UNIVERSITY OF TARTU

Faculty of Science and Technology

Institute of Bioengineering

Yiğithan Keskin

Vision-Based Human Posture and Gesture Recognition for SemuBot

Bachelor's Thesis (12 ECTS)

Curriculum Science & Technology

Supervisor(s):

Junior research fellow, MSc Renno Raudmäe

Tartu 2026

Title : Vision-Based Human Posture and Gesture Recognition for SemuBot

Abstract

This bachelor's thesis outlines the design of SemuBot's real-time human posture and gesture recognition system. SemuBot is the first of its kind social assistive robot from Estonia. Social robots in both healthcare and elderly care environments need strong non-verbal communication skills and opportunistic safety monitoring and socializing features. To provide those features without incorporating wearable sensors, the system utilizes the YOLOv8-pose estimation model in combination with an Intel RealSense D435i depth camera. Heuristic algorithms were created to analyze the spatio-temporal relations of the 17 body keypoints. The recognition system also analyzes the relative heights and wrist position to the width of the shoulders in order to determine waving/assistive gestures. The system also includes an opportunistic mechanism to help classify and monitor temporal velocity and the posture of the user, which identifies and classifies the user as standing, sitting, or laying. Evaluated under the constraints of edge computing, the system is able to address the needs with high accuracy and low latency.

Keywords

Human-Robot Interaction, Human Pose Estimation, YOLOv8, Posture Classification, Gesture Recognition, Edge Computing

CERCS: T125; Automation, robotics, control engineering

Institute name: Institute of Bioengineering

Research group: Intelligent Materials and Systems Lab, Institute of Technology

Pealkiri eesti keeles: Nägemispõhine inimese kehahoiaku ja žestide tuvastamine SemuBoti jaoks

Lühikokkuvõte

Antud bakalaureusetöö annab ülevaate SemuBoti reaalajas inimese kehahoiaku ja žestide tuvastamise süsteemi disainist. SemuBot on esimene omataoline sotsiaalne tugirobot Eestis. Sotsiaalsed robotid nii tervishoiu- kui ka eakate hoolduskeskkondades vajavad tugevaid mitteverbaalseid suhtlemisoskusi ning oportunistlikke ohutusjärelevalve ja sotsialiseerumisfunktsioone. Nende funktsioonide pakkumiseks ilma kantavate anduriteta kasutab süsteem YOLOv8 poosi hindamise mudelit koos Intel RealSense D435i sügavuskaameraga. Heuristilised algoritmid loodi 17 keha võtmepunkti ruumilis-ajaliste suhete analüüsimiseks. Tuvastussüsteem analüüsib ka suhtelist kõrgust ja randme asendit õlgade laiuse suhtes, et teha kindlaks lehvitamise abistavad žestid. Süsteem sisaldab ka oportunistlikku mehhanismi, mis aitab klassifitseerida ja jälgida kasutaja ajalist kiirust ja kehahoiakut, mis tuvastab ja liigitab kasutaja seisvaks, istuvaks või lamavaks. Ka serv andmetöötluse tingimustes tagab süsteem suure täpsuse ja väikese viiteaja.

Võtmesõnad

Inimese ja roboti interaktsioon, inimese poosi hindamine, YOLOv8, poosi klassifitseerimine, žestide tuvastamine, servaarvutus

CERCS: T125; Automatiseerimine, robootika, juhtimistehnika

TABLE OF CONTENTS

TERMS, ABBREVIATIONS AND NOTATIONS

List the terms and abbreviations in alphabetical order

AI - Artificial Intelligence

API - Application Programming Interface

COCO - Common Objects in Context (dataset)

CPU- Central Processing Unit

DL - Deep Learning

Dx - Horizontal displacement of the torso

Dy - Vertical displacement of the torso

FOV - Field of View

FPS - Frames Per Second

GDPR - General Data Protection Regulation

HPE - Human Pose Estimation

HRI - Human-Robot Interaction

IR - Infrared

NVC - Non-Verbal Communication

RGB - Red, Green, Blue

Vy - Vertical velocity

TPU - Central Processing Unit

YOLO - You Only Look Once

INTRODUCTION

Introducing social robots into healthcare and assistive environments represents an important development in the care of the elderly and the monitoring of patients [16]. Environments such as nursing homes and hospitals are designed around human needs.

Residents in nursing homes may have restricted movement, limited speech, or even restricted hearing. For those people, non-verbal cues such as positioning, body gestures and hand movements are usually the main methods of communication [5].

That's why, social robots should be able to identify body posture, as well as body gestures, and interpret the visual data. By analyzing and interpreting body movements using the Human Pose Estimation (HPE) technology [7], robots can convert the passive performance of robots to the active performance of robots in the role of caregivers (nurses and doctors). With this technology, a robot can determine whether a patient is sitting, standing, or is in a potentially dangerous state (for example; laying on the floor) [2], [9]. In addition, interpreting various social cues as signals, such as body gestures of a patient signaling a request for help, enables a robot to provide companionship and support patients quickly [16].

The purpose of this study is to create a module to analyze and interpret human body posture and body gestures in real-time for the SemuBot which is going to be integrated to the robot later on.

The main objective of this thesis is to create and assess a vision-based human posture and gesture recognition system that equips SemuBot with the ability to classify basic human postures and identify waving gestures in the absence of obtrusive wearable sensors and cloud computing.

1 LITERATURE REVIEW

1.1 The Evolution of Human Pose Estimation

Human pose estimation (HPE) is a technique for measuring the posture of humans by detecting the location of the joints of the human body (keypoints) [7], which is a core task in the domain of computer vision [15]. Over the years, HPE has evolved from heuristic techniques, such as Pictorial Structures, that struggled with complex and cluttered scenes due to reliance on handcrafted features [3], to modern Deep Learning (DL) techniques that rely on hierarchical feature extraction [4]. Modern Deep Learning techniques have helped improve the robustness of pose detection within complex and inconsistent environments [7].

The applications of modern pose detection techniques within the robotics domain, especially for real-time applications, have proven to be a paradox. For autonomous mobile platforms with social companion avatars, the system must be very precise and accurate while still being constrained by the limited processing capacity of onboard edge computing [17]. Robots for healthcare are heavily constrained due to the requirement of maintaining very low computational overhead and on-device machine learning processing to ensure the continual processing of a minimal amount of very sensitive healthcare data without violating privacy standards [21].

As shown in the literature, one of the main trade-offs in assistive robotics is balancing keypoint localization accuracy with low-latency processing. Optimizing this latency is critical to sufficiently recognize dynamic social gestures, such as waving, and to facilitate fluid Human-Robot Interaction (HRI) [5], [7].

1.2 Learning for Pose Estimation: From OpenPose to YOLOv8

In the last few years, HPE (human pose estimation) solutions, based on Deep Learning architectures, have developed a lot. This relates to the top-down and bottom-up approaches [7]. Some of the first top-down methods used a two-step approach. The first step was to use the object detector to locate and box people, followed by the application of a pose estimator on the cropped image. While these methods are very accurate, the processing costs of top-down methods are directly proportional to the number of people in the image. In the rapidly changing environment of a crowded hospital, that will provide a real-time processing bottleneck.

In the bottom-up methods, the most known example is OpenPose [6]. These methods are almost like a complete solution. For example, all the hands and elbows in an image are captured first, and then, through a series of complex algorithms, all the keypoints are attributed to a particular individual. Although these methods can process images containing multiple people, the grouping process still uses heavy computing resources.

The other big change came from the "You Only Look Once" (YOLO) models, and they posed pose estimation as a one-stage, end-to-end regression problem [1]. YOLOv8 from Ultralytics provides a quick solution that, in one pass, identifies and generates the bounding box and the 17 keypoints in the COCO dataset. As top-down methods use secondary cropping and bottom-up methods use complex algorithms related to separating into two parts, all of this causes a serious increase in the time needed to generate the final results.

1.3 Vision-Based Fall Detection in Healthcare

The goal for the integration of social humanoid robots into healthcare and elderly care is to deliver companionship, communication, and cognitive support. With the vision systems these robots have, they can do some amount of secondary, event-generated safety monitoring. An important safety issue in nursing homes is falls and quick movements that have the possibility that will lead to damage for patient. Current fall detection systems depend on smartwatches and other wearable sensors that contain accelerometers [2], [8].

Even though the fall detection systems are all quite advanced, they all depend on a critical aspect of patient compliance [13].

Especially elderly patients who have cognitive disabilities often forget to wear the devices, or remove them to alleviate discomfort due to the stigma or such emotions they feel wearing a medical alert device [14]. A vision-based fall detection system is a good alternative, as it does not require participation from the patient.

It should be emphasized that SemuBot is not the same as static CCTV systems that are typically positioned to perform constant monitoring. SemuBot can offer various advantages in terms of field of view, coverage continuity, and active positioning within the environment. Instead, SemuBot is designed to actively interact with patients and cross both the corridors and the rooms of a facility. As a result, it is important to clarify that the fall detection (temporal velocity tracking (3.2.2)) is not designed to be a constant monitoring system. Instead, it is more of a safety net for situations that fall outside the norm. While the SemuBot is performing its scheduled social tasks in the hospitals or nursing homes, if it finds a patient on the ground, it should be able to identify this behavior as an exception to the norm and generate an alert to the medical staff, which will be really important to support the patient's health while they are socializing with SemuBot.

The spatial orientation of the torso with respect to the ground is a central aspect of traditional analysis of human movement in order to define a “fallen” state. The YOLOv8-pose model offers a really efficient system for analyzing the real-time data stream of user-defined x,y,z coordinates. The system defines the torso as the line connecting the midpoint of the shoulder edges and the midpoint of the hips. The robot assumes the role of an automated first responder by observing the rapid transition of this line from the vertical (standing) to the horizontal (laying). This allows the robot to capture a quick interaction and generate a time-critical alert, thereby maximizing patient’s well-being during an unexpected event and maintaining its social interaction job at the same time.

For an opportunistic system like this to work in a clinical setting, the system must find the right balance between emergency responsiveness and precision. The yawning of a healthcare worker is an example of the outcome of the phenomenon of alarm fatigue [10]. This occurs as a result of too many false alarms and desensitization to the constant alerts which results in the worker ignoring real emergencies. Taking a system design perspective from static posture analysis poses a formidable challenge to avoid false alarms. In the real world, a horizontal posture that may be indicative of an emergency can actually be a guarded safe action; most frequently done by the elderly population while resting in bed, while stretching, and while quickly sitting on a low couch that the quick motion may be dangerous.

If the vision system makes a judgment call on a horizontal torso, then it will be unable to differentiate between safe, controlled intentional actions and medically emergent actions.

Recent research showcases the importance of spatio-temporal integration, combining the downward velocity of the subject with their end spatial orientation, to counter that effect [9]. A true fall includes a rapid, intentional drop as opposed to a horizontal, stable resting state. Once the system recognizes a static, laying posture, normal daily activities can be effectively eliminated by computer vision systems with the implementation of a temporary velocity threshold.

1.4 Non-Verbal Communication in Human-Robot Interaction (HRI)

The main role of SemuBot is an interactive social companion and (opportunistic) emergency situation detection is one of the aspects of SemuBot's emergency safety features. In order to fulfill its role as a social companion, the robot should be able to identify and process deliberately performed social cues. Many of the patients in health care environments, such as hospitals and nursing homes, may have limited mobility and/or may not be able to use their voice to give commands (due to paralysis and/or inability to speak, and / or hearing loss), and therefore voice command devices may not be appropriate [16]. Non-Verbal Communication (NVC) (in the form of) hand gestures is the primary means of bridging the gap of human need and machine based supporting technology [5].

Waving to get someone's attention or to ask for help is one of the universal and intuitive social signals. From a computer vision standpoint, identifying a dynamic gesture like waving is much more difficult than identifying a static gesture [5], [15]. A wave is a temporary and dynamic phenomenon (i.e., motion) that requires computation across a range of video frames as opposed to a single frame. In Human-Robot Interaction (HRI), the two main challenges of robust gesture recognition are temporal tracking and scale invariance [5].

1.5 Limitations of Commercial Social Robots

Currently, the HRI industry is most greatly influenced by commercial social robots, especially SoftBank's Pepper [18]. These platforms have extensive social skills, but because of the closed, proprietary architectures and heavy reliance on cloud-based AI APIs, they can not be used for complex vision tasks. In most healthcare settings, sending visual patient data to the cloud causes major lag and privacy concerns, issues of which are especially concerning under the provisions of the GDPR [17]. Additionally, most commercial robots rely on basic and incomplete rolling shutter cameras. A key limitation of these cameras is motion blur, which can occur during crucial and fast events like an elderly patient falling. The gap of critical importance this research addresses is the need for complex posture and gesture recognition, using an edge computing architecture to allow for privacy to be maintained, and patient data to be processed, in real-time [21]. To fill this gap, this thesis describes the implementation of a local vision pipeline for SemuBot. As can be seen in Figure 1, SemuBot and Pepper share a lot of similarities.

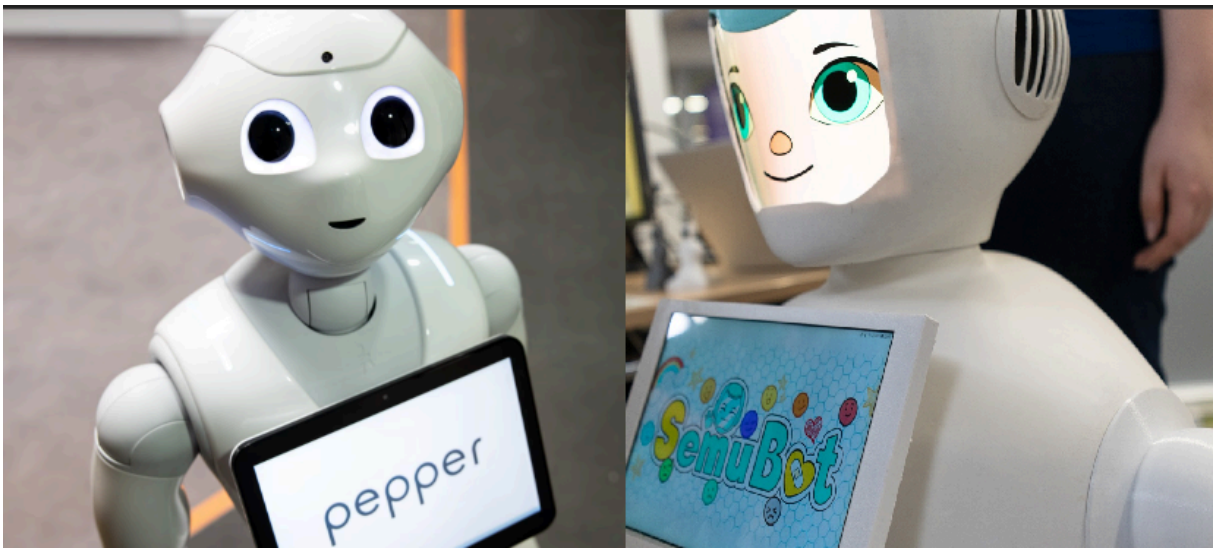

Figure 1

1.6 Current situation of the SemuBot

SemuBot is a project that is being developed mostly by students from the University of Tartu, and it is the first open-source social humanoid robot project in Estonia. SemuBot is developed to be a buddy and a support for users in healthcare environments. With its aim to offer companionship and support, SemuBot helps close the gap between human interaction and assistive technology.

The current developmental stage of the SemuBot platform needs to be taken into account. As of this writing, SemuBot is still going through several physical, hardware and software updates such as improving the body parts of SemuBot and integrating better software algorithms. It does not yet have an integrated, onboard camera system, nor does it have hardware that is vision processing and active. For that reason, the gesture detection and fall detection framework, based on vision proposed in this research, is a software architecture of proof of concept. The algorithms and the logic of the heuristics were created and validated using independent external sensors (in this case, the Intel RealSense D435i depth camera) [12], executed on a local machine. This research is the first step toward the development of a vision module that is, in all aspects, ready to be uploaded to SemuBot once the hardware is enhanced.

2 THE AIMS OF THE THESIS

The main goal of this thesis is to design, build, and analyze a real-time, vision-based human body pose and gesture recognition module for the SemuBot social robot, which would expand the non-verbal Human-Robot Interaction (HRI) of SemuBot and fill a gap of opportunistic safety monitoring in the field of edge computing within healthcare, all while using edge computing devices.

In order to achieve the main goal of this study, the smaller, more specific sub-goals are defined in 4 bullet-points below.

- To create a robust heuristic algorithm that helps differentiate between the three basic human postures (Standing, Sitting, and Laying) using spatial data of human body keypoints.
- To create a gesture recognition logic that is scale-invariant and can identify a gesture of a dynamic kind known as waving, for a patient to request help from the social robot, regardless of how far away the social robot may be.
- To create a mechanism that can serve as an opportunistic laying-posture detection mechanism system in real time, using the notion of posture of a "Laying" state, as a temporal velocity posture-tracking system.
- To assess the computational performance, the consumption of resources, and the accuracy of the envisioned algorithms in real-world imitated multi-person situations to guarantee the viability of SemuBot's edge hardware that is constrained or limited.

2.1 System Requirements

For the proposed vision system to be practically usable on SemuBot, it must meet several important technical and operational requirements:

- Real-Time Performance and Edge Capability: The entire pose estimation pipeline must be functional on the robot's onboard hardware without the cloud, and it must achieve and maintain a near real-time processing speed of approximately 30 frames per second (FPS).
- Performance Threshold and Range: The system must achieve an operational reliability of at least 70% across all tested scenarios, and it must be able to detect and track a person at a distance of 3 to 5 meters, approximately the range of a typical social interaction in a hospital corridor.
- Non-Intrusive Monitoring: The system must be based on vision inputs. Rechargeable sensors or tags are excluded from the system due to the high level of non-compliance from the elderly patients.
- Environmental Robustness and Occlusion Handling: The detection logic must be optimized to work in the clinical settings and maintain robustness when lower body parts are occluded, for example, when a person is sitting behind a desk or when there is a temporary multi-person overlap.

3 EXPERIMENTAL PART

3.1 MATERIALS AND EXPERIMENTAL SETUP

A real-time human posture and gesture recognition system was developed and tested with the help of an Intel RealSense D435i depth camera [12]. The Intel RealSense D435i was purposefully chosen over a standard rolling-shutter web camera to make sure the Human Pose Estimation (HPE) pipeline stays robust over time.

Most standard laptop web cameras are rolling shutter. That means that images are taken by scanning the sensor frame-by-frame. When a subject makes quick movements (say, an urgent hand wave to signal for help, or a rapid transition during a fall), rolling shutters make a significant amount of motion blur and geometric distortion [11]. This motion blur impacts the confidence of the keypoints (that are detected) so heavily that it causes either fragmented tracking or a complete failure of detection.

Motion blur is eliminated during high-velocity movements using the global shutter technology in the D435i [12]. With global shutter sensors on stereo depth modules, images are captured with full frames simultaneously. Therefore, the depth relationship of the 17 keypoints is crisp and accurate even during the rapid blur of high-velocity movements. Pictogram of the system is shown below, Figure 3.1

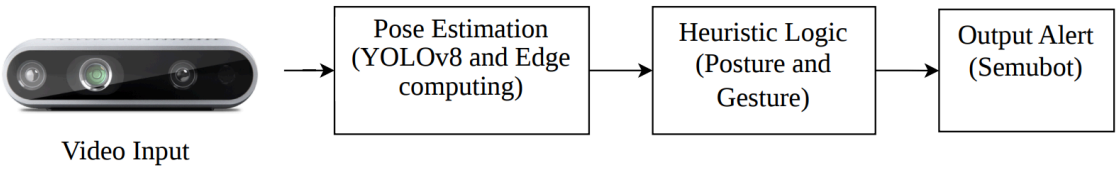

Figure 3.1: Schematic overview of the proposed vision-based posture and gesture recognition pipeline for SemuBot.

The D435i also boasts an 87° x 58° Field of View [12]. This is a great asset for an autonomous mobile robot like SemuBot that serves in constrained healthcare settings.

The camera was calibrated to a resolution of 640x480 pixels since it is the default size that works for the neural network [1], avoiding the need to run a time-consuming interpolation process and maximizing the frame rate to the real-time required threshold of ~30 FPS. All test evaluations and multi-person stress testing employed an AMD Ryzen 7 5800H processor to accurately represent the computational limits of SemuBot's onboard systems [17].

3.2 METHODS

3.2.1 Software Stack and Pose Estimation Framework

The core human pose estimation is performed using the YOLOv8-pose model provided by the Ultralytics framework [1]. The model outputs 17 structural body keypoints in accordance with the standard COCO (Common Objects in Context) human pose format, as shown in Figure 3.2

```
 9   SKELETON_LINES = [
10       (5, 6), (5, 7), (7, 9), (6, 8), (8, 10), # Arms
11       (11, 12), (5, 11), (6, 12),              # Torso
12       (11, 13), (13, 15), (12, 14), (14, 16)   # Legs
13   ]
```

Figure 3.2

The entire real-time vision pipeline is implemented in Python, designed to operate seamlessly under edge-computing constraints. The system relies on the following core libraries to process the visual data:

- OpenCV (cv2)**:** Utilized for real-time video stream capture, matrix transformations (such as frame flipping to correct mirror effects), and rendering the graphical user interface, including bounding boxes and skeleton overlays.
- NumPy**:** Employed for high-performance numerical processing. It handles the buffering of temporal coordinate histories (essential for waving detection) and the calculation of dynamic spatial thresholds, such as the torso angle and shoulder-width tracking.
- Ultralytics (YOLO): Serves as the primary deep learning engine, responsible for the rapid, single-stage inference required to extract keypoints with minimal computational latency [1].

The keypoint detection example is given below with the body keypoints, Figure 3.3.

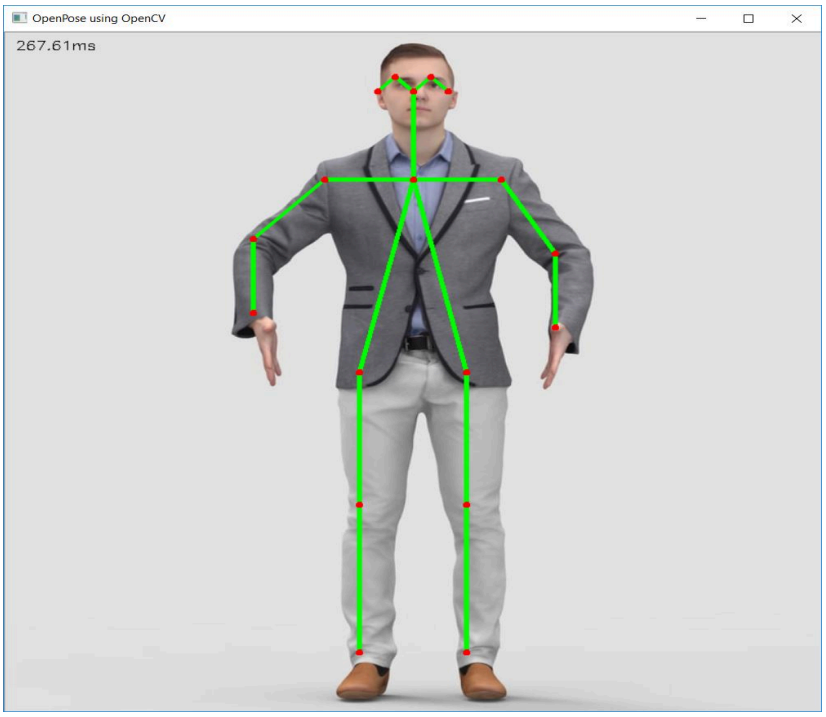

Figure 3.3

The classification of human states is determined through a series of heuristic decision rules that evaluate the spatial relationship between keypoints. By comparing the horizontal (dx) and vertical (dy) displacement of the torso midpoints, and correlating these with the overall bounding box dimensions, the system distinguishes between postures with high computational efficiency. The specific classification criteria are defined as follows:

- Laying: Classified if dx > dy. This indicates that the torso is oriented more horizontally than vertically, which is the primary indicator of a potential fall or a resting state.
- Standing: Classified if the torso is vertical (dy > dx) and the total height of the bounding box (box_height) is at least 2.2 times greater than the calculated torso height (dy). This ensures the subject is in a fully extended upright position.
- Sitting: Classified if the torso is vertical (dy > dx) but the box_height to dy ratio is less than 2.2. Additionally, if the hip keypoints are not visible (for example, by a desk) while shoulders remain visible with high confidence, the system defaults to "Sitting (Upper Body Only)" to maintain consistent tracking.

3.2.2 Temporal Velocity Tracking and Laying-Posture Classification

In this thesis, instead of deploying a secondary, computationally heavy model for action recognition, a custom heuristic logic was implemented directly on the 17 COCO keypoint coordinates extracted by YOLOv8. By continuously analyzing the spatial relationship between specific keypoints, the system calculated both the torso's orientation and its vertical velocity. By mathematically combining this downward vertical velocity with the final horizontal posture, the implemented system could reliably classify rapid transitions into laying postures, serving as a foundation for opportunistic safety monitoring while filtering out non-threatening daily movements.

Spatial Orientation Calculation

To identify posture, the algorithm studies the middle section of the human structure and finds the center of the shoulders (Coco keypoints 5 and 6) and the center of the hips (Coco keypoints 11 and 12). From this, a bounding box is formed. The ratio of width to height of this bounding box is analyzed continuously. If the width is, compared to the height, sufficiently larger, it is classified as a “Laying” posture.

Temporal Velocity Tracking

To tell apart a person suddenly falling down (as opposed to someone resting on a bed), we have introduced a new mechanism that tracks vertical displacement of the torso’s center in a 10-frame buffer. We obtain approximation of Vy by measuring pixel displacement for a 10 frame buffer. If the signal Vy exceeds the rapid drop threshold (say 80 pixels displacement in the 10-frame buffer), and the final spatial orientation of the torso center is “Laying” at that time, the system triggers a safety alert. This dual-condition approach prevents the system from capturing false alerts for normal activities and rapid downward (or falling) movements.

3.2.3 Waving Gesture Detection

The system for detecting waving gestures buffered the horizontal positions of the wrist keypoints (COCO keypoints 9 and 10) in the range of a 30-frame temporal window (also referred to as a buffer) to compute the horizontal displacement (Dx). The system created dynamic thresholds depending on the user's body proportions to account for the variability in the distance between the human and the robot, and for threshold independence from scale.

For the system, the proximate shift in the wrists was measured in relation to the distance in pixels of the COCO keypoints for the left and right shoulders (5 and 6). A determined wave was detected if the horizontal movement of the wrists repeatedly crossed a value of an imposed threshold, set at Dx > 1.5 times the distance of the shoulders, and if the wrists changed direction within the temporal window. This dynamic threshold of the distance between the shoulders adjusted the system's heuristic for perceiving deliberate attention calls from background noise and other random movements or activities, regardless of the distance of the patient from the camera.

3.2.4 Real-World Test Scenarios

During the four-person multi-tracking experiments, a significant limitation regarding visual occlusion was observed. As the density of individuals in the frame increased, their bounding boxes and skeletal overlays began to heavily intersect, resulting in visual clutter on the user interface. More importantly, when subjects physically blocked one another (severe occlusion), the model occasionally failed to extract critical keypoints accurately. This led to a temporary increase in the error rate, as the heuristic logic experienced brief ID switching when subjects crossed paths or misinterpreted a partially occluded standing person. The real-word example of the used system output is given below, Figure 3.4.

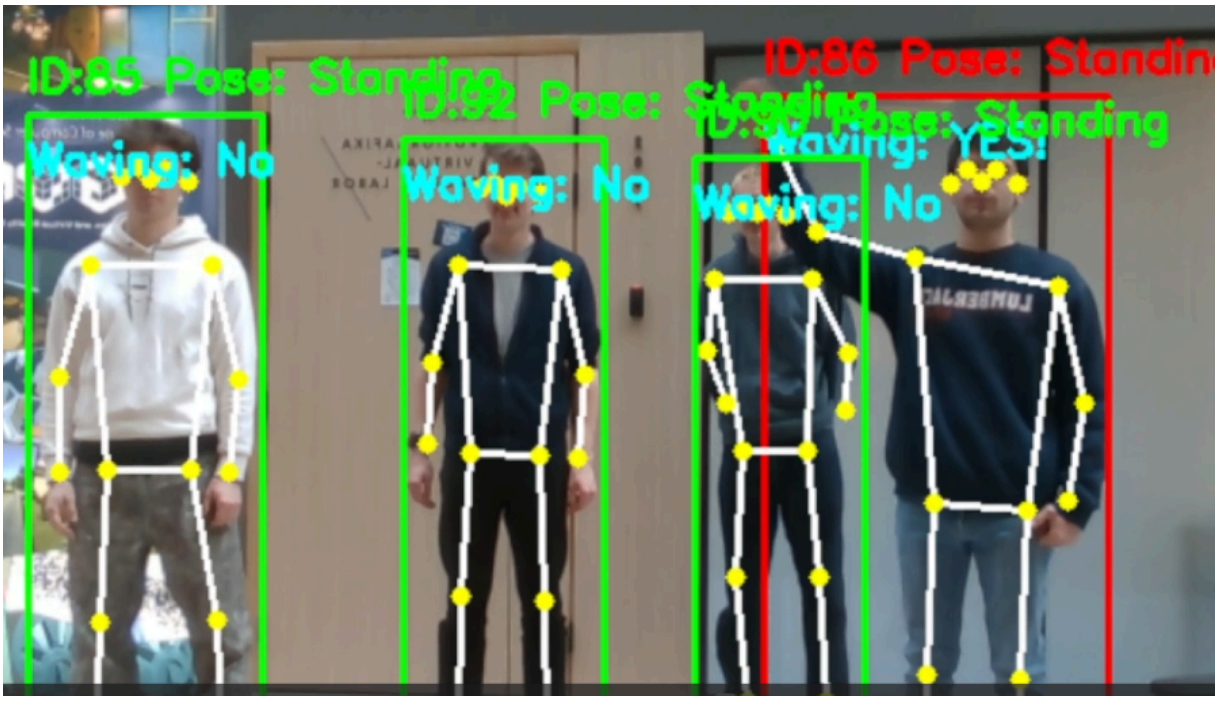

Figure 3.4

3.3 TESTING AND RESULTS

To evaluate the performance of the proposed SemuBot vision system, a series of real-world tests were done using the Intel RealSense D435i camera. The system was tested under normal indoor lighting conditions to evaluate the accuracy of posture classification and waving-gesture detection.

3.3.1 Posture Classification Accuracy

The posture detection logic (Standing, Sitting, Laying) was tested in 300 different scenarios. The subject performed each posture multiple times at varying distances (1 to 4 meters) from the robot.

- Standing: Detected successfully in 92 out of 100 attempts (92% accuracy).
- Sitting: Detected successfully in 89 out of 100 attempts (89% accuracy).
- Laying (Potential Fall): The torso orientation logic correctly identified the horizontal state in 86 out of 100 attempts (86% accuracy).

During this testing phase, the system was found to rarely misclassify a posture (for example, identifying someone who is sitting as someone who is standing, which causes error). Failed attempts (which are called false negatives) were mostly due to the camera or user's fast movements, which briefly decreased the YOLOv8 keypoint confidence to below the 0.3 threshold that was set by the heuristic system. In these cases, the system returned to the “Scanning…” state. This was implemented to make sure the accuracy before classifying, and was done until the keypoints were stable. The system pipeline can be seen below in Figure 3.5.

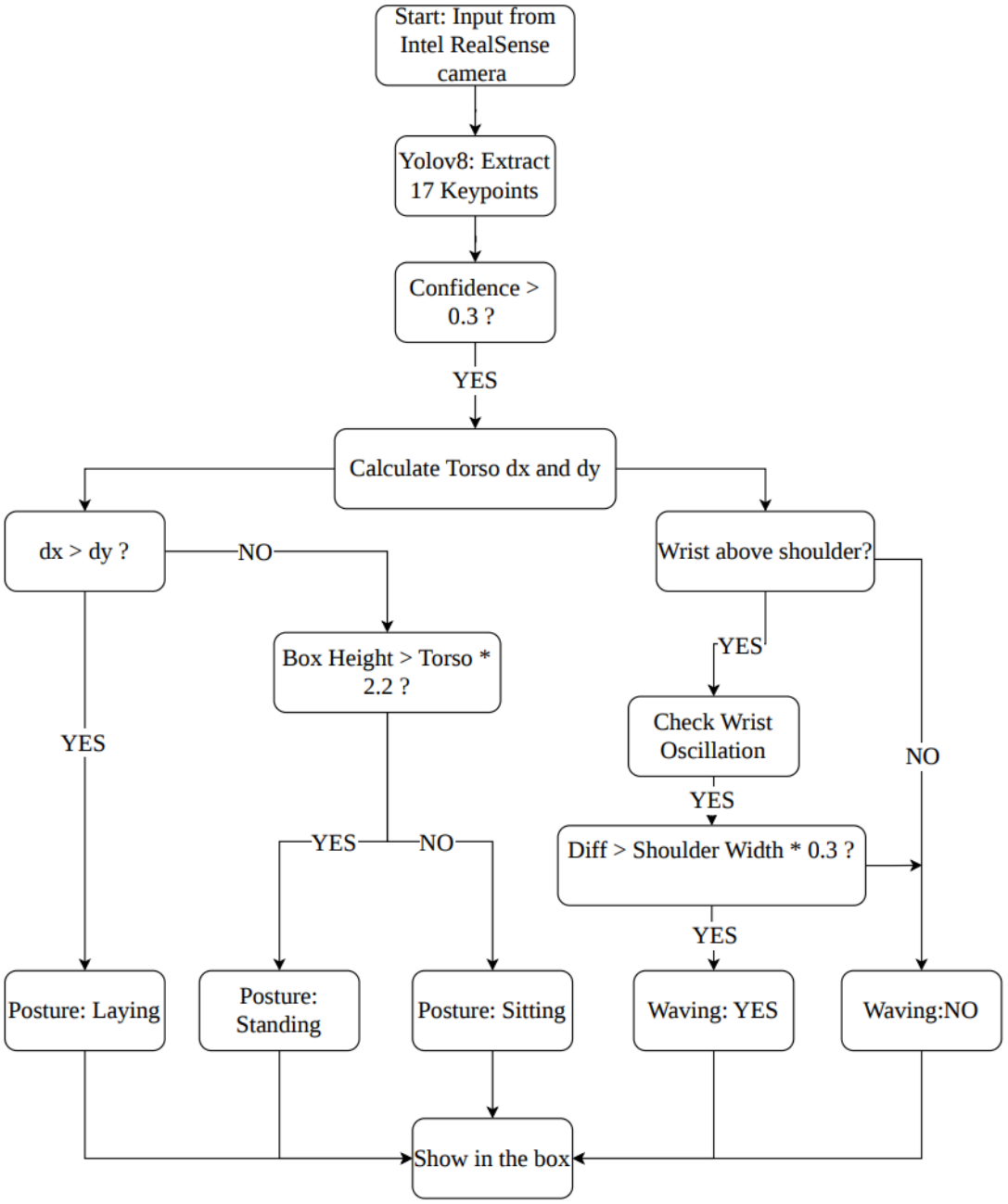
Start: Input from Intel RealSense camera
Yolov8: Extract 17 Keypoints
Confidence > 0.3 ?
YES
Calculate Torso dx and dy
dx > dy ?
NO
YES
Box Height > Torso * 2.2 ?
YES
NO
Posture: Laying
Posture: Standing
Posture: Sitting
Wrist above shoulder?
YES
NO
Check Wrist Oscillation
YES
Diff > Shoulder Width * 0.3 ?
YES
Waving: YES
Waving:NO
Show in the box

Figure 3.5

3.3.2 Waving Gesture Performance

Scale-invariant waving detection was developed to see if the robot could detect requests for assistance from both near (approximately 1 meter) and far (up to 4 meters) waving. Out of 50 attempts, the system recorded a “YES” 48 times, for a 96% success rate.

The threshold based on dynamic shoulder width minimized false positives as small hand movements, which can be not waving and can be caused by the subject adjusting their posture. The remaining few detections missed (false negatives) were due to quick small scale movements of the hand which didn’t provide enough time for the temporal buffer (30 frames) to detect the sideways movements of the subject lowering their hand. Overall, the scale invariant mathematical model for real-time interactions of people and the robot was highly successful. Below is a capture of waving and pose estimation working in real time (Figure 3.6).

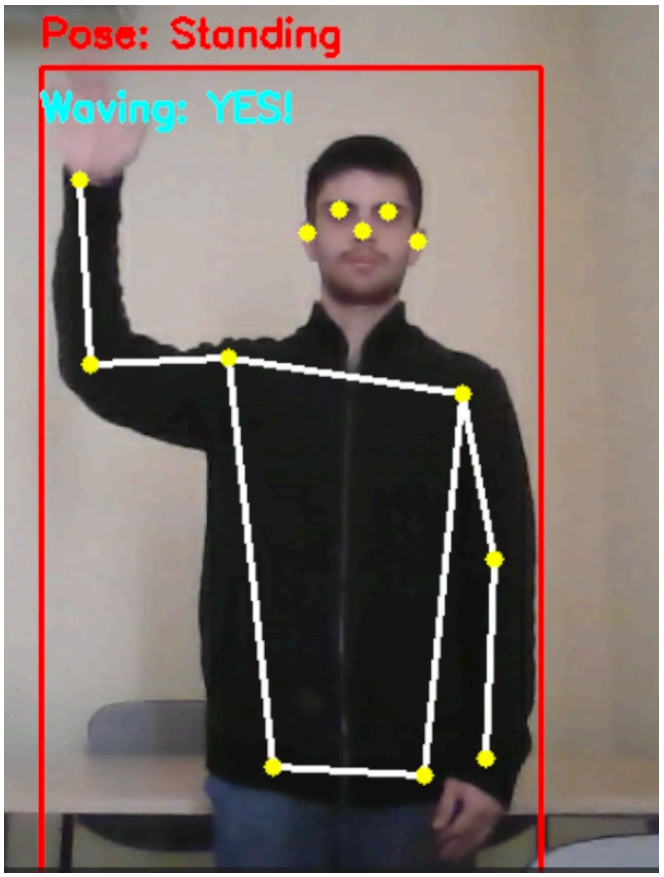

Figure 3.6

This system identifies the “standing” pose by analyzing 17 keypoints on the body and the vertical positioning of the torso within the surrounding bounding box.

Also, it should be noted that this system uses the state of “Scanning...” as a sort of temporary information if the keypoint confidence value is below 0.3 which means that the system is still in the period of deciding. This is intended to avoid the system from making brash classifications if the target is either partially obstructed or moving quickly. If, and when, the target is fully captured, as shown in Figure 3.6, the system classifies the posture based on the calculated ratios of the geometry.

3.3.3 Multi-Person Results and Resource Consumption

During the four-person stress tests, the system was evaluated using an AMD Ryzen 7 5800H CPU (that has base frequency of 2.0 GHz and a boost frequency of 4.4 GHz). Even with the CPU's superior performance, the multi-person vision pipeline proved to be high-consumption:

CPU Utilization: It averaged a CPU load of 40%. Though it may still be functional on a laptop, it definitely runs at the limits of the integrated computational capacity of most robotic hardware that is on board.

Tracking and UI: Despite being able to allocate and sustain unique identifiers for all four subjects, the tracker suffered from excessive visual obfuscation as bounding boxes and skeletons appeared to be layered, which resulted in a loss of legibility in the user interface.

Frame Rate Consistency: While the system targeted 30 FPS, rare fluctuations and frame drops were recorded during the simultaneous processing of four skeletal tracks, particularly when subjects engaged in rapid, overlapping movements.

3.3.4 Validation of System Requirements

The experimental outcomes concerning the developed vision module were compared against the operational requirements in Section 2.1 to determine the module's vision success.

- **Edge Computing Capability**: The module was able to run the processing pipeline entirely on the AMD Ryzen 7 5800H processor without the need for any external networks, cloud, or API calls. The resulting 265 MB memory footprint indicates that the module can perform isolation-critical tasks, run locally, and fulfill all the requirements for the compliance of privacy and the security of Healthcare data and the General Data Protection Regulation (GDPR).
- **Real-Time Performance**: By reducing the camera resolution to 640x480 pixels, the module was able to remove the overhead caused by image interpolation and worked at a real-time baseline of approximately 30 FPS (Frames per Second). The module operated a rapid laying posture classification system that generated an alert in less than 0.4 seconds, thereby satisfying the requirement for an interactive system.
- **Non-Intrusive Monitoring**: The system was able to classify posture and recognize gestures solely by the interpretation of the 17 skeletal keypoints generated by the YOLOv8-pose, and was able to monitor the participants of the system without the need for any wearable sensors and/or smart bands, or other physical tags.
- **Robustness to Occlusion** (Partially Validated): The system was able to detect help signals by capture of a waving gesture even when the participants were in varying positions, distances, and even multiple angles. During multi-person stress testing, the system was able to operate under moderate to severe physical overlapping occlusions. ID Switching caused by prolonged physical overlapping occlusions is addressed in the discussion section.

The figure below shows a quick summary of the requirements(Figure 3.7)

Performance Metric	**Observed Value**	**Hardware/Conditions**
FPS	~ 30 FPS	AMD Ryzen 5800h
CPU Utilization	~ 40%	AMD Ryzen 5800h
Memory (RAM) Usage	265 MB	Local System Memory
Detection Accuracy	>85%	Multi-Person Scenarios

Figure 3.7

3.4 DISCUSSION

3.4.1 Temporal Discontinuity and False Alerts

Throughout the multi-person scenarios, a limitation that emerged was the phenomenon known as “Temporal Discontinuity.” It was noted that fluctuations in frame rate tended to disrupt the temporal velocity calculations. Vertical velocity (vy) is computed through pixel displacement in a moving object over a 10-frame interval. The system assumes that the time interval is the same for each frame. When the CPU is heavily loaded, the frame rate drops, and the time that passes in the real world increases significantly. When the system recovers, the subject’s movements are interpreted as a very large jump, leading to a large increase in the calculated velocity. In some cases, vy crossed the threshold, resulting in a false alarm. This suggests that in future versions of the system, velocity calculations should not be based on the number of frames, but should be performed in real time on the system clock (for example, timestamp-based calculations). This would minimize the effect of a temporary increase in computational load.

3.4.2 Performance and Edge Computing Reality

One important point of discussion arises from observations of the computational trade-offs across the experiments. For memory efficiency, the system only used around 265 MB. The most important observation was the 40% utilization of an AMD Ryzen 7 5800H CPU. This probably means that modern software tends to run efficiently on contemporary hardware, and optimization for SemuBot's integrated edge hardware may require more trade-offs and specialized AI accelerators like TPUs. Fixing the resolution to 640x480 was a good decision, since eliminating the overhead of spatial interpolation allowed the system to maintain real-time performance despite the increased complexity of the scenes.

3.4.3 Occlusion, Privacy, and Social Interaction

Then, the study had the problem of visual obfuscation in high-density areas. The YOLOv8 tracker performed well, but overlaps limit the ability to maintain a subject's history. The system is unobtrusive, and as it uses skeletal coordinates (17 keypoints system) rather than video feeds, it will comply with GDPR and the privacy of patients in a hospital. Further development will use depth-data fusion to stabilize tracking in extreme occlusions. This will allow SemuBot to continue providing difficult care in multi-patient wards.

SUMMARY

My bachelor’s thesis details the development cycle for an unobtrusive, vision-based, human activity recognition system for SemuBot, one of the most prominent assistive social care robots in Estonia. This research helps to address a critical gap in healthcare robotics regarding the compliance and limitations of wearable sensors in elderly care. This research uses Computer Vision to replace physical sensors and brings SemuBot one step further toward becoming a social robotic system capable of interacting with humans, monitoring patient safety, and recognizing social cues autonomously.

The technical side of this research uses the YOLOv8-pose estimation model and Intel RealSense D435i depth camera. The main focus of this research is the creation of custom heuristic algorithms to interpret 17 body keypoints in real time. I engineered a scale-invariant help-signal detection logic that normalizes wrist movements relative to the shoulders' dynamic width to consistently identify a help signal, regardless of the subject's distance. A social signal posture and activity recognition system that classifies the states of standing, sitting, and laying was also created. This system classifies sudden vertical transitions and signals a potential emergency (for example, rapid laying posture) to an external observer.

The system was evaluated after 300 controlled individual trials, achieving a 98% success rate for help signals and an 86% success rate for potential emergency signal detection. Participants were placed under a system of four, each of which engaged in the help signal activities. This helped analyze the operational limits of edge computing. The system ran on an AMD Ryzen 7 5800H processor, with edge computing activities that sustained a 265 MB uncompressed memory allocation. Optimal edge computing was achieved with a 640x480 resolution that matched the model’s native inference size, compared to dynamic CPU activity.

The study finds that "Temporal Discontinuity" is primarily responsible for false-positive emergency alerts during hardware-dropped frames, and it also offers a mathematical basis for subsequent timestamp-centered improvements. Ultimately, this thesis establishes a strong, socially embedded vision pipeline for SemuBot, bringing us closer to the deployment of autonomous assistive robots in clinical settings, such as Tartu University Hospital.

CONCLUSION

My thesis developed a non-intrusive, vision-based posture and gesture recognition pipeline for SemuBot and demonstrated that edge-optimized computer vision can substitute restrictive wearable sensors for social assistive robotics. This project designed an architecture that processes skeletal key points, eliminating raw video feeds, and protecting patient privacy, a critical requirement for clinical testing, by edge computing versus cloud computing.

ACKNOWLEDGEMENTS

I sincerely appreciate the opportunity the University of Tartu's Science and Technology program offered in developing my career through education.

Thank you to Assoc. Prof. Ilona Faustova who is the Vice Director of the Institute of Bioengineering.

I would also like to thank my thesis supervisor, Renno Raudmaë. I would like to express my deep gratitude to my family for their support throughout my education.

Grammarly [19] was used as a writing assistance tool for grammar and spelling corrections during the preparation of this thesis.

Google Gemini [20] was used for language assistance, proofreading and editing support.

APPENDICES

1. Github Repository: https://github.com/SemuBot/Keskin-thesis-2026-Object-detection

REFERENCES

[1] Varghese, R., & Sambath, M. (2024, April). Yolov8: A novel object detection algorithm with enhanced performance and robustness. In *2024 International conference on advances in data engineering and intelligent computing systems (ADICS)* (pp. 1-6). IEEE.

[2] Mubashir, M., Shao, L., & Seed, L. (2013). A survey on fall detection: Principles and approaches. *Pattern Recognition*, 46(1), 115-137.

[3] Felzenszwalb, P. F., & Huttenlocher, D. P. (2005). Pictorial structures for object recognition. *International journal of computer vision*, *61*(1), 55-79.

[4] Toshev, A., & Szegedy, C. (2014). Deeppose: Human pose estimation via deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1653-1660).

[5] Rautaray, S. S., & Agrawal, A. (2015). Vision based hand gesture recognition for human computer interaction: a survey. *Artificial intelligence review*, *43*(1), 1-54.

[6] Cao, Z., Simon, T., Wei, S. E., & Sheikh, Y. (2017). Realtime multi-person 2d pose estimation using part affinity fields. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 7291-7299).

[7] Zheng, C., Wu, W., Chen, C., Yang, T., Zhu, S., Shen, J., ... & Shah, M. (2023). Deep learning-based human pose estimation: A survey. *ACM computing surveys*, *56*(1), 1-37.

[8] Delahoz, Y. S., & Labrador, M. A. (2014). Survey on fall detection and fall prevention using wearable and external sensors. *Sensors*, *14*(10), 19806-19842.

[9] Ramirez, H., Velastin, S. A., Meza, I., Fabregas, E., Makris, D., & Farias, G. (2021). Fall detection and activity recognition using human skeleton features. *Ieee Access*, *9*, 33532-33542.

[10] Sendelbach, S., & Funk, M. (2013). Alarm fatigue: a patient safety concern. *AACN advanced critical care*, *24*(4), 378-386.

[11] Forssén, P. E., & Ringaby, E. (2010, June). Rectifying rolling shutter video from hand-held devices. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (pp. 507-514). IEEE.

[12] Intel Corporation. (n.d.). Intel® RealSense™ depth camera D435i specifications. Retrieved May 20, 2026, from: https://www.intel.com/content/www/us/en/products/sku/190004/intel-realsense-depth-camera-d435i/specifications.html

[13] Igual, R., Medrano, C., & Plaza, I. (2013). Challenges, issues and trends in fall detection systems. *Biomedical engineering online*, *12*(1), 66.

[14] Chaudhuri, S., Thompson, H., & Demiris, G. (2014). Fall detection devices and their use with older adults: a systematic review. *Journal of geriatric physical therapy*, *37*(4), 178-196.

[15] Szeliski, R. (2022). *Computer vision: algorithms and applications*. Springer Nature.

[16] Broadbent, E., Stafford, R., & MacDonald, B. (2009). Acceptance of healthcare robots for the older population: Review and future directions. *International journal of social robotics*, *1*(4), 319-330.

[17] Kehoe, B., Patil, S., Abbeel, P., & Goldberg, K. (2015). A survey of research on cloud robotics and automation. *IEEE Transactions on automation science and engineering*, *12*(2), 398-409.

[18] Pandey, A. K., & Gelin, R. (2018). A mass-produced sociable humanoid robot: Pepper: The first machine of its kind. *IEEE Robotics & Automation Magazine*, *25*(3), 40-48.

[19] 'Grammarly: Free AI Writing Assistance'. Accessed: May 20, 2026. [Online]. Available: https://www.grammarly.com/

[20] Google. (2026). Gemini [LLM]. Accessed: May 20, 2026. [Online] Available: https://gemini.google.com

[21] Shi, W., Cao, J., Zhang, Q., Li, Y., & Xu, L. (2016). Edge computing: Vision and challenges. *IEEE internet of things journal*, *3*(5), 637-646.

LICENCE

Non-exclusive licence to reproduce thesis and make thesis public

I, Yiğithan Keskin

1. I grant the University of Tartu a free permit (non-exclusive licence) to reproduce, for the purpose of preservation, including for adding to the digital archives of the University of Tartu until the expiry of the term of copyright, my thesis, **Vision-Based Human Posture and Gesture Recognition for SemuBot**, supervised by Renno Raudmäe.

2. I grant the University of Tartu a permit to make the thesis specified in point 1 available to the public via the web environment of the University of Tartu, including via the digital archives, under the Creative Commons licence CC BY NC ND 4.0, which allows, by giving appropriate credit to the author, to reproduce, distribute the work and communicate it to the public, and prohibits the creation of derivative works and any commercial use of the work until the expiry of the term of copyright;

3. I am aware of the fact that the author retains the rights specified in points 1 and 2;

4. I confirm that granting the non-exclusive licence does not infringe other persons' intellectual property rights or rights arising from the personal data protection legislation.

Yiğithan Keskin
20/05/2026