EINDHOVEN UNIVERSITY OF TECHNOLOGY

Department of Mathematics and Computer Science
Interconnected Resource-aware Intelligent Systems Research Group

Towards domain agnostic Wi-Fi CSI gesture classification

Master's Thesis

Yvan Putra Satyawan

Supervisors:
Prof. Dr. Ir. Nirvana Meratnia
Your second Committee Member
Your Third Committee Member

Draft version

Eindhoven, February 2023

Abstract

THIS IS MY ABSTRACT

Abstract needs to be completed

Preface

We choose to write this thesis. We choose to write this thesis, in this semester and do the other things, not because they are easy, but because they are hard.

actually write a proper preface

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In this ever more connected climate that we find ourselves in, IoT devices everywhere are adding little conveniences to our every day lives. With IoT devices becoming ever more common and reaching a forecasted 27 billion devices by 2025 [9], the dream of ubiquitous computing and sensing is transitioning from a mere dream to the reality of our every day lives. Additionally, approximately 19% of new devices bought in 2020 also utilize some form of Wi-Fi radio for communications with a forecasted increase to 24% by 2025.

It is clear that this trend towards integrating computing technology into everyday objects will only accelerate in the future. The increased convenience and efficiency may be the biggest boon of such technologies. For example, smart thermostats can predict heating requirements and adjust accordingly, leading to lower heating costs in a house while maintaining the convenience of having a well heated space.

With all these connected devices becoming and edge computing ability comes ubiquitous sensing, enabling new modalities of interaction and improving data collection and analytics. It is now possible to envision households with complete presence detection coverage,

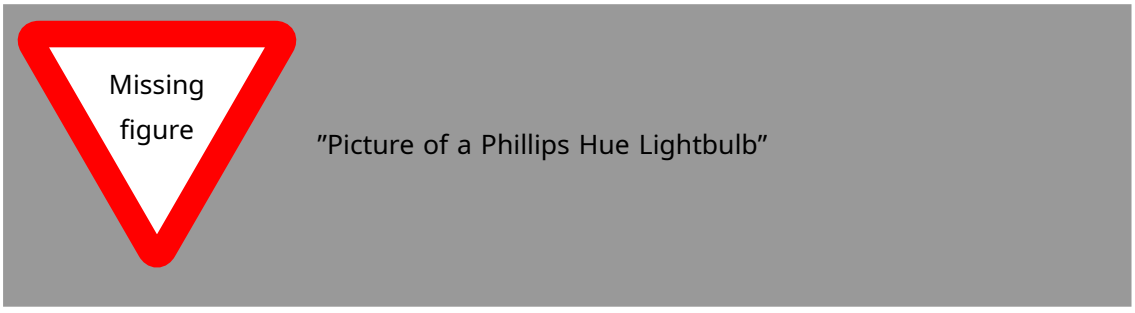

Figure 1.1: A Phillips Hue Lightbulb, one example of an IoT smart lightbulb with an integrated Wi-Fi Radio

for example, through the use of smart motion sensors, enabling increased efficiency by intelligently identifying which rooms require heating and lighting and which do not. Always-on voice control systems, such as Amazon Alexa and Google Assistant speakers are also increasingly common, making a connected AI-assistant only one call away. The always-connected nature of these sensors also enable the gathering and analysis of vast amounts of user data, potentially providing valuable insights into various aspects of our lives.

One challenge that has continued to plague ubiquitous devices is the lack of a ubiquitous user interface which does not require input devices. It is not too common to use the end device itself as the input. Even in the case of voice control systems, a dedicated smart speaker is still required and must be placed in every room from which interaction is desired. For example, in the case of smart lights or smart thermostats, the end device would be the light bulbs and heating system, respectively. In both cases, a separate control unit, the light switch or thermostat, respectively, is still required.

With these challenges in mind, Wi-Fi-based sensing provides one potential solution. Many IoT devices already contain some sort of Wi-Fi radio, such as the Phillips Hue Light bulb in Figure 1.1, and it would be a rather safe assumption to make that spaces with IoT devices would also have some sort of Wi-Fi infrastructure in place as well. With this in mind, the idea of a gesture-based interface based around Wi-Fi becomes rather appealing. The use of Channel-State Information (CSI) also enables of finer-grain signals to be extracted from consumer-grade Wi-Fi radios, enough to enable reliable gesture recognition [1]. This method does, though, suffer from the domain-shift problem, achieving the best accuracy only in cases with a prediction model fine-tuned to a specific person and environment is used.

With this thesis, we aim to explore the use of CNN architectures and domain-shift mitigation methods to improve the state-of-the-art in Wi-Fi CSI-based gesture classification. Specifically, we will look at using preprocessing methods to transform the input signal from CSI into an image using table-to-image and signal-to-image transformations, the use of traditional signal processing algorithms to process the incoming CSI signal, and the use of Reinforcement Learning (RL) to perform domain auto-labeling and provide the CNN classifier with additional information.

1.1 Context and Background

IoT devices are without a doubt increasingly prevalent in everyday life. The Atlas building at the Technical University of Eindhoven (TU/e), for example, uses centrally networked lights for all of its lighting fixtures powered through Power over Ethernet (PoE) and this is at least

partially credited as a reason the building has the best efficiency of any academic building in Europe when it was constructed [7].

All over the building, presence detection, in the form of motion detection sensors, is also used to automatically set appropriate lighting levels for each room. This is just one example of how ubiquitous computing and sensing has now entered the mainstream and is no longer a dream of a few enthusiasts and IoT evangelists. Developments in AI and big data processing has also made the usefulness of ubiquitous computing much more evident, legitimizing its use in everyday objects.

Finally, the deployment of 5G networks in densely populated areas is working towards enabling faster speeds and lower latency, essential in many ubiquitous computing and sensing applications.

With these advances, the question has shifted towards what sort of interface we should utilize to provide an always-available non-intrusive experience for the users. One possible solution is gesture-based interfaces. With any ubiquitous computing and sensing product, especially in the consumer space, minimal setup on the user's part is desired; otherwise the product will not become something which is widely accepted and used.

Wi-Fi gesture recognition can solve these issues, providing a gesture-based interface requiring potentially zero additional setup requirements. As a bonus, this would also be a low-cost solution which many IoT devices already having the necessary hardware for regardless.

There also exists a task group for Wireless Local Area Network (WLAN) sensing, called 802.11bf, within the IEEE 802.11 working group, the group which sets the standards for WLAN, with members from large companies including Huawei, Qualcomm, and Meta [6]. This shows there is genuine interest in the industry to utilize WLAN for these purposes. With an approval date set for September 2024, it is clear that WLAN sensing is not just some theoretical possibility confined to a lab, but rather a very real technology that may soon become widespread.

To make such an approach possible, we utilize machine learning (ML) to process the incoming CSI data and classify user gestures. Wi-Fi technology, when boiled down, is just a really complex radio and what is radar but a different form of very complex radio. It naturally, or not so naturally, follows then, can Wi-Fi be used for remote sensing analogously to radar technology? The answer to this question, according to [1] and [5], the answer is a resounding yes!

However, ML suffers from degraded performance when faced with domain-shifts. When dealing with new, unseen users and environments, gesture classification accuracy degrades significantly. As such, factors to mitigate this domain-shift problem are required in any citation needed

implementation outside of a pristine laboratory setting.

For the purposes of our thesis, the Wi-Fi information we utilize for gesture classification is known as the Channel State Information (CSI). This comes in the form of two signals, an amplitude and phase shift, for each receiver access point (AP) from each transmitting AP. CSI itself is a description of the multipath effects of a signal traveling from the transmitting AP to the receiving AP. In the realm of WLAN, the estimated CSI of incoming signals is used to correct for these multipath effects, making it possible for the system to adapt to current environmental conditions. Importantly, human activity in an environment also affects the CSI, making it possible to infer activity through CSI.

1.2 Motivation

There are significant potential commercial and practical benefits to enabling domain-agnostic Wi-Fi gesture classification.

From a commercial perspective, the 802.11bf standard, which standardizes the hardware technology required for Wi-Fi sensing, is on the cusp of being released. If we are to take advantage of these new technologies, it is imperative that its practical applications be researched.

I don't like the wording here

Wi-Fi sensing is also unique in that the hardware for it is already commercially available and widespread, though not necessarily standardized. As such, it is in a unique place where it is an almost completely software-based solution for preexisting, mass-adopted hardware enabling completely new modes of interaction. This makes it unique and, ultimately, much more commercially desirable as a technology to be adopted.

With respect to domain-agnosticism, the motivating factor is that the largest hurdle for Wi-Fi gesture classification is its loss of performance in new, unseen domains. A model that cannot adapt to such issues will inevitably be nonviable for commercial adaptation lest the end-user be required to perform the same gesture hundreds of times in various positions after installing every single smart IoT device.

We also wish to further the state-of-the-art in domain-agnostic models. The results of this thesis is not only applicable to the field of Wi-Fi gesture classification. Domain-agnostic systems are required for many industry applications, such as [what]. The conclusions of this thesis will hopefully be able to advance the field and provide new insight into what future avenues of research may end up being fruitful.

Figure out what

Finally, a Sci-Fi future where all your devices are controlled through your thoughts might seem like a dream, but in reality, we are not too far from this future. The use of Wi-Fi sensing to detect gestures is one step closer to this futuristic dream and in addition to all the poten-

tial commercial and practical benefits this technology could bring, the sheer "coolness" of the technology should not be underestimated as a motivation to perform research.

A bit less academic, but hopefully just as viable as a motivation

1.3 Problem Statement

Models already exist to classify gestures through Wi-Fi CSI data. Ideally models are solely influenced by those factors which contribute to the correct classification of the gesture, this is, in reality, not the case. These models are influenced by "domain factors" such as the subject performing the gesture and the environment where this is taking place.

Replace "this is taking place" with something else

These domain factors cause feature domain shifts between the training data and the data encountered during actual use or inference despite both domains containing the phenomenon of interest, for example the gesture we wish to classify. Due to these shifts, performance of the model is degraded. It is thus interesting to be able to create a domain-agnostic model whose output is independent of the aforementioned domain factors.

citation needed

There exist proposals to mitigate this degradation including using very large datasets. This is one method especially championed by large companies such as Tesla and OpenAI which have vast resources at their disposal. However, the gathering of such large datasets is only feasible for specific scenarios. In the case of Tesla, for example, having a large fleet of vehicles capable of recording what is essentially supervised training data makes it possible to collect the vast amounts of data required to build a dataset usable for the training of self-driving vehicles. Various OpenAI projects, on the other hand, simply scrape large amounts of websites and collect these as part of their dataset. Within the scope of our research, due to its nature, such widespread data collection methods are non-viable, and we must resort to more novel approaches towards data-agnosticism.

specific citation needed

specific citation needed

Most alternative methods to simply using very large datasets rely on a ground truth domain label being provided [13, 25]. In these approaches, a "discriminator" network is used to predict domain labels from the latent embedding of the data and an adversarial training procedure is then used. The "generator" which produces these latent embeddings must thus generate embeddings which contain no domain information while still maintaining enough information that a classifier model can use its output to correctly classify target features.

In [14], a method is presented which does not require manually labeled domain labels to be provided. Instead, a CNN encoder and state machine neural network are used and their output is fed into a recurrent neural network (RNN) to provide a classification. The RNN is trained through reinforcement learning (RL) to predict features correctly and independently of domain factors.

We hypothesize that we can extend the RL component of [14] to facilitate domain auto-

labeling and eliminate the use of a state machine neural network by using signal-to-image preprocessing methods. Towards these goals, we specify our research questions as follows:

1. To what extent can a reinforcement learning agent utilize the latent signal representation produced by the CNN to accurately produce a latent representation of the domain space, measured by performance metric difference in a domain factor leave-out cross validation setting between a classifier provided the domain space representation and one which has not?

2. To what extent would changing the latent representation of the domain space from a one-hot encoding to a probability measure affect the performance of the classifier, measured by the performance metric difference in a domain factor leave-out cross validation between both domain space representation types?

3. To what extent can signal-to-image transformation replace the state machine neural network presented in [14], measured by comparing the performance metric difference in a domain factor leave-out cross validation setting between the model presented in [14] and our approach provided no self-label?

The rest of this thesis will present a literature review, background on required knowledge, the proposed methodology, experimental results, and discussions and conclusions of those results.

Chapter 2

Literature Review

In this chapter we review important works in the literature which form the foundation of this thesis. We first discuss the initial set of works which cover Wi-Fi activity detection as well as other related works which do not use CSI data specificially before discussing those works which do utilize CSI data and publicly available datasets for this purpose. We then look at various signal-to-image transformation methods which may be applied to time-series signal data, enabling the use of techniques from the image processing domain. Finally, we look into domain shift mitigation methods and specifically reinforcement learning for domain shift mitigation.

2.1 Wi-Fi for Activity Detection

Past works have investigated the use of Wi-Fi signals generally for the purposes of activity detection.

The first work regarding the use of Wi-Fi signals for the detection of humans subjects we could find is the work of Chetty, Smith, and Woodbridge in 2012 [5]. Their work utilized passive Wi-Fi signals propagating through a building with receivers placed outside the building for presence detection. This method achieved reasonable results and proved that Wi-Fi signals could be used to detect human presence in buildings, although it required the indoor and outdoor APs to be synchronized through wires and was unable to detect precise activities of the human subjects.

The first work we could find discussing the use of Wi-Fi for activity detection is the work from Fadel Adib and Dina Katabi, published in 2013 [1]. This work shows the potential of using signals which could be produced by Wi-Fi APs to detect human activity from through a wall. The most important idea in this work is the elimination of the radio “flash” which

comes with the signal hitting a wall and bouncing back towards the transceiver. Their work focused more on the radar technology implications and not on the use of consumer Wi-Fi APs for gesture detection. They did, though, show that using matched filters was enough to perform rudimentary gesture recognition, given coarse enough gestures.

In the same year, a different group published a paper showing how to use signals in the 2.4 GHz range, i.e., compatible with Wi-Fi transceivers, for simple gesture detection using Doppler shift identification [18]. This paper proposes the use of a narrowband pulse with a very narrow bandwidth of only a few Hertz and detecting the Doppler shift from the returned signal. Using this method, the researchers were able to identify 9 different gestures with a claimed 94% accuracy.

The same group as [1] also published a separate paper in 2014 detailing the use of a custom-built Wi-Fi based device which could detect course body motions by leveraging the geometric position of its antennas and measuring values through a Time of Flight (ToF) approach [2].

Finally, it is also important to note that IEEE has a task group 802.11bf assigned specifically to standardize Wi-Fi sensing technologies [6]. This group is focused on standardizing the hardware requirements, specifically enabling CSI accessibility and specific measurement procedures which future devices can implement. Their target is to standardize these requirements for future devices both in the sub-7 GHz range and in the 60 GHz range. They additionally provide suggestions for what methods can be then be used to interpret the data provided, including the use of Fast Fourier Transform (FFT) algorithms to calculate a Channel Impulse Response from CSI and a Doppler FFT, which may be directly used for gesture recognition. The standards for 802.11bf is set to be ratified and published by September 2024.

2.2 Wi-Fi CSI for Gesture Recognition

A selection of works which utilize Wi-Fi CSI data specifically for the task of gesture recognition are discussed in this section.

To the best of our knowledge, the first work discussing the use of CSI for gesture recognition was published in 2015 by He et al. [10]. This work looks into the use of CSI and outlier detection to detect gestures, achieving 92% gesture recognition accuracy on four gestures in a line-of-sight experiment and 88% accuracy in a non-line-of-sight experiment.

The 2019 work titled Person-in-WiFi from Wang et al. proposes the use of an array of three transmitter and three receiver antennas to directly predict body segmentation and pose estimation of persons located in between the aforementioned antennas [23]. In this

work, they used an RGB camera to provide ground truth annotations. The ground truth body segmentation masks were generated using Mask-RCNN while the Body-25 model of OpenPose was used for pose detection. This work, shows that body segmentation and pose estimation is possible with only CSI data, achieving an mAP of 0.38 for body segmentation and around 0.1 meter error for join estimation. Qualitatively, the results are quite impressive and it is clear that at the very least, the model performs well given that its input data is one-dimensional.

Using the same dataset, Geng, Huang, and De La Torre published DensePose in 2022 performs similarly, but instead produces UV coordinates of the subjects. This work also provides some interesting preprocessing steps on the raw CSI data to improve prediction performance.

WiGAN, published in 2020, proposes the use of a Generative Adversarial Network (GAN) to augment training data using the generator as well as using the discriminator as a feature fusion and extraction module [12]. The output of the discriminator is then used fed into a Support Vector Machine (SVM) which classifies then gesture seen in the input data. The authors claim that by using the discriminator, which fuses together multiple layers of a CNN module through yet another convolution, their method is able to "learn and recognize the importance of different depth features by itself". On publicly available datasets, WiGAN indeed achieves better results than the competing methods at the time, achieving 98% accuracy on Widar 3.0 with known subjects and ≈8–10% lower performance on new domains.

The same group, in the same year, published DeepMV which proposes the use of multiple APs and audio sources, in the form of ultrasound signals, with a domain discriminator and embedding generator [25]. This work is based on the intuition that the fusion of sensor data from multiple APs and audio sources placed around the room, which intuitively should provide more data. The embedding generator produces a latent representation of the action which can then be used by a fully connected module to classify the action being performed while the domain discriminator uses the latent representation to predict the domain of the action. A minimax game is played between the embedding generator and domain discriminator to minimize the maximum accuracy of the domain discriminator while maximizing the minimum accuracy of the action classifier. On their self-collected dataset, they were able to achieve 83.7% mean accuracy, outperforming all other benchmarked methods.

The 2021 paper by Zhuravchak et al. proposes the use of an LSTM as a classifier. In this method approach, CSI data is provided as an input with a fixed length and the LSTM provides a single output representing the action detected within the input window. Their method achieves 87.8% accuracy on a self-collected dataset.

Ma et al., published in 2021, proposes the use of a CNN and neural network state machine encoder and a LSTM trained using RL to eliminate the need of domain specific information [14]. This work also contains a curated benchmark of many previous works in this field. Their proposal is the use of a neural network state machine relies on the assumption that there is a temporal dependency within and across CSI segments. For example, it is unlikely that a person who is currently standing will immediately be sitting within the next CSI segment without a sit-down transition segment in between. Although unlikely, the probability is non-zero, due to incontinuities in the data and mislabeling in the ground-truth labels, and thus a neural network state machine is used. The results show >97% mean accuracy on in-domain test samples with a drop of 14–17% on out-of-domain samples.

Additionally, a few bachelors thesis' fromt he past years at the IRIS research cluster at TU/e have focused on this problem as well. The 2022 thesis by van den Biggelaar proposes the use of reinforcement learning with Deep Q-Networks as the gesture classifier [4]. Their result shows ∼88% mean accuracy on Widar 3.0, dropping by ∼4% on out-of-domain test samples.

The 2022 thesis by Oerlemans compares how different preprocessing methods appear to affect gesture recognition performance [17]. Specifically, signal filtering through a finite impulse response filter and phase unwrapping, transformation to a Doppler frequency spectrum (DFS), and transformation using Gramian Angular Fields (GAF) was explored. On the SignFi dataset, they show that each of the above steps do indeed significantly improve model performance with the GAF transformation coupled with signal filtering resulting in the best performance.

2.3 Wi-Fi CSI datasets for Gesture Recognition

Three Wi-Fi CSI datasets for gesture recognition were considered for this thesis and they are each explained in this section.

The Widar 3.0 dataset, forthwith referred to as the Widar dataset for brevity, presents a dataset with developed specifically for "cross-domain learning solutions" [28]. The solution provided in this dataset is two-fold: 1) the (relatively) high number of domains that the data was collected with, and 2) the proposal of Body-coordinate Velocity Profile (BVP) with is a theoretically domain-independent representation of the data. More details of BVP is discussed in Section . Finally, this paper also provides a baseline model to compare against.

Desecribe in background section

SignFi is a dataset of Wi-Fi CSI data specifically for sign language recognition [15]. This dataset contains over 276 sign gestures in a lab and home environment with five different users. A baseline CNN model is also presented in this paper, capable of achieving a ∼87%

mean accuracy over 150 sign gestures.

Person-in-WiFi is the dataset used in the paper by Wang et al. [23]. This dataset was made public and includes both Wi-Fi CSI data and RGB camera data of the activity from a fixed position. This dataset was collected specifically for pose estimation solutions and is not meant specifically for activity recognition, as no activity labels are provided in the dataset.

2.4 Signal-to-Image Transformations

Different preprocessing methods have been investigated to transform raw tabular data into images for deep learning. Four state-of-the-art approaches are DeepInsight [20], REFINED [3], GAF, and MTF [24]. A search of the current body of literature did not yield any research into a direct comparison of these techniques on a common dataset. Instead, a previous unpublished work by the author of this thesis for the Seminar course at the TU/e has shown that these four methods performed best among state-of-the-art signal-to-image transformations [19]. A more thorough description of each of these methods are described in Section

Describe in background section

2.5 Domain Shift Mitigation Methods

There are a number of papers focusing on domain shift mitigation methods. A selection of these papers, specifically those which are highly related to our problem domain, are discussed in this section.

The 2021 paper by Zinys et al. focuses on the use of GANs for domain shift mitigation, called Adversarial Domain Adaptation (ADA) [29]. In this work, the discriminator attempts to predict gesture and domain while the generator produces sample data. The discriminator is provided a loss function based on ground truth data and accurately identifying generated data while the generator produces sample data which is in the same domain and gesture as its provided input. Their results, tested on Widar show significantly better performance than the baseline Widar model on unseen domains.

Zhang et al., in 2022, proposes the use of federated learning for domain shift mitigation [26]. The concept is to allow for each user to train their own neural networks and using matched average federated learning to combine all user models together. Tested on the Widar dataset, they show performance competitive with state-of-the-art techniques.

Van Berlo et al. discusses attempts at using mini-batch alignment to generate domain factor independent latent representations of the data. They showed that unfortunately, the

proposed mini-batch alignment pipeline did not lead to better performance across domains. The authors believe this may be due to a lack of sufficient domain factor information, leading to poor mini-batch alignment. Alternatively, the assumptions of the underlying probability distributions may be incorrect. In any case, it seems that this method, given current publicly available datasets, does not improve the SOTA.

Finally, the 2022 bachelors thesis by Sips investigates the use of network pruning and quantization for domain shift mitigation [21]. The basic concept is to improve model robustness by enforcing sparsity and utilizing mixed precision training. Tested against a baseline where sparsity and mixed precision training was not used, the modified networks performed slightly worse on in-domain test samples while they had mixed results on out-of-domain samples.

2.6 Domain Shift Mitigation

The following section contains some selected works in domain shift mitigation, mainly those related directly with RL of self-supervised techniques.

Zhang et al. in 2021 published research into using RL based features election for domain shift mitigation [27]. The proposed method would be able to select the most relevant features across two domains by employing Q-learning to learn policies for feature selection, utilizing the performance of a domain discriminator as its reward function. By doing so, they attempt to align the feature manifolds between both domains. Benchmarked on publicly available datasets, this method achieves the best mean accuracy among SOTA methods.

Martini et al. published a technique called "Domain-Adversarial Neural Networks" in 2021 [16]. This technique uses a feature extractor, a domain classifier, and a label predictor. The feature extractor maps the input to a latent representation, the domain classifier predicts the domain given the latent representation, and the label predictor provides a class prediction given the extracted features. The loss function of the model balances the label predictor loss, using Cross Entropy Loss, and the domain classifier loss, also using Cross Entropy Loss, with the goal of providing a latent representation from the feature extractor which is domain-invariant, yet still discriminative such that the domain classifier is still capable of accurately classifying the domain. Additionally, the paper proposes the use of Maximum Mean Discrepancy (MMD) to measure the difference between two domains. The MMD measures the kernel-based difference between feature means of each domain. This work proposes the use of MMD with the goal of minimizing this distance while maintaining the distinctness of each domain, allowing for the domain classifier to still work.

Chapter 3

Methodology

The general intuition behind our approach is the lack of ability for a model to perform on unseen domains without any additional information. The general architecture of our approach can be seen in Figure 3.1. We first propose a signal preprocessing and signal-to-image transformation pipeline, described in Section 3.1 and 3.2. This transforms the input signal into an image that is then passed through a CNN encoder, as described in Section sec:methodology-signal-encoding, encoding the signal into a latent representation. This latent representation is then used in two different ways: As an input for the multi-task learning heads, described in Section 3.5, and as the state observation for our RL agent, described in Section 3.4.

Two mitigate domain-shift, our RL agent is tasked with producing the best possible embedding of the domain D_r as its action given the signal latent representation as its state observation.

To provide the RL agent with a reward function, inspired by triplet loss, we use two different multi-task learning modules. One of these modules is provided D_0, representing a vector of zeros, and the other D_r.

This chapter is not finalized, due to the possibility of changing things around so will be kept as bullet points at this stage.

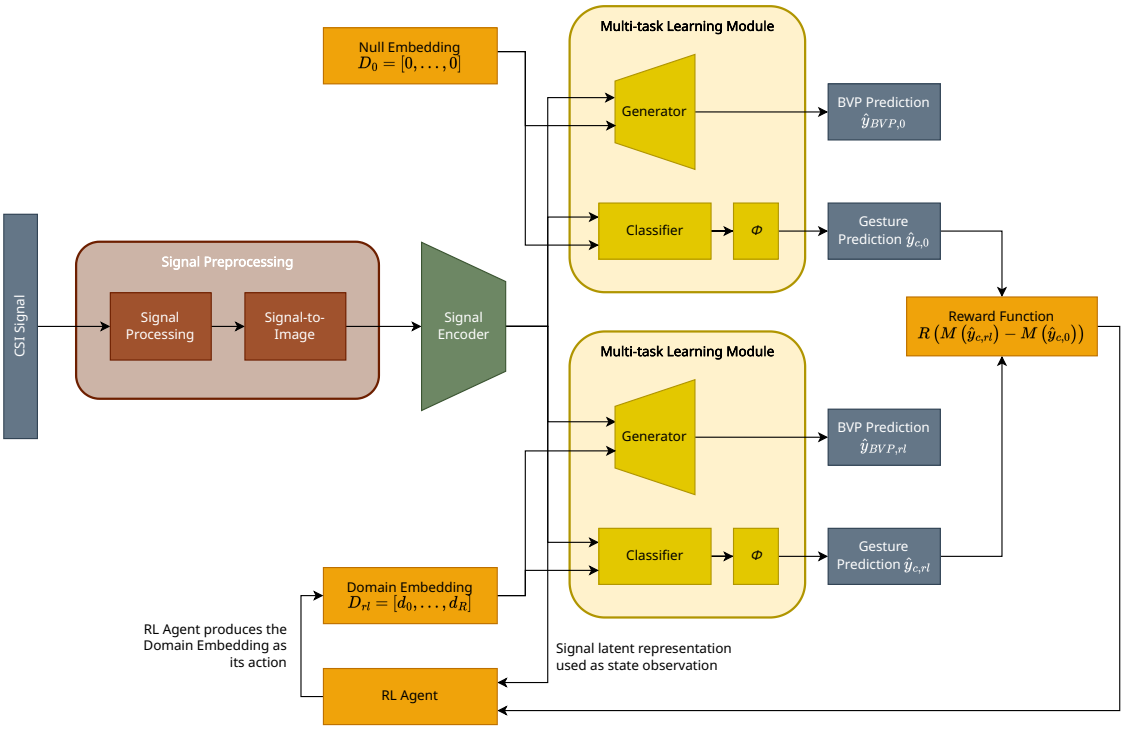

Figure 3.1: An abstracted diagram of the proposed architecture in this thesis. Red-brown components are for signal processing and signal-to-image transformation. Green components are for the signal encoding, yellow components are the multi-task-learning modules, orange components are for RL, and blue components are either input or outputs of the model.

3.1 Signal Processing

As the old adage goes, garbage in, garbage out. As such, we propose the following steps from traditional signal processing to clean up the input signal for our ML algorithm.

- Noise filtering with a low-pass filter, since high-freq noise is most likely not human movement
- CSI Phase unwrapping and linear fitting as suggested by [8]
- CSI Phase derivative, to keep values from changing magnitude as in the case of unwrapping
- DWT seems to have been attempted in some papers, so let's see if that helps.
- We'll do an ablation study to see if these actually help

The signal can now be considered clean, or at least clean enough that we can continue with further steps.

3.2 Signal-to-Image Transformation

Four crazy signal-to-image transformation methods! You won't believe number three!

If there are no objections, this will be in the final paper

The next stage in our method is transform the signal into an image, leveraging advances from computer vision. There are many works from the literature which show that there are certainly improvements that can be gained from performing image processing on temporal data. We will experiment with the following four methods for signal-to-image transformation: DeepInsight [20], REFINED [3], GAF/FGAF (Feature-wise GAF) [24, 19], and MTF [24].

3.2.1 DeepInsight

DeepInsight [20] applies t-SNE on the columns of V, resulting in each AP k being mapped to a point in 2D space. The entire space is then rotated to fit within the minimum rectangular bounding box. These rotated points then become the pixel coordinates for a given AP. The value of the RSSI of a given AP k is then directly mapped to the value of its corresponding pixel. If multiple features share a pixel location, the average value of these features is then used as the value of the pixel.

3.2.2 REFINED

REpresentation of Features as Images with NEighborhood Dependencies [3] uses a technique the authors call "Bayesian Multidimensional Scaling". A 2D embedding of the data V_{embed} is calculated using Multidimensional Scaling (MDS) with a Euclidean distance metric. A pixel grid P of p^2 pixels, where $p = \lceil \sqrt{K} \rceil$ is then produced with dimensions $p \times p$. A mapping of features to pixels is calculated by considering all permutations which minimize the Euclidean distance of the pixel mapping to the feature location in V_{embed} while keeping at most one feature per pixel iteratively. This final mapping is calculated using a hill-climb algorithm.

3.2.3 GAF

Gramian Angular Fields [24] first transform the incoming signal for each AP to polar coordinates with

$$\vec{w_{i,k}} = \begin{cases} \phi_{i,k} = \arccos(V_{n,k}) \\ r_{i,k} = i/I \end{cases} \tag{3.1}$$

where $\vec{w_{i,k}}$ is the resulting vector of a signal, $V_{i,k}$ the value of the signal from AP K at sample i, i the sample number of the fingerprint, and I the total number of timesteps at the given sample location. For the case of our dataset, there are 6 timesteps for each location.

The Gramian is than calculated as

$$G_k = \begin{bmatrix} w_{1,k} \cdot w_{1,k} & \cdots & w_{1,k} \cdot w_{I,k} \\ \vdots & \ddots & \vdots \\ w_{I,k} \cdot w_{1,k} & \cdots & w_{I,k} \cdot w_{I,k} \end{bmatrix} \tag{3.2}$$

for each AP k creating an image with k channels.

We discovered in a previous project that calculating vector $\vec{w}$ across features, i.e., channels, instead of across timesteps can also be useful. This results in

$$\vec{w_{n,k}} = \begin{cases} \phi_{n,k} = \arccos(V_{n,k}) \\ r_{n,k} = k/K \end{cases} \tag{3.3}$$

where n is the sample index and k is the feature index. This leads to the Gramian for each sample

$$G_n = \begin{bmatrix} w_{n,1} \cdot w_{n,1} & \cdots & w_{n,1} \cdot w_{n,K} \\ \vdots & \ddots & \vdots \\ w_{n,K} \cdot w_{n,1} & \cdots & w_{n,K} \cdot w_{n,K} \end{bmatrix} \tag{3.4}$$

We call this transformation Feature-wise GAF (FGAF).

In either case, G is finally normalized.

3.2.4 MTF

Markovian Transition Fields [24] first quantizes the signal into q bins, each bin representing an interval of RSSI strength. A Markovian transition matrix M_t is then constructed where each row represents a fingerprint and each column a bin. The values in M_t are the size of each bin for a given sample. M_t is then normalized and aligned along the temporal axis such that in this new matrix M, $M_{i,j}$ represents the probability of a transition from bin i to bin j. M is then an image of size $q \times q$.

Regardless of the chosen method, the signal has now been transformed into an image and we can proceed with the encoding of this image into a latent space.

3.3 Signal encoding

The encoding of the signal, now an image, into a latent space is performed using CNN-based image processing methods.

- The signal encoding backbone is a standard CNN
- This will probably be something simple, like ResNet since the signal shouldn't be too complex that it requires something very SOTA
- Alternatively, mobilenet may be used as well
- This may actually be the least important aspect of this thesis

The latent representation of this signal is then passed to both the reinforcement learning agent as its state observation as well as to the multi-task learning modules. How the RL agent uses this state observation is described in Section 3.4 while its use by the multi-task learning module is described in Section 3.5.

3.4 Unsupervised Domain Representations through Reinforcement Learning

To mitigate domain shift, we implement a novel method for unsupervised domain representations through reinforcement learning.

- Base terminology: Agent, action, state observation (state), and reward
- The encoder-decoder architecture described in Section 3.3 and 3.5 is the environment.

- RL loop:
  1. The agent receives the image embedding produced by the encoder/backbone as its state
  2. The reinforcement learning agent produces an embedding of the domain as its action
  3. The environment is trained with one pair of heads receiving the action of the agent while the other pair receives senseless values (either all 0s if the representation is one-hot or a uniform distribution if the representation is a probability measure)
  4. After training of the agent, the RL agent is provided a reward from the environment which is used to improve the agent
- Let the loss function $\mathscr{L}_w, \mathscr{L}_{wo}$ represent the loss function of the heads with and without the action provided by the agent, respectively
- Then, the reward function $\mathscr{R}$ of the agent is $\mathscr{R} = \mathscr{L}_{wo} - \mathscr{L}_w$
  - The intuition is, the reward is based on how much better the head pair with the action should perform than the head pair without the action
- To speed up training, we take a page from AutoML competitions and the environment is trained within 1 minute and be focused on improving performance as fast as possible instead of achieving the best performance, at least during hyperparam optimization
- After hyperparameters are chosen, then we train properly and fully.
- A reward is then calculated using the function $R(M(\hat{y}_{rl}, y) - M(\hat{y}_0, y))$, where M is the metric used to calculate the performance of the gesture classifier in each multi-task learning module and R is some reward function.
- We do not use the absolute difference, as we are interested in the having negative values representing the module provided D_0 having better performance.

3.5 Multi-task Learning

The actual gesture classification module is built using a multi-task learning module.

- The idea is to enforce some sort of representation that *can* be used to get a domain-independent representation of the data

- We utilize multi-task learning for this, where one head is used to predict BVP, which is theoretically domain-independent [28]
  - We are not actually interested in the BVP, but having it as part of our training will enforce the need for a representation which is theoretically domain-independent.
  - [16] suggests the use of MMD and an adversarial approach to training the model to ensure that while the representation is domain-invariant, it still retains enough discriminatory powers such that the domain classifier can still perform.
  - Instead of this approach, we use the BVP prediction to enforce a representation which is domain-invariant and we use an RL agent instead of the domain classifier suggested in [16].
- The other head is a classifier head and classifies gestures
- It's been shown that doing multi-task learning like this leads to good results with latent representations which are more robust [22]
- We duplicate this pair of heads to enable a reward function for the RL agent inspired by triplet loss
- Loss function for the gesture classifier is cross entropy loss and for the BVP generator, it is binary-cross entropy loss.

Bibliography

[1] Fadel Adib and Dina Katabi. “See through walls with WiFI’. In: *Proceedings of the ACM SIGCOMM 2013 conference on SIGCOMM*. 2013, pp. 75–86.

[2] Fadel Adib et al. “3D tracking via body radio reflections”. In: *11th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 14)*. 2014, pp. 317–329.

[3] Omid Bazgir et al. “Representation of features as images with neighborhood dependencies for compatibility with convolutional neural networks”. In: *Nature communications* 11.1 (2020), pp. 1–13.

[4] Lieke A.J. van den Biggelaar. “Gesture recognition using Deep Q-Neworks on WiFi-CSI data”. MA thesis. Eindhoven University of Technology, 2022.

[5] Kevin Chetty, Graeme E Smith, and Karl Woodbridge. “Through-the-wall sensing of personnel using passive bistatic wifi radar at standoff distances”. In: *IEEE Transactions on Geoscience and Remote Sensing* 50.4 (2011), pp. 1218–1226.

[6] Rui Du et al. “An overview on IEEE 802.11 bf: WLAN sensing”. In: *arXiv preprint arXiv:2207.04859* (2022).

[7] Technical University of Eindhoven. *Atlas most sustainable education building in the world Atlas most sustainable education building in the world Atlas most sustainable education building in the world*. Apr. 2019. URL: https://www.tue.nl/en/our-university/tue-campus/buildings/atlas/atlas-most-sustainable-education-building-in-the-world/.

[8] Jiaqi Geng, Dong Huang, and Fernando De la Torre. “DensePose From WiFi”. In: *arXiv preprint arXiv:2301.00250* (2022).

[9] Mohammad Hasan. *State of IoT 2022 Number of connected IoT devices growing 18% to 14.4 billion globally*. URL: https://iot-analytics.com/number-connected-iot-devices/ (visited on 05/18/2022).

[10] Wenfeng He et al. "WiG: WiFi-based gesture recognition system". In: *2015 24th International Conference on Computer Communication and Networks (ICCCN)*. IEEE. 2015, pp. 1–7.

[11] Chip Huyen. *Designing Machine Learning Systems*. USA: O'Reilly Media, 2022. ISBN: 978-1801819312.

[12] Dehao Jiang, Mingqi Li, and Chunling Xu. "Wigan: A wifi based gesture recognition system with gans". In: *Sensors* 20.17 (2020), p. 4757.

[13] Wenjun Jiang et al. "Towards environment independent device free human activity recognition". In: *Proceedings of the 24th annual international conference on mobile computing and networking*. 2018, pp. 289–304.

[14] Yongsen Ma et al. "Location-and person-independent activity recognition with WiFi, deep neural networks, and reinforcement learning". In: *ACM Transactions on Internet of Things* 2.1 (2021), pp. 1–25.

[15] Yongsen Ma et al. "Signfi: Sign language recognition using wifi". In: *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2.1 (2018), pp. 1–21.

[16] Mauro Martini et al. "Domain-Adversarial Training of Self-Attention-Based Networks for Land Cover Classification Using Multi-Temporal Sentinel-2 Satellite Imagery". In: *Remote Sensing* 13.13 (2021). ISSN: 2072-4292. DOI: 10.3390/rs13132564. URL: https://www.mdpi.com/2072-4292/13/13/2564.

[17] C.H.J.M Oerlemans. "The Effect of Data Preprocessing On the Performance of Few-shot Learning for Wi-Fi CSI- based Gesture Recognition The Effect of Data Preprocessing on the Performance of Few-shot Learning for Wi-Fi CSI-based Gesture Recognition". MA thesis. Tilburg University, 2022.

[18] Qifan Pu et al. "Whole-home gesture recognition using wireless signals". In: *Proceedings of the 19th annual international conference on Mobile computing & networking*. 2013, pp. 27–38.

[19] Yvan Putra Satyawan. "CNNs and Preprocessing: The Dynamic Duo of Wi-Fi Localization". Jan. 2023.

[20] Alok Sharma et al. "DeepInsight: A methodology to transform a non-image data to an image for convolution neural network architecture". In: *Scientific reports* 9.1 (2019), pp. 1–7.

[21] Suze E. Sips. "Impact analysis of network pruning and quantization on the domain shift problem in a recognition context, using WiFi CSI data". MA thesis. Eindhoven University of Technology, 2022.

[22] Lukas Tuggener et al. “The DeepScoresV2 dataset and benchmark for music object detection”. In: *2020 25th International Conference on Pattern Recognition (ICPR)*. IEEE. 2021, pp. 9188–9195.

[23] Fei Wang et al. “Person-in-WiFi: Fine-grained person perception using WiFi”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2019, pp. 5452–5461.

[24] Zhiguang Wang and Tim Oates. “Imaging time-series to improve classification and imputation”. In: *24th International Joint Conference on Artificial Intelligence*. 2015.

[25] Hongfei Xue et al. “DeepMV: Multi-view deep learning for device-free human activity recognition”. In: *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4.1 (2020), pp. 1–26.

[26] Weidong Zhang, Zexing Wang, and Xuangou Wu. “WiFi signal-based gesture recognition using federated parameter-matched aggregation”. In: *Sensors* 22.6 (2022), p. 2349.

[27] Youshan Zhang, Hui Ye, and Brian D Davison. “Adversarial reinforcement learning for unsupervised domain adaptation”. In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 2021, pp. 635–644.

[28] Yue Zheng et al. “Zero-effort cross-domain gesture recognition with Wi-Fi”. In: *Proceedings of the 17th annual international conference on mobile systems, applications, and services*. 2019, pp. 313–325.

[29] Augustinas Zinys, Bram van Berlo, and Nirvana Meratnia. “A domain-independent generative adversarial network for activity recognition using wifi csi data”. In: *Sensors* 21.23 (2021), p. 7852.

Appendix A

Plan

The plan involves of the following steps, which are elaborated upon in the Gantt Chart seen in Table A.1.

1. Initial infrastructure setup. This involves building the train-validate-test loops, building the data ingest/transformation pipelines, and building the model-building pipelines. This is feasible in the given timespan since much of the code will be taken from the IRIS Seminar project.
2. Integration of all modules as well as integration of hyperparameter optimization and training-tracking code complete. Initial training/debugging of the network can begin.
3. Parallelization of reinforcement-learning and deep-learning components of the network are complete. This involves making it possible to run the RL and the DL models on separate computers during training, if this improves performance.
4. Get initial results, bugs will be found and "beta-testing" of the framework starts. All results at this point are taken with a massive grain of salt since something will cause the results to be wrong, speaking from experience.
5. Deeper investigations through hyperparameter optimization and changes to the model architecture should now start or is already ongoing.
6. Final results from the experiments should be completed.
7. First draft of the paper.
8. Second draft the paper.
9. Final draft of the paper.

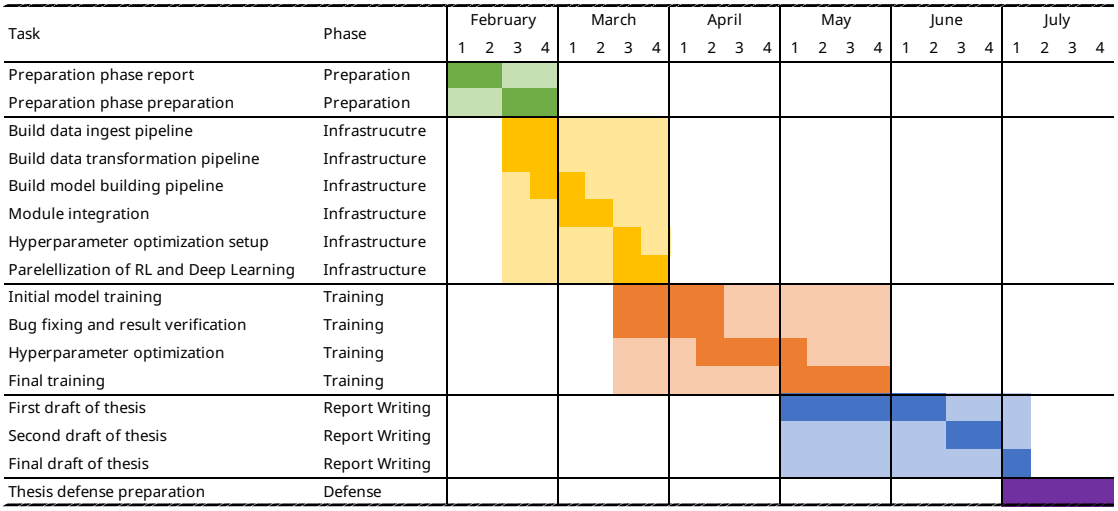

Figure A.1: A Gantt chart showing the plan for this thesis

10. Thesis defense preparation.

Appendix B

Risk Assessment

Risks identified for each phase of the plan and mitigation options are investigated in this section. Risk mitigation strategies are built from the author's previous experience working in an AI research lab and as a data scientist and software developer in industry. Additionally, strategies are also developed from theory learned in his computer science bachelor's.

Initial infrastructure setup During the initial infrastructure setup, the entire pipeline will be developed. This includes a modular data-ingest/transformation pipeline and model-building pipeline. Risks include data availability/usability and bugs in the data pipeline code. Additionally, the model will be completely modular and not prebuilt, increasing the chances of a bug appearing during this building phase but increasing flexibility of the model being investigated.

Data availability/usability refers to the fact that while the data is a public and published dataset, it is nonetheless quite large and it would not be feasible to download the entire dataset and place it on TU/e's HPC server. This means that some way to compress the data must be done. It is also possible that the transformed version of the data can be compressed more efficiently and this is what will end up being the dataset we work with for the majority of the project.

The data pipeline is also modular, allowing for data augmentation to be added "on-the-fly" instead of being hard-coded. This increases flexibility, but introduces the risk of bugs in unexpected circumstances. Mitigation factors include having written similar pipelines multiple times in the past and reuse of old, known-good code from the IRIS Seminar project and 2AMM10 Deep Learning course. Additionally, a similar approach has been use in previous research that we have completed, and we have significant experience in similarly modular pipelines.

Similarly, the model is built only at run-time, allowing for more flexibility and the possibility to fine-tune the model architecture using hyperparameter optimization techniques. This increases the chance that a good model architecture is chosen and strengthens the reasoning behind the chosen model architecture through empirical performance. Mitigation factors are the same as for the data pipeline.

Module Integration The most complex part of this infrastructure is to ensure that all modules work well together and there are no bugs in the hand-off step between modules.

To mitigate these factors, we take some advice from Chip Huyen's book Designing Machine Learning Systems [11]. To ensure that errors are not made during this integration, data flow will be closely monitored and visualized at every step through a UI, such that it is easy to see if anything went wrong at any step. The application of this will essentially be most of the interim steps being given some sort of output so we can visualize their result and track how data is transformed throughout the entire process.

Parallelization This is potentially unnecessary and may take up time that could be better used elsewhere. The idea is essentially that it might make sense to have the reinforcement-learning model and the gesture-classification model run on separate computers and having them communicate through some network.

Mitigation for this being unnecessary is providing only a limited amount of time to do this and the understanding that if this seems too difficult/may take too long, then we will immediately shelf the idea.

General bugs As with any software-based project there will inevitably be bugs in the code. Software engineering principle which lead to fewer bugs, such as proper use of debugging tools (but not test suites as we don't believe they will be necessary for a project with a limited scope such as this), will be used throughout work on this thesis. Additionally, use of "magic numbers" will be limited and as many parameters as appropriate will be assigned from variables.