UNIVERSITY OF TARTU
Institute of Computer Science
Computer Science curriculum

Matis Ottan

Leading vehicle length estimation using pressure data for use in autonomous driving

Bachelor's Thesis (9 ECTS)

Supervisor: Naveed Muhammad

Tartu 2022

Leading vehicle length estimation using pressure data for use in autonomous driving

Abstract:

Overtaking vehicles is a risky manoeuvre for human drivers and an even more difficult challenge for autonomous cars. The algorithms for overtaking require extensive information about the surrounding environment including knowing the length of a leading vehicle. The usual sensing modalities used in autonomous vehicles (vision, radar, LiDAR) are not suitable for estimating that length. In literature, flow sensing has been shown to aid underwater robots in navigation and localization. This suggests that flow sensing could also provide useful information for autonomous vehicles. This study investigates air flow data behind truck-sized bluff bodies using data acquired from Computational Fluid Dynamics (CFD) simulations. The proposed features for classification are based on Fast-Fourier transforms. The results show that pressure data can be used to differentiate between various truck lengths, indicating that flow sensors could aid autonomous vehicles in overtaking.

Keywords:

Kujundus, paigutus, mall

CERCS: P170

Eessõitva sõiduki pikkuse hindamine rõhuandmete põhjal autonoomsetes sõidukites

Lühikokkuvõte:

Möödasõit on ohtlik ja keerukas manööver. Veelgi keerukam on see aga isesõitvate autode jaoks. Selleks on algoritmidel vaja hulgaliselt infot ümbritseva keskkonna kohta ning sealhulgas on vaja arvestada ka möödasõidetava sõiduki pikkusega. Tavapäraste autonoomsetes sõidukites kasutatavate sensoritega (kaamera, radar, LiDAR) pole võimalik seda pikkust aga hinnata. Teadustööd on näidanud, et veealuste robotite navigatsiooni ja asukoha määramise täpsust on võimalik parandada vedeliku voolu tajumise meetoditega. Sarnaseid meetodeid võib olla võimalik kasutada ka isesõitvates autodes. Selles töös uuritakse õhuvoolust tingitud rõhu muutumist veoauto taga, kasutades selleks arvutusliku vedeliku dünaamika simulatsioone. Klassifitseerimiseks kasutatavad tunnused põhinevad

Fourier’ kiirteisendusel. Töö tulemused näitasid, et rõhuandmete alusel on võimalik eristada erineva pikkusega veoautosid ning et selliseid meetodeid oleks võimalik kasutada ka isesõitvates autodes.

Võtmesõnad:

klassifitseerimine, õhuvoolu tajumine, sõiduki pikkuse hindamine, autonoomsed sõidukid, arvutuslik vedeliku dünaamika

CERCS: P170

Table of contents

1. Introduction

Many of the biggest car manufacturers are now introducing level 2 autonomy [1] (partial automation – e.g., adaptive cruise control, lane-keep assist) to their vehicles [2]. Such vehicles are still largely dependent on a human driver. However, some companies including Tesla [3, 4], Audi and Waymo [5] are already targeting autonomy levels beyond 2, where the vehicle can take full control of parts of a journey. According to [6], autonomous vehicles (AVs) have the potential to increase traffic efficiency, reduce pollution and eliminate over 90% of car accidents. AVs have the advantage of better perception and faster and more accurate decisionmaking and execution [7]. Yet, they aren't perfect either, for example in complex driving situations, humans might still perform better than AVs. There are also new risks that have to be considered, e.g., cyberattacks [7].

While higher levels of autonomy have been demonstrated, they still require testing and development to be considered safe. They can meet the requirements in controlled environments, but may fail to perform in unexpected and more dynamic situations [8]. One such situation is overtaking [9], which is considered to be one of the most complex maneuvers in autonomous driving. It consists of consecutive lane change and lane keeping maneuvers, both of which require different controllers and algorithms [10]. The vehicle needs extensive information about the surrounding environment to be able to execute the manoeuvre safely. Weather conditions, road quality and other drivers' unexpected behavior must also be considered [8].

Overtaking large vehicles (i.e., trucks) is even more complex as visibility is limited and it is therefore difficult to see vehicles in the oncoming lane. The length of a truck also determines the execution time of the maneuver, which is a crucial factor for safety. Studies have shown that over 40% of overtaking accidents are caused by the overtaking vehicle colliding with oncoming traffic (for human drivers) [11]. While the usual sensing modalities in AVs (i.e., camera, radar and LiDAR) are able to construct a detailed model of the environment, they are not able to estimate the length of a leading vehicle.

The goal of this study is to investigate the use of flow sensors as a complementary sensing modality for autonomous vehicles. We present a method of estimating the length of a leading vehicle (i.e., truck) using flow sensors placed on a trailing vehicle. The flow sensors are used to acquire pressure data over a period of time, which allows analysis of the frequency

domain. By extracting flow features from the data, we show that it is possible to classify and predict the length of a leading truck.

This work has also been submitted to the IEEE 27th International Conference on Emerging Technologies and Factory Automation [12], and is under review at the time of writing.

The thesis is divided into five parts. The second chapter gives an overview of the importance of flow sensing and its current uses in robotics. Previous work in the area of autonomous driving is also discussed. The third chapter gives an overview of the used software, the simulation scenes and setup parameters, and describes the methodology. The fourth chapter presents the results of various models tested in this study. The final chapter concludes the work and presents possible avenues for future investigations.

2. Background & related work

The first section gives an overview of the importance of flow sensing in the natural environment, which has led to development of biomimetic systems used in robotics. The second and third sections present examples of the use of flow sensing in underwater and aerial robotics, respectively. The final section discusses work in autonomous driving research.

2.1 Flow sensing in nature

Animals are not always able to rely on vision to sense their surrounding environment. Vision is affected in the dark as well as in other low visibility conditions (e.g. in water). Some animals have developed special organs to be able to live in these conditions. For example, bats use echolocation and some fish use electrolocation to sense their environment [13]. This is especially important in a deep sea environment, where fish have to be able to navigate, avoid predators, find food and reproduce [14]. The ability to do so relies on multiple sensory organs. One of these organs is called lateral line which allows the fish to sense water flow and low frequency vibrations [15]. Lateral line consists of structures called neuromasts and they are found on the skin (superficial neuromasts) or in canals (canal neuromasts). Some fish species have less than 50 superficial neuromasts while others can have several thousand. Using the lateral line, fish can detect the wakes of other aquatic animals. Some fish can also use lateral line to detect the motion direction, speed, size and shape of moving objects [16]. In recent years, research about lateral line has lead to the development of artificial lateral lines (ALLs) which can be a low-cost alternative to sonar and camera systems [17].

2.2 Flow sensing in underwater robotics

Autonomous underwater vehicles (AUVs) are unmanned vehicles used for surveying, monitoring and mapping the underwater environment [18]. They are used for both commercial and military applications in places where using manned vehicles would be too dangerous or expensive. Marine environments are often extremely unpredictable and therefore AUVs must be able to adapt to these varying conditions. They must be able to perform tasks and navigate while adjusting for sudden currents and downdrafts [18].

AUVs usually use dead reckoning[1] and inertial measurement units (IMUs) to estimate their position. However, IMUs have issues including sensor drifting and cumulative errors. Sensors such as sonars, pressure sensors and cameras are used to further enhance localization, however these systems are expensive and heavy, making them impractical for use on small AUVs. Insufficient lighting conditions and poor visibility also means that these systems cannot always be relied on [19].

Other methods of underwater localization include acoustic, GPS and visual positioning. Acoustic positioning system requires periodic maintenance and is also difficult to install on small-sized robotics due to its size and weight [20]. Underwater GPS positioning requires a separate floating buoy since radio waves cannot pass through water [19]. However, this system may not be accurate enough for small UAVs, as the location is determined by the buoy [20]. Many of these methods (e.g., acoustic positioning and sonar) perform active sensing, which means they are not very energy-efficient [21] and can consume over 40% of an AUVs power [22]. There is a need for sensors that are smaller and use less power to be used on today's smaller AUVs [21].

Pressure sensors have previously only been used for determining the depth, but in recent years, the previously mentioned ALLs have been integrated into underwater robots as well [18]. This was demonstrated in [23] and [24], where robots were able to detect and navigate along obstacles using the wall effect. The advantage of pressure sensors is low power consumption and small size, which makes it easier to install on existing robotic systems [18].

[25] introduces a biomimetic hair sensor design, which can be used in large arrays while maintaining compact dimensions (the hairs are 1 mm long). This artificial flow-sensor has the advantage of high sensitivity and low power consumption. Experiments showed that the sensor could accurately detect parallel and perpendicular velocity components of a vibration source and determine the distance to the source [25].

[17] presents a method of localization with flow sensing as the only perception modality. Their method uses flow-speed maps generated from CFD simulations in a model based on the real-world environment. They then equipped a fish-shaped object with an array of 16 absolute pressure sensors (ALL) to perform measurements in a real-world environment. The

[1] https://www.sciencedirect.com/topics/earth-and-planetary-sciences/dead-reckoning

data from the pressure sensors was used to estimate flow speed around the robotic body. A particle filter algorithm was then used for localization. It was shown that localization is possible in environments, where structural maps allow for simulation data to be acquired beforehand. The results indicated that flow sensing could be used for localization or be integrated into an underwater navigation system as a complementary sensing modality [17].

G. Liu et al. [15] propose an ALL system for detecting the coordinates, amplitudes and frequencies of underwater vibration sources. Twenty-five pressure sensors were used for the system. Simulations were first done to see how different positions of a vibration source affect the pressure distribution in the lateral line system. Experimental data showed similar results to the data from simulations. A neural network learning algorithm was used for localization and detection of frequency and amplitude components. It was also found that the effective range of the system is limited as the pressure decreases with increasing distance from the source. Results concluded that flow field sensing may be useful for improving identification accuracy of vibration sources and that this method could therefore have practical applications in deep-sea environments [15].

In their later work [26], G. Liu et al. designed a new shape for the robot based on the body of a boxfish (see figure 1). Its shape made it made it convenient to adjust the position of the ALL sensor array. Hydrodynamic simulations were done to optimize the placement of the system. This further increased the sensitivity and accuracy of the ALL. Neural networks were used to determine the motion parameters of the robot. Experiments showed that the bionic fish was able to accurately sense its environment [26].

Figure 1. (a) boxfish, (b) bionic boxfish for the ALL system [26]

2.3 Flow sensing in aerial robotics

Due to their small size, drones have found uses in a large variety of fields such as delivery services, agricultural monitoring and inspection of buildings [27]. However, the small size means their performance can easily be affected by wind. It is therefore necessary to have flight control systems that can counteract these external factors quickly and precisely. Knowing the wind direction and velocity can help improve the stability of these systems [27].

K. Haneda et al. [27] present a new type of airflow sensor for use in drone flight control systems. The disadvantages of other types of flow sensors are also discussed. For example, both a wind turbine-type sensor and an ultrasonic-type sensor are difficult to scale down to fit on drones. The latter is also expensive. Previously mentioned hair sensors are too fragile to be used in high wind speeds [27].

Differential pressure sensors (called Pitot tubes) that measure the pressure between two inlets are used in airplanes. The problem with these is that drones can fly in all directions, while planes fly in one direction only. The flight speed of drones is also much lower, and the sensor needs to be lightweight. The work by K. Haneda et al. presents an airflow sensor that is based on similar principles as a Pitot tube and that can be fitted on small drones [27].

C. G. Hooi et al.[28] use flow velocity measurements to estimate the height of a rotorcraft in ground effect, providing an additional sensing method for hovering and landing [28]. Ground effect is a condition where the air downwash from the rotors creates more lift for the rotorcraft [29]. They used differential pressure airspeed sensors to measure the flow speed and aerodynamic modelling to estimate the height using a Bayesian filter [28].

S. Zahran et al. [30] present an unconventional use for hall-effect and mass air flow sensors. They use the two sensors to estimate the forward velocity and create an odometer for quadcopters. This is combined with existing inertial measurement units (IMUs) to minimize cumulative errors. The mass air flow sensor is used for measurements at lower velocities because these sensors are sensitive to the direction of air flow. At higher wind velocities, the quadcopter starts to tilt to counteract the wind, in which case the hall effect sensor is used. The two sensors and the IMU measurements are fused using an extended Kalman filter (EKF) [30].

2.4 Flow sensing in autonomous driving research

Previous results on the use of flow sensing in autonomous vehicles are presented in [31]. The goal was to investigate the potential use of flow sensing for estimating the length of a vehicle (i.e., a truck) in front of an autonomous vehicle. Simulations were conducted for two different speeds (60 km/h and 90 km/h) and two different sized trucks (10 m and 17 m lengths). Both the truck length and the speed were found to affect the pressure levels behind the truck. Pressure levels also depend on the distance from the truck.

However, in some cases the pressure levels decreased as the truck length increased, and vice versa. Due to this, it was concluded that there was insufficient data to accurately predict the length. Additional simulations were also done for a windy environment and a third truck length, which showed expected results.

Results indicated that the pressure values are distinguishable in some cases, but further research is required. Proposed future work included using larger data sets and acquiring data in a real-world environment [31].

3. Methodology

We propose a method for estimating the length of a leading truck using features extracted from airflow data. The methodology relies on the use of computational fluid dynamics (CFD). First, simulations were done with trucks of various lengths. Pressure data from the simulations was then acquired at specific points on the trailing vehicle (car) over a period of time. This would allow us to analyse the frequency domain of the changing pressure values that could aid in estimating the truck length. Features used in this work are based on the Fast Fourier transform (FFT) [32]. After extracting the features, the resulting data was divided into training and testing data for use in classification and regression models. Further testing was done to find an optimal time window length for the data samples.

Furthermore, the distance between the car and the truck in the simulations is constant and this work assumes that this distance is always known. This is reasonable to assume as the existing sensors used in most autonomous vehicles can already detect this distance accurately [33].

The following sections give an overview of the used software, physics related to flow sensing, simulation setup, proposed features and prediction models used in the thesis.

3.1 Computational fluid dynamics (CFD) and software

CFD

The data used for analysis in this thesis has been acquired using computational fluid dynamics (CFD). It is the process of mathematically modelling fluid-flow phenomena based on the laws of physics. CFD is used widely by engineers for studying aerodynamics and hydrodynamics [34].

SimScale

The simulation software used in this work is SimScale[2]. It is a cloud-based platform for engineering applications in fluid dynamics, solid mechanics and heat transfer. It is based on the OpenFOAM toolbox[3]. SimScale offers a free Community account with limited runtime and computing power. For this work, the Academic Plan of SimScale was used, which is

[2] https://www.simscale.com/

[3] https://www.openfoam.com/

offered for free to students writing a thesis. This allowed us to run longer and more detailed simulations.

ParaView

ParaView[4] is an open-source data analysis and visualization application. It was used to analyse the simulation data as SimScale's web interface provides a rather limited toolset for extracting results. The simulation data downloaded from SimScale were about 40 gigabytes per simulation. ParaView was used to export the pressure data at specific points into .csv files (one file per timestep). We later merged these files into a single file using a simple Python script. All further analysis was done in MATLAB[5].

3.2 Simulation scene, truck and car models

The simulation scene contained a truck followed by a car. The distance between the vehicles is 10 meters and both vehicles were later (in SimScale) set to move at 25 meters per second (90 kilometers per hour). In the work by R. Matvejev [31], a 50-meter distance was used, which is the safe distance calculated based on the Estonian Traffic Act. In reality, the safe distance is often ignored by drivers prior to overtaking. Because of this, we decided to investigate a shorter distance between the vehicles as it might include stronger and more distinguishable frequency features that could dissipate with the increasing distance. Figure 2 shows the scene of the 12.5-meter truck.

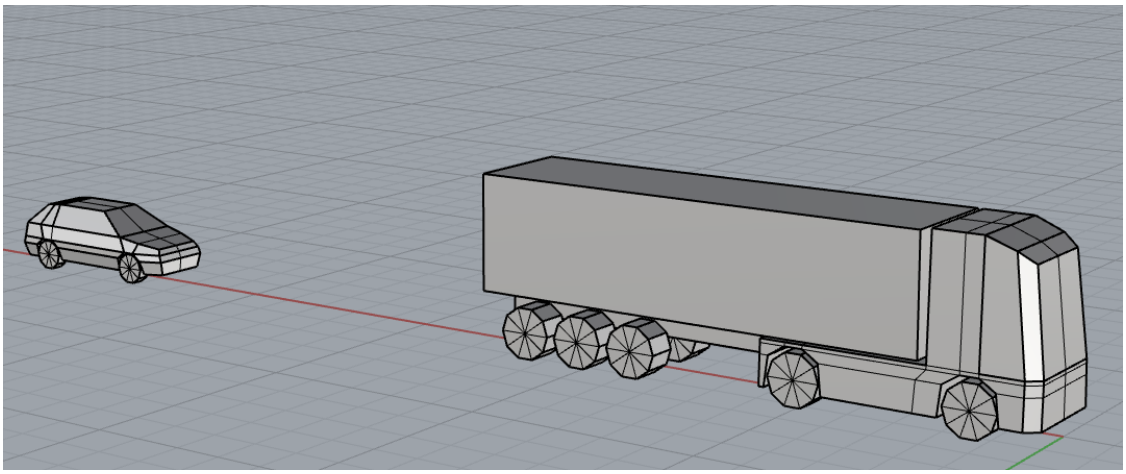

Figure 2. Simulation scene of the 12.5-meter truck.

[4] https://www.paraview.org/

[5] https://se.mathworks.com/products/matlab.html

To further verify the hypothesis that the aerodynamic effects are more distinguishable at a 10-meter distance, we used the first two simulations (10-meter and 20-meter trucks) to compare the pressure values. A pressure difference graph (shown in figure 3) indicated that the highest pressure difference was at a 12-meter distance. This varies between different simulations, so we concluded that the chosen 10-meter distance was a reasonable distance.

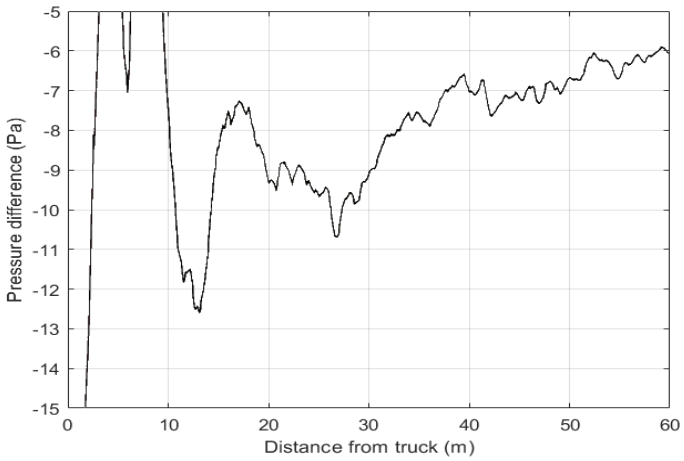

Figure 3. Pressure difference between 20-meter and 10-meter trucks.

In the first simulations, the truck was represented by a simple 3D bounding box with the dimensions of an average truck (2.5-meter width and 4-meter height). These simulations did not include a car model, which allowed us to extract data from any desired distance. However, this raised the question of how a car would affect the aerodynamic flow in the simulation and whether it would be necessary to include one. Two simulations were done to determine that. Figure 4 shows pressure maps with and without a car. It is evident that it does affect the flow and it was therefore decided to include the car in the final simulations.

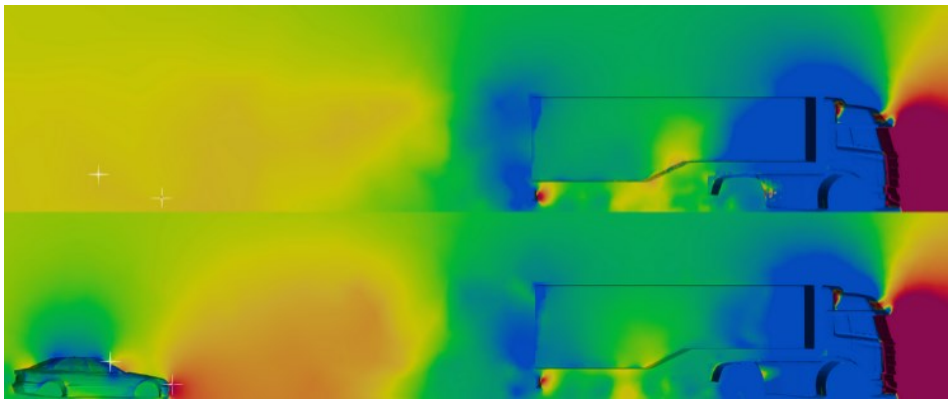

Figure 4. Pressure map with a car (bottom) and without a car (top). Warmer tones indicate higher pressure.

To achieve comparable results to the work by R. Matvejev [31], we first decided to use the same models he had used in his thesis (the ones in figure 4). However, these models proved to be too detailed (>100,000 faces) for running the transient simulations long enough to acquire usable amount of data for this thesis. The paid version of SimScale could be used for longer and more detailed simulations.

As the selection of free 3D models is very limited online, we created our own low-poly models of a truck and a car for the purpose of this thesis. This would allow us to significantly optimize simulation time, while preserving the most important details and features of both vehicles. The modelling was done in the open-source 3D software Blender[6]. The truck model was based on the popular Volvo FH series and the car was based on a Volkswagen Golf. The resulting models had 628 and 320 faces, respectively. The models can be seen in figure 2.

We created five trucks of different lengths for the simulations: 10, 12.5, 15, 17.5 and a 20-meter truck (total length of the truck, including the cabin). The length was changed by adjusting the trailer's length. Other parts of the truck were unchanged. The models were exported as .obj files and the final scenes were created in Rhinoceros 3D[7] software whose native file format was found to work well with SimScale.

To gain insight into how the proposed classification and regression models would work in a real-world scenario, we created a modified version of the truck cabin. The hypothesis was that this would slightly affect the aerodynamic flow and would give us a better idea of how our classifier would work in a real-world setting. The two models can be seen in figure 5.

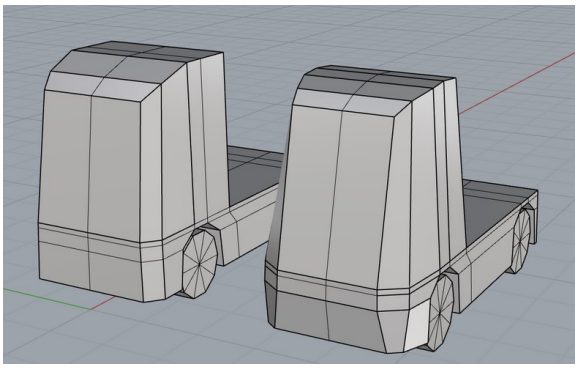

Figure 5. Original (left) and modified (right) truck models.

[6] https://www.blender.org/

[7] https://www.rhino3d.com/

3.3 Physics and simulation parameters in SimScale

The first step in simulating was to determine whether the flow behind the truck is laminar or turbulent. For this, we use the Reynolds number, which is a dimensionless number used to categorize flow [35]. It is defined as follows:

$$Re = \frac{\rho v L}{\mu} \tag{1}$$

In the formula, ρ is the fluid density, v is the maximum flow velocity, L is the characteristic length and μ is the dynamic viscosity. In our case these parameters are:

- $\rho_{air} = 1.225\ \frac{kg}{m^3}$
- $v = 25\ \frac{m}{s}$
- $L = 10\ m$
- $\mu_{air} = 1.81 \cdot 10^{-5}\ \frac{kg}{m \cdot s}$

There are many interpretations to what should be chosen as the characteristic length depending on the shape and size of the flow channel as well as the object being investigated [36]. As we need to determine if laminar flow is even possible, we can choose this to be a shorter dimension such as the width of the truck (2.5 meters).

From our parameters, the Reynolds number calculates to $Re \approx 4{,}230{,}000$. A flow is considered laminar if $Re \leq 2100$ [35]. Therefore, this flow is turbulent and requires a turbulence model for the simulations. The model chosen for the simulations was the k-omega SST model, which has been shown to be the most accurate among turbulence models used for automotive applications [37].

The idea to investigate the pressure changes behind the truck first came from the Kármán vortex street phenomena [38]. It is defined as "the periodic detachment of pairs of alternate vortices from a bluff-body immersed in a fluid flow, generating an oscillating wake, or vortex street, behind it" [39]. Kármán streets are observed in tall buildings [40] as well as in tidal currents in the ocean [41].

Vortex shedding frequency can be calculated from the Strouhal number. The equation for the Strouhal number is as follows:

$$St = \frac{fd}{V} \tag{2}$$

In the formula, f is the vortex shedding frequency, d is the width perpendicular to the flow direction (width of the truck) and V is the flow velocity (25 meters per second). Experiments have shown that the Strouhal number around bluff rectangular bodies is about 0.2 [42]. Assuming that, the shedding freuquency calculates to $f = 2\,Hz$. However, the Strouhal number depends on many factors, and for high Reynolds number flows there is little experimental data on it [42]. Because of this, we still decided to run simulations at a higher sampling rate of 50 hertz. This would allow us to investigate the importance of higher frequencies while not exceeding the runtime limitations of SimScale.

To achieve this sampling frequency, write interval was set to 0.02. Delta *t* (rate at which the simulation equations are solved) was set to 0.005 seconds. We first decided to run 60-second simulations to maximize the size of the data sets. However, at this sampling rate, SimScale's service ran out of this space at about 30-40 seconds. The final usable data set size for each simulations was 30 seconds.

Each CFD simulation requires an enclosure in which the fluid flow is set to be simulated. The dimensions of this enclosure were set close to SimScale's automatic suggestion for the longest truck model. These dimensions were kept constant across all the simulations to avoid affecting the flow in any way.

An issue with the simulations was that all of them exceeded the Courant number limit of 1. The Courant-Friedrichs-Lewy (CFL) condition states that information traveling through the simulation mesh elements must not skip any cells (i.e., information can only propagate to an adjacent cell) [43]. The solution is to either lower the timestep or use a coarser mesh for the simulation [43]. These were not possible, because meshing was already set to the coarsest possible setting to optimize runtime. Lowering the timestep would have also greatly increased this runtime. Fortunately, the CFL condition was only exceeded in parts of the simulation and the mean value was 0.15, which is well below the limit.

3.4 Feature definition

The initial simulation results of the five different length trucks showed that the pressure levels for the 12.5-meter and the 17.5-meter trucks were most similar, while both the 12.5-meter and 20-meter trucks showed significantly lower pressure data. Based on this, it was concluded that it is difficult to predict the length of the truck based solely on the pressure levels.

However, it was noted that the frequencies and amplitudes of the signals were visually distinguishable, which lead to the idea of using frequency domain analysis to analyse these frequency components. The proposed features used in this work are based on the Fast-Fourier transform (FFT) algorithm, which converts the signal data from time domain to frequency domain, which shows the signal's energy distribution over the frequency range [44]. Figure 6 shows an example signal with a length of 32 seconds, the calculated FFT and the resulting histogram feature. The histogram contains 100 bins with each bin corresponding to a range of 2.5 hertz. The proposed feature is similar to the one presented in [45], where histogram distances were later used to perform localization in an underwater environment.

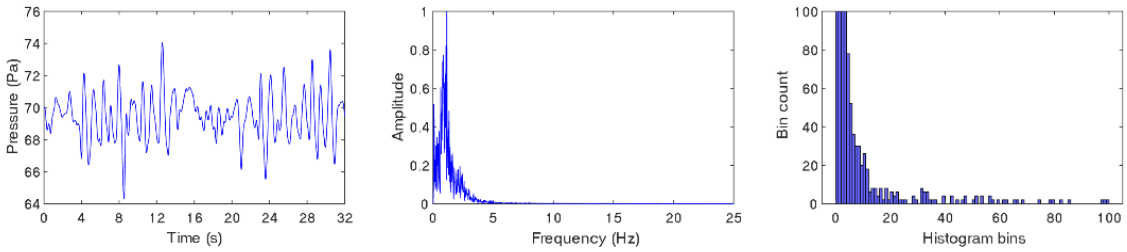

Figure 6. Pressure data (left), FFT (middle) and resulting feature (right).

3.5 Random forest models

In this study, the random forest algorithm was used to build a prediction model. It was chosen because of two reasons: the random selection of subsets and averaging of results helps avoid overfitting, and it is more accurate than most classifiers [46]. It is also used for both classification and regression problems, both of which we wanted to test in this thesis. For classification, the output is the most popular tree output and for regression, the output is the average of all tree outputs [46].

The disadvantage of the algorithm is speed. Having multiple decision trees can become very time-consuming [46]. In the scope of this thesis, this was not an issue, as the data sets were small and predictions took a fraction of the analysed signal length. However, speed could become an issue when using larger data sets and testing in a real-world environment.

The results section presents results of both a classification and a regression model.

3.6 Histogram distance classifier

In addition to the random forest model, a classifier based on histogram distances was also proposed. For this, we would use the same histogram features and compare these features against various histogram distance metrics. A paper that provides an overview of various histogram distance methods is [47]. To classify truck lengths, the extracted feature was compared against reference features from different truck length simulations. This classification method is discussed further in the results section.

4. Results and discussion

There were 10 simulations in total that were used for testing the prediction models (five different truck lengths and two different truck models). A comparison of the pressure data of different truck lengths can be seen in figure 7. The figure shows a substantial drop in pressure in the first few seconds and then stabilizing at about 5 seconds. This is because it takes time for the flow to develop and stabilize at the beginning of the simulation. For this reason, the first 5 seconds of data were removed from the data. The resulting dataset was 30 seconds for each simulation.

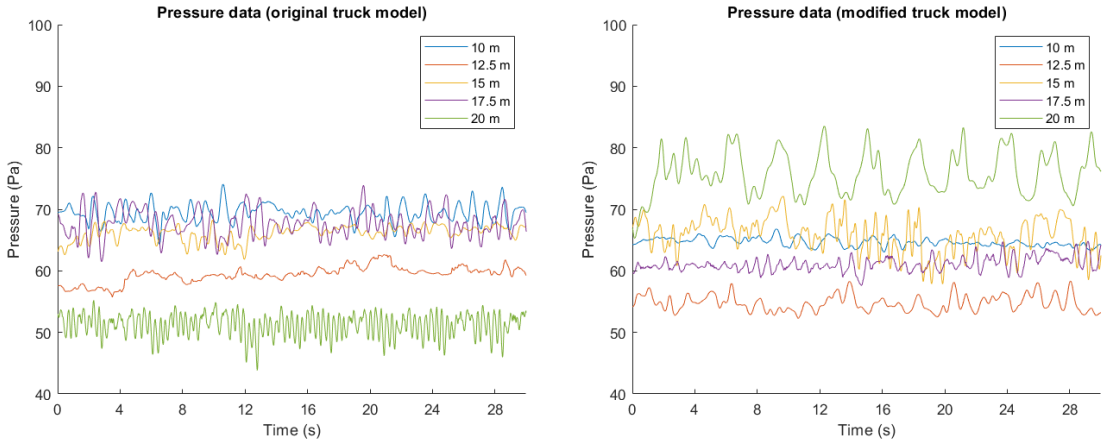

Figure 7. Pressure data comparison of different truck lengths.

4.1 k-fold cross-validation

The first presented classification and regression model results use only the data from the original truck model simulations. To have less biased results, we used k-fold cross-validation for training and testing the model. It is a method of estimating the accuracy of a model on unseen data (data that was not used during training) [48]. The data is split into *k* groups, one by one, each group is used as the testing data while others are used for training. An illustration of this is shown in figure 8.

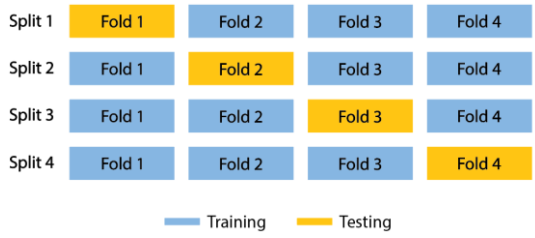

Figure 8. k-fold cross-validation

We chose $k = 4$, as this produced a data set of 7.5 seconds per fold and still allowed us to generate data samples with varying lengths (data sample length selection is discussed in section 4.4). From each fold, 100 random data samples were generated. Features were generated for each of the samples.

A random model forest was trained based on the features and the lengths of the corresponding truck simulations.

Further results are presented where simulation data from both models are used: the original and the modified truck model for training and testing, respectively.

The predictions were saved in a 5-by-5 matrix to later analyze any mispredictions.

4.2 Classification results

In the first test, the classifier was trained using the FFT features as predictors. Each histogram bin was used as a separate predictor. This is a standard approach when training decision trees using histogram data [49]. Using only the FFT features allowed us to determine whether the frequency data has any distinguishable information for estimating the truck length. Results are presented in table 1.

Table 1. Random forest classification model results (predictors: histogram bins). Prediction results as a percentage - columns show which length the corresponding simulation in the row was predicted as.

		Prediction				
		10 m	***12. 5m***	***15 m***	***17.5 m***	***20 m***
Simulation	***10 m***	**85**	10	0	4	1
	12.5 m	31	**38**	10	5	16
	15 m	5	22	**27**	14	32
	17.5 m	0	5	8	**81**	6
	20 m	13	18	9	0	**60**

The average accuracy across all tests is 0.58. The 15-meter truck has the lowest prediction accuracy. It was most often mispredicted as a 20-meter and a 12.5-meter truck. The 12.5-meter truck also has a low accuracy, and it was often predicted as a 10-meter and a 20-meter truck. This means that the prevalent frequencies in the data were similar across these simulations. Both 10 and 17.5-meter trucks showed high accuracy, meaning the data from these simulations was most distinct.

The next test featured the median pressure of the data sample as an additional predictor. The hypothesis here was that this would significantly increase accuracy as pressure levels between the data sets were quite distinct.

As expected, the accuracy improved greatly with a mean value of 0.89. However, these results are based on a very limited data set and ideal conditions (constant speed and distance, no crosswind or other external factors) and it therefore it does not reflect the real-world performance of the model. Because of this, it was proposed to run the additional modified truck simulations to be able to better separate the training and testing data (results shown in section 4.6).

Table 2. Random forest classification results (predictors: histogram bins, median pressure)

		Prediction				
		10 m	***12. 5m***	***15 m***	***17.5 m***	***20 m***
Simulation	***10 m***	**94**	0	6	0	0
	12.5 m	0	**96**	4	0	0
	15 m	10	0	**80**	7	3
	17.5 m	0	0	17	**83**	0
	20 m	7	0	1	0	**92**

4.3 Regression results

Because trucks lengths vary in the real world, it was also decided to train and test a regression model, which would predict intermediate values. The results of this can be seen in figures 9 and 10.

Same predictors were used as in the classification model. The first test used the FFT features (histogram bins) and median pressure values. Like in the respective classification model, the results are much more accurate than without the pressure values. A noteworthy difference is that none of the truck lengths were predicted to be at the other end of the length range (e.g., as in table 1, where the 20-meter truck was predicted to be a 10-meter truck 13% of the time). In figure 10, only FFT features were used as predictors. The median prediction values were within a few meters of the correct length, but overall the variance was quite high, meaning the results are unreliable.

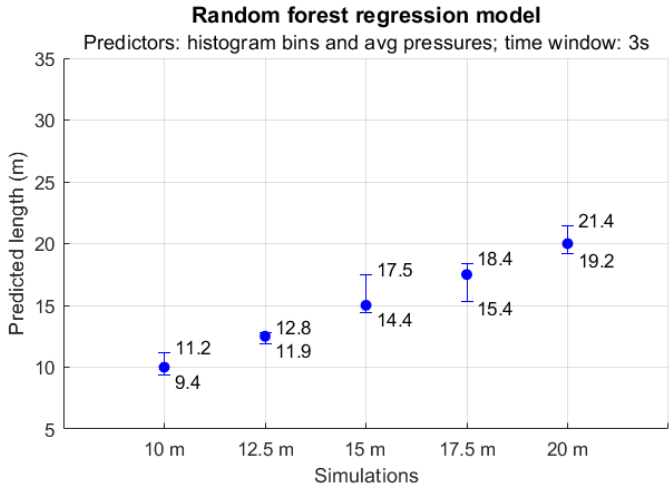

Figure 9. Regression model results. Dots show the median prediction values and the top and bottom caps show maximum and minimum prediction values, respectively.

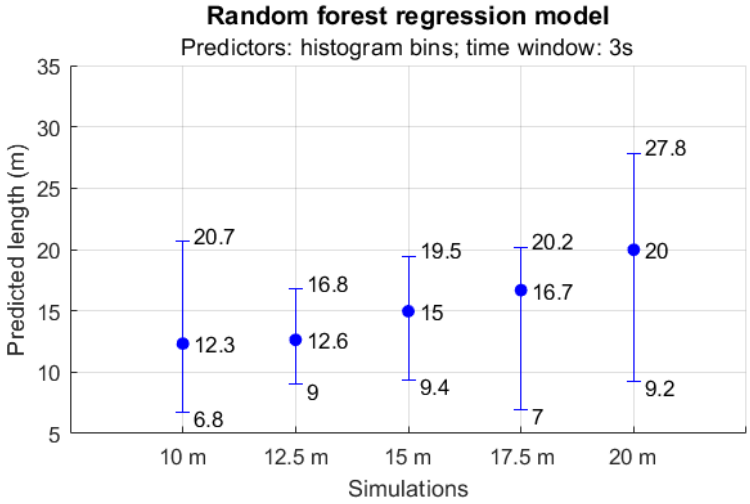

Figure 10. Regression model results with FFT features as the only predictors.

4.4 Optimal time window

To determine an optimal data sample length, tests were done to compare the average accuracy at varying sample lengths. For this, the classification model was used. Data samples were generated with lengths from 0.25 to 5 seconds with an interval of 0.5 seconds. This was performed separately for each truck lengths. Figure 11 shows the accuracy graph for the 10-meter truck simulation. The optimal sample length was determined to be about 3 seconds, which was used for all of the testing in this project (including the results presented

above). A longer sample length was not used, because it would create too much overlap between different samples.

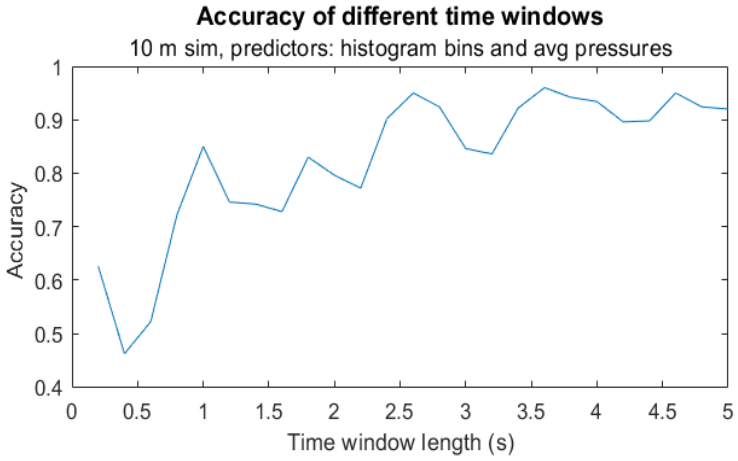

Figure 11. Accuracy of classification at different time window lengths.

4.5 Predictor importance

MATLAB provides a method for estimating individual predictor importance by permutation of predictor observations [50]. Figure 12 shows the importance graph of the random forest model with histogram bins and median pressure as predictors. The x-axis corresponds to the predictors. Indices 0 to 100 are the histogram bins and the 101th index is the median pressure predictor. It can be noted from this graph that the lower frequency components provide more meaningful information for classification. The pressure predictor importance is far above the rest, because this was most distinct between simulations. Tests were also done with higher frequencies (> 10 hertz) removed. This showed some increase in the accuracy.

Experiments were also done with different number of histogram bins. However, the results were largely similar (e.g., between 100-bin and 20-bin histograms), so the number of bins were not changed for further testing.

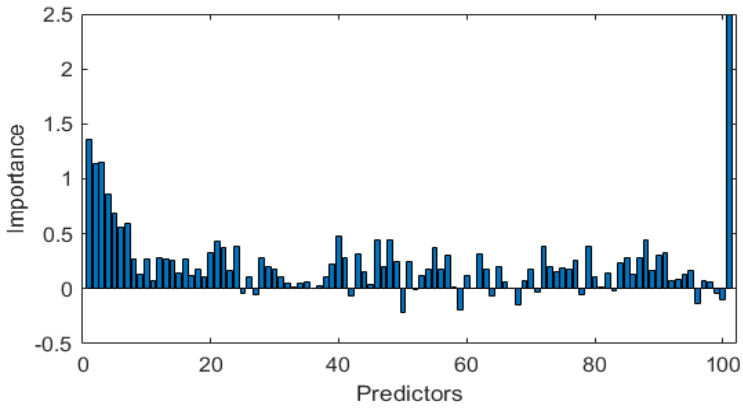

Figure 12. Predictor importance graph. Bins 0 to 100 represent the histogram bins and the 101st bin is the median pressure predictor.

4.6 Modified truck results

The simulation data from the modified truck model was quite different from the original: the peak-to-peak amplitudes were much lower and the pressure levels also varied by about 10 pascals. However, visually, some of the frequencies did look similar. Data comparison of the 17.5-meter truck simulations is shown in figure 13. The results of the prediction model are presented in table 3.

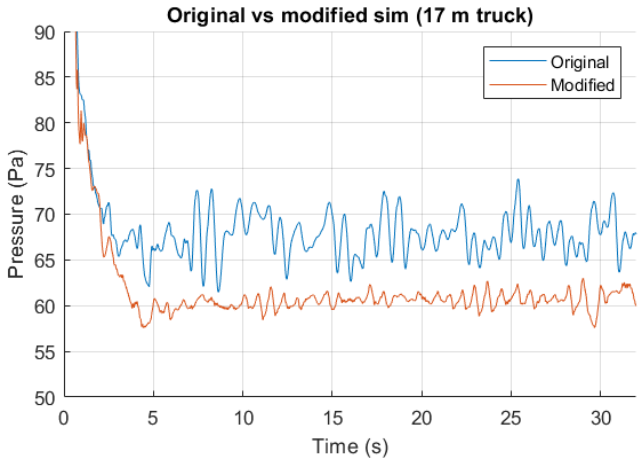

Figure 13. Pressure data comparison between the two different models.

Table 3. Random forest classification model on modified truck model (predictors: histogram bins). Prediction results as a percentage - original truck simulations as training data and modified truck simulations as testing data.

		Prediction				
		10 m	***12. 5m***	***15 m***	***17.5 m***	***20 m***
Simulation	***10 m***	**24**	0	54	22	0
	12.5 m	15	**4**	31	15	35
	15 m	14	3	**30**	4	49
	17.5 m	19	2	46	**26**	7
	20 m	8	8	6	76	**2**

This test proved to be a limitation of this study. The data varied too greatly to be able to accurately classify the trucks. It is likely that this is because of inaccurate simulations results due to low quality meshing and a large timestep (as described earlier). It is possible that better results could be achieved with larger datasets and more detailed simulations that were unfortunately out of scope for this project.

4.7 Classification using histogram distances

Another proposed method was to use histogram distance measures to compare and classify the data. Various distance measures including intersection, city block distance, Euclidean distance and Chebyshev distance were tested, but cosine similarity proved to be the most accurate. Cosine similarity is defined as follows:

$$\cos(\theta) = \frac{\mathrm{A}\cdot\mathrm{B}}{||\mathrm{A}||\cdot||\mathrm{B}||} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}} \tag{3}$$

To first see whether this is a feasible method for classification, we first compared cosine similarities between data from the same simulation, and then between data from different simulations. First, 500 random data samples from the 10-meter simulation were extracted. The FFT features were calculated. The features were then compared against each other in a 500-by-500 matrix. After that, the median value of the similarity matrix was calculated (not taking into account similarities equal to 1, i.e., the diagonal elements). Next, 500 random samples from other simulations were extracted. These were then compared against the features from the 10-meter simulation.

This process was repeated for sample lengths from 0.5 to 30 seconds with an interval of 0.1 seconds. The results are shown in figure 14.

Results show that features extracted from the same simulation data have an average similarity of 0.89. Similarities are significantly lower when comparing to data from other simulations, with an average similarity of 0.56. Results are consistent throughout the length range. It was expected that at the lower end of the range, similarities within the same simulation would be lower. Similarity compared to different simulations starts to drop when nearing the length of the dataset, and similarity within the same simulations approaches 1 (because at a sample length of 30 seconds, all generated samples will match).

These results indicated that classification could be possible using the cosine similarity metric.

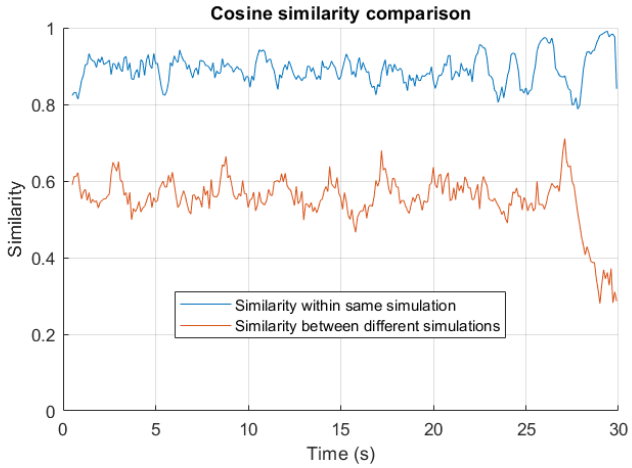

Figure 14. Cosine similarity comparisons over varying time lengths. Blue line shows similarities between samples from the same simulation and orange line shows similarities between samples from different simulations.

Classification testing was done by acquiring a random data sample and then comparing it against 1000 random samples from each of the truck simulations. This was repeated 100 times for all truck simulations. The results are presented in table 4.

Table 4. Classification using cosine similarity. Results as a percentage.

		Prediction				
		10 m	***12. 5m***	***15 m***	***17.5 m***	***20 m***
Simulation	***10 m***	**78**	8	0	4	10
	12.5 m	27	**49**	4	5	15
	15 m	6	21	**38**	2	33
	17.5 m	5	11	5	**73**	6
	20 m	17	22	11	1	**49**

The resulting accuracy is similar to the random forest's results: average prediction accuracy across all truck lengths was 0.57. This is a promising result, which could be investigated further in a future work.

4.8 Combination of histogram distances and the random forest model

The two classifiers were also run on the same data samples to compare their performance. This was done to see if the algorithms predict correctly in different cases. The results are presented in table 5.

Table 5. Accuracy comparisons between histogram distance algorithm and random forest model. Results as a percentage.

	Similarity	**Random forest**	**At least one correct**	**Both correct**
10 m	68	70	79	60
12.5 m	25	25	38	12
15 m	32	45	52	10
17.5 m	57	76	77	56
20 m	30	22	44	8

The average accuracies for the similarity classification and the random forest model are 42% and 47%, respectively. Both of the models were correct only 36% of the time, while at least one was correct 57% of the time. This indicates that an algorithm combining the two classifiers could therefore be used to further improve prediction accuracy in a future work.

5. Conclusion and future work

5.1 Conclusion

This study presents a method for estimating the length of a leading truck using features based on the Fast-Fourier transform. Air flow data is obtained from CFD simulations of five different truck lengths. Random forest models (classification and regression) are used to predict the length of the truck. The simulation data shows that the pressure data contains frequency components that can be used to differentiate between different truck lengths. This indicates that such methods could be useful in a real-world scenario as well and that flow sensors could complement the usual sensing modalities found in autonomous vehicles.

5.2 Future work

As shown in this study, the lower frequency signals were most important for classification and should therefore be studied further. Other methods such as discrete wavelet transforms (DWTs) and empirical mode decomposition (EMD) could be used to further analyze the frequency domain. Histogram distance classification showed equally accurate results and should also be studied further.

It would likely be beneficial to first have larger datasets. Longer and more detailed simulations could be run, but it is advised to also start gathering data from a real-world environment to see how the data compares to simulations. A brief overview and examples of possible sensors to use for real-world testing is given the appendix.

As discussed in section 4.8, a random forest model could also be used in combination with histogram distance metrics to increase confidence in cases where the model results are almost identical.

6. References

[1] SAE International Releases Updated Visual Chart for Its “Levels of Driving Automation” Standard for Self-Driving Vehicles [Internet]. SAE International; 2018 [cited 2022 May 8] Available from: https://www.sae.org/news/press-room/2018/12/sae-international-releases-updated-visual-chart-for-its-“levels-of-driving-automation”-standard-for-self-driving-vehicles

[2] Autonomous Vehicle Outlook 2022 [Internet]. TOMORROW’S WORLD TODAY®. 2022 [cited 2022 May 8]. Available from: https://www.tomorrowsworldtoday.com/2022/01/10/autonomous-vehicle-outlook-2022/

[3] Tesla Likely to Achieve Level 4 Autonomy in 2022, Says Elon Musk [Internet]. TESMANIAN. Available from: https://www.tesmanian.com/blogs/tesmanian-blog/tesla-likely-to-achieve-level-4-autonomy-in-2022-says-elon-musk

[4] Tesla. Autopilot [Internet]. Tesla.com. 2019. Available from: https://www.tesla.com/autopilot

[5] Hyatt K. Self-driving cars: A level-by-level explainer of autonomous vehicles [Internet]. Roadshow. CNET; 2018. Available from: https://www.cnet.com/roadshow/news/self-driving-car-guide-autonomous-explanation/

[6] Bonnefon J-F, Shariff A, Rahwan I. The social dilemma of Autonomous Vehicles. Science. 2016;352(6293):1573–6.

[7] Kalra N, Paddock S. Driving to safety: How many miles of driving would it take to demonstrate autonomous vehicle reliability? 2016

[8] New algorithm makes overtaking smoother and faster for self-driving cars | TalTech [Internet]. taltech.ee. [cited 2022 May 4]. Available from: https://taltech.ee/en/news/new-algorithm-makes-overtaking-smoother-and-faster-self-driving-cars

[9] Malayjerdi E, Sell R, Malayjerdi M, Udal A, Bellone M. Practical path planning techniques in overtaking for autonomous shuttles. Journal of Field Robotics. 2022 Jan 11.

[10] Petrov P, Nashashibi F. Modeling and Nonlinear Adaptive Control for Autonomous Vehicle Overtaking. IEEE Transactions on Intelligent Transportation Systems. 2014 Aug;15(4):1643–56.

[11] Richter T, Ruhl S, Ortlepp J, Bakaba E. Causes, consequences and countermeasures of overtaking accidents on two-lane rural roads. Transportation Research Procedia. 2017;25:1989–2001.

[12] Ottan M, Muhammad N. Leading vehicle length estimation using pressure data for use in autonomous driving. Under review at IEEE 27th International Conference on Emerging Technologies and Factory Automation (ETFA). 2022.

[13] Tan D, Patton P, Coombs S. Do blind cavefish have behavioral specializations for active flow-sensing? Journal of Comparative Physiology A. 2011 Mar 23;197(7):743–54.

[14] Marranzino AN, Webb JF. Flow sensing in the deep sea: the lateral line system of stomiiform fishes. Zoological Journal of the Linnean Society. 2018 Jan 15;183(4):945–65.

[15] Liu G, Gao S, Sarkodie-Gyan T, Li Z. A novel biomimetic sensor system for vibration source perception of autonomous underwater vehicles based on artificial lateral lines. Measurement Science and Technology. 2018 Oct 23;29(12):125102.

[16] Bleckmann H, Zelick R. Lateral line system of fish. Integrative Zoology. 2009 Mar;4(1):13–25.

[17] Muhammad N, Fuentes-Perez JF, Tuhtan JA, Toming G, Musall M, Kruusmaa M. Map-based localization and loop-closure detection from a moving underwater platform using flow features. Autonomous Robots. 2018 Aug 24;43(6):1419–34.

[18] Hoth J, Kowalczyk W. Determination of Flow Parameters of a Water Flow Around an AUV Body. Robotics. 2019 Jan 28;8(1):5.

[19] Dang F. Flow Sensing Based Environmental Perception of Autonomous Underwater Robots. ProQuest, 2021.

[20] He Y, Zhu L, Sun G, Qiao J. Visual positioning system for small-scaled spherical robot in underwater environment. Microsystem Technologies. 2018 Jun 9;25(2):561–71.

[21] Kottapalli AGP, Asadnia M, Barbastathis G, Triantafyllou M, Miao JM, Tan CW. Polymer MEMS pressure sensor arrays for fish-like underwater sensing applications. Micro & Nano Letters. 2012 Dec 1;7(12):1189–92.

[22] Zhang X, Shan X, Xie T, Miao J, Du H, Song R. Harbor seal whisker inspired self-powered piezoelectric sensor for detecting the underwater flow angle of attack and velocity. Measurement. 2021 Feb;172:108866.

[23] Yen W-K, Sierra DM, Guo J. Controlling a Robotic Fish to Swim Along a Wall Using Hydrodynamic Pressure Feedback. IEEE Journal of Oceanic Engineering. 2018 Apr;43(2):369–80.

[24] Xu Y, Mohseni K. A Pressure Sensory System Inspired by the Fish Lateral Line: Hydrodynamic Force Estimation and Wall Detection. IEEE Journal of Oceanic Engineering. 2017 Jul;42(3):532–43.

[25] Dagamseh AMK, Lammerink TSJ, Kolster ML, Bruinink CM, Wiegerink RJ, Krijnen GJM. Dipole-source localization using biomimetic flow-sensor arrays positioned as lateral-line system. Sensors and Actuators A: Physical. 2010 Aug;162(2):355–60.

[26] Liu G, Wang M, Xu L, Incecik A, Sotelo MA, Li Z, et al. A new bionic lateral line system applied to pitch motion parameters perception for autonomous underwater vehicles. Applied Ocean Research. 2020 Jun;99:102142.

[27] Haneda K, Matsudaira K, Noda R, Nakata T, Suzuki S, Liu H, et al. Compact Sphere-Shaped Airflow Vector Sensor Based on MEMS Differential Pressure Sensors. Sensors. 2022 Jan 30;22(3):1087.

[28] Chin Gian Hooi, Lagor FD, Paley DA. Flow sensing, estimation and control for rotorcraft in ground effect. 2015 IEEE Aerospace Conference. 2015 Mar;

[29] IGE and OGE in aviation [Internet]. Helis.com. [cited 2022 May 4]. Available from: https://www.helis.com/howflies/igeoge.php

[30] Zahran S, Moussa A, El-Sheimy N. Enhanced UAV navigation using hall-magnetic and air-mass flow sensors in indoor environment. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences. 2019 May 29;IV-2/W5:187–94.

[31] Matvejev R, Muhammad Y, Muhammad N. Air-flow sensing for vehicle length estimation in autonomous driving applications. 2021. Paper presented at 26th International Conference on Emerging Technologies and Automation, Västerås, Sweden.

[32] Maklin C. Fast Fourier Transform [Internet]. Medium. 2019 [cited 2020 May 26]. Available from: https://towardsdatascience.com/fast-fourier-transform-937926e591cb

[33] Reliable distance measurements with radar sensor [Internet]. Sentech. [cited 2022 May 5]. Available from: https://www.sentech.nl/en/sensor-technology/reliable-distance-measurements-with-radar-sensor

[34] What is CFD | Computational Fluid Dynamics? [Internet]. SimScale. Available from: https://www.simscale.com/docs/simwiki/cfd-computational-fluid-dynamics/what-is-cfd-computational-fluid-dynamics/

[35] Shashi Menon E. Fluid Flow in Pipes. Transmission Pipeline Calculations and Simulations Manual. 2015;149–234.

[36] What is the best characteristic length to calculate Reynolds number? [Internet]. ResearchGate. [cited 2022 May 5] Available from: https://www.researchgate.net/post/What-is-the-best-characteristic-length-to-calculate-Reynolds-number

[37] Igali D, Mukhmetov O, Zhao Y, Fok SC, Teh SL. Comparative Analysis of Turbulence Models for Automotive Aerodynamic Simulation and Design. International Journal of Automotive Technology. 2019 Sep 16;20(6):1145–52.

[38] Alexander D E. Fluid biomechanics. Nature's Machines. 2017:59-97.

[39] Von Kármán vortex shedding - Encyclopedia of Mathematics [Internet]. Encyclopediaofmath.org. 2020 [cited 2022 Apr 24]. Available from: https://encyclopediaofmath.org/wiki/Von_K%C3%A1rm%C3%A1n_vortex_shedding

[40] Irwin PA. Vortices and tall buildings: A recipe for resonance. Physics Today. 2010 Sep;63(9):68–9.

[41] Maul G A. Intra-americas sea. Encyclopedia of Ocean Sciences. 2001;1354-1363.

[42] Vibration Induced by Cross-Flow. Flow-induced Vibrations. 2014;29–115.

[43] The CFL Condition and How to Choose Your Timestep Size [Internet]. SimScale. 2017 [cited 2022 May 4]. Available from: https://www.simscale.com/blog/2017/08/cfl-condition

[44] Practical Introduction to Frequency-Domain Analysis - MATLAB & Simulink Example - MathWorks Nordic [Internet]. se.mathworks.com. [cited 2022 May 4]. Available from: https://se.mathworks.com/help/signal/ug/practical-introduction-to-frequency-domain-analysis.html

[45] 1. Muhammad N, Strokina N, Toming G, Tuhtan J, Kamarainen J-K, Kruusmaa M. Flow feature extraction for underwater robot localization: Preliminary results. 2015 IEEE International Conference on Robotics and Automation (ICRA). 2015 May;

[46] Team TA, Team TA. Why Choose Random Forest and Not Decision Trees – Towards AI — The Best of Tech, Science, and Engineering [Internet]. Available from: https://towardsai.net/p/machine-learning/why-choose-random-forest-and-not-decision-trees

[47] Cha S. Taxonomy of nominal type histogram distance measures. MATH'08: Proceedings of the American Conference on Applied Mathematics. 2010;325-330

[48] [k] Brownlee J. A Gentle Introduction to k-fold Cross-Validation [Internet]. Machine Learning Mastery. 2018. Available from: https://machinelearningmastery.com/k-fold-cross-validation/

[49] Gurund R B, Lindgren T, Boström H. Learning Decision Trees from Histogram Data Using Multiple Subsets of Bins. Proceedings of the Twenty-Ninth International Florida Artificial Intelligence Research Society Conference. 2016;430-435

[50] Estimates of predictor importance for classification tree - MATLAB - MathWorks Nordic [Internet]. se.mathworks.com. [cited 2022 May 5]. Available from: https://se.mathworks.com/help/stats/compactclassificationtree.predictorimportance.html

[51] Pressure or airflow sensors? [Internet]. [cited 2022 May 10]. Available from: https://sps.honeywell.com/content/dam/honeywell-edam/sps/common/en-us/industries/manufacturing/industrial-equipment/documents/sps-his-pressure-or-airflow-sensors.pdf

[52] Michèle Beyer. Difference between gauge pressure and absolute pressure - WIKA blog [Internet]. WIKA blog. 2019. Available from: https://blog.wika.com/knowhow/difference-between-gauge-pressure-and-absolute-pressure-measurement/

Appendix

I. Flow sensors

This section gives a brief overview of various sensors used for measuring airflow that could be used in future work to acquire real-world flow data. In general, there are two types of sensors that can be used for measuring airflow: mass air flow sensors and pressure sensors. [51] provides a comparison between the two sensor types.

A mass flow sensor has an inlet and an outlet for measuring flow. Gas flow between the two ports creates a thermal imbalance, which is measured by a circuit inside the sensor. Pressure sensors also have two ports, but only require the small gas flow created by compression and expansion of gases. Inside of the sensor is a diaphragm, whose deflection is measured to derive the airflow rate. Mass airflow sensors do not have a linear output curve and they have better resolution at lower flow rates. The output curve of pressure sensors is practically linear. Airflow sensors also require the gas to be free of contaminants as even small amounts of dirt can cause errors in the output. Contaminants in a pressure sensor may cause problems with frequency response, however the output remains correct. It is noted that sometimes both types of sensors are used together to create a fail-safe. The frequency response for most airflow sensors is typically below 100 Hz while pressure sensors have a faster response of 1 kHz. This means the output of the pressure sensor may require filtering to provide a noise-free signal [51].

The CPT2500[8] USB pressure sensor could be used to perform testing. It features multiple measuring ranges and it can measure gauge pressure. Gauge pressure is the difference between absolute pressure and atmospheric pressure [52]. The results in this work showed a pressure variance of about 50 Pa, but this could vary in the real world. The suitable measuring range for this sensor would therefore be 0 … 25 mbar, which corresponds to 0 … 2500 Pa. In the documentation, it is noted that for ranges < 100 mbar, it has an accuracy of 0.2 % FS. For the chosen range, the accuracy would therefore be ±5 Pa, which could be accurate enough for testing purposes. It also has a maximum sampling rate of up to 1 kHz meaning that higher frequencies can be investigated as well.

[8] https://www.wika.com/media/Data-sheets/Calibration/Precision-pressure-measuring-instruments/ds_ct0501_en_co.pdf

II. License

Non-exclusive licence to reproduce the thesis and make the thesis public

I, Matis Ottan,

1. grant the University of Tartu a free permit (non-exclusive licence) to

reproduce, for the purpose of preservation, including for adding to the DSpace digital archives until the expiry of the term of copyright, my thesis

Leading vehicle length estimation using pressure data for use in autonomous driving,

supervised by Naveed Muhammad.

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

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

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

Matis Ottan

10/05/2022