

Design Specification

Gabriel Anderberg
Karl Hilding
Gustaf Härold
Klara Kastenson
Tai Ta
Michael Yasemi

December 16, 2022

Version 1.0



Status

Reviewed	Gustaf Härold	November 20, 2022
Approved	Martin Skoglund	November 20, 2022

Project Identity

Group E-mail: gusha385@student.liu.se

Homepage: <http://www.isy.liu.se/tsrt10/group>

Orderer: Martin Skoglund, Linköpings universitet
Phone: +46 13 281890
E-mail: martin.skoglund@liu.se

Customer: Sergi Rotger Griful, Eriksholm Research Centre
Phone: -
E-mail: segr@eriksholm.com

Supervisor: Johanna Wilroth
Phone: +4670-894 48 49
E-mail: johanna.wilroth@liu.se

Course Responsible: Daniel Axehill
Phone: +46 13 28 40 42
E-mail: daniel.axehill@liu.se

Participants of the group

Name	Responsible	E-mail
Tai Ta	Responsible for the Documentation (DOC)	taita693@student.liu.se
Klara Kastensson	Responsible for the Design (DES)	klaka376@student.liu.se
Karl Hilding	Responsible for the Testing (TEST)	karhi203@student.liu.se
Michael Yasemi	Responsible for the Hardware (HW)	husya078@student.liu.se
Gabriel Anderberg	Responsible for the Software (SW)	gaban592@student.liu.se
Gustaf Härold	Project Leader (PL)	gusha385@student.liu.se

CONTENTS

1	Introduction	1
1.1	Definition of terms	1
2	System overview	2
2.1	Orientation representation	4
2.2	State space model	4
3	Implementation of new hardware	6
3.1	Tobii Pro Glasses 3	6
3.2	Hearing aid with internal accelerometer	6
3.3	Bias estimations	6
4	Simulation Environment	9
4.1	Measurement model and noise distributions	9
4.2	Sound physics	9
4.3	Implementation of new sensors	9
4.4	Simulation Validation	13
4.5	Usability	13
5	Eye Tracking	15
5.1	Saccade detection	16
5.2	Fixation detection	17
5.3	Smooth pursuit detection	17
5.4	Listening detection using pupil dilation	19
6	Tracking and localization	21
6.1	Real-time optimization	21
6.2	Distance perception	22
6.3	New statistical tracking method	22

DOCUMENT HISTORY

Version	Date	Changes made	Sign	Reviewer
0.1	2022-10-10	First draft.	All	GH
0.2	2022-11-04	Second draft.	All	GH, KK
1.0	2022-11-20	First version.	All	GH

1 INTRODUCTION

As a collaboration between Linköping University and Oticon, a company developing hearing aid technology, this project has been an ongoing development process for the recent years. By further development of existing features the final goal is to have a hearing aid product better suited for noisy environments. New for this year is access to new hardware with improved capabilities. This project is part of the course TSRT10 at Linköping University and will follow the project model LIPS. The design specification document is a part of LIPS and lists the requirements for the final product.

1.1 Definition of terms

Table 1 shows the definitions of acronyms and terms used in the requirement specification.

Table 1: Definition of terms

Terms	Meaning
sim-env	The simulation environment is a combination of Gazebo with ROS, where instructions from ROS controls the Gazebo environment.
GUI	Graphical User Interface, in which the user can interact with the program.
SLAM	Simultaneous Localization And Mapping.
EKF	Extended Kalman Filter.
IMU	Inertial Measurement Unit.
ISY	The department of electrical engineering at Linköping University.
DP	Decision point.
LIPS	Project model, containing rules, instructions and templates to conduct a project.
User	Refers to the person using the Tobii Pro Glasses.
Speakers	Refers to people talking in the environment of the user.
G2	Tobii Pro Glasses 2.
G3	Tobii Pro Glasses 3.
HA	Hearing aid.
Gazebo	Open source software of a 3D-robot-simulator.
ROS	Open source software for robotic operating system.

2 SYSTEM OVERVIEW

This section describes how the different subsystems work and the cooperation between them as well as what the main focus is to improve. A more detailed explanation of how the modules and subsystem will evolve is described in Sections 3 to 6.

The hardware used in this project includes sensors used for the different modules. The system currently uses an IMU and front camera to estimate user's head orientation. Previous projects had introduced a face mesh-module to estimate more information about the environment, especially mouth movements during speaking. This project will ensure the compatibility of the modules with the new hardware and the new magnetometer found in the Tobii G3. A new HA-prototype will be introduced with an internal accelerometer for this year.

The eye tracking system module will be enhanced by considering the impact of the environments light on the pupil size. This project will also focus on the eye fixation and eye saccade concepts, in order to better interpret eye movements.

The simulation environment is made to test modules and algorithms during development to ease future testing. To get more realistic data from the simulation environment, different noise model will be included to every signal. The simulation environment will also be applicable to the new hardware for this project. For this implementation accelerometer data must be simulated in the simulation environment since the Tobii 3 glasses has a magnetometer sensor.

An overview of the system is shown in Figure 1. In the figure, blue boxes show different subsystems, green boxes existing modules and red boxes new modules.

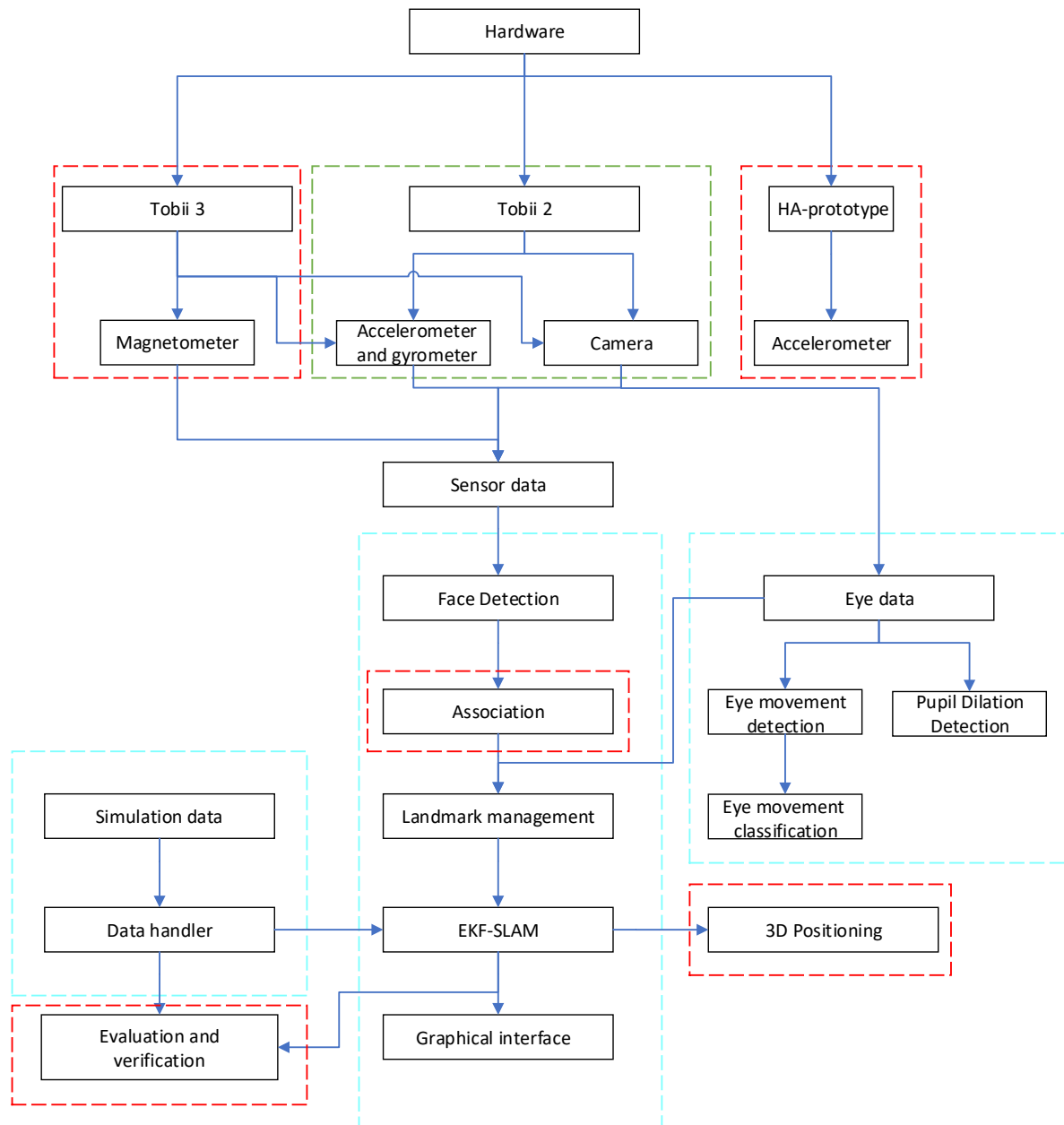


Figure 1: A schematic of the system

2.1 Orientation representation

To represent the systems multiple coordinate systems relations must be modeled as follows:

- **Global(G):** The global frame is earth fix right hand coordinate system with the z-axis parallel to gravity in opposite direction. Represented by (x_G, y_G, x_G) .
- **Body(B):** The body frame is the orientation of the object in the simulation environment and is defined as the right hand system with origin in the center of the IMU. The coordinate system is represented by (x_B, y_B, x_B) in Figure 2.

Defining coordinate frames (G) is done with unit quaternion where the orientation is described as a vector $q = [q_0 \ q_1 \ q_2 \ q_3]^T$, where q_0 is a scalar and q_1, q_2, q_3 are complex with one imaginary axis each [1]. The rotation matrix R^{GB} from the global frame (G) to the body frame (B) can be described as:

$$R^{GB}(q) = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1q_2 - q_0q_3) & 2(q_1q_3 + q_0q_2) \\ 2(q_1q_2 - q_0q_3) & q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1q_3 + q_0q_1) \\ 2(q_1q_3 + q_0q_2) & 2(q_2q_3 + q_0q_1) & q_0^2 + q_1^2 - q_2^2 - q_3^2 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \quad (1)$$

The time derivative of q can be expressed given the angular velocity $\omega = [\omega_x \ \omega_y \ \omega_z]^T$ and can then be expressed by:

$$\dot{q} = \frac{1}{2}S(\omega)Q = \frac{1}{2}\bar{S}(q)\omega \quad (2)$$

Where $S(\omega)$ and $\bar{S}(q)$ are skew-symmetric matrices:

$$S(\omega) = \begin{bmatrix} 0 & -\omega_x & -\omega_y & -\omega_z \\ \omega_x & 0 & -\omega_z & \omega_y \\ -\omega_y & -\omega_z & 0 & -\omega_x \\ \omega_z & -\omega_y & \omega_x & 0 \end{bmatrix} \quad (3)$$

$$\bar{S}(q) = \begin{bmatrix} -q_1 & -q_2 & -q_3 \\ q_0 & q_3 & -q_2 \\ -q_3 & q_0 & q_1 \\ q_2 & -q_1 & q_0 \end{bmatrix} \quad (4)$$

2.2 State space model

Using equations (3) and (4) the IMU-sensors can be modeled. For accelerometer, gyroscope and magnetometer measurements the following state space model is obtained

$$\begin{bmatrix} \dot{p} \\ \dot{v} \\ \dot{a} \\ \dot{q} \\ \dot{\omega} \\ \dot{m} \\ \dot{b}^{acc} \\ \dot{b}^{gyro} \\ \dot{b}^{mag} \end{bmatrix} = \begin{bmatrix} 0 & I & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & I & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & I & -\frac{1}{2}S(\omega) & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & I & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} p \\ v \\ a \\ q \\ \omega \\ m \\ b^{acc} \\ b^{gyro} \\ b^{mag} \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} v^a \\ v^\omega \\ v^{acc} \\ v^{gyro} \\ v^{mag} \end{bmatrix} \quad (5)$$

where p is position, v is velocity, a is acceleration, ω is angular velocity, m is a magnetic field, b^{acc} , b^{gyro} and b^{mag} is bias for the accelerometer, gyrometer and magnetometer respectively. Further v^a , v^ω , v^{acc} , v^{gyro} and v^{mag} is process noise for the acceleration, angular velocity, the acceleration bias, the gyrometer bias, and the magnetometer bias respectively.

Discretising equation (2) the following is obtained

$$q_{k+1} = q_k + \frac{T_s}{2} \bar{S}(q_k) \omega_k. \quad (6)$$

Discretising equation (5) by using (6) the discretised state space model is obtained

$$\underbrace{\begin{bmatrix} p_{k+1} \\ v_{k+1} \\ a_{k+1} \\ q_{k+1} \\ m_{k+1} \\ \omega_{k+1} \\ b_{k+1}^{acc} \\ b_{k+1}^{gyro} \\ b_{k+1}^{mag} \end{bmatrix}}_{x_{k+1}} = \underbrace{\begin{bmatrix} I & T_s I & \frac{T_s^2}{2} I & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & I & T_s I & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & I & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & I & \frac{T_s}{2} \bar{S}(q_k) & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & I & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & I & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & I & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & I & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & I \end{bmatrix}}_F \underbrace{\begin{bmatrix} p_k \\ v_k \\ a_k \\ q_k \\ m_{k+1} \\ \omega_k \\ b_k^{acc} \\ b_k^{gyro} \\ b_{k+1}^{mag} \end{bmatrix}}_{x_k} + \underbrace{\begin{bmatrix} \frac{T_s^3}{6} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{T_s^2}{2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & T_s & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & T_s & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{T_s^2}{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & I & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & I & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & I & 0 \end{bmatrix}}_G \underbrace{\begin{bmatrix} w_k^p \\ w_k^v \\ w_k^a \\ w_k^m \\ w_k^\omega \\ w_k^{bias,acc} \\ w_k^{bias,gyro} \\ w_k^{bias,mag} \\ w_k \end{bmatrix}}_{w_k} \quad (7)$$

where x_k is the discretized state vector, and w_k is the process noise.

3 IMPLEMENTATION OF NEW HARDWARE

Two new pieces of hardware are available for use in the project this year. The first of these is an updated version of the Tobii glasses called Tobii Pro Glasses 3 (G3). The new glasses have better sensors than G2 as well as a magnetometer to measure magnetic fields. The second piece of new hardware is a prototype of a hearing aid(HA) with an internal accelerometer.

3.1 Tobii Pro Glasses 3

With improved sensors and the new magnetometer, the glasses will be able to get more accurate data which can lead to improved localization and orientation estimations.

As there is a difference between the sensors in G2 and G3, all of the existing modules must be looked over in order to ensure compatibility with the new glasses. This will be done by comparing the IMU output data between the two versions and updating the code to work with both versions. It is also necessary to calculate the biases of the IMU-sensors. How this is to be done is described in Section 3.3.2 and 3.3.3.

Since the G3 also has a new magnetometer the estimation algorithms must be updated to reflect this. The magnetometer will primarily be used in order to improve the orientation estimation by comparing the measured magnetic field around the glasses to the earth's magnetic field. As with the IMU, it is also necessary to calculate the bias of the magnetometer, which is described in Section 3.3.4

3.2 Hearing aid with internal accelerometer

By introducing the new hearing aid it will be possible to improve the accuracy of the systems localization and orientation estimations. This will be done by fusing the measurements from the HAs internal accelerometer with the sensors in the Tobii glasses. The bias of the accelerometer will be calculated according to Section 3.3.3

It is also possible to use the accelerometer to detect specific user actions such as taps on the HA. This function can be used as an input to the system to manually access different modes and settings.

3.3 Bias estimations

As new sensors are to be used, it is important to calculate and compensate for any biases they may have. This can be done by taking measurements of the sensor readings while the sensor is in a known constant configuration. Once the measurements are done they can be compared to the expected values for the configuration. The bias can then be calculated as the difference by the mean from the sensor measurements and the correct values. The tests differ from each other based on which sensor is tested. The bias of each sensor will be assumed to be constant. The specific methods are explained further in Sections 3.3.2-3.3.4 below.

3.3.1 Estimators

The mean μ of each sensor is estimated using

$$\hat{\mu} = \frac{1}{n} \sum_{i=1}^n x_i. \quad (8)$$

It is also possible to estimate the variance σ^2 of each sensor using

$$\hat{\sigma}^2 = \frac{1}{n-1} \sum_{i=1}^n (x_i - \hat{\mu})^2 \quad (9)$$

where x_i is each measurement. Using the information obtained about the mean and variance of each sensor, it is possible to implement simulations of each sensor with noise in the sim-env.

3.3.2 Gyroscope bias

To estimate the bias and variance of the gyroscope sensors, the Tobii Glasses must be put laying still on a table. Since the true value of the angular velocity then is 0 m/s, the bias is then the mean of the sensor output, using equation (8), i.e

$$b^{gyro} = \begin{bmatrix} b_x \\ b_y \\ b_z \end{bmatrix} = \begin{bmatrix} \hat{\mu}_x^{gyro} \\ \hat{\mu}_y^{gyro} \\ \hat{\mu}_z^{gyro} \end{bmatrix}. \quad (10)$$

The variance of the sensor is obtained by using equation (9).

3.3.3 Accelerometer bias

To estimate the bias and variance of the accelerometer sensors, the Tobii Glasses must be put laying still on a table. The true value of the acceleration is then $a_x = a_y = 0 \text{ m/s}^2$, and $a_z = -9.82 \text{ m/s}^2$, where x , y and z are described in the sensors inertial frame. The bias is then the mean of each sensor output subtracted with the true value, using equation (8), i.e

$$b^{acc} = \begin{bmatrix} b_x \\ b_y \\ b_z \end{bmatrix} = \begin{bmatrix} \hat{\mu}_x^{acc} \\ \hat{\mu}_y^{acc} \\ \hat{\mu}_z^{acc} \end{bmatrix} - g^0, \quad (11)$$

where $g^0 = [0 \ 0 \ -9.82]^T$. The variance of the sensor is obtained by using equation (9).

It can be predicted that the bias of the accelerometer will not be easy to estimate correctly. This is because of the difficulties of knowing the exact orientation of the sensor. If it is deemed impossible to calculate the individual components of each bias, a damping constant c^{acc} will be calculated as

$$c^{acc} = \frac{9.82}{\sqrt{(\hat{\mu}_x^{acc})^2 + (\hat{\mu}_y^{acc})^2 + (\hat{\mu}_z^{acc})^2}} \quad (12)$$

3.3.4 Magnetometer bias

By describing the magnetic field as a 3D-field locally integration with *World Magnetic Model* can be used. According to this model [2] the magnetic field in Linköping is

$$B_r^0 = 49\mu T, \quad (13)$$

$$B_\theta^0 = 16\mu T, \quad (14)$$

and thus the size of the magnetic field is

$$|B^0| = 51\mu T. \quad (15)$$

The bias of the sensors can then be calculated by pointing the x-axis of the body frame to the north, see Section 2.1, as

$$b^{mag} = \begin{bmatrix} b_x \\ b_y \\ b_z \end{bmatrix} = \begin{bmatrix} \hat{\mu}_x^{mag} - B_r^0 \\ \hat{\mu}_y^{mag} - B_\theta^0 \\ \hat{\mu}_z^{mag} \end{bmatrix}. \quad (16)$$

It can be predicted that the bias of the magnetometer will not be easy to estimate correctly. This is because of the difficulties of knowing the exact orientation of the sensor. If it is deemed impossible to calculate the individual components of each bias, the expression in equation (15) will be used to instead calculate a damping constant c^{mag} to compensate with as

$$c^{mag} = \frac{|B^0|}{\sqrt{(\hat{\mu}_x^{mag})^2 + (\hat{\mu}_y^{mag})^2 + (\hat{\mu}_z^{mag})^2}}. \quad (17)$$

4 SIMULATION ENVIRONMENT

There currently exists a simulation environment developed by previous year's project groups, which is to be further developed. The main advantage of having a simulation environment is that it is easy and fast to early test new algorithms and different situations. For example, it will be used for verification of the algorithms developed in the project, but also to test what impact noise has on the validity of the estimations.

4.1 Measurement model and noise distributions

A large part of the work done in the simulation environment will cover noise and its impact on the estimations. The general measurement model is

$$y_k = h_k + e_k, \quad (18)$$

where y_k is the measured signal, h_k is the measured quantity and e_k is noise with some arbitrary distribution. All measurements will be assumed to follow equation (18). In Table 2 there is a summary of the distributions of different types of noise that will be implemented, as well as their mean and covariance.

Table 2: Noise distributions.

Noise type	PDF	Mean	Variance
Gaussian	$\frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$	μ	σ^2
Exponential	$\lambda e^{-\lambda x}$	$\frac{1}{\lambda}$	$\frac{1}{\lambda^2}$
Laplace	$\frac{1}{2b}e^{-\frac{ x-\mu }{b}}$	μ	$2b^2$

A Gaussian mixture model will also be simulated and analyzed. This will be done by combining N Gaussian distributed stochastic variables as follows

$$X = \sum_{i=1}^N \pi_i X_i, \quad X_i \sim \mathcal{N}(\mu_i, \sigma_i), \quad (19)$$

where $\sum_{i=1}^N \pi_i = 1$, and then X is a Gaussian mixture. Each mixture component represents a unique cluster specified by (π_k, μ_k, σ_k) .

4.2 Sound physics

The simulation environment should also be able to simulate several sources of noise. This will be dealt with by adding noise to each sound source that is used in the simulation environment according to equation (18). The distributions used for the noise are summarized in Table 2. The implementation of this is described in Section 4.5.

4.3 Implementation of new sensors

With the new hardware of Tobii G3 the simulation must be fit to simulate similar data as the glasses. Tobii G3 has an additional sensor, a magnetometer, that must be simulated in the Gazebo environment. This can be done by

implementing magnetic fields in the simulation environment, more precise an absolute world magnetic field must be implemented. Magnetometer, accelerometers and gyroscopes can all be used to measure motion of the glasses. The physics described in this section will have the world generated by the simulation environment's inertial frame.

4.3.1 Orientation representation for simulated environment

A representation of the simulated environment must be modeled, as an extension from Section 2.1 a new coordinate system is introduced.

A representation of the simulation environment will be modeled so that it is possible to simulate a magnetic field, acceleration, and angular velocity. Therefore as an extension of 2.1 a new coordinate system for the simulation environment is introduced.

- Environment plane(E): Has the origin in the center of the sim-env with a right hand system represented as (x_E, y_E, z_E) in Figure 2.

The relationship between the body frame (B) and the plane in the simulation environment (E) is defined by the rotational matrix R^{EB} and a translation vector $t = [x_E, y_E, z_E]^T$:

$$R^{EB} = \begin{bmatrix} \cos(180^\circ + \theta) & -\sin(180^\circ + \theta) & 0 \\ \sin(180^\circ + \theta) & \cos(180^\circ + \theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} -\cos(\theta) & \sin(\theta) & 0 \\ -\sin(\theta) & -\cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (20)$$

An illustration is shown in Figure 2.

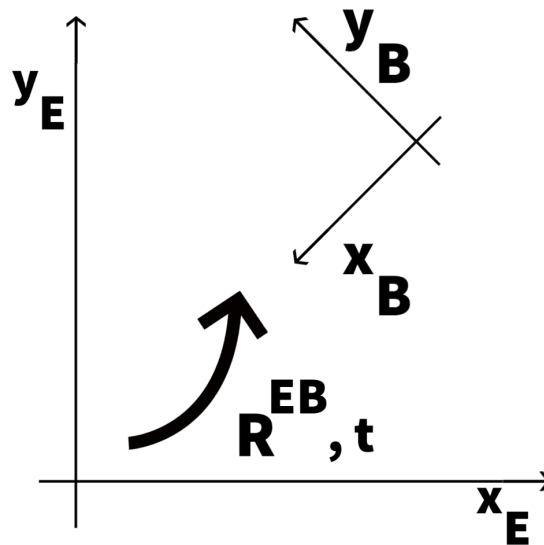


Figure 2: Rotation from the simulation environment coordinate frame to body coordinate frame.

4.3.2 Simulation of gyroscope data

The angular velocity in the simulation environment coordinate system will be simulated according to

$$\omega = \frac{r \times v}{|r|^2}. \quad (21)$$

Using Euler backward to differentiate equation (21)

$$\dot{\theta}(t) \approx \frac{\theta(t) - \theta(t - T_s)}{T_s} \quad (22)$$

where θ is some arbitrary measured quantity and T_s is the sample time, we can approximate equation (21) for the special case when it is a plane circular movement, i.e, $z \equiv 0$, to

$$\omega(t) \approx \frac{\sqrt{(x(t) - x(t - T_s))^2 + (y(t) - y(t - T_s))^2}}{RT_s} \mathbf{z}_E, \quad (23)$$

where \mathbf{z}_E is the z-vector in the environment plane, compare with Figure 3.

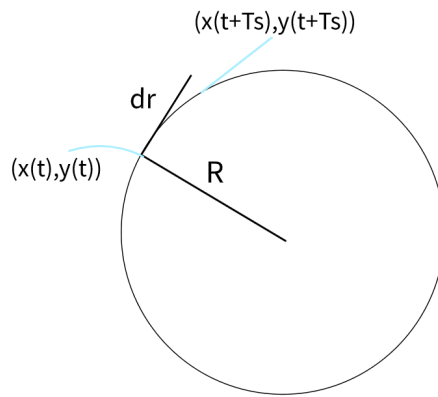


Figure 3: Angular velocity.

The measurement for the gyroscope can then be defined as:

$$y^{gyro} = R^{GB} \omega + b^{gyro} + e^{gyro}, \quad e^{gyro} \sim N(0, R^{gyro}), \quad R^{gyro} = \begin{bmatrix} (\sigma_x^{gyro})^2 & 0 & 0 \\ 0 & (\sigma_y^{gyro})^2 & 0 \\ 0 & 0 & (\sigma_z^{gyro})^2 \end{bmatrix}, \quad (24)$$

where the angular velocity ω is given from equation (23) and the variances of the noises e^{gyro} and bias estimation b^{gyro} will be estimated as discussed in Section 4.1 and 3.3.2.

4.3.3 Simulation of accelerometer data

The general form of acceleration is

$$\mathbf{a}(t) = \frac{d^2 \mathbf{r}(t)}{dt^2}, \quad \mathbf{r}(t) = x(t)\hat{\mathbf{x}}_E + y(t)\hat{\mathbf{y}}_E + z(t)\hat{\mathbf{z}}_E \quad (25)$$

where $\hat{\mathbf{x}}_E$, $\hat{\mathbf{y}}_E$ and $\hat{\mathbf{z}}_E$ are time constant unit vectors in the simulation environment's coordinate system. Iterated Euler backward as defined in equation (22) can then be used to approximate the expression in equation (25).

By discretizing equation (5) the accelerometer can be stated as equation (26). The accelerometer data from the IMU in the Tobii Glasses will be estimated using

$$\mathbf{y}_t^{acc} = R^{GB}(\mathbf{a} - \mathbf{g}) + \mathbf{b}^{acc} + \mathbf{e}^{acc}, \quad \mathbf{e}^{acc} \sim N(0, R^{acc}), \quad R^{acc} = \begin{bmatrix} (\sigma_x^{acc})^2 & 0 & 0 \\ 0 & (\sigma_y^{acc})^2 & 0 \\ 0 & 0 & (\sigma_z^{acc})^2 \end{bmatrix}. \quad (26)$$

i.e the noise of each sensor is assumed to have mean 0 and to be independent of all other sensors.

Assuming a 2-dimensional circular motion, the accelerometer will measure an acceleration of

$$\mathbf{y}_{t,circ}^{acc} = R^{GB} \begin{bmatrix} R\omega_t \\ R \frac{\omega_t - \omega_{t-T_s}}{T_s} \\ -9.82 \end{bmatrix} + \begin{bmatrix} b_x \\ b_y \\ b_z \end{bmatrix} + \mathbf{e}^{acc} \quad (27)$$

i.e the biases b_x , b_y and b_z are assumed constant. The variance of the noise \mathbf{e}^{acc} and the bias of each sensor will be estimated as in discussed in Section 3.3.3.

4.3.4 Simulation of magnetometer data

The magnetic potential is given by

$$V(r) = -\frac{m \cos \theta}{4\pi r^2} \quad (28)$$

where $m = 8 \times 10^{22} \text{ Am}^2$ is the dipole moment at the earth's center, $r = 6371 \times 10^3 \text{ m}$ is the radius of the earth and θ is the angle in radians from the earth's north pole to the position of the object. The magnetic field at a given position is then

$$\mathbf{B}(\mathbf{r}) = -\mu_0 \nabla V(\mathbf{r}) \quad (29)$$

where $\mu_0 = 4\pi \times 10^{-7} \frac{\text{kgm}}{\text{A}^2 \text{s}^2}$. This can be simplified to the radial component B_r of the magnetic field

$$B^r = -\mu_0 \frac{\partial V}{\partial r} = -\frac{\mu_0 2 \cos \theta}{4\pi r^3} \quad (30)$$

and the tangential component B_θ of the magnetic field

$$B^\theta = -\frac{\mu_0}{r} \frac{\partial V}{\partial \theta} = -\frac{\mu_0 \sin \theta}{4\pi r^3}, \quad (31)$$

where t is each time instant.

The size of the magnetic field $|B|$ can also be estimated using

$$|B| = \sqrt{(B^r)^2 + (B^\theta)^2} = \frac{\mu_0 m}{4\pi R^3} \sqrt{1 + 3 \cos^2 \theta}. \quad (32)$$

The magnetometer measurements are defined as:

$$y^{mag} = R^{GB} m^G a^{acc} + b^{mag} + e^{mag}, \quad e^{mag} \sim N(0, R^{mag}), \quad R^{mag} = \begin{bmatrix} (\sigma_x^{mag})^2 & 0 & 0 \\ 0 & (\sigma_y^{mag})^2 & 0 \\ 0 & 0 & (\sigma_z^{mag})^2 \end{bmatrix}. \quad (33)$$

Where the variances of the noises e^{mag} will be estimated as discussed in Section 4.1. The bias for the magnetometer is hard to estimate. A dampening constant a^{acc} , derived in Section 3.3.4, is thereby used instead to simulate realistic data.

4.4 Simulation Validation

The new modules and algorithm will be validated in the simulation environment by simulating a defined scenario, i.e. the position of user and target as well as the movement of the target. The sensor data from real testing will be the input together with the algorithm created to replicate the real world.

All modules developed in the project shall be evaluated in the sim-env and evaluated against ground truth. This will be done by simulating IMU measurements in the sim-env by using equations (24) and (26). The magnetometer measurements will be simulated by using equation (33). This will then be used together with the already existing target tracking modules, and new modules to be developed, to evaluate the algorithms.

4.5 Usability

The usage of the GUI will make the simulation tool easier to adapt for new users as well as making it more time-efficient. Further expansion of the GUI must be made to be able to implement the new modules and sound physics mentioned in Section 4.3 and 4.2.

Sound files of different noise types must be added as a available source in the python code thus creating a sound-vector, the method is described in section 4.2. Secondly the GUI must be configured to have a selection of sound-files to run within the simulation. This can be done in the *Kivy* and *KivyMD* framework. One file is for implementing the appearance, see Figure 4 and 5 for the desired appearance and usage. The other file is a python file to implement functions that interact with the GUI.

To start up the simulation environment and the GUI it uses ROS and Gazebo simultaneous. A new bash script will be implemented and explained in the *User Manual* to ease the start up process. To start the simulation predefined test will be implemented and can be called upon directly in the terminal for a faster use.

Simulation Menu

Noice
Gaussian ▼

Objects

Name1	Edit	Remove
Name2	Edit	Remove

Run Simulation

New Object

End Simulation

Figure 4: GUI start screen.

Object

Name:

Initial position: X: Y: Z:

End position: X: Y: Z:

Movement: Linear ▼

Soundfile: speaking.mpy ▼

Objects

Name1	Edit	Remove
Name2	Edit	Remove

Back

Set Object

Figure 5: GUI creating an object in the simulation environment.

5 EYE TRACKING

Tobii Pro Glasses 2 and 3 [3] uses two cameras per eye to provide gaze data. The output data includes a time stamp, a 3D gaze vector for each individual eye, the gaze origin and a 2D-representation of the gaze. The 2D-representation uses the camera view as coordinate system. When analyzing eye movements both the gaze vectors and the 2D gaze representation can be used. By analyzing the output data five modules describing the movements and behavior of the eyes will be developed. The technical specification of the Tobii pro glasses 3 is shown in Table 3.

Table 3: Technical Specifications for G3

Eye tracking technique	Corneal reflection, dark pupil, stereo geometry
Binocular eye tracking	Yes
Sampling rate	50 Hz or 100 Hz
Calibration procedure	One point
Accuracy	0.6°
Parallel compensation tool	Automatic
Slippage compensation	Yes, 3D eye tracking model
Pupil measurement	Yes, absolute measure
Gaze recovery time	1 frame (immediate)
Blink recovery time	1 frame (immediate)

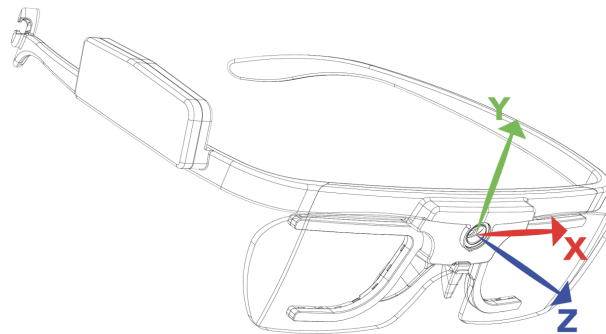


Figure 6: Display of gaze coordinate system of G2. Image rights: Tobii AB.

The coordinate system shown in Figure 6 displays the origin of the body frame coordinate system. The gaze origin describes the location of the center of the pupil for each eye in millimeters from the body fixed origin. The gaze vector describes the direction of the gaze. Gaze direction vectors of both eyes individually are obtained from the measurements, where index L represents the left eye and index R represents the right eye. The gaze vectors are normalized and has their origin in the gaze origin.

$$G_L = \begin{bmatrix} x_L \\ y_L \\ z_L \end{bmatrix} \quad G_R = \begin{bmatrix} x_R \\ y_R \\ z_R \end{bmatrix} \quad (34)$$

The gaze origin vectors provides the origin of the gaze vectors with respect to the body fixed coordinate system (in millimeters). Index L represents the left eye and index R represents the right eye.

$$G_{o,L} = \begin{bmatrix} x_{o,L} \\ y_{o,L} \\ z_{o,L} \end{bmatrix} \quad G_{o,R} = \begin{bmatrix} x_{o,R} \\ y_{o,R} \\ z_{o,R} \end{bmatrix} \quad (35)$$

5.1 Saccade detection

Saccades are rapid eye movements that shifts the center of the gaze from one point to another. Both eyes move according to the shift of gaze. Humans cannot get a clear image of the whole visual field just by fixating the eyes on a single point [4]. Therefore saccades are performed frequently to gather the visual information necessary. The initial focus is to study both horizontal and vertical saccades. For this particular case the saccade velocity (in radians per second) will be studied. A linear relation between the saccade amplitude and duration can be seen since longer saccades takes longer time to complete. By utilizing these connections together with the measurements a detection algorithm can be constructed. Here the gaze direction vectors will be utilized. The vectors provided represents the direction of the gaze at each time instant. Since the measurements are discrete, the difference in direction of the current and the previous vector will be used. Since the saccades are studied in the domain of angles a model to transfer the measured data into angular representations is constructed. The model describes the angular velocity for each eye. Index t describes the current measurement and T_s is the time between the measurements.

$$\dot{\theta}_{L,t} = \frac{\arccos(G_{L,t} \cdot G_{L,t-1})}{T_s} \quad (36)$$

$$\dot{\theta}_{R,t} = \frac{\arccos(G_{R,t} \cdot G_{R,t-1})}{T_s} \quad (37)$$

The average of the angular velocities of the eyes will be used and a moving average filter will be applied to neglect possible spikes and make the result smoother. Here t is the current measurement and N is the moving average window i.e the number of measurements taken into account. The moving average takes measurements before and after the current measurement into account.

$$\dot{\theta}_{eyes,t} = \left| \frac{1}{2(2N+1)} \sum_{k=-N}^N \dot{\theta}_{L,t-k} + \dot{\theta}_{R,t-k} \right| \quad (38)$$

The detection of a saccade will be made using a velocity threshold V_{te} . If the current average eye velocity is higher than V_{te} a saccade is occurring. When the velocity is lower than the threshold, a fixation is taking place. The velocity threshold will be decided by experiments.

5.1.1 Average saccade frequency

The saccade frequency will here be defined as the number of saccades conducted per second. Typically the frequency varies depending of the activity of the user [4]. When focusing on a single speaker for example the saccade frequency tends to drop while when performing activities that require the person keep track of the surroundings the saccade frequency tends to increase. The average saccade frequency can be set to a nominal value or decided through experiments

or being continuously updated. Since the saccade frequency depends both on the individual user and the current environment an average will be calculated during usage. This information can be used to detect deviations in the behavior of the studied person. The method used is to count the number of saccades ($n_{s,i}$) during a set time interval ($T_{s,i}$). The idea is to first establish a reliable average saccade frequency and then detect deviations quickly by comparing to the current average during a set time window. For a saccade to be counted it needs to be detected, using the earlier stated algorithm. The nominal average saccade frequency will be continuously updated during use. The nominal saccade frequency $\bar{S}_{f,n}$ at measurement n will be calculated as follows where $T_{s,tot}$ is the total measurement time and $n_{s,tot}$ is the number of completed saccades during the measurement time:

$$\bar{S}_{f,n} = \frac{n_{s,tot}}{T_{s,tot}} \quad (39)$$

The current saccade frequency will be calculated as an average over a set time interval of past measurements and will be continuously updated during use. The number of completed saccades $n_{s,i}$ will be counted during the set time interval $T_{s,i}$:

$$\bar{S}_{f,n_i} = \frac{n_{s,i}}{T_{s,i}} \quad (40)$$

5.2 Fixation detection

Eye fixation can be defined as the eye maintaining its gaze at a single position. Fixation occurs typically when a person tries to focus on a single object or entity. The fixations will be detected through the saccade detection algorithm because of the definition that when a saccade is not occurring a fixation is. During fixation micro-saccades are normally occurring and by choosing a velocity threshold high enough for the saccade detection algorithm the micro saccades will not be detected.

To save locations with eye fixation a combination of the sensor measurements are necessary to utilize. The already existing distance perception modules will be utilized. A fixation will be detected. The idea is to save the the angle of the gaze in the xz-plane as can be seen in Figure 6. The distance modules will be used to get the distance from the origin Figure 6 to the object. The location of the fixation will then be saved in global coordinates to be used in a map for future display.

5.3 Smooth pursuit detection

Besides fixation or saccades, there is also a concept known as smooth pursuit. Smooth pursuit is when the eye follows the movement of a moving object. Knowing when the smooth pursuit occurs, it is possible to use this and adjust the sound amplifier to the moving object.

The pattern of eye movement during smooth pursuit mostly looks like a ramp, with the eye position on vertical axis and time in horizontal axis [5]. However, considering that the users head is also going to move, the sum of the eye and head movement in degrees become equivalent to the targets movement in degrees.

$$\theta_{eye} + \theta_{head} = \theta_{object} \quad (41)$$

The velocity vector for object is calculated by taking difference of two measurements and dividing it by the sample time T_s .

$$S = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad \dot{S} = \begin{bmatrix} \frac{x_2 - x_1}{T_s} \\ \frac{y_2 - y_1}{T_s} \\ \frac{z_2 - z_1}{T_s} \end{bmatrix} \quad (42)$$

$$\dot{\theta}_{object} = 2 * \arctan\left(\frac{\dot{S}}{2}\right) \quad (43)$$

$$\dot{\theta}_{head} = \begin{bmatrix} \frac{\theta_{hx2} - \theta_{hx1}}{T_s} \\ \frac{\theta_{hy2} - \theta_{hy1}}{T_s} \\ \frac{\theta_{hz2} - \theta_{hz1}}{T_s} \end{bmatrix} \quad (44)$$

$$\dot{\theta}_{eye} = \begin{bmatrix} \frac{\theta_{ex2} - \theta_{ex1}}{T_s} \\ \frac{\theta_{ey2} - \theta_{ey1}}{T_s} \end{bmatrix} \quad (45)$$

The equation (41) implies the equation (46)

$$\dot{\theta}_{eye} + \dot{\theta}_{head} = \dot{\theta}_{object} \quad (46)$$

where \dot{S} represents the target velocity and $\dot{\theta}$ the angular velocity. We call the model above *Smooth Pursuit model*. To determine if an eye movement is a smooth pursuit, the model above must apply with an error less than 10%.

$$e = \|\dot{\theta}_{eye} + \dot{\theta}_{head} - \dot{\theta}_{object}\| < 0.1 * \|\dot{\theta}_{object}\| \quad (47)$$

Figure 7 shows the structure for decision making of Smooth Pursuit.

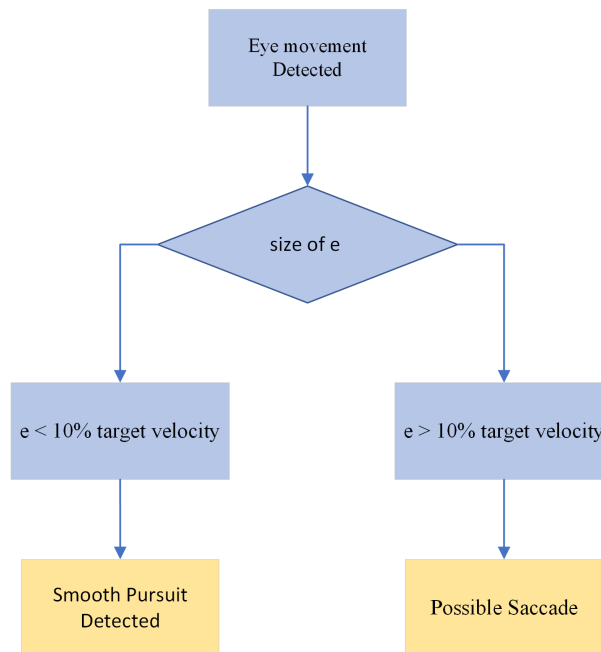


Figure 7: Decision making for Smooth Pursuit.

5.4 Listening detection using pupil dilation

The Tobii glasses provides a pupil diameter measurements in millimeters. However, the pupil size, has never been used in previous years projects. In the recent years, many studies regarding pupil dilation have been published [6]. It has been proved that the pupil dilation can have a direct link to listening [7]. The goal is to use this information to detect listening effort. Making conclusions of this detection, one can say that the user have difficulties in hearing and is making an effort to hear clearly. Using this assumption, it is reasonable to amplify the sound of interest even more. However, it is not just the effort of listening that affects the pupil size. In this work two other factors has been also considered. One, is when the user looks at different objects and is in saccade. The time eye is in saccade considered not relevant to listening effort.

Second, is the environments light, that is also an important factor in pupil dilation. To find out if the pupil dilation is a result of listening or change of environments light, the environments light intensity is going to be measured using the front camera located on the G2/G3. Figure 8 illustrates this step.

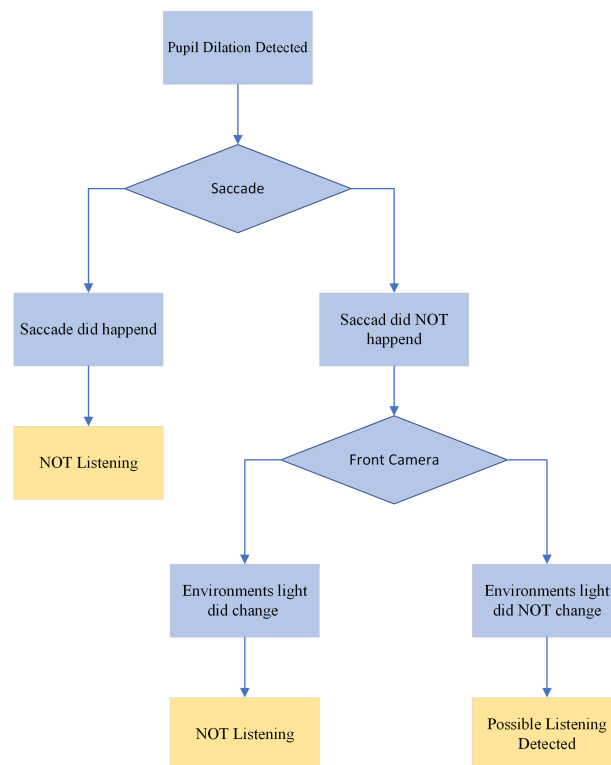


Figure 8: Decision making for Possible Listening Detection.

Additionally, the enlarged pupil size in the case of listening effort is also a function of the environment's light intensity. In other words, the pupil will enlarge differently in dark respective light environment when there is a listening effort. The mean pupil dilation in case of listening effort in the dark environment is 0.07 mm with a peak dilation of 0.16 mm. In a light environment however, the mean is 0.05 mm with a peak of 0.07 mm [7]. More information is represented in Table 4. The used light intensity in the dark in this study is 0.1 cd/m² and in the light circumstances

is 360 cd/m² [7]. However, in the real life scenario we have a range of different light intensities. Therefore, the dark environment is assumed to imply a light intensity less than 50 cd/m² and all greater than 50 cd/m² assumes to represent light environments. Figure 9 shows the final step in the listening effort detection.

Table 4: The Pupil Dilation[7]

Condition	Mean Lower (2.5%)	Mean Upper (97.5 %)	Peak Lower (2.5%)	Peak Upper (97.5 %)
Dark	0.05	0.09	0.14	0.19
Light	0.03	0.08	0.04	0.09

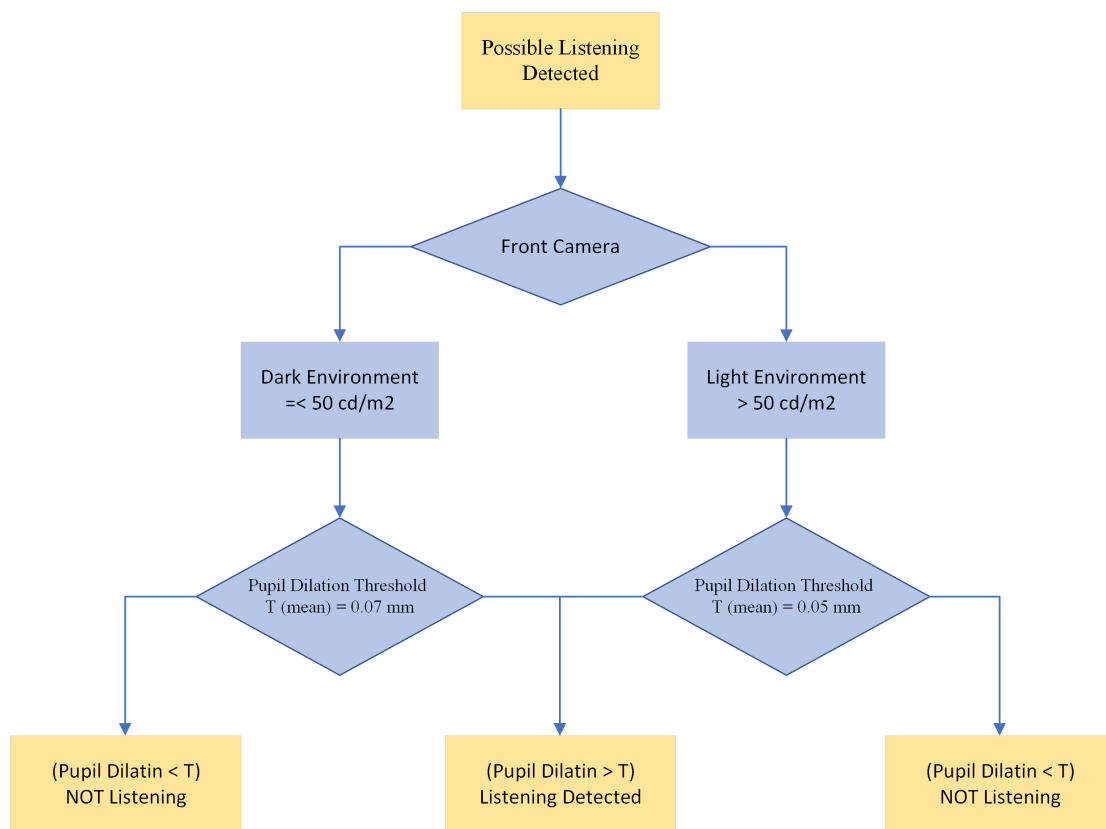


Figure 9: Decision making for Listening Detection.

6 TRACKING AND LOCALIZATION

The tracking and localization problem concern the ability to construct a map of an unknown environment while keeping track of the position of the person wearing the glasses. In previous years, this has been done using the SLAM-method EKF-SLAM, yielding the linearized model shown in equation (48)

$$x_{k+1} = F_k x_k + G_k w_k, \quad Cov(w_k) = Q_k \quad (48a)$$

$$m_{k+1} = m_k \quad (48b)$$

$$y_k = C_k^x x_k + C_k^m (c_k^{1:I_k}) m_k + e_k, \quad Cov(e_k) = R_k \quad (48c)$$

In these equations, the state-space model in (48a) is the same as in equation (7) in Section 2.1. The measurements y_k are dependent on landmark association, which will change when landmarks are introduced or removed. In order to propagate the EKF in time, the statistics in (49) are used.

$$\hat{z}_{k|k-1} = \begin{pmatrix} \hat{x}_{k|k-1} \\ \hat{m}_{k|k-1} \end{pmatrix}, P_{k|k-1} = \begin{pmatrix} P_{k|k-1}^{xx} & P_{k|k-1}^{xm} \\ P_{k|k-1}^{mx} & P_{k|k-1}^{mm} \end{pmatrix} \quad (49)$$

By utilizing equations (48) and (49), the update and time update can be obtained. The measurement update is

$$S_k = C_k^x P_{k|k-1}^{xx} C_k^{xT} + C_k^m P_{k|k-1}^{mm} C_k^{mT} + C_k^m P_{k|k-1}^{mx} C_k^{xT} + C_k^x P_{k|k-1}^{xm} C_k^{mT} + R_k \quad (50a)$$

$$K_k^x = \left(C_k^x P_{k|k-1}^{xx} + C_k^m P_{k|k-1}^{mx} \right)^T S_k^{-1} \quad (50b)$$

$$K_k^m = \left(C_k^x P_{k|k-1}^{xm} + C_k^m P_{k|k-1}^{mm} \right)^T S_k^{-1} \quad (50c)$$

$$\varepsilon_k = y_k - C_k^x \hat{x}_{k|k-1} - C_k^m \hat{m}_{k|k-1} \quad (50d)$$

$$\hat{z}_{k|k} = \hat{z}_{k|k-1} + \begin{pmatrix} K_k^x \\ K_k^m \end{pmatrix} \varepsilon_k \quad (50e)$$

$$P_{k|k} = P_{k|k-1} - \begin{pmatrix} K_k^x \\ K_k^m \end{pmatrix} S_k \begin{pmatrix} K_k^x \\ K_k^m \end{pmatrix}^T \quad (50f)$$

and the time update is

$$\hat{z}_{k+1|k} = \begin{pmatrix} F_k & 0 \\ 0 & I \end{pmatrix} \hat{z}_{k|k} \quad (51a)$$

$$P_{k+1|k} = \begin{pmatrix} F_k P_{k|k}^{xx} F_k^T + G_k Q_k G_k^T & F_k P_{k|k}^{xm} \\ P_{k|k}^{mx} F_k^T & P_{k|k}^{mm} \end{pmatrix} \quad (51b)$$

6.1 Real-time optimization

The current modules are running without parallel processing. This means that the face detection, face mesh, pose estimation and localization algorithms are running in series. This result in a considerable large time lag for when

data is processed making the real time performance of the glasses obsolete. By making use of multi-threading, the computational time can be lowered significantly if processes are parallelized appropriately. The task mainly involves how to split the modules on the threads without concurrency problems. Sampling rate of the EKF might be lowered to lower the computational effort. Therefore, optimizations to the current implementation will be done.

6.2 Distance perception

Previous years implementation relied on the face mesh to produce 3D coordinates on the face and the 2D images to produce the distance to the camera user. This means that estimates of distance to a speaker is only produced when the speaker is facing the camera and have the face visible. In situation where face mesh is not possible, distance estimate is not possible either.

The solution for this is deep neural networks for monocular vision based on a paper from Virgili University [8]. Monocular depth estimation problem is ill-posed since it's impossible to recover depth from geometry from a mono camera. Other cues has to be used such as light reflection, object size, texture, shade etc to estimate the depth. A neural network can be trained using supervised learning to learn these features by using ground truth data from the KITTI dataset. This results in the monoDepth network which is able to estimate relative position. In order to go from relative to absolute distance, the paper [8] describes parameter fitting of relative distances to reflect reality. monoDepth is used in parallel with YOLOv5 for object identification and association of a distance to the object. The expected accuracy is 0.203 RMSE of correct absolute distance.

6.3 New statistical tracking method

The system currently utilizes simple target tracking methods based on EKF to keep track of speakers and objects in the environment. These modules use modules for estimation of position and orientation together with face detection and face meshes. These methods will be expanded upon by introducing statistical methods for multi-target tracking. Three examples of these methods are presented below.

The first method is Multi-Hypothesis Tracking (MHT). MHT can conceptually be explained as generating all possible association hypotheses, computing the probability of the available options, and then reducing the number of possible hypotheses by merging similar ones and removing those with a low probability. However, generating all possible hypotheses requires a lot of computational power and the method is therefore rather inefficient. To rectify this, two different principal implementations of the method exist. The first one is Hypothesis-Oriented MHT (HO-MHT) which only generates the best hypotheses and ignore those who will be deleted. The second is Track Oriented MHT (TO-MHT) where the MHT is built from the possible tracks, which is more efficient as several hypotheses usually contain identical tracks. [9]

The second method is a Global Nearest Neighbor (GNN) tracker. In the GNN method, the most likely association hypothesis is chosen. The chosen hypothesis is thereafter used to update the track logic and all tracks with associated measurements. This method makes a hard association decision, meaning that it is optimal when a correct decision is made. However, the opposite is also true as a wrong association could make the entire method break down. The method is relatively easy to implement and requires a small amount of computational power. It does however only work well when targets are well separated and will cause problems when the separation is poor. [10]

The final method is a Joint Probabilistic Data Association (JPDA) Filter. In a JPDA filter, all hypotheses are merged

into a single hypothesis weighted with their probability of being true. The JPDA method works well when targets are well separated, but neighboring tracks may become identical when targets are close together. Unlike GNN, this filter makes a soft decision. This makes the method more robust, but it also means that it is sub-optimal when correct associations are made. The method is more computationally complex than GNN, but it requires less computational power [10]

In order to decide on which of these methods to implement, more research into their pros and cons will be done. A decision will then be made based on which method seems to be most suited for the complete system. If time allows it, more than one method may be implemented. In that case, a comparison between their performance will be done.

REFERENCES

- [1] F. Gustafsson, *Statistical Sensor Fusion*, 3rd ed. Studentlitteratur AB, 2018.
- [2] N. C. F. E. Information. (2022) World magnetic model. [Online]. Available: <https://www.ngdc.noaa.gov/geomag/calculators/magcalc.shtml#igrfwmm>
- [3] Tobii AB. (2021) How do tobii eye trackers work? [Online]. Available: <https://www.tobiipro.com/learn-and-support/learn/eye-tracking-essentials/how-do-tobii-eye-trackers-work/>
- [4] American Academy of Ophthalmology. (2022) Saccade. [Online]. Available: <https://eyewiki.aao.org/Saccade>
- [5] D. A. Robinson, “The mechanics of human smooth pursuit eye movement,” *J Physiol*, 1965.
- [6] A. A. Zekveld, T. Koelewijn, and S. E. Kramer, “The pupil dilation response to auditory stimuli: Current state of knowledge,” *Trends in Hearing*, 2018.
- [7] P. Książek, A. Zekveld, D. Wendt, L. Fiedler, T. Lunner, and S. Kramer, “Effect of speech-to-noise ratio and luminance on a range of current and potential pupil response measures to assess listening effort,” *Trends in Hearing*, 2021.
- [8] A. Masoumian, D. G. Marei, S. Abdulwahab, J. Cristiano, D. Puig, and H. A. Rashwan, “Absolute distance prediction based on deep learning object detection and monocular depth estimation models,” *Frontiers in Artificial Intelligence and Applications*, 2021.
- [9] G. Hendeby and R. Karlsson, “Lecture notes on multi target tracking: multi-hypothesis tracking,” April 2019. [Online]. Available: <https://www.control.isy.liu.se/student/graduate/targettracking/file/le5.pdf>
- [10] —, “Lecture notes on multi target tracking: single hypothesis tracking,” February 2019. [Online]. Available: <https://www.control.isy.liu.se/student/graduate/targettracking/file/le4.pdf>