth complements RGB. Requires depth input, IMU-based coordinate transformation and post-processing, which increases system complexity.

The literature shows that vision improves the prediction horizon of wearable robot control, but it also shows a trade-off between detail and deployability. RGB-D and geometric methods can estimate step dimensions, but they require depth quality, calibration and post-processing. Large CNNs can improve accuracy on some datasets, but wearable systems need low latency and low computational cost. Laschowski et al. explicitly compared CNN architectures for exoskeleton and prosthesis environment recognition and found that MobileNetV2 provided the best balance between accuracy, parameters and operations, even though EfficientNetB0 had the highest raw accuracy [4]. The StairNet work further supports the use of lightweight mobile-oriented CNNs for stair perception, achieving high stair-recognition accuracy on a large egocentric dataset [5].

For this project, MobileNetV3 is therefore selected as the visual training backbone. MobileNetV3 was designed for mobile vision through hardware-aware architecture search and NetAdapt, and the MobileNetV3-Large model improved ImageNet accuracy while reducing latency compared with MobileNetV2 [8]. This fits the project because the exoskeleton perception module should be lightweight enough for wearable or edge deployment, while still strong enough to learn visual differences between stairs and flat ground. The project also needs a feature extractor that can be fused with IMU and FSR features. A MobileNetV3 classifier can be modified by replacing its final classification layer with a feature output, allowing image features to be concatenated with sensor features in a compact fusion model. Since the current task is a high-level confidence estimate over up stairs, down stairs and flat ground, MobileNetV3 is a better fit than a heavier geometric stair-estimation pipeline.

2.2.1 Dataset Considerations

After selecting a lightweight visual backbone, the next problem is the choice of training data. In exoskeleton and prosthesis vision research, ExoNet and StairNet are frequently used because both are egocentric wearable-camera datasets designed for human locomotion environments. ExoNet was collected using

a chest-mounted wearable RGB camera and contains more than 5.6 million images, of which approximately 923,000 were manually annotated into a 12-class hierarchical label structure [9]. These classes include level ground, incline stairs, decline stairs, doors, walls, seats and transition cases. This makes ExoNet broad and useful for general environment recognition, and it is also the dataset used in the CNN comparison by Laschowski et al. [4].

For the present project, the main limitation of ExoNet is not its size, but the consistency of the image and label quality for a focused stair-assistance task. The StairNet authors reported that ExoNet labels contained overlapping classes that limited stair-recognition performance, and later sections of the same work note that unlabeled ExoNet images could include irrelevant content and significant camera obstructions [5]. In addition, ExoNet is designed as a broad environment database rather than a clean stair-specific pretraining set. A model trained directly on ExoNet may therefore learn general scene context, but it can also be affected by ambiguous transitions, partial stair visibility, camera obstruction and uneven class definitions.

StairNet was developed to address these issues for stair recognition. It uses images from ExoNet, but re-labels them with new class definitions for level ground (LG), level-ground transition to incline stairs (LG–IS), incline stairs (IS), and incline-stairs transition to level ground (IS–LG). The dataset contains more than 515,000 manually labeled RGB images and was processed using multiple annotation pass-throughs to improve label precision. Images that did not contain level ground or incline stairs, or that had significant camera obstructions, were removed [5]. Compared with ExoNet, StairNet is therefore more suitable for training a stair-aware visual backbone because it has cleaner task-specific labels, fewer irrelevant samples and demonstrated high model accuracy on stair recognition.

However, StairNet still does not fully match the needs of this project. The published StairNet classes focus on incline stairs and level-ground transitions, and inspection of the StairNet images used in this project showed that the stair examples primarily describe an upward or incline-stair situation. There is no separate decline-stair class corresponding to walking down stairs. This is a critical mismatch because the exoskeleton perception module must distinguish `up`, `down` and `flat`. StairNet can therefore be used as a useful binary stair/flat or stair-aware pretraining source, but it cannot by itself provide the complete three-class supervision required by the final project model.

Figure 2.2: StairNet incline-stair example illustrating that the available stair images do not cover the required down-stair class.

2.3 Summary

This chapter introduced the two non-visual sensor types and the visual perception direction used in the project. IMU measurements describe body motion through acceleration, angular velocity and orientation-related signals, while FSR measurements describe force-contact patterns. These sensors are useful for understanding the user's current gait state, but they do not by themselves provide a long preview of the upcoming environment.

Vision-based perception addresses this limitation by observing the walking environment before physical contact. Prior exoskeleton and prosthesis research shows that CNN-based methods can classify walking environments and stairs effectively, while RGB-D and geometric methods can additionally estimate stair dimensions at the cost of more sensing and post-processing complexity. Based on the need for a lightweight, trainable and sensor-fusion-friendly model, MobileNetV3 is selected as the visual backbone for the exoskeleton environment perception task.

The dataset analysis shows that existing exoskeleton vision datasets are useful but not sufficient on their own. ExoNet provides scale and diverse walking environments, but its broad and partly overlapping labels make it less suitable as a clean stair-specific source. StairNet improves on ExoNet by re-labeling and filtering images for stair recognition, making it valuable for stair-aware pretraining. Its limitation is that it focuses on level ground and incline-stair situations, while this project requires separate up-stair, down-stair and flat-ground predictions. The final system therefore needs project-specific HoloLens recordings with synchronized IMU and FSR signals to cover the required classes.

Chapter 3

Project Scope

This chapter outlines the focus of the project, including the necessary delimitations. It introduces the proposed vision–sensor system for exoskeleton-assisted walking, describes the available hardware and data flow, and narrows the project around the dataset and training challenges identified in the problem analysis.

3.1 Vision–Sensor System for Exoskeleton-Assisted Walking

The Vision–Sensor System for Exoskeleton-Assisted Walking (VSS-EAW) is the project system concept for environment perception. It combines Microsoft HoloLens 2 as the visual acquisition device with an IMU–FSR sensor strap that records leg-motion and contact-related data. The HoloLens 2 provides RGB images of the walking environment, while the IMU and FSR sensors provide motion and pressure information from the user's lower limb.

The central design idea is feature fusion. Image features extracted from the visual model are combined with sensor features extracted from the IMU and FSR streams. This fused representation is then used to estimate the walking environment class. The goal is not to control the exoskeleton directly in this project, but to develop and evaluate an environment-recognition module that can output confidence values for walking down stairs, walking on flat ground and walking up stairs.

Figure 3.1: VSS-EAW environment setup for exoskeleton-assisted walking perception.

3.1.1 Microsoft HoloLens 2

Microsoft HoloLens 2 is selected as the visual device because it can be worn by the user and can capture the scene from an egocentric viewpoint. According to Microsoft's hardware specification, HoloLens 2 includes an 8 MP photo camera with 1080p30 video capability, four visible-light cameras for head tracking, a 1 MP time-of-flight depth sensor, eye-tracking infrared cameras and an integrated IMU containing an accelerometer, gyroscope and magnetometer [10]. It also supports Wi-Fi communication, which is relevant for transferring recording data to the PC used in the training workflow [10].

In this project, HoloLens 2 is delimited to RGB acquisition. The project uses exported personal-video frames and frame metadata rather than implementing a complete real-time mixed-reality application on the headset. Although HoloLens 2 can capture video at 30 FPS, the training datasets are sampled at lower rates so that images can be synchronized with the external sensor logs and processed efficiently.

Figure 3.2: Microsoft HoloLens 2 used as the visual acquisition device.

3.1.2 IMU–FSR Sensor Strap

The IMU and FSR sensors are introduced in section 2.1, and their working principles are not repeated here. Within the project scope, the IMU–FSR strap is treated as an external sensor system that records motion and pressure channels during walking. The sensor electronics communicate with an ESP32, which transfers the collected signals to the PC. These sensor logs are then used together with the HoloLens 2 RGB frames during dataset construction and model training.

This setup creates a practical synchronization challenge. The visual stream is recorded through the HoloLens 2 and transferred over Wi-Fi, while the sensor stream is collected through the ESP32-to-PC communication path. Since these streams are not produced by the same device clock or communication channel, the project must align them in post-processing before they can be used as a fused training input.

3.2 Dataset Limitations and Model Selection

The dataset analysis in subsection 2.2.1 showed that existing exoskeleton vision datasets contain many labelled images that are useful for model training, but that specific classes are still missing for this project. ExoNet provides broad egocentric walking-environment labels, while StairNet improves the stair-specific label quality. However, StairNet is still mainly useful for learning generic stair-versus-level-ground features, because the available stair images do not provide the separate down-stair class required by this project.

For this reason, the project uses StairNet as a pretraining source. In this context, pretraining means that the visual backbone is first trained on a related source task, namely binary `stair` versus `flat` recognition. The learned visual features are then transferred to the project-specific task and fine-tuned using HoloLens 2 images. The final HoloLens-based dataset must contain the three target classes `down`, `flat` and `up`, because the exoskeleton perception module needs to distinguish walking down stairs from walking up stairs.

The model selection is also delimited by the findings in section 2.2 and Table 2.1. Since the system

is intended for wearable or edge deployment, the visual model should be lightweight, trainable on the available data and compatible with feature fusion. MobileNetV3 is therefore selected as the training model. It provides a mobile-oriented visual backbone, can be initialized from the StairNet pretraining task, and can be modified so that image features can be concatenated with IMU and FSR features before classification.

3.3 Problem Statement

The motivation to give an exoskeleton environment-awareness served as the starting point for the problem analysis in chapter 2. In that chapter, the relevant sensors were introduced, and several visual models and datasets were compared. These findings are now delimited to fit the project focus, from which the following problem statement is formulated.

How can a MobileNetV3-based visual system be combined with an IMU–FSR sensor system to generate an environment-state assessment for exoskeleton environment perception?

3.3.1 Objectives

Objective 1: RGB dataset generation

Generate an RGB dataset containing the three environment classes required by the project: walking up stairs, walking on flat ground and walking down stairs. The dataset should be formatted so it can be used with the MobileNetV3 training pipeline, including class-specific train, validation and test splits and image preprocessing compatible with the model input. This dataset is therefore used to train, validate and test the exoskeleton's visual environment-recognition capability.

Objective 2: Synchronization of sensor data and RGB data

Synchronize the IMU–FSR sensor data with the HoloLens 2 RGB data. The sensor data are collected through ESP32 communication with the PC, while the HoloLens 2 recordings are transferred through a different device and communication path. These two acquisition routes cannot guarantee native synchronization. Therefore, the dataset builder should align image frames and sensor samples by timestamp, so that both streams describe the same walking state when used as model input.

Objective 3: Synchronized model output

Train a model that uses synchronized RGB and sensor features to output an environment-state assessment. The output should estimate confidence values for `down`, `flat` and `up`. Since real-time exoskeleton activity recognition has been demonstrated with an overall latency of 0.506 s per recognition, corresponding to approximately 2 Hz, the perception module should target at least 3 Hz and preferably 5 Hz to provide a faster update rate than this published real-time reference [11]. In the current project implementation, 3–5 Hz is treated as the relevant operating range for synchronized perception output.

Objective 4: Testing of the perception system

After training, evaluate the model accuracy on held-out test data. The test procedure should include sufficient ablation experiments to identify which subsystem combination is most useful for exoskeleton environment recognition. The relevant comparisons include RGB-only, sensor-only, RGB+IMU,

RGB+FSR and RGB+IMU+FSR configurations, as well as nearest-sample and window-based sensor features.

Chapter 4

Implementation

4.1 System Overview

The implemented pipeline is organized as an offline data and training flow. Recordings are first exported from the HoloLens 2 format, then converted into RGB-only and synchronized RGB+sensor datasets, and finally used to train and evaluate classifiers.

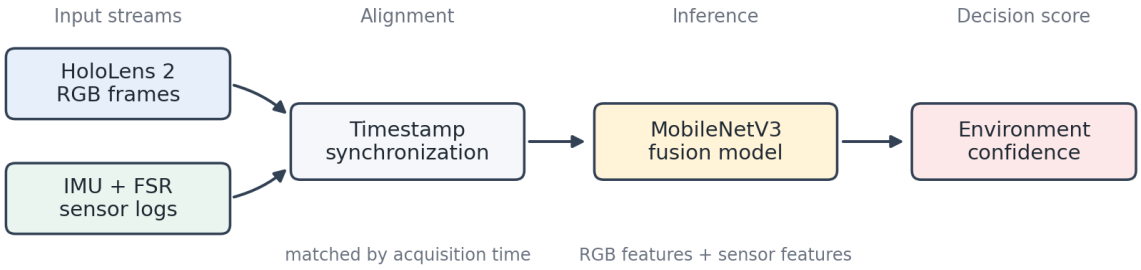


Figure 4.1: Implemented perception pipeline.

The final output is a three-dimensional confidence vector:

$$\mathbf{p} = [p_{\text{down}}, p_{\text{flat}}, p_{\text{up}}], \tag{4.1.1}$$

where the elements are produced by a softmax layer. The largest element is used as the predicted environment class.

4.2 RGB Dataset Generation

As described in the previous chapters, the project uses two RGB data sources. The first source is a StairNet subset used for visual pretraining. The second source is the project-specific HoloLens 2 dataset used for the final three-class exoskeleton environment-recognition task. The purpose of this section is to describe how these RGB datasets were prepared before synchronization with IMU and FSR data.

4.2.1 StairNet Pretraining Subset

A 20,000 image subset was sampled from StairNet and converted into a binary pretraining dataset with the labels `flat` and `stair`. The dataset is split according to a train:validation:test ratio of 8:1:1. Transition classes were excluded so that the pretraining task focuses on learning general visual differences between level ground and stairs. The split is balanced and isolated by video ID.

Table 4.1: StairNet binary subset used for visual pretraining.

Split	Flat	Stair	Total
Train	8000	8000	16000
Validation	1000	1000	2000
Test	1000	1000	2000
Total	10000	10000	20000

The binary model is not directly suitable for the final task because it cannot distinguish walking up stairs from walking down stairs. It is used only to initialize the visual backbone with stair-relevant features.

4.2.2 Project RGB Dataset

The project-specific RGB dataset was generated from HoloLens 2 recordings. The HoloLens 2 captures the walking environment as a video stream from an egocentric viewpoint. The raw recordings are stored in session folders containing `hololens2_raw_session`. The export script creates `pv.mp4` and frame metadata, after which the RGB dataset builder samples image frames from the exported videos.

The resulting RGB images are organized into the three target classes `down`, `flat` and `up`. These classes correspond to walking down stairs, walking on flat ground and walking up stairs. During training, the images are transformed to match the MobileNetV3 input pipeline, including resizing/cropping to the model input size and applying the same normalization used by the visual backbone. The RGB-only dataset contains 651 sampled frames from 15 sessions, as shown in Table 4.2.

Table 4.2: RGB-only dataset frame counts.

Split	Down	Flat	Up	Total
Train	104	138	144	386
Validation	66	45	32	143
Test	35	44	43	122
Total	205	227	219	651

Because the HoloLens 2 records a continuous video stream, some frames occur during transitions between stair walking and flat-ground walking. These transition frames can appear in both up-stair-to-flat and down-stair-to-flat recordings, and they may not visually represent the folder-level class label. Incorrect labels and ambiguous transition frames can reduce the model's ability to learn a stable environment boundary.

To address this issue, the project uses a frame-level label correction file. Ambiguous transition intervals are discarded when they do not clearly belong to one of the three target classes. For frame ranges that were manually inspected and found to have the wrong label, the correction file overrides the original label derived from the recording folder. This keeps the final RGB dataset closer to the intended visual classes before it is used for training, validation and testing.

4.3 Sensor Data and RGB Synchronization

The synchronized dataset combines the HoloLens 2 RGB stream with the measurements from the IMU–FSR sensor band. Each recording session contains one exported RGB video, one frame metadata file and two sensor log files. The sensor logs are stored as time series and each sample is associated with a relative recording time. This time field is the basis for synchronizing the body-worn sensor measurements with the video frames.

4.3.1 Measured Sensor Signals

The IMU log describes the movement and orientation of the lower-limb sensor band. For each sample, the recorded values include the gravity vector, angular velocity, quaternion orientation, linear acceleration and Euler angles. These values describe how the leg segment moves during walking and stair negotiation. During stair ascent and descent, the IMU signals are expected to contain different rotation and acceleration patterns because the leg is lifted, placed and loaded differently in the two motions.

The FSR log contains eight force-sensitive resistor channels. These channels measure changes in contact pressure on the sensor band. Compared with the IMU, the FSR channels provide information about loading and contact timing. During stair walking, these signals can indicate when the foot or lower limb enters a support phase and how the pressure distribution changes between ascent, descent and flat-ground walking.

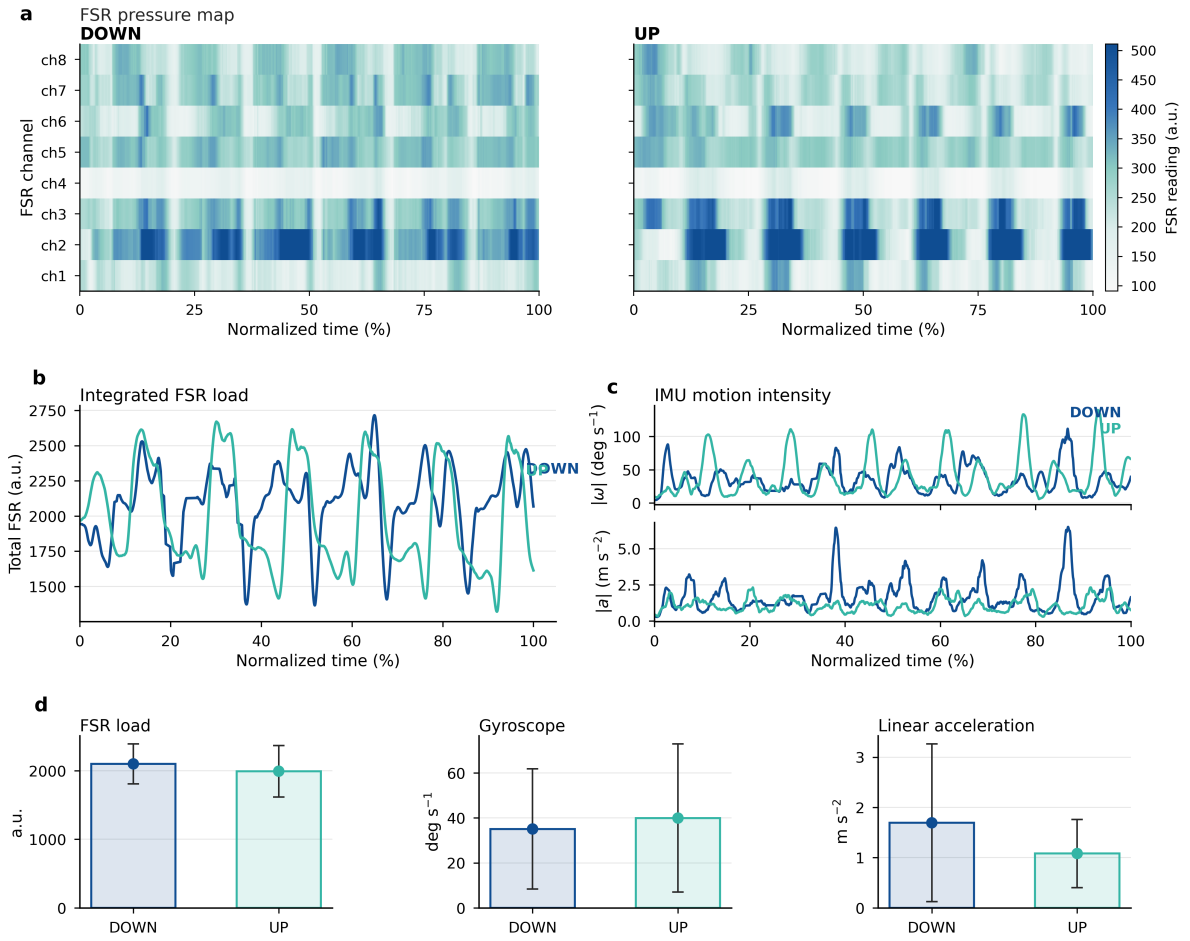


Figure 4.2: Measured FSR and IMU signals during stair descent and stair ascent.

Figure 4.2 shows that stair ascent and descent produce similar cyclic structures, but the detailed sensor patterns are different. In the FSR pressure maps, both motions show repeated loading phases, with channel 2 carrying the strongest pressure response. However, descent shows a slightly more distributed pressure pattern across several channels, while ascent contains more clearly separated high-pressure bands. The integrated FSR load is therefore broadly comparable for the two directions, with descent being slightly higher in the summarized load plot. The IMU measurements provide a clearer directional difference: ascent shows stronger and more repeated angular-velocity peaks, which is consistent with the leg rotating to lift the body onto the next step, while descent shows larger linear-acceleration bursts,

which is consistent with impact and controlled lowering when stepping down. This indicates that FSR features mainly describe the loading timing and contact pattern, whereas IMU features add motion-direction information that can help separate up-stair and down-stair walking.

4.3.2 Synchronization Method

The HoloLens 2 recording is first exported into a video file and a frame metadata file. The metadata file contains the RGB frame index, image resolution, HoloLens timestamp and a relative frame time denoted as `record_time_s`. The IMU and FSR files also contain `record_time_s`. Therefore, synchronization is performed in the recording-time domain rather than by matching frame numbers or packet indices.

For each recording, the RGB frame rows, IMU rows and FSR rows are sorted by `record_time_s`. Let the timestamp of an RGB frame be t_i^v and let the timestamps of a sensor stream be t_j^s. The nearest synchronized sensor sample is found by:

$$j^* = \arg\min_j \left| t_j^s - t_i^v \right|. \tag{4.3.1}$$

The time difference is stored as:

$$\Delta t_i = t_{j^*}^s - t_i^v. \tag{4.3.2}$$

If the absolute time difference is larger than 0.05 s, the nearest-sample feature values are not used for that frame. This prevents a video frame from being paired with a sensor sample that is too far away in time. In addition to the nearest sample, a centered 0.20 s window is extracted around each RGB frame:

$$W_i = \left\{ j \mid t_i^v - 0.10 \le t_j^s \le t_i^v + 0.10 \right\}. \tag{4.3.3}$$

For all IMU and FSR value columns inside this window, the mean, standard deviation, minimum and maximum are calculated. These window statistics preserve short-term movement and pressure context around each image frame while still producing a fixed-length feature vector for the fusion model.

The synchronization procedure therefore follows five steps. First, the HoloLens 2 video and frame metadata are loaded for each session. Second, the corresponding IMU and FSR logs are loaded and sorted by their recording timestamps. Third, each RGB frame is matched with its nearest IMU and FSR samples and with short-window IMU and FSR statistics. Fourth, frames marked as ambiguous or transitional by the frame-level label correction file are skipped, while manually corrected frame ranges receive the corrected label. Finally, the synchronized rows are sampled at 3 FPS and written to the training manifest together with the extracted RGB image path and sensor features.

The full synchronization manifest contains 6060 RGB frame rows. After sampling at 3 FPS and removing rows without usable sensor windows, the training manifest contains 596 image rows from 14 synchronized sessions.

Table 4.3: Synchronized 3 FPS dataset frame counts.

Split	**Down**	**Flat**	**Up**	**Total**
Train	107	84	147	338
Validation	62	41	33	136
Test	36	43	43	122
Total	205	168	223	596

Table 4.4: Sensor features in the synchronized dataset.

Feature group	Columns	Description
IMU nearest	17	Timestamp delta plus gravity, angular velocity, quaternion, linear acceleration and Euler-angle values.
FSR nearest	9	Timestamp delta plus eight FSR channels.
IMU window	65	Window count plus mean, standard deviation, minimum and maximum for each IMU value channel.
FSR window	33	Window count plus mean, standard deviation, minimum and maximum for each FSR channel.

4.4 Model Architecture

4.4.1 RGB Model

The RGB baseline uses MobileNetV3-Large. The final classifier layer is replaced to match the number of target classes. Training uses 224 by 224 input crops, ImageNet normalization, random resized crop, horizontal flip and mild color jitter for training. Evaluation uses resize and center crop.

The RGB-only model is trained with AdamW, a learning rate of $3 \cdot 10^{-4}$, weight decay of 10^{-4}, batch size 16 and 25 epochs. For experiments using StairNet initialization, compatible weights are loaded from the binary checkpoint and incompatible classifier-head weights are skipped.

4.4.2 Fusion Model

The fusion model combines a MobileNetV3-Large visual branch with an MLP sensor branch. The MobileNet classifier is replaced by an identity layer so that image features can be concatenated with learned sensor features. The sensor branch uses two linear layers with LayerNorm, Hardswish activations and dropout. The concatenated representation is passed through a final classifier MLP.

Three sensor-source configurations are implemented:

- `imu`: IMU features only;
- `fsr`: FSR features only;
- `both`: IMU and FSR features together.

Three feature-set options are implemented:

- `nearest`: only the nearest synchronized sensor sample;
- `window`: only window statistics;
- `all`: nearest and window features together.

A sensor-only MLP baseline is also implemented. It uses the same normalized sensor feature columns but no RGB branch. This baseline tests whether IMU and FSR can solve the current classification task without visual input.

Chapter 5

Tests

5.1 Test Strategy

The tests are designed to evaluate whether the implemented perception pipeline can classify the walking environment and whether sensor fusion improves the RGB baseline. The evaluation is divided into four parts. First, the StairNet pretraining test checks whether the MobileNetV3 backbone can learn generic stair-versus-flat visual features. Second, the transfer-learning test evaluates the three-class RGB model after the StairNet-initialized backbone is fine-tuned on the project-specific HoloLens 2 dataset. Third, the ablation test compares RGB-only, RGB+IMU, RGB+FSR, RGB+IMU+FSR and sensor-only configurations on the same synchronized dataset. Finally, the video test inspects a generated prediction video to evaluate temporal consistency in continuous video input.

The following test principles are used:

- train, validation and test splits are isolated by recording session where possible;
- transfer-learning performance is reported on the held-out HoloLens RGB test split;
- all ablation models are evaluated on the same synchronized 3 FPS test split;
- ablation results report mean and standard deviation over seeds 42, 7 and 123;
- confusion matrices are used to identify which classes are confused;
- the video test is treated as a temporal behavior check rather than a replacement for the held-out test set.

5.2 StairNet Pretraining Test

The first test evaluates the external pretraining stage. The StairNet binary pretraining task uses MobileNetV3-Large to classify images as either `flat` or `stair`. This test is important because the following transfer-learning stage depends on whether the visual backbone has learned useful stair-related features before it is adapted to the project-specific HoloLens 2 data.

The model was trained for 10 epochs. The best validation accuracy was 98.25%. On the test split, the confusion matrix in Figure 5.1 shows 991 correct flat predictions out of 1000 flat images and 952 correct stair predictions out of 1000 stair images. The resulting test accuracy is 97.15%.

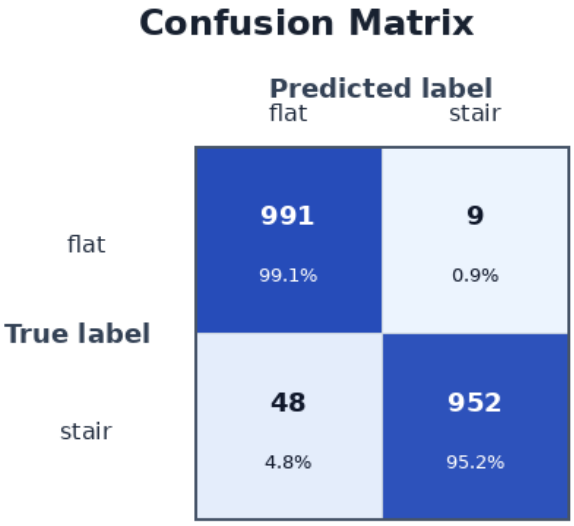


Figure 5.1: Confusion matrix for the StairNet binary pretraining run.

The result confirms that the selected backbone can learn the general visual difference between stairs and level ground from a large external dataset. However, this test only contains two classes and therefore does not solve the final project task. It is used as a feature-initialization step before transferring the model to the three-class HoloLens 2 dataset.

5.3 Transfer-Learning Test

After the StairNet binary pretraining stage, the MobileNetV3-Large backbone is transferred to the project-specific HoloLens 2 task. The binary classifier head is replaced by a three-class head for `down`, `flat` and `up`. The model is then fine-tuned on the corrected HoloLens RGB dataset.

The transfer-learning run reached a best validation accuracy of 98.60%. On the held-out test split, the model classified 118 of 122 frames correctly, corresponding to a test accuracy of 96.72%. The detailed confusion matrix is shown in Figure 5.2.

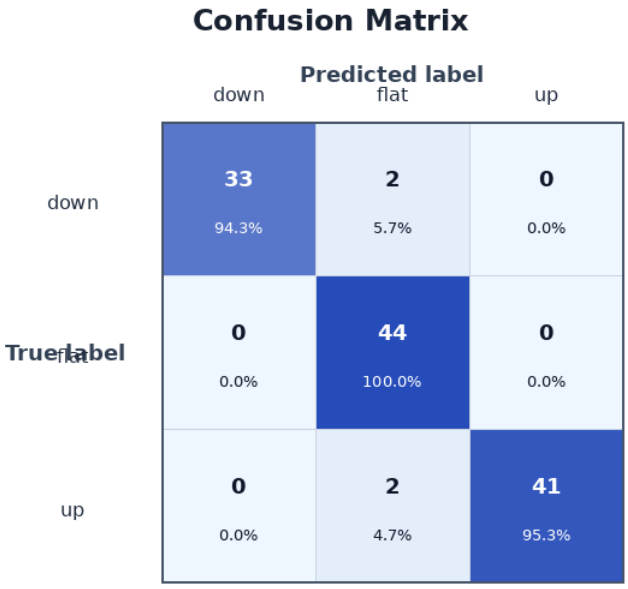


Figure 5.2: Confusion matrix for the MobileNetV3 transfer-learning test on the HoloLens RGB dataset.

The result shows that transfer learning provides a strong RGB baseline for the final environment-recognition task. The model correctly classifies all 44 flat-ground test frames. For stair descent, 33 of 35 frames are classified correctly, while two frames are predicted as flat. For stair ascent, 41 of 43 frames are classified correctly, while two frames are predicted as flat. No test frame is confused directly between `up` and `down`. This is important for an exoskeleton application because a direct up-versus-down error would be more harmful than a temporary stair-versus-flat uncertainty.

The remaining errors are mainly located at the boundary between stair walking and flat-ground walking. This matches the dataset limitation discussed in the implementation chapter: continuous HoloLens video can contain transition frames whose visual content is not always perfectly represented by a single folder-level label. The transfer-learning test therefore confirms that the visual model is effective, but it also shows why frame-level label correction and later sensor fusion are relevant.

5.4 Ablation Test

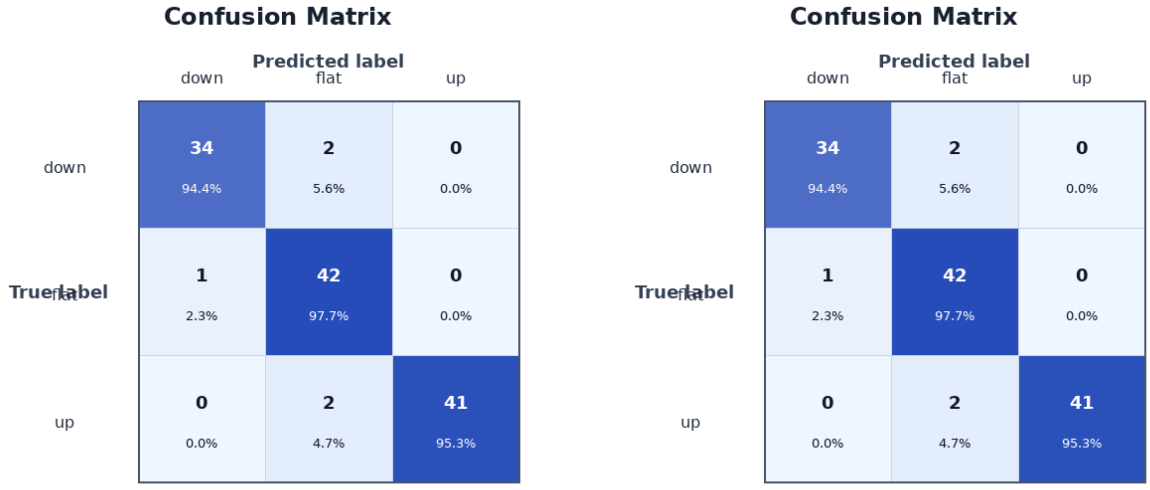
The ablation test evaluates which input combination is most useful for exoskeleton environment recognition. All ablation models use the same synchronized 3 FPS dataset and the same held-out test split. The RGB and fusion models use the StairNet binary checkpoint as visual initialization, while the sensor-only baseline uses only the synchronized IMU and FSR feature columns.

Table 5.1 summarizes the multi-seed ablation study. Accuracy and recall values are averaged over seeds 42, 7 and 123.

Table 5.1: Ablation results on the synchronized 3 FPS test split.

Experiment	Test acc.	Std.	Down recall	Flat recall	Up recall
RGB only	95.36%	0.95%	92.59%	97.67%	95.35%
RGB+IMU nearest	95.90%	0.82%	94.44%	96.12%	96.90%
RGB+FSR nearest	94.81%	1.25%	90.74%	97.67%	95.35%
RGB+IMU+FSR nearest	95.90%	0.82%	94.44%	96.90%	96.12%
RGB+IMU+FSR window	95.90%	0.82%	94.44%	97.67%	95.35%
Sensor only nearest	59.84%	1.42%	86.11%	0.00%	97.67%

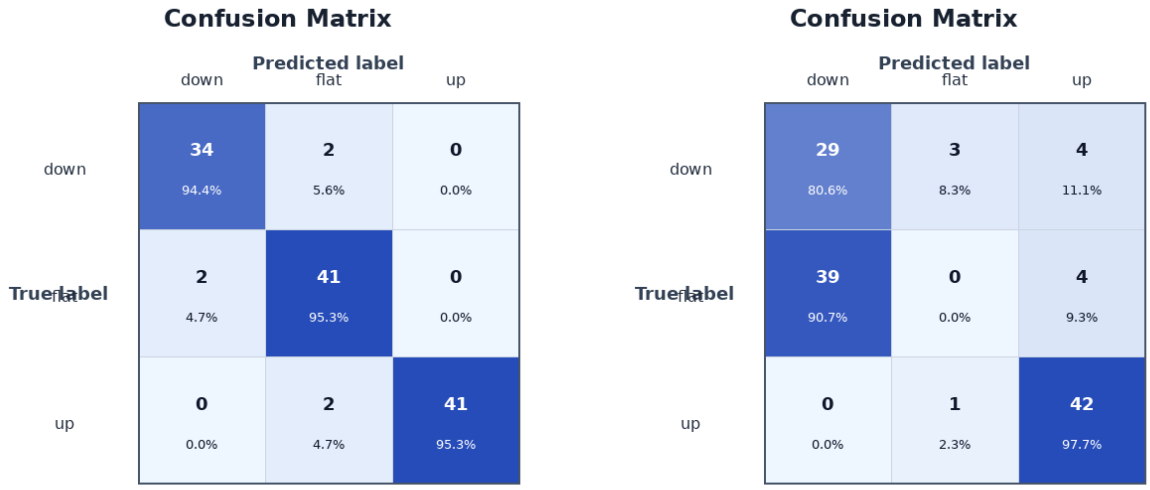
The ablation results show that RGB is the dominant signal in the current dataset. The RGB-only model already reaches 95.36% mean test accuracy. Adding IMU nearest-sample features improves the mean accuracy to 95.90%, which is a small but consistent improvement. Adding FSR alone does not improve over the RGB baseline, and the full RGB+IMU+FSR configurations match the RGB+IMU result rather than clearly exceeding it.



(a) RGB only (b) RGB+IMU nearest

Figure 5.3: Representative confusion matrices for RGB-only and RGB+IMU models, seed 42.

Figure 5.3 shows that both RGB-only and RGB+IMU nearest make few errors for seed 42. In this representative seed, both models classify 34 of 36 down frames, 42 of 43 flat frames and 41 of 43 up frames correctly. The mean over three seeds is therefore more informative than a single confusion matrix, because a difference of one or two frames can noticeably change the accuracy on the small test split.



(a) RGB+IMU+FSR window (b) Sensor only

Figure 5.4: Representative confusion matrices for full window fusion and sensor-only models, seed 42.

Figure 5.4 explains the main limitation of the sensor-only model. IMU and FSR features can distinguish many stair-related samples, but the sensor-only model fails to recall flat-ground samples in the multi-seed result. This indicates that the current sensor feature representation does not contain enough stable information to solve the three-class environment task alone. The best interpretation of the ablation is therefore that the RGB stream provides the main environment cue, while IMU and FSR can provide complementary motion/contact information. In the current implementation, the best subsystem choice is a visual model with IMU support, with the full RGB+IMU+FSR model remaining useful when pressure information is needed for later gait-phase analysis.

5.5 Video Test

The previous tests evaluate sampled image frames. A separate video test is used to inspect how the trained RGB model behaves on continuous video input. An annotated prediction video was generated for one stair-ascent recording. The video overlays the predicted class and class probabilities on each frame, and the corresponding probability timeline is shown in Figure 5.5.

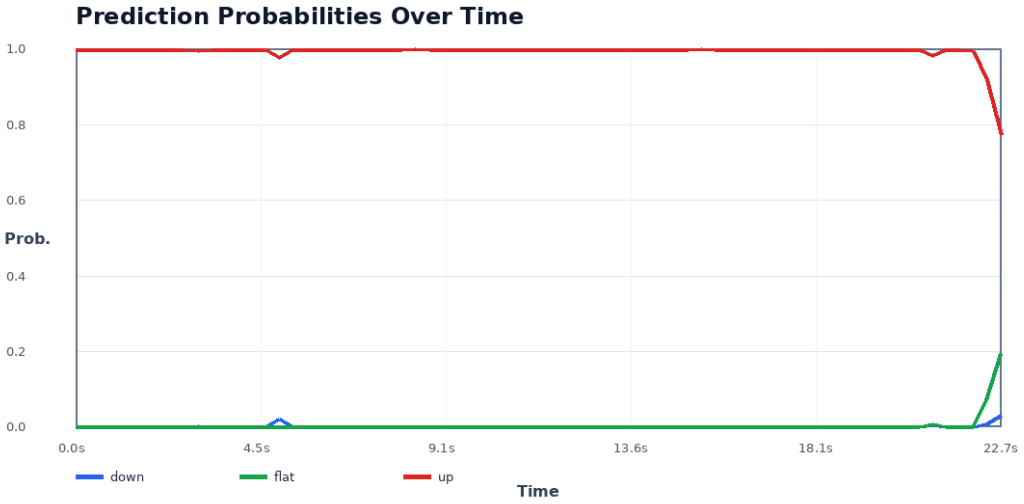


Figure 5.5: Prediction probabilities over time for the generated stair-ascent video.

The probability curve shows that the model is temporally stable during most of the stair-ascent sequence. The `up` probability remains close to 1.0 for nearly the entire video, with only a small confidence drop around the middle of the sequence. This indicates that the transfer-learned visual model can produce a consistent prediction when the user is continuously walking upstairs.

The end of the video contains a transition toward flat ground. The frame-level label correction file marks frames 650–680 of this recording as `flat`. In the probability plot, the `flat` probability increases near the end of the video, while the `up` probability decreases. However, `up` still remains the maximum-probability class. If the corrected frame-level labels are used as the reference, the sampled video contains 69 evaluated frames: 65 `up` frames and 4 corrected `flat` frames. Since the prediction remains `up`, the estimated frame-level video accuracy is 65/69, or 94.2%. For the stair-ascent portion alone, the accuracy is 100%.

This video test supports the image-based test results: the model is reliable during clear stair-ascent motion, but transitions between stairs and flat ground are more difficult. For a real exoskeleton assistance system, this means that frame-by-frame classification should be combined with temporal smoothing, confidence thresholds or sensor-fusion logic before the output is used by a controller. The video test is therefore useful as a qualitative and temporal validation step, while the held-out image and ablation tests remain the main quantitative evaluation of model accuracy.

Chapter 6

Discussion

6.1 Interpretation of Results

The experiments indicate that visual information is the dominant signal for the current dataset. After StairNet pretraining, the transfer-learned RGB model reached 96.72% test accuracy on the project-specific HoloLens RGB dataset. In the synchronized ablation study, RGB-only classification reached 95.36% mean test accuracy, and the best fusion variants reached 95.90%. The improvement from adding IMU and FSR is therefore real but small in the available experiments. This can be interpreted in two ways. First, the HoloLens 2 RGB stream already contains strong cues about stairs and flat ground. Second, the current dataset may not be large or diverse enough for the sensor branch to learn a consistently stronger complementary representation.

The video test supports the same interpretation. During the clear stair-ascent part of the generated prediction video, the model output remained stable and the `up` confidence stayed close to 1.0. Near the end of the sequence, where the recording transitions toward flat ground, the `flat` confidence increased but did not become the final predicted class. This shows that the model can behave consistently on continuous video, but it also confirms that transition frames are the most difficult part of the task.

The sensor-only baseline is useful because it prevents an overly optimistic interpretation of the sensor data. Although the sensor-only model has high mean recall for `down` and `up`, it has 0.00% mean recall for `flat`. This suggests that motion/contact features alone are not reliable enough for the current three-class environment classifier. A future sensor model may need temporal sequence modelling, gait-phase information, better normalization across users or more balanced flat-ground motion examples.

6.2 Dataset Limitations

The synchronized dataset used for the final experiments contains 596 sampled frames. This is enough to test the implemented pipeline, but it is small for drawing general conclusions about exoskeleton perception. The test split contains 122 frames, so a difference of one or two misclassified frames can change accuracy by around one percentage point. For that reason, the multi-seed mean is more important than a single run.

The session-level split reduces frame leakage, but it does not prove that the model generalizes to all users, buildings, lighting conditions or camera mounting positions. The StairNet subset helps the visual model learn stair/flat features, but it does not contain the final `up` and `down` direction labels. The final direction distinction still depends on the local HoloLens recordings.

The video test should also be interpreted carefully. It is useful for inspecting temporal behavior and confidence trends, but it is based on one selected stair-ascent recording and is not an independent replacement for the held-out test split. Its main value is showing how the classifier behaves during a continuous transition, where frame-level labels can change faster than the model confidence.

6.3 Sensor Fusion Limitations

The implemented fusion model uses nearest samples and simple window statistics. This is a good first representation, but it discards temporal order inside each sensor window. Stair ascent and descent are dynamic activities, so a sequence model such as a temporal convolution, recurrent model or transformer-style encoder may extract information that a static MLP cannot.

The current confidence output is also uncalibrated. A model can be accurate while still assigning overconfident probabilities. Before using the confidence vector in an exoskeleton control loop, calibration and temporal smoothing should be tested. A practical system should also include an uncertainty or rejection mechanism for frames that are blurred, occluded or outside the training distribution.

6.4 What Has Not Been Claimed

The report does not claim that the model is ready for safety-critical exoskeleton control. It also does not claim that the system estimates step geometry, predicts gait phase or selects actuator torques. The completed contribution is an implemented and evaluated perception pipeline that estimates a high-level environment belief from HoloLens 2 RGB frames and synchronized IMU/FSR measurements.

Chapter 7

Conclusion

This project implemented an environment perception pipeline for an exoskeleton scenario using HoloLens 2 RGB data together with synchronized IMU and FSR measurements. The perception task was defined as a three-class confidence estimation problem for walking up stairs, walking down stairs and walking on flat ground.

A 20,000 image StairNet subset was used to pretrain a MobileNetV3-Large backbone for binary stair/flat recognition. The project-specific HoloLens recordings were then converted into RGB-only and synchronized RGB+IMU+FSR datasets. The final synchronized dataset contains 596 sampled 3 FPS frames from 14 synchronized sessions, with each image paired with nearest sensor values and short-window sensor statistics.

The ablation experiments show that the RGB stream is already strong for the available data. RGB-only reached 95.36% mean test accuracy over three seeds. The best fusion configurations reached 95.90% mean test accuracy, with RGB+IMU, RGB+IMU+FSR nearest and RGB+IMU+FSR window giving the strongest results. The sensor-only baseline reached 59.84% mean test accuracy and failed to recall flat-ground samples, showing that IMU and FSR alone are not sufficient in the current implementation.

The completed work therefore provides a reproducible perception pipeline and a measured baseline for future exoskeleton environment-awareness work. The most important next steps are to collect more diverse synchronized recordings, evaluate temporal sensor models, calibrate the confidence output and test how the perception module should interface with an actual exoskeleton control strategy.

Bibliography

[1] Manon Kok, Jeroen D. Hol, and Thomas B. Schön. Using inertial sensors for position and orientation estimation. *Foundations and Trends in Signal Processing*, 11(1–2):1–153, 2017.

[2] Interlink Electronics. *FSR Integration Guide*. Document part number EIG-10000 Rev. C. Accessed: 2026-05-27.

[3] Sachin Negi, Shiru Sharma, and Neeraj Sharma. FSR and IMU sensors-based human gait phase detection and its correlation with EMG signal for different terrain walk. *Sensor Review*, 41(3):235–245, 2021.

[4] Brokoslaw Laschowski, William McNally, Alexander Wong, and John McPhee. Environment classification for robotic leg prostheses and exoskeletons using deep convolutional neural networks. *Frontiers in Neurorobotics*, 15:730965, 2022.

[5] Andrew Garrett Kurbis, Dmytro Kuzmenko, Bogdan Ivanyuk-Skulskiy, Alex Mihailidis, and Brokoslaw Laschowski. Stairnet: Visual recognition of stairs for human-robot locomotion. *BioMedical Engineering OnLine*, 23(20), 2024.

[6] Chen Wang, Zhongcai Pei, Shuang Qiu, and Zhiyong Tang. Deep leaning-based ultra-fast stair detection. *Scientific Reports*, 12:16124, 2022.

[7] Chen Wang, Zhongcai Pei, Shuang Qiu, and Zhiyong Tang. RGB-D-based stair detection and estimation using deep learning. *Sensors*, 23(4):2175, 2023.

[8] Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, Quoc V. Le, and Hartwig Adam. Searching for MobileNetV3. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1314–1324, 2019.

[9] Brock Laschowski, William McNally, Alexander Wong, and John McPhee. ExoNet database: Wearable camera images of human locomotion environments. *Frontiers in Robotics and AI*, 7:562061, 2020.

[10] Microsoft Learn. *HoloLens 2 hardware*. Accessed: 2026-05-28.

[11] Ismael Espinoza Jaramillo, Jin Gyun Jeong, Patricio Rivera Lopez, Choong-Ho Lee, Do-Yeon Kang, Tae-Jun Ha, Ji-Heon Oh, Hwanseok Jung, Jin Hyuk Lee, Won Hee Lee, and Tae-Seong Kim. Real-time human activity recognition with IMU and encoder sensors in wearable exoskeleton robot via deep learning networks. *Sensors*, 22(24):9690, 2022.