Acoustic Odometry for Wheeled Robots on Loose Sandy Terrain

Andreu Gimenez Bolinches
Graduate School of Science and Technology
Keio University, Japan
andreu@keio.jp

Genya Ishigami
Graduate School of Science and Technology
Keio University, Japan
ishigami@mech.keio.ac.jp

***Abstract*—This work paves the way toward inexpensive robust robot localization by proposing a system capable of estimating the longitudinal velocity of a wheeled robot on granular non-cohesive soil using only acoustic data. Many future ground mobile robots will be equipped with microphones for human-robot interaction purposes and there already exist a wide variety of efficient methods for audio signal processing. The proposed system can be combined with other self-localization methods to strengthen their robustness with a minimal additional price.**

The proposed system consists of an audio feature extraction module, based on gammatone filterbanks, and a prediction module, based on a convolutional neural network. Experiments in a single wheel testbed with a wheel driving up to speeds of 0.07 m/s over loose sandy terrain with a wide range of slippage, show that the system is a feasible auxiliary source of odometry with an average drift of 5 mm/s. The system is able to make a prediction from a single audio frame with a duration of 15 ms in only 2 ms on a user-level commercially available CPU. Additional experiments with white Gaussian noise show that high noise power (signal-to-noise ratio of -10 dB) only affects significantly the prediction of speeds close to 0.

A qualitative evaluation of the proposed system against other sources of odometry shows that acoustic and visual methods' vulnerabilities do not overlap, which indicates that their combination would be beneficial from the robustness point of view. While the proposed system can recognize slippage even using devices not present during training, the estimated magnitude of the longitudinal velocity depends on the device used and the power of the sensed audio signal. This indicates that combining the proposed system with wheel odometry input could significantly increase the performance and generalization of the system while keeping a small computational cost.

***Index Terms*—component, formatting, style, styling, insert**

I. Introduction

A. Problem

Robots are machines that resemble living creatures in being capable of moving independently (as by walking or rolling on wheels) and performing complex actions (such as grasping and moving objects) [12]. In accomplishing a defined mission, a robot physically interacts with its operating environment. Robot operating environments can be classified into predefined, semi-structured, and unstructured [11]. In an unstructured environment, the robot has no prior knowledge about it and has to rely on its sensory and navigation systems to operate autonomously. Hence the problem of robot localization is defined as the process of determining where a mobile robot is located in its environment [19]. Self-localization is one of the most fundamental competencies required by an autonomous robot as the knowledge of its location is an essential precursor to making decisions about future actions.

Wheels or tracks still form the basis for robot locomotion, although strides have been made into exotic forms of legged robots [10]. Wheeled mobile systems are useful for practical applications compared with legged systems because of the simplicity of the mechanisms and control systems and efficiency in energy consumption [7]. However, wheeled systems' performance depends on the traction between the wheels and the ground. If there is not enough traction, the wheel will slip and the efficiency will decrease. Traction is a special concern when the robot is expected to move over granular non-cohesive loose soil. Which is the case for planetary missions [8], construction site applications, and agricultural robots, among others.

One of the simplest forms of self-contained localization is wheel odometry, based on wheel encoders that are mounted on a robot to track the number of revolutions each wheel has made. The number of revolutions is integrated into a dynamic model to determine the robot's current position relative to the starting point [27]. But it performs poorly in the presence of wheel slippage, accumulating position error (drift).

Another form of self-contained localization is visual odometry, which operates by incrementally estimating the pose of the vehicle through examination of the changes that motion induces on the images of its onboard cameras [13]. Similarly, laser odometry estimates the ego-motion of a vehicle by scan-matching of consecutive laser scans. Unlike wheel odometry, both visual odometry and laser odometry, are not affected by wheel slip. They have their drawbacks: visual odometry suffers from poor illumination and low textured environments; laser might struggle in degenerated scenes where planar areas are prevalent; both are sensitive to dynamic environments.

Research that improves the robustness of robot localization can be divided into two trends: To fuse information from different sensors to overcome their individual limitations [30, 36, 6] and to make better use of the available sensor information using deep learning and other computationally expensive algorithms [35, 37]. Therefore, robot localization robustness has a price. Whether it is the explicit price of extra sensors or the implicit cost of extra computational resources.

B. Motivation

This work intends to minimize the price of robust ground robot localization in unstructured environments by using sensors that are relatively inexpensive, imply a low overhead, and might be needed for other functionalities of the robot: acoustic odometry. Many ground mobile robots will be equipped with sound sensors for the purpose of human-robot interaction [17], at the same time, audio signals have been extensively studied, resulting in very efficient methods for their processing.

Increasing the robustness of robot localization in unstructured environments without significantly increasing its price would make this technology more accessible. Which complies with the 9th and 10th Sustainable Development Goals: Promotes innovation without increasing the inequalities between small and big producers [40].

C. Related work

Audio-based Odometry is a relatively unexplored field, but robot audition is present in self-localization and navigating tasks using Sound Source localization, where the robot's ego-noise is seen as a nuisance. Moreover, robot-terrain interaction sound has been used in Terrain Classification, which indicates that it carries significant information for environment understanding.

a) Audio-based Odometry: In [20], the authors propose a classification framework to associate ego-noise captured with an onboard microphone to a set of predefined velocity profiles. Additionally, they are able to detect a change in the inclination of the surface the robot is moving. However, the application of this framework is rather limited. On the other hand, [25] proposes a system capable of estimating ground robots' linear and angular velocities using onboard audio sensors. It uses deep neural networks to regress the motion of a vehicle from feature representations (based on Gammatone filterbanks) of the sensed audio. The authors claim that their work demonstrates an absolute error lower than 0.07 m/s and 0.02 rad/s and conclude that audio-based odometry systems should be useful auxiliary sources of odometry on the side of more traditional systems. However, the number of experiments and their evaluation is limited.

b) Sound Source localization: In [24] and [21], the authors propose an algorithm to simultaneously localize a robot and map its environment (SLAM) using onboard audio sensors that perceive sound sources in its environment. [18] restricts the application to an in-pipe robot with a combination of orientation estimates from an inertial measurement unit and traversed distance estimations achieving as well both, self-localization and mapping of the pipeline. The distance is estimated using the time of flight of a reference sound generated with a loudspeaker at the entrance of the pipeline, which is measured with an onboard microphone. Alternatively, [14] perceives the robot's intrinsic noise to localize it using external audio sensors. Combinations with other self-localization methods are proposed in [38], where onboard sound sensors identify and remove the effect of dynamic obstacles for Visual SLAM, and [15], which localizes sounds using onboard microphones and uses them as navigation goals while using Visual SLAM for self-localization.

c) Terrain Classification: Multiple works propose to identify the terrain type of a robot's environment using onboard audio sensors. [26] proposes a deep learning framework, based on a convolutional neural network, that uses only sound from vehicle-terrain interactions to classify a wide range of indoor and outdoor terrains. This method is extended in [31], where an unsupervised classifier that learns from vehicle-terrain interaction sounds supervises a pixel-wise semantic image classifier. Similarly, [34] proposes a multi-modal self-supervised learning technique that switches between audio and image features to cluster terrain types. Extended as well by [33] using a multi-modal variational autoencoder and a Gaussian mixture model clustering algorithm on audio-visual data. It proposes as well to use gammatone-based filtering methods to extract audio features like in [25].

D. Objectives and contributions

The goal of this work is to study the feasibility of an audio-based odometry system, since it has the potential of providing inexpensive robust robot localization in the future, as mentioned in section I-B. More concretely, the objectives of this thesis are:

- To design a system capable of estimating the longitudinal velocity of a wheel driving over loose sandy terrain using only acoustic information.
- To quantitatively evaluate the performance of acoustic odometry against other odometry systems.
- To qualitatively evaluate the feasibility of using audio-based odometry in a real-world situation.

Due to the difficulty in finding publicly available audio odometry datasets, the problem of robot localization is simplified to a single dimension over a unique terrain type: A wheel that can only move along a longitudinal axis over loose sandy terrain. As this scenario can be easily reproduced with the available experimental equipment, making it possible to gather a whole new dataset.

Section II describes the proposed system, including section II-A, where it is described how the data used in this work was gathered, **??**, which explains how that data is processed and section II-C, where one can find details about how processed data is used to predict the longitudinal velocity of a wheeled robot. Section III objectively states its evaluation results with a subjective discussion in section III-C. Finally, section IV summarizes the most important insights from this research as well as recommendations for future research on the field of acoustic odometry.

II. Methods

This work proposes a system capable of estimating motion from audio signals. Shown in figure 1, it is a modular system that consists of two components: a feature extraction module composed of several *extractor* submodules. It receives a multi-channel audio signal of a fixed length and outputs a group of feature vectors, one per extractor submodule; and a prediction

Index	Device
1	Intel RealSense™ Depth Camera D435i
2	Intel RealSense™ Tracking Camera T265
3	RØDE VideoMic™ NTG front
4	RØDE VideoMic™ NTG back
5	RØDE VideoMic™ NTG top
6	RØDE SmartLav+ wheel axis
7	RØDE SmartLav+ top
8	HP Elite Dragonfly built in microphone array

Table I: Devices used in the Wheel Testbed Experiment 2

model that receives a segment composed of a finite number of feature frames and outputs a motion estimation. Between both modules, previous feature frames are stored in memory.

The sections present in this chapter deepen in the different modules that compose the proposed system. Starting with the audio signal, section II-A describes how the data used in this work was gathered. **??** describes the feature extraction module and the segment composition. Finally, section II-C describes the different prediction models used in this work.

A. Experimental Setup

This section describes the different experimental setups used to gather the data needed to build the datasets described in **??**.

1) Wheel Testbed Experiment 2: This final experiment addressed some of the issues found in the sensor setup of **??**.

On one hand, the number of sensors used was increased again to a total of 5 external microphones plus the sensor workstation internal microphone array and two cameras, one of them a tracking camera with built-in visual-based simultaneous localization and mapping (Visual SLAM). The final list of devices can be found in table I while their positioning is shown in figure 2. Their positioning takes into account ideal positions for a microphone attempting to capture the sound generated by the wheel interaction with the terrain (devices 4 and 3), feasible positions for a microphone in a mobile robot (devices 5, 6, and 7), and a reasonable position for a microphone dedicated for human-robot interaction (device 8).

On the other hand, some maintenance and reparations were applied to the wheel testbed. The ball screw shaft was reassembled with tighter screws and grease was applied to the ball bearings. Additionally, the cables were rearranged in a way that they do not appear in the field of view of the camera during the recording.

The experiment is composed of different recordings on the wheel testbed. Recordings are performed under different driving conditions, which are defined by the combination of the following controlled variables: *slip ratio* as defined in **??**. Ranging between -0.3 and 0.6. Being free or uncontrolled slip an additional option; *load* measured in kg added to the base carriage weight (which is 11.2 kg) taking values of 0 kg, 5 kg and 10 kg; *wheel angular velocity* ranging from 5 deg/s to 30 deg/s in steps of 5 deg/s; *contact*, whether the wheel is making contact with the ground or is suspended in the air.

Combinations of said variables resulted in 168 different recordings that make a total of 1 hour for each of the 8 recording devices. It took approximately 13 working hours to perform. The data gathered is used to build the different datasets shown in **??**.

B. Feature extraction

Audio data comes in the shape of vectors of audio samples, one per microphone channel. These vectors contain the audio amplitude for a single time instant. The time between consecutive audio samples is defined by the sampling frequency (or sample rate) of the audio recording.

However individual audio samples are not very informative, the key information lies behind the oscillation of consecutive samples. That's why it is a very common practice to represent audio features in the frequency domain.

The term *extractor* will be used to abstract the system used to extract features from audio data. Although in this work only one type of extractor has been implemented and tested, the system is designed to be generic. Additional feature extractors, like the traditional Mel-frequency Cepstrum Coefficients (MFCC) or a simple Short-Time Fourier Transform, can be easily implemented as an *extractor*. This work opts to use a feature extractor based on Gammatone filterbanks since the performance of classification systems relying on MFCCs is greatly reduced in the presence of noise [25].

a) Gammatone filterbanks: An approximation to the human cochlear frequency selectivity originally introduced in [2]. Time-independent features are obtained by filtering the audio waveform with a bank of gammatone band-pass filters. The impulse response of a gammatone filter centered at frequency f_c is given by equation (1), where n indicates the order of the filter which largely determines the slope of the filter's skirts; and b is the bandwidth of the filter and largely determines the duration of the impulse response; a is the amplitude and ϕ is the phase.

$$g(t, f_c) = at^{n-1}e^{-2\pi bt}\cos 2\pi f_c t + \phi \tag{1}$$

This work implements a bank of fourth order gammatone filters with its corresponding bandwidth b of 1.019 ERB where ERB is the equivalent rectangular bandwidth scale [3]. Said implementation is written in C++ as a Python extension module. It is based on a Matlab MEX function implemented in C by Ma et al. [9], which at the same time is based on Martin Cooke's Ph.D work [4]. The filters are distributed over a predefined frequency range linearly on the ERB scale. The number of filters used is equivalent to the number of features to be extracted.

1) Segmenting: Section II-A described how 168 recordings were gathered from the **??** with different driving conditions. These recordings vary in length but they are all in the order of magnitude of tenths of seconds. However, an odometry estimation model should be able to make new predictions several times per second. Therefore the model should be fed new data several times per second as well.

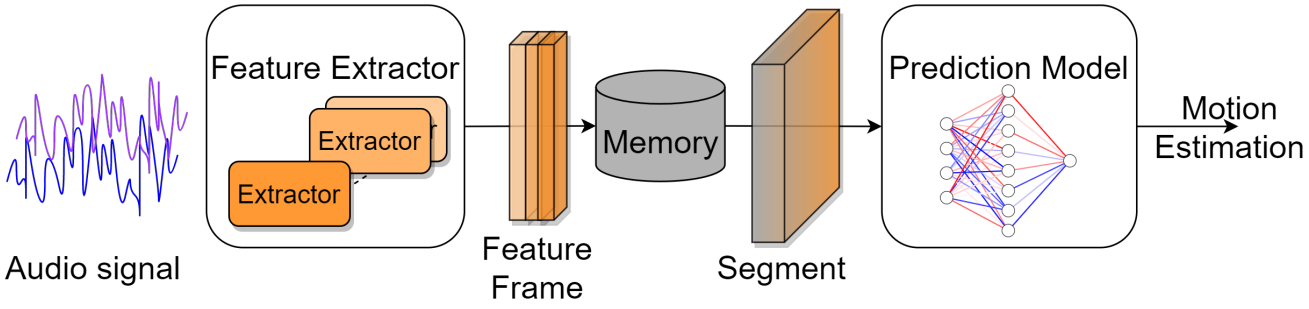
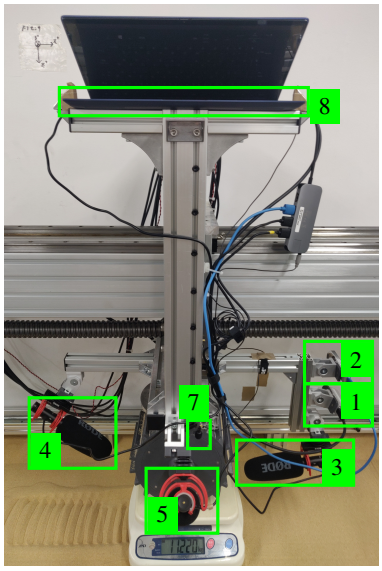
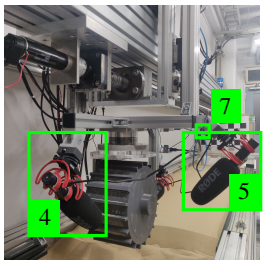


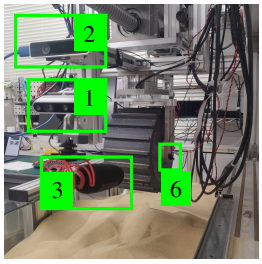
Figure 1: Proposed system



(a) Lateral view



(b) Back view



(c) Front view

Figure 2: Wheel Testbed Experiment 2 setup: *a)* shows a lateral view of the wheel testbed carriage while being weighted. *b)* shows a back view of the wheel and sensor setup while *c)* shows a front view of it.

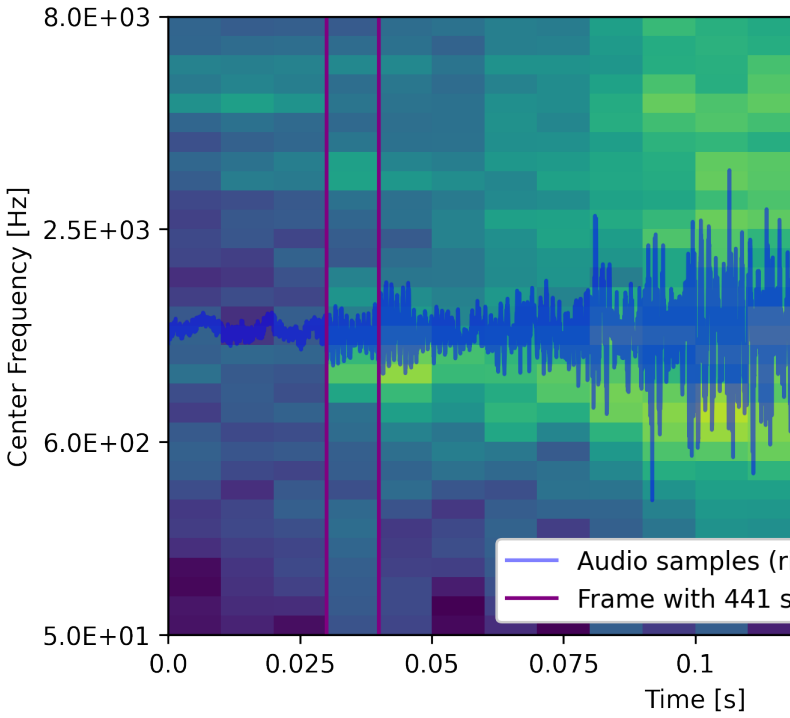


Figure 3: Audio signal (in blue) over its features extracted from frames with a duration of 10 ms with a single Gammatone filterbanks extractor.

The term *frame* is defined as the group of vectors of features extracted from a finite number of samples. One vector per *extractor* (as defined in section II-B) used. Figure 3 illustrates how an audio signal is divided into frames. The number of samples per frame can be tied together with the sample rate of the audio signal to define frames in terms of time. Frame duration is one of the parameters of the dataset as well as the number of features extracted per frame. **??** lists all the different parameters that define a dataset.

The term *segment* is defined as a group of consecutive frames. Segments will be the input for prediction models. Segments can be overlapped in a way that consecutive segments share a percentage of their frames. **??** how the audio frames are grouped into segments that overlap each other.

2) Labeling: *Dataset* is defined in the context of this work as a collection of segments annotated with their corresponding ground truth. The variables of interest are the longitudinal velocity V_x, the wheel angular speed V_ω and the slip ratio s. However, as it has been explained in section II-B1, these segments are relatively large in terms of time. In a way that the measured variables can oscillate significantly during the segment. This work annotates the segments with a weighted average of the measurements within its duration. Consider $x[t]$ the measurement of a given variable x at time t, being t in the set of measurement times $t \in T$. Equation (2) describes how the weighted average of that variable x is computed for

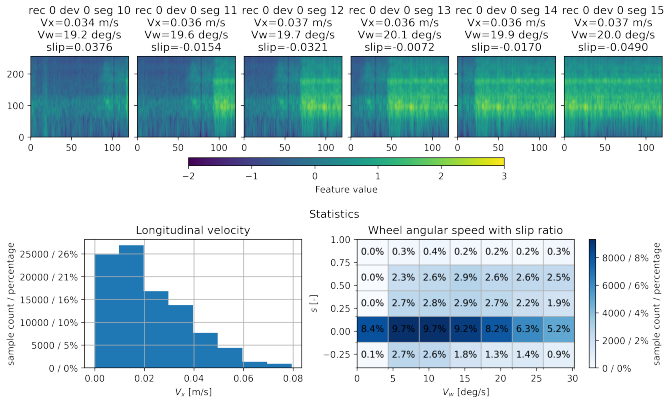


Figure 4: Visualization of consecutive samples from a dataset together with the distribution of the variables of interest. This dataset was processed using all available recordings from Wheel Testbed Experiment 2, a single Gammatone filterbanks extractor with 256 features per frame, a frame duration of 10 ms and a segment length of 100 frames with 80% overlapped frames between consecutive segments.

segment k where t_k is the vector of measurement times inside the segment boundaries and N its size.

$$x_k = \frac{\sum_{i=1}^{N} i x[t_k[i]]}{\sum_{i=1}^{N} i}, \text{ where } i \in \mathbb{N} \tag{2}$$

One can visualize in **??** the segments with their longitudinal velocity V_x annotations computed from the measurements that lay inside the segment boundaries as described by equation (2). Moreover, figure 4 shows consecutive segments from one of the datasets used in this work with their complete annotations. Figure 4 also shows the distribution of the values of the variables of interest in the dataset. This distribution can be altered employing Data augmentation and **??** in order to obtain a more favorable distribution for model training.

3) Data augmentation: The performance of most Machine Learning models depends on the quality, quantity, and relevancy of the training data. Artificially augmenting the data in a dataset can be useful to improve the performance of a model. In this work datasets are augmented by altering the raw recordings before the Feature extraction and Segmenting operations. The term *transform* is defined as the function to be applied to the raw recording. Two transform functions for data augmentation are used in this work:

- Gain: Add a random gain in the range [-10, 10]. Illustrated by the second row of samples in **??**.
- Noise: Add white noise with a random signal-to-noise ratio in the range [0, 1] in the decibel scale. Illustrated by the third row of samples in **??**.

?? illustrates the effect of the data augmentation on the dataset segments. The size of the dataset is essentially multiplied by the number of transformations applied. While this improves the training data quantity, it may decrease its quality.

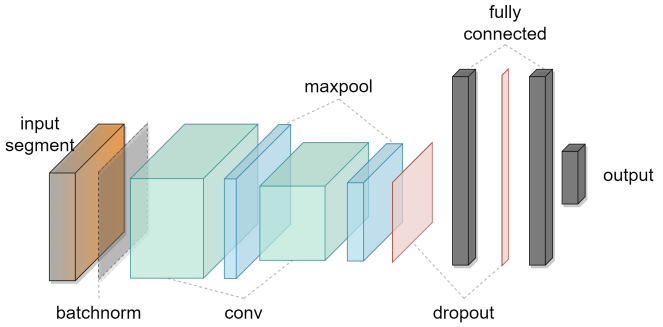


Figure 5: Convolutional Neural Network Architecture used in this work. Different layer sizes are used as well as different input segment dimensions. The output of the last fully connected layer changes with the model Task too.

?? describes the techniques used to split the dataset, controlling the quality of the training data.

C. Models

One can find in this section descriptions of the different model designs used in this work. Three main design choices are taken into account: The dataset used with its **??**; The model Task; and its Architecture. All models are implemented using PyTorch [28].

The main architecture is a shallow Convolutional Neural Network [1] composed by two convolution layers, each of them followed by a max pooling layer, and two fully connected layers, each of them preceded by a dropout, as shown in figure 5 except for the batch normalization layer. The input dimensions depend on the dataset parameters, which take different values for the evaluation. Different layer sizes are evaluated as well, defined in table II. The output of the last fully connected layer depends on the Task.

1) Task: One can find here a description of the different tasks implemented and tested in models. Tasks define the goal of the model and the way its loss is computed.

a) Classification: Consists in classifying the longitudinal velocity given a set of possibilities. The different classes are ranges of longitudinal velocities, being these ranges a hyperparameter of the model. Cross entropy loss is computed between the predicted class probabilities and the class corresponding to the target longitudinal velocity and the predicted class. The output of the model is therefore a vector of probabilities corresponding to each class.

b) Ordinal classification: Consists in classifying the longitudinal velocity given a set of possibilities like in Classification, with different classes being ranges of longitudinal velocities. But the order of the class matters. This method was introduced in [5], where standard classification algorithms are extended to make use of the order of the classes. The output of this model is a vector of binary values that can be decoded into a class position by making use of a ranking rule. The loss is computed with the mean square error between the target class position and the predicted class position.

Name	Conv 1		Conv 2		Hidden units	Size [MB]
	Filters	Kernel	Filters	Kernel		
base	64	5	128	5	512	412.6
M	32	5	64	5	256	103.2
S	16	5	32	5	256	51.5
XS	8	5	16	5	128	12.9
XXS	4	3	8	3	64	3.4
XXXS	2	3	4	3	32	0.87

Table II: Different sizes used for the model Architecture in this work. *Conv 1* corresponds to the first convolutional layer, *Conv 2* to the second. *Hidden units* corresponds to the units between the two last fully connected layers. The input of the first fully connected layer is determined by the output elements of the second convolutional layer while the output of the second fully connected layer is determined by the model Task. Sizes in megabytes correspond to models with input segments of 120 frames with 256 features from 1 extractor and classification output with 7 classes.

2) Architecture: This section describes the different model architectures implemented and evaluated. A common point of them all is simplicity. It is out of the scope of this work to find an optimal architecture for acoustic odometry. But it is interesting to evaluate different simple options.

a) CNN with normalized input: This architecture is identical to the **??** except for the fact that it contains a batch normalization layer [16] as shown in figure 5.

III. Results

This evaluation is computed using recordings never available as training data. The training pipeline is deterministic, meaning that all randomly generated numbers follow the same seed unless otherwise specified. Nevertheless, when the frame duration or the segment length is changed, the produced datasets have a different number of samples. Comparing the performance of models trained using datasets with different sizes can lead to wrong conclusions.

A. Metrics

This work measures the performance of the trained models using commonly used odometry and simultaneous localization and mapping metrics: [29].

B. Selected model

After the experiments listed in **??** this work selects a model trained on a dataset of segments of 50 frames, each of them spans over 15 ms and has 64 features from a single Gammatone filterbanks extractor. The extractor is applied on the average of the audio signal channels with a frequency range of [50, 80000] Hz and features on te Bel scale. Training data from all available recordings is selected using the `with-laptop` strategy split described in **??**, a Ordinal classification task with 28 different linearly distributed longitudinal velocity ranges and a CNN with normalized input architecture with a small size described as `S` in table II.

	Mean ATE [m]	Mean RPE [m/s]
Acoustic Odometry	$1.02e-03$	$5.10e-03$

Table III: Selected model average performance across all evaluation recordings and all devices. **??** is computed between frames with a duration of 15 ms and **??** is computed using time windows 1s long.

Table III shows the average performance of the selected model across all seven evaluation recordings and devices. Figure 6 shows the its results on an evaluation recording characterized by high slippage driving conditions using two different devices.

a) Noise: One can see in figure 7 the behavior of the selected model against white noise. Generally, white noise increases the value of the estimated motion. Very small values of signal-to-noise ratio (-20 dB, which is a noise power 100 times higher than the signal) impede the model to recognize situations where the wheel is not moving.

b) Computational cost: Tests with the selected model on a CPU show that the time to compute the feature extraction is a 40% of the total time to process a new frame, which is 2.03 ms on an Intel® Core™ i7-9750H CPU at 2.60GHz. The prediction time takes up a 55% of the total time while the remaining 5% corresponds to loading the new frame into memory The system is able to run 7.5 times faster than real time on this particular hardware. Using hardware acceleration with CUDA the total time to process a new frame is slightly lower: 1.93 ms on a NVIDIA GeForce RTX 2060 GPU.

c) Compared with other models: The selected model is evaluated against two other odometry methods: Wheel odometry computed from the ground truth wheel angular speed; Visual SLAM using Intel® RealSense™ Tracking Camera T265, which is based on visual odometry. Figure 8 is the evaluation in an undisturbed scenario for all methods, no acoustic noise and no lightning changes or dynamic objects. It shows that the selected model can perform with an accuracy comparable to state-of-the-art commercial visual-based methods in this simplified scenario. Figure 9 instead corresponds to a scenario that is challenging for visual odometry. Dynamic objects are moved in the camera field of view and lightning conditions are changed during the recording. One can see in this evaluation that the vulnerabilities of audio-based odometry and visual-based odometry do not overlap. Which makes Acoustic Odometry more accurate in certain scenarios where the visual odometry vulnerabilities are exploited.

C. Discussion

One can find in this section insights extracted from the presented results. First of all, the proposed Acoustic Odometry system proves to be a viable auxiliary source of odometry for wheeled robots in loose sandy terrain. Figure 8 and figure 9 show that audio-based odometry is more accurate than wheel odometry, and its accuracy can be comparable to state-of-the-art commercially available visual-based methods and that it

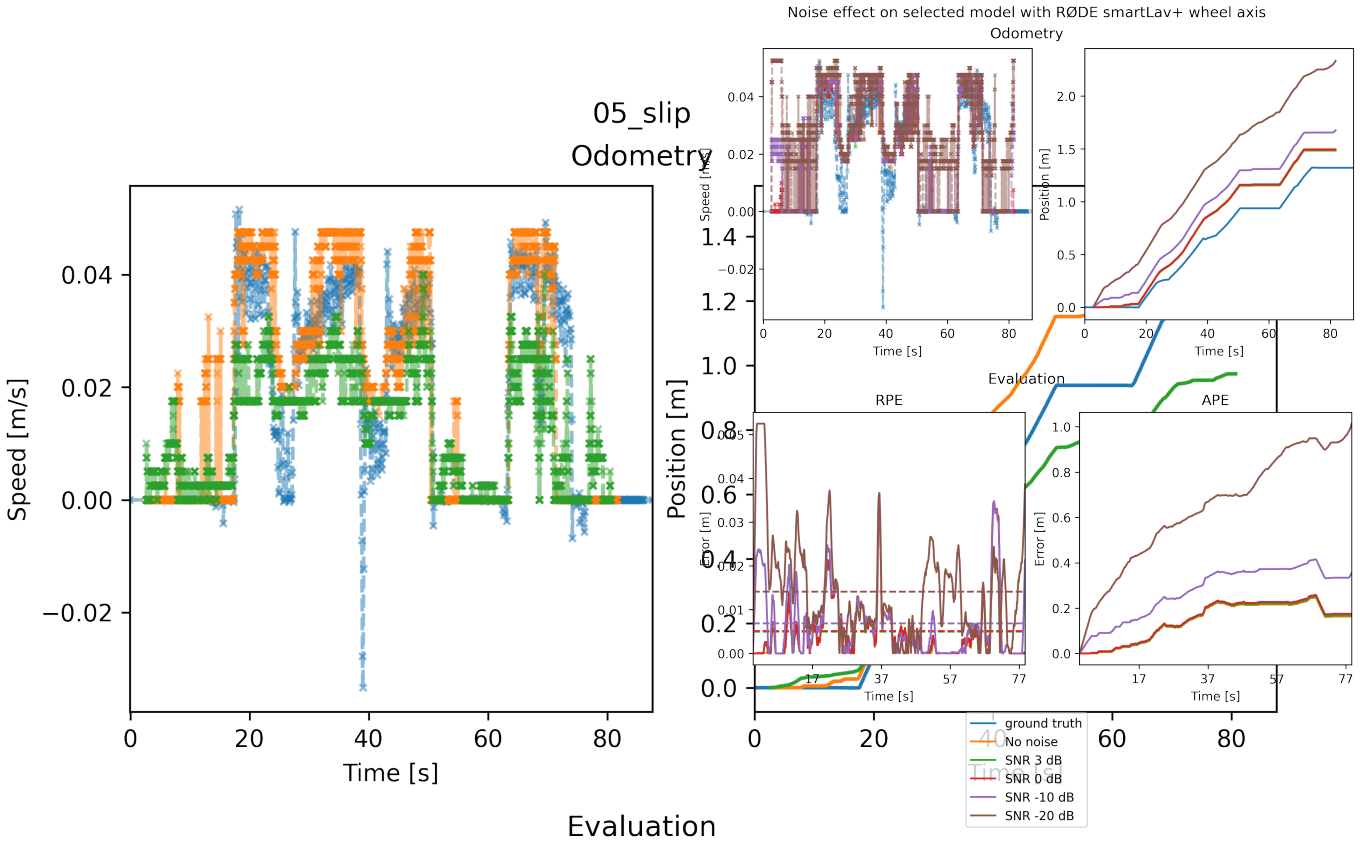


Figure 7: Selected model evaluated on a recording with added white noise with varying signal-to-noise Ratio values.

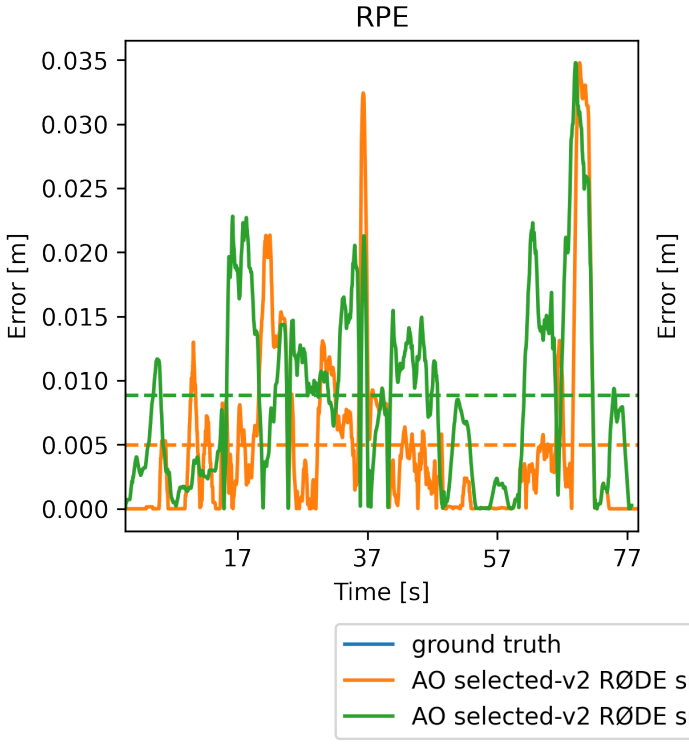


Figure 6: Selected model evaluated on a recording characterized by high slippage conditions using two different devices.

can be more accurate than visual odometry in the presence of dynamic objects and lightning.

Secondly, the selected model behavior changes from one device to another. Devices that output an audio signal with a higher power tend to result in higher speed predictions, as can be seen in figure 6, while the overall predicted curve is very similar. Meaning that even if the absolute value of the predicted velocity is not accurate, the model is able to recognize slippage conditions across different devices. Even with devices not present during training, as is the case in figure 8, where none of the devices shown in the evaluation were present during training.

Thirdly, white noise affects the performance of the proposed system, as could be expected. Nevertheless, a considerably large noise power compared to the signal power does not make the selected model unusable. As it can be seen in figure 7, where a noise with a power 10 times the one of the audio signal (SNR -10 dB) only significantly affects the predictions on speeds close to 0. This indicates that white noise only significantly affects the prediction of speeds under a threshold determined by the signal-to-noise power ratio.

Finally, this work can be compared with other odometry methods. On one hand, only [25] proposes an audio-based odometry method, claiming an average absolute error of 0.065 m/s when predicting velocities, which is comparable to the

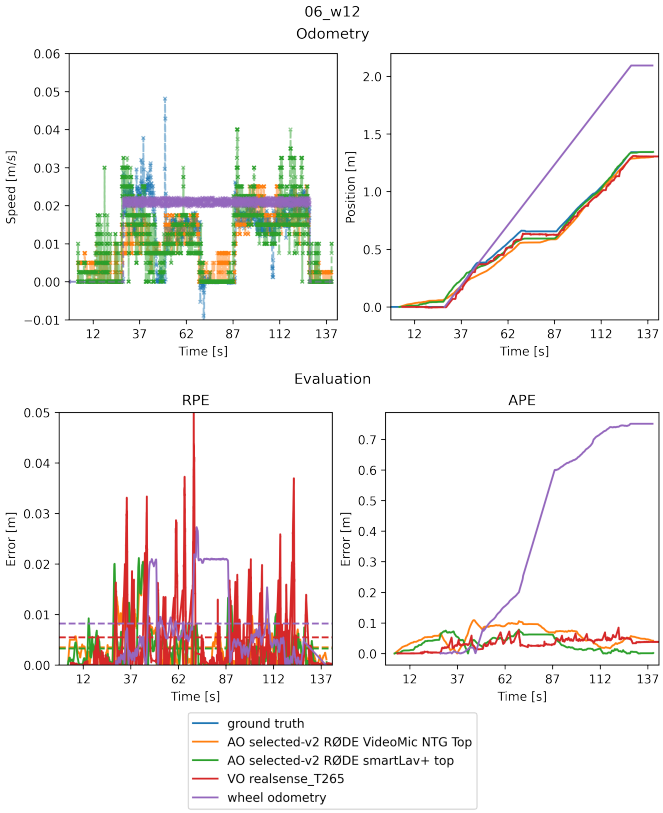


Figure 8: Selected model evaluated evaluated against Visual SLAM (based on visual odometry) and wheel odometry.

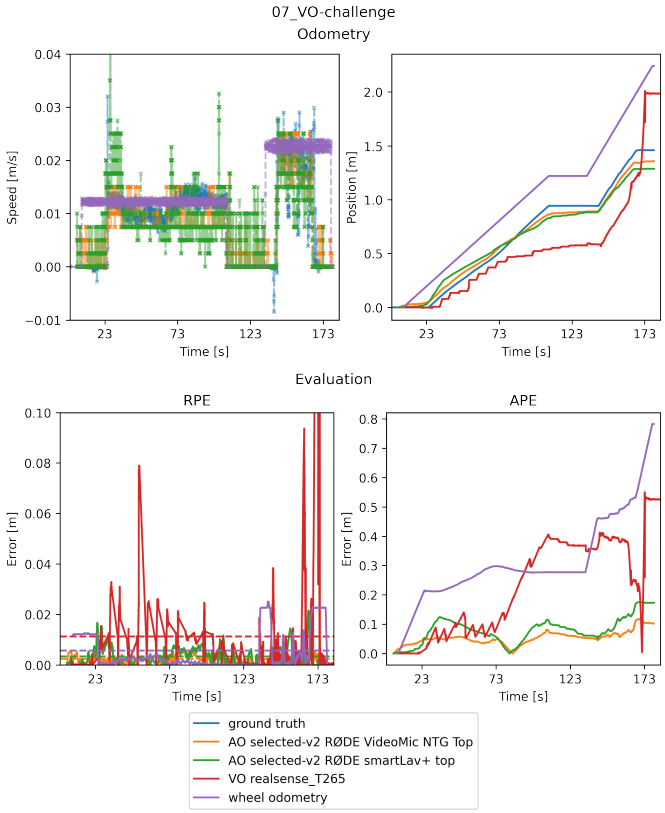


Figure 9: Selected model evaluated against Visual SLAM (based on visual odometry) in a scenario where visual odometry vulnerabilities are exploited: Dynamic objects are moved in the camera field of view and lightning conditions are changed during the recording.

average **??** 0.001 m accumulated over the 66 15 ms long audio frames contained in one second. But this work also shows that the actual average **??** in a 1 second window is 0.005 m. However, it remains unknown the performance of the proposed system on rotational movement and under the presence of multiple moving wheels. The comparison should also be taken with a grain of salt as no common benchmarking data exists.

On the other hand, the method presented in [6], where wheel slippage is identified using the motor current and compensated in the wheel odometry computation, still outperforms the proposed system while being a computationally inexpensive method. They demonstrate accumulated drift of up to 1% of the traveled distance using the same range of longitudinal velocities under 0.07 m/s, while the proposed system results in an average error of 16% of the traveled distance across all devices and evaluation recordings. On the other hand, the authors mention that current-based slippage detection fails to correctly identify a wheel slipping over rocks instead of sand. Audio-based methods do have the potential to identify wheel slippage in both scenarios and **??** IV-0a discusses several improvements that could be applied to the proposed system in order to improve its performance.

IV. Conclusions

This work demonstrates that it is feasible to estimate odometry from acoustic data with a computationally inexpensive system. Many future ground mobile robots will be equipped with audio sensors for human-robot interaction purposes. In that context, robot ego-noise is just noise. Being able to estimate motion using byproduct data from the human-robot interaction system can add robustness to the localization system without significantly altering the hardware cost with additional sensors.

In this thesis, we propose a system that can estimate the longitudinal velocity of a wheeled robot on loose sandy terrain using gammatone-based features extracted from robot ego-noise data and an ordinal classification model implemented as a convolutional neural network. Said system has been trained and evaluated in a new multi-modal dataset collected using a single wheel testbed and several microphones and cameras. The evaluation contains high wheel slippage scenarios where the proposed system successfully identifies when the wheel slips. The proposed model is also evaluated in the presence of white noise and demonstrates that it still can successfully predict longitudinal velocities in the presence of high noise power. Moreover, the system is compared against wheel odometry and

a commercially available visual simultaneous localization and mapping system satisfactorily.

a) Recommendations: This work shows that performance does not change significantly with model size. This indicates that there is still plenty of room for improvement in terms of model architecture: One one side, machine learning may not even be needed, a simpler algorithm may be used with sufficient accuracy and lower computational cost. On the other side, the fact that the behavior of the selected model changes significantly between devices indicates that fine-tuning [22] a model with the device where it will be deployed might significantly increase its performance. Similarly, using wheel odometry to estimate the wheel angular speed while using the proposed system to identify the wheel slippage may improve the performance as well.

From the model point of view, this work only evaluates the performance of a shallow convolutional neural network with different layer sizes. But other architectures be more suitable for the task. Specially transformers [23] which have been used in speech recognition [39] and visual odometry tasks [32].

Finally, in recent years, single-robot simultaneous localization and mapping research is steadily moving toward systems that can build metric-semantic maps. Acoustic odometry could be combined with terrain classification in order to provide both, an auxiliary source of odometry and semantic information for a simultaneous localization and mapping system. Similarly, a system that not only estimates motion based on ego-noise but also is able to identify and subtract said ego-noise from the audio signal would be useful to improve the performance of speech recognition in collaborative ground robots.

References

[1] K. Fukushima. "Neural network model for a mechanism of pattern recognition unaffected by shift in position - Noecognitron". In: *Trans. IECE* A-(10).J62 (1979), pp. 658–665.

[2] J. Holdsworth et al. "Implementing a gammatone filterbank". In: *Annex C of the SVOS Final Report: Part A: The Auditory Filterbank* 1 (1988), pp. 1–5.

[3] Brian R Glasberg and Brian C.J Moore. "Derivation of auditory filter shapes from notched-noise data". In: *Hearing Research* 47.1 (1990), pp. 103–138. ISSN: 0378-5955. DOI: https://doi.org/10.1016/0378-5955(90)90170-T.

[4] Martin Cooke. "Modelling auditory processing and organisation". In: *Distinguished dissertations in computer science*. 1993.

[5] Ling Li and Hsuan-tien Lin. "Ordinal Regression by Extended Binary Classification". In: *Advances in Neural Information Processing Systems*. Ed. by B. Schölkopf, J. Platt, and T. Hoffman. Vol. 19. MIT Press, 2006.

[6] L. Ojeda et al. "Current-Based Slippage Detection and Odometry Correction for Mobile Robots and Planetary Rovers". In: *IEEE Transactions on Robotics* 22.2 (2006), pp. 366–378. DOI: 10.1109/TRO.2005.862480.

[7] Masayoshi Wada. "Studies on 4WD Mobile Robots Climbing Up a Step". In: *2006 IEEE International Conference on Robotics and Biomimetics*. 2006, pp. 1529–1534. DOI: 10.1109/ROBIO.2006.340156.

[8] Faiz Ben Amar et al. "Towards an advanced mobility of wheeled robots on difficult terrain". In: *International Journal of Factory Automation, Robotics and Soft Computing* hal-03135894 (2007), pp. 40–45.

[9] Ning Ma et al. "Exploiting correlogram structure for robust speech recognition with multiple speech sources". In: *Speech Communication* 49 (Dec. 2007), pp. 874–891. DOI: 10.1016/j.specom.2007.05.003.

[10] Gary Boucher and Luz Maria Sanchez. "Mobile Wheeled Robot with Step Climbing Capabilities". In: *Mobile Robots*. Ed. by XiaoQi Chen, Y.Q. Chen, and J.G. Chase. Rijeka: IntechOpen, 2009. Chap. 3. DOI: 10.5772/6988. URL: https://doi.org/10.5772/6988.

[11] X.Q. Chen, Y.Q. Chen, and J.G. Chase. "Mobiles Robots - Past Present and Future". In: *Mobile Robots*. Ed. by XiaoQi Chen, Y.Q. Chen, and J.G. Chase. Rijeka: IntechOpen, 2009. Chap. 1. DOI: 10.5772/6986. URL: https://doi.org/10.5772/6986.

[12] Merriam-Webster.com. *robot*. 2011. URL: https://www.merriam - webster . com / dictionary / robot (visited on 07/25/2022).

[13] Davide Scaramuzza and Friedrich Fraundorfer. "Visual Odometry [Tutorial]". In: *IEEE Robotics and Automation Magazine* 18.4 (2011), pp. 80–92. DOI: 10.1109/MRA.2011.943233.

[14] Brian F. Allen et al. "Localizing a mobile robot with intrinsic noise". In: *3DTV-CON 2012* hal-00732764 (Oct. 2012).

[15] Gautam Narang, Keisuke Nakamura, and Kazuhiro Nakadai. "Auditory-aware navigation for mobile robots based on reflection-robust sound source localization and visual SLAM". In: *2014 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. 2014, pp. 4021–4026. DOI: 10.1109/SMC.2014.6974560.

[16] Sergey Ioffe and Christian Szegedy. *Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift*. 2015. DOI: 10.48550/ARXIV.1502.03167. URL: https://arxiv.org/abs/1502.03167.

[17] Matthew R. Walter et al. "A Situationally Aware Voice-commandable Robotic Forklift Working Alongside People in Unstructured Outdoor Environments". In: *Journal of Field Robotics* 32.4 (2015), pp. 590–628. DOI: https://doi.org/10.1002/rob.21539. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/rob.21539. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/rob.21539.

[18] Yoshiaki Bando et al. "Sound-based online localization for an in-pipe snake robot". In: *2016 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*. 2016, pp. 207–213. DOI: 10.1109/SSRR.2016.7784300.

[19] Shoudong Huang and Gamini Dissanayake. "Robot Localization: An Introduction". In: *Wiley Encyclopedia*

of Electrical and Electronics Engineering. John Wiley and Sons, Ltd, 2016, pp. 1–10. ISBN: 9780471346081. DOI: https://doi.org/10.1002/047134608X.W8318. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/047134608X.W8318.

[20] Antonio Pico et al. "How do I sound like? forward models for robot ego-noise prediction". In: *2016 Joint IEEE International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)*. 2016, pp. 246–251. DOI: 10.1109/DEVLRN.2016.7846826.

[21] Daobilige Su et al. "Robust sound source mapping using three-layered selective audio rays for mobile robots". In: *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2016, pp. 2771–2777. DOI: 10.1109/IROS.2016.7759430.

[22] Karl Weiss, Taghi M. Khoshgoftaar, and DingDing Wang. "A survey of transfer learning". In: *Journal of Big Data* 3 (May 2016), p. 9. DOI: 10.1186/s40537-016-0043-6.

[23] Ashish Vaswani et al. *Attention Is All You Need*. 2017. arXiv: 1706.03762 [cs.CL].

[24] Christine Evers and Patrick A. Naylor. "Acoustic SLAM". In: *IEEE/ACM Trans. Audio, Speech and Lang. Proc.* 26.9 (Sept. 2018), pp. 1484–1498. ISSN: 2329-9290. DOI: 10.1109/TASLP.2018.2828321. URL: https://doi.org/10.1109/TASLP.2018.2828321.

[25] L Marchegiani and P Newman. "Learning to listen to your ego-(motion): Metric motion estimation from auditory signals". In: Institute of Electrical and Electronics Engineers, 2018.

[26] Abhinav Valada, Luciano Spinello, and Wolfram Burgard. "Deep Feature Learning for Acoustics-Based Terrain Classification". In: *Robotics Research: Volume 2*. Ed. by Antonio Bicchi and Wolfram Burgard. Cham: Springer International Publishing, 2018, pp. 21–37. ISBN: 978-3-319-60916-4. DOI: 10.1007/978-3-319-60916-4_2. URL: https://doi.org/10.1007/978-3-319-60916-4_2.

[27] Sherif A. S. Mohamed et al. "A Survey on Odometry for Autonomous Navigation Systems". In: *IEEE Access* 7 (2019), pp. 97466–97486. DOI: 10.1109/ACCESS.2019.2929133.

[28] Adam Paszke et al. "PyTorch: An Imperative Style, High-Performance Deep Learning Library". In: *Advances in Neural Information Processing Systems 32*. Curran Associates, Inc., 2019, pp. 8024–8035. URL: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

[29] David Prokhorov et al. *Measuring robustness of Visual SLAM*. 2019. DOI: 10.48550/ARXIV.1910.04755. URL: https://arxiv.org/abs/1910.04755.

[30] Michelle Valente, Cyril Joly, and Arnaud de La Fortelle. *Deep Sensor Fusion for Real-Time Odometry Estimation*. 2019. DOI: 10.48550/ARXIV.1908.00524. URL: https://arxiv.org/abs/1908.00524.

[31] Jannik Zürn, Wolfram Burgard, and Abhinav Valada. "Self-Supervised Visual Terrain Classification from Unsupervised Acoustic Feature Learning". In: (2019). DOI: 10.48550/ARXIV.1912.03227. URL: https://arxiv.org/abs/1912.03227.

[32] Xiangyu Li et al. *Transformer Guided Geometry Model for Flow-Based Unsupervised Visual Odometry*. 2020. arXiv: 2101.02143 [cs.CV].

[33] Reina Ishikawa et al. "Single-modal Incremental Terrain Clustering from Self-Supervised Audio-Visual Feature Learning". In: *2020 25th International Conference on Pattern Recognition (ICPR)*. 2021, pp. 9399–9406. DOI: 10.1109/ICPR48806.2021.9412638.

[34] Akiyoshi Kurobe et al. "Audio-Visual Self-Supervised Terrain Type Recognition for Ground Mobile Platforms". In: *IEEE Access* 9 (2021), pp. 29970–29979. DOI: 10.1109/ACCESS.2021.3059620.

[35] Ran Long et al. "RigidFusion: Robot Localisation and Mapping in Environments With Large Dynamic Rigid Objects". In: *IEEE Robotics and Automation Letters* 6.2 (Apr. 2021), pp. 3703–3710. DOI: 10.1109/lra.2021.3066375. URL: https://doi.org/10.1109%2Flra.2021.3066375.

[36] Elizabeth Vargas et al. "Robust Underwater Visual SLAM Fusing Acoustic Sensing". In: *2021 IEEE International Conference on Robotics and Automation (ICRA)*. 2021, pp. 2140–2146. DOI: 10.1109/ICRA48506.2021.9561537.

[37] Huangying Zhan et al. *DF-VO: What Should Be Learnt for Visual Odometry?* 2021. DOI: 10.48550/ARXIV.2103.00933. URL: https://arxiv.org/abs/2103.00933.

[38] Tianwei Zhang et al. *AcousticFusion: Fusing Sound Source Localization to Visual SLAM in Dynamic Environments*. 2021. DOI: 10.48550/ARXIV.2108.01246. URL: https://arxiv.org/abs/2108.01246.

[39] Sehoon Kim et al. *Squeezeformer: An Efficient Transformer for Automatic Speech Recognition*. 2022. arXiv: 2206.00888 [eess.AS].

[40] United Nations. *Sustainable Development Goals*. URL: https://sdgs.un.org/goals (visited on 07/25/2022).