



POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Improvements of a Cognitive Model of the Human Driver

TESI DI LAUREA MAGISTRALE IN
AUTOMATION AND CONTROL ENGINEERING - INGEGNERIA
DELL'AUTOMAZIONE

Author: **Ahmed El Hanafi**

Student ID: 995597
Advisor: Prof. Alessandro Colombo
Academic Year: 2022-23

Abstract

Many developments have been done in the autonomous driving field recently, yet it is still a technology that is not mature enough to be used in everyday real world scenarios, making human drivers still in charge of such task. This creates a safety problem because humans are prone to errors, something that could lead to accidents while driving. To counteract this problem safety systems have been invested on and developed to assist the drivers, ADAS (Advanced Driver Assistance Systems) systems in particular have helped in the reduction of accidents, still it is hard for them to determine whether intervention is necessary or not, due to the difficulty in understanding the driver's intentions and their understanding of the surrounding environment.

This problem has been tried to be solved by developing algorithms capable of understanding if the driver is attentive or not, still this does not solve the problem since a driver could be attentive but still make mistakes. This is the problem that this thesis tries to solve, by the introduction of a mathematical model capable of simulating the Perceptive and Cognitive processes of a human observing a walking pedestrian.

The Perceptive process is responsible for the visual measurements done while looking at a target, and it simulates the error made in doing so, while the Cognitive process is responsible for the understanding of the targets' movement and how they are predicted.

In this thesis a Cognitive model for the pedestrian's movement is introduced, where the model will be developed as a Bayesian filter, in particular starting from the structure of an Extended Kalman Filter it will be attempted to develop a nonoptimal filter, to better simulate the human's cognitive process. The model's parameters will be then identified with a dataset developed for this purpose. Finally, an experiment campaign will be conducted in a virtual environment using a Virtual Reality headset, so to collect information about the Cognitive process when observing a cyclist.

Keywords: Cognitive model, Perceptive model, Bayesian filter, Observer ,EKF, ADAS

Abstract in lingua italiana

Recentemente molti sviluppi sono stati fatti nel campo della guida autonoma, tuttavia è una tecnologia non abbastanza matura per essere utilizzata in molti scenari quotidiani, rendendo il guidatore umano ancora responsabile di tale compito. Questo crea un problema di sicurezza, poiché gli esseri umani sono inclini a sbagliare, qualcosa che mentre si guida potrebbe causare incidenti. Per contrastare questo problema, sono state investite molte risorse e sviluppati sistemi di sicurezza per assistere i guidatori, in particolare i sistemi ADAS (Advanced Driver Assistance Systems) hanno contribuito a ridurre gli incidenti. Tuttavia fanno fatica a determinare se necessario intervenire, a causa della difficoltà nel comprendere le intenzioni del guidatore e la su comprensione dell'ambiente circostante. Si è cercato di risolvere questo problema sviluppando algoritmi in grado di comprendere se il guidatore è attento o meno, ma ciò non risolve il problema dal momento che un guidatore potrebbe essere attento ma comunque commettere errori. Questo è il problema che questa tesi cerca di risolvere, introducendo un modello matematico in grado di simulare il processo Percettivo e Cognitivo di un umano che osserva un pedone che cammina. Il processo Percettivo è responsabile delle misurazioni visive fatte mentre si guarda un corpo esterno, e simula l'errore commesso nel farlo, mentre il processo Cognitivo è responsabile della comprensione del movimento del corpo esterno e di come lo si predice. In questa tesi si introduce un modello cognitivo per il movimento del pedone, sviluppato come un filtro Bayesiano, più nello specifico partendo dalla struttura di un Filtro di Kalman Esteso si cercherà di sviluppare un filtro non ottimale, in modo da simulare meglio il processo cognitivo umano. I parametri del modello saranno quindi identificati con un dataset sviluppato per questo scopo. Infine, una campagna sperimentale sarà condotta in un ambiente virtuale utilizzando un visore di Realtà Virtuale, in modo da raccogliere informazioni sul processo cognitivo quando si osserva un ciclista.

Parole chiave: Processo Cognitivo, Processo Pecettivo, Filtro Bayesiano, Osservatore, EKF, ADAS

Contents

Abstract	i
Abstract in lingua italiana	iii
Contents	v
1 Introduction	1
1.1 Problem Introduction	1
1.2 State of the Art on ADAS	4
1.3 The Problem of Understanding the Driver	6
1.4 Statement of contributions	8
1.5 Thesis Structure	8
2 Introduction to the Model and Dataset	11
2.1 Introduction	11
2.2 Perceptive Model	11
2.3 Cognitive Model	15
2.4 Cognitive Experiment	17
3 Kinematic Model	27
3.1 Introduction	27
3.2 Starting Models	27
3.3 Development of the Unicycle Model	28
3.3.1 Linear Velocity Model	28
3.3.2 Polynomial Velocity Models	28
3.3.3 Distance Model	29
4 Cognitive Model	31
4.1 Introduction	31
4.1.1 EKF	31

4.1.2	Modified Extended Kalman Filter	33
4.1.3	Luenberger Observer	33
5	Results	37
5.1	Introduction	37
5.2	Optimization Procedure	37
5.3	Results	40
5.3.1	Comparison of the Kinematic Models	40
5.3.2	Comparison of the Filters	47
5.3.3	Comparison with dataset	55
6	Experiment Design	59
6.1	Introduction	59
6.2	Equipment and Software	59
6.3	Experiment Design	59
6.4	Experiment Conduction	66
6.5	Comparison with the Previous Experiment	69
6.6	Experiment Observations	69
6.6.1	Discarded Data	70
6.6.2	Pointing difficulties	71
6.6.3	Normality Tests	72
7	Conclusions	75
 Bibliography		77
List of Figures		81
List of Tables		83
Ringraziamenti		85

1 | Introduction

1.1. Problem Introduction

In recent years many developments have been done in the *Autonomous Vehicles* (AV) and *Machine Learning* (ML) fields. According to [25] the advent of such technology promises many advantages over conventional vehicles, such as fewer accidents, less traffic and fewer emissions. Still, safety is a concern due to how ML models work and are trained, indeed as stated in [33], pattern recognition, that is a key factor in ML applied in AV, suffers from the so called long-tail problem, this means that given an object, this could be recognized with some specific parameters (e.g. lighting, weather), but this can not cover every case, due to a huge number of edge cases. This makes AVs a possible liability, since a single edge case could lead to an accident.

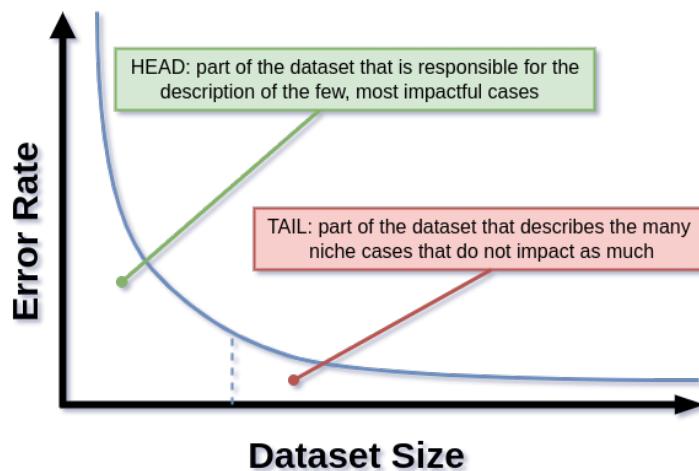


Figure 1.1: Qualitative graph describing the long-tail problem, where it is shown how given a dataset to train a model, the most common cases will be taken care from a small batch, and consequently this will also drastically reduce the error of the trained model. The many uncommon, or niche, cases though require a much larger dataset since they are hard to spot, moreover there is a diminishing return in the error reduction. This problem is not exclusive to Machine Learning and is known with other names such as Pareto Principle, or twenty-eighty rule.

A solution to reduce this problem could be to build cities in such way to reduce the possible plethora of conditions AVs could face, but this contrasts the current vision on how urbanization should evolve, as stated in [22]. New urbanization plans focus on narrower streets, more vegetation and more in general the implementation of solutions catered toward walkable cities (see [15] and [32]), making wide and easy to read streets possible only in car centric areas such as highways (see [23]).



Figure 1.2: Image taken in the city of Utrecht, Netherlands. The Netherlands is a country renowned for its bike friendly and walkable cities, making it a prime example of how urbanization is evolving. Looking at this image it is visible how the road is designed to be small, irregular and visibly residential, thanks to the use of brick instead of asphalt. This has the double effect of slowing down drivers, while making pedestrian feel, and be, safer on the street. Image taken from [34]

It is then clear how cars will still be driven by humans, making the development of safety systems fundamental. According to the *National Highway Traffic Safety Administration* among many causes of accidents between light vehicles 94% of the cases are attributed to drivers' errors. From that 94%, 41% is due to recognition errors, 33% to decision errors and 11% to performance errors (see [29]), this makes the introduction of safety systems, catered toward the assistance of drivers, fundamental and indeed many investments and research has been and is done on this matter.

Safety systems are categorized in this way: Driver Assistant Systems (DAS), that rely on egocentric data, and Advanced Driver Assist Systems (ADAS), that instead focus on

the surrounding environment. ADAS' introduction in particular has been crucial in the reduction of accidents, as stated in [21].

1.2. State of the Art on ADAS

ADAS is a collection of passive and active safety systems catered toward the assistance of the driver, their aim is not to substitute the driver, rather help him when it is necessary. To do so, a plethora of sensors are used to gather information ,better than a human could. For example, as stated in [30], ADAS systems are equipped with: RADARS to sense better the surroundings that a driver could not see, SONARS to detect objects where there is no visibility, cameras and LiDARs to see in all directions at each instant. Moreover, satellites are used to collect information about the position and speed of the car.

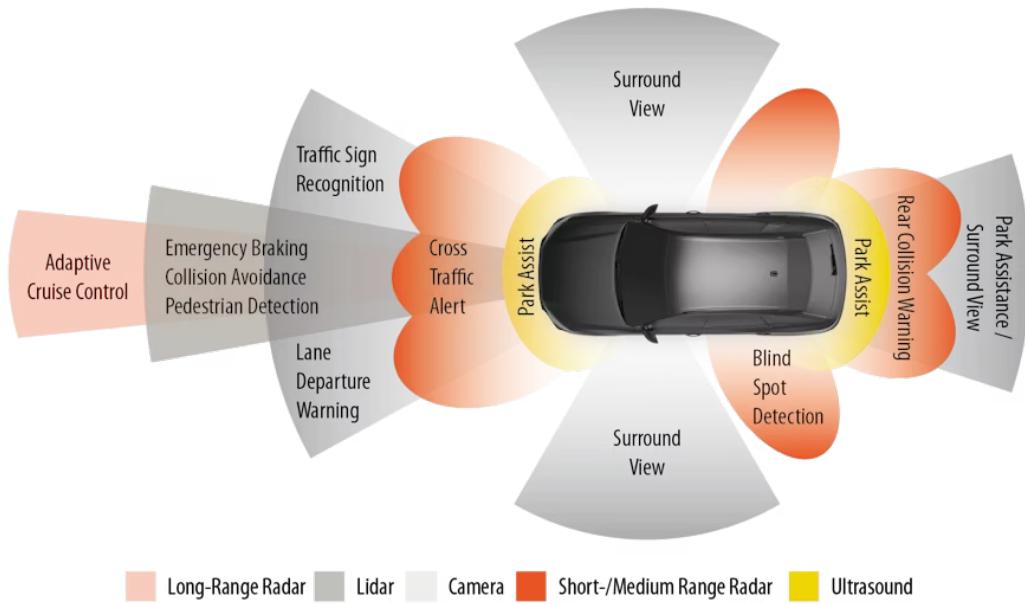


Figure 1.3: Example of sensors equipped on a car and their use. Image taken from [30].

The information gathered by the sensors is then processed by a computer, that has to make decisions in real time regarding the necessity of intervention based on the level of danger detected. ADAS systems are divided in two categories: passive and active safety systems:

- The passive safety systems are responsible for the notification of the driver of potential hazardous situations, without actually operating the vehicle. Examples of

passive ADAS are Forward Collision Warning (FCW), Lane Departure Warning (LDW), Blind Spot Monitoring (BSM) and Driver Attention Warning (DAW). There are two problems regarding the passive system though, as stated in [7], one being the promptness of the driver to react to the warnings, and the possible false alarms caused by software bugs or environment factors, that could lead to the driver ignoring the system.

- The active safety systems are responsible for the actual intervention on the vehicle, on top of notifying the driver, this is done to both prevent accidents and increase driving comfort. Examples of active ADAS are Adaptive Cruise Control (ACC), Lane Keeping Assist (LKAS), Automatic Emergency Braking (AEB), and Traffic Jam Assist (TJA). ADAS have some flaw though, as stated in [6], such as the high dependency on sensors and their well functioning, moreover even when intervening the driver is still supposed to be ready to take control of the vehicle, indeed these systems can only assist the driver, not substitute him/her.

It is then clear how both active and passive ADAS require an attentive driver. Regarding this problem there are many examples of research in literature (see [19] and [2] for a comprehensive overview). An example can be seen in [1], where an algorithm, called Driver Inattention Monitoring System, has been developed that determines the attentiveness of a driver based on his/her gaze, if the level of attention goes below a certain level the driver is then considered distracted. Many of the examples seen in literature follow this approach, where a driver is considered either attentive or not, this is problematic though since it does not consider the case when the driver is actually paying attention and is not drowsy, yet not capable of understanding well enough the surrounding situation.

Another problem regarding ADAS systems is the difficulty in understanding when a situation is actually dangerous and when instead the driver is in control while driving aggressively. An attempt to solve this problem is done in [37], where an adaptive algorithm is developed to understand the driver's intention and control on the car. On a more general note, the problem of understanding the driver's capability of analyzing the surrounding could also help in this case, so to make ADAS intervene only when the driver is clueless about a possible hazardous situation.

1.3. The Problem of Understanding the Driver

The problem of analyzing the driver's understanding of the surrounding is, as much as the writer is aware, not well studied in literature. Such problem is tackled in this thesis, continuing the work that has been done in [17] and [18].

The work so far has been focused on the creation of a mathematical model describing the process of a human observing a target, where the target is either a pedestrian or a cyclist, since both are vulnerable road users. To do so two models have been introduced:

- The Perceptive model: this describes how an external object is measured by a human when seen, more specifically it computes the error made in doing so, called perceptive error.
- The Cognitive Model: this describes how a pedestrian understands the movement of an external target, and how it is predicted.

Regarding the Perceptive model, so far it has been developed based on optic physics, and it is capable of estimating the perceptive error starting from the position of the target with respect to the observer, more in particular the distance, and the eccentricity with respect to the visual axis of the observer are what determines the computed error.

This model has then been used to determine the perceptive error when observing a cyclist, where since the heading angle was an important parameter to compute as well some geometrical considerations were made that brought to the final model.

Regarding the Cognitive model instead a basic structure based on the *Extended Kalman Filter* (EKF) has been introduced. This allows to divide the problem in two parts, modelled through the prediction and update parts that compose an EKF.

The prediction part of the algorithm is catered toward the modelling of how a person understands the motion of the target, and such knowledge is modeled through a nonlinear set of difference equations, the kinematic model. It is important to notice the difference between the understanding of a person and the actual motion of a target, indeed the kinematic model is not supposed to be accurate to how the target moves, rather to how a person thinks it does.

The update part of the algorithm instead is the part that corrects the estimations done in the prediction phase, and to do so it uses the measurements of the target done through the perceptive model.

So far the Cognitive model has been developed just for the bicycle case.

This idea of modelling the Cognitive process as an EKF is not completely new indeed it has already been seen in [9], but a model that describes it when observing a target has not been found in literature. What is easily found instead, if the target considered is a pedestrian, is the prediction of the actual trajectory, and this has been approached with different means, mainly through an optimal approach ([11], [36], [24], [10]), and statistical models ([13], [14]), where the prediction is usually long term, so more than three seconds.

The closest work found in literature is [27], where the motion of a pedestrian is predicted using an EKF and an *Interacting Multiple Model* (MM).

There have also been developed and conducted two experiments meant to gather information about the cognitive and perceptive process. These experiments have been conducted in a virtual environment, using a Virtual Reality headset (VR headset). This allowed to freely design the scenario where the experiments were conducted, and to gather data in a controlled environment. The data gathered has been used to estimate the parameters of the developed Perceptive and Cognitive models.

1.4. Statement of contributions

The work done in this thesis is the continuation of what has been done in [17] and [18] by former students Ludovico Rozza and Matteo Depaola. What will be introduced here is the result of my work and my advisor's, Prof. Alessandro Colombo and Ph.D. student Nicolò Dozio.

More specifically the contributions of this thesis are:

- The readaptation of the perceptive model for the pedestrians' case
- The introduction of different kinematic models for the cognitive model for the pedestrian
- The development of a Cognitive model that is based on a generic observer, and not only on the EKF in particular, so to remove the hypothesis of optimality that the EKF has.
- The development and conduction of a new experiment to gather data for the Cognitive model for the bicycle case

1.5. Thesis Structure

The thesis is structured as follows:

- **Chapter 2:** a comprehensive introduction to the model developed will be given, then the experiment that resulted in the dataset used to estimate the parameters of the models will be introduced, finally the dataset will be analyzed.
- **Chapter 3:** the different kinematic models used for the Cognitive model will be introduced.
- **Chapter 4:** the different Cognitive models structures will be introduced.

- **Chapter 5:** the different kinematic and Cognitive models will be tested on the dataset and the results will be compared.
- **Chapter 6:** the experiment conducted to gather data for the Cognitive model for the bicycle case will be introduced.

2 | Introduction to the Model and Dataset

2.1. Introduction

In this chapter a comprehensive introduction to the Perceptive and Cognitive models for the pedestrian will be given, then the experiments conducted in [17] for the Cognitive model will be introduced.

2.2. Perceptive Model

The Perceptive model is a fundamental part of the overall mathematical structure that is being developed, indeed it is responsible for the computation of the perceptive error that a person makes when observing a target. A model formulated for the observation of a point mass object will be used, where just the position on the ground will be considered, then the heading will be added later on.

Before introducing the mathematical model some hypotheses must be done:

- The position of a body is measured as its projection of the geometric center on the ground.
- The perception process of an object involves projecting it into oneself's retina, leading to an error that is caused due to the difference between the projection with respect to the position of the fovea.
- The plane onto which the points will be projected is assumed to be orthogonal to the pupillary axis, if the head can only rotate around its yaw angle.
- The *Useful Field of View* (UFOV), the part of the field of view where a human can perceive external objects, is going to be the only one where a person can see, outside it nothing is visible. The UFOV it is going to be modelled as a cone with a range of $\pm 30^\circ$.

Before introducing the mathematical model, three reference systems must be introduced:

- World Reference Frame (WRF): this parametrizes the position of the observer with respect to the other objects in space. The position in WRF is defined as $\mathbf{x} \in \mathbb{R}^3$, where (x_1, x_3) parametrize the horizontal plane where the observer is standing, and x_2 parametrizes the vertical direction. For simplicity, it will only be considered the case that the observer is standing at the origin.
- Head Reference Frame (HRF): this is placed at the height of the pedestrian's head, more specifically its origin is placed at the intersection of the sagittal, transversal and frontal plane of the head, and it follows the head's movement. The position in this frame is defined as $\mathbf{z} \in \mathbb{R}^3$, where z_1 is contained between the frontal and transversal planes, z_2 between the frontal and sagittal planes and z_3 between the sagittal and transversal planes. Defining the head rotation around x_2 as θ , and the position of the head in WRF as \mathbf{x}_{head} , to change coordinates from WRF to HRF the following rotation matrix and translation vector are used:

$$\mathbf{T}(\theta) = \begin{bmatrix} \cos(\theta) & 0 & \sin(\theta) \\ 0 & -1 & 0 \\ -\sin(\theta) & 0 & \cos(\theta) \end{bmatrix} \quad (2.1)$$

$$\mathbf{x}_{world} = \begin{bmatrix} 0 \\ x_s \\ 0 \end{bmatrix} \quad (2.2)$$

$$\mathbf{z} = T(\theta)(\mathbf{x} - \mathbf{x}_{head}) \quad (2.3)$$

where x_s is the head's height from the ground.

- Retina Reference Frame (RRF): this represents the observed object's position on the retina, the position is defines as $\mathbf{y} \in \mathbb{R}^2$, where (y_1, y_2) is parallel to the (z_1, z_2) plane. The origin of the frame is placed in the intersection between the (y_1, y_2) plane and the visual axis when it is parallel to the ground. To move from the HRF to the RRF, the following transformation is used:

$$\mathbf{y} = s(z_1, z_3) = \begin{bmatrix} \frac{z_1}{z_3} \\ \frac{z_2}{z_3} \end{bmatrix}. \quad (2.4)$$

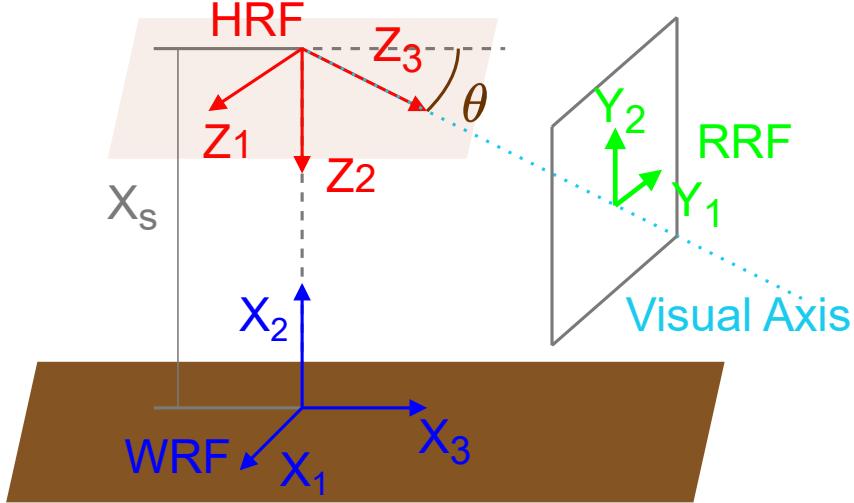


Figure 2.1: The three reference frames used in the Perceptive Model.

Now a mathematical model that describes the Perceptive Process can be introduced. As already stated, the perception error is defined as the difference between the perceived position and the real one, where in the Perceptive model it is described as a Gaussian distribution. More specifically in the HRF it is defined as $\nu_y \sim \mathcal{N}(\mu_y, \Sigma_y)$, where both μ_y and Σ_y depend on $y = [y_1, y_2]$ and $y_m = [0, y_{f,2}]$, the position of the fovea, more specifically they are formulated as:

$$\mu_y = \begin{bmatrix} -k_1 y_1 y_2 \\ k_2 y_1^2 (y_2 - y_{f,2} - k_3) - (y_2 - y_{f,2}) (1 - e^{(-k_4(y_2 - y_{f,2})^2)}) \end{bmatrix} \quad (2.5)$$

$$\Sigma_y = \begin{bmatrix} ((1 + c_1 y_1^2) n_1)^2 & 0 \\ 0 & ((1 + c_2 (y_2 - y_{f,2})^2) n_2)^2 \end{bmatrix} \quad (2.6)$$

c_1	c_2	n_1	n_2	k_1	k_2	k_3	k_4
4.196	28.391	0.027	0.013	0.332	0.608	0.055	2.46

Table 2.1: Parameters identified for the perceptive model. The values have been computed in [18], ch.7

Since these matrices are computed in RRF, they must be transformed in WRF, so to be compatible with the Cognitive Model, which instead is modelled only in WRF. An

intermediate step to change coordinates to HRF will be made:

$$\mathbf{J}_s(z_1, z_3) = \begin{bmatrix} \frac{1}{z_3} & -\frac{z_1}{z_3^2} \\ 0 & -\frac{z_2}{z_3^2} \end{bmatrix} \quad (2.7)$$

$$(\mu_{z_1}, \mu_{z_3}) = \mathbf{J}_s(z_1, z_3)^{-1} \boldsymbol{\mu}_y \quad (2.8)$$

$$\begin{bmatrix} \sigma_{z_1}^2 & \sigma_{z_1 z_3} \\ \sigma_{z_1 z_3} & \sigma_{z_3}^2 \end{bmatrix} = \mathbf{J}_s(z_1, z_3)^{-1} \boldsymbol{\Sigma}_y \mathbf{J}_s(z_1, z_3) \quad (2.9)$$

Where (2.7) is obtained from the Taylor expansion of (2.4):

$$\mathbf{y} + \boldsymbol{\nu}_y = s(\mathbf{z} + \boldsymbol{\nu}_z) = s(z_1, z_3) + \mathbf{J}_s(z_1, z_3) \boldsymbol{\nu}_z + H.O.T. \quad (2.10)$$

From HRF to WRF the computations are the following:

$$(\mu_{x_1}, \mu_{x_3}) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} \mu_{z_1} \\ \mu_{z_3} \end{bmatrix} \quad (2.11)$$

$$\mathbf{R} = \begin{bmatrix} \sigma_{x_1}^2 & \sigma_{x_1 x_3} \\ \sigma_{x_1 x_3} & \sigma_{x_3}^2 \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} \sigma_{z_1}^2 & \sigma_{z_1 z_3} \\ \sigma_{z_1 z_3} & \sigma_{z_3}^2 \end{bmatrix} \begin{bmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{bmatrix}. \quad (2.12)$$

The obtained result is the perception error in WRF, but it just considered the x_1 and x_3 coordinates, so now the heading angle will be added. In order to do so it will be assumed that there is no bias in the perception of the heading:

$$\mathbf{b} = \begin{bmatrix} \mu_{x_1} \\ \mu_{x_3} \\ 0 \end{bmatrix} \quad (2.13)$$

Regarding the covariance instead three new parameters will be added:

$$\mathbf{R} = \begin{bmatrix} \sigma_{x_1}^2 & \sigma_{x_1 x_3} & r_1 \\ \sigma_{x_1 x_3} & \sigma_{x_3}^2 & r_2 \\ r_1 & r_2 & r_3 \end{bmatrix}. \quad (2.14)$$

The addition of the heading angle is a necessity since the overall model would not work properly otherwise, moreover removing it would mean assuming that the Perceptive process does not measure it, which is not true.

2.3. Cognitive Model

The Cognitive model is responsible for the modeling of how a person understands and predicts the movement of a moving target. The following hypotheses will be used to model it:

- A generic observer is used to model the cognitive process
- The filter uses the subject's prior experience about how the foreign object moves to predict its movement, and this will be done through a mathematical model.
- Since this model is meant to work on digital hardware the model will be developed in discrete time, even though the cognitive process is continuous. Where the model is hypothesized to run with a time step $T_s = 0.01$ s.

The general structure of the model is described in alg. 2.1:

The algorithm, as a Bayesian filter, is divided in Prediction and Update, where the latter occurs only when a new position of the pedestrian is perceived, since a parameter ϵ nullifies the filter's gain otherwise.

The matrices \mathbf{Q} and \mathbf{R} , that in a Bayesian filter are related to the process and measurement noise, here represent the perceptive and cognitive model's uncertainty.

In the prediction phase, given the last states $\xi(t-1)$, the model computes the next steps' states using the kinematic equation $f(\hat{\xi}(t-1))$, this is the mathematical model that describes the understanding of the pedestrian's movement, and as already stated in ch. 1.3, it is not supposed to be accurate to the target's actual kinematics, rather it describes how the observer thinks the target moves, or in other words the perceived kinematics.

During the prediction phase the algorithm evolves the covariance dynamics as well, and this results in its eigenvalues increase, indeed this models the growing uncertainty the more the target is not visible

If the pedestrian is sighted, where the measurement ζ is the perception model's output, $\epsilon = 1$ and the model updates the states' values using the computed gain in the equation $\mathfrak{K}(\cdot)$. At the same time the covariance evolves, where the equation describing this process is known as Joseph form, resulting in a reduction of its eigenvalues.

In the next chapters, 3 and 4, different formulae for the perceived kinematic model $f(\hat{\xi}(t-1))$ and gain's formulation $\mathfrak{K}(\cdot)$ will be presented. It will always be assumed that \mathbf{Q} is diagonal, where the number of elements will be the same as the number of states used in $f(\hat{\xi}(t-1))$.

Algorithm 2.1 Cognitive Process, generic observer structure

2.4. Cognitive Experiment

The *Cognitive Experiment* (CXP) was conducted in [17], and an in depth analysis on the dataset can be found in [8]. In this chapter it will be thoroughly described, and this it will serve for two reasons:

- The dataset is used to optimize the parameters of the model developed in the next chapters.
- A new experiment will be introduced in ch. 6 that will improve some aspects of the design.

First the experiment design will be introduced, then the dataset will be analyzed.

Cognitive Experiment Design

The experiment was conducted in a virtual environment using a Virtual Reality headset, where the setting of the experiment was composed by an intersection with two roads crossing, and sidewalks on the sides. On the roads there were placed road signs, more specifically two cross roads, a right-of-way and centerlines. (see fig. 2.3a).

The candidates where placed as it can be seen in fig. 2.3a, and the height of the candidates point of view was set to 2m, so make it easier to indicate the targets (see fig. 2.3b).

In front of the candidates, one at a time a pedestrian or a bicycle, the possible targets of the experiment, appeared and started a trajectory, each composed by a straight part, lasting $T_{Straight} = 3$ s, then a *Maneuver of Interest* (MOI), this consisted in a curve that could be clockwise or counterclockwise, and it either lasted $T_{est} = 2$ s or $T_{est} = 3$ s, where T_{est} is called *estimation time*. During the MOI the target was visible for a certain amount of time, called *visibility time*, T_v , that could last: 0.25 s, 0.5 s, 0.75 s, 1 s or 1.25 s, then the target disappeared, still continuing the maneuver until the estimation time end. This period of time when the target was not visible is defined as *prediction time*, and it is defined as $T_p = T_{est} - T_v$ (see fig. 2.2).

Each trajectory was preceded by a stereo audio cue, indicating its beginning and helping the candidates finding the spawn position of the target, at the end of each trajectory another audio cue, with a different tone, is played so to indicate the end of the trajectory, this was needed since at this point the target was invisible.

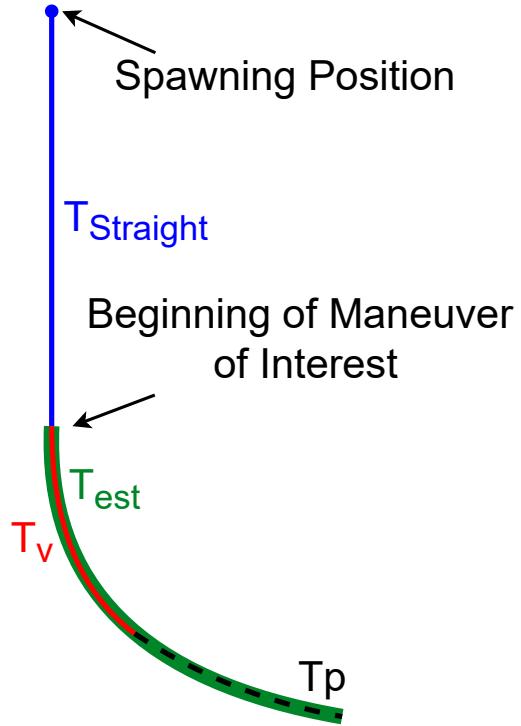
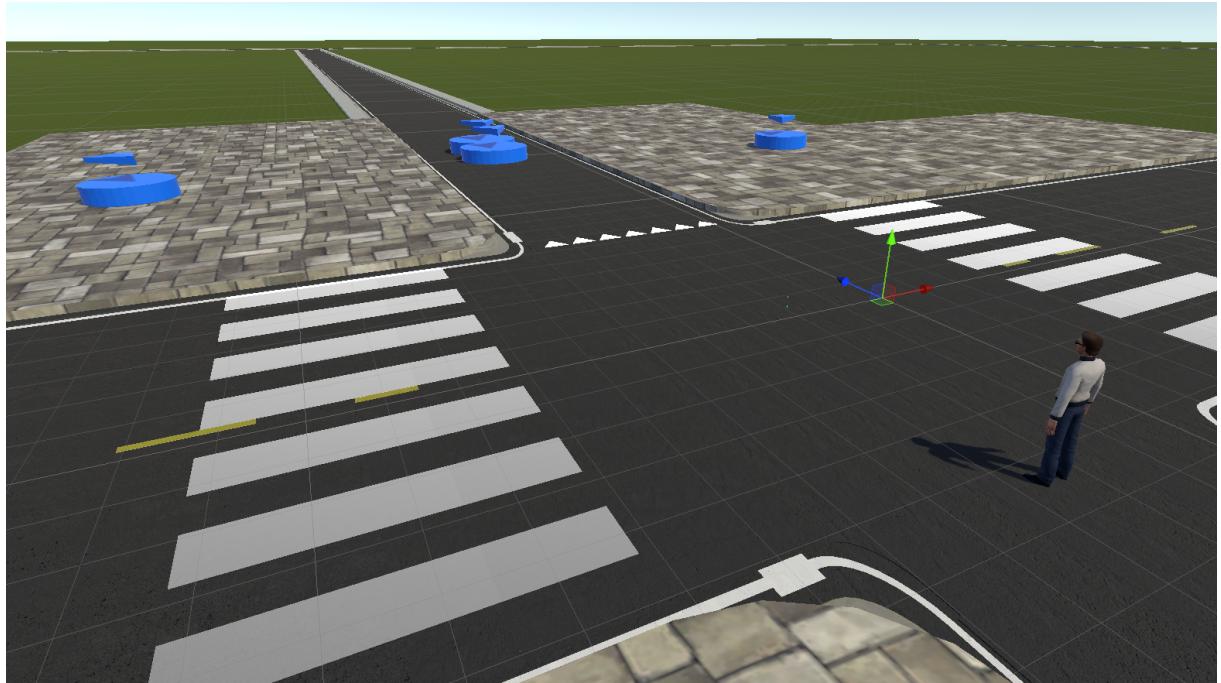
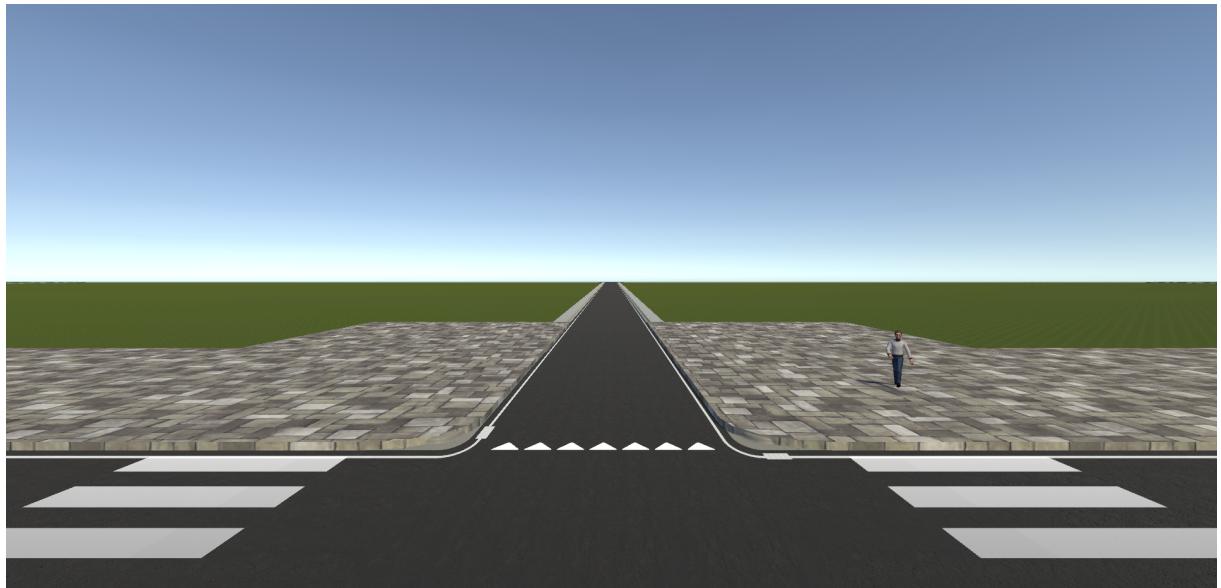


Figure 2.2: Trajectory illustration, where for each part the corresponding duration is indicated with the same color.



(a) CXP scene. The candidates were placed where the man is standing, in coordinates (-2.15 m , 0 m , -5.4 m). The reference system, seen in the image, has as x-axis the red one, as y-axis the green one and as z-axis the blue one.



(b) Point of view of the candidates during the CXP. The height of the point of view of the pedestrian is of 2 m.

Figure 2.3: CXP setting.

The pedestrians and cyclists spawned inside circular areas, where the spawn position was randomized inside them with a uniform distribution, more specifically the pedestrian's spawning areas were positioned on the sidewalks in front of the candidates, and where of radius 1 m, while the cyclists' were positioned on the road, with radius 1.7 m. The spawning areas are indicated in fig. 2.4.

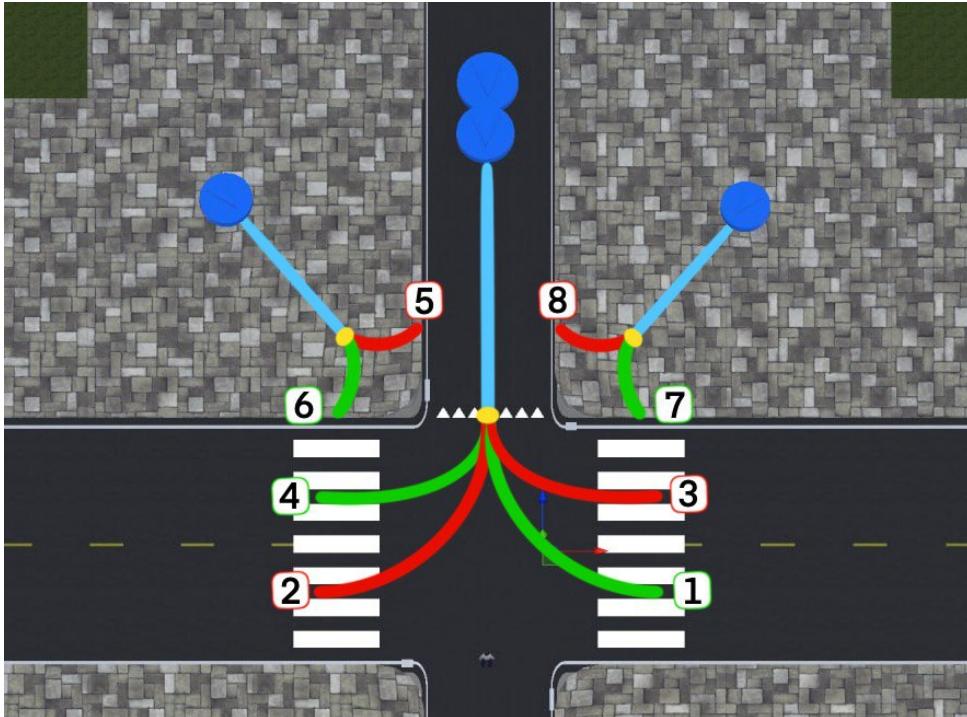


Figure 2.4: Top view of the Perception experiment scene. The pedestrians spawning areas, the ones on the sidewalks, have as center (-14.11 m , 0 m , 16.42 m) and (9.04 m , 0 m , 16.1 m), while the cyclists' one, on the road, have as center (-0.35 m , 0 m , 24.9 m) and (-0.35 m , 0 m , 27.4 m).

The spawning position determined the initial heading angle of the target, more specifically, pedestrians spawning from the left spawning area had an initial heading angle of 225° , while the ones spawning of the right of 315° , all the cyclists instead spawned with a heading of 270° .

The MOI indicated in the figure are: 1. Big-Left 2. Big-Right 3. Small-Left 4. Small-Right 5. Left left 6. Left right 7. Right left 8. Right right.

Both pedestrian and cyclists had 4 possible trajectories, differentiated by the spawning position and MOI. In the pedestrian's case each one moved with a longitudinal velocity of 1.8 m/s, then for each MOI a curved trajectory of radius ± 5.5 m was performed, where the sign of the angular velocity depended on the specific MOI. The possible trajectories for a pedestrian were:

- Left-left
- Left-right
- Right-left
- Right-right

Where the first direction indicates the side from which the pedestrian appeared with respect to the players' point of view, while the second the verse of the MOI's curve with respect to the pedestrian's point of view.

In the cyclists' case instead the MOI where classified in this way in [18]:

- Small/Left
- Big/Left
- Big/Right
- Small/Right

where big stands for a steering angle of ± 0.1297 rad, while small of ± 0.1034 rad, the direction instead is indicated with respect to the cyclist initial heading. The longitudinal speed instead was always 4 m/s (see fig. 2.4).

The task assigned to the candidates was to follow with a laser, that the candidates could point using a controller, the movement of the target until it disappeared, then continue its trajectory indicating where the target was presumed to be. The final positions pointed at by the candidates where collected, for the dataset and a distribution for each MOI, T_v and T_{est} combination was computed, given that there were 8 MOIs, 5 possible visibility times and 2 possible estimation times, 80 distributions where computed, 40 for each target type.

Data Analysis

The collected dataset related to the pedestrians will be now analyzed. Some Normality tests were done so to check if the distributions are Gaussian or not. This is done since, as seen in ch. 2.3, the Cognitive models output a Gaussian distribution, where the mean is the state estimate $\hat{\xi}$ and the covariance is P_0 . This does not mean that it is necessary to have a dateset composed by Gaussian distributions, but it is better to have it, since the final result will be better describing the real process. Obviously, if the data is far from being Gaussian, the model might be not suitable for the task.

Three statistical tests where done: the Lilleforth test [16], the Jarque-Bera test [12] and

the Chi-Square test [31]. These are statistical tests that could be used to check if a dataset is Gaussian or not. They were all performed with a significance level $\alpha = 0.05$ and a null hypotheses stating that the dataset distributions are Gaussian.

Since the data is bivariate, and these tests are deigned for univariate distributions, each distribution was projected along 10 directions(see [28]) and the Bonferroni correction (see [4]) was applied, this means that the significance level was divided by the number of tests, so $\alpha' = 0.005$, and for each distribution the tests where applied 10 times, one for each direction, with the new significance level. A score is then computed as the sum of the failed tests, where the maximum score is 10, when all the tests fail, and the minimum 0, when they all succeed.

The test's results, fig. 2.5, show that many distributions passed the test, more specifically when $T_{est} = 3$ s the best scores where obtained. Still it can be noticed that some distributions, in particular many of the Left right and Right left MOIs with $T_{est} = 2$ s, failed a significant number of tests. This result might be explained by analyzing how the original experiment was designed. Indeed, as stated in ch. 2.4, the setting presented road signs, and in particular there were two crosswalks that, in the case of the Left right and Right left MOIs, where approached by the pedestrians. The problem is that while the MOIs where deigned solely based on the kinematic considerations made in [18], some candidates, might have been influenced by the presence of crosswalks, and imagined a trajectory that made sense if a pedestrian was actually trying to cross the street, and not simply curving with a constant radius of curvature. This effect seems to be more accentuated when $T_{est} = 2$ s, and when T_v is small, so where the candidates had less time to elaborate a trajectory.

It is also to be noticed that the Cognitive model that will be developed in the next chapters, does not contain any information based on the external environment, rather it is based solely on the kinematic model of the perceived motion and the gain of the filter, meaning that ideally the experiment should be conducted in a setting that does not contain any information that could influence the candidates, something that these crosswalks did. It has been not considered the possibility of cleaning the data, mainly to avoid data snooping, but also because when comparing different models of the cognitive model, the use of the cleaned dataset as well for the parameter identification procedure does not introduce any benefits, indeed if comparing two different models and one works better with the raw data, it is expected that it does the same with the clean as well, if that is not the case then the data might be far from normal, and consequently the model is not suitable for the task of describing it. Obviously there is also the presence of outliers, or data that

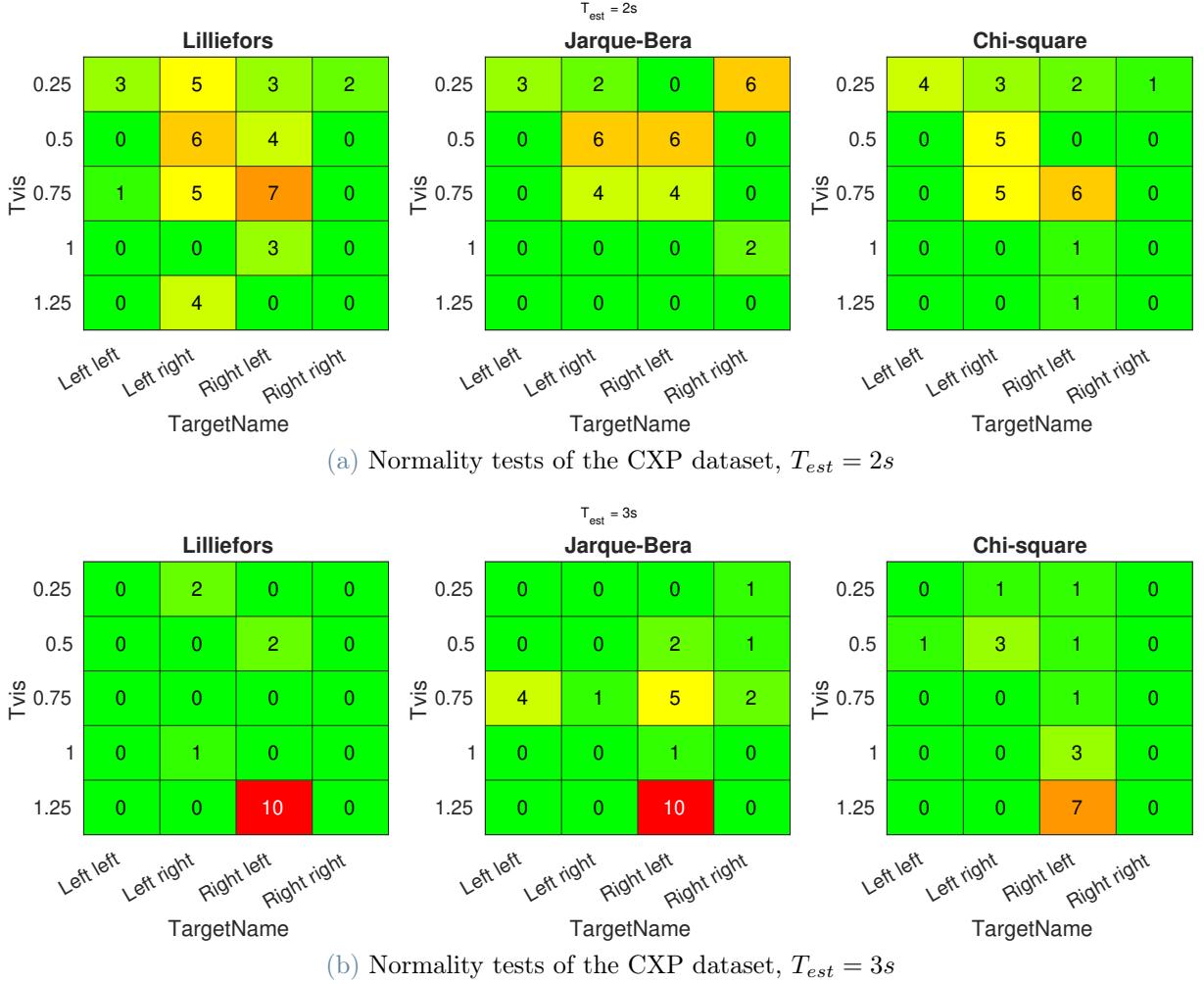


Figure 2.5: CXP dataset normality tests.

was influenced by factors outside the experiment, that might get the distributions to be less Gaussian, but since there is no way to distinguish it, it is better not to manipulate the dataset.

The final results distributions can be seen in fig. 2.7 and fig. 2.8, where the mean and covariance of each distribution has been plotted. It can be noticed how the final distribution's means are all ahead of the real trajectories, implying that the Cognitive process accelerates the pedestrian, moreover the smaller T_v the more it seems like the perceived trajectory is less curved. This could be explained by the fact that the candidates, as soon as the pedestrian disappeared, continued the trajectory along the tangent, and just later make a correction in the curvature. As expected the covariances, when $T_{est} = 3$ s, are bigger, moreover generally speaking the bigger the visibility time the smaller the area of the standard deviation are (as seen in fig. 2.6).

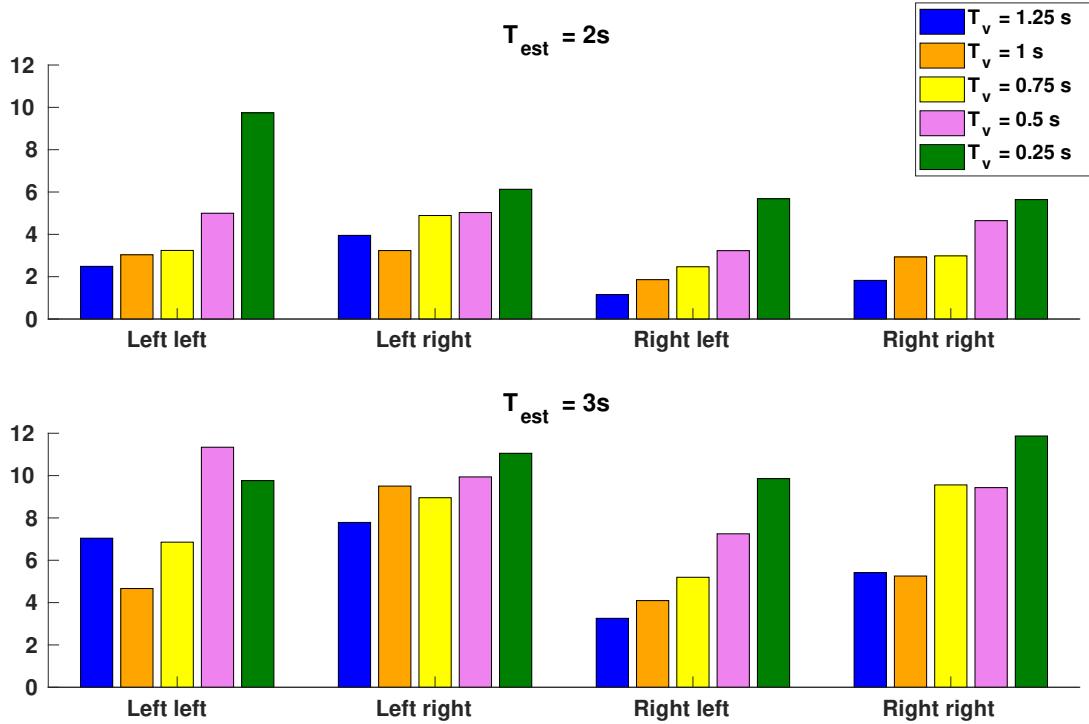
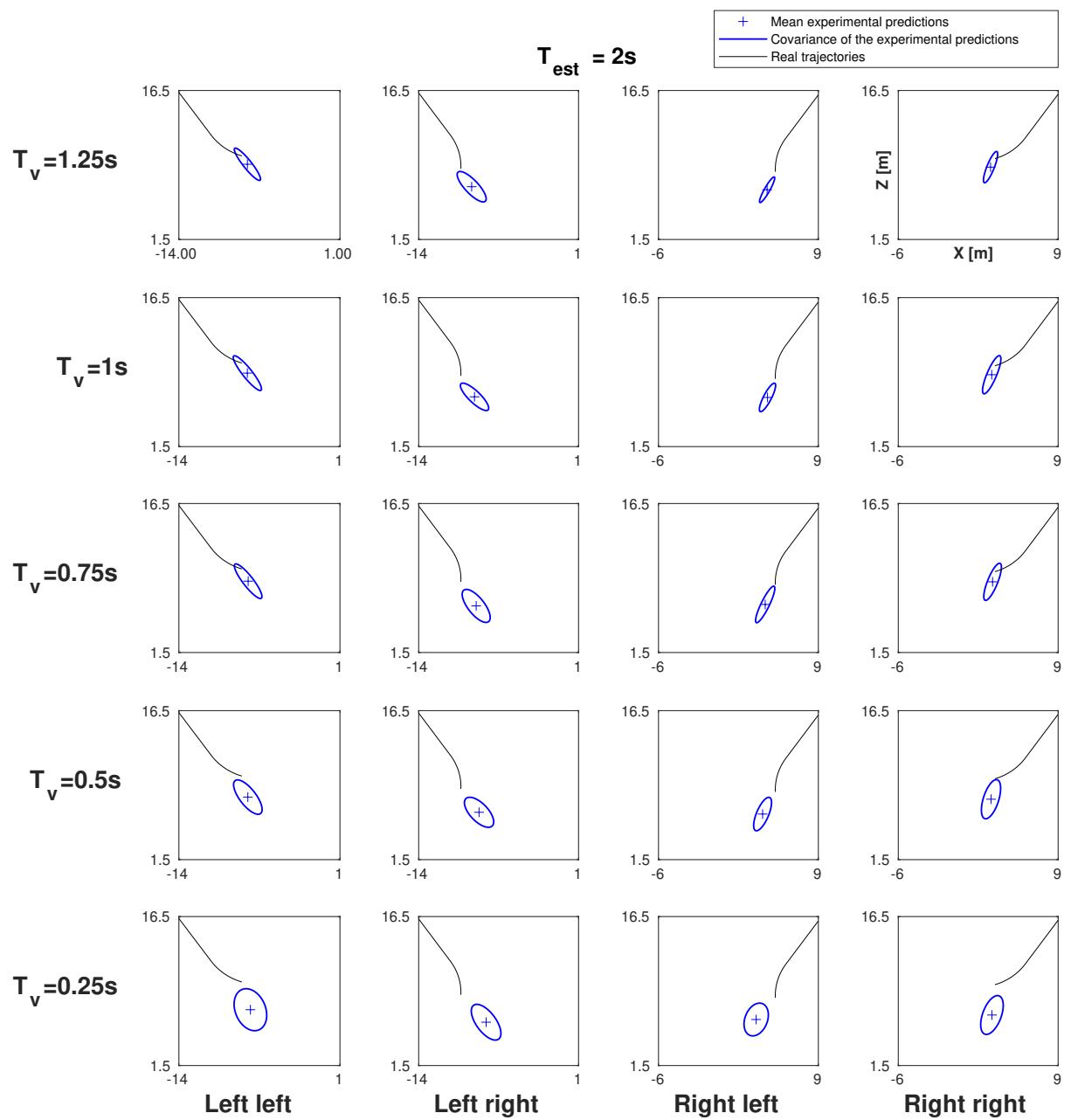
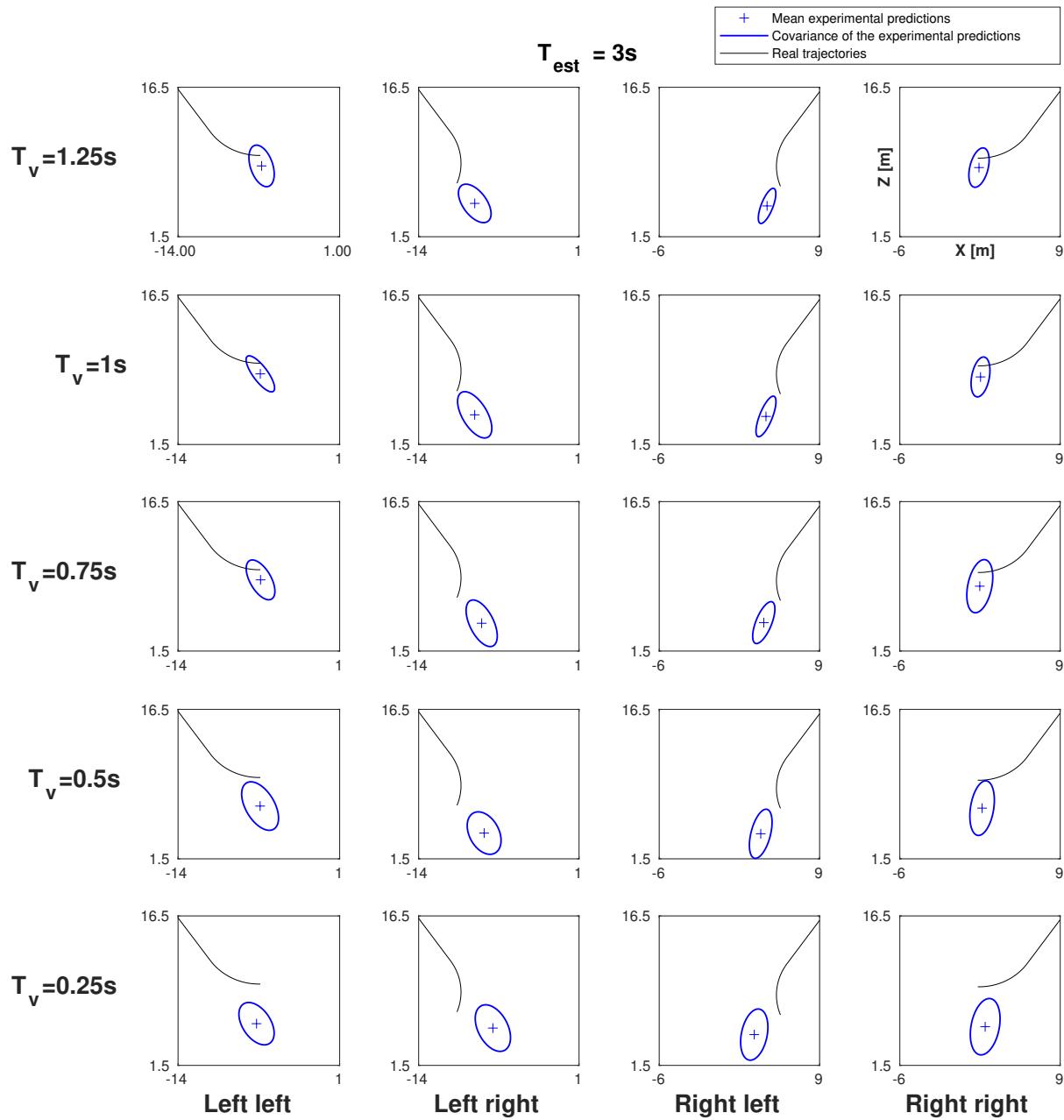


Figure 2.6: Areas of the first standard deviations of the CXP dataset.

This data set will now be used to identify the parameters of the models that will be introduced in the next chapters, where the optimization will focus on obtaining similar final distributions as the dataset's.

It can be noticed how the dataset only contains information about the final guesses of the candidates, and not about the intermediate steps. This is a limitation since, as it will be seen, in ch. 5.3.2, this lack of data will not allow to determine the performance of the models except for the final distributions they output.

Figure 2.7: Dataset distributions for $T_{est} = 2s$.

Figure 2.8: Dataset distributions for $T_{est} = 3\text{s}$.

3 | Kinematic Model

3.1. Introduction

An important part of the Cognitive model is the kinematic equation used. Its job is to describe the motion of the pedestrian in the same way as an observer would imagine it and, mathematically speaking, it determines how the model performs when $\epsilon = 0$ in alg. 2.1.

In this chapter some attempts at finding a suitable kinematic model are presented.

3.2. Starting Models

Having referenced the literature (see [11], [27], [13]) the picked starting point for the kinematic model is the *Unicycle model* (UM):

$$\begin{aligned}\xi_1(t+1) &= \xi_1(t) + \xi_5(t)T_s \cos(\xi_3(t)) \\ \xi_2(t+1) &= \xi_2(t) + \xi_5(t)T_s \sin(\xi_3(t)) \\ \xi_3(t+1) &= \xi_3(t) + \xi_4(t)T_s \\ \xi_4(t+1) &= \xi_4(t) \\ \xi_5(t+1) &= \xi_5(t)\end{aligned}\tag{3.1}$$

where ξ_1 and ξ_2 correspond to the position on the (x_1, x_3) plane in the WRF, ξ_3 is the heading angle, or yaw angle, ξ_4 is the angular velocity and ξ_5 is the longitudinal velocity. This model has the characteristic of not having a heading angle defined by a convoluted equation, and this is accurate since a human does not have any particular kinematic constraints on how he/she can turn.

In this model no acceleration was added, neither to the longitudinal velocity nor to the angular velocity, this was done since a “normal” pedestrian’s motion would likely have no sudden longitudinal and angular velocity changes, rather slow increases or decreases, making this task easily manageable just by the filter correction.

3.3. Development of the Unicycle Model

In this section some attempts at finding a better kinematic model are presented, where the starting point is the UM. More specifically the proposed solutions will try to mimic the tendency of the candidates to overestimate the velocity of the pedestrian. All the states are numbered the same as for (3.1).

3.3.1. Linear Velocity Model

As seen in [18], in order to account for the distance between the dataset and the model's means, a constant coefficient is added to change the velocity over time with a constant rate, where this is supposed to accelerate the pedestrian:

$$\begin{aligned}\xi_1(t+1) &= \xi_1(t) + \xi_5(t)T_s \cos(\xi_3(t)) \\ \xi_2(t+1) &= \xi_2(t) + \xi_5(t)T_s \sin(\xi_3(t)) \\ \xi_3(t+1) &= \xi_3(t) + \xi_4(t)T_s \\ \xi_4(t+1) &= \xi_4(t) \\ \xi_5(t+1) &= d_1\xi_5(t).\end{aligned}\tag{3.2}$$

This model will be called *Linear Velocity* (LV).

3.3.2. Polynomial Velocity Models

The *Polynomial Velocity Models* (PV) are not based on a physical observation, rather they simply use a higher order polynomial to better fit the data:

$$\begin{aligned}\xi_1(t+1) &= \xi_1(t) + \xi_5(T)T_s \cos(\xi_3(t)) \\ \xi_2(t+1) &= \xi_2(t) + \xi_5(T)T_s \sin(\xi_3(t)) \\ \xi_3(t+1) &= \xi_3(t) + \xi_4(T)T_s \\ \xi_4(t+1) &= \xi_4(t) \\ \xi_5(t+1) &= \sum_{i=1}^n d_i \xi_5(t)^i\end{aligned}\tag{3.3}$$

Where n is the order of the polynomial picked. The orders that will be tested are the 2nd and 3rd.

3.3.3. Distance Model

The *Distance Models* (DM) are formulated based on the hypothesis of the velocity being estimated differently based on the distance with respect to the observer. Two attempts are done, the first one, called DM1, is the following:

$$\begin{aligned}\xi_1(t+1) &= \xi_1(t) + \xi_5(t)T_s \cos(\xi_3(t)) \\ \xi_2(t+1) &= \xi_2(t) + \xi_5(t)T_s \sin(\xi_3(t)) \\ \xi_3(t+1) &= \xi_3(t) + \xi_4(t)T_s \\ \xi_4(t+1) &= \xi_4(t) \\ \xi_5(t+1) &= \xi_5(t)[1 + d_1\sqrt{\xi_1(t)^2 + \xi_2(t)^2}]\end{aligned}\tag{3.4}$$

where the perceived acceleration is tied to the distance from the observer. A second attempt, called DM2, is done:

$$\begin{aligned}\xi_1(t+1) &= \xi_1(t) + \xi_5(t)T_s \cos(\xi_3(t)) \\ \xi_2(t+1) &= \xi_2(t) + \xi_5(t)T_s \sin(\xi_3(t)) \\ \xi_3(t+1) &= \xi_3(t) + \xi_4(t)T_s \\ \xi_4(t+1) &= \xi_4(t) \\ \xi_5(t+1) &= \xi_5(t)[1 + d_1\xi_1(t)^2 + d_2\xi_2(t)^2]\end{aligned}\tag{3.5}$$

where the acceleration is still perceived differently based on the distance from the observer, but this time a distinction is made along the ξ_1 and ξ_3 states, or x_1 and x_3 axes. The distances that appear in the equation of ξ_5 are elevated by 2 so to nullify the effect of the sign on the velocity.

4 | Cognitive Model

4.1. Introduction

The Cognitive model has two important tasks to accomplish:

- Predict the pedestrian's motion using the kinematic model (see ch. 3), when the pedestrian is not in sight
- Correct the estimates using the measurements of the pedestrian's position and heading angle, where these are given from the Perceptive process (see ch. 2.2).

These two tasks must be accomplished giving the same performance as the average human would, so the aim is to mimic the same performance as the candidates' in the CXP dataset (see ch. 2.4), not design a filter capable of filtering the actual position of the pedestrian. Since the Cognitive model divides the Cognition in two parts, prediction and update, a natural fit for the Cognitive model is a Bayesian filter.

4.1.1. EKF

The EKF is, given a dynamical system affected by white noise, a mathematical model capable of estimating the state in the current timestep (or in short, filtering) while keeping the covariance of such estimate as low as possible (see [3] and [20]), for this reason the KF and EKF are called optimal filters. In [17] the *Kalman Filter* (KF) and the EKF were introduced as possible filters, and in [18] all the formulated models use the EKF. This is a good starting point for the Cognitive model, since it is a Bayesian filter designed to work on dynamical systems, but it inherits the optimality of the estimation as well, something that is not necessary (see alg. 4.1).

It can be noticed how, compared to alg. 2.1, the covariance dynamics evolves with a different formula, this is the case since the gain used by the EKF allows for some simplifications (see [3]).

The ability in mimicking the human observer of this filter will depend on the values of

Algorithm 4.1 Extended Kalman Filter (EKF)

Require: Initial state estimate $\hat{\xi}(0)$,
 Initial State Covariance $\mathbf{P}(0)$,
 Perception Model Covariance \mathbf{R} ,
 Cognitive Model Covariance \mathbf{Q}

```

1: for  $k = 1$  to  $N$  do
2:   Prediction:
3:    $\hat{\xi}(t|t-1) = f(\hat{\xi}(t-1))$                                 {Predicted state estimate}
4:    $\mathbf{F}(t) = \frac{\partial f}{\partial \xi}(\hat{\xi}(t-1))$                 {State transition matrix}
5:    $\mathbf{P}(t|t-1) = \mathbf{F}(t)\mathbf{P}(t-1)\mathbf{F}(t)^T + \mathbf{Q}$     {Predicted state covariance}
6:   Update:
7:    $\mathbf{H}(t) = \frac{\partial h}{\partial \xi}(\hat{\xi}(t|t-1))$                 {Measurement Jacobean matrix}
8:   if  $\zeta(t)$  is known then
9:      $\epsilon = 1$ 
10:   else
11:      $\epsilon = 0$ 
12:   end if
13:    $e(t) = \zeta(t) - \mathbf{H}(t)\hat{\xi}(t|t-1)$                       {Residual computation}
14:    $\mathbf{S}(t) = \mathbf{H}(t)\mathbf{P}(t|t-1)\mathbf{H}(t)^T + \mathbf{R}$         {Innovation covariance}
15:    $\mathbf{K}(t) = \epsilon \mathbf{P}(t|t-1)\mathbf{H}(t)^T \mathbf{S}(t)^{-1}$       {Gain computation}
16:    $\hat{\xi}(t) = \hat{\xi}(t|t-1) + \mathbf{K}(t)e(t)$                          {Updated state estimate}
17:    $\mathbf{P}(t) = (\mathbf{I} - \mathbf{K}(t)\mathbf{H}(t))\mathbf{P}(t|t-1)$        {Updated state covariance}
18: end for
19: return  $\hat{\xi}(t)$  and  $\mathbf{P}(t)$ 

```

the \mathbf{Q} matrix, since it influences how the filter's gain is computed, and consequently how the state's covariance evolves and the speed at which the state estimate converges to the measured position and heading.

4.1.2. Modified Extended Kalman Filter

The *Modified Extended Kalman Filter* (MEKF) takes the EKF and tries to adapt the gain so to make the Cognitive model fit the results better. To do so a Hadamard product, or element wise product (\odot), is performed so to weight the elements of the filter's gain through the matrix \mathbf{K}_w .

The reason why the MEKF is introduced is that, while trying to make the EKF less optimal it still retains its relationship with \mathbf{P} . This is important since it means that the filter computes the gain also to accommodate how the state's covariance evolves, so if for example it is too big the gain will be computed accordingly to dimension it with a proper rate (See alg. 4.2).

It is noticeable in the algorithm that the covariance dynamics does not use the same formula as the EKF, rather it uses the Joseph form as in alg. 2.1, this is the case since the gain is not the optimal one anymore.

4.1.3. Luenberger Observer

The Luenbergher observer is a mathematical model capable of estimating the states of a linear system, or one well approximated by its linearization, using a static gain \mathbf{k} . More specifically it is possible to compute a gain such that the dynamics of the error between the state and its estimate is asymptotically stable when null (see [26]).

In the Cognitive model's case using a static gain to filter is an alternative way to try recreating a nonoptimal process, where the relationship between the gain \mathbf{K} and the covariance \mathbf{P} is not present anymore, consequently the current value of \mathbf{P} does not influence the update phase of the filter, this means that for example if the initial covariance value \mathbf{P}_0 is too big the filter might not be able to reduce it in a reasonable number of steps, since \mathbf{k} will not adapt to reduce it quicker. It is also to be noticed that a gain not dependent on the covariance decouples the estimated states' dynamics from the covariance's dynamics, so the filter's performance is not influenced by the covariance's evolution anymore.

Algorithm 4.2 Modified Extended Kalman Filter (MEKF)

Require: Initial state estimate $\hat{\xi}(0)$,
 Initial State Covariance $P(0)$,
 Perception Model Covariance R ,
 Cognitive Model Covariance Q ,
 Gain multiplicative modifier K_w

```

1: for  $k = 1$  to  $N$  do
2:   Prediction:
3:      $\hat{\mathbf{x}}(t|t-1) = f(\hat{\mathbf{x}}(t-1))$                                 {Predicted state estimate}
4:      $\mathbf{F}(t) = \frac{\partial f}{\partial t}(\hat{\mathbf{x}}(t-1))$                       {State transition matrix}
5:      $\mathbf{P}(t|t-1) = \mathbf{F}(t)\mathbf{P}(t-1)\mathbf{F}(t)^T + \mathbf{Q}$           {Predicted state covariance}
6:   Update:
7:      $\mathbf{H}(t) = \frac{\partial h}{\partial \mathbf{x}}(\hat{\mathbf{x}}(t|t-1))$                 {Measurement Jacobean matrix}
8:     if measurement available at time step  $k$  then
9:        $\epsilon = 1$ 
10:    else
11:       $\epsilon = 0$ 
12:    end if
13:     $\mathbf{e}(t) = \zeta(t) - \mathbf{H}(t)\hat{\mathbf{x}}(t|t-1)$                          {Residual computation}
14:     $\mathbf{K}(t) = \epsilon \mathbf{P}(t|t-1) \mathbf{H}(t)^T (\mathbf{H}(t)\mathbf{P}(t|t-1)\mathbf{H}(t)^T + \mathbf{R})^{-1} \odot \mathbf{K}_w$  {Unoptimized
        Kalman gain}
15:     $\hat{\mathbf{x}}(t) = \hat{\mathbf{x}}(t|t-1) + \mathbf{K}(t)\mathbf{e}(t)$                      {Updated state estimate}
16:     $\mathbf{P}(t) = (\mathbf{I} - \mathbf{K}(t)\mathbf{H}(t)) \mathbf{P}(t|t-1) (\mathbf{I} - \mathbf{K}(t)\mathbf{H}(t))^T + \mathbf{K}(t)\mathbf{R}\mathbf{K}(t)^T$  {Updated
        state covariance}
17: end for
18: return  $\hat{\mathbf{x}}(t)$  and  $\mathbf{P}(t)$ 

```

Algorithm 4.3 Luenberger Observer

5 | Results

5.1. Introduction

In this chapter an overview on how the parameters of the models have been obtained will be given, then the results regarding the Cognitive model with the different kinematic models and filter gains will be shown.

5.2. Optimization Procedure

Each model has its own set of parameters that need to be identified, and to do so an optimization problem has been formulated (see equations (5.1a) to (5.1p)).

Before the optimization begins all the trajectories, that as states in ch. 2.4 are uniformly randomly spawned in a circular area, are shifted in the nominal spawning point, or the center of the circle, and consequently the estimate of the trajectories are as well by the same amount. This is done to avoid influencing the final distribution ν_E by the random spawn.

$$\min_{\lambda} \mathbf{C}(\boldsymbol{\nu}_E, \boldsymbol{\nu}_M) \quad (5.1a)$$

s.t.

$$\hat{\xi}_i(t|t-1) = f(\hat{\xi}_i(t-1), \boldsymbol{\lambda}) \quad \forall t = 0, \dots, N_T \quad \forall i = 1, \dots, N_D \quad (5.1b)$$

$$\hat{\xi}_i(0|0) = \boldsymbol{\xi}_0 \quad \forall i = 1, \dots, N_D \quad (5.1c)$$

$$\mathbf{F}_i(t) = \frac{\partial f}{\partial \hat{\xi}_i}(\hat{\xi}_i(t-1), \boldsymbol{\lambda}) \quad \forall t = 0, \dots, N_T \quad \forall i = 1, \dots, N_D \quad (5.1d)$$

$$\mathbf{P}_i(t|t-1) = \mathbf{F}_i(t) \mathbf{P}_i(t-1) \mathbf{F}_i(t)^T + \mathbf{Q}(\boldsymbol{\lambda}) \quad \forall t = 0, \dots, N_T \quad \forall i = 1, \dots, N_D \quad (5.1e)$$

$$\mathbf{P}_i(0|0) = \mathbf{P}_0 \quad \forall i = 1, \dots, N_D \quad (5.1f)$$

$$\mathbf{H} = \frac{\partial h}{\partial \hat{\xi}_i}(\hat{\xi}_i(t|t-1)) \quad \forall t = 0, \dots, N_T - 1 \quad \forall i = 1, \dots, N_D \quad (5.1g)$$

$$\epsilon_i(t) = \begin{cases} 1 & \text{if } \boldsymbol{\zeta}_i(t) \text{ is available} \\ 0 & \text{otherwise} \end{cases} \quad \forall t = 0, \dots, N_T \quad \forall i = 1, \dots, N_D \quad (5.1h)$$

$$\mathbf{e}_i(t) = \boldsymbol{\zeta}_i(t) - \mathbf{H}_i(t) \hat{\xi}_i(t|t-1) \quad \forall t = 0, \dots, N_T \quad \forall i = 1, \dots, N_D \quad (5.1i)$$

$$\mathbf{K}_i(t) = \epsilon \mathbf{R}(\cdot) \quad \forall t = 0, \dots, N_T - 1 \quad \forall i = 1, \dots, N_D \quad (5.1j)$$

$$\hat{\xi}_i(t|t) = \hat{\xi}_i(t|t-1) + \mathbf{K}_i(t) \mathbf{e}_i(t) \quad \forall t = 0, \dots, N_T - 1 \quad \forall i = 1, \dots, N_D \quad (5.1k)$$

$$\mathbf{P}_i(t|t) = \mathbf{P}_i(t) = (\mathbf{I} - \mathbf{K}(t) \mathbf{H}) \mathbf{P}(t|t-1) (\mathbf{I} - \mathbf{K}(t) \mathbf{H})^T + \mathbf{K}(t) \mathbf{R}(\boldsymbol{\lambda}) \mathbf{K}(t)^T \quad \forall t = 0, \dots, N_T \quad \forall i = 1, \dots, N_D \quad (5.1l)$$

$$\boldsymbol{\mu}_{E,i} = \hat{\xi}_i(N_T|N_T) \quad \forall i = 1, \dots, N_D \quad (5.1m)$$

$$\Sigma_{E,i} = \begin{bmatrix} \mathbf{P}_{1,1,i}(N_T|N_T) & \mathbf{P}_{1,2,i}(N_T|N_T) \\ \mathbf{P}_{2,1,i}(N_T|N_T) & \mathbf{P}_{2,2,i}(N_T|N_T) \end{bmatrix} \quad \forall i = 1, \dots, N_D \quad (5.1n)$$

$$\boldsymbol{\nu}_{E,i} = \mathbf{N}(\boldsymbol{\mu}_{E,i}, \Sigma_{E,i}) \quad \forall i = 1, \dots, N_D \quad (5.1o)$$

$$\boldsymbol{\nu}_E = \begin{bmatrix} \boldsymbol{\nu}_{E,1} \\ \vdots \\ \boldsymbol{\nu}_{E,N_D} \end{bmatrix} \quad (5.1p)$$

Where N_T and N_D are respectively the time steps and the number of trajectories in the dataset.

The constraints of the problem are the equations of the Cognitive model, where the function $f(\hat{\xi}(t), \boldsymbol{\lambda})$ is the kinematic model used, and $\mathbf{K}_i(t) = \epsilon \mathbf{R}(\cdot)$ is the filter used.

$\boldsymbol{\lambda}$ is the set of parameters, that will be identified so to minimize the cost function

$\mathbf{C}(\boldsymbol{\nu}_E, \boldsymbol{\nu}_M)$. Each kinematic model and filter gain combination has a different set of parameters, that will be introduced in this chapter.

The cost function $\mathbf{C}(\boldsymbol{\nu}_E, \boldsymbol{\nu}_M)$ used for all the models is the Log Likelihood one:

$$LL = \sum_{i=1}^{N_D} \frac{1}{2} (\boldsymbol{\mu}_{E,i} - \boldsymbol{\mu}_{M,i})^T \Sigma_{E,i}^{-1} (\boldsymbol{\mu}_{E,i} - \boldsymbol{\mu}_{M,i}) + \frac{1}{2} \log(2\pi \det(\Sigma_{E,i})). \quad (5.2)$$

The value of (5.2) is a measure of how well the model's distribution $\boldsymbol{\nu}_E$ fits the dataset one $\boldsymbol{\nu}_M$, so it will be also used to compare the different models.

Another cost function has been used to assist in the optimization of some models, the square of the 2-norm of the difference between the means of the model's and dataset's distributions:

$$MD = \sum_{i=1}^{N_D} \|\boldsymbol{\mu}_{E,i} - \boldsymbol{\mu}_{M,i}\|_2^2. \quad (5.3)$$

This cost function was introduced to assist in the optimization procedure in case it was hard to find a suitable set of initial conditions for $\boldsymbol{\lambda}$. Its objective is to identify a set of parameters that gets the model to yield results where the mean values are as close as possible to the experimental ones, so when LL is applied afterwards the optimization starts from a result that is not random.

An alternative way of optimizing the models when it was not known where to start with the initial conditions was the use of a genetic algorithm.

Both procedures described now to begin the optimization process have their problems, indeed in the MD case the risk was of obtaining a model that gets close to the mean, yet with covariance values so high that using LL afterwards was problematic. The genetic algorithm use instead had the flaw of being extremely slow, moreover numerical errors could block the algorithm even after a long time running.

The optimization procedure was done using the *fmincon* function on *MATLAB R2023a* with an interior point method. For the genetic algorithm instead *ga* was used.

5.3. Results

Since there are many models to compare this procedure will be followed:

1. the different kinematic models will be compared based on the Log Likelihood Cost function score and the distribution of the Mahalanobis distances between the model's distributions and the dataset's means, using a common filter gain to make the comparison fair
2. the different filters will be compared using the best kinematic model found in the previous step
3. the best result will be tested with respect to the dataset using the Kolmogorov Smirnov

5.3.1. Comparison of the Kinematic Models

In this chapter the different Kinematic Models will be compared, using the EKF as common filter.

In all the models the Cognitive model covariance will have this form:

$$\mathbf{Q} = \begin{bmatrix} q_1 & 0 & 0 & 0 & 0 \\ 0 & q_2 & 0 & 0 & 0 \\ 0 & 0 & q_3 & 0 & 0 \\ 0 & 0 & 0 & q_4 & 0 \\ 0 & 0 & 0 & 0 & q_5 \end{bmatrix} \quad (5.4)$$

where the diagonal values are all parameters to be determined. On top of \mathbf{Q} there will be other parameters such as the ones introduced by the different Kinematic models, called d_i , and the 3 parameters introduced for the Perceptive model, r_1 , r_2 and r_3 (see ch. 2.2).

In all the optimization procedures the initial conditions where:

$$\mathbf{P}_0 = \begin{bmatrix} 100000 & 0 & 0 & 0 & 0 \\ 0 & 100000 & 0 & 0 & 0 \\ 0 & 0 & 100000 & 0 & 0 \\ 0 & 0 & 0 & 100000 & 0 \\ 0 & 0 & 0 & 0 & 100000 \end{bmatrix} \quad (5.5)$$

$$\boldsymbol{\xi}_0 = \left[\zeta_1(1) \ \zeta_2(1) \ 0 \ 0 \ 0 \right]^T$$

In the implementation of the EKF the Joseph form has been favorite with respect to the form seen in alg. 4.1.1 of the update of \mathbf{P} , since the latter is more prone to numerical errors, so (5.11) is accurate even when using the EKF.

	Unicycle	Linear Velocity	2 nd Order Polynomial	3 rd Order Polynomial	Distance Model 1	Distance Model 2
Total Cost	5349.8050	4985.2352	4790.7459	4790.0375	4962.9468	5053.3937
Cost $T_{est} = 2$ s	2323.3049	2041.9935	1927.3825	1926.0131	2026.7464	2068.6937
Cost $T_{est} = 3$ s	3026.5000	2943.2417	2863.3633	2864.0244	2936.2004	2984.7000

Table 5.1: Final Log Likelihood cost function values. The kinematic model used is indicated in the first row, the filter used is the EKF.

The first row is the total cost, the second and third are the cost when $T_{est} = 2$ s and $T_{est} = 3$ s respectively.

Looking at the results in tab. 5.1 it is clear that the polynomial models are the better performing ones. Since the difference in cost between the two polynomial models is very low then the 2nd Order Polynomial will be considered as the better one.

To better appreciate the improvements done on the velocity equation the 2nd Order Polynomial model, the best result obtained, will be compared with the Unicycle model, the starting point of the design of the kinematic model. The results obtained by both models are plotted in fig. 5.1, fig. 5.2, fig. 5.3 and fig. 5.4, where it is clearly visible how the Unicycle model is not capable of keeping up with the pace of the dataset, while the 2nd Order Polynomial model is.

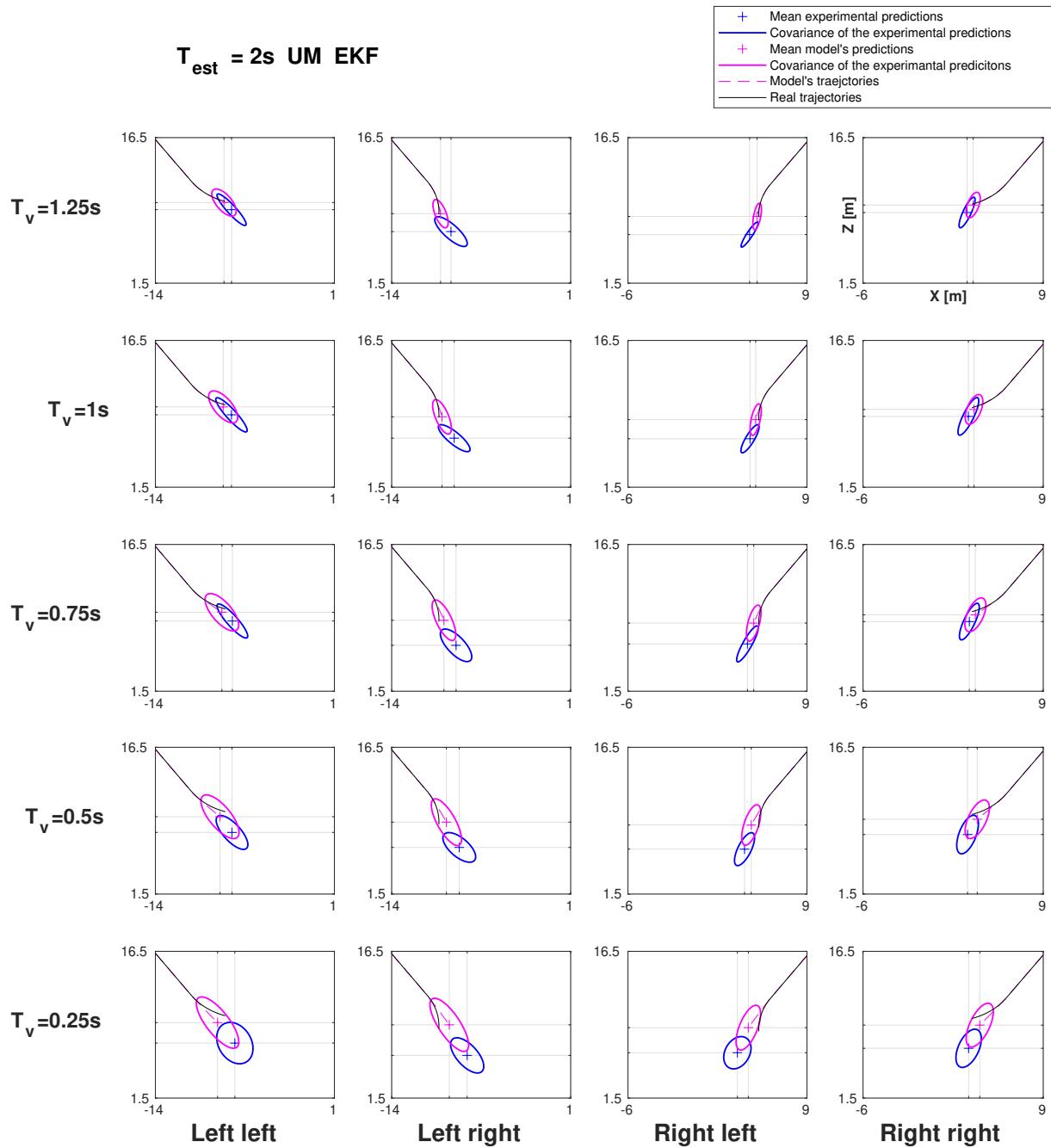


Figure 5.1: Results obtained with the Unicycle Kinematic Model and EKF filter, $T_{est} = 2\text{s}$

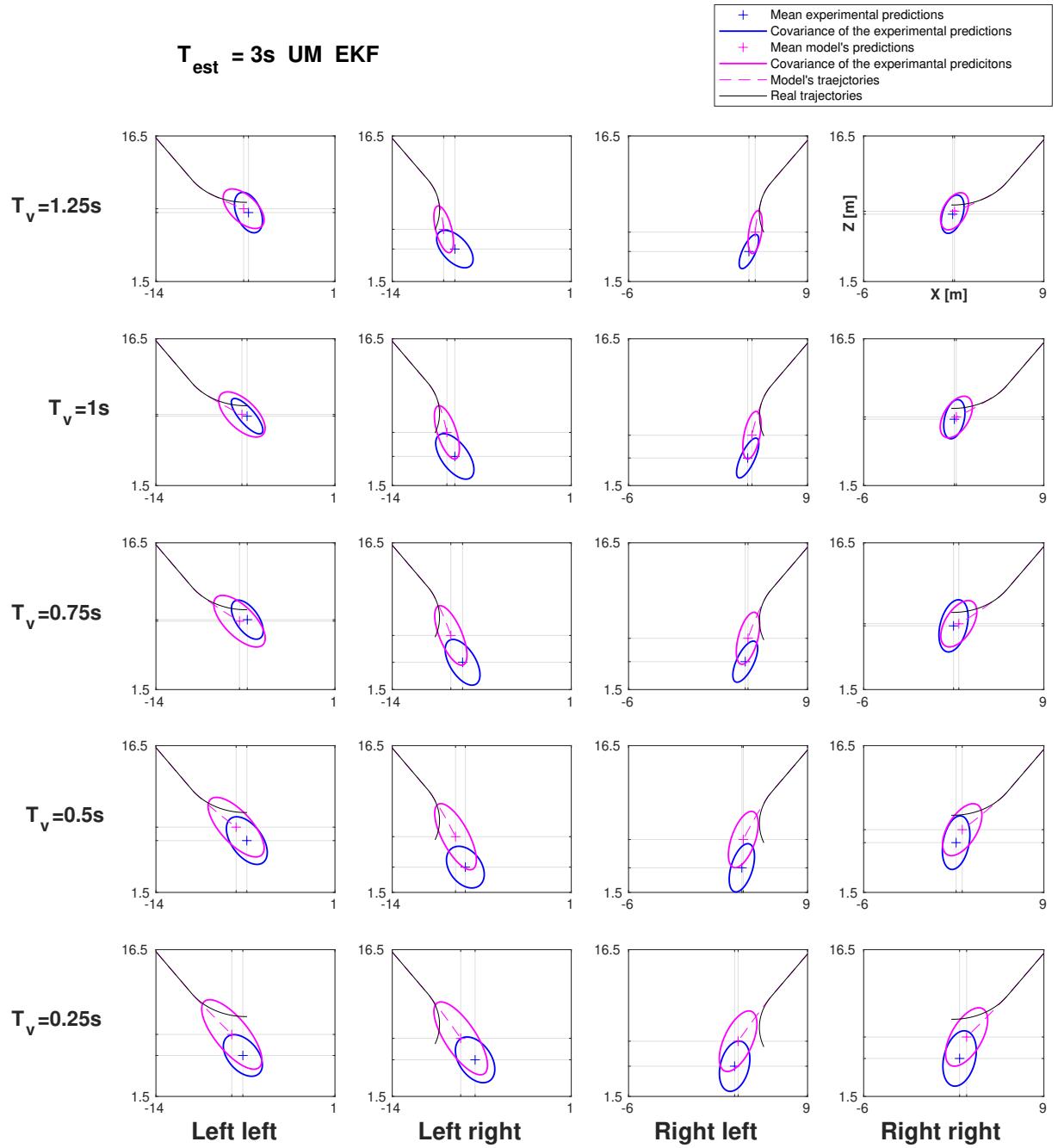


Figure 5.2: Results obtained with the Unicycle Kinematic Model and EKF filter, $T_{\text{est}} = 3\text{s}$

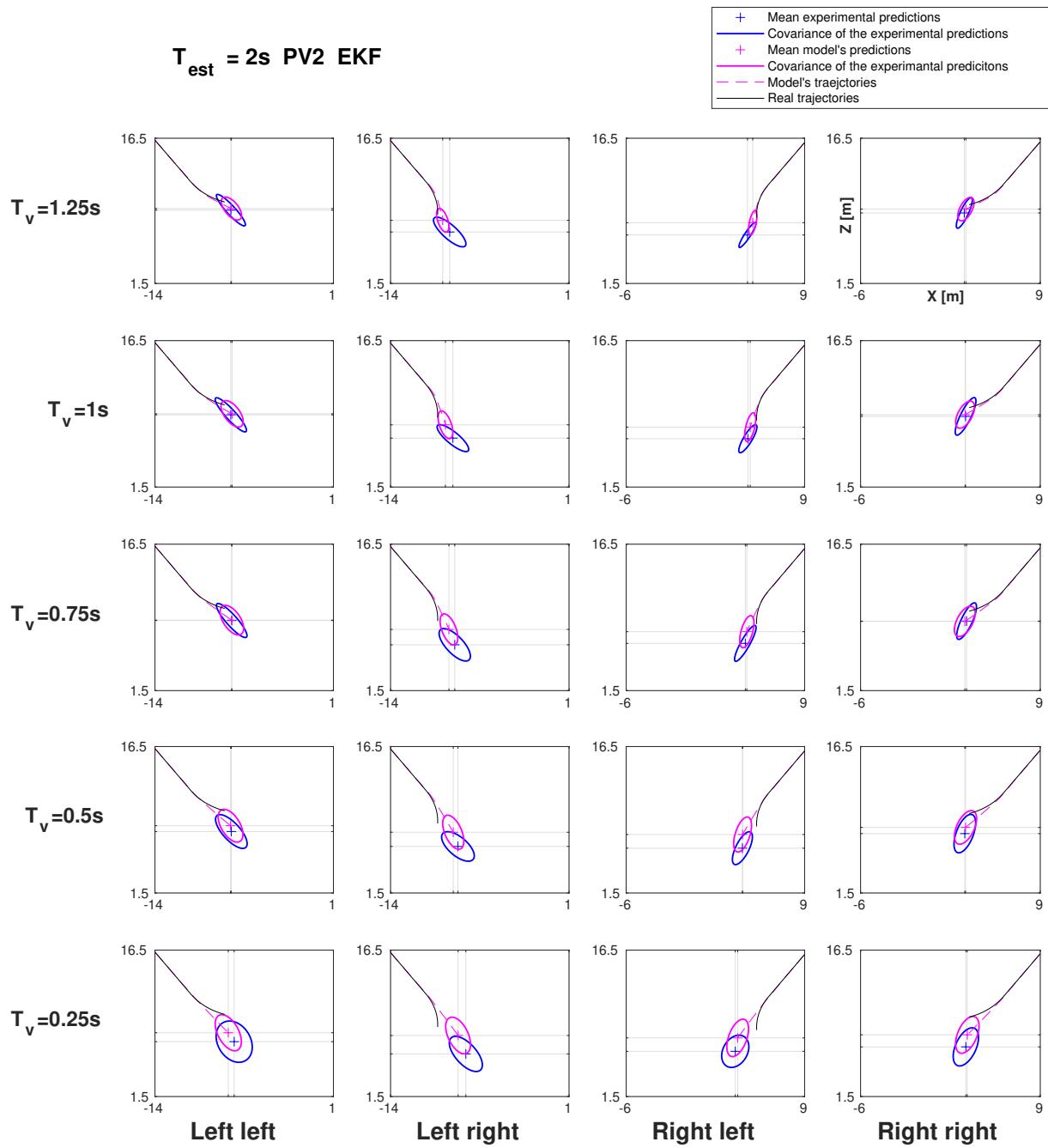


Figure 5.3: Results obtained with the 2nd Order Polynomial Kinematic Model and EKF filter, $T_{\text{est}} = 2\text{s}$

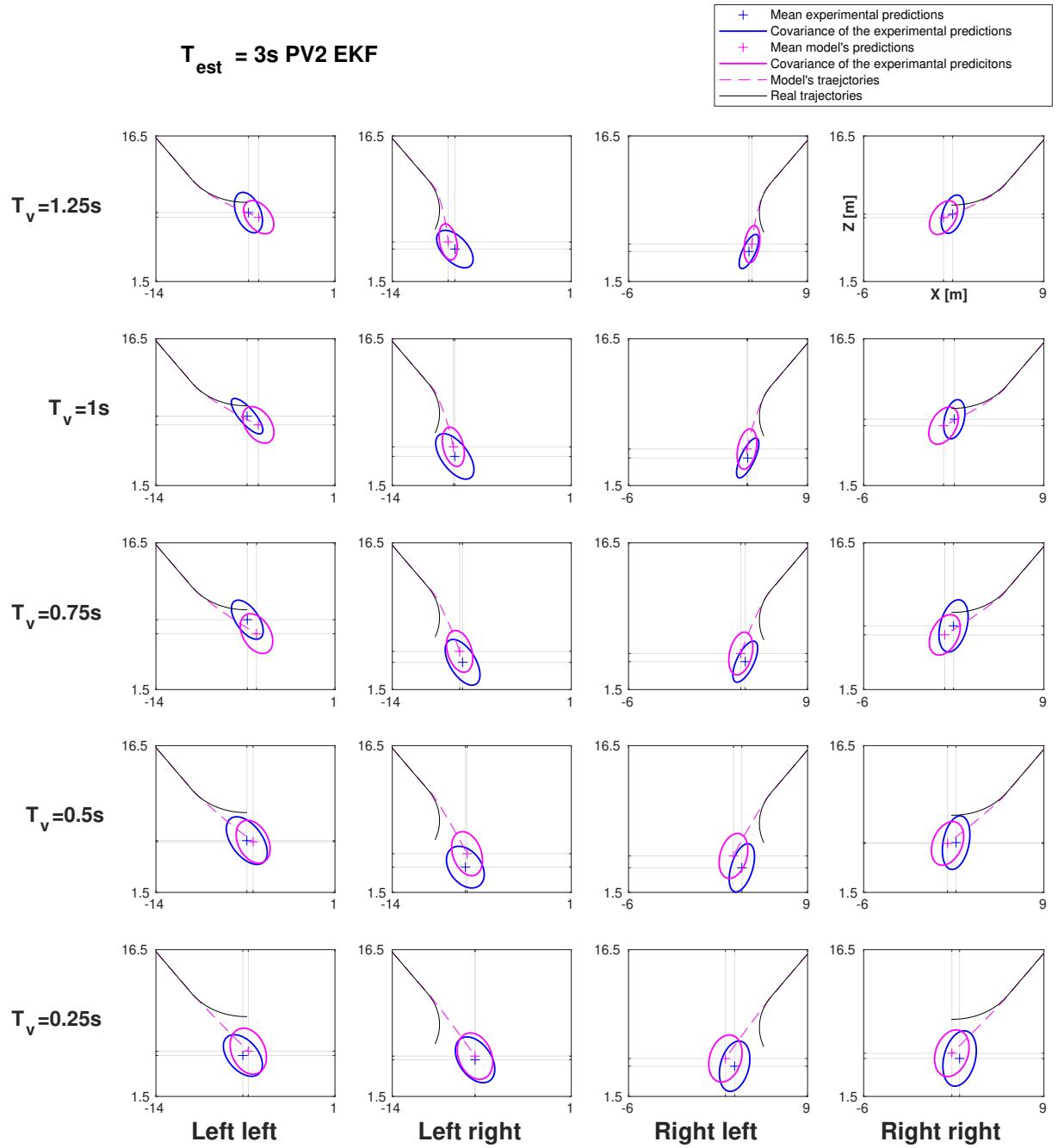


Figure 5.4: Results obtained with the 2nd Order Polynomial Kinematic Model and EKF filter, $T_{est} = 3\text{ s}$

In tab. 5.1 it is noticeable how the Log-Likelihood cost of the trajectories when $T_{est} = 3$ s are all bigger. This is due to the fact that the covariances when $T_{est} = 3$ s are bigger than the ones when $T_{est} = 2$ s, as seen in tab. 2.6. This matters since it means that the cost function will be weighted more by the trajectories that have $T_{est} = 3$ s, and consequently the optimizer will favor a solution that improves those rather than the ones with $T_{est} = 2$ s, this fact is noticeable in the Mahalanobis distances distributions, as it can be seen in fig. 5.5, where in all the models the distances when $T_{est} = 3$ s are smaller than the ones when $T_{est} = 2$ s.

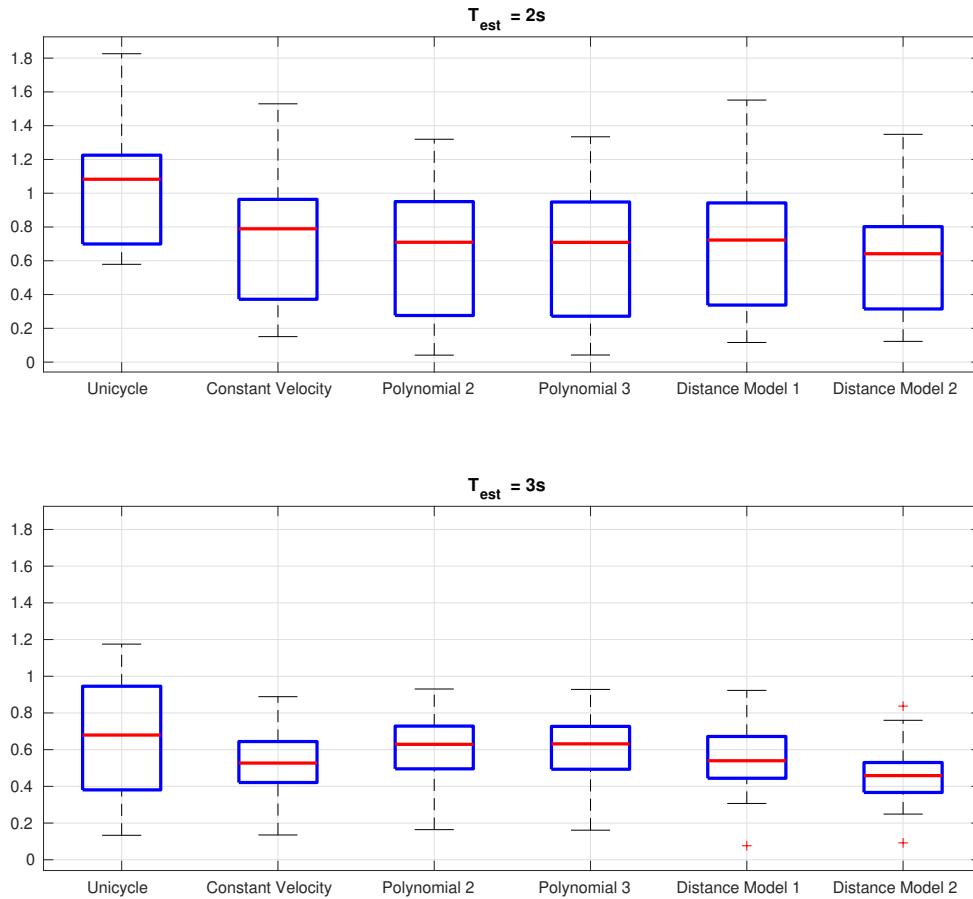


Figure 5.5: Boxplot of the Mahalanobis distances between the means of the dataset and the model's distributions

Looking at fig. 5.5 it can also be noticed that the distributions given by the MOI Left right and Right left when $T_{est} = 2$ s yield worse results. This observation might be quantified by considering the Mahalanobis distance between the model's distributions and dataset's

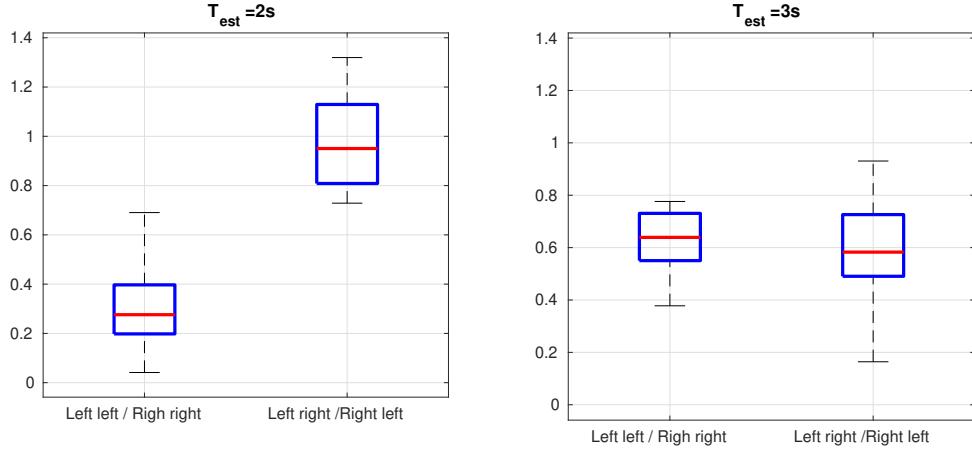


Figure 5.6: Boxplot of the Mahalanobis distances between the means of the dataset and the model's distributions, where the dataset is divided based on the MOI, as indicated on the abscissa.

Kinematic Model	q_1	q_2	q_3	q_4	q_5	d_1	d_2	d_3
Unicycle	1.1985e-4	1.399e-2	1.8625e-4	4.8580e-11	2.2640e-3	X	X	X
Linear Velocity	6.5355e-11	9.5401e-3	1.8044e-4	1.9830e-12	5.4698e-4	X	X	X
Polynomial (2 nd Order)	8.6116e-11	1.0403e-2	1.7581e-4	9.2550e-12	2.5210e-2	1.0157	-0.0063	X
Polynomial (3 rd Order)	9.0035e-9	1.0357e-2	1.7694e-4	9.2679e-10	2.4052e-2	1.0246	-0.0141	0.0016
Distance Model 1	1.3415e-8	9.4669e-3	1.7778e-4	2.0217e-10	6.1178e-4	8.7469e-5	X	X
Distance Model 2	1.1985e-4	1.3989e-2	1.8625e-4	4.8581e-11	2.2639e-3	-4.2870e-6	-1.5283e-8	9.4884e-6

Table 5.2: Cognitive Model parameters, q_i are the diagonal values of the covariance matrix, d_i are the parameters introduced by the kinematic models

means, where the dataset is divided based on the MOI, as it can be seen in fig. 5.6. This result might be given by the problem states in ch. 2.4, where the crossroads might have influenced the Cognitive process of the candidates.

In tab. 5.2 and tab. 5.3 the parameters of the models are shown, where the q_i are the diagonal values of the Cognitive model covariance matrix, the d_i are the parameters introduced by the kinematic models, and the r_i are the parameters introduced by the perceptive model.

5.3.2. Comparison of the Filters

Now the 2nd order polynomial kinematic model will be used to compare the different filters, where the EKF has already been seen in the previous chapter.

Starting from the MEKF, the parameters to identify are the same as the ones introduced by the kinematic model with the addition of the values of the matrix \mathbf{K}_w introduces in alg. 4.2:

Kinematic Model	r_1	r_2	r_3
Unicycle	1.5025e-8	3.5405e-2	3.5405e-2
Linear Velocity	1.9702e-9	1.3249e-8	3.5229e-2
Polynomial (2 nd Order)	7.2112e-9	3.3222e-8	4.0832e-2
Polynomial (3 rd Order)	7.2128e-7	3.3918e-6	4.0439e-2
Distance Model 1	1.8782e-4	4.0931e-1	4.9500e-1
Distance Model 2	1.5026e-8	8.8567e-7	3.5404e-2

Table 5.3: Perceptive Model parameters

$$\mathbf{K}_w = \begin{bmatrix} k_{1,1} & k_{1,2} & k_{1,3} \\ k_{2,1} & k_{2,2} & k_{2,3} \\ k_{3,1} & k_{3,2} & k_{3,3} \\ k_{4,1} & k_{4,2} & k_{4,3} \\ k_{5,1} & k_{5,2} & k_{5,3} \end{bmatrix}. \quad (5.6)$$

The initial conditions are the same as for ch. 5.3.1.

For the Luenberger model instead, the gain itself becomes parametrized, where each element in the K matrix is obtained through the optimization procedure:

$$\mathbf{K} = \begin{bmatrix} k_{1,1} & k_{1,2} & k_{1,3} \\ k_{2,1} & k_{2,2} & k_{2,3} \\ k_{3,1} & k_{3,2} & k_{3,3} \\ k_{4,1} & k_{4,2} & k_{4,3} \\ k_{5,1} & k_{5,2} & k_{5,3} \end{bmatrix}. \quad (5.7)$$

Regarding the initial conditions, they are not the same as for the other models, indeed, as already stated in ch. 4.1.3, due to its inability to reduce the covariance in few steps the initial covariance has been set to low values, where the exact values are not important. The picked \mathbf{P}_0 used in the results is the following:

$$\mathbf{P}_0 = \begin{bmatrix} 7.07e-3 & 0 & 0 & 0 & 0 \\ 0 & 1.42e-2 & 0 & 0 & 0 \\ 0 & 0 & 6.52e-6 & 0 & 0 \\ 0 & 0 & 0 & 3.80e-6 & 0 \\ 0 & 0 & 0 & 0 & 2.20e-4 \end{bmatrix} \quad (5.8)$$

where these values were obtained by optimizing the elements in \mathbf{P}_0 as well, still the exact values are not important. The states initial conditions instead remain the same.

	Extended Kalman Filter	Modified Extended Kalman Filter	Luenberger
Total Cost	4790.7459	4571.8204	4729.7592
Cost $T_{est} = 2 \text{ s}$	1927.3825	1811.4940	1874.1231
Cost $T_{est} = 3 \text{ s}$	2863.3633	2760.3263	2855.6361

Table 5.4: Final Log Likelihood cost function values. The filter used is indicated in the first row, the kinematic model is the 2nd order polynomial.

The first row is the total cost, the second and third are the cost when $T_{est} = 2 \text{ s}$ and $T_{est} = 3 \text{ s}$ respectively.

The results, in tab. 5.4, show that the MEKF is the best performing filter, and this proves that a nonoptimal filter is more capable of mimicking the Cognitive process than an optimal one.

The result obtained from the Luenberger gain is also interesting, indeed using a static gain it was possible to get a result comparable to the EKF's.

Something to pay attention to is how the model that uses the Luenberger gain generates trajectories that do not track as well the real one, indeed if its results (fig. 5.7 and fig. 5.8) are compared with the ones obtained with the MEKF (fig. 5.9 and fig. 5.10) and the EKF's (fig. 5.3 and fig. 5.4) it is clear that it is the only model doing so. Due to the way how the data was collected in the CXP, where just the last positions of the trajectories were sampled, there is no way of telling which kind of trajectory is more accurate to the one perceived by the candidates. In fig. 5.11 there is an example of how much the Luenberger gets far from the trajectory with respect to the MEKF.

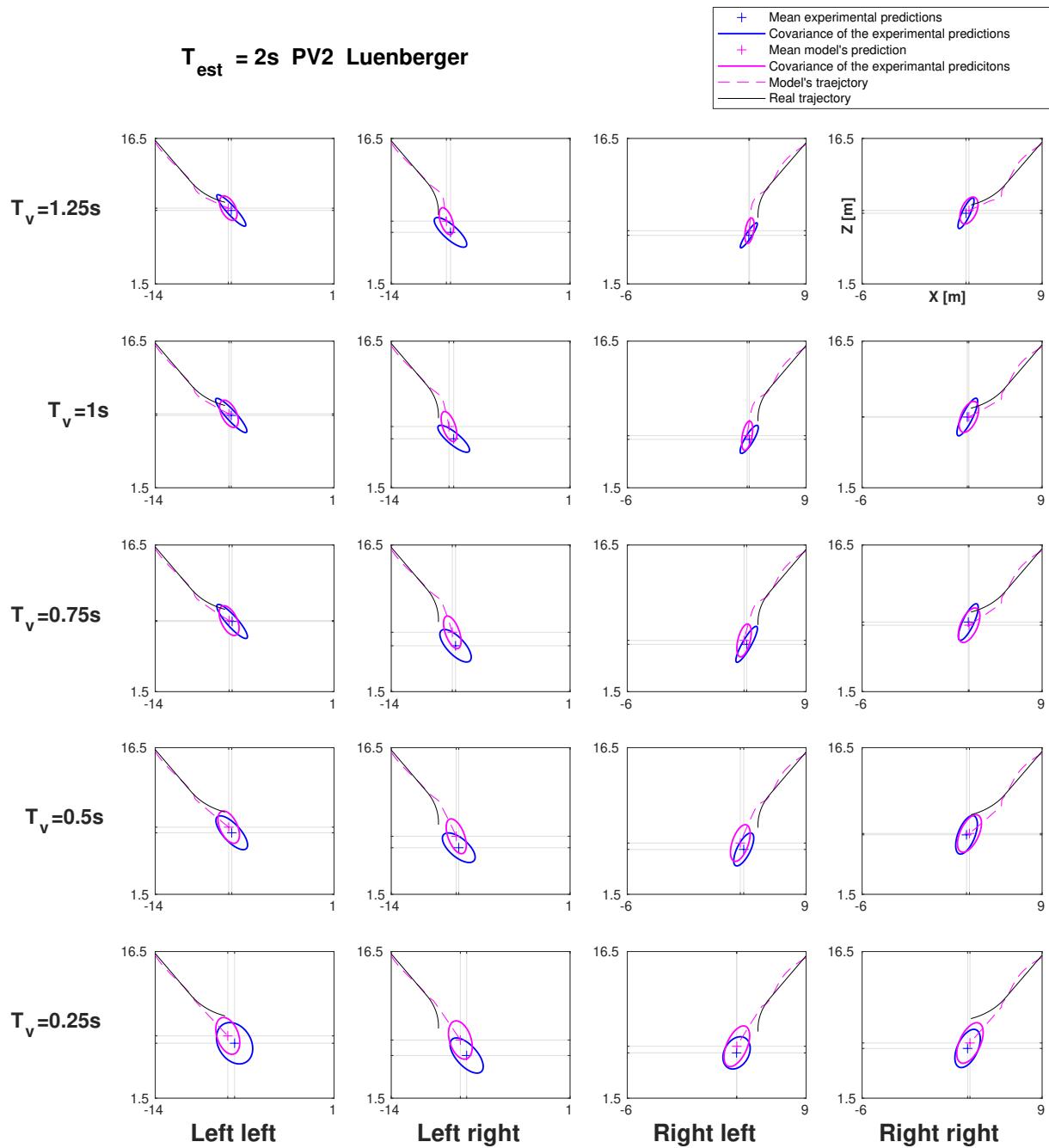


Figure 5.7: Results obtained with the 2nd Order Kinematic Model and Luenberger Observer, $T_{est} = 2\text{ s}$

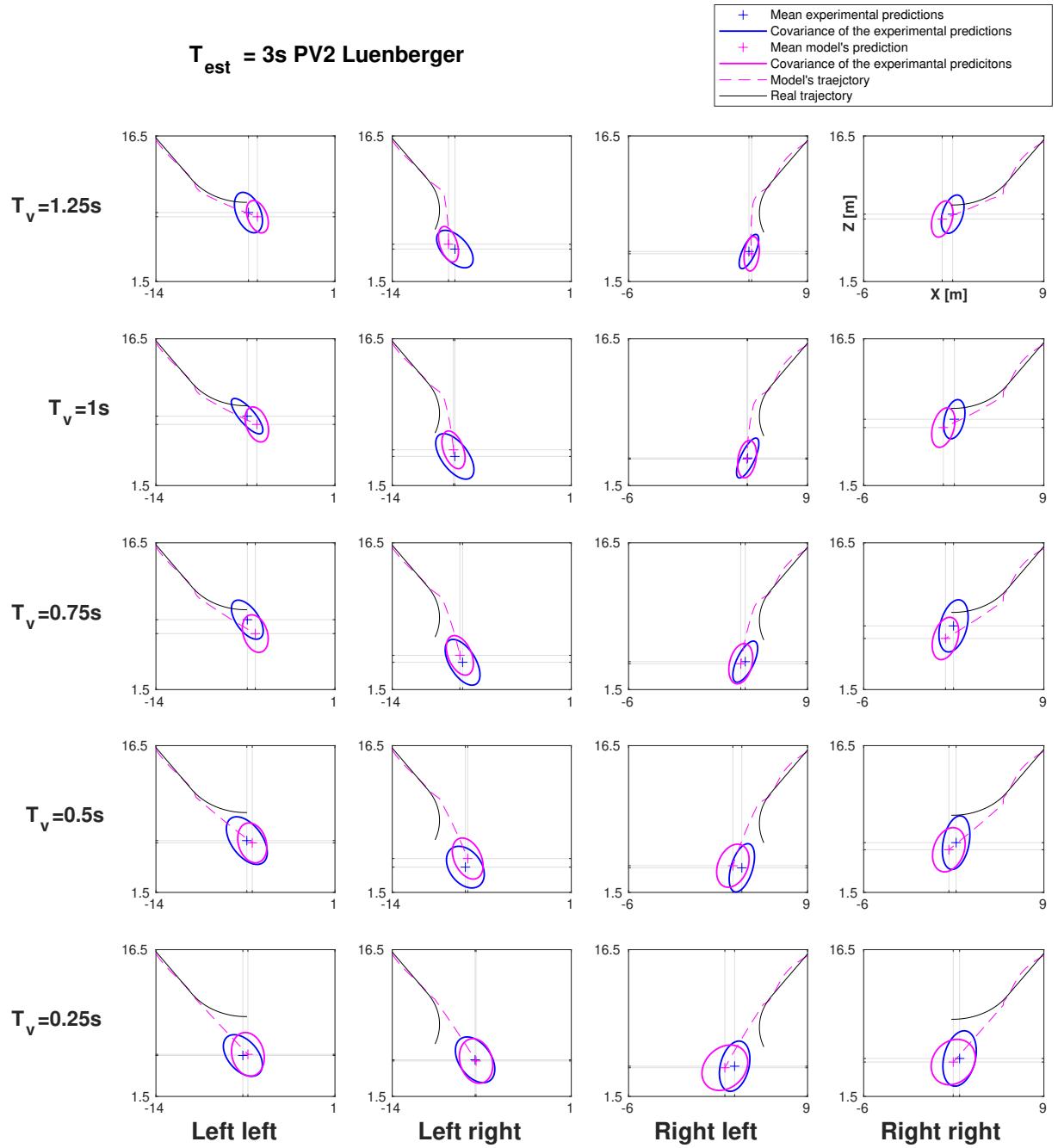


Figure 5.8: Results obtained with the 2nd Order Kinematic Model and Luenberger Observer, $T_{est} = 3\text{ s}$

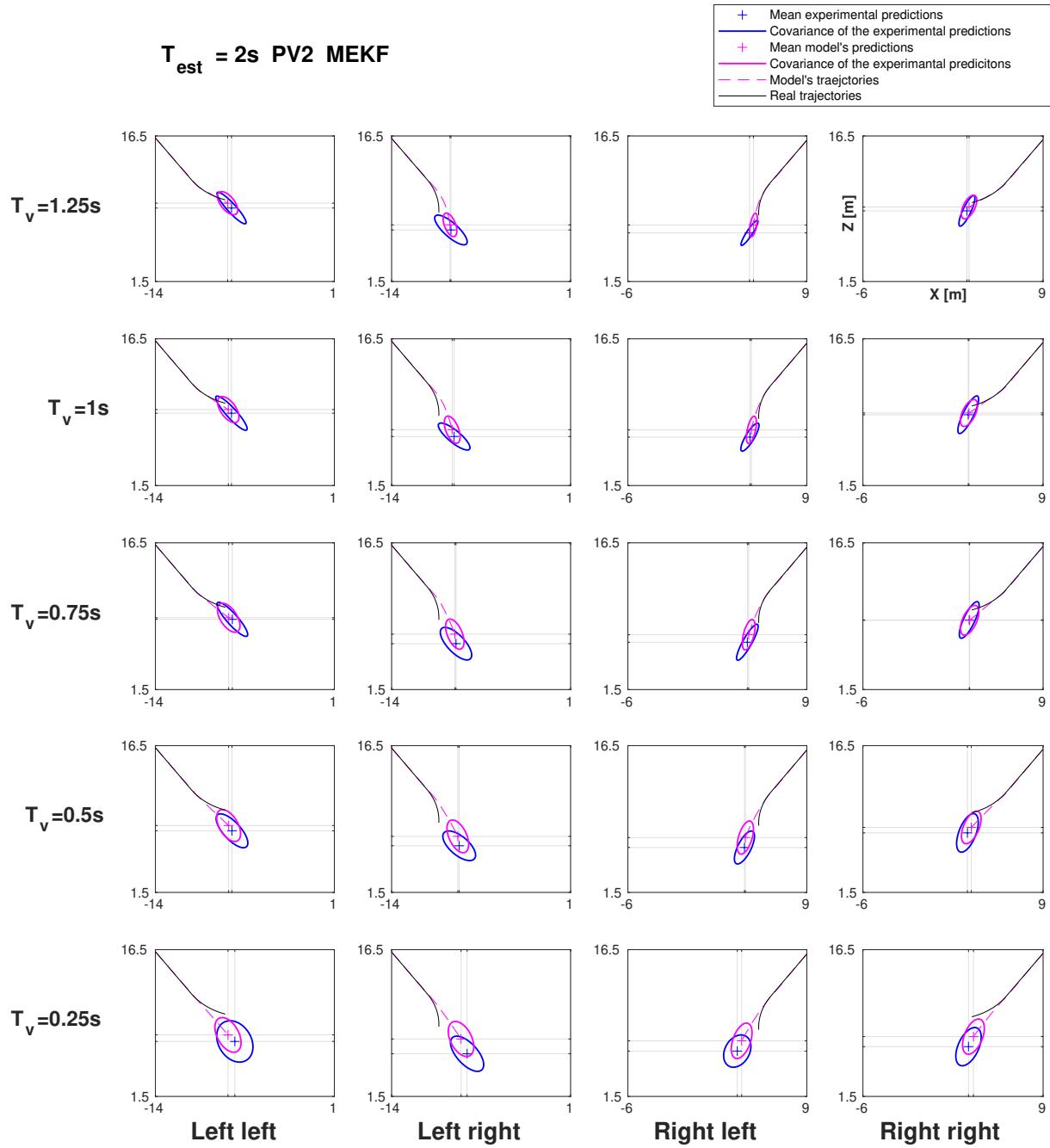


Figure 5.9: Results obtained with the 2nd Order Polynomial Kinematic Model and MEKF filter, $T_{est} = 2$ s

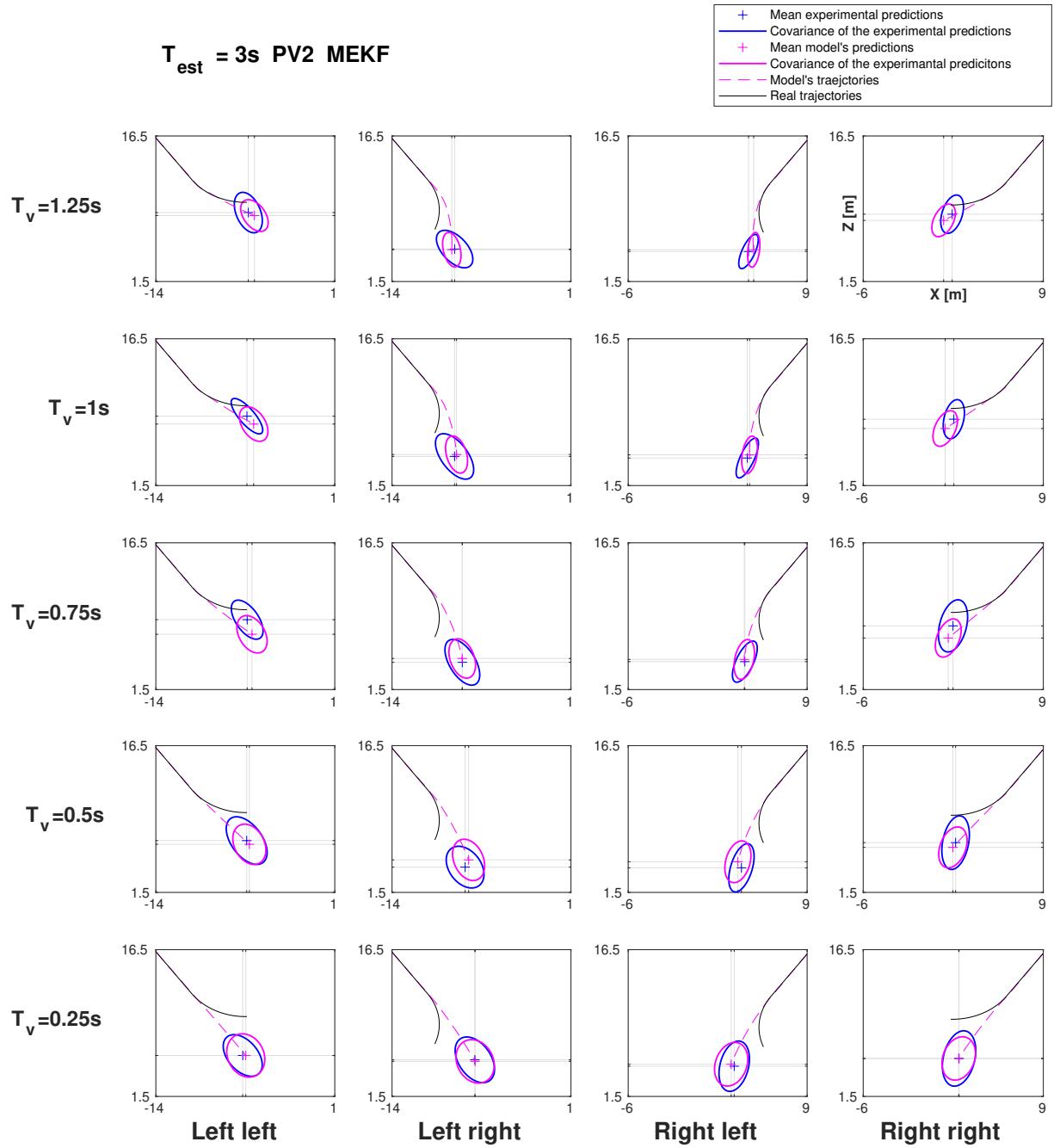


Figure 5.10: Results obtained with the 2nd Order Polynomial Kinematic Model and MEKF filter, $T_{\text{est}} = 3\text{ s}$

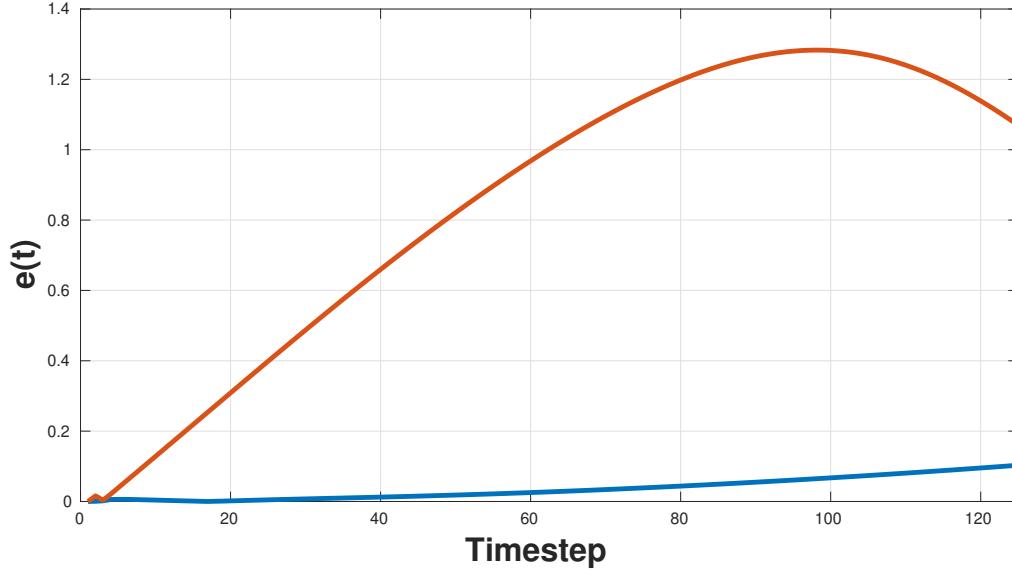


Figure 5.11: Distance between the perceived trajectory and the one computed by the Cognitive model in trajectory Right right $T_v = 1.25$ s, during visibility time
 Blue line: Cognitive model with 2nd order polynomial and MEKF
 Orange line: Cognitive model with 2nd order polynomial and Luenberger

Filter Model	q_1	q_2	q_3	q_4	q_5	d_1	d_2
EKF	8.6116e-11	1.0403e-2	1.7581e-4	9.2550e-12	2.5210e-2	1.0157	-0.0063
MEKF	2.1311e-10	1.0586e-2	3.1617e-11	4.0491e-11	5.1952e-2	1.0276	-1.0806e-2
Luenberger	6.2212e-8	1.0959e-2	6.1333e-9	3.0417e-9	9.9931	1.0538	-2.0506e-2

Table 5.5: Cognitive Model parameters, q_i are the diagonal values of the covariance matrix, d_i are the parameters of the 2nd

Filter Model	r_1	r_2	r_3
EKF	1.5025e-8	3.5405e-2	3.5405e-2
MEKF	2.3149e-3	7.1386e-3	3.2799e-2
Luenberger	6.7623e-5	4.1109e-2	1.1497e-8

Table 5.6: Perceptive Model parameters

Filter Model	$k_{1,1}$	$k_{1,2}$	$k_{1,3}$	$k_{2,1}$	$k_{2,2}$	$k_{2,3}$	$k_{3,1}$	$k_{3,2}$
MEKF	1.0422	8.9311e-1	-6.3776e-1	1.0422	9.6764e-1	-4.8677e-1	1.5500	1.4208
Luenberger	-2.3329e-2	-1.2611e-2	-2.9012e-1	8.5789e-4	2.4019e-2	1.3981e-1	1.1191e-2	2.2337e-3

Table 5.7: New parameters values, the parameters refer to the \mathbf{K}_w matrix in the MEKF case, and to the \mathbf{K} matrix in the Luenberger case

Filter Model	$k_{3,3}$	$k_{4,1}$	$k_{4,2}$	$k_{4,3}$	$k_{5,1}$	$k_{5,2}$	$k_{5,3}$
MEKF	9.2155e-1	1.0500	9.0207e-1	1.9901	1.0579	9.0774	2.7633
Luenberger	7.8488e-2	2.4297e-3	4.6670e-4	1.6708e-2	2.9329e-2	4.5330e-3	3.4995e-1

Table 5.8: New parameters values, the parameters refer to the \mathbf{K}_w matrix in the MEKF case, and to the \mathbf{K} matrix in the Luenberger case

5.3.3. Comparison with dataset

The best result obtained Cognitive model, the 2nd order polynomial kinematic model and the MEKF as filter, will be evaluated with respect to the dataset using the one sample *Kolmogorov Smirnov* (KS) test. This is a non-parametric goodness of fit test, where the null hypothesis is that the sampled dataset's cumulative distribution function is equal to the one of the model's distribution, with a significance level α .

The same considerations will be done as for the tests done in ch. 2.4 since the data is bivariate, so each distribution will be projected in 10 directions and tested, moreover the Bonferroni correction will be applied.

The used significance level is $\alpha = 0.01$. Two score definitions will be used:

- *Failed Test Score* (FTS): this score simply counts how many failed tests there are given each combination of MOI and visibility time. The best score is 0 and the worst is 10. This test is not capable of giving an idea about how much the sample has failed, so there is no way to distinguish a model that failed in many directions but slightly from one that failed the same number of tests but by much.
- *Test Statistical Score* (TSS): this score takes the statistic values of the KS test, where each one is the maximum absolute vertical displacement between the experimental and the model's cumulative distribution. This score still has values that range between 0 (best) and 10 (worst), but it can assume any real value in between as well, since every KS test is a real value between 0 and 1. This is supposed to make the test less harsh and since now tests that slightly fail will score less than ones that failed completely.

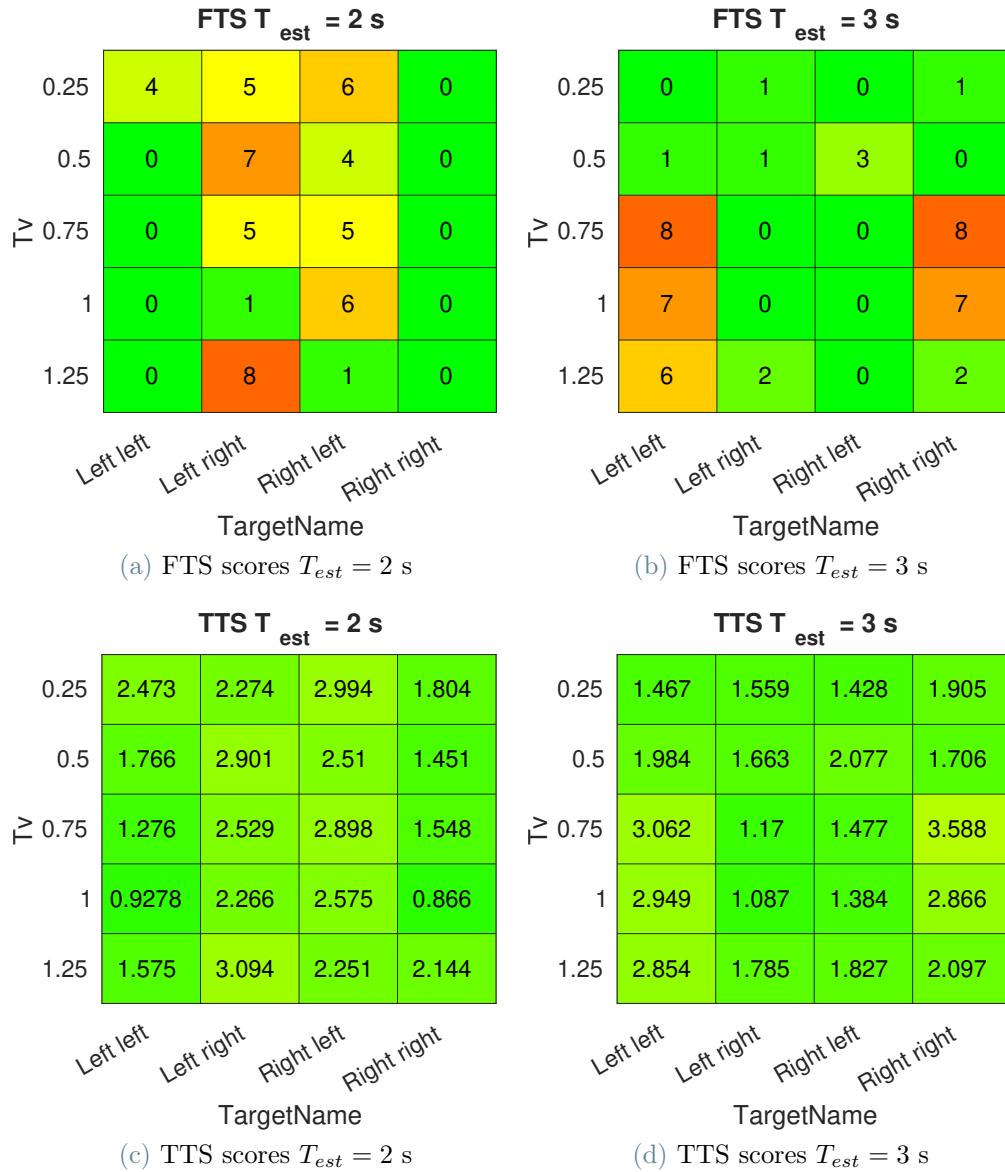


Figure 5.12: Testing of the Cognitive model with the best set of kinematic models and filter.

The results in fig. 5.12 show that while the TTS values are generally low, some FTS scores are not. This is especially visible in:

- Left right and Right left MOI when $T_{est} = 2$ s, where these results are to be expected since as seen in ch. 2.4 these MOI scored the lowest in the normality tests.
- Left left and Right right MOI when $T_{est} = 3$ s. This is a surprising thing since the dataset passed well the normality tests. Consequently, these fails must be attributed to the model.

If the graphs in fig. 5.9 and fig. 5.10 are analyzed, the trajectories that failed the test are also the ones that have the furthest means with respect to the dataset's, this consequently affect the corresponding first standard deviation level lines, that can not adapt well to the experimental ones.

In fig. 5.9 and fig. 5.10 the results obtained by the MEKF are shown, where the experimental distributions are confronted with the model's ones. Qualitatively speaking it can be stated that the model performed well when it is capable of reaching the experimental mean, since it can adapt better its covariance to mimic the sampled one. It could also be stated that in the trajectories where the model performed poorly, the distributions are not too wrong with respect to the experimental ones, and this is also confirmed by the TTS scores not being too high.

In conclusion, it can be stated that a Cognitive model capable of describing well enough the Cognitive process has been developed, where most of the trajectories sampled in the dataset are described well, as it can be concluded from the overall results in fig. 5.12. This has been achieved by the introduction a kinematic model capable of describing the way an observer perceived the motion of a pedestrian, where this means a model that accelerates its motion over time, and this has been achieved with the 2nd order polynomial kinematic model. Moreover, a suboptimal Bayesian filter has been introduced as Cognitive model, where the suboptimality has been shown to describe better the Cognitive process. This has been achieved by the use of the MEKF, a filter that takes the EKF's gain computation, than rescales the values to make the filter suboptimal.

6 | Experiment Design

6.1. Introduction

After the modeling work done in [18] a new dataset was required in order to develop the model even further and test the model obtained so far. As for the previous experiment, the dataset was collected in a virtual environment, where the candidates were placed on the sidewalk of an intersection. The task remained the same, to point a target moving in the environment until it disappeared, then keep on pointing where it is thought the cyclist is, then both the real trajectory and the perceived one were collected. This time though the targets were all cyclists. In the next chapters a more in depth description of the experiment will be given, starting from the Hardware and Software used, then the experiment design, the experiment conduction, a comparison will be done with the previous experiment, and finally the results will be analyzed.

6.2. Equipment and Software

The experiment was conducted using a *Virtual Reality Headset* (VR Headset), the *Meta Quest 2*, with its controllers. The Headset was connected to a computer that ran the experiment through the *Unity* Game Engine.

6.3. Experiment Design

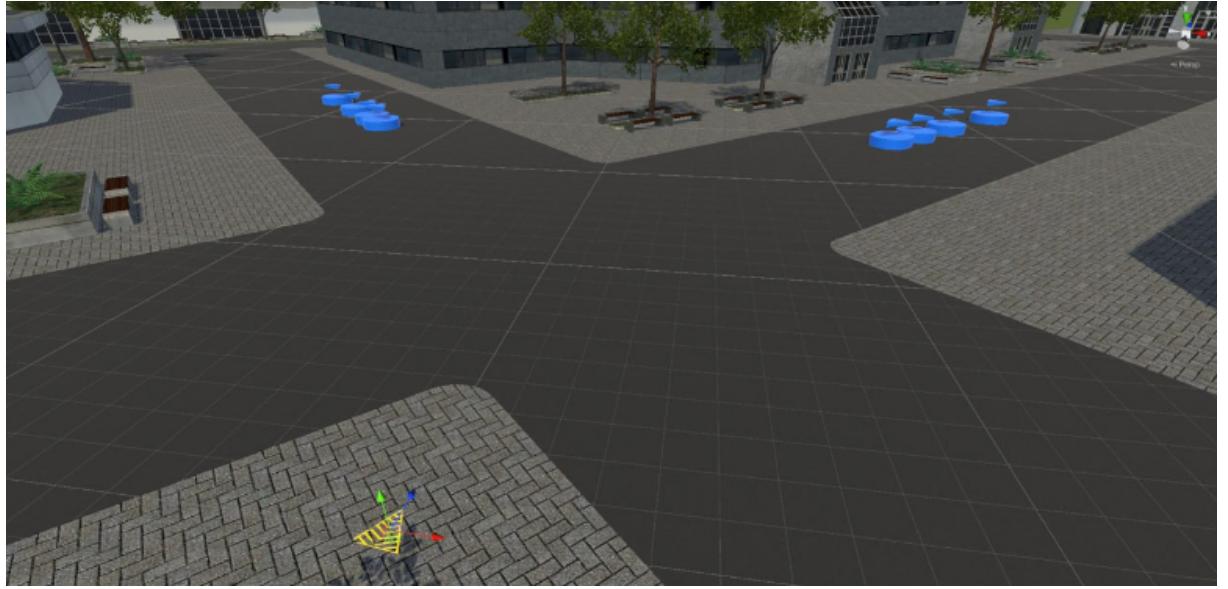
As already stated in ch. 6.1, the candidates were placed in an urban environment (see fig. 6.1a) where they had to point a target, that in this experiment could only be a cyclist, moving in the environment until it disappeared, then keep on pointing where they think it is. As for the old experiment (see ch. 2.4) Cyclists appeared one at a time, preceded by a stereo audio cue that allowed understanding where the cyclist was coming from with respect to the current head position. At the end of each trajectory another audio cue, with a different tone, was played to indicate the end of it. Both the trajectory and the one pointed by the candidates were sampled with a period of 0.25 s.

The candidates were placed on a sidewalk of an intersection in an urban environment, heading toward the two streets where the spawning areas were placed. The origin of the reference frame was placed where the candidates were standing, and it had the x and z axes parallel to the ground, and the y axis pointing upward, perpendicular to the ground, see fig. 6.1a. The candidates point of view was positioned at 1.7m of height (see fig. 6.1b).

As for the old experiment, trajectory was composed by two parts: a preview straight trajectory, then a MOI, that could be either a curve (either left or right) or straight. The preview lasted $T_{prev} = 2$ s, while the estimate could only last $T_{est} = 3$ s. The bicycle disappeared during the estimate time, and this happened after a visibility time T_v (defined as in ch. 2.4), that could last 0.25 s, 0.5 s, 0.75 s, 1 s or 1.25 s. Between each trajectory there was, a second of pause, where the candidate could prepare for the next trajectory. All the trajectories were picked from [5]. This is a dataset comprised of sampled trajectories of pedestrians, cyclists and cars, collected in different intersections. For our purpose all the trajectories were picked in the same intersection fig. 6.2, which was picked due to its conformation, making it similar to how the virtual intersection was designed, and consequently making it easier for the trajectories to be compatible with the virtual intersection modelled.

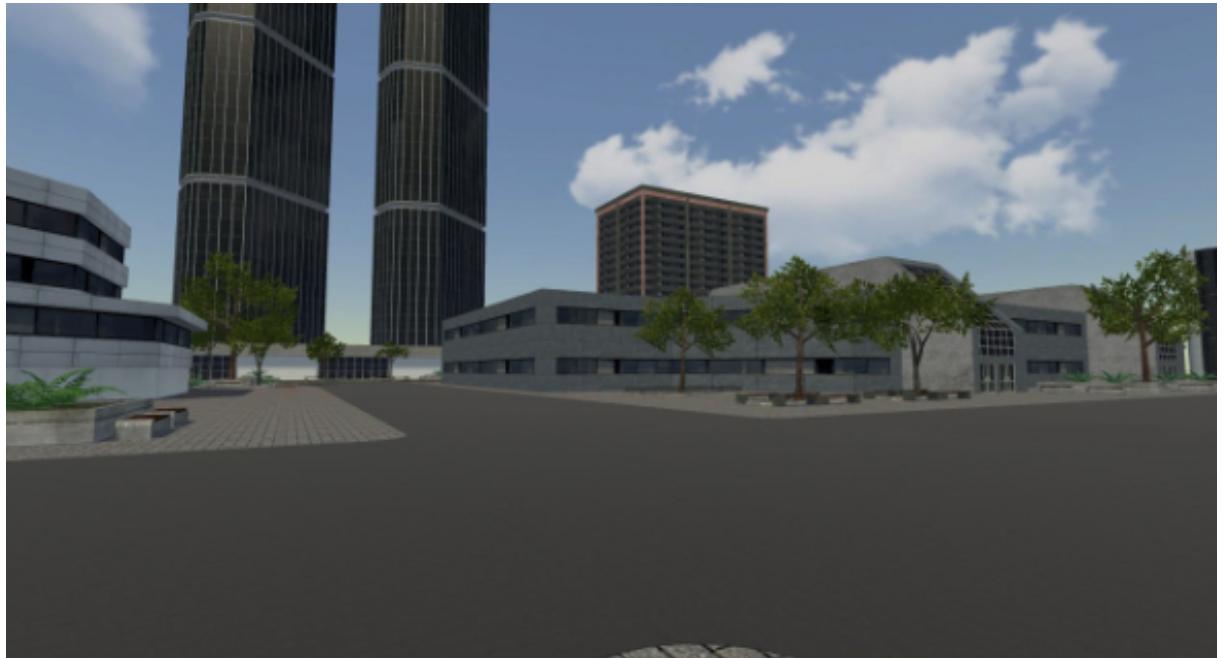
In total 6 trajectories were picked, 4 curving and 2 straight (visible in fig. 6.3). Since the dataset's trajectories are sampled at a frequency of 25 fps, all of them were interpolated to 60 fps, using *interp1* on *MATLAB*, so to make the animations more fluid in the experiment. Even though the dataset sampled many kinematic quantities, just the positions were used since the other quantities were not as accurate, for example the heading was computed afterwards since the sampled ones were not always aligned with the direction the pedestrian was moving toward.

Starting from the positions, the velocities at each instant were computed, (6.1) and (6.2), then the heading, using the velocity components along the x and z axes (6.4). To make the cyclists' animations more natural the roll angle was added as well to its animation (6.6).



(a) Image of the virtual environment where the experiment was conducted. The candidate is placed where there is a yellow marking on the ground. The blue areas are the spawning points of the trajectories, where each bicycle spawned with a heading angle of 235° if spawned in the left street, and 315° if spawned in the right street.

The Global Reference System is indicated by the red, green and blue arrows. The red arrow is the x axis, the blue one is the z axis and the green one is the y axis. The origin of the reference system is where the candidate is standing.



(b) Point of View of the candidates in the virtual environment.

Figure 6.1: New Perception experiment scene.



Figure 6.2: Intersection where the trajectories of the experiment's dataset were picked.
Image taken from [5].

$$v_x(k) = \frac{x(k) - x(k-1)}{T_s} \quad (6.1)$$

$$v_z(k) = \frac{z(k) - z(k-1)}{T_s} \quad (6.2)$$

$$v(k) = \sqrt{v_x(k)^2 + v_z(k)^2} \quad (6.3)$$

$$\theta(k) = \arctan\left(\frac{v_z(k)}{v_x(k)}\right) \quad (6.4)$$

$$\dot{\theta}(k) = \frac{\theta(k) - \theta(k-1)}{T_s} \quad (6.5)$$

$$\rho(k) = \arctan\left(\frac{v(k)^2}{g}\right) \quad (6.6)$$

$$g = 9.81 \frac{m}{s^2} \quad (6.7)$$

$$T_s = 0.01s \quad (6.8)$$

As for the old experiment each trajectory had assigned a spawn area with a uniform probability of radius 1 m, so to make the trajectories less recognizable, and since each trajectory had a different speed profile, each spawn was placed with a different distance from the intersection so to make the bicycle curve about the beginning of the intersection, and avoid them going on the sidewalk.

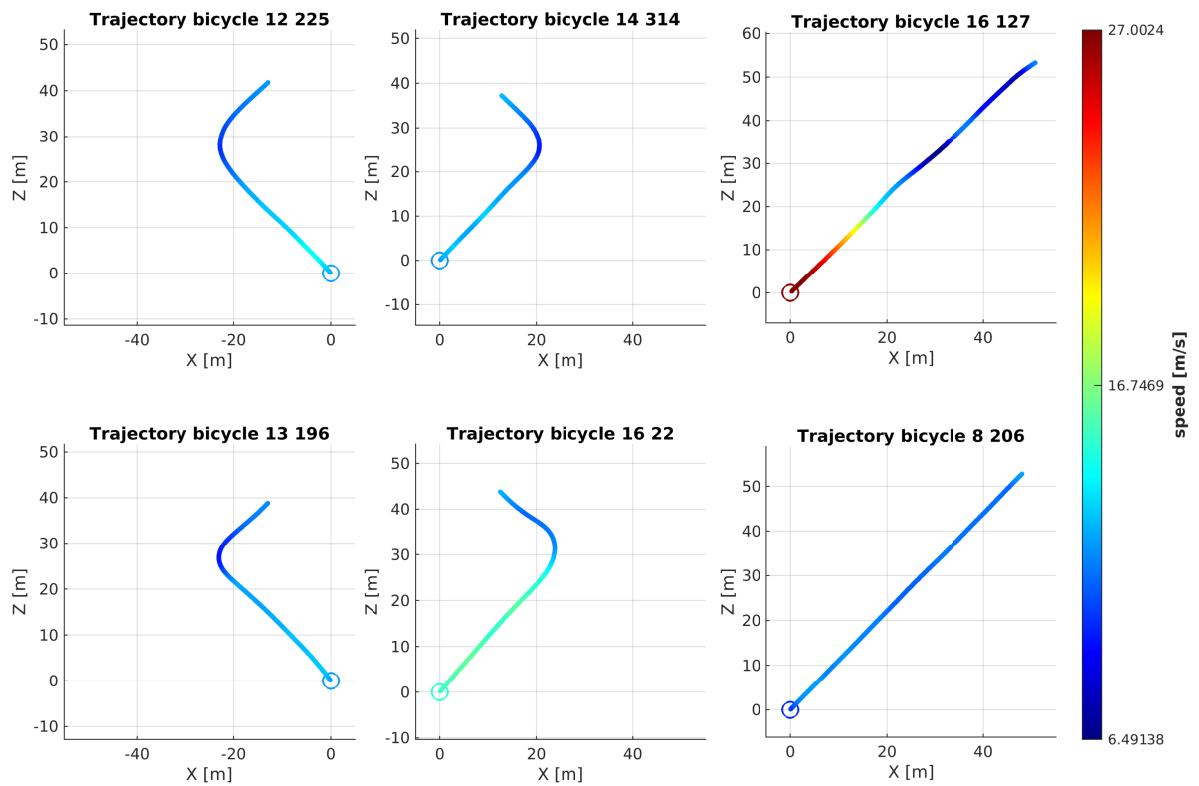


Figure 6.3: Plot of the 6 trajectories taken from [5] and used for the new Cognitive experiment. The starting point of each trajectory is circled.

These trajectories are to be translated and rotated accordingly on the spawn areas of the virtual environment, see fig. 6.4.

Trajectory Name			
Sampled Trajectory	Spawn Direction	Rotation Direction	Spawn Coordinates (X-Z Plane)
12 225	Left	right	(-15.6 m , 28.61 m)
12 225	Right	right	(15.8 m , 28.96 m)
14 314	Right	left	(14.27 m , 27.53 m)
14 314	Left	left	(-14.41 m , 27.54 m)
13 196	Left	right	(14.09 m , 27.32 m)
13 196	Right	right	(-14.34 m , 27.31 m)
16 22	Left	left	(-18.03 m , 31.15 m)
16 22	Right	left	(18.05 m , 31.15 m)
8 206	Right	straight	(14.14 m , 27.37 m)
8 206	Left	straight	(-1.58 m , 27.65 m)
16 127	Right	straight	(21.7 m , 34.69 m)
16 127	Left	straight	(-21.51 m , 34.6 m)

Table 6.1: List of the 12 trajectories of the experiment. The names of each is composed by: the number identifying the trajectory in the dataset, the position of the spawn with respect to the candidate, the rotation direction with respect to the cyclist's point of view. In the last column there is the spawn coordinates of each trajectory.

Each trajectory could spawn in both streets, as seen in tab. 6.1, resulting in 12 different combinations of trajectory and spawn areas.

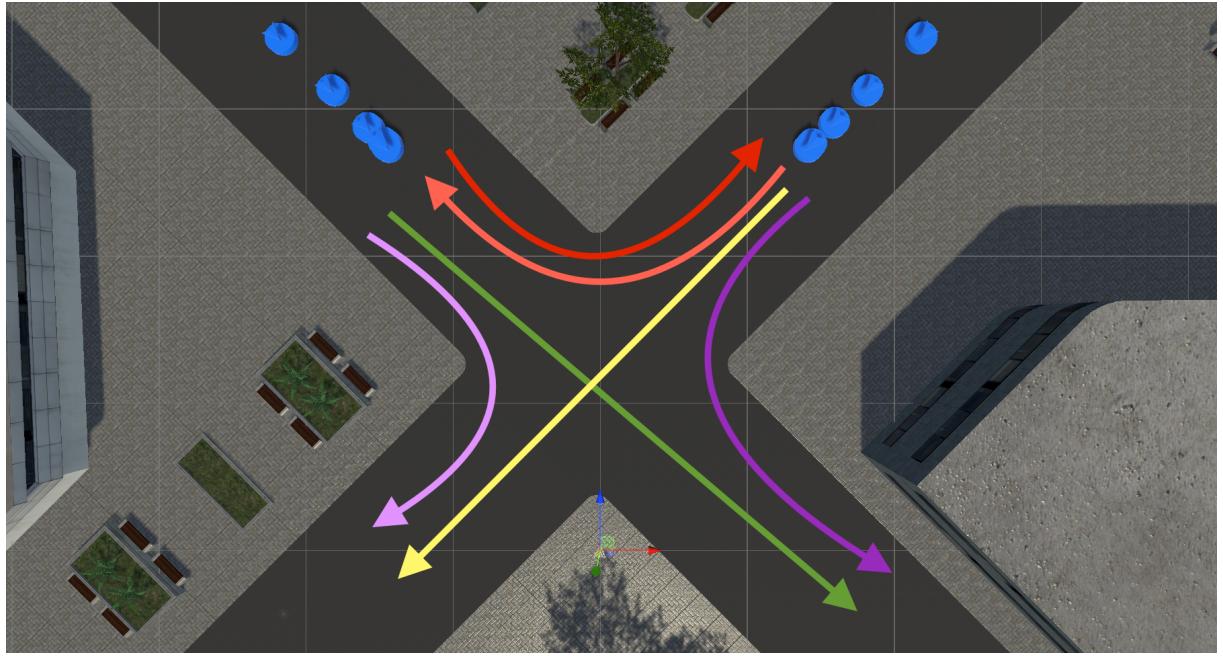


Figure 6.4: Representation of the spawn areas, the blue circles, and a qualitative representation of the trajectories of the dataset.

More specifically: the Red arrow represents 14 314 Left left and 16 22 Left left, orange arrow represent 12 225 Right right and 13 196 Right right, the violet arrow represents 14 314 Right left and 16 22 Right left, the rose arrow represents 12 225 Left right and 13 196 Left right, the yellow arrow represents 8 206 Right straight and 16 127 Right straight, and the green arrow represents 8 206 Left straight and 16 127 Left straight.

The blue areas are the spawning areas of the trajectories, listed in tab. 6.1.

To recreate the trajectories in the virtual environment each one had assigned a matrix containing the velocity and angular velocity values, v_x , v_z and $\dot{\theta}$, then in Unity a fixed frame rate of 60 fps was imposed, finally at each frame the values were read one at a time.

In each of the 4 curving trajectories there had to be defined the beginning of the MOI, or when the estimate time starts. To do so it was decided to consider the radius of curvature of the trajectory, more specifically when the curvature was consistently greater than 10 m, then the MOI starts (see fig. 6.5).

The two straight trajectories instead were picked based on their velocity profile, indeed 8 206 is the slowest trajectory in the dataset, while 16 127 is the fastest. Another difference between the two trajectories is that the fast one has a sudden deceleration that occurred about the beginning of T_p . This choice was done so to see if velocity and acceleration could influence the perception of the candidates.

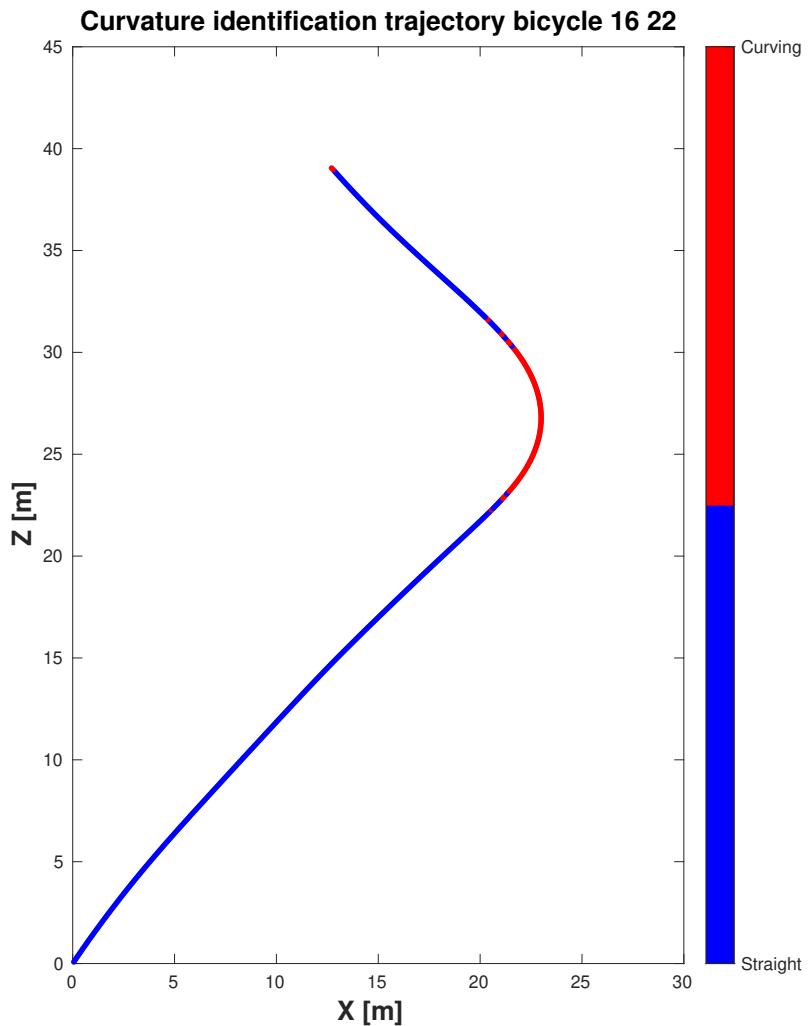


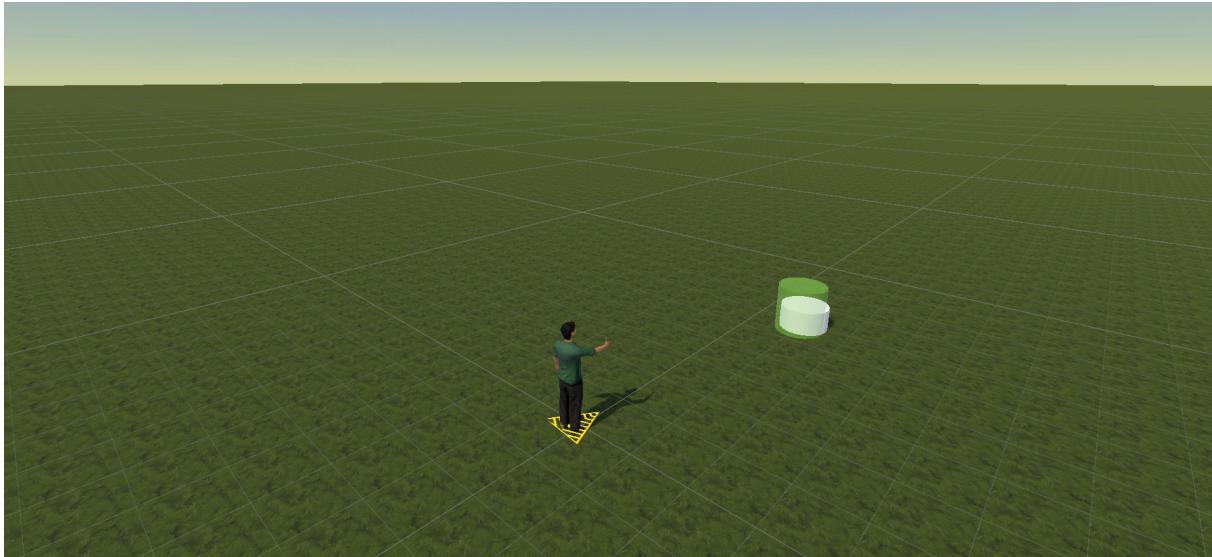
Figure 6.5: The plot shows in red the part of the trajectory that are considered as curving, so curvature greater than 10 m, and not curving in blue. The MOI starts when the trajectory becomes consistently red, meaning that when blue and red are alternating the MOI has not started yet.

6.4. Experiment Conduction

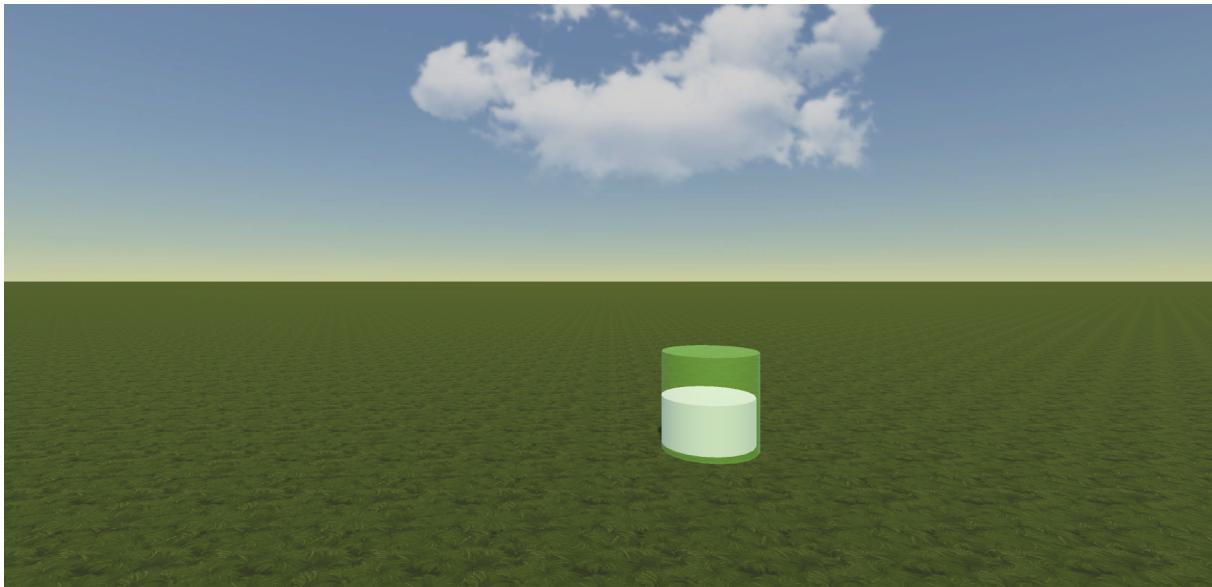
In total 54 people participated in the experiment, where the first 13 participated in a pilot experiment, where the environment was tested and the difficulties of the participants in pointing where noticed. The experiment lasted around 15 minutes for each candidate, and it was divided in tree parts:

1. a training meant to get the participants used to the pointing system, this consisted in following with the pointer a cylinder that moved in a circular trajectory. This part of the experiment was set in an empty field, as seen in fig. 6.6.
2. An extract of the experiment where both T_{prev} and T_{est} where set to 1.5 s and each trajectory was played once, so to get the participants acquainted to the task
3. the experiment itself.

The training was added after a pilot of 13 participants, since it was noticed that the participants had some difficulties at pointing when the bicycle was going far from them.



(a) Image of the player performing the training procedure. The green cylinder was spawned where the laser intersected the ground so to help to understand better where the laser was pointing at.



(b) Point of View of the candidates during the training procedure.

Figure 6.6: Training scene

6.5. Comparison with the Previous Experiment

This experiment was designed starting from the previous one so to improve some aspects and consequently get a more meaningful dataset.

In the previous experiment only the last position of the pointer was recorded, at the end of the trajectory, while now the pointer's position was sampled each 0.25 s, so to avoid having to repeat each combination of trajectory and visibility time to get an estimate with both

$T_{est} = 2s$ and $T_{est} = 3s$. The sampling of all the trajectory is also needed to understand better the perceptive ability with a moving target, indeed the models developed so far are based on only the precision of the distribution of the final position's guess, without actually knowing how precise the candidates are when the target is visible (in ch. 5.3.2 the results obtained by the Luenberger observer are an example on why this is important). Another important improvement is the introduction of naturalistic trajectories. This is important since it removes any doubt about how they could be wrongly perceived due to them being artificial, moreover this introduces a wide range of velocities in them, adding information about how velocity and acceleration could influence perception, something that is completely missing in the old dataset.

As already stated in ch. 2.4 the old dataset was not reliable in some trajectories due to the presence of road signs, where in the case of the pedestrian the cross roads influenced the Cognitive process of the participants. Since the Cognitive model developed so far only the knowledge about the target's movement is considered, this is a problem. In the new environment the street does not give any hint about where the cyclists should ride while traversing it.

The candidates were lowered to a height of 1.7 m, so to make the point of view of the candidates closer to an average person [35].

The last improvement is the enrichment of the surrounding environment. The buildings around the street indeed are meant to improve the spacial awareness of the candidate by giving them a sense of scale. This was a needed improvement also because lowering the candidates could have made pointing harder.

6.6. Experiment Observations

Some preliminary analysis was done on the results, so to spot possible problems with the experiment design.

6.6.1. Discarded Data

Even though the data was sampled each 0.25 s, not all the data is useful, since when the bicycle spawned the candidates needed some time to point at it, consequently the first seconds must be cut from the dataset. This could have been avoided by either making the candidates manually start each trajectory, or use a longer estimate time. The drawback is that both methods increase the duration of the experiment.

In the trajectories where the bicycles went further with respect to the candidates there are many points in the dataset that are way off the target's position, more specifically

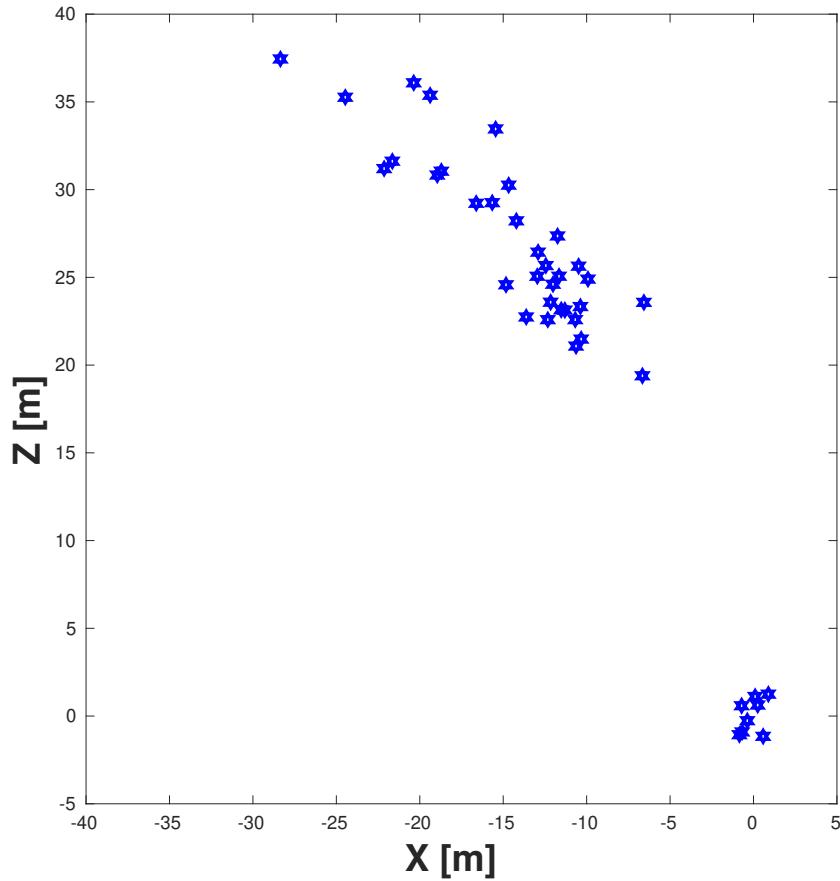


Figure 6.7: Data collected at the end of trajectory 12 225 Right right with $T_v = 0.25$ s. The data collected around (0,0) is caused by the pointing problem that affected the experiment.

these points are collected around the origin of the coordinate system. This is caused by a technical problem in the experiment, indeed after some inspection it seemed that the laser could be pointed at most 50 m far from the candidate, any further and the registered data is not reliable.

This makes the data collected very unreliable, so to solve this problem the dataset will be considered reliable up to $T_{est} = 2.5$ s, a value chosen after visually inspecting it.

6.6.2. Pointing difficulties

Some candidates had some difficulties when pointing far objects due to the fact that a far target is harder to follow with a pointer. Indeed, when pointing far, small movements of the hand cause big movements of the pointer on the ground. This might have influenced

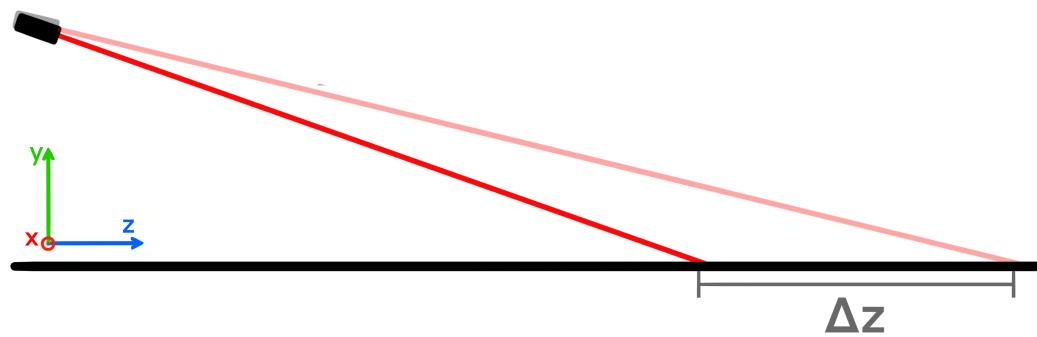
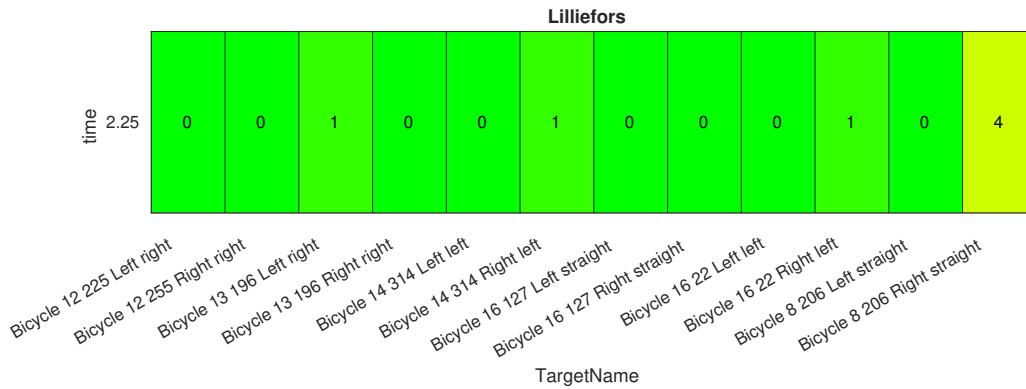
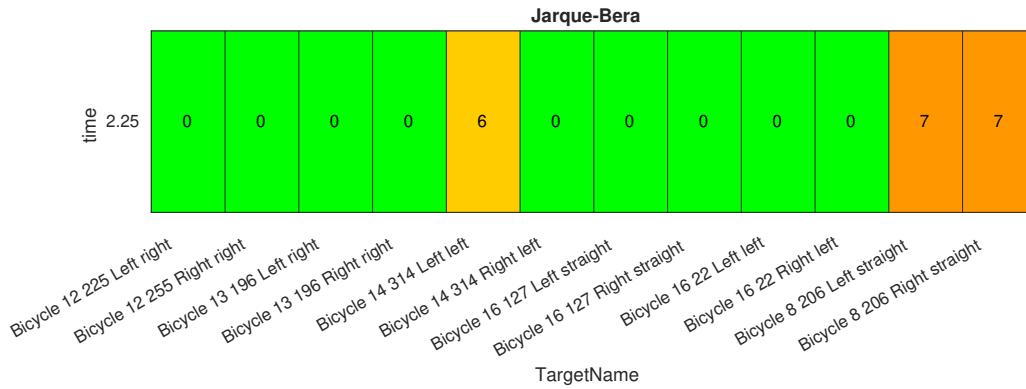
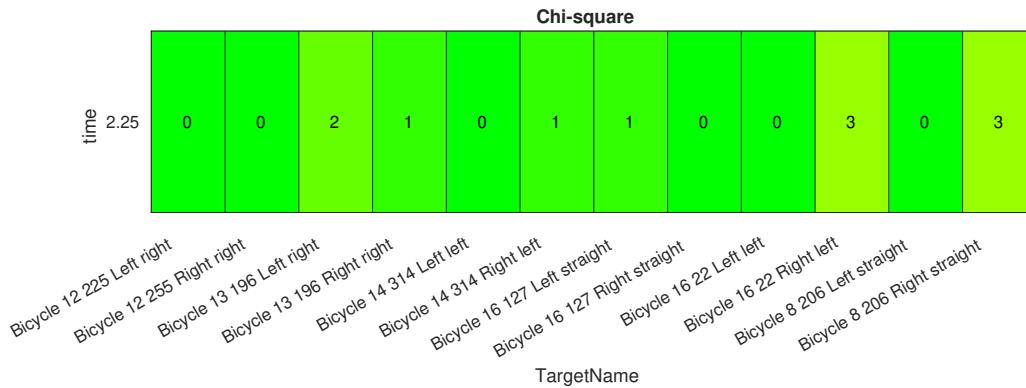


Figure 6.8: This image shows how a small movement of the laser generates a big difference in the pointed position.

the dataset in the trajectories that ended with the bicycle far from the target.

6.6.3. Normality Tests

As for ch. 2.4 some normality tests were performed so to check if the data is Gaussina. Defining t as the time from the trajectory beginning, the tests will be done when $t = 2.25$ s, the last instant where the trajectory is visible for all T_v , and when $t = 4.5$ s.

(a) Lilliefors test performed on the new dataset when $t=2.25$ s(b) Jaque-Bera test performed on the new dataset when $t=2.25$ ss(c) Chi-square test performed on the new dataset when $t=2.25$ sFigure 6.9: Normality tests when $t = 2.25$ s.

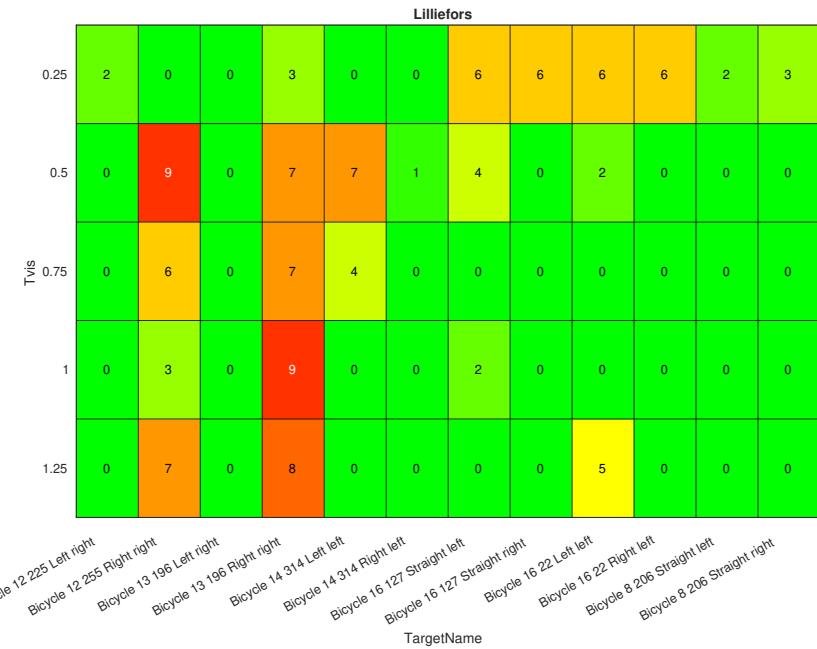


Figure 6.10: Lilliefors test performed on the new dataset when $t=4.25$ s.

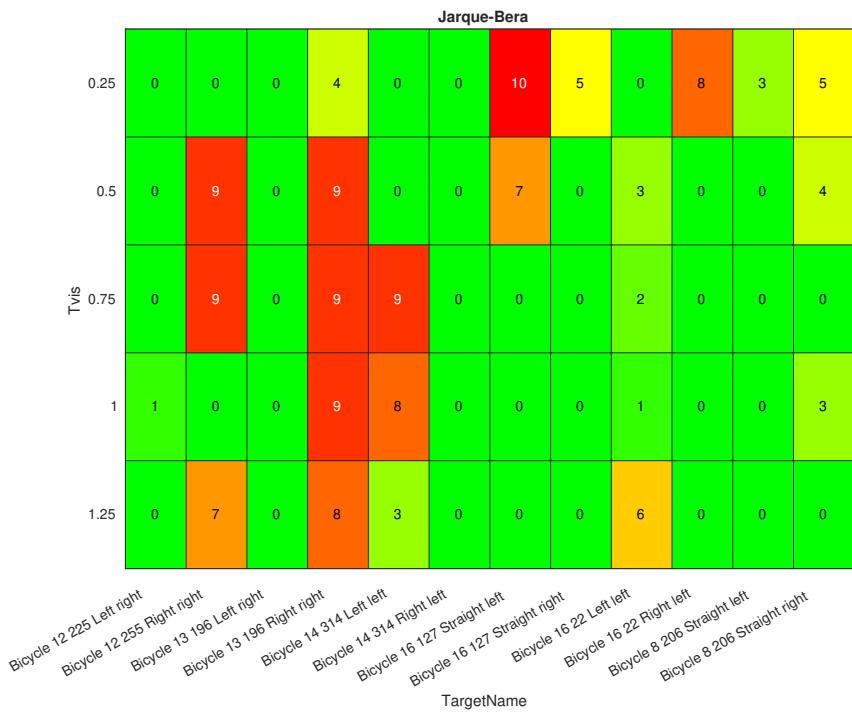


Figure 6.11: Jaque-Bera test performed on the new dataset when $t=4.25$ s.

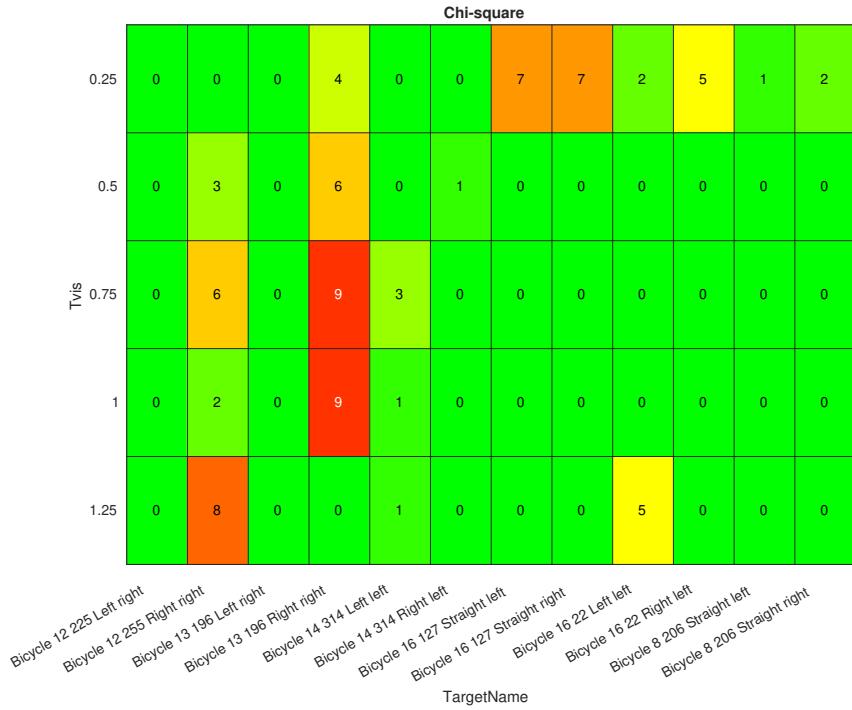


Figure 6.12: Chi-square test performed on the new dataset when $t=4.25$ s.

The results (from fig. 6.9 to fig. 6.12) on the raw data show that the data is overall Gaussian, with some exceptions, namely: 12 225 Right right, 13 196 Right right and 14 314 Left left when $t=4.5$ s. In all of these trajectories the cyclists got further from the candidates, bringing out the already discussed difficulties in pointing.

7 | Conclusions

A new Cognitive model has been developed, capable of simulating the real Cognitive process with a good degree of fidelity. This was developed to be integrated in an ADAS system so to give it a better insight about the driver's understanding of its surrounding and consequently, allow the ADAS system to understand if a safety intervention is necessary. This should both make the system safer and more comfortable for the driver, since unnecessary intervention can be avoided.

The model has been developed starting from the structure of a generic observer, more specifically starting from the structure of an EKF. A kinematic model describing how an observer perceives a pedestrian's movement has been developed, then the gain of the filter has been modified so to remove its optimality, making the final model a generic nonoptimal observer. The results show that the generic observers gave a better performance with respect to the EKF.

Possible next developments for the model are:

- In the CXP the pedestrians were all walking at the same speed, and all performed a MOI with the same angular velocity. This lead to a dataset with very little information about the kinematic model of the pedestrian. A new dataset is then needed so a more accurate kinematic model can be developed.
- In the models obtained in this work, the parameters of the Perceptive model regarding the heading angle of the pedestrian, r_1 , r_2 and r_3 , where estimated using the CXP dataset with the Cognitive model parameters. This has lead to values that favored the performance of the Cognitive model with the given dataset, but it does not guarantee the Accuracy of the Perceptive model with respect to the Perceptive process. A model catered toward a better description of the heading angle in the Perceptive model is needed. Since the current PXP does not collect any data about the perception of the heading angle, the development of a better Perceptive model also requires a new dataset.

A new dataset for the Cognitive model of the bicycle was created to develop the already existing model even further. The experiment was conducted so to improve some

aspects about the previous one, such as the introduction of naturalistic trajectories, the improvement of the environment so to remove any element that could have influenced the candidates in their Cognitive process, the sampling of the whole trajectory rather than just the last position and the lowering of the point of view of the candidate to a height closer to the average Italian height.

The introduction of these design changes has also introduced some problems that need to be addressed, such as the pointing problems given by the limited range the of laser and the possible pointing problems given by the high distance of the bicycle from the candidates.

In the future a dataset for the Cognitive process of the pedestrian that introduce the improvements of this dataset must be created, so to develop the Cognitive model of the pedestrian even further, moreover some changes in the experiment deign must be done to solve the encountered problems.

Bibliography

- [1] C. Ahlstrom, K. Kircher, and A. Kircher. A gaze-based driver distraction warning system and its effect on visual behavior. *IEEE Transactions on Intelligent Transportation Systems*, 14(2):965–973, 2013. doi: 10.1109/TITS.2013.2247759.
- [2] J. G. V. Ana“is Halin and M. V. Droogenbroeck. Survey and synthesis of state of the art in driver monitoring. *Sensors*, 2021.
- [3] B. Bittanti. *Model Identification and Data Analysis*, chapter 9 - Kalman Filtering and Prediction, pages 209–279. Wiley, 2019.
- [4] J. M. Bland and D. G. Altman. Multiple significance tests: The bonferroni method. *BMJ*, 310(6973):170, 1995.
- [5] J. Bock, R. Krajewski, T. Moers, S. Runde, L. Vater, and L. Eckstein. The ind dataset: A drone dataset of naturalistic road user trajectories at german intersections. In *2020 IEEE Intelligent Vehicles Symposium (IV)*, pages 1929–1934, 2020. doi: 10.1109/IV47402.2020.9304839.
- [6] Caradas. Active adas systems and driver safety features. <https://caradas.com/active-adas-systems/>, 2023.
- [7] Caradas. Passive adas systems and safety features. <https://caradas.com/passive-adas-systems/>, 2023.
- [8] N. D. S. L. C. Ferrise. Localization and prediction of visual targets’ position in immersive virtual reality. *PRESENCE*, 2023.
- [9] V. Goussev. Does the brain implement the kalman filter? *Behavioral and Brain Sciences*, 27:404–405, 2004.
- [10] D. Helbing, I. Farkas, and T. Vicsek. Simulating dynamical features of escape panic. *Nature*, 407(6803):487–490, 2000. doi: 10.1038/35035023.
- [11] N. L. P. F. Ivo Batkovic, Mario Zanon. A computationally efficient model for pedestrian motion prediction. *European Control Conference (ECC)*, 201.

- [12] C. M. Jarque and A. K. Bera. A test for normality of observations and regression residuals. *International Statistical Review / Revue Internationale de Statistique*, 55(2):163–172, 1987.
- [13] V. Karasev, A. Ayvacı, B. Heisele, and S. Soatto. Intent-aware long-term prediction of pedestrian motion. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2543–2549, 2016. doi: 10.1109/ICRA.2016.7487409.
- [14] K. M. Kitani, B. D. Ziebart, J. A. Bagnell, and M. Hebert. Activity forecasting. In A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, and C. Schmid, editors, *Computer Vision – ECCV 2012*, pages 201–214, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg. ISBN 978-3-642-33765-9.
- [15] U. D. Lab. The future of urban design: Trends and challenges. <https://urbandesignlab.in/the-future-of-urban-design-trends-and-challenges/#:~:text=Public%20space%20design%3A,lanes%20and%20pedestrian%2Dfriendly%20streets>, 2023.
- [16] H. W. Lilliefors. On the kolmogorov-smirnov test for normality with mean and variance unknown. *Journal of the American Statistical Association*, 62(319):399–402, 1967.
- [17] A. C. Ludovico Rozza. Elaborazione di un modello cognitivo per il conducente. Master’s thesis, Politecnico di Milano, 2021.
- [18] A. C. Matteo Depaola. A cognitive model of the human driver. Master’s thesis, Politecnico di Milano, 2022.
- [19] E. Michelaraki, C. Katrakazas, S. Kaiser, T. Brijs, and G. Yannis. Real-time monitoring of driver distraction: State-of-the-art and future insights. *Accident Analysis and Prevention*, 192:107241, 2023. ISSN 0001-4575. doi: \url{https://doi.org/10.1016/j.aap.2023.107241}. URL <https://www.sciencedirect.com/science/article/pii/S0001457523002889>.
- [20] A. P. A. Mohinder S. Grewal. *Kalman Filtering: Theory and Practice Using MATLAB*. Wiley-Interscience, 2001.
- [21] Nationide. Reducing collisions with adas. <https://www.mylosscontrolservices.com/learning-center/articles/reducing-collisions-with-adas>.
- [22] P. Newman. Driverless vehicles and pedestrians don’t mix. so how do we re-arrange our cities? <https://theconversation.com/>

- driverless-vehicles-and-pedestrians-dont-mix-so-how-do-we-re-arrange-our-cities
2019.
- [23] K. P. Nick Oliver and T. Calvard. To make self-driving cars safe, we also need better roads and infrastructure. <https://hbr.org/2018/08/to-make-self-driving-cars-safe-we-also-need-better-roads-and-infrastructure>, 2018.
 - [24] S. Pellegrini, A. Ess, K. Schindler, and L. van Gool. You'll never walk alone: Modeling social behavior for multi-target tracking. In *2009 IEEE 12th International Conference on Computer Vision*, pages 261–268, 2009. doi: 10.1109/ICCV.2009.5459260.
 - [25] S. Saha, D. De. Practical self-driving cars: Survey of the state-of-the-art. *Preprints*, 2022(2022020123), 2022. doi: 10.20944/preprints202202.0123.v1. URL <https://doi.org/10.20944/preprints202202.0123.v1>.
 - [26] R. Scattolini. *Advanced and Multivariable Control*, chapter 6. Pitagoga edirice Bologna, 2014.
 - [27] N. Schneider and D. M. Gavrila. Pedestrian path prediction with recursive bayesian filters: A comparative study. In J. Weickert, M. Hein, and B. Schiele, editors, *Pattern Recognition*, pages 174–183, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg. ISBN 978-3-642-40602-7.
 - [28] C. Shalizi. Undergraduate advanced data analysis, multivariate distributions. <https://www.stat.cmu.edu/~cshalizi/uADA/19/>, 2012.
 - [29] S. Singh. Critical reasons for crashes investigated in the national motor vehicle crash causation survey. technical report. Technical report, National Highway Traffic Safety Administration, 2015.
 - [30] G. M. Smith. What is adas (advanced driver assistance systems)? <https://dewesoft.com/blog/what-is-adas>, 2023.
 - [31] M. A. Stephens and R. B. D'Agostino. *Goodness-of-Fit Techniques*. CRC Press, 1986.
 - [32] R. Steuteville. Ten features of walkable communities. <https://www.cnu.org/publicsquare/2023/12/22/ten-features-walkable-communities>, 2023.
 - [33] A. Tsakpinis. State of the art, issues and trends of autonomous driving. Master's thesis, Technische Universität München, 2022.
 - [34] R. Weetman. Nice cities, liveable places...and how we might try to

create them. <https://robertweetman.wordpress.com/2019/03/19/i-want-my-street-to-be-like-this/>, 2019.

- [35] WorldData.info. Average height and weight by country. <https://www.worlddata.info/average-bodyheight.php>, 2020.
- [36] K. Yamaguchi, A. C. Berg, L. E. Ortiz, and T. L. Berg. Who are you with and where are you going? In *CVPR 2011*, pages 1345–1352, 2011. doi: 10.1109/CVPR.2011.5995468.
- [37] K. Yang, C. Al Haddad, R. Alam, T. Brijs, and C. Antoniou. Adaptive intervention algorithms for advanced driver assistance systems. *Safety*, 10:10, 2024. doi: 10.3390/safety10010010. URL <https://doi.org/10.3390/safety10010010>.

List of Figures

1.1	Qualitative graph describing the long-tail problem, where it is shown how given a dataset to train a model, the most common cases will be taken care from a small batch, and consequently this will also drastically reduce the error of the trained model. The many uncommon, or niche, cases though require a much larger dataset since they are hard to spot, moreover there is a diminishing return in the error reduction. This problem is not exclusive to Machine Learning and is known with other names such as Pareto Principle, or twenty-eighty rule.	2
1.2	Image taken in the city of Utrecht, Netherlands. The Netherlands is a country renowned for its bike friendly and walkable cities, making it a prime example of how urbanization is evolving. Looking at this image it is visible how the road is designed to be small, irregular and visibly residential, thanks to the use of brick instead of asphalt. This has the double effect of slowing down drivers, while making pedestrian feel, and be, safer on the street. Image taken from [34]	3
1.3	Example of sensors equipped on a car and their use. Image taken from [30].	5
2.1	The three reference frames used in the Perceptive Model.	13
2.2	Trajectory illustration, where for each part the corresponding duration is indicated with the same color.	18
2.3	CXP setting.	19

2.4	Top view of the Perception experiment scene. The pedestrians spawning areas, the ones on the sidewalks, have as center (-14.11 m , 0 m , 16.42 m) and (9.04 m , 0 m , 16.1 m), while the cyclists' one, on the road, have as center (-0.35 m , 0 m , 24.9 m) and (-0.35 m , 0 m , 27.4 m). The spawning position determined the initial heading angle of the target, more specifically, pedestrians spawning from the left spawning area had an initial heading angle of 225°, while the ones spawning of the right of 315°, all the cyclists instead spawned with a heading of 270°. The MOI indicated in the figure are: 1. Big-Left 2. Big-Right 3. Small-Left 4. Small-Right 5. Left left 6. Left righth 7. Right left 8. Right right.	20
2.5	CXP dataset normality tests.	23
2.6	Areas of the first standard deviations of the CXP dataset.	24
2.7	Dataset distributions for $T_{est} = 2s$	25
2.8	Dataset distributions for $T_{est} = 3s$	26
5.1	Results obtained with the Unicycle Kinematic Model and EKF filter, $T_{est} = 2 s$	42
5.2	Results obtained with the Unicycle Kinematic Model and EKF filter, $T_{est} = 3 s$	43
5.3	Results obtained with the 2 nd Order Polynomial Kinematic Model and EKF filter, $T_{est} = 2 s$	44
5.4	Results obtained with the 2 nd Order Polynomial Kinematic Model and EKF filter, $T_{est} = 3 s$	45
5.5	Boxplot of the Mahalanobis distances between the means of the dataset and the model's distributions	46
5.6	Boxplot of the Mahalanobis distances between the means of the dataset and the model's distributions, where the dataset is divided based on the MOI, as indicated on the abscissa.	47
5.7	Results obtained with the 2 nd Order Kinematic Model and Luenberger Observer, $T_{est} = 2 s$	50
5.8	Results obtained with the 2 nd Order Kinematic Model and Luenberger Observer, $T_{est} = 3 s$	51
5.9	Results obtained with the 2 nd Order Polynomial Kinematic Model and MEKF filter, $T_{est} = 2 s$	52
5.10	Results obtained with the 2 nd Order Polynomial Kinematic Model and MEKF filter, $T_{est} = 3 s$	53

5.11	Distance between the perceived trajectory and the one computed by the Cognitive model in trajectory Right right $T_v = 1.25$ s, during visibility time Blue line: Cognitive model with 2 nd order polynomial and MEKF Orange line: Cognitive model with 2 nd order polynomial and Luenberger	54
5.12	Testing of the Cognitive model with the best set of kinematic models and filter.	56
6.1	New Perception experiment scene.	61
6.2	Intersection where the trajectories of the experiment's dataset were picked. Image taken from [5].	62
6.3	Plot of the 6 trajectories taken from [5] and used for the new Cognitive experiment. The starting point of each trajectory is circled. These trajectories are to be translated and rotated accordingly on the spawn areas of the virtual environment, see fig. 6.4.	64
6.4	Representation of the spawn areas, the blue circles, and a qualitative representation of the trajectories of the dataset. More specifically: the Red arrow represents 14 314 Left left and 16 22 Left left, orange arrow represent 12 225 Right right and 13 196 Right right, the violet arrow represents 14 314 Right left and 16 22 Right left, the rose arrow represents 12 225 Left right and 13 196 Left right, the yellow arrow represents 8 206 Right straight and 16 127 Right straight, and the green arrow represents 8 206 Left straight and 16 127 Left straight. The blue areas are the spawning areas of the trajectories, listed in tab. 6.1.	65
6.5	The plot shows in red the part of the trajectory that are considered as curving, so curvature greater than 10 m, and not curving in blue. The MOI starts when the trajectory becomes consistently red, meaning that when blue and red are alternating the MOI has not started yet.	67
6.6	Training scene	68
6.7	Data collected at the end of trajectory 12 225 Right right with $T_v = 0.25$ s. The data collected around (0,0) is caused by the pointing problem that affected the experiment.	70
6.8	This image shows how a small movement of the laser generates a big difference in the pointed position.	71
6.9	Normality tests when $t = 2.25$ s.	72
6.10	Lilliefors test performed on the new dataset when $t=4.25$ s.	73
6.11	Jaque-Bera test performed on the new dataset when $t=4.25$ s.	73
6.12	Chi-square test performed on the new dataset when $t=4.25$ s.	74

List of Tables

2.1	Parameters identified for the perceptive model. The values have been computed in [18], ch.7	14
5.1	Final Log Likelihood cost function values. The kinematic model used is indicated in the first row, the filter used is the EKF. The first row is the total cost, the second and third are the cost when $T_{est} = 2$ s and $T_{est} = 3$ s respectively.	41
5.2	Cognitive Model parameters, q_i are the diagonal values of the covariance matrix, d_i are the parameters introduced by the kinematic models	47
5.3	Perceptive Model parameters	48
5.4	Final Log Likelihood cost function values. The filter used is indicated in the first row, the kinematic model is the 2^{nd} order polynomial. The first row is the total cost, the second and third are the cost when $T_{est} = 2$ s and $T_{est} = 3$ s respectively.	49
5.5	Cognitive Model parameters, q_i are the diagonal values of the covariance matrix, d_i are the parameters of the 2^{nd}	54
5.6	Perceptive Model parameters	54
5.7	New parameters values, the parameters refer to the \mathbf{K}_w matrix in the MEKF case, and to the \mathbf{K} matrix in the Luenberger case	55
5.8	New parameters values, the parameters refer to the \mathbf{K}_w matrix in the MEKF case, and to the \mathbf{K} matrix in the Luenberger case	55
6.1	List of the 12 trajectories of the experiment. The names of each is composed by: the number identifying the trajectory in the dataset, the position of the spawn with respect to the candidate, the rotation direction with respect to the cyclist's point of view. In the last column there is the spawn coordinates of each trajectory.	63

Ringraziamenti

Desidero ringraziare il mio relatore, il Prof. Alessandro Colombo, per avermi seguito e supportato durante il lavoro di tesi, e per essere stato molto paziente con me durante tutto il percorso, soprattutto durante la stesura.

Ringrazio inoltre il dottorando Nicolò Dozio che mi ha affiancato nella progettazione e conduzione dell'esperimento, senza il quale non sarei riuscito a portare a termine il lavoro. Ringrazio la mia famiglia per avermi permesso di intraprendere questo percorso universitario e per avermi supportato in ogni momento, anche in quei momenti nei quali ero intrattabile a causa delle sessioni.

Ringrazio Giorgia per essermi stata vicino durante questo percorso, soprattutto quando mancava pochissimo alla consegna della tesi e, in piena crisi ansiogena, ha scritto al relatore da parte mia, risolto i problemi di compilazione e caricato i documenti sul portale appena in tempo (ha detto che le ha messo più ansia la consegna della mia tesi che della sua).

Ringrazio Antonio che per essere stato un amico molto generoso con me, permettendomi di condurre gli esperimenti usando il suo PC, senza mai chiedere nulla in cambio. Senza il suo aiuto non sarei riuscito a finire gli esperimenti in tempo. Ringrazio tutti i miei amici, quelli che ho avuto il piacere di conoscere in questi ultimi anni e quelli che mi sono vicini da praticamente una vita. In particolare ringrazio Ochoa, Kevin, Stefano, Riccardo, Emanuele e Matteo, ovvero i miei amici di meccanica. Abbiamo sofferto assieme durante la triennale, ma è grazie anche a voi se è stato possibile superare ogni ostacolo.

Ringrazio inoltre gli amici dell'Agorà, in particolare Lorenzo, Valerio, Martina, Fede Sbravati, Francesca, Bianca, Zazzi, Zahraa, Fede Cirelli, Sergio, Sok, il bot del Razzismo e infine il bot del Voltaggio (SI DICE TENSIONE!), per avermi accompagnato durante le lunghissime sessioni di esami.

Ringrazio infine i miei amici più recenti, quelli conosciuti in magistrale, ovvero Alessandro Cüopo, Davide, Giacomo Walker Ramses II Obi-Wan Kenobi Colombo, Francesco, Tancredi, Pex e la quota napoletana Edoardo e Raffaele. Abbiamo fatto molti corsi divertenti assieme (come quel fantastico corso sui PLC) e lavorato su tanti progetti ben riusciti (in particolare quello sul Maglev).