Enabling Contactless Sleep Studies at Home using Wireless Signals

by

Shichao Yue

B.Eng., Tsinghua University (2016)
S.M., Massachusetts Institute of Technology (2018)

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2021

Author .....
Department of Electrical Engineering and Computer Science
May 14, 2021

Certified by .....
Dina Katabi
Professor of Electrical Engineering and Computer Science
Thesis Supervisor

Accepted by .....
Leslie A. Kolodziejski
Professor of Electrical Engineering and Computer Science
Chair, Department Committee on Graduate Students

Enabling Contactless Sleep Studies at Home using Wireless Signals

by

Shichao Yue

Submitted to the Department of Electrical Engineering and Computer Science
on May 14, 2021, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Abstract

Sleep studies help doctors diagnose a variety of sleep-related disorders, such as insomnia and sleep apnea. Most disorders can be managed once they are correctly diagnosed. However, sleep studies usually introduce discomfort and high cost, as patients need to go to hospitals, sleep in unfamiliar beds, and wear sensors all over the body. Such high bars may discourage patients from taking sleep studies.

In this thesis, we explore alternative solutions that make sleep studies comfortable and affordable, and at the same time, deliver clinically meaningful measurements. Specifically, we propose four systems that leverage radio frequency (RF) signals to remotely and automatically monitor the user's sleep in the user's bedroom. First, we present EZ-Sleep, the first contactless sleep sensor that automatically identifies bed entries and exits, and monitors key insomnia parameters like sleep latency and sleep efficiency. Further, we propose RF-Sleep, which demonstrates a much superior sleep stages classification accuracy than state-of-art methods. Next, we introduce DeepBreath, the first wireless respiration monitoring system that can recover the breathing signals of multiple individuals even when they right next to each other. And finally, we have BodyCompass, the first RF-based system that provides accurate sleep posture monitoring overnight.

Collectively with all four systems, we turn the user's bedroom into a contactless sleep lab. We provide doctors tools to diagnose and keep track of sleep-related symptoms without any additional efforts from the patient.

Thesis Supervisor: Dina Katabi
Title: Professor of Electrical Engineering and Computer Science

Acknowledgments

I have an unforgettable and unbelievable Ph.D. journey. It is hard. But with help from so many people, I am not alone. And I am finally reaching my destination.

First and foremost, this thesis would not have been possible if my adviser Dina, had not believed in me. Six years ago, I nearly knew nothing about research and wireless sensing. Yet Dina trusted in my potential and brought me here to MIT. Over these years, she encouraged me a hundred times whenever I felt frustrated. I still remembered the time when I was stuck with the BodyCompass project. I tried so hard but could not get any improvements. I was so discouraged that I felt it was impossible. Dina said to me that I should believe in myself, and she also believed in me. She even brought me a gift, an eGPU! Something I always wanted to have and never had an excuse to buy. I was totally shocked, staring at that eGPU for a long time and could not believe it. It became my driving force and encouraged me not to stop. Ultimately, we figured it out, and the BodyCompass project was a big success. I felt blessed to have Dina as my advisor.

I was also very fortunate to work with all my excellent collaborators. Mingmin taught me how to do outstanding research with his dedication and expertise in my first year at MIT. Chen-Yu is like a brother to me, as he feels warm and encouraging, and he can always point out problems in the most acceptable way. Coding with Hao He is such a memorable and amusing experience. Hao Wang is a reliable information source, and he kindly explained all my questions in great detail. Working with Peng is another unique experience. It is basically two-shift work, and there is always someone working. Zach provides me the best logistics services. Whenever I break something, He can always fix it back. And for Rumen and Rahul, hopefully, we can continue to collaborate for the years to come.

I am proud to be a member of NETMIT. We usually spent hours discussing ideas and papers. Those valuable discussions and brainstorms helped me overcome many challenges. They also made me confident during conference presentations. If they think my presentation is acceptable, then the rest of the world should probably think it is perfect. I also want to thank our group admin, Mary, for taking care of everything. She is truly helpful and warmhearted.

And I want to say thanks to all my friends. When I was down, they were here to cheer me up. I remembered the time when I was photographing at the white mountain, passing the time at Niagara falls, throwing axes, and playing bowling. They made my life rich and colorful.

And finally, I owe all my accomplishments to my family. My parents provide me the best possible environment for me, and I can do whatever I want to do. They are always here to support me and love me. They are the best parents in the world. And Yajing, I can never thank you enough. You bear all my shortcomings and make me a better man. Thank you.

Previously Published Materials

Chapter 2 revises a previous publication [62]: Chen-Yu Hsu, Aayush Ahuja, **Shichao Yue**, Rumen Hristov, Zachary Kabelac, and Dina Katabi. "Zero-effort in-home sleep and insomnia monitoring using radio signals." Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies, 2017

Chapter 3 revises a previous publication [179]: Mingmin Zhao, **Shichao Yue**, Dina Katabi, Tommi S. Jaakkola, and Matt T. Bianchi. "Learning Sleep Stages from Radio Signals: A Conditional Adversarial Architecture." In International Conference on Machine Learning, 2017

Chapter 4 revises a previous publication [172]: **Shichao Yue**, Hao He, Hao Wang, Hariharan Rahul, and Dina Katabi. "Extracting Multi-Person Respiration from Entangled RF Signals." Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies, 2018.

Chapter 5 revises a previous publication [173]: **Shichao Yue**, Yuzhe Yang, Hao Wang, Hariharan Rahul, and Dina Katabi. "Bodycompass: Monitoring sleep posture with wireless signals." Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies, 2020.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivations

Sleep plays an important role in our physical and mental health. Having a good night of sleep can not only make us feel energetic, but also keep our body healthy and prevent diseases. For example, sleep improves brain functions by removing toxins that build up while you are awake [165], and it speeds up muscle recovery by effecting hormone secretion [32].

Unfortunately, many people in the world suffer from poor sleep quality due to various sleep-related problems. For example, insomnia and sleep deprivation are common health problems in the US. One in every three Americans do not get enough sleep, and about 10% of the population suffers from chronic insomnia [108]. Sleep apnea, which affects approximately 18-22 million Americans [129], is another common, potentially serious sleep disorder. People with untreated sleep apnea stop breathing repeatedly during their sleep, sometimes hundreds of times during the night. Sleep apnea may also cause severe health problems if left untreated, including hypertension [49], stroke [167], etc.

To diagnose and treat sleep disorders, the patient usually needs to undergo a sleep study. A sleep study monitors vital signals throughout the night, e.g. the respiration signal, sleep stages and sleep postures. Those signals provide doctors more conclusive evidences about certain type of sleep disorders [137]. Sleep studies may also be used to help initiate or adjust treatment plans.

However, prevailing approaches for monitoring sleep are inconvenient and intrusive.

The medical gold standard relies on overnight polysomnography (PSG), which is typically conducted in a hospital or sleep lab, and requires the patient to wear a plethora of sensors, such as EEG-scalp electrodes, an ECG monitor, multiple chest bands, and nasal probes. As a result, patients can experience sleeping difficulties, which render the measurements unrepresentative, especially for insomnia patients (the first night effect [8]). Furthermore, the cost and discomfort of the PSG limit the potential for long-term sleep studies.

For at-home long-term sleep-tracking, doctors usually ask the patient to keep a sleep diary. However it puts much overhead on the patient, as the patient needs to record many parameters everyday, e.g. bed time, lights-out time and time it takes to fall asleep [135]. Therefore it is hard to sustain a sleep diary for a long period of time. Making it even worse, research shows that people's impression of their sleep can be wildly wrong, off by hours in some cases [139, 17]. Alternative approaches use wearable sensors, such as actigraphy, which leverage accelerometers to monitor body movements and infer sleep quality. However, many people do not feel comfortable sleeping with wearable devices [23]. And motion alone can only provide limited information; for example, it is known to be a poor metric for estimating sleep stages [114].

1.2 Proposed Solutions

In this thesis, we aim to monitor the user's sleep fully automatically and contactlessly at home, all without scarifying accuracy. Our systems work by transmitting low-power radio-frequency signals (RF signals), and analyzing RF signals that bounced back from the environment. Since the user's motion, like walking and breathing, will alter how RF reflects in the space, the RF reflection encodes rich information about the user. With novel algorithms and machine learning models, we demonstrate the ability of our system to capture different aspects of the sleep accurately, including sleep parameters, sleep stages, respiration and postures during sleep.

Our approach brings multiple advantages: compared to PSG which requires the user to go to hospital and wear sensors all over the body, our approach is much more comfortable and affordable, as the user can just sleep normally at home. And compare to wearable

sensors and user self-reported information, our approach is more accurate, with lower overhead. Users no longer need to manually write down their memories and their sleep will not be interrupted by wearables.

In the following, we briefly describe all the vital signs that can be monitored with our system.

1.2.1 Sleep Parameters

Sleep parameters are usually used for insomnia assessment. They consist of the time between going to bed and falling asleep, or sleep latency (SL), the percentage of sleep time to the time in bed, or sleep efficiency (SE), the total sleep time (TST), and the amount of wakefulness after falling asleep (WASO).

In Chapter 2, we introduce *EZ-Sleep* that measures sleep parameters accurately without any user overhead. It works by automatically detecting the location of the user's bed, identifying when she goes to bed and when she leaves the bed, and monitoring the key insomnia assessment parameters SL, SE, TST, and WASO. Its design involves four components that work together to deliver the application:

(a) Monitoring the user via RF signals: EZ-Sleep uses an FMCW radio and an antenna array to separate RF signals based on the location of the reflecting body. This allows it to track both the location of the user and her breathing, and use both to extract the user's sleep patterns.

(b) Identifying bed location: EZ-Sleep introduces a novel algorithm for automatically detecting where the user goes to sleep. The algorithm first identifies locations where the user stays stationary like chairs, desks, couches and beds. It then classifies these locations as bed vs. non-bed.

(c) Detecting bed entries and exits: EZ-Sleep employs a Hidden Markov Model (HMM), where location measurements act as observations of the hidden state which takes one of two values: in-bed or out-of-bed.

(d) Computing sleep parameters: Once it knows the user is in bed, EZ-Sleep zooms in on the RF signal reflected from the bed region. EZ-Sleep feeds this signal to a deep neural

network model to detect whether the user is asleep.

1.2.2 Sleep Stages

Sleep progresses in cycles that involve four sleep stages: Awake, Light sleep, Deep sleep and Rapid Eye Movement (REM). Different stages are associated with different physiological functions. For example, deep sleep is essential for tissue growth, muscle repair, and memory consolidation, while REM helps procedural memory and emotional health.

In Chapter 3, we introduce *RF-Sleep* that can robustly estimate sleep stages of the user only by analyzing reflected RF signals. It involves a new model that delivers a significantly higher accuracy and generalizes well to new environments and subjects. The model adapts a convolutional neural network (CNN) to extract stage-specific features from RF spectrograms, and couples it with a recurrent neural network (RNN) to capture the temporal dynamics of sleep stages.

To enhance its generalizability, we introduce a new adversarial training regime that discards extraneous information specific to individuals or measurement conditions, while retaining all information relevant to the predictive task –i.e., the adversary ensures conditional independence between the learned representation and the source domains.

Our training regime involves three players: the feature encoder (CNN-RNN), the sleep stage predictor, and the source discriminator. The encoder plays a cooperative game with the predictor to predict sleep stages, and a minimax game against the source discriminator. Our source discriminator deviates from the standard domain-adversarial discriminator in that it also takes as input the predicted distribution of sleep stages in addition to the encoded features. This dependence facilitates accounting for inherent correlations between stages and individuals, which cannot be removed without degrading the performance of the predictive task.

1.2.3 Respiration

Respiration is an important health metric used for tracking diseases in many areas, such as sleep [106, 140], pulmonology [117], and cardiology [19, 171]. However, past work on

RF-based respiration monitoring requires monitored people to be located away from each other. This requirement precludes respiration monitoring during sleep for people who share the same bed.

In Chapter 4, we introduce *DeepBreath*, the first system that disentangles mixed RF-based breathing signals. DeepBreath can monitor the respiration signals of multiple people even if they have zero distance between them. It also provides continuous monitoring throughout the night for people who share the same bed.

The design of DeepBreath combines three components as described below.

(a) Breathing Separation: We present an approximation that allows us to formulate the problem as blind source separation and solve it using independent component analysis (ICA). DeepBreath uses its formulation of the mixing process to recover the original breathing signals from such observations using ICA.

(b) Motion Detection: DeepBreath needs to automatically detect movements and run its breathing separation module on each stable period separately. We use a convolutional neural network that is trained to identify movements of the monitored people.

(c) Identity Matching: The recovered breathing signals with ICA do not have identity. Yet identity of the breathing segments is needed to reconstruct full night breathing signal. We formulate this problem as an optimization problem, where the solution maximizes a similarity metric between the person's breathing signals across different periods.

1.2.4 Sleep Posture

Significant clinical research has shown that sleep posture is a valuable marker of disease progression, and has a significant impact on health. For instance, supine position (facing upwards) is considered the worst posture for sleep apnea patients as it imposes unfavorable airway geometry and reduces lung volume [95, 104].

In Chapter 5, we present *BodyCompass*, the first RF-based system that provides accurate sleep posture monitoring overnight in users' own homes. The key enabler of the system is to use the multipath effect, a known phenomenon in RF communication systems that refers to the fact that RF signals bounce off different objects and obstacles in the envi-

ronment and reach the receiver through multiple paths. As the RF signal is incident upon the human body, it reflects from the body based on the body orientation and bounces off the surrounding objects and walls creating a multipath signature indicative of the body posture. Our system learns an inverse map that receives the reflected multipath profile and attempts to infer the body posture.

1.3 Thesis Contributions

This thesis and its core modules make the following key contributions:

1. EZ-Sleep (Chapter 2) is the first RF-based sleep sensor that automatically detects bed locations, identifies bed entries and exits, and monitors key insomnia parameters like sleep latency, sleep efficiency, total sleep time, and wake after sleep onset.

2. RF-Sleep (Chapter 3) learns sleep stages from RF signals and achieves a significant improvement over the state-of-the-art methods. It adapts and extends a CNN architecture to extract stage-specific features from RF reflections, and couples it with RNNs to capture the dynamic evolution of sleep stages.

3. DeepBreath (Chapter 4) is the first system that recovers the detailed respiration signals of multiple people sharing the bed or sitting with zero separation. This result renders much recent research using RF signals to monitor respiration, sleep, and apnea applicable to a significantly larger segment of the population.

4. BodyCompass (Chapter 5) is the first RF-based system that provides accurate sleep posture monitoring overnight in users' own homes. It achieves high accuracy without sacrificing privacy and sleep comfort.

Beyond the above applications, some of the algorithms developed in this thesis apply generally to other applications and data modalities, as explained below.

1. We introduce and analyze a conditional adversarial approach to domain adaptation in machine learning. Our design removes domain-dependent information that is extraneous to the task of interest, while preserving information relevant to the task. In contrast, prior work on domain-adaptation using adversarial learning can remove both extraneous information as well as information relevant to the predictive task. Our conditional adversarial setup is suitable for problems that require aligning domain distributions in situations where labels may be correlated with domains (Chapter 3).

2. We propose a new algorithm for disentangling radio signals received by a Frequency Modulated Carrier Wave (FMCW) radio without knowledge of their corresponding channel information. In particular, FMCW radios can receive radio signals reflected off multiple co-located targets, which interfere with each other and prevent target detection and identification. We prove that for scenarios where the co-located targets exhibit small perturbations, the problem can be reduced to blind source separation, and the targets can be resolved using independent component analysis (ICA). Our analysis can be applied generally to wireless sensing using FMCW (Chapter 4).

3. We propose a dynamic-programming algorithm for identity matching in longitudinal sensing over discrete intervals. Many RF-based sensing scenarios aim to use episodic sensing to track multiple targets longitudinally. This creates a need for matching the RF signal from each target across different episodes. Such matching is difficult in the absence of a priori known RF signature for each target. We formulate this problem as an optimization problem and introduce a dynamic programming algorithm for finding the best match across time (Chapter 4).

4. Data augmentation is a general approach for improving the robustness and generalizabilty of machine learning models. We propose a new approach for data augmentation that we call target-focused data augmentation. It augments data from a target

domain without breaking the spatial structure of the features. The method is helpful when the traditional augmentation methods (e.g., rotation, cropping, flipping) cannot generate valid samples. Instead, we sample training data that are similar to the labeled data from the target domain, and treat it as labeled target data. (Chapter 5).

Chapter 2

Measuring Sleep Parameters

2.1 Introduction

Insomnia and sleep deprivation are common health problems in the US. One in every three Americans do not get enough sleep, and about 10% of the population suffers from chronic insomnia [108]. Chronic insomnia increases the risk of heart disease, kidney failure, high blood pressure, diabetes, and stroke [91]. The statistics are even worse among the elderly, where 50% of seniors experience periods of insomnia that last for weeks, months or even years [108, 163]. In-home sleep monitoring is important for both detecting insomnia and treating it. People's impression of their sleep can be wildly wrong, off by hours in some cases [139, 17]. On the other hand, older patients may not report their insomnia symptoms in the first place [108].

Further, there is an increasing interest in replacing drug-based treatment with Cognitive Behavior Therapy for Insomnia (CBT-I). CBT-I is based on gradual adjustments of sleep schedule and the time spent in bed, and hence in-home monitoring of those parameters is beneficial for treatment tracking [55].

Monitoring insomnia is difficult. The gold standard in sleep monitoring is overnight Polysomnography (PSG), conducted in a hospital or sleep lab, where the subject sleeps with EEG, ECG, EMG, respiration, and pulse monitors. PSG does not work well for insomnia because sleeping away from one's bed with obtrusive sensors causes difficulties in falling asleep that are unrelated to insomnia (the first night effect [123, 8]). In fact, most

insomnia studies use patient diaries, requiring people to keep daily records of when they go to bed, how long it takes them to fall asleep, how often they wake up at night, etc. This approach creates significant overhead and is hard to sustain over long periods. More recently, actigraphy-based solutions have been used to track motion and infer sleep patterns, but they must instrument the user with accelerometers at the wrist or hip [107]. Many people do not feel comfortable sleeping with wearable devices [23], and older adults are encumbered by wearable technologies and may simply take the device off creating adherence issues.

Insomnia and long term sleep monitoring should be zero effort without sacrificing accuracy. It should provide in-home continuous monitoring without requiring the user to wear a device or write a diary. It should also measure the key sleep parameters used for insomnia assessment. This means it needs to measure the time between going to bed and falling asleep, or sleep latency (SL), the percentage of sleep time to the time in bed, or sleep efficiency (SE), the total sleep time (TST), and the amount of wakefulness after falling asleep (WASO).

In this chapter, we introduce EZ-Sleep, a sleep sensor that achieves these goals. EZ-Sleep is zero effort – all that the user has to do is to put EZ-Sleep in her bedroom and plug it to the power outlet. EZ-Sleep works by transmitting radio frequency (RF) signals and listening to their reflections from the environment. By analyzing these RF reflections, EZ-Sleep automatically detects the location of the user's bed, identifies when she goes to bed and when she leaves the bed, and monitors the key insomnia assessment parameters SL, SE, TST, and WASO. Further, because it directly measures the user's bed routine, i.e., when she enters and exits the bed, EZ-Sleep can be used in CBT-I to monitor patient compliance with prescribed changes in her bed schedule.

The design of EZ-Sleep builds on recent advances in wireless systems, which show that by transmitting a wireless signal and analyzing its reflections, one can localize a person and track her vital signs without any wearables [3, 4]. However, past solutions that leveraged these advances in the context of sleep have limited themselves to analyzing the user's vital signs (mainly breathing) as extracted from the RF signals [120, 87, 147, 179]. In contrast, EZ-Sleep uses the RF signal to extract both the user's breathing and location, and combines both to infer the user's bed schedule and sleep quality. In the absence of information on

when the user goes to bed and leaves the bed, past work cannot compute key insomnia parameters like sleep latency, sleep efficiency and WASO. Furthermore, most past solutions rely on the Doppler effect which is highly sensitive to interference from other sources of motion in the environment such as potential neighbors or flatmates (see Sec. 2.9.1 for empirical results). In contrast, by leveraging RF-based localization, EZ-Sleep is not only robust to motion in the environment but can also monitor the sleep of multiple subjects at the same time.

Figure 2-1: EZ-Sleep setup in one of our subjects' bedroom.

We designed EZ-Sleep as a standalone sensor as shown in Fig. 2-1. Our design involves four components that work together to deliver the application:

- *Monitoring the user via RF signals:* EZ-Sleep uses an FMCW radio and an antenna array to separate RF signals based on the location of the reflecting body. This allows it to track both the location of the user and her breathing, and use both to extract the user's sleep patterns.

- *Identifying bed location:* EZ-Sleep introduces a novel algorithm for automatically detecting where the user goes to sleep. The algorithm first identifies locations where the user stays stationary like chairs, desks, couches and beds. It then classifies these locations as bed vs. non-bed. To do so, EZ-Sleep leverages RF-based localization to analyze how the user uses the space. It builds a map of where the user spends her time. Using tools from image processing and machine learning, EZ-Sleep segments the map into meaningful areas.

- *Detecting bed entries and exits:* Knowing when the person goes to bed is essential for both assessing insomnia and monitoring compliance with CBT-I. But, since RF-localization error can be as high as one meter, we cannot simply rely on the location estimate to detect when the person goes to bed. Instead, EZ-Sleep employs a Hidden Markov Model (HMM), where location measurements act as observations of the hidden state which takes one of the two values: in-bed or out-of-bed.

- *Classifying sleep and awake periods, and computing sleep parameters:* Once it knows the user is in bed, EZ-Sleep zooms in on the RF signal reflected from the bed region. EZ-Sleep feeds this signal to a deep neural network model, which operates in two phases. In the first phase it learns to classify the time in bed into sleep and awake epochs. In the second phase, it zooms in on the first transition from awake to sleep, and trains a second model customized for learning sleep onset. Once it knows entry and exit from the bed, the sleep onset, and any later awakenings, EZ-Sleep can compute all of the insomnia assessment parameters described above.

We have implemented EZ-Sleep and evaluated it in 8 homes. We collected a total of 100 nights of sleep, including 30 nights with an EEG-based FDA-approved sleep monitor [136]. Our results show correct bed identification in all places. The results also show that EZ-Sleep is highly accurate. Specifically, its average errors in computing SE, SL, TIB, TST, and WASO are 2.8%, 4.9 min, 3.2 min, 10.3 min, 8.2 min respectively. These results show that EZ-Sleep's accuracy is comparable or better than medical grade actigraphy devices [34, 28, 80].[1] Thus, EZ-Sleep can deliver clinically meaningful sleep parameters without asking the user to wear any sensor or record her sleep data. Furthermore, the results also show that EZ-Sleep can monitor the sleep of two subjects simultaneously, making it the first RF-based sleep monitor that works with multiple people.

Contributions: To our knowledge, EZ-Sleep is the first RF-based sleep sensor that automatically detects bed locations, identifies bed entries and exits, and monitors key insomnia parameters like sleep latency, sleep efficiency, total sleep time, and wake after sleep onset.

[1]Medical grade actigraphy errors are as follows: SE: 3~5%, SL: 0.2~3 min, TST: 8~19 min, and WASO: 12 min [34, 28, 80]. Since actigraphy cannot measure TIB, these estimates require the user to push a button to indicate going-to-bed and leaving-bed.

It is also the first RF-based sleep sensor that can monitor multiple users simultaneously. These properties are enabled by the observation that sleep monitoring can benefit from combining location tracking with temporal analysis of the user's breathing signal, both extracted from RF reflections. We introduce new algorithms that use tools from image processing and machine learning to analyze the user's location and breathing signals and infer her bed schedule and sleep quality. We further implement our design as a standalone sensor which we deploy and evaluate in eight homes, demonstrating its high accuracy.

2.2 Background and Related Work

We first provide background on insomnia monitoring and then describe past solutions, both in industry and academia.

2.2.1 Key Insomnia Parameters

Insomnia and sleep deprivation are typically assessed using the following sleep parameters [80, 130]:

- *Sleep latency (SL)* is the time between going to bed and falling asleep. Sleep latency is perhaps the most common metric for assessing insomnia. A sleep latency that is longer than 30 minutes for more than two nights per week is a sign of insomnia.
- *Total sleep time (TST)* is the total time in bed actually spent in sleep. It captures whether the person gets enough sleep.
- *Time in Bed (TIB)* which is also called time from light out to light on. It puts an upper bound on the sleep time and can show that the person does not allow herself enough time to sleep.
- *Sleep efficiency (SE)* is the ratio of the total sleep time to the total time in bed, (i.e., TST/TIB).
- *Wake after sleep onset (WASO)* is the total duration of wakefulness occurring after sleep onset. It captures sleep fragmentation, or the inability to sustain sleep. For

example, some older people wake up after a few hours of sleep and cannot go back to sleep.

- *Number of awakenings:* This metric counts the number of awakenings that last for more than 5 minutes (NA > 5). In this chapter, we experiment with healthy individuals, and have not had any reports of night awakenings. Hence, we do not report results for this metric. However, our design can compute this metric because it classifies all 30-second epochs as awake or asleep, as explained in section 2.7.

Figure 2-2 shows the relation between these sleep parameters.

Figure 2-2: The definition of all sleep parameters.

2.2.2 Sleep Monitoring Solutions

Laboratory-based Polysomnography (PSG) is the medical gold standard for sleep monitoring. It requires the person to spend the night in a sleep lab connected to a dozen sensors, including EEG scalp electrodes, an ECG monitor, a respiratory chest band, a nasal probe, etc. PSG monitors many aspects of sleep including sleep stages, sleep parameters, and sleep apnea. Yet, PSG is both highly obtrusive and impractical for long term studies [71]. The discomfort with the sensors changes the user's sleep behavior, making it hard to measure insomnia. Further, since night-to-night sleep variability is a important factor for diagnosing insomnia [15], a single night of study is often insufficient.

Patient diaries are commonly used for monitoring insomnia. The person is asked to keep daily records of when they go to bed, how long it takes them to fall asleep, how often they wake up at night, etc. Writing sleep diaries requires a significant effort and it is hard to sustain for long periods of time. Medical grade actigraphy has been used to track user movement to infer sleep. Actigraphy measures user motion using accelerometers tied to

a person's wrist or hip. However, some people feel encumbered to sleep with wearable devices [23] , and older people and kids may simply take the device off during the night.

The consumer industry has developed wellness devices that track sleep. They encompass activity trackers such as FitBit and Jawbone, smartphone-based solutions like Sleepbot and Sleep Cycle, bed side sensors or bed pads inserted under the sheets such as Beddit [42, 67, 133, 134, 12]. These solutions have lower accuracy than the medical grade devices [52]. Some of these solutions require the user to input when they are physically in bed [141] or may have lower accuracy if this information is missing [12].

The research community has shown a great interest in monitoring sleep using smartphones or systems that sense environmental factors [24, 23, 54, 98, 56, 69]. They infer sleep quality using data from microphones, accelerometers, cameras, phone usage, etc. However, recording audio and video information throughout the night may still be considered obtrusive for some people due to privacy and comfort reasons [120]. Also, these systems may not be easy to configure for certain sectors of the populations, like the elderly or children.

In terms of technology, EZ-Sleep is closest to past work on using radio signals for monitoring sleep stages or in-bed movements [120, 82, 87, 147, 179]. All of these systems work by extracting a person's breathing using radio signals that bounce off the person's body. They then classify the night into periods that correspond to different sleep stages or body movements. EZ-Sleep differs from these systems along four axes. First, EZ-Sleep analyses sleep behavior both in time and space. It does so by combining RF capabilities to extract breathing and location. Second, EZ-Sleep computes sleep parameters unavailable to past systems including TIB, SL, and WASO. Third, EZ-Sleep introduces new algorithms that automatically identify where a person sleeps and track her bed entries and exists. Fourth, EZ-Sleep can simultaneously monitor multiple users with one device.

2.3 EZ-Sleep Overview

EZ-Sleep is an in-home sleep monitoring system that requires zero user effort. It operates by transmitting a low power wireless signal and capturing its reflections off users in the

vicinity of the device. By analyzing the signals, it learns the bed location, when the user goes to bed, and when she falls asleep. Based on this knowledge, it generates sleep parameters including Time in bed (TIB), Sleep latency (SL), Total sleep time (TST), Sleep efficiency (SE), and Wake after sleep onset (WASO).

The EZ-Sleep sensor is a software-hardware system whose operation involves the following key steps:

1. Capturing the user's location and breathing using RF signals.
2. Inferring bed areas by observing the user's movements through space.
3. Detecting bed entry and exit events by interpreting noisy location measurements.
4. Classifying sleep versus awake when the person is in bed and estimating sleep parameters.

In the following sections, we explain each step in detail.

2.4 Extracting Location and Breathing from RF Signals

The relationship between motion and sleep is intuitive and constitutes the core of all actigraphy-based sleep sensors. But sleep sensors, so far, have looked only at the amount of motion over time – i.e., actigraphy. We observe however that some sleep parameters, such as going-to-bed, can be better inferred by analyzing motion as a function of space. Wireless signals are a powerful tool to capture a person's motion both in time and space. When the person is stationary, her motion over time is mostly her breathing motion [5]. Thus, in designing EZ-Sleep, we leverage these RF capabilities. The literature already shows the capacity of RF signals to capture both location [4, 3, 153, 72] and breathing [5, 120, 155]. We introduce these techniques briefly and refer the reader to [3, 5] for more information.

EZ-Sleep uses a combination of an FMCW radio and an antenna array. The FMCW technique measures the distance of the reflecting body from the device, whereas the antenna array measures the spatial direction of the reflector with respect to the device. Thus, together, they allow us to divide the x-y plane into pixels and separate RF reflections from different pixels in space.

Once we have separated the signal from each x-y pixel, we need to first identify human reflections from reflections of other objects in the environment such as walls and furniture. To do so, we apply two different filters to the signals to capture two types of human motions. The first is a high pass filter that captures fast and non-periodic human motion, including walking and any hand or limb movements. The second is a band pass filter around the breathing frequency that captures a user's chest motion when she is breathing and not actively moving other body parts. Note that since furniture and walls neither move at a periodicity similar to human breathing, nor have non-periodic motion, their reflections disappear at the output of both filters. Applying these filters to the signal allows us to detect whether the x-y pixel has a human at that time, and whether the human is only breathing or he is performing a bigger non-periodic motion.

In our implementation, we generate one measurement per pixel every 50 milliseconds. Each measurement contains a time sample of the RF signal reflected from that pixel at that time, and a tag that states whether the pixel has a moving person, a stationary person, or is empty (i.e., has no person). We project the area covered by EZ-Sleep into a 500-by-500 pixel image, where each square pixel occupies a 0.024-by-0.024 square meter space.

Finally, we note that RF-based localization suffers from environment noise and multipath problems, and its 90 percentile error can be more than one meter [3, 72, 153]. Similarly, breathing signals extracted from RF reflections are highly sensitive to body motion, and can be erroneous when the user moves her limbs [5]. EZ-sleep is designed to be robust to these phenomena and can measure sleep parameters accurately in the presence of noise.

2.5 Bed Identification

In this section, we describe EZ-Sleep's algorithm for detecting beds in the home. The algorithm first identifies areas in the home where the user is mostly stationary. Such areas span the bed and various seating places such as chairs and couches. Next, EZ-Sleep identifies which of these stationary areas are beds.

Potential bed areas are places where people stay stationary for long periods of time. Thus, we would like to identify areas in the home where the radio sees the person station-

ary for significant periods. Recall that at every 50 millisecond, we have a tag that identifies every pixel as being stationary, moving, or empty. Thus, the first step of our algorithm computes a 2D histogram of the stationary pixels in the x-y plane. The histogram is computed every day. Regions where the person sits or lies down will accumulate many stationary tags and hence will show up as peaks in the histogram.

Fig. 2-3 shows example histograms from two of the homes where we deployed EZ-Sleep. The locations of EZ-Sleep are shown as the blue rectangles and the black lines indicate walls or boundaries of beds or desks. The darker the pixels are the more stationary time is spent in that location. The figure reveals three observations: First, the distribution of stationary pixels does indeed reveal the location of actual seating or sleeping areas. For example, the dark regions in the figure refer to real beds, couches and desks. Second, EZ-Sleep can detect more than one bed and can identify beds and seating areas outside the room where the device is installed. Third, the stationary pixels do not span the whole bed region. This is because most people sleep on a particular side of the bed. Besides, stationary pixels correspond to locations where there is some breathing signal – i.e., the chest, face, and abdomen. This description of the sleeping area is more relevant to our task than the whole physical space of the bed. Fourth and most importantly, the stationary regions can be connected in the histogram. For example, one of the beds in the figure appears connected to the desk. Simply by looking at the histogram, EZ-Sleep cannot tell whether this is one big bed or two distinct stationary areas. Thus, our next step is to separate the distinct seating or sleeping areas observed by the device.

In the second step, our algorithm takes the 2D histogram as input and extracts distinct seating or sleeping areas. We frame this problem as an image segmentation problem – i.e., we would like to assign each pixel a label such that pixels with the same label share the same area identity (i.e., desk, couch, or bed). Common approaches to image segmentation take initial labels of a few pixels, and finds boundaries dividing regions with different initial labels. Such initial labels are called markers. But, how can we assign initial labels to the images in Fig. 2-3a such that the bed and the desk have different labels automatically? Simply using local maxima as initial distinct labels will lead to over segmentation because each area can have multiple local maxima. Smoothing the image first does not help because

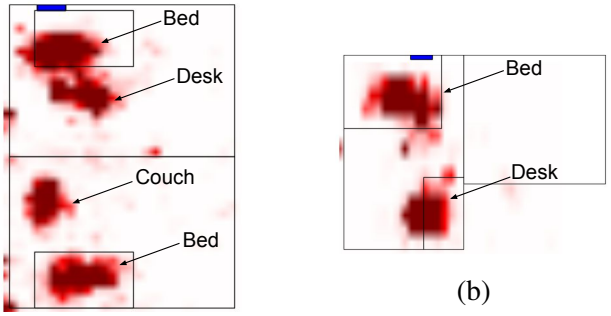


Figure 2-3: Images of the distribution of stationary location measurements from two homes. The locations of EZ-Sleep are shown as the blue rectangles and the black lines indicate walls or boundaries of beds or desks. Dark regions indicate areas where people stay stationary for long periods of time, such as beds, desks or couches.

it blurs region boundaries.

To assign initial labels for the segmentation algorithm, we use a processing pipeline explained in Fig. 2-4. To estimate the initial labels, we binarize the input image (Fig. 2-4a) into foreground and background (red and white regions in Fig. 2-4b) using Otsu thresholding [111]. In the binarized image (Fig. 2-4b), the desk and the bed are still connected. We can make the distinction between two areas bigger by computing the distance transform, i.e., the nearest distance to the background, for each pixel of this image. Given the transformed image (Fig. 2-4d), we can apply a threshold to get the central regions of both the desk and the bed (Fig. 2-4e). Note that to set a threshold that generalizes to different areas in different homes, we normalize the transformed image with respect to the size of each connected component (Fig. 2-4c) before thresholding. After thresholding, the connected components of the thresholded image (Fig. 2-4e) can be used as the initial labels for the segmentation algorithm. Fig. 2-4f shows the final segmentation result using Watershed algorithm [97]. We reject regions whose area is too small to be a bed. The remaining regions are the potential bed areas.

The third and last step in our algorithm is to identify beds from other areas where the user may spend some time while stationary, like a couch or a desk. People may spend long time in areas other than beds. Using only the amount of time spent in an area as the classifying feature is not enough. Also, people's schedule changes from day to day. A user

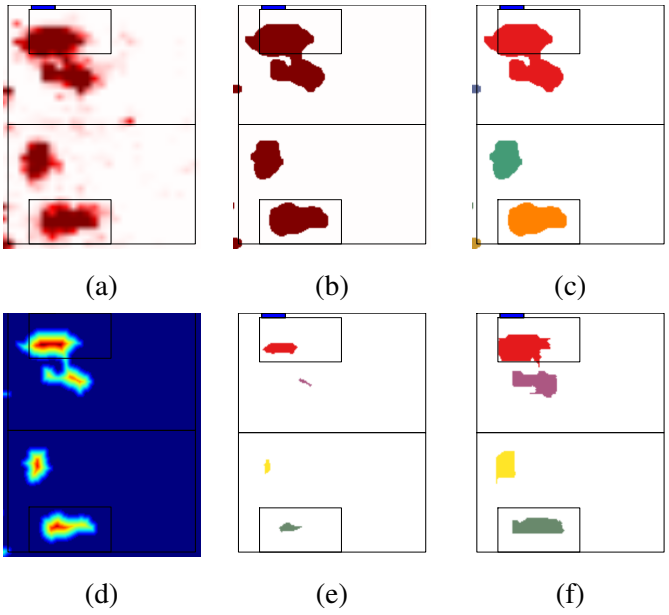
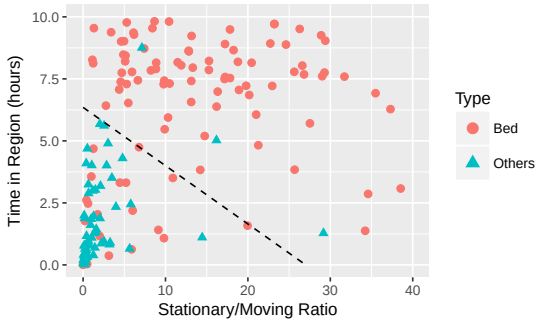


Figure 2-4: Estimating potential bed areas. The image of the stationary location distribution is shown in (a). This image is binarized using Otsu thresholding, resulting in (b) where the red pixels form the foreground. To obtain the markers for watershed, we first compute the connected components as shown in (c). Computing a distance transform results in (d) which is thresholded to obtain the markers in (e). The watershed segmentation outputs the potential bed areas as shown in (f).

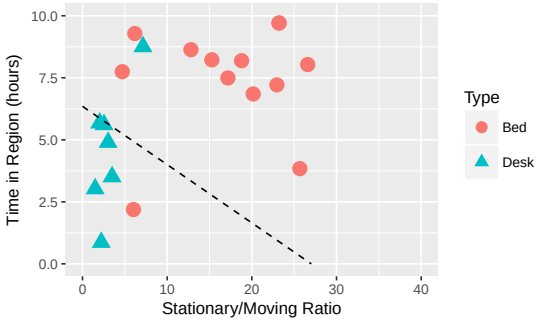
can spend more time on a couch watching TV than sleeping in the bed during the weekend. Similarly, before a deadline a user can spend more time working at her desk than sleeping in her bed. The task is further complicated by the fact that we want to discover multiple beds in the same house and different people may have very different schedules.

To address these problems, we analyze the temporal activities in each potential bed area to extract additional features and look at the consistency of features across multiple days. Specifically, in addition to the amount of time the user stays in an area, we also use the ratio of time being stationary versus moving. To compute this feature, we take the sum of all stationary tags for all pixels in the area and divide it by the sum of all moving tags for all pixels in the area. Fig. 2-5a visualizes around two weeks of data for 10 potential bed areas identified at the final output of the segmentation pipeline in Fig. 2-4. The two axes refer to the two features: total amount of time, and ratio of stationary time to moving time spent in that area. Each day contributes one data point to each detected area. The circles refer to real beds, whereas the triangles refer to couch and desk areas. We remove data

points where people spent zero amount of time from this figure. We can see that the desk and couch areas form a cluster in the lower left corner. There are outliers as a person may sleep less on some days or a couch may look like a bed occasionally. To better understand this, we highlight data from one bed region and one desk region in Fig. 2-5b. If we look across multiple days, data from the bed and the desk stay in their respective clusters.



(a) Figure shows the two features for 10 potential bed areas. Using the two features, we can separate the bed and non-bed areas.



(b) Figure shows the two features for a specific bed and desk region. We use data across multiple days to classify a region as bed or non-bed.

Figure 2-5: Identifying actual beds.

Given the analysis, we use the above two features to train a bed classifier using SVM with a linear kernel. To determine if a region is a bed, we look at the its predicted labels over the past D days. We identify the region as bed if the percentage of days labeled as bed is greater than γ:

$$\hat{y_i} = \begin{cases} Bed, & if \ \frac{\sum_{d=0}^{D-1} \|f(A_i;d)\|}{D} > \gamma \\ Not\ Bed, & otherwise \end{cases}, \tag{2.1}$$

where $\hat{y_i}$ is the final prediction for area A_i, $f(.)$ is the classifier, $f(A_i; d)$ is the binary predicted results for area A_i on the dth day. Our default is $D = 7$ and $\gamma = 5/7$. Fig. 2-5 shows the classification boundary of our classifier.

2.6 Detecting Bed Entry and Exit

After identifying the bed areas, we consider the problem of detecting when a user goes to bed and when she leaves the bed. RF-based localization returns the location of the people in the environment. However, as mentioned in Section 2.4 their errors can be as large as one meter. An error of plus or minus one meter is pretty much the size of the bed. It can easily move a person from the bed to outside the bed. Similarly, a person who is changing his clothes next to the bed may look already in bed. These localization errors have little impact on detecting the bed region because in that case we can use the histogram of location data over a whole day, which allows us to average out the errors. In contrast, when detecting entries and exits from the bed, we would like to detect them as quickly as possible because any delay in detecting such event will appear as an error in our estimate of sleep latency.

So, how do we accurately track when the user enters or leaves her bed? Instead of directly mapping the measured locations to in or out of the bed, EZ-Sleep considers the location measurements as noisy observations of the true state. The true state –i.e., whether the user is in or out of bed – is hidden. EZ-Sleep uses a Hidden Markov Model (HMM) to infer the true state from the observations. Below we give a short primer on HMM followed by our particular HMM design.

2.6.1 HMM Background

A hidden Markov model is a statistical model that tries to explain a sequence of observations with a sequence of hidden states. Time progresses in steps. In each time step, the model is presented with a new observation which it uses to decide whether to stay in the current state or transition to some other state. The key assumption underlying an HMM is that the probability of an observation given the current state is independent of any other observation or state. Mathematically, let us model the observation at time, t, as o_t and the hidden state as s_t. If $P(X)$ denotes the probability of event X, then the assumption can be represented as:

$$P(o_t|s_0, s_1, s_2, ..., s_{t-1}, s_t) = P(o_t|s_t) \tag{2.2}$$

Thus, the joint probability of having a sequence of states, $\{s_t\}_{t=1}^{N_t}$ and observations, $\{o_t\}_{t=1}^{N_t}$ is given by:

$$P(\{o_t, s_t\}_{t=1}^{N_t}) = P(s_0)P(o_0|s_0)\prod_{t=1}^{N_t} P(o_t|s_t)P(s_t|s_{t-1}) \tag{2.3}$$

To model a problem as an HMM one has to define a set of possible states $\mathcal{S} = \{\mathcal{S}_1, ..., \mathcal{S}_M\}$, a set of observations $\mathcal{O} = \{\mathcal{O}_1, ..., \mathcal{O}_N\}$, a $M \times M$ transition probability matrix $\mathbf{T}$, and a $M \times N$ emission probability matrix $\mathbf{E}$. Transition probability is the probability of transitioning from state, $\mathcal{S}_i$ to $\mathcal{S}_j$ in consecutive time steps, i.e., $\mathbf{T}_{ij} = P(s_t = \mathcal{S}_i|s_{t-1} = \mathcal{S}_j)$. The emission probability is the probability of an observation given a state, i.e, $\mathbf{E}_{ij} = P(o_t = \mathcal{O}_j|s_t = \mathcal{S}_i)$. The set of observations and states is typically picked by the designer, and the transmission probabilities and emission probabilities are learned from the data.

2.6.2 Design of Our HMM

We use the HMM hidden states to represent whether the user is in the bed or outside the bed at each time step t. We denote the state corresponding to being in the bed as $\mathcal{S}_0$ and outside the bed as $\mathcal{S}_1$.

We define our observations by first dividing the space around the bed region into three areas: center area (R_0), buffer area (R_1), and outer area (R_2). The center area is the bed region we found in section 2.5, and the buffer area is a 50-centimeter-wide area that encircles the center area. The outer area is the rest of the space. We define 9 possible observations which correspond to the user transitioning from one of the three areas to another or staying in the same area. Thus, the observations for the HMM are the set of tuples

$$\mathcal{O} = \{(R_s, R_e)\ \forall R_s, R_e \in R_0, R_1, R_2\}\,.$$

To detect which observation has occurred in each time step, we divide time into consecutive windows of 5 seconds each. For each window we detect the location of the user from the moving and stationary pixels. We average the location in each second and map it to one of the three area: R_0, R_1, or R_2. Then R_s denotes the area that corresponds to the first second of the window while R_e is the area that correspond to the last second of the window.

We specify the buffer area around the bed because we are often unsure of the person's exact location. To decide a person's state, we need to look a bit into the past and the future. Having the buffer area helps us encode different beliefs in these observations as opposed to simply dividing the space into inside versus outside the bed. Given the learned bed regions from the last section, this choice of observations does not require any information specific to the home. Thus, we can use the same HMM for all homes. In addition, we transform raw location data into discrete observations for better generalization of our model. If we use a continuous input instead (e.g., the distance to the center of the bed), because people stay at different locations away from beds in different homes, the model trained at few places cannot generalize well to different homes.

If multiple bed regions are detected, we generate different observations for each bed. EZ-Sleep uses mechanisms similar to [3] to track raw locations of users to form observations for the HMM.

Learning and Inference: The goal of the learning algorithm is to learn the transition probabilities matrix, T, and the emission probabilities matrix, E. In the absence of priors, the typical way to learn T and E is to compute their statistics using labeled data. Thus, we manually label the training data with the correct states and use them to learn T and E.

After the learning phase, the model can be used independent of the environment given the bed region. In the inference phase, we give the HMM the sequence of observations for one day and find a sequence of states that best explain those observations using the Viterbi algorithm [44]. This means that we can give the HMM the observations extracted from the areas around each detected bed, and have the HMM predict for us when a person enters the bed (i.e., the state transition from $\mathcal{S}_1$ to $\mathcal{S}_0$) and when she exits the bed (a transition from $\mathcal{S}_0$ to $\mathcal{S}_1$).

2.7 Estimating Sleep Parameters

So far, we have learned the bed area and when the user enters and leaves the bed. To estimate the sleep parameters in Section 3.2, we still need to detect when the user falls asleep, and the sleep-awake intervals throughout a night. To do so, we divides the time in

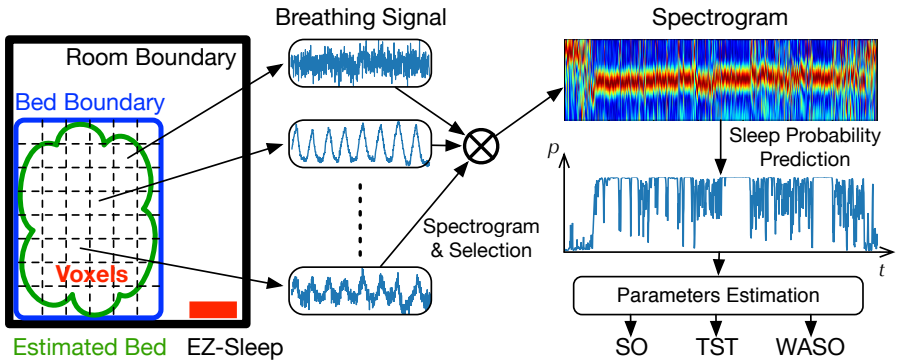


Figure 2-6: Sleep parameters estimation system. Breathing signals are collected from bed pixels, and provided to the neural network classifier as different channels. The classifier outputs the probability of sleep at each time epoch. Finally, a second layer estimation algorithms take sleep probability as input, and output sleep parameters: Sleep Onset Time (SO), Total Sleep Time (TST) and Wake After Sleep Onset (WASO).

bed into 30-second epochs, as typical in sleep studies. We would like to classify each epoch into sleep vs. awake and identify the first epoch in which the person falls asleep, which is the sleep onset.

The information of whether a person is asleep or awake is encoded in her breathing pattern and body movements [120, 147, 54]. To extract this information, we need to ensure that the RF signal that we provide to the classifier, in each epoch, reflects the user's breathing and movements and is filtered from any source of interference. This is challenging because there are many other sources of motion in the environment, e.g., rotating fans, neighbors whose movements may impact the signal, and even curtains moving due to air conditioning. These extraneous movements can impact the signal and completely overshadow the small movements caused by human breathing. As we show in Section 2.9.1, having the neighbor walking in another room from behind the wall can easily interfere with the breathing signal.

To separate breathing from extraneous interference, EZ-Sleep uses the identified bed areas as spatial filters to filter out motion coming from outside the bed. Specifically, we zoom in on the pixels in the bed areas and analyze only the signals reflected from those pixels. For each epoch, we consider the signals reflected from pixels tagged as stationary or moving (i.e., non empty). We use these signals as estimates of the user's breathing pattern and body movements.

For our classifier, we build on recent success of convolutional neural network (CNN) in classifying time signals [1]. The benefit of this classifier is that we do not need to pick features manually. We can directly pass the signal in each epoch to the classifier. We build a CNN classifier with a 14-layer residual network model [57]. Our network architecture is similar to the 18-layer network described in [57] but with no repetition for the conv4 and conv5 layers. The classifier takes the spectrogram of the signal in each epoch as input, and outputs the probability of the person being asleep. Hence for each night, we obtain a series of probabilities $\{p_i\}_n$, where n is the total number of epochs for that night, and i is the i_{th} epoch. We use this CNN classifier as a building block for estimating sleep onset time, Total sleep time (TST), and Wake after sleep onset (WASO).

Sleep Onset Time: Predicting the onset of sleep, using the CNN classifier alone would not give us enough accuracy. The reason is that in training the CNN, the model tries to maximize the overall accuracy of predicting asleep-awake for the entire night, but not the accuracy of detecting the exact sleep onset time. Thus, we use this first CNN classifier as the first phase in detecting sleep onset. Specifically, for each night, we consider the first epoch in which the probability of sleep p_i is larger than 0.5. We then take a window of 15 minutes before and 15 minutes after that epoch. This gives us the sequence of epochs around the sleep onset time.

We build a Gradient Boosting Regressor on top of the above CNN with special focus on the sleep onset transition. The regressor takes as input the signal in the epochs in the above window to learn the exact sleep onset. In this case, we would like to learn which epoch is the sleep onset. Since this function is an impulse function, it cannot be directly learned by the regressor. It has to be smoothed first. Thus, we smooth it by convolving it with a Gaussian kernel. For each epoch, the regressor predicts the probability of being the sleep onset epoch. The epoch with the largest probability score is considered the sleep onset epoch.

TST and WASO: With the predicted sleep probabilities $\{p_i\}_n$ and sleep onset time κ, we can estimate TST and WASO. Specifically, TST is the total duration of all epochs for which $p_i > 0.5$, starting after epoch κ. Similarly, WASO is the total time durations of all epochs

for which $p_i \leq 0.5$ starting after epoch κ.

SE and SL: Finally, sleep efficiency (SE) is computed directly as TST/TIB, and sleep latency (SL) is the time difference between the last entry to bed before epoch κ and the beginning of epoch κ.

We use an implementation of residual networks from [73]. Our network architecture is similar to the 18-layer network described in [57] but with no repetition for the conv4 and conv5 layers. For estimating sleep onset time, we use the default parameters for Gradient Boosting Regressor from the scikit-learn library. The standard deviation of the Gaussian kernel is chosen to be 3 epochs (90 seconds).

2.8 Evaluation Setup

We evaluated EZ-Sleep through actual deployments in 8 homes. We collected more than 100 nights of sleep data from 10 healthy subjects whose age spans 23 to 45 years. Out of these, all subjects slept with SleepProfiler [136, 143], which is an FDA-approved medical grade sleep monitoring device, for a total of 30 nights to obtain the ground truth for our sleep parameters. We install the EZ-Sleep device in the bedroom of the subject. Figure 2-8 shows the location of the device in all 8 homes. In our deployments, two of the homes have two beds in the device's coverage area. All subjects have been consented in accordance with our IRB.

SleepProfiler (used to obtain the ground-truth) has three frontal EEG electrodes to measure brain activity, accelerometers for detecting motion, a chest band for breathing, and a pulse rate sensor for monitoring heart rate. The device has a push button for the user to indicate when she goes to bed and when she leaves the bed. The subjects are instructed to push the button when they enter and leave the bed. Other parameters can be extracted directly from the sleep report provided by the device. The subject may make mistakes in setting up the sleep profiler or wearing the sensors. For example, they may forget to attach the adhesives that prevent the EEG electrodes from moving, or remove the sensors and go back to sleep. To ensure that all nights considered in the study do not have such errors, we ask the subjects to keep a sleep diary, in which they record when they go to bed and

when they wake up, whether they experienced any awakenings during the night, and other comments that regarding removal of the device. The diary is used to check for consistency with the sleep profiler data and exclude inconsistent nights.

The ground truth data is also used to train the classifiers. Training and testing are done on different people and different homes –i.e., for each home, we test a classifier trained on the other homes.

2.9 Results

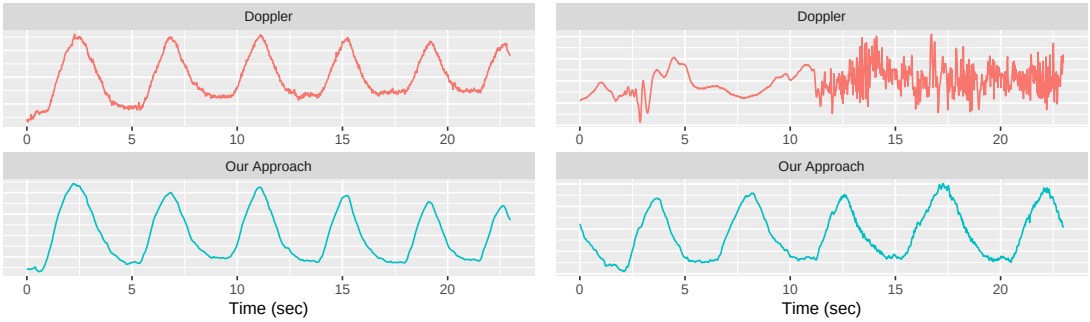
Below we start by showing that EZ-Sleep is more robust to interference than past solutions that use RF-based signals but rely on the Doppler effect. We then evaluate EZ-Sleep's ability to detect the bed region and extract the sleep parameters.

2.9.1 Sensor Robustness

There are two benefits for using the spatial information extracted from RF-signals. The first is the ability to detect where and when the user goes to bed. The second, is the fact that we can separate the RF reflections from different pixels in space. This latter property allows EZ-Sleep to eliminate interference from other sources of motion, such as fans, neighbors, etc. In this section, we demonstrate empirically the added robustness due to this spatial filtering. We also provide mathematical reasons why Doppler-based solutions common in past work [120, 82, 87, 141], can be easily confused in the presence of extraneous motion.

We did an experiment where we ask a person to lie on a bed in the same room as the radio device. We ask the person to be stationary, so that we can focus only on his breathing motion. The RF signal is measured both using the EZ-Sleep radio and a similar radio, where the only difference is that we replace the FMCW and antenna array setup with measuring the Doppler effect as described in [120]. Figure 2-7a shows that both the Doppler radar and EZ-Sleep capture the person's breathing when he is the only source of motion in the environment. We repeat the same experiment, but this time with another person present in an adjacent room. Figure 2-7b shows the results. The second person is stationary in the first half of the experiment, and starts walking around the 12th second. For the Doppler-based

approach, even before the second person starts walking, his presence disturbs the signal making it hard to track the breathing of the person in the bed. This gets worse when the person starts walking, as shown in the second half of Figure 2-7b. In contrast, EZ-Sleep's signal stays clean and clearly reflects the breathing of the subject in the bed.



(a) Single Person Scenario (b) Two Person Scenario

Figure 2-7: EZ-Sleep vs. Doppler based approaches. The signal using the Doppler effect gets disturbed in the presence of other motion in the environment.

The above results can be explained as follows. A Doppler-based sensor transmits a single frequency and detects the chest displacement $x_0(t)$ by looking at the baseband signal $B(t)$ at the receiver [120]:

$$B(t) = A_0 cos(\frac{4\pi x_0(t)}{\lambda} + C_0), \tag{2.4}$$

where A_0 is proportional to the amplitude of the reflection, λ is the wavelength, and C_0 is a constant that depend on the person's distance from the device. The chest motion $x_0(t)$ can be inferred from the phase of the signal $B(t)$. However, in the presence of another source of motion with displacement $x_1(t)$, the baseband signal becomes:

$$\widetilde{B}(t) = A_0 cos(\frac{4\pi x_0(t)}{\lambda} + C_0) + A_1(t) cos(\frac{4\pi x_1(t)}{\lambda} + C_1(t)) \tag{2.5}$$

where $A_1(t)$ is proportional to the amplitude of the second reflection and $C_1(t)$ is related to the second person's distance. Without knowing how the second person moves, both $A_1(t)$ and $C_1(t)$ are unknown, and one can no longer infer $x_0(t)$. Moreover, since both terms could change over time, estimating the periodicity of $x_0(t)$ becomes challenging. Hence,

Doppler-based approaches cannot estimate the breathing rate accurately in this scenario. In contrast, combining FMCW with an antenna array (Section 2.4) allows EZ-Sleep to separate the two reflections using the fact that they come from different spatial pixels. Therefore, we can extract cleaner breathing signal, as shown in Figure 2-7b, even in the presence of other people.

2.9.2 Identifying Bed Areas

We compare the bed areas identified by EZ-Sleep with the ground truth location of the bed. The ground truth is obtained using careful measurements of the location of the bed and the layout of each home using laser distance meters with an accuracy of $0.06"$. To train our bed model we divide the data into two sets, each cover 4 homes. We train the model on one set and test it on the other. We then swap the training and test sets. Thus, the model is asked to predict the bed area for new homes that it did not see in the training phase.

Figure 2-8 shows the results of our bed identification. The figure shows the floor plans from all 8 homes. The location of the EZ-Sleep device is illustrated using a blue rectangle. We label the ground truth bed areas using the red boxes and show the bed areas identified by EZ-Sleep in dark green. Note that in the homes in (c) and (g), EZ-Sleep monitor two beds simultaneously.

As shown in Figure 2-8, EZ-Sleep correctly identified the bed areas for all 8 homes, even at places where there are two beds in the coverage area (2-8c and 2-8g). Note that the identified bed area usually covers a subregion of the physical bed. As explained in Section 2.5, these areas are obtained using the location data when the person is stationary and breathing is the only motion. These areas represent the chest, abdomen and face of the person. Having this ability to accurately estimate the spacial pixels that reflect the user's breathing allows us to zoom in on the subject during their sleep to estimate the sleep parameters avoiding a scan of a larger 3D space.

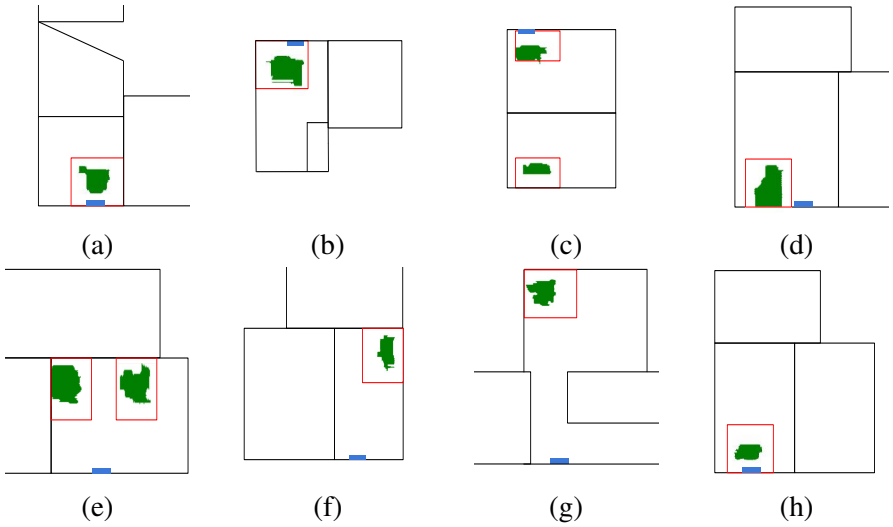


Figure 2-8: Bed areas identified by EZ-Sleep. Locations of EZ-Sleep are illustrated with blue rectangles. The red boxes represent the ground truth bed areas and the green areas show the identified bed areas.

2.9.3 Accuracy of Sleep Parameters

In this section, we evaluate EZ-Sleep's accuracy of detecting bed entry and exit events, and estimating the various sleep parameters. We distinguish scenarios in which the sensor monitors one subject and one bed from scenarios in which the sensor monitors two subjects in two beds.

Single User Scenarios

Let us start by reporting the accuracy of detecting bed entry and exit events and the resulting time in bed (TIB). As explained earlier, EZ-Sleep uses an HMM based approach to identify bed entries and exits. The average error of the HMM in detecting bed entries and exits are 1.8 minutes and 1.3 minutes, respectively. The entries and exits are used to compute the TIB, which has an average error of 3.15 minutes.

To show the individual errors for all subjects and all nights, we show in Figure 2-9 a scatter plot of the predicted TIB values versus the actual TIB values. The TIB values in our dataset range from 273 minutes to 564 minutes. Most of the points in Figure 2-9 lie on the diagonal line. This shows that EZ-Sleep achieves high accuracy of TIB estimation across a wide range of sleep schedules.

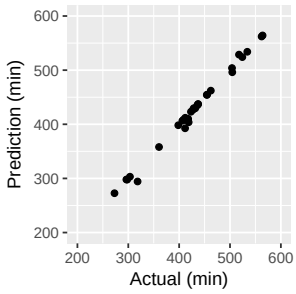
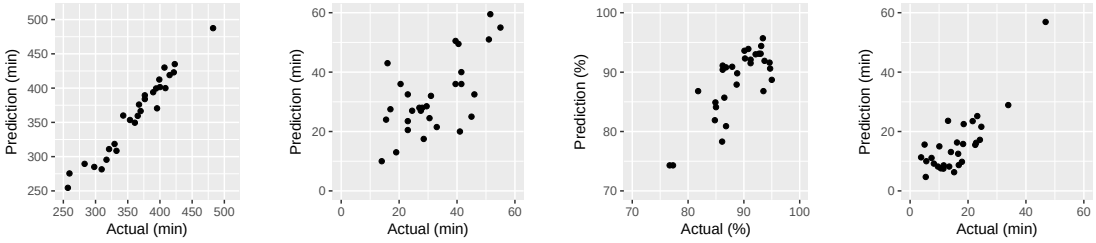


Figure 2-9: Predicted versus actual Time in Bed.

Next, we compare the other sleep parameters predicted by EZ-Sleep to their ground truth values. Figure 2-10 shows scatter plots of the predicted sleep parameters vs. their ground truth values taken over all subjects and all nights. Table 2.1 summarizes the statistics in the scatter plots and presents the average, mean and standard deviation of the prediction error, for the various sleep parameters. The table shows that EZ-Sleep has high accuracy. Specifically, its average error in predicting TST, SL, SE, and WASO is 10.3 min, 4.9 min, 2.8%, and 8.2 min, respectively. These results are comparable to medical grade actigraphy-based insomnia monitors and within the clinically meaningful ranges [80].

Table 2.1: Accuracy of Sleep Parameters for Single User Scenarios

Sleep Parameter	Average Error	Median Error	Std of Error
Time in Bed (TIB)	3.15 (min)	0.14 (min)	6.11 (min)
Total Sleep Time (TST)	10.3 (min)	8.5 (min)	7.7 (min)
Sleep Latency (SL)	4.9 (min)	4.3 (min)	3.1 (min)
Sleep Efficiency (SE)	2.8 (%)	2.6 (%)	2.1 (%)
Wake After Sleep Onset (WASO)	8.2 (min)	6.2 (min)	8.5 (min)



(a) Total Sleep Time (b) WASO (c) Sleep Efficiency (d) Sleep Latency

Figure 2-10: Scatter plots of the predicted sleep parameters versus their ground truth values.

Monitoring Multiple Users

Finally, we show that EZ-Sleep can simultaneously monitor multiple subjects sleeping in their corresponding beds. We have two homes where the RF signal covers two beds, as shown in Figure 2-8c and 2-8g. Table 2.2 presents the error statistics for simultaneously monitoring two people sleeping in the same home. The table shows that the errors in TST, SL, and WASO are a few minutes higher than in the case of a single person. This is expected because while mechanisms like antenna array separate signals from different directions, the separation is not perfect (due to the array's side lobes). Thus, the presence of a second person can add a small disturbance to the signal from the other person, particularly if the two people are close to each other. However, the errors are still small and comparable to medical grade actigraphy [80]. This result is particularly interesting since it shows that EZ-Sleep is the first RF-based sleep sensor that is capable of monitoring multiple people simultaneously.

Table 2.2: Accuracy of Sleep Parameters for Multi User Scenarios.

Sleep Parameter	Average Error	Median Error	Std of Error
Time in Bed (TIB)	0.2 (min)	0.1 (min)	0.2 (min)
Total Sleep Time (TST)	15.8 (min)	15.5 (min)	11.8 (min)
Sleep Latency (SL)	7.6 (min)	7.0 (min)	6.4 (min)
Sleep Efficiency (SE)	1.8 (%)	2.2 (%)	1.2 (%)
Wake After Sleep Onset (WASO)	13.1 (min)	13.5(min)	12.1 (min)

2.10 Conclusion

We have presented EZ-Sleep, a new approach to monitoring insomnia and sleep deficiencies. EZ-Sleep is both accurate and easy to use. All that the user has to do is to put the device in her bedroom and plug it to the power outlet. EZ-Sleep automatically figures out where the user sleeps, and continuously monitors her sleep parameters. It has high accuracy in monitoring sleep latency, sleep efficiency, total sleep time, time in bed, and wake after sleep onset.

Chapter 3

Estimating Sleep Stages

3.1 Introduction

Sleep progresses in cycles that involve multiple sleep stages: Awake, Light sleep, Deep sleep and REM (Rapid Eye Movement). Different stages are associated with different physiological functions. For example, deep sleep is essential for tissue growth, muscle repair, and memory consolidation, while REM helps procedural memory and emotional health. At least, 40 million Americans each year suffer from chronic sleep disorders [103]. Most sleep disorders can be managed once they are correctly diagnosed [103]. Monitoring sleep stages is beneficial for diagnosing sleep disorders, and tracking the response to treatment [18].

Prevailing approaches for monitoring sleep stages are inconvenient and intrusive. The medical gold standard relies on Polysomnography (PSG), which is typically conducted in a hospital or sleep lab, and requires the subject to wear a plethora of sensors, such as EEG-scalp electrodes, an ECG monitor, multiple chest bands, and nasal probes. As a result, patients can experience sleeping difficulties, which renders the measurements unrepresentative [58]. Furthermore, the cost and discomfort of PSG limit the potential for long term sleep studies.

Recent advances in wireless systems have demonstrated that radio technologies can capture physiological signals without body contact [68, 5, 177]. These technologies transmit a low power radio signal (i.e., 1000 times lower power than a cell phone transmission)

and analyze its reflections. They extract a person's breathing and heart beats from the radio frequency (RF) signal reflected off her body. Since the cardio-respiratory signals are correlated with sleep stages, in principle, one could hope to learn a subject's sleep stages by analyzing the RF signal reflected off her body. Such a system would significantly reduce the cost and discomfort of today's sleep staging, and allow for long term sleep stage monitoring.

There are multiple challenges in realizing the potential of RF measurements for sleep staging. In particular, we must learn RF signal features that capture the sleep stages and their temporal progression, and the features should be transferable to new subjects and different environments. The problem is that RF signals carry much information that is irrelevant to sleep staging, and are highly dependent on the individuals and the measurement conditions. Specifically, they reflect off all objects in the environment including walls and furniture, and are affected by the subject's position and distance from the radio device. These challenges were not addressed in past work which used hand-crafted signal features to train a classifier [174, 147]. The accuracy was relatively low ($\sim$64%) and the model did not generalize beyond the environment where the measurements were collected.

This chapter presents RF-Sleep, a new model that delivers a significantly higher accuracy and generalizes well to new environments and subjects. The model adapts a convolutional neural network (CNN) to extract stage-specific features from RF spectrograms, and couples it with a recurrent neural network (RNN) to capture the temporal dynamics of sleep stages.

However, a CNN-RNN combination alone would remain liable to distracting features pertaining to specific individuals or measurement conditions (i.e., the source domains), and hence would not generalize well. To address this issue, we introduce a new adversarial training regime that discards extraneous information specific to individuals or measurement conditions, while retaining all information relevant to the predictive task –i.e., the adversary ensures conditional independence between the learned representation and the source domains.

Our training regime involves 3 players: the feature encoder (CNN-RNN), the sleep stage predictor, and the source discriminator. The encoder plays a cooperative game with

the predictor to predict sleep stages, and a minimax game against the source discriminator. Our source discriminator deviates from the standard domain-adversarial discriminator in that it takes as input also the predicted distribution of sleep stages in addition to the encoded features. This dependence facilitates accounting for inherent correlations between stages and individuals, which cannot be removed without degrading the performance of the predictive task.

We analyze this game and demonstrate that at equilibrium, the encoded features discard all extraneous information that is specific to the individuals or measurement conditions, while preserving all information relevant to predicting the sleep stages. We also evaluate our model on a dataset of RF measurements and corresponding sleep stages. Experimental results show that our model significantly improves the prediction accuracy of sleep stages. In particular, our model has a prediction accuracy of 79.8% and a Cohen's Kappa of 0.70, whereas the best prior result for predicting sleep stages from RF signals [147] has an accuracy of 64% and a Cohen's Kappa of 0.49.

Contributions: The chapter makes the following contributions:

- We present RF-Sleep, which delivers a significant improvement in prediction accuracy and is able to transfer the learning to new subjects and different environments.

- It adapts and extends a CNN architecture to extract stage-specific features from RF reflections, and couples it with RNNs to capture the dynamic evolution of sleep stages.

- It introduces and analyzes a new adversarial setup for removing subject and environment specific information from the latent features while preserving the ability to predict sleep stages. The adversarial model is conditioned on the predicted distribution of stage labels rather than the labels themselves allowing it to deliver a fine-grained distribution matching.

Table 3.1: Automated Sleep Staging Systems

Signal Source		Accuracy (*acc*/κ)[1]	Comfort
EEG		High $(83\%/0.76)$[2]	Low
Cardiorespiratory		Medium $(71\%/0.56)$	Medium
Actigraphy		Low $(65\%/\text{-})$[3]	High
RF	State-of-the-art	Low $(64\%/0.49)$	High
	RF-Sleep	**High (**$79.8\%/0.70)$	**High**

[1] Four-class subject-independent classification accuracy on every 30-second segment.
[2] Some studies achieve accuracy over 90% [31] but they discard artifacts and use segments from the same night to train and test.
[3] Three-class classification based on 5-minute segment.

3.2 Related Work

(a) Sleep Staging: The gold standard in sleep staging is based on Polysomnography (PSG) conducted overnight in a hospital or sleep lab. The subject has to sleep while wearing multiple sensors including an EEG monitor, an EMG monitor, an EOG monitor, nasal probes, etc. A sleep technologist visually inspects the output of the sensors and assigns to each 30-second window a stage label [121].

A few past proposals have tried to automate the process and reduce the number of sensors. These solutions can be classified into four categories according to their source signal: EEG-based, Cardiorespiratory-based, Actigraphy-based, or RF-based. Table 3.1 summarizes the state of the art performance in each category. The table shows both the classification accuracy and the Cohen's Kappa coefficient, κ. The most accurate methods rely on EEG signals [40, 45, 115, 128]. However, EEG monitors are also the most intrusive because they require the subject to sleep with a skullcap or a head-band equipped with multiple electrodes, which is uncomfortable and can cause headaches and skin irritation.The second category requires the subject to wear a chest-band and analyzes the resulting cardiorespiratory signals. It is more comfortable than the prior method but also less accurate [89]. The third approach is based on actigraphy; it leverages accelerometers in FitBit or smart phones [56, 54] to monitor body movements and infer sleep quality. Yet, motion is known to be a poor metric for measuring sleep stages and quality [114]. The

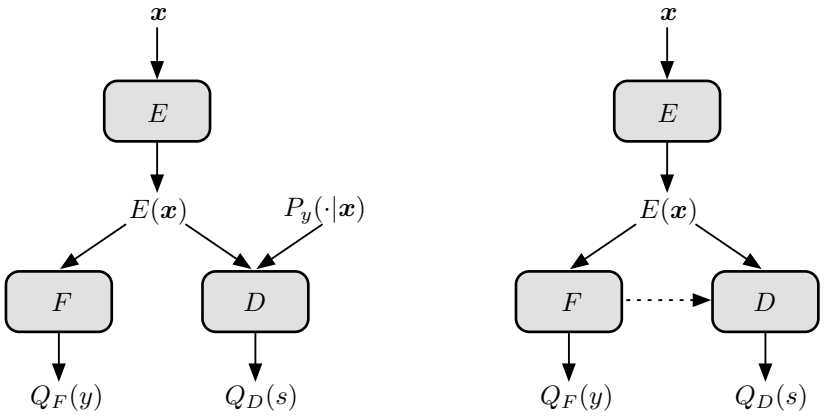
last approach relies on RF signals reflected off the subject body during her sleep. It allows the subject to sleep comfortably without any on-body sensors. Yet past approaches in this category have the worst performance in comparison to other solutions.

RF-Sleep builds on the above literature but delivers significant new contributions. In comparison to methods that use sources other than RF signals, RF-Sleep enables accurate monitoring of sleep stages while allowing the subject to sleep comfortably in her own bed without sensors on her body. Furthermore, due to differences between RF signals and other signal sources, our model has to eliminate extraneous information that are specific to the environment and irrelevant to sleep stage classification. In comparison to past work on learning sleep stages from RF signals [120, 147, 87], our approach significantly improves the prediction accuracy as shown in Table 3.1. This improvement is due to intrinsic differences between past models and the model in this chapter, which avoids hand-crafted features, and learns features that capture the temporal dependencies and transfer well to new subjects and different environments.

(b) Representation Learning: We build on a rich body of literature on CNNs and RNNs which have been successfully used to model spatial patterns [145, 57] and temporal dynamics [144], including combinations of the two [113]. Our CNN differs slightly in terms of convolutions that are adapted to our domain while, architecturally, our RNN is a standard variety LSTM.

Our work also contributes to learning invariant representations in deep adversarial networks. Adversarial networks were introduced to effectively train complex generative models of images [50, 119, 22] where the adversary (discriminator) was introduced so as to match generated samples with observed ones. The broader approach has since been adopted as the training paradigm across a number of other tasks as well, from learning representations for semi-supervised learning [94], and modeling dynamic evolution [154, 118] to inverse maps for inference [37, 38], and many others. Substantial work has also gone into improving the stability of adversarial training [96, 10, 9].

On a technical level, our work is most related to adversarial architectures for domain adaptation [46, 47, 150, 151]. Yet, there are key differences between our approach and the above references, beyond the main application of sleep staging that we introduce. First, our



(a) Model (Ideal Game) (b) Extended Game

Figure 3-1: Model and Extended Game. Dotted arrow indicates that the information does not propagate back on this link.

goal is to remove conditional dependencies rather than making the representation domain independent. Thus, unlike the above references which do not involve conditioning in the adversary, our adversary takes the representation but is also conditioned on the predicted label distribution. Second, our game theoretic setup controls the information flow differently, ensuring that only the representation encoder is modified based on the adversary performance. Specifically, the predicted distribution over stages is strategically decoupled from the adversary (conditioning is uni-directional). Third, we show that this new conditioning guarantees an equilibrium solution that fully preserves the ability to predict sleep staging while removing, conditionally, extraneous information specific to the individuals or measurement conditions. Guarantees of this kind are particularly important for healthcare data where the measurements are noisy with a variety of dependencies that need to be controlled.

Finally, our work is naturally also related to other non-adversarial literature on multi-source domain adaptation [30, 88], and work on metrics for measuring distance between distributions [13, 41].

3.3 Model

Let $x \in \Omega_x$ be an input sample, and $y \in \{1, 2, ..., n_y\}$ an output label. Let $s \in \{1, 2, ..., n_s\}$ denote an auxiliary label that refers to the source of a specific input sample. We define

$\boldsymbol{x} = [x_1, x_2..., x_t] \in \Omega_{\boldsymbol{x}}$ as the sequence of input samples from the beginning of time until the current time t.

In the context of our application, the above notation translates into the following: The input sample x is a 30-second RF spectrogram, and the output label y is a sleep stage that takes one of four values: Awake, Light Sleep, Deep Sleep, or REM. The vector $\boldsymbol{x}$ refers to the sequence of RF spectrograms from the beginning of the night until the current time. Since RF signals carry information about the subject and the measurement environment, we assign each input x an auxiliary label s which identifies the subject-environment pair, hereafter referred to as the source.

Our goal is to learn a latent representation (i.e., an encoder) that can be used to predict label y; yet, we want this representation to generalize well to predict sleep stages for new subjects without having labeled data from them. Simply making the representation invariant to the source domains could hamper the accuracy of the predictive task. Instead we would like to *remove conditional dependencies between the representation and the source domains*.

We introduce a multi-domain adversarial model that achieves the above goal. Our model is shown in Fig. 3-1a. It has three components: An encoder E, a label predictor F, and a source discriminator D. Our model is set up as a game, where the representation encoder plays a cooperative game with the label predictor to allow it to predict the correct labels using the encoded representation. The encoder also plays a minimax game against the source discriminator to prevent it from decoding the source label from the encoded representation.

A key characteristic of our model is the conditioning of the source discriminator on the label distribution, $P_y(\cdot|\boldsymbol{x})$ (see Fig. 3-1a). This conditioning of the adversary allows the learned representation to correlate with the domains, but only via the label distribution –i.e., removes conditional dependencies between the representation and the sources.

The rest of this section is organized as follows. We first formally define three players E, F, and D and the representation invariance they are trained to achieve. In Sec. 3.3.1, we analyze the game and prove that at equilibrium the encoder discards all extraneous information about the source that is not beneficial for label prediction (i.e., predicting y).

Training the ideal model in Fig. 3-1a is challenging because it requires access to the label distribution $P_y(\cdot|\boldsymbol{x})$. To drive an efficient training algorithm, we define in Sec. 3.3.2 an extended game where the source discriminator uses the output of the label predictor as an approximation of the posterior probabilities, as shown in Fig. 3-1b. We prove that the equilibriums of the original game are also equilibriums in the extended one.

Encoder E: An encoder $E(\cdot) : \Omega_{\boldsymbol{x}} \to \Omega_{\boldsymbol{z}}$ is a function that takes a sequence of input samples $\boldsymbol{x}$, and returns a vector summary of $\boldsymbol{x}$ as $\boldsymbol{z} = E(\boldsymbol{x})$.

Label Predictor F: A label predictor $F(\cdot) : \Omega_{\boldsymbol{z}} \to [0, 1]^{n_y}$ takes a latent representation $E(\boldsymbol{x})$ as input and predicts the probability of each label y associated with input $\boldsymbol{x}$ as $Q_F(y|E(\boldsymbol{x}))$. The goal of an ideal predictor F is to approximate $P_y(\cdot|\boldsymbol{x})$ with $Q_F(\cdot|E(\boldsymbol{x}))$.

The loss of the label predictor, F, given the encoder E, is defined as the cross-entropy between the label distribution $P_y(\cdot|\boldsymbol{x})$ and $Q_F(\cdot|E(\boldsymbol{x}))$:

$$\mathcal{L}_f(F; E) = \mathbb{E}_{\boldsymbol{x},y}[-\log Q_F(y|E(\boldsymbol{x}))] \tag{3.1}$$

During training, the encoder E and predictor F play a co-operative game to minimize the label prediction loss.

Source Discriminator D: We define a source discriminator as $D(\cdot, \cdot) : \Omega_{\boldsymbol{z}} \times [0, 1]^{n_y} \to [0, 1]^{n_s}$. It takes the latent representation $E(\boldsymbol{x})$ and the label distribution $P_y(\cdot|\boldsymbol{x})$ as inputs, and predicts which source domain (i.e., subject and environment) they are sampled from as $Q_D(\cdot|E(\boldsymbol{x}), P_y(\cdot|\boldsymbol{x}))$.

Next, we define the desired representation invariance.

Definition 1 (Representation invariance)**.** *We say that representation E is invariant if $E(\boldsymbol{x})$ contains no information about s beyond what is already contained in $P_y(\cdot|\boldsymbol{x})$; that is, $Q_D(\cdot|E(\boldsymbol{x}), P_y(\cdot|\boldsymbol{x})) = Q_D(\cdot|P_y(\cdot|\boldsymbol{x}))$ for the optimal D.*

To measure the invariance of an encoder E, we define the loss of the source discriminator D as the cross-entropy between $P_s(\cdot|\boldsymbol{x})$ and $Q_D(\cdot|E(\boldsymbol{x}), P_y(\cdot|\boldsymbol{x}))$:

$$\mathcal{L}_d(D; E) = \mathbb{E}_{\boldsymbol{x},s}[-\log Q_D(s|E(\boldsymbol{x}), P_y(\cdot|\boldsymbol{x}))] \tag{3.2}$$

During training, encoder E and discriminator D play a minimax game: while D is trained to minimize the source prediction loss, encoder E is trained to maximize it in order to achieve the above invariance.

3.3.1 Ideal Game

During training, encoder E plays a co-operative game with predictor F, and a minimax game with discriminator D. We define a value function of E, F and D with $\lambda > 0$:

$$\mathcal{V}(E, F, D) = \mathcal{L}_f(F; E) - \lambda \cdot \mathcal{L}_d(D; E) \tag{3.3}$$

The training procedure can be viewed as a three-player minimax game of E, F and D:

$$\min_E \min_F \max_D \mathcal{V}(E, F, D) = \min_{E,F} \max_D \mathcal{V}(E, F, D) \tag{3.4}$$

Proposition 2 (Optimal predictor)**.** *Given encoder E,*

$$\mathcal{L}_f(E) \triangleq \min_F \mathcal{L}_f(F; E) \geq H(y|E(\boldsymbol{x})), \tag{3.5}$$

where $H(\cdot)$ is entropy.

The optimal predictor F^* that achieves equality is:

$$Q_{F^*}(y|E(\boldsymbol{x})) = p(y|E(\boldsymbol{x})) \tag{3.6}$$

Proof.

$$
\begin{aligned}
&\mathcal{L}_f(F;E)\\
=&\mathbb{E}_{\boldsymbol{x},y}[-\log Q_F(y|E(\boldsymbol{x}))]\\
=&\mathbb{E}_{E(\boldsymbol{x}),y}[-\log Q_F(y|E(\boldsymbol{x}))]\\
=&\mathbb{E}_{\boldsymbol{z}\sim P(E(\boldsymbol{x}))}\,\mathbb{E}_{y\sim P(y|\boldsymbol{z})}[-\log Q_F(y|\boldsymbol{z})]\\
=&\mathbb{E}_{\boldsymbol{z}\sim P(E(\boldsymbol{x}))}[H(y|\boldsymbol{z})+D_{\mathrm{KL}}(P(y|\boldsymbol{z})\,\|\,Q_F(y|\boldsymbol{z}))]\\
\geq&\mathbb{E}_{\boldsymbol{z}\sim P(E(\boldsymbol{x}))}[H(y|\boldsymbol{z})]\\
=&H(y|E(\boldsymbol{x}))
\end{aligned}
$$

The equality holds when $D_{\mathrm{KL}}(P(y|E(\boldsymbol{x}))\,\|\,Q_F(y|E(\boldsymbol{x})))=0$ for almost every $\boldsymbol{x}\in Supp(P_{\boldsymbol{x}})$. That is $Q_{F^*}(y|E(\boldsymbol{x}))=p(y|E(\boldsymbol{x}))$ for almost every y and $\boldsymbol{x}\in Supp(P_{\boldsymbol{x}})$. □

Similarly we can prove the following Proposition.

Proposition 3 (Optimal discriminator)**.** *Given encoder E,*

$$\mathcal{L}_d(E)\triangleq\min_D\mathcal{L}_d(D;E)\geq H(s|E(\boldsymbol{x}),P_y(\cdot|\boldsymbol{x})) \tag{3.7}$$

The optimal discriminator D^* that achieves this value is:

$$Q_{D^*}(s|E(\boldsymbol{x}),P_y(\cdot|\boldsymbol{x}))=P(s|E(\boldsymbol{x}),P_y(\cdot|\boldsymbol{x})) \tag{3.8}$$

Corollary 3.1. *$H(s)$ is an upper bound of the loss of the optimal discriminator D^* for any encoder E.*

Next, we state the virtual training criterion of the encoder.

Proposition 4. *If predictor F and discriminator D have enough capacity and are trained to achieve their optimal losses, the minimax game (3.4) can be rewritten as the following training procedure of encoder E:*

$$\min_E[H(y|E(\boldsymbol{x}))-\lambda\cdot H(s|E(\boldsymbol{x}),P_y(\cdot|\boldsymbol{x}))] \tag{3.9}$$

Proof. Based on the losses of the optimal predictor F^* and the optimal discriminator D^* in Proposition 2 and Proposition 3, the minimax game (3.4) can be rewritten as (3.9). Thus, encoder E is trained to minimize a virtual training criterion $C(E) = H(y|E(\boldsymbol{x})) - \lambda \cdot H(s|E(\boldsymbol{x}), P_y(\cdot|\boldsymbol{x}))$. □

Next, we describe the optimal encoder.

Theorem 5 (Optimal encoder)**.** *If encoder E, predictor F and discriminator D have enough capacity and are trained to reach optimum, any global optimal encoder E^* has the following properties:*

$$H(y|E^*(\boldsymbol{x})) = H(y|\boldsymbol{x}) \tag{3.10a}$$

$$H(s|E^*(\boldsymbol{x}), P_y(\cdot|\boldsymbol{x})) = H(s|P_y(\cdot|\boldsymbol{x})) \tag{3.10b}$$

Proof. Since $E(\boldsymbol{x})$ is a function of $\boldsymbol{x}$:

$$\mathcal{L}_f(E) = H(y|E(\boldsymbol{x})) \geq H(y|\boldsymbol{x}) \tag{3.11a}$$

$$\mathcal{L}_d(E) = H(s|E(\boldsymbol{x}), P_y(\cdot|\boldsymbol{x})) \leq H(s|P_y(\cdot|\boldsymbol{x})) \tag{3.11b}$$

Hence, $C(E) = H(y|E(\boldsymbol{x})) - \lambda \cdot H(s|E(\boldsymbol{x}), P_y(\cdot|\boldsymbol{x})) \geq H(y|\boldsymbol{x}) - \lambda \cdot H(s|P_y(\cdot|\boldsymbol{x}))$. The equality holds if and only if both (3.10a) and (3.10b) are satisfied. Therefore, we only need to prove that the optimal value of $C(E)$ is equal to $H(y|\boldsymbol{x}) - \lambda \cdot H(s|P_y(\cdot|\boldsymbol{x}))$ in order to prove that any global encoder E^* satisfies both (3.10a) and (3.10b).

We show that $C(E)$ can achieve $H(y|\boldsymbol{x}) - \lambda \cdot H(s|P_y(\cdot|\boldsymbol{x}))$ by considering the following encoder E_0: $E_0(\boldsymbol{x}) = P_y(\cdot|\boldsymbol{x})$. It can be examined that $H(y|E_0(\boldsymbol{x})) = H(y|\boldsymbol{x})$ and $H(s|E_0(\boldsymbol{x}), P_y(\cdot|\boldsymbol{x})) = H(s|P_y(\cdot|\boldsymbol{x}))$. □

Adversarial training of D can be viewed as a regularizer, which leads to a common representation space for multiple source domains. From Theorem 5, the optimal encoder E^* using adversarial training satisfies $H(y|E^*(\boldsymbol{x})) = H(y|\boldsymbol{x})$, which is the maximal discriminative capability that any encoder E can achieve. Thus, we have the following corollary.

Corollary 5.1. *Adversarial training of the discriminator does not reduce the discriminative capability of the representation.*

Remark 5.1. *During the proof of Theorem 5, we construct an encoder $E_0(\boldsymbol{x}) = P_y(\cdot|\boldsymbol{x})$ that can achieve the optimal value of $C(E)$. However, we argue that training will not converge to this trivial encoder in practice. This is because $P_y(\cdot|\boldsymbol{x})$ is a mapping from the full signal history to the distribution over stages at the current step, therefore itself highly complex. Since we use the RNN state as the encoding $E(\boldsymbol{x})$, and it feeds into the LSTM gates, distribution over stages at previous step does not represent a sufficient summary of the history until the current one. Therefore, $E(\boldsymbol{x})$ must be able to anticipate the temporal evolution of the signal and contain a more effective summary than $P_y(\cdot|\boldsymbol{x})$ would be.*

Corollary 5.2. *If encoder E and predictor F have enough capacity and are trained to reach optimum, the output of F is equal to $P_y(\cdot|\boldsymbol{x})$.*

Proof. When predictor F is optimal (Proposition 2), $Q_F(y|E(\boldsymbol{x})) = p(y|E(\boldsymbol{x}))$. When E is optimal (Theorem 5), $H(y|E(\boldsymbol{x})) = H(y|\boldsymbol{x})$, that is $p(y|E(\boldsymbol{x})) = p(y|\boldsymbol{x})$. Therefore, $Q_F(y|E(\boldsymbol{x})) = p(y|\boldsymbol{x})$. □

3.3.2 Extended Game

In practice, estimating the posterior label distribution $P_y(\cdot|\boldsymbol{x})$ from labeled data is a non-trivial task. Fortunately however our predictor F and encoder E are playing a cooperative game to approximate this posterior label distribution $P_y(\cdot|\boldsymbol{x})$ with $Q_F(\cdot|E(\boldsymbol{x}))$. Therefore, we use $Q_F(\cdot|E(\boldsymbol{x}))$, the output of predictor F, as a proxy of $P_y(\cdot|\boldsymbol{x})$ and feed it as input to discriminator D (Fig. 3-1b).

An extended three-player game arises: while encoder E still plays a cooperative game with predictor F and a minimax game with discriminator D, discriminator D depends strategically on predictor F but not vice versa. The dotted line in Fig. 3-1b illustrates this dependency.

The relationship between the ideal minimax game (Sec. 3.3.1) and the extended one is stated below.

Algorithm 1 Encoder, predictor and discriminator training

Input: Labeled data $\{(\boldsymbol{x}_i, y_i, s_i)\}_{i=1}^{M}$, learning rate η.
Compute stop criterion for inner loop: $\delta_d \leftarrow H(s)$
for number of training iterations **do**
  Sample a mini-batch of training data $\{(\boldsymbol{x}_i, y_i, s_i)\}_{i=1}^{m}$

$$\mathcal{L}_f^i \leftarrow -\log Q_F(y_i|E(\boldsymbol{x}_i))$$
$$\boldsymbol{w}_i \leftarrow Q_F(\cdot|E(\boldsymbol{x}_i)) \blacktriangleright \textit{stop gradient along this link}$$
$$\mathcal{L}_d^i \leftarrow -\log Q_D(s_i|E(\boldsymbol{x}_i), \boldsymbol{w}_i)$$
$$\mathcal{V}^i = \mathcal{L}_f^i - \lambda \cdot \mathcal{L}_d^i$$

  Update encoder E:

$$\theta_e \leftarrow \theta_e - \eta_e \nabla_{\theta_e} \frac{1}{m} \sum_{i=1}^{m} \mathcal{V}^i$$

  Update predictor F:

$$\theta_f \leftarrow \theta_f - \eta_f \nabla_{\theta_f} \frac{1}{m} \sum_{i=1}^{m} \mathcal{V}^i$$

  repeat
    Update discriminator D:

$$\theta_d \leftarrow \theta_d + \eta_d \nabla_{\theta_d} \frac{1}{m} \sum_{i=1}^{m} \mathcal{V}^i$$

  until $\frac{1}{m} \sum_{i=1}^{m} \mathcal{L}_d^i \leq \delta_d$
end for

Proposition 6. *If encoder E, predictor F and discriminator D have enough capacity, the solution that encompasses the optimal encoder, E^*, predictor, F^* and discriminator, D^*, in the ideal minimax game is also an equilibrium solution of the extended game.*

Proof. By Corollary 5.2, when encoder E and predictor F are optimal, $Q_F(\cdot|E(\boldsymbol{x}))$ is equal to $P_y(\cdot|\boldsymbol{x})$. Thus, the extended game becomes equivalent to the ideal game, and E^*, F^* and D^* is an equilibrium solution of both games. □

3.3.3 Training Algorithm

We implement the extended three-player game with iterative updates of the players (Algorithm 1). Note that, since the output of the label predictor is a proxy of the underlying posterior, and since the source discriminator depends strategically on the predictor but not vice versa, the gradient does not back-propagate from the discriminator to the predictor (i.e., the dotted link in Fig. 3-1b).

The number of training steps in the inner loop usually needs to be carefully chosen [50]. A large number of steps is computationally inefficient but a small one will cause the model to collapse. This is because the outer players, E and F, can be over-trained against a non-

optimal inner player D, and they will try to maximize $\mathcal{L}_d$ at the cost of increasing $\mathcal{L}_f$. To prevent the model collapse phenomenon, we use an adaptive number of training steps in the inner loop and adjust it dynamically based on $\mathcal{L}_d$ (Algorithm 1). The idea is to use the upper bound in Corollary 3.1 as the stopping criterion for the inner loop.

3.3.4 Discussion of the Model Benefits

While we described our model in the context of sleep staging, we believe the model can be applied more broadly. Our model is characterized by the 3-way game and the adversarial conditioning on the label distribution. This combination yields the following benefits: 1) It guarantees an equilibrium solution that fully preserves the ability to perform the predictive task while removing any distracting information specific to the source domains. Guarantees of this kind are particularly important in healthcare where the measurements are noisy and have a variety of dependencies that need to be controlled. 2) It allows to properly leverage the adversarial feedback even when the target labels are uncertain. For example, in the sleep staging problem, each 30-second window is given one label. Yet, many such windows include transitions between sleep stages, e.g., a transition from light to deep sleep. These transitions are gradual and hence the transition windows can be intrinsically different from both light and deep sleep. It would be desirable to have the learned representation capture the concept of transition and make it invariant to the source (see the results in Sec. 3.4.5). 3) It allows the conditioning to remain available for additional guiding of representations based on unlabeled data. The model can incorporate unlabeled data for either semi-supervised learning or transductive learning within a unified framework.

3.4 Experiments

In this section, we empirically evaluate our model.

3.4.1 RF-Sleep Dataset

RF-Sleep is a dataset of RF measurements during sleep with corresponding sleep stage labels. All studies that involve human subjects were approved by our IRB.

Study setup: The sleep studies are done in the bedroom of each subject. We install a radio device in the bedroom. It transmits RF signals and measure their reflections while the subject is sleeping alone in the bed.

Ground truth: During the study, each subject sleeps with an FDA-approved EEG-based sleep monitor [115], which collects 3-channel frontal EEG. The monitor labels every 30-second of sleep with the subject's sleep stage. This system has human-level comparable accuracy [115], and has already been used in several sleep studies[90, 127].

Size of dataset: The dataset collects 100 nights of sleep from 25 young healthy subjects (40% females). It contains over 90k 30-second epochs of RF measurements and their corresponding sleep stages provided by the EEG-based sleep monitor. Each epochs has one of four labels Awake, REM, Light Sleep (N1 or N2) and Deep Sleep (N3).

3.4.2 Parameterization

We parameterize encoder E, predictor F and discriminator D as neural networks. Encoder E is parameterized by a hybrid CNN-RNN model. We adapt a residual networks architecture [57] with 24 convolutional layers to extract features from each 30-second RF spectrogram, and an RNN with LSTM cell [59] that takes sequences of CNN features as input. Both predictor F and discriminator D are parameterized by networks with two fully-connected layers.

3.4.3 Classification Results

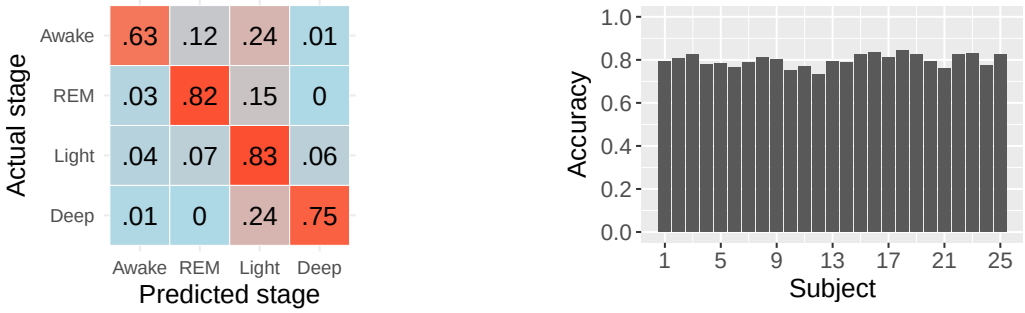
We evaluate the model on every subject while training on the data collected from the other subjects (i.e., the model is never trained on data from the test subject). The training data is randomly split into a training set and validation set ($75\%/25\%$).

We use two metrics commonly used in automated sleep staging, namely Accuracy and Cohen's Kappa. While accuracy measures the percent agreement with ground truth, Co-

hen's Kappa coefficient κ [27] takes into account the possibility of the agreement occurring by chance and is usually a more robust metric. $\kappa > 0.4$, $\kappa > 0.6$, $\kappa > 0.8$ are considered to be moderate, substantial and almost perfect agreement [77].

Table 3.2: Sleep Stage Classification Accuracy and Kappa

Approach	*Accuracy*	κ
Tataraidze et al. [147]	0.635	0.49
Zaffaron et al. [174]	0.641	0.45
Ours	**0.798**	**0.70**



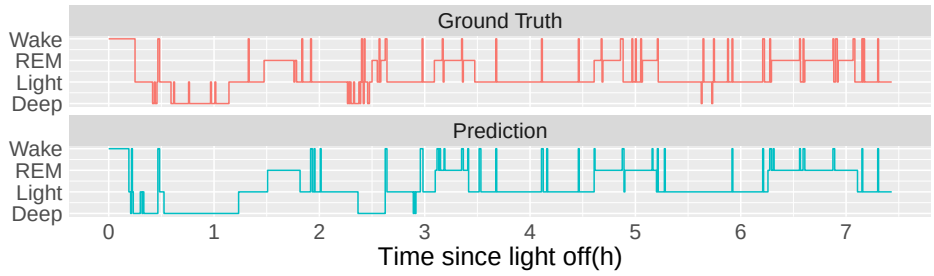
(a) Confusion Matrix

(b) Accuracy on each subject

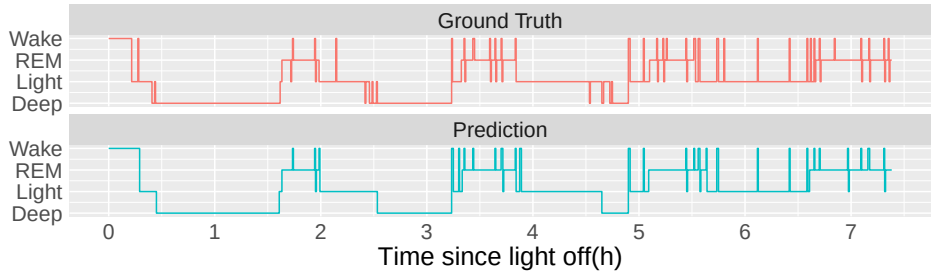
Figure 3-2: Fig. 3-2a shows that our model can distinguish deep and light sleep with high accuracy. And Fig. 3-2b illustrates that our model works well for different subjects and environments.

Table 3.2 shows the accuracy and Cohen's Kappa of our model compared to the state-of-the-art in classifying sleep stages using RF reflections. Since neither the dataset nor the code used in past papers is publicly available, we compare with their published results. We note however that the Cohen's Kappa provides some normalization since it accounts for the underlying uncertainty in the data. The table shows that our model has an accuracy of 79.8% and a $\kappa = 0.70$, which significantly outperforms past solutions.

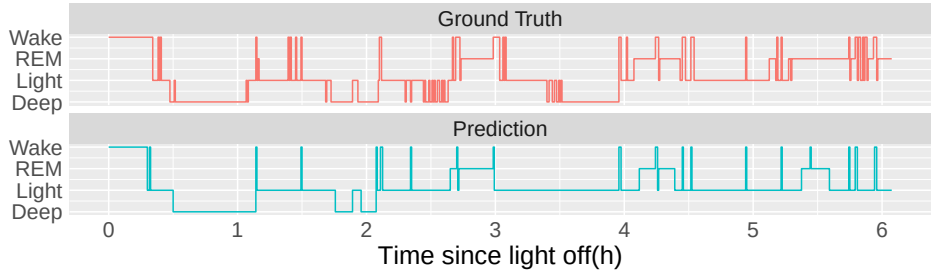
Fig. 3-2a shows the confusion matrix of our model. Fig. 3-2b also shows the accuracy on each subject. It has a standard deviation of 2.9%, suggesting that our model is capable of adapting to different subjects and environments.



(a) Average Accuracy (80.4%)



(b) Best Accuracy (91.2%)



(c) Worst Accuracy (71.2%)

Figure 3-3: Three examples of full night predictions corresponding to the average, best and worst classification accuracy.

Finally, we show in Fig. 3-3 the full-night predictions along with the ground truth for the average, best, and worst classification accuracy.

3.4.4 Understanding the Role of CNN & RNN

We analyze the role of CNN and RNN in predicting sleep stages. To do so, we use t-SNE embedding [92] to visualize the response of our network after CNN and RNN, respectively. Fig. 3-4 shows the visualization results from one of the subjects. Data points are randomly sub-sampled for better viewing. The result shows that the CNN succeeds at separating the Wake, REM from Light and Deep Sleep. However it fails at separate Light Sleep and Deep

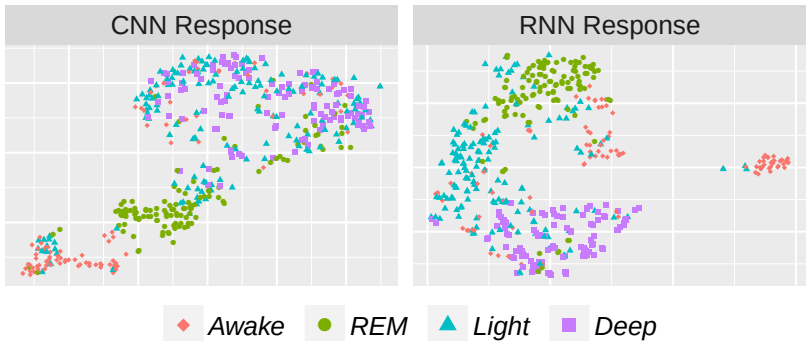


Figure 3-4: Visualizations of the CNN and RNN responses. CNN can separate Wake REM and from the other stages, yet Deep and Light Sleep can only be distinguished by RNN.

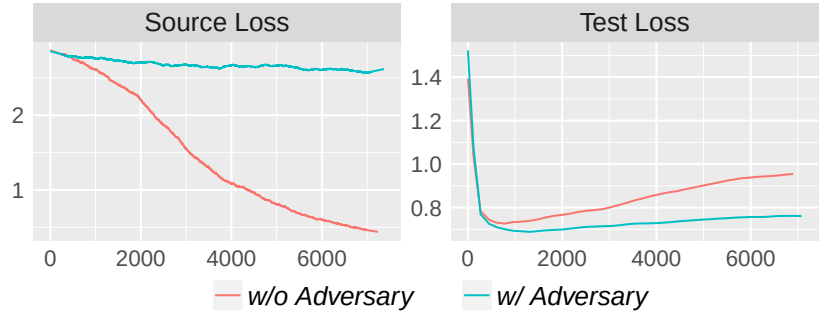


Figure 3-5: Baseline model and ours are evaluated on same dataset. A higher source loss indicates the removal of source specific information, and a lower test loss shows that the proposed setup can better avoid overfitting.

Sleep from each other. In contrast, Light Sleep and Deep Sleep form different clusters in the RNN response. These results demonstrate the role of CNN and RNN in our model: CNN learns stage-specific features that can distinguish Wake, REM and from Deep and Light Sleep. RNN captures the dynamics of those features to further determine whether the sleep is light or deep. Note that Light and Deep Sleep are more similar to each other and are typically referred to as NREM, i.e., non-REM.

We have trained a similar model without the RNN layer on top of CNN. In this case, the overall accuracy decreases by 12.8%, specifically the precision light and deep sleep decreases by 23.5%. This suggests that there are stage-specific information embedded in the temporal dynamics of the RF measurements, and therefore can only be captured

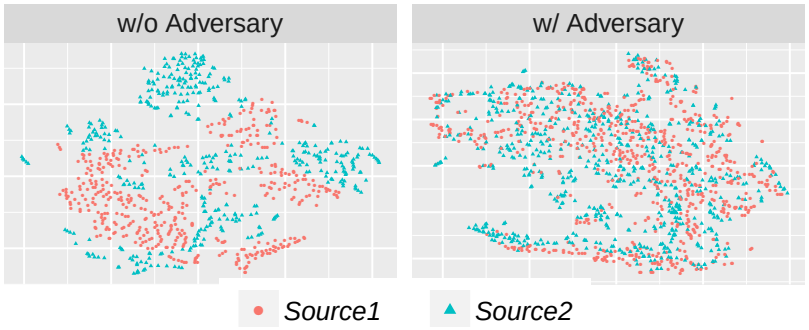


Figure 3-6: Visualization of learned latent representations from two sources. Data-points are separated when no adversary, yet they are well aligned by proposed setup.

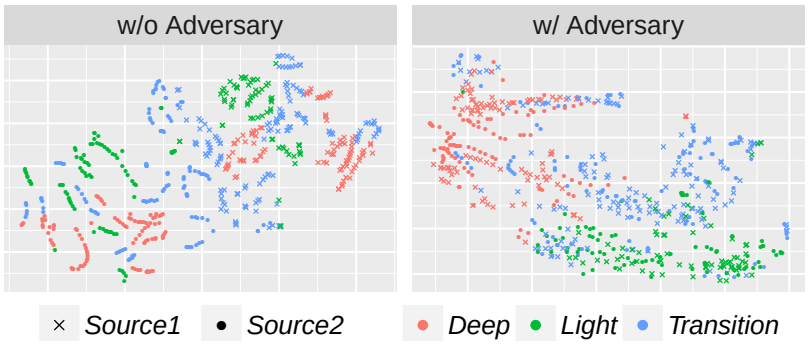


Figure 3-7: Visualization of fine-grained alignment on test data. Our model, which conditions the adversary on the posterior distribution, not only aligns deep and light stages, but also aligns the transition periods, which are not directly specified by the labels.

and exploited with RNN. Moreover, these temporal dynamics are particularly crucial for distinguishing light and deep sleep. Indeed, there are known temporal patterns that govern the progression of light and deep sleep through the night [16]. For example, the probability of being in deep sleep decreases as sleep progresses. Also, people usually need to go through light sleep before they can get into deep sleep. Moreover, people tend to have less body movements during deep sleep. These temporal dynamics of sleep stages can be captured by RNN and might be exploited to distinguish light and deep sleep.

To illustrate those patterns, we segment each night into several stable NREM periods with requirement that all periods should be longer than $30min$ and do not contain more than $2min$ of continuous REM/Awake epochs, and then we calculate the occurrence likelihood

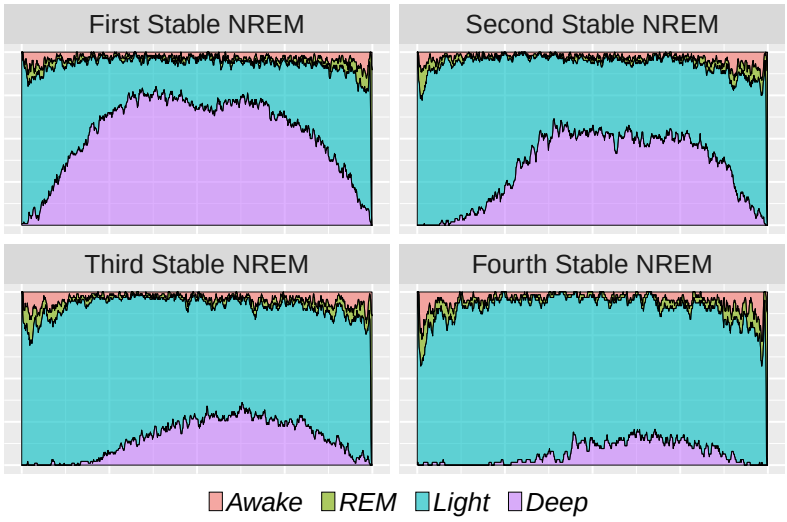


Figure 3-8: The x-axis is time normalized over one night of sleep from 0% to 100%, and the height of each region shows its likelihood of its occurrence. The figure shows that deep sleep is more likely to happen in the first half of the night, and is surrounded by light sleep.

of each stage based on our dataset. As shown in Figure 3-8, likelihood of deep sleep reaches its maximum in the middle of each stable NREM period, and is decreasing with respect to time. Also the figure shows that there are more movements during the light sleep period. On average, people move about $90sec/h$ during the light sleep stage, but only $24sec/h$ during the deep sleep stage. So if the person moves frequently, he or she is less likely to be in deep sleep.

It is worth noticing that two types of awake epochs: before sleep and during the sleep. In CNN they are mixed together, in contrast, the first type of wake is separated far away from others by RNN, because it is a common pattern that it takes a while for people to go sleep after they turn off the light.

3.4.5 Role of Our Adversarial Discriminator

We evaluate the role of our adversarial discriminator in learning transferable features for predicting sleep stages. We first look at the losses on the validation set as training progresses to check whether the extraneous information specific to the individuals and environments can be removed. As a baseline, we compare with a version of our model without the source discriminator. For this baseline, we train a (non-adversarial) discriminator to

determine the source of features. Fig. 3-5 shows that the loss of the source discriminator in the baseline model decreases very quickly while ours stays high (upper bounded by $H(s) = 2.81$ in this case), suggesting that our learned representation is invariant across sources. The figure also shows that adding an adversarial discriminator increases the performance on the test set and can be helpful in reducing over-fitting.

To check that our adversarial model has learned transferable features, we visualize the learned features $E(\boldsymbol{x})$ on the test data for both models. Color-coding the sources, Fig. 3-6 shows that our learned features have almost the same distribution on different sources, while the baseline model learns features that are separable.

Next, we illustrate the benefits of conditioning on the posterior distribution, and that it can recover underlying concepts not specified in the labels. We consider the learned features for transition periods between light and deep sleep, which might be a class that is different from both light and deep sleep. We define transition periods as epochs that have both light and deep sleep as neighbors. We visualize it with a different color. Color-coding stages and shape-coding sources, Fig. 3-7 shows the learned features from transition periods are segregated, as those from light sleep and deep sleep. This indicates that our learned features have recovered the concept of a transition period, which is helpful in understanding and predicting sleep stages.

3.5 Conclusion

This chapter introduces RF-Sleep, a new predictive model that learns sleep stages from RF signals and achieves a significant improvement over the state-of-the-art. We believe this work marks an important step in sleep monitoring. We also believe that the proposed adversarial setup, which extracts task-specific domain-invariant features, is applicable to other predictive tasks, particularly in health sensing where variations across subjects and measurement conditions could be a major challenge.

Chapter 4

Extracting Multi-Person Respiration

4.1 Introduction

Breathing is an important health metric used for tracking diseases in many areas, such as sleep [106, 140], pulmonology [117], and cardiology [19, 171]. It can also provide useful insights about the psychological state of an individual [79, 76]. Respiration monitoring during sleep is particularly useful. For example, as explained in Chapter 3, the respiration signal can be used to infer the person's sleep stages (light, deep or REM sleep), and diagnose sleep disorders [14, 43]. Monitoring can also be used to track sleep disturbed breathing which has been associated with cardiac arrest in heart failure patients [170, 101]. Furthermore, respiration monitoring is important for diagnosing and tracking sleep apnea, a condition in which the person stops breathing for short intervals during the night [16].

Traditional approaches for respiration monitoring are relatively cumbersome. They require the person to wear a breathing belt and/or a nasal probe and sleep with it. However, recent advances in RF-based sensing have demonstrated the possibility of monitoring breathing without any sensor on the person's body [120, 5]. Such systems transmit a low power wireless signal and capture its reflections. The signal bounces off the people in the environment and gets modulated by their breathing. The receiver uses the reflected signal to track the person's breathing without any sensors on the body.

However, past work on RF-based respiration monitoring requires monitored people to be located away from each other. This requirement precludes respiration monitoring during

sleep for people who share the same bed. Hence, it prevents a large fraction of the population benefiting from such technology including couples that share the same bed, and new mothers who sleep with their infants.

Tracking respiration when people are next to each other is a difficult problem. RF reflections off two bodies super-impose (i.e. add up) over the wireless medium and interfere at the receiver. For relatively distant people, one can use antenna arrays and/or Frequency-Modulated Continuous Wave (FMCW) radios [93] to zoom in on the location of a particular individual and receive the signal reflected off his/her body with limited to no interference. However, as people get closer, interference becomes too high. Hence, even systems that combine antenna arrays and FMCW still require a minimum separation of 1.5 to 2 meters [5]. Fundamentally, today there is no solution that can disentangle mixed RF-based breathing signals. Hence, all systems require a minimum distance between monitored users.

This chapter introduces DeepBreath, the first system that disentangles mixed RF-based breathing signals. DeepBreath can monitor the respiration signals of multiple people even if they have zero distance between them. It also provides continuous monitoring throughout the night for people who share the same bed.

The design of DeepBreath combines three components as described below.

(a) Breathing Separation: DeepBreath has a breathing separation module that reconstructs the correct breathing signals of multiple co-located individuals. To design this component, we first model how RF-based breathing signals mix over the wireless medium. We show that the mixing is complicated by changes in the wireless channels due to movements. We present an approximation that allows us to formulate the problem as blind source separation and solve it using independent component analysis (ICA). Given this new formulation, DeepBreath uses an FMCW radio equipped with an antenna array to zoom in on RF signals from different locations in space. It calls those RF signals observations. DeepBreath uses its formulation of the mixing process to recover the original breathing signals from such observations using ICA.

(b) Motion Detection: When people move, their location changes and the contribution of their breathing to each observation changes. Hence the mapping from observations to

original breathing signals has to be recomputed. Note that even a small motion – such as turning in bed – changes the mapping. Thus, DeepBreath has to automatically detect such movements and run its breathing separation module on each stable period separately. To detect motion reliably DeepBreath uses a convolutional neural network that is trained to identify movements of the monitored people and ignore irrelevant motion, e.g, a fan or the HVAC (Heating, ventilation and air conditioning).

(c) Identity Matching: For continuous monitoring, a night of sleep has to be segmented into multiple periods of stable breathing with no motion. Yet, when running ICA on each such period, the recovered breathing signals can be permuted in the output. For example, say there are 2 people in the environment. During the first period, the first ICA component may refer to the first person's breathing and the second component to the second person's breathing. However, in the next period, the situation may flip. How do we identify the ICA components that correspond to the breathing of the same individual in different periods?

We formulate this problem as an optimization problem, where the solution maximizes a similarity metric between the person's breathing signals across different periods. We solve this optimization using dynamic programming. The final solution provides a robust system that disentangles the breathing signals of multiple people and delivers continuous respiration monitoring throughout the night.

We have implemented DeepBreath and evaluated it with 13 couples.[1] The couples sleep in the same bed, with the bed size ranging from twin (1m wide) to king (2m). To obtain ground truth breathing, each subject sleeps with an FDA-approved respiration monitoring belt [138]. In total, we have collected 21 nights of sleep, and 151 hours of data. To measure the accuracy of the reconstructed breathing signal, we compute its correlation with the ground truth signal from the respiration belt. Note that the correlation between two breathing belts on the same subject, one on the chest and one on the belly is around 0.915. Thus a comparable correlation value indicates a high accuracy of the recovered breathing signals.

Our experiments show that DeepBreath recovers the breathing signals of couples sharing the bed with high accuracy. Specifically, the recovered breathing signal has an average

[1] All experiments are approved by our IRB.

correlation of 0.914 with the breathing belt signal measured on the subject's chest. DeepBreath also recovers the breathing rates of the monitored subjects with an average error as small as 0.140 breaths per minute.[2]

Further, DeepBreath scales beyond two people. We conduct experiments with 5 subjects sitting shoulder-to-shoulder on one couch. DeepBreath can recover the breathing signals of all 5 subjects with an average correlation of 0.922 with the ground truth breathing signals.

To the best of our knowledge, DeepBreath is the first system that recovers the detailed respiration signals of multiple people sharing the bed or sitting with zero separation. This result renders much recent research using RF signals to monitor respiration, sleep, and apnea applicable to a significantly larger segment of the population including couples sharing the same bed and recent mothers sleeping with their babies.

4.2 Illustrative Example

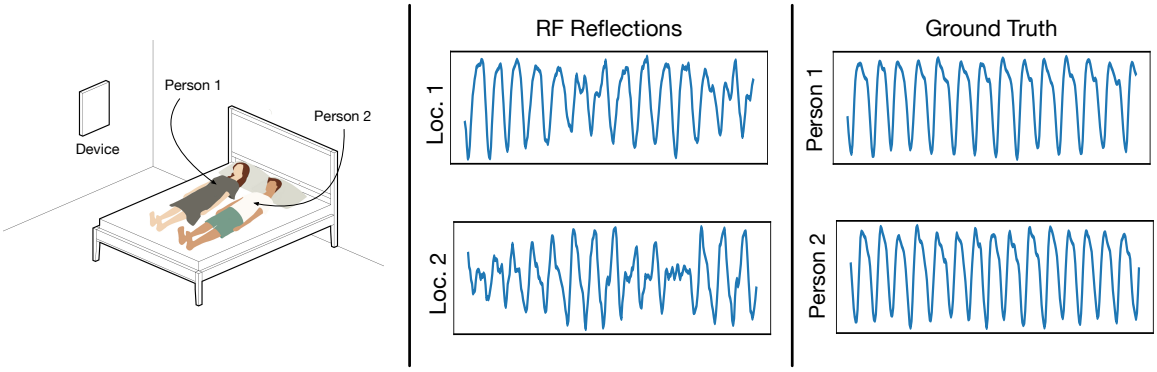
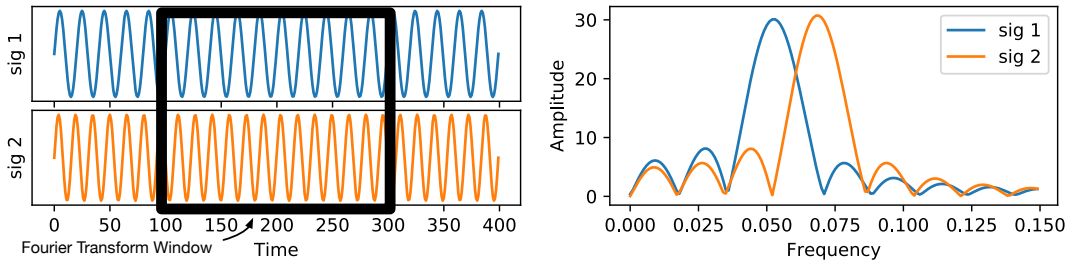


Figure 4-1: A scenario of two people sharing a bed. Loc. 1 and Loc. 2 represent the locations of Person 1 and Person 2's bodies. During this moment, both Person 1 and Person 2 are breathing normally, yet the reflected signal from Loc. 2 indicates Person 2 has an apnea event.

Consider a scenario in which a couple shares the bed. Let us examine the accuracy of monitoring their breathing using the state of the art RF techniques. To do so, we use an FMCW radio equipped with an antenna array as in [5]. Such a radio divides the space

[2]Average breathing rate of all the subjects is 15.68 breaths per minute, with a standard deviation of 2.18 breaths per minute. So our system's breathing rate percentage error is 0.89%.



(a) Two sine waves with different frequency (b) Windowed Fourier Transform result

Figure 4-2: A sinc example. Fig. 4-2a shows two sine waves with slightly different frequency. The black box on top of the signals indicates the Fourier Transform window. Fig. 4-2b shows the result of the Fourier Transform. Two signals are sincs in frequency domain, and when the frequencies of the signals are similar, their sincs are close to each other.

into voxels, and separates RF signals received from different spatial voxels. We deploy the radio in the couple's bedroom as shown in the first column in Fig. 4-1. We also ask both subjects to wear respiration belts to obtain their ground truth breathing.

We measure the exact location of both subjects with respect to the radio using the Bosch GLM50 laser distance measurement tool [29]. We then make the FMCW radio and antenna array zoom in on the two voxels in space that are centered on the chest of each subject. This gives us the best estimates of the subjects' breathing signals based on past work, under the assumption that past work is able to perfectly localize the subjects. We plot those RF signals in the middle column in Fig. 4-1. The figure shows that by looking at RF reflections from the two subjects, it seems that person 2 has experienced an apnea event – i.e., a period during which he did not breath.

Now let us look at the actual breathing signals from the two belts on the subjects, which we show in the third column in Fig. 4-1. The ground truth shows that both subjects have regular breathing and neither of them had an apnea event, which contradicts the behavior shown in the RF signals reflected off their chests. This can lead to serious errors in reporting a person's health condition.

This experiment shows that even if one zooms in on the signal from the exact location of the person's chest, one cannot eliminate the effect of the signal reflected off a nearby person. The reason we cannot completely separate the two signals is that antenna arrays and FMCW are filters over space. However, no filter is perfect. In particular, both antenna

arrays and FMCW filter signals using the Fourier transform [93, 4]. Since the Fourier transform is applied over a finite window, it will result in a *sinc* in the frequency domain. This causes nearby signals to mix with each other due to the sinc tail, as shown in Fig. 4-2. The closer the two people are, the closer the frequency response after taking the Fourier Transform, and the more the corresponding sinc functions leak into each other.[3]

4.3 Related Work

DeepBreath is related to past work on RF-based sensing and the general literature on blind source separation.

4.3.1 RF-Based Sensing

Recent years have witnessed much interest in using RF signals to monitor people's movements and vital signs without any on-body sensors. Researchers have developed systems that transmit low power RF signal and monitor its reflections. They used these reflections to infer a person's location [4, 75], breathing [5, 26, 155], heart rate [5, 26] and even emotions [177].

DeepBreath is closest to past work that uses RF signals to monitor breathing. Studies using Doppler Radio [120], FMCW [5, 62, 179], millimeter waves [168, 26], and WiFi signals [155, 87] all demonstrate accurate monitoring of a single person's breathing. Respiration monitoring using radio signal is also closely related to monitoring breathing using acoustic signals, which tends to use FMCW techniques and demonstrates good accuracy for a single person [102, 149]. Further, some recent papers demonstrate that the breathing signal extracted from RF can be used to infer additional health metrics [177, 126]. In particular, a larger number of papers use the breathing extracted from RF to monitor sleep quality, track insomnia, and infer sleep stages (Chapter 3).

A key challenge in passive wireless monitoring is the interference between different

[3]Note that leakage through the sinc tail occurs even when two objects are separated by a distance larger than the resolution of the antenna array or the FMCW radio. Said differently, the resolution of antenna array or FMCW radio identifies the minimum distance for two objects to be both detectable (but it does not mean that their reflections are completely separated.)

people. RF signals reflect off all people in the environment. Those reflections super-impose over the wireless medium and add-up in the received signal. To avoid this problem, much of the past work assumes a single person in the environment [177, 4, 179]. Papers that show results for multiple users require people to have a minimum separation of 1.5 to 2 meters [5, 62, 168]. Having a minimum separation between subjects reduces the mixing between the RF signals reflected off their bodies, and allows the receiver to zoom in on each individual using techniques like FMCW and antenna arrays. The need for such minimum separation prevents past work from monitoring users sitting next to each other or sleeping in the same bed.

The only past studies, that we know of, that report breathing results for users in the same bed are [155] and [102]. The former requires the users to lie in a specific position and have significantly different breathing rates. The latter uses audio signals as opposed to RF signals, and hence is vulnerable to environmental audio sources and is affected by thick comforters and clothes. Furthermore, both studies only report the breathing rate, which is an average value over a long interval. Recovering the exact details of the breathing signal is a much more demanding task.

More fundamentally, all past work relies on the existence of good RF/audio signals that do not suffer from interference and mixing, and no past work can disentangle multiple breathing signals when they are mixed together. In contrast, by modeling different reflections of FMCW as linear combinations of breathing signals, and applying independent component analysis, our system can disentangle breathing signals from mixtures of such signals. Hence, it can accurately monitor multiple users' respiration signals without a minimum separation requirement.

4.3.2 Blind Source Separation and ICA

DeepBreath leverages the literature on blind source separation (BSS) [25]. The problem is defined as the recovery of a set of source signals from a set of mixture signals, without knowing the properties of the original sources or how they are mixed together. If the sources are independent and have non-Gaussian distributions, and the mixing is linear, then one can

use independent component analysis (ICA) to recover the source signals from the mixture signals. ICA recovers the original sources by maximizing the statistical independence of the estimated components in the mixtures [64, 65].

ICA has been used in many domains including speech and image processing, text mining, financial data processing, communication systems, and EEG processing [146, 125, 142, 100, 109, 35]. In this chapter, we extend the applications of ICA to include the recovery of the breathing signals of collocated individuals.

4.4 Modeling Mixtures of RF-Based Breathing Signals

In this section, we discuss how to separate mixtures of RF breathing signals.

4.4.1 Primer

Breathing signals have a variety of patterns that are unpredictable in advance, and the results of multiple people breathing in an environment are seen in a combined form in the observed RF signals. Identifying the individual breathing signals from the combined RF signals falls under the broad framework of *Blind Source Separation*. A common example of such blind source separation is the *cocktail party problem*, where the audio signals from multiple speakers are combined in the environment, and the goal is to separate the audio signals corresponding to each individual speakers. In the particular case that the sources are independent, non-Gaussian, and combine linearly, we can perform Blind Source Separation efficiently using a technique called *Independent Component Analysis (ICA)*.

Traditionally, ICA is defined as follows: Let there be N independent time varying sources, $s_i(t), i = 1 \dots N$, and M different observations, $x_i(t), i = 1 \dots M$. For T time units ($t = 1 \dots T$), we can define the source signals as a $N \times T$ matrix,

$$S_{N \times T} = \begin{bmatrix} s_{11} & s_{12} & \dots \\ \vdots & \ddots & \\ s_{N1} & & s_{NT} \end{bmatrix}$$

and the observations as a $M \times T$ matrix,

$$X_{M\times T} = \begin{bmatrix} x_{11} & x_{12} & \dots \\ \vdots & \ddots & \\ x_{M1} & & s_{MT} \end{bmatrix}$$

The observations X are produced by the sources S combining via a mixing matrix $W_{M\times N}$, such that we can write:

$$X_{M\times T} = W_{M\times N} \times S_{N\times T} \tag{4.1}$$

The goal of ICA is to recover the sources S and the mixing matrix W given only the observations X, provided the sources are independent and non-Gaussian.

4.4.2 The Challenge in Applying ICA to RF Breathing

At first blush, it might seem that applying ICA to recover mixed breathing signals is simple. The RF signals reflected off people's bodies add up linearly over the wireless medium. Further, the breathing signals of different subjects are generally independent and not Gaussian.[4] The problem, however, is that Eq. 4.1 assumes that the mixing matrix W is the same at every time instance. ICA leverages the distribution of the sources during the period T to impose additional constraints to solve the problem. Unfortunately, this assumption does not hold for our problem.

In our problem, the sources are the RF signals reflected off each person's torso. At any point in time, antennas on the radio receive linear mixtures of these reflections. The mixing coefficients, W, are the wireless channels from the torso of each person to each antenna. When the person breathes, his/her torso moves. As a result, the channels change and the mixing coefficients are no longer constant. This means that there is not a single mixing matrix that can be used independent of time in Eq. 4.1.

In the remaining parts of this section, we describe the mathematical structure of the RF signal, how the signal changes with multiple reflectors, and how we can reformulate the mixing problem to allow source identification using ICA.

[4] In the evaluation section, we show the distribution of a person's breathing signal.

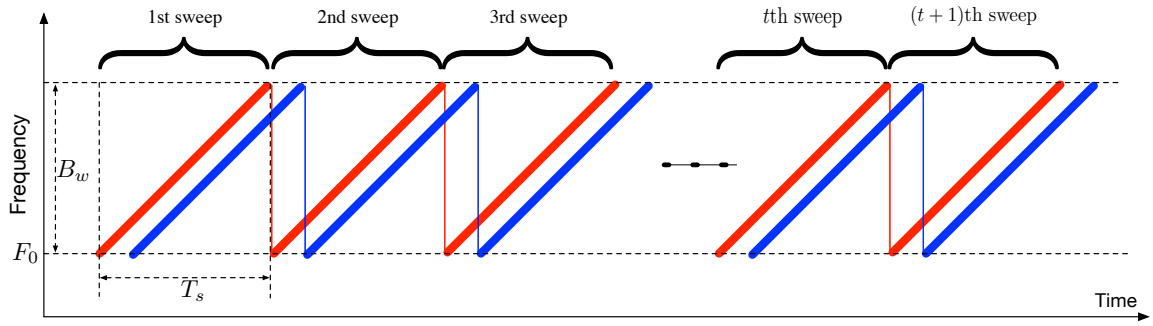


Figure 4-3: Transmitted and received FMCW signal. The red line is the transmitted signal, and the blue line is the received signal reflected by a single reflector. FMCW operates by transmitting a sequence of sweeps, and comparing the frequency difference between the transmitted signal and received signal.

4.4.3 FMCW with a single reflector

We will analyze RF reflections assuming a Frequency-Modulated Carrier Waves (FMCW) radio, which is widely used in the literature [5, 62, 179]. FMCW operates by transmitting a sequence of sweeps. During each sweep, the frequency of the transmitted signal changes linearly with time, as illustrated by the red lines in Fig. 4-3. The human body reflects the signal back. The reflected signal, depicted in blue, is a time-shifted version of the transmitted sweep, where the shift is equal to the time it takes the signal to travel to the reflector and back to the radio. Because the duration of each sweep is usually very short ($\sim 0.1ms$), we assume the reflector does not move within each sweep. For a single reflector at distance $d(t)$ during the t^{th} sweep, the signal's time-in-flight is $\tau(d(t)) = 2d(t)/C$, where C is the speed of light. In this case, the received FMCW signal within the t^{th} sweep can be written in the time domain as [112]:

$$m(d(t), u) = Ae^{j2\pi(F_0\tau(d(t))+K_s\tau(d(t))u-1/2K_s\tau^2(d(t)))}, u \in [0, T_s], \tag{4.2}$$

where A is the amplitude of the received signal, F_0 is the smallest frequency in the sweep, T_s is the *sweeping period*, and $K_s = B_w/T_s$ is sweeping rate defined as the swept bandwidth divided by the sweeping time. For convenience, we also define the following quantities: *carrier-to-bandwidth ratio* $R = F_0/B_w$, and *resolution* $U = C/2B_w$.

In order to compute the frequency response of this time domain signal, we can compute

the Fourier Transform of the signal within each sweep.

For simplification, we use d to denote $d(t)$. Recall inside each sweep $u \in [0, T_s]$, with a reflector at distance d, the received FMCW signal can be written as:

$$m(d,u) = Ae^{j2\pi(F_0\tau(d)+K_s\tau(d)u-1/2K_s\tau(d)^2)}, u \in [0,T_s], \tag{4.3}$$

Because the signal travels with the speed of light, $\tau(d)^2$ is a small amount and can be ignored. Then we have:

$$m(d,u) \approx Ae^{j2\pi(F_0\tau(d)+K_s\tau(d)u)}, u \in [0,T_s] \tag{4.4}$$

Next we compute the frequency response of $m(d,u)$ at frequency f by applying continuous Fourier Transform.

$$l(d,f) = \int_0^{T_s} m(d,u)e^{-j2\pi fu}du = Ae^{j2\pi F_0\tau(d)} \int_0^{T_s} e^{j2\pi(K_s\tau(d)+f)u}du \tag{4.5}$$

Define $v(d,f) \triangleq T_s(K_s\tau(d)+f) = d/U + T_sf$, then we have:

$$\begin{aligned} l(d,f) &= Ae^{j2\pi F_0\tau(d)} \int_0^{T_s} e^{j2\pi v(d,f)u/T_s}du \\ &= Ae^{j2\pi F_0\tau(d)} \frac{e^{j2\pi v(d,f)u/T_s}\big|_{T_s} - e^{j2\pi v(d,f)u/T_s}\big|_0}{j2\pi v(d,f)/T_s} \\ &= AT_se^{j2\pi F_0\tau(d)} \frac{e^{j2\pi v(d,f)}-1}{j2\pi v(d,f)} \end{aligned} \tag{4.6}$$

Noticing that

$$F_0\tau(d) = \frac{F_0}{B_w}(B_w\tau(d)+T_sf-T_sf) = R(d/U+T_sf)-RT_sf = Rv(d,f)-RT_sf \tag{4.7}$$

We can rewrite the Eq. 4.6 as following:

$$l(d,f) = AT_s e^{-j2\pi RT_s f} \cdot e^{j2\pi Rv(d,f)} \frac{e^{j2\pi v(d,f)} - 1}{j2\pi v(d,f)} \tag{4.8}$$

Define $h(f) \triangleq AT_s e^{-j2\pi RT_s f}$, then we have:

$$l(d,f) = h(f) \cdot e^{j2\pi Rv(d,f)} \frac{e^{j2\pi v(d,f)} - 1}{j2\pi v(d,f)} \tag{4.9}$$

Given the fact that

$$\begin{aligned} e^{j2x} - 1 &= (\cos 2x - 1) + j \sin 2x = -2\sin^2 x + 2j(\sin x \cos x) \\ &= 2\sin x(j\cos x - \sin x) = 2j \sin x(\cos x + j \sin x) = 2j \sin x e^{jx} \end{aligned} \tag{4.10}$$

The frequency response at a subcarrier f for a reflector at distance $d(t)$ can be written as:

$$\begin{aligned} l(d,f) &= h(f) \cdot e^{j2\pi Rv(d,f)} \frac{2j \sin(\pi v(d,f)) e^{j\pi v(d,f)}}{2j\pi v(d,f)} \\ &= h(f) \cdot sinc(v(d,f)) \cdot e^{j2\pi(R+1/2)v(d,f)} \end{aligned} \tag{4.11}$$

4.4.4 FMCW with multiple reflectors

Now consider the case with multiple reflectors $r_1, r_2, \ldots, r_N$, at distances $d_1, d_2, \ldots, d_N$. In such a case, the total reflected signal for the N reflectors is simply the sum of the corresponding time domain signals, *i.e.*, $M(t,u) = \sum_{i=1}^{N} m(d_i(t), u)$.

Since Fourier Transform is linear, the frequency domain representation of $M(t)$ can be obtained simply by summing the individual frequency distributions, *i.e*

$$L(d(t), f) = \sum_{i=1}^{N} l(d_i(t), f). \tag{4.12}$$

As can be seen from Eq. 4.11, the function $L(d(t), f)$ can no longer be written as a

linear sum of N independent sources, *i.e.*, one cannot define sources $g(d_1(t)), \ldots, g(d_N(t))$ (note that $g(.)$ are not a function of frequency f) such that we can write the output signal

$$L(d(t), f) = \sum_{i=1}^{N} w_i(f) g(d_i(t)), \tag{4.13}$$

for every f, where the $w_i(f)$ are the mixing coefficients and remain constant for all the sweeps. This prevents us from directly applying ICA to the signal $L(d(t), f)$, as described in Section 4.4.2.

4.4.5 Making ICA work for FMCW with multiple reflectors

Our key insight, however, is that we can create such a decomposition by considering only the small linear motions involved with breathing. Specifically, we can write $l(d_i(t), f)$ as $l(D_i + \delta_i(t), f)$ where D_i is the mean position of the reflector (i.e., the mean position of the chest during breathing), and $\delta_i(t)$ are the small time varying motions corresponding to breathing. Using the Taylor series expansion till the first order term, this function can be approximated as: $l(d_i(t), f) = l(D_i, f) + \delta_i(t) l'(D_i, f)$, where $l'(D_i, f)$ is the derivative with respect to distance. The total frequency response for all the N reflectors can then be written as:

$$L(d(t), f) = \sum_{i=1}^{N} l(D_i, f) + \sum_{i=1}^{N} l'(D_i, f) \delta_i(t) \tag{4.14}$$

In Eq. 4.14, note that the first term is simply the mean frequency response across all time units, which we can subtract from the signal without affecting the estimation of the breathing signal. The second term is exactly of the form in Eq. 4.13 desired by ICA, where the derivative term corresponds to the desired ICA mixing coefficient, and the $\delta_i(t)$ corresponds to the time varying source relevant to the breathing motion.

Thus, we can apply ICA algorithms to this modified signal, *i.e.* after subtracting the mean across time for each frequency and recover the individual breathing motions of all the sources.

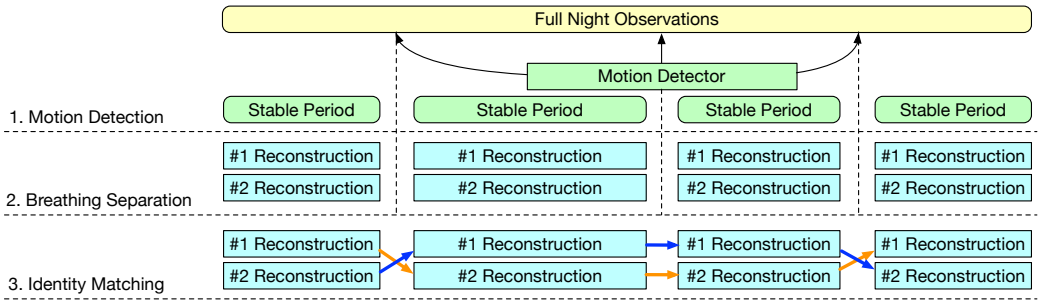


Figure 4-4: The architecture of DeepBreath. DeepBreath has three components: a motion detector to detect motions disruptive to ICA and generate stable periods, a breathing separation module that adapts ICA to recover breathing signal candidates, and an identity matching module to stitch all reconstructed breathing signals that belong to the same person.

Formally, we define our observations for the ICA as:

$$O(f,t) = L(d(t),f) - \sum_{i=1}^{N} l(D_i,f) = \sum_{i=1}^{N} l'(D_i,f)\delta_i(t). \tag{4.15}$$

As described above, these observations are linear combinations of the source signals. We can therefore use ICA to separate the source signals.

4.5 DeepBreath

DeepBreath is the first RF-based full night breathing reconstruction system that can accurately monitor multiple people even when they share the same bed. The breathing signals extracted by DeepBreath can be directly analyzed to learn the health status of the individuals, or fed to a sleep analysis system like the one described in [179].

DeepBreath runs on top of an FMCW radio equipped with an antenna array. The radio transmits a low power wireless signal and captures its reflections from users in the vicinity. Multi-antenna FMCW radios allow us to capture the RF signal from a particular location in space [93] – i.e., they allow us to obtain multiple concurrent RF signals, each corresponding to the signal reflected from a particular voxel in space. We use this property to focus on RF signals from voxels in and around the bed. Each such RF signal is a mixture of the original breathing signals. We call such mixtures, "observations". The difference between

observations stems from differences in the contribution of each breathing signal to that particular observation, which depends on the position of the person and the location that the observation focuses on.

DeepBreath takes as input a set of full-night observations and returns the breathing signals of each person. It does so by following a three-step process illustrated in Fig. 4-4 and summarized below:

- *Motion Detection:* The motion detection component takes as input observations, identifies motion intervals, and splits the observations into a series of stable periods, as shown in the first row of Fig. 4-4.

- *Breathing Separation:* This module processes the observations during each stable period to disentangle the breathing signals of different people. It outputs the reconstructed breathing signals during that period, as shown in the second row of Fig. 4-4.

- *Identity Matching:* The breathing separation module is not aware whom each reconstructed signal belongs to. In order to create full-night reconstruction, DeepBreath has a special module that stitches all reconstructed breathing signals that belong to the same person, as shown in the third row of Fig. 4-4.

Below we describe these components in detail.

4.5.1 Robust Human Motion Detector

When there are large motions, ICA will fail for two reasons. First, the frequency response of a large motion can no longer be approximated by the first order Taylor expansion, therefore its corresponding observations are no longer linear related to each other. Second, a large motion usually represents a change of posture of the user. He may move slightly away from his original position, and cause a change in the signal mixing pattern. Therefore, we should segment the signal when we detect large motions and apply ICA to each stable period separately.

Ideally, we want a motion detector that raises a flag only when one of the people moves but ignores other sources of motion. This is because only a movement of one of the breath-

ing people disrupts the ICA. Other sources of motion in the environment (e.g., the HVAC) do not affect the contribution of each breathing signal to the observations.

To distinguish movements of the monitored individuals from other sources of motion, we rely on the following intuition. When one of the monitored people moves, the motion affects the breathing signal in every observation. On the contrary, environmental motions are usually further away from the person and they affect the breathing signal due to the sinc effect. Thus when an environmental motion occurs, the breathing pattern will be detectable in some observations.

In order to utilize the above intuition, we define the following terms:

Definition 7. Short Observation *A short observation is a small period of a observation with a fixed duration.*

We assume that the human's breathing rate is constant during a short period. This means that breathing can be assumed periodic during a short observation. Therefore, the periodicity of a short observation can represent its signal quality. Then we have:

Definition 8. Short term Breathing-to-Noise Ratio (s-BNR) *$\mathcal{B}_s$ of a short observation o is defined as the ratio of breathing energy to the overall energy in observation o.*

We computed B_s by first taking an FFT of the short observation signal. We then find the FFT bin with maximal energy within the human breathing range. We compute the ratio between that bin's energy and the energy sum of all FFT bins. In our computation, we use 15 seconds for the default duration of short observations and 10 to 30 breaths/minute for the human breathing range [48, 124]. The larger the s-BNR, the more periodic the short observation is, then more likely the short observation contains good breathing signal.

Thus, an observation is a time series of short (15-second) observations, each of which has a particular B_s. Next, we define "a motion tableau" over all observations as follows:

Definition 9. Motion Tableau *is a matrix with columns representing discretized time slots and rows representing observations at different locations. Each (i, j) cell represents the short observation at the jth location during ith time slot. The value of cell (i, j) is the s-BNR of that short observation.*

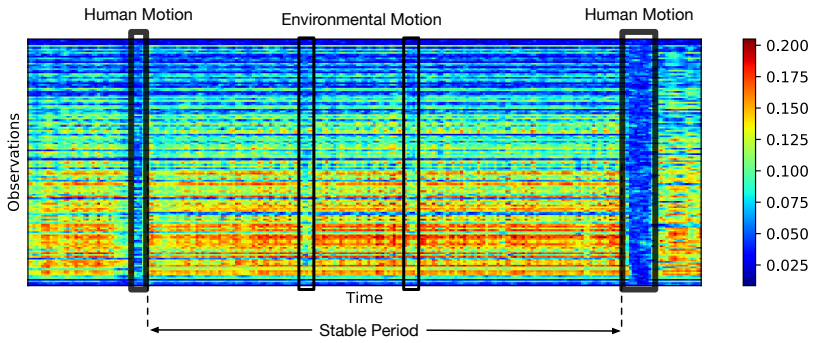


Figure 4-5: An example of Motion Tableau. More reddish the cell is, higher the s-BNR of that short observation is. During this period, we can see three typical patterns. When there are no motion, a large part of short observations have high s-BNR. When there are environmental motions, some observations are affected, but there still exists high s-BNR observations. And when there are human motions, nearly all observations are destroyed. We call the period between two human motions as a stable period.

Figure 4-5 shows an example of motion tableau computed over a full night of sleep of a couple in our dataset. The figure shows that the motion tableau has the following desirable property: Different kinds of motion have different patterns. Human motion usually causes all short observations during that period to have low s-BNR, whereas environmental motion does not affect the s-BNR much.

Then, instead of using hand-crafted features that are hard to tune, we can treat the motion tableau as an image and train a Convolutional Neural Network(CNN) based classifier to detect human motion. CNNs are well-known for its good accuracy and robustness on image classification tasks. Here, we adopt a classic VGG16 [131] architecture with adjustment of the input size and output dimension. The CNN classifies each column in the motion tableau to identify whether it experiences human motion. To classify the ith column, we feed the CNN a small image that includes all rows and the columns $[i-k, i+k]$, where k is a small number that allows the CNN to look at the context around the particular column of interest. The CNN outputs "one" to indicate movements of the monitored people and "zero" otherwise.

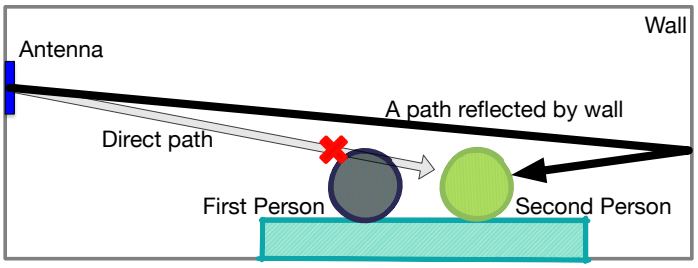


Figure 4-6: A multipath example. The direct path from the device to the second person is blocked by the first person. Yet the second person can be reached by signals reflected by a wall.

4.5.2 Breathing Separation

After motion detection, we can divide a single night into a series of stable periods. Ideally, we can apply ICA to each stable period and obtain the breathing reconstructions. Yet, there are still two practical challenges:

1. The direct path of the farther person may be completely blocked. Thus, we cannot assume breathing signals are always located in the bed area.

2. Noise may completely overwhelm breathing signals in some observations. Such observations no longer satisfy the linearity property and are harmful to the reconstruction.

Therefore we propose a breathing quality filter to address both challenges.

Dealing with Blockage

To solve the blockage problem, we leverage a well-known phenomenon called multipath. Wireless signals can be reflected by walls and furniture, and can reach the receiver even though the direct path is not available. Fig. 4-6 shows a typical example. Although multipath signals have a different time-of-flight compared to the direct path signal, they are all linearly related to the origin breathing motion. Therefore, multipath signals provide the same level of information as direct path signals. So, instead of only focusing on bed area, we use observations from the larger space around the bed to gather as much information as possible.

Filtering Noisy Observations

As explained earlier, we need to consider observations in the larger area around the bed, so that we may recover the breathing of a person whose direct path to the receiver is occluded. On the other hand, for observations that are located far away from the person, the breathing signals will be suppressed and noise will become the main component. Feeding such noisy observations to ICA is harmful since ICA will then try to reconstruct the noise pattern.

To filter out noisy observations and keep the useful ones, we again rely on the periodicity of human breathing. Specifically, a good observation should have many short observations (sequences of 15 seconds) that exhibit high s-BNR. Thus, we propose the following definition to measure the signal quality during long observation, i.e. the observation over a whole stable period:

Definition 10. Long term Breathing-to-Noise Ratio (l-BNR) *$\mathcal{B}_l$ of a long observation o is computed as the average of the s-BNR of the series of short observations that constitute the long observation.*

A low l-BNR indicates a low possibility that the observation contains useful breathing information, and hence such observations should not excluded in recovering breathing signals. Once we filter out such low l-BNR observations, we can apply our model introduced in Sec. 4.4 and reconstruct the breathing signals of the people in the scene.

4.5.3 Identity Matching

After we apply the breathing separation algorithm to each stable period, we get several ICA components for each of them. Assuming, in each stable period, that the ICA components are all successfully reconstructed breathing signals, the last step to get the full night breathing of every person is to figure out which ICA component corresponds to which person. We call this problem, *identity matching*.

Identity matching is non-trivial even in the case of two people. Say we have two people p_1 and p_2. In each stable period, we have two ICA components c_1, c_2. There are two choices when matching these components: c_1 is p_1's breathing, c_2 is p_2's breathing or vice versa. One now needs to make such a choice for each stable period. Because we have at least two

Table 4.1: All key notations of identity matching module

Symbol	Meaning
N	number of people
M	number of stable periods
K	number of ICA components in each stable periods
p_n	nth person ($n = 1, .., N$)
q_m	mth stable periods ($m = 1, .., M$)
c_m^k	kth ICA component in stable period q_m ($k = 1, .., K$)
$\mathcal{M}(c_m^k)$	mixing vector of ICA components c_m^k
$\mathcal{M}(c_m^k)_i$	ith element in mixing vector $\mathcal{M}(c_m^k)$
$\mathcal{I}(\cdot, \cdot)$	identity consistency metric for two ICA components
$\mathcal{I}_\sigma(\cdot, \cdot)$	identity consistency metric for two ICA assignments
$\sigma_m : \{1, .., N\} \rightarrow \{1, .., K\}$	ICA assignment in stable periods q_m

possible matchings in each stable period, the valid solution space grows exponentially with the number of stable periods. Things can become much more complex when the number of people or the number of ICA components are larger than two.

To solve the identity matching problem, we start by defining a consistency metric that captures the likelihood that two ICA components represent the breathing signal of the same person. We then formulate identity matching as an optimization problem that assigns identities to maximize the consistency metric. We show how to solve the optimization problem efficiently using dynamic programming. We explain this process in detail below. Table 4.1 summarizes the notations used this section.

Identity Consistency Metric.

We define an identity consistency metric $\mathcal{I}$. Metric $\mathcal{I}$ takes two ICA components c_1, c_2 (from different stable periods) as input, and gives a consistency score $\mathcal{I}(c_1, c_2)$ as output. The higher the consistency score, $\mathcal{I}(c_1, c_2)$ is, the more likely the ICA components c_1, c_2 are the breathing of the same person during two stable periods.

To compute $\mathcal{I}$ we leverage the mixing vector of the ICA components. The mixing vector $\mathcal{M}(c)$ is one column of the mixing matrix (W in section 4.4) that corresponds to ICA component c. The length of the mixing vector is the number of original observations we used to generate the ICA components. Each entry of the mixing vector represents how this ICA component c contributes to its corresponding observation. Two ICA components

having similar mixing vectors means that they contribute similarly to the RF observations. Thus they are more likely to represent the same source signal.[5]

Then we mathematically define the ICA component identity consistency metric $\mathcal{I}$ as follows:

$$\mathcal{I}(c_1, c_2) = \sum_i \frac{\min(\mathcal{M}(c_1)_i, \mathcal{M}(c_2)_i)}{\max(\mathcal{M}(c_1)_i, \mathcal{M}(c_2)_i)} \tag{4.16}$$

where the index i runs over all the entries of the corresponding mixing vectors.

Consistent Identity Assignments.

We introduce the ICA assignment σ_m for stable periods q_m. Each assignment can be viewed as a permutation of the ICA components. Specifically, $\sigma_m(n) = k$ means ICA component c_m^k is assigned as person p_n's breathing in ICA assignment σ_m.

We compute the consistency score of two assignments for two stable periods by computing the consistency between the components they assign to the same person, as follows:

$$\mathcal{I}_\sigma(\sigma_m, \sigma_{m'}) \triangleq \sum_{n=1}^{N} \mathcal{I}(c_m^{\sigma_m(n)}, c_{m'}^{\sigma_{m'}(n)}), \tag{4.17}$$

where the sum is over all people and $\mathcal{I}(c_m^{\sigma_m(n)}, c_{m'}^{\sigma_{m'}(n)})$ is the consistency between a pair of ICA components that were assigned the same identity by assignments σ_m and $\sigma_{m'}$.

The score $\mathcal{I}_\sigma(\sigma_m, \sigma_{m'})$ is used to measure whether ICA assignment σ_m and $\sigma_{m'}$ reorder the ICA components in a *consistent* way. By saying *consistent*, we mean that after the reordering, ICA components having the same order in two stable periods do actually represent the breathing of the same person.

Objective Function and Solution.

Our goal is to find ICA assignments σ_m for each stable periods such that after reordering, an ICA component having the same order in all stable periods is the same person's breathing.

[5]Note that since ICA is not sensitive to scale, mixing vectors of different ICA components may have different scales. Thus before doing comparison, we have to normalize the mixing vectors.

Thus, we define the following objective function $\mathcal{J}$.

$$\mathcal{J}(\sigma_1, .., \sigma_M) = \sum_{m=1}^{M} \sum_{m'=m+1}^{\min(M, m+h)} \mathcal{I}_\sigma(\sigma_m, \sigma_{m'}) \tag{4.18}$$

The objective $\mathcal{J}$ simply sums all the ICA assignment identity consistency scores for all pairs of ICA assignments in two different and *temporally close* stable periods. Here, two stable periods q_m, $q_{m'}$ are considered as *temporally close* if there are less than h stable periods between these two stable periods .i.e $|m-m'| \leq h$ (our default is h=12). The reason why we limit ourselves to nearby stable periods is that the consistency metric I is valid only over a short period. Specifically, the consistency metric captures the fact that when a person tosses around in bed, or moves on a chair or couch, he is still in the same general location and hence the mixing matrix of his breathing changed by a small amount. However, if the person keeps moving, then eventually the mixing matrix can change significantly.

The above objective function captures the idea that if each pair of ICA assignments orders ICA components in a consistent way then all the ICA assignments order the ICA components consistently. Thus, the optimal set of assignments can be represented as:

$$\sigma_1^*, .., \sigma_M^* = \arg\max \mathcal{J}(\sigma_1, .., \sigma_M) \tag{4.19}$$

To solve the optimization in Eq.(4.19), we design a dynamic programming algorithm which performs two passes, forward and backward. In the forward pass, we have initial conditions f_h and DP-equation f_m and g_m as follow:

$$f_h(\sigma_1, .., \sigma_h) = \sum_{i=1}^{h} \sum_{j=i+1}^{h} \mathcal{I}_\sigma(\sigma_i, \sigma_j) \tag{4.20}$$

$$f_m(\sigma_{m-h+1}, .., \sigma_m) = \max_{\sigma_{m-h}} f_{m-1}(\sigma_{m-h}, .., \sigma_{m-1}) + \mathcal{I}_\sigma(\sigma_{m-h}, \sigma_m), \quad m = h+1, h+2, .., M \tag{4.21}$$

$$g_m(\sigma_{m-h+1}, .., \sigma_m) = \arg\max_{\sigma_{m-h}} f_{m-1}(\sigma_{m-h}, .., \sigma_{m-1}) + \mathcal{I}_\sigma(\sigma_{m-h}, \sigma_m), \quad m = h+1, h+2, .., M \tag{4.22}$$

In the backward pass, we can get optimal solution $\sigma_1^*, .., \sigma_M^*$ in a reverse order based on

the DP-functions computed in forward pass.

$$\sigma^*_{M-h+1}, .., \sigma^*_M = \arg\max f_M(\sigma_{M-h+1}, .., \sigma_M) \quad (4.23)$$

$$\sigma^*_m = g_{m+h}(\sigma^*_{m+1}, .., \sigma^*_{m+h}), \quad m = M-h, M-h-1, .., 1 \quad (4.24)$$

4.6 Evaluation

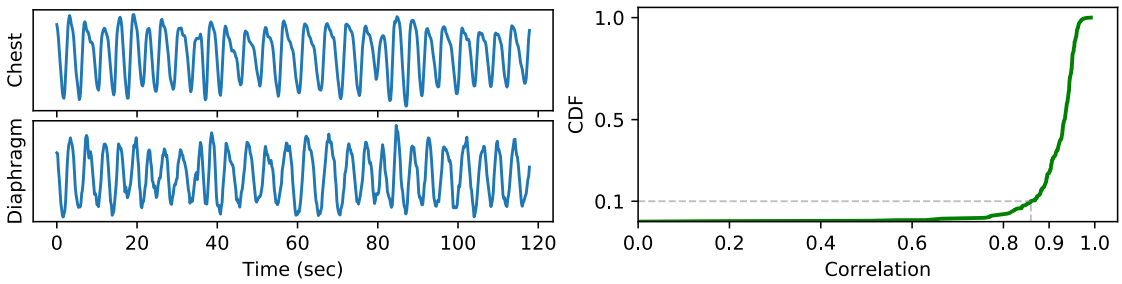
In this section, we empirically evaluate our theoretical model and DeepBreath's performance. All experiments with human subjects are approved by our IRB and we have obtained informed consent from every subject. For the hardware, we have implemented a state-of-art FMCW radio that sweeps from $5.7GHz$ to $7.2GHz$ and transmits at sub-milliwatt power in accordance with the FCC regulations. The radio is equipped with an antenna array to improve spacial resolution.

Ground Truth: To obtain the ground truth breathing of people, we use SleepView[138], an FDA-approved sleep and apnea monitor. The user sleeps while wearing a special belt that uses respiratory inductance plethysmography to measure the expansion of the chest, which it translates to breathing signal.

Metrics: To evaluate our breathing reconstruction results, we use the following two metrics:

- *Pearson correlation coefficient (correlation):* We evaluate the similarity between the reconstructed signal and the ground truth by computing their correlation. The correlation is computed over stable periods. The aggregate per-night correlation is computed by taking a weighted average of the correlation over stable periods where the weight corresponds to the length of each period (normalized by the total duration of stable periods throughout the night).

  The optimal value for the correlation is 1. This would happen if the reconstructed signal is exactly the same as the ground truth. Of course in practice, the correlation is never 1. Thus, it is natural to ask what correlation value is good enough. To answer this question we compare the correlation between two respiration belts on the same



(a) Signals captured from chest and diaphragm (b) Correlation CDF between two signals

Figure 4-7: Comparison between chest level and diaphragm level breathing signals. Fig. 4-7a shows a 2 minutes sample of two breathing signals, and Fig. 4-7b shows the the correlation CDF between two breathing signals.

person: one is strapped around the chest and the other around the diaphragm area. Both belts capture the person's breathing but at slight different locations on the body. Fig. 4-7a plots an example that shows the breathing signal of the same person captured at the chest and diaphragm levels. The figure shows small differences between the two signals. For quantitative results, we plot in Fig. 4-7b a CDF of the the correlation between two breathing signals. The data in the figure are taken over 2-minute windows from 6 individuals over 6 nights. As can be seen from the figure, the two breathing signals are slightly different –i.e. their correlation is not 1. The average correlation is 0.915 and 90th percentile is 0.860. This CDF provides a point of comparison. Specifically, since the RF signal captures reflections from multiple areas of the human body, an ideal reconstruction would show a correlation comparable to the correlation between the breathing signals captured in different locations on the body.

- *Breathing Rate:* Breathing rate is the number of breaths per minute. We compare our reconstructed signal to the ground truth breathing rate as captured by the respiration belt. As in past work, we compute the breathing rate over 2-minute windows.

4.6.1 Empirical Validation of Our Model

The use of ICA to separate mixed signals is subject to three requirements: 1) the mixing is linear, 2) the sources are independent, and 3) the distributions of the source signals are not Gaussian.

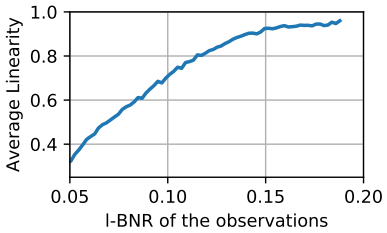


Figure 4-8: Linearity of an observation as a function of its l-BNR. Blue line represents the average linearity of observations with certain l-BNR and grey area shows the standard deviation. The figure shows that observations with high l-BNR are linear functions of the breathing signals.

We empirically check that our formulation of the breathing separation problem satisfies all three requirements. We conduct an experiment in which two people sit shoulder-to-shoulder on a couch. Each subject is asked to wear a breathing belt to record his ground-truth breathing signal. The subjects breathe normally with no restrictions. Each trial lasts for 5 minutes. In total, we invite 10 people to participate in this experiment and conduct 20 trials.

(a) Linearity of the Observations: To check whether the observations are linear combinations of breathing signals, we measure the ***linearity*** of a given observation $\mathcal{O}(t)$ as follows:

$$l_{\mathcal{O}} = \max_{a_1,a_2} |\rho(a_1 s_1(t) + a_2 s_2(t), \mathcal{O}(t))|, \tag{4.25}$$

where $s_1(t)$ and $s_2(t)$ are the ground truth breathing signals, $\mathcal{O}(t)$ is the observation signal, and $\rho()$ is the Pearson correlation coefficient function. This metric computes the correlation between the observation and the closest linear combination of the signals.

We plot in Fig. 4-8 the linearity of observations as a function of their l-BNR. The figure shows that the linearity of an observation increases with l-BNR. This is expected because observations with low l-BNR are dominated by noise. The figure also shows that when the l-BNR is higher than 0.15, linearity is higher than 0.9. Thus, we can treat high l-BNR observations as linear functions of the breathing signals.

(b) Independence of the Source: Second, we examine the correlation between the breathing of two subjects as a function of the duration of an observation. For each duration t, we divide the duration of the experiment to non-overlap periods of length t. For each such

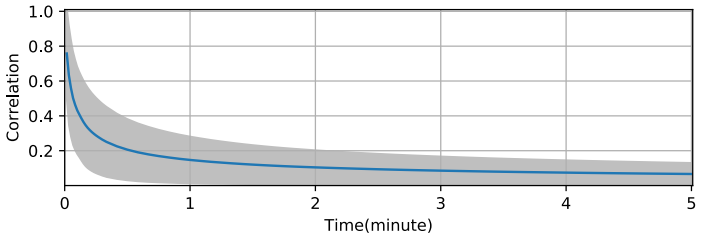
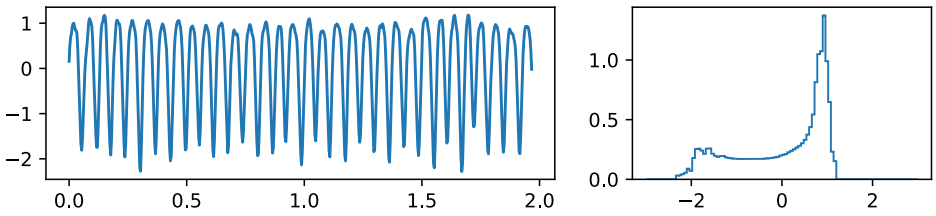


Figure 4-9: Correlation between the breathing of two subjects over windows with different duration. Blue line represents the average correlation between subjects and and grey area shows the standard deviation.



(a) Example Breathing Signal (b) Example Distribution

Figure 4-10: Figure shows that marginal distribution of a subject breathing is non-Gaussian

period, we compute the correlation between the breathing of the two subjects in the experiment. We repeat this computation for all subjects and trials. Fig. 4-9 plots the relationship between the correlation of the subject's ground-truth breathing and the duration of the observation. The blue line represents the averaged correlation, and the grey area shows the standard deviation. The figure shows that the correlation of the subjects breathing signals decreases as the duration of the observation increases, and is lower than 0.1 when the duration is larger than 2 minutes. Thus, it is safe to assume that the breathing signals of different subjects are statistically independent over periods equal or larger than two minutes.[6]

(c) Non-Gaussianity of the Sources: Finally, we look at the distribution of the breathing signals of our subjects. We show a representative example of the ground truth breathing of a subject in Fig. 4-10a and its distribution in Fig. 4-10b. Clearly the distribution is not Gaussian. The distributions of other subjects are similarly non-Gaussian.

[6]Note that two signals can look similar but still have low correlation if the frequency of breathing is slight different or the breathing cycles are unsynchronized.

4.6.2 Evaluation of DeepBreath's Performance

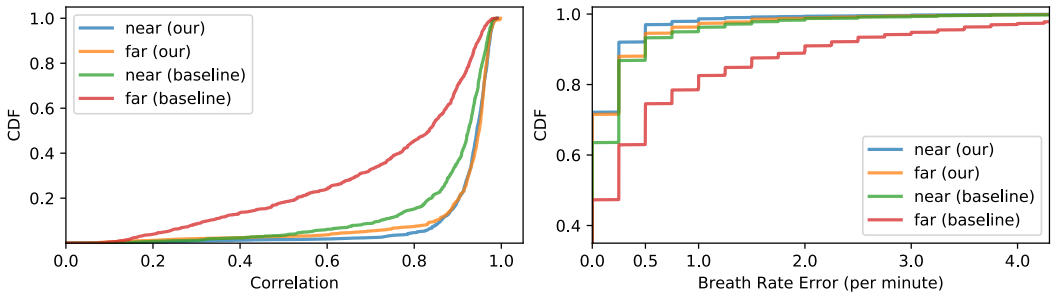
We deploy our system and evaluate it with 13 different couples. The experiments are run in the couple's own homes. All monitored couples sleep in the same bed. The bed size ranges from twin ($1m$ wide) to king ($2m$). Each subject wears a breathing belt for the whole night to obtain the ground truth breathing. We install our device at the bedside. We refer to the subject who sleeps closer to the device as the *near* subject and the other subject as the *far* subject. In total, we have collected 21 nights, and 151 hours of data.

We first generate stable periods for each night by detecting all motion using the motion detector. We remove periods with motion and operate on stable periods. Our analysis shows that, on average, 11% of a night exhibits motion. This is compatible with medical literature [34].[7] We then process the stable periods and compare them with the ground truth breathing signals.

We also compare the performance of DeepBreath with a baseline that uses an oracle to iterate over all voxels in the bed area, and for each person, it zooms in on the voxel that results in the most accurate breathing signal for that person. This baseline highlights the importance of disentangling the breathing signals as opposed to zooming in on the best RF signal that corresponds to the person's breathing, as in past work. Note that the baseline is a conservative representation of past work since we allow it to look at the actual ground truth breathing as it picks the best signal.

Fig. 4-11a plots a CDF of the correlation between the reconstructed breathing signals and the ground truth for both DeepBreath and the baseline. The figure shows that DeepBreath's average correlation is 0.920 and 0.908 for the near and far subjects respectively. This is comparable with the average correlation between two belts that capture breathing at two locations on the same person. In comparison, the baseline achieves a correlation of 0.874 and 0.733 for the near and far subjects respectively. This performance is significantly worse than DeepBreath's, particularly for the far person. Note that, as mentioned earlier, the baseline is allowed to look at the ground truth signals so that it can zoom in on the RF signals with the best correlation, which is not feasible in practice. Despite this extra

[7] Healthy people experience around 30 minutes of intermittent awakenings each night [34]. Since we monitor a couple, we detect motion when either of them moves.



(a) CDF of breathing correlation

(b) CDF of breathing rate errors

Figure 4-11: Performance comparison between DeepBreath and an oracle-based baseline.

information, it is still unable to match DeepBreath's performance.

Fig. 4-11b plots a CDF of the error in estimating the breathing rate for both DeepBreath and the baseline. The median breathing rate error in DeepBreath is 0.121 and 0.158 for the near and far subjects. In contrast, the baseline shows a breathing rate error of 0.200 and 0.633 for the near and far subjects respectively.

There are two reasons for the lower performance of the baseline. First, for most cases, since the subjects are fairly close, their breathing signals are mixed in all locations in space. Thus, sometimes it is not possible to find a good voxel that includes breathing from only one subject. Second, the signal from the near person can be much stronger than the far person, and the body of the near person can sometimes occlude the body of the far person. DeepBreath does not suffer from these problems because it is intrinsically designed to separate mixed signals. Further, it combines information across all observations and can leverage multipath effects to reconstruct the breathing of the far person.

Finally, Fig. 4-12 shows a few examples of our reconstructed breathing signals and the corresponding ground truth signals. The figure shows that DeepBreath is able to separate a couple's respiration signals even when their breathing patterns look similar. It also shows that DeepBreath can deal with irregular breathing patterns such as those experienced by the far person in couple 2 and couple 3.

4.6.3 Evaluation of the Components of DeepBreath

The above result shows that DeepBreath can correctly recover the breathing of each individual even when they share the bed. In this section, we zoom in on the identity matching

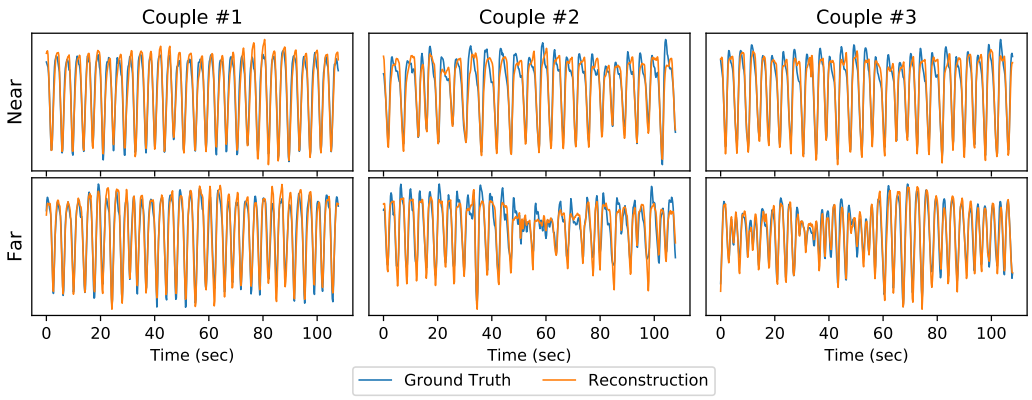


Figure 4-12: Examples of recovered breathing signals and the original ground truth signals from the breathing belt. The figure shows that DeepBreath is accurate even when the monitored couple have similar breathing rates or their breathing signals have irregular patterns.

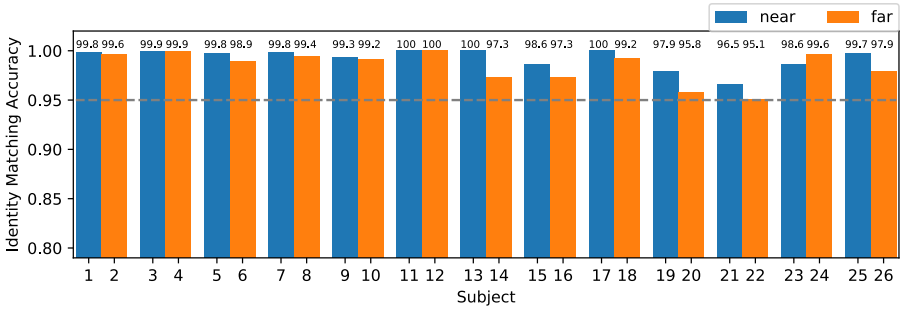


Figure 4-13: Accuracy of identity matching. Each bar shows the identity matching accuracy of each subject.

and motion detection components to understand their performance.

Evaluation of Identity Matching: We report the accuracy of identity matching for all stable periods in the collected 21 nights. We compute the average accuracy weighted by the duration of each stable period. The total averaged accuracy is 99.1% and accuracy for each subject is shown in Fig. 4-13. Accuracies of all the subjects are above 95%, which shows the robustness of our identity matching algorithm.

Evaluation of Motion Detection: We also evaluate the motion detection component. To obtain the ground truth motion we have asked 4 subjects to sleep while wearing accelerometers on each ankle and each wrist. In this experiment, each subject sleeps in a separate bed to ensure that the motion captured by the accelerometers is as close to the ground truth as possible. Each subject also wears a breathing belt for ground truth breathing.

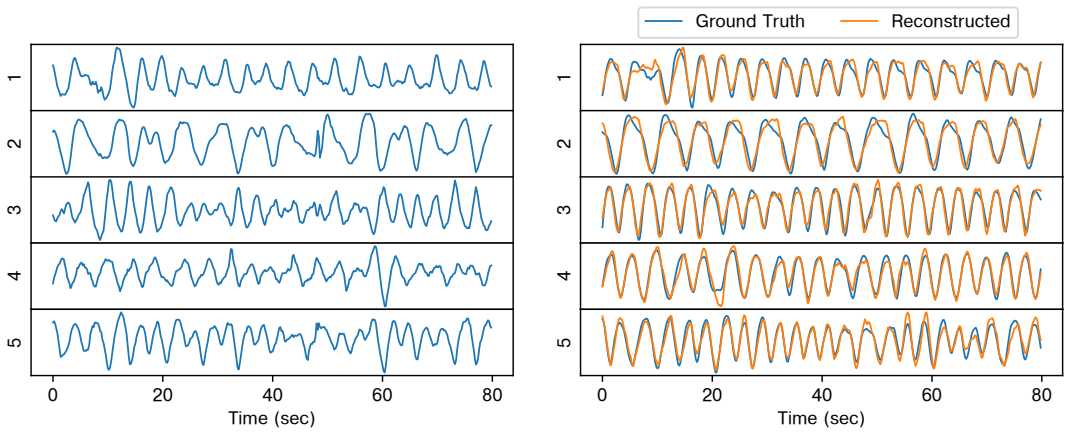
Figure 4-14: Experiment setup. 5 subjects are sitting shoulder-to-shoulder on a $1.9m$ couch.

Accelerometers are affected by gravity. To eliminate this effect we subtract the average acceleration and consider only changes. Because the reading from the accelerometer is noisy, for each 30 second period, we say there is a human motion if the acceleration is 2 times larger than the standard deviation. We declare motion if any accelerometer experiences a motion event. We compare the output of our motion detector to the result of predicting motion based on the four accelerometers. On average we achieve a precision of 0.933 and a recall of 0.954. This result shows that our motion detector successfully detects motion.

4.6.4 Breathing Separation with Many People

Finally, we would like to show that DeepBreath is not limited to two people and can scale to a larger number of users. We conduct an experiment with 5 subjects sitting shoulder-to-shoulder on a $1.9m$ couch, as shown in Fig. 4-14. The subjects wear a breathing belt to capture the ground truth breathing signals. In total we invited 8 people to participate in this experiment, and we conducted 3 trials with different subject combinations. Each trial lasts for 5 minutes.

Fig. 4-15a shows the observations at the voxels centered on each of the five subjects. Each observation is a different mixture of the breathing signals of the subjects. Fig. 4-15b plots the ground truth and the reconstructed breathing signals. As can be seen, even



(a) Observations at each subject's location

(b) Ground truth and reconstructed signals

Figure 4-15: Five people experiment. Observations at each subject's location are plotted in Fig. 4-15a. Because there are no gaps between subjects, breathing signals are mixed together. But DeepBreath can reconstruct the breathing of each subject accurately, as shown in Fig. 4-15b, with an average correlation of 0.92.

though every observation is a mixture of multiple subject's breathing, our algorithm can still reconstruct the breathing of each subject accurately. On average, we achieve 0.922 correlation and 0.034 breathing rate error with respect to the ground truth breathing signals. These results demonstrate that DeepBreath can reconstruct the breathing of at least 5 people even when there is nearly zero distance between them.

4.7 Conclusion

This chapter presents DeepBreath, the first wireless system that disentangles breathing signals of multiple individuals from mixtures of such signals. As a result, it can provide contactless respiration monitoring of multiple people even when they have zero distance between them. A user study with 13 couples who used DeepBreath for 21 nights in their own homes shows that DeepBreath is highly accurate.

We believe DeepBreath opens the door for contactless multi-person sleep studies. Its output can directly serve as the input for sleep stage classifiers [179, 89], or apnea detectors [152, 7]. As such it extends the benefits of such contactless monitoring to couples and individuals sharing the same bed.

Chapter 5

Monitoring Sleep Postures

5.1 Introduction

Shown by significant clinical research, sleep posture is a valuable marker of disease progression, and has a significant impact on health. For instance, patients with Parkinson's disease often suffer from loss of axial movement; and less frequent nocturnal turnovers and longer periods spent recumbent or supine (*i.e.*, facing up) are associated with deterioration in the condition of Parkinson's patients [99]. Similarly, infrequent changes in sleep posture can lead to pressure ulcers in the elderly and post-surgery patients [51]. Studies have also demonstrated that sleeping in a supine position can reduce back pain since it is the position in which the muscles have the least amount of work to do to maintain one's posture against the force of gravity [36]. In contrast, if one has obstructive sleep apnea (OSA), the supine position becomes the worst posture because it imposes unfavorable airway geometry and reduces lung volume. Studies have shown that more than half of all OSA cases can be classified as supine related [95, 104]. Improper sleep posture can even be fatal – sleeping on the stomach can boost the risk of sudden infant death syndrome (SIDS) [39] and sudden death in epilepsy patients [81, 70]. These examples highlight the importance of continuous and fully automatic sleep posture monitoring. Such monitoring can provide doctors with information to better manage patient conditions; it can also provide people themselves information to adjust their posture and reduce their health risks.

Unfortunately, today, there is no good way to provide such sleep posture monitoring.

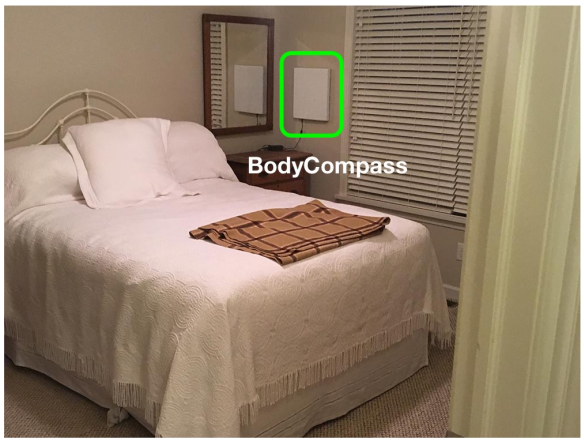


Figure 5-1: BodyCompass in one of our deployments. The white box mounted on the wall is the radio. It uploads the RF signals to the cloud where the model processes them to extract sleep posture.

Doctors typically resort to asking patients about their sleep posture, an error-prone mechanism since people routinely, and unknowingly, change their postures while sleeping. Automated monitoring systems primarily fall into two categories. The first category is vision-based. These methods use a camera to monitor the user's sleep, then extract postures from recorded videos with a machine learning system. Deploying cameras in people's bedrooms, however, is privacy-intrusive. Furthermore, cameras have difficulties tracking body posture if the person is covered or lighting is bad, both of which are typical scenarios when sleeping. The second category uses various kinds of on-bed sensors. Such methods require the user to fix the sensor to the surface of the mattress, which can affect sleep comfort.

Ideally, one desires a system that is non-contact, non-intrusive, and works even in dark scenarios typical of sleeping conditions. In this chapter, we present *BodyCompass*, an RF-based system for sleep posture monitoring. BodyCompass analyzes the reflections of RF signals to infer subjects' sleep postures. It does so without requiring users to wear or be in contact with any sensors. It is not invasive of privacy, and can also work in the dark. Unlike much previous work, BodyCompass has also been demonstrated to work in the wild - with real subjects sleeping in their own homes, and can generalize to new environments with minimal additional training. Fig. 5-1 shows BodyCompass deployed in the home of one of our users.

But how can one extract the sleep posture from radio signals? Our idea is to use the

multipath effect, a known phenomenon in RF communication systems that refers to the fact that RF signals bounce off different objects and obstacles in the environment and reach the receiver through multiple paths. Past work has shown that the human body acts as a reflector in the low-GHz frequencies, commonly used in commodity radios [2, 178]. As the RF signal is incident upon the human body, it reflects from the body based on the body orientation and bounces off the surrounding objects and walls creating a multipath signature indicative of the body posture. Our objective is to learn an inverse map that receives the reflected multipath profile and attempts to infer the body posture. A key challenge in delivering this idea is that the RF signal bounces off many objects in the environment, not just the human body. Only a subset of the signal path involves reflections from the human body, and hence is relevant to the sleep posture. Thus, one has to extract only the RF reflections that bounced off the human body either directly or indirectly in order to determine the sleep posture.

To address this challenge, we leverage DeepBreath (Chapter 4) to extract a person's breathing from RF signals. Our intuition is that all paths that bounce off a person's trunk (e.g. chest and belly) during their sleep are modulated by the person's breathing, and hence we can use this property to disentangle these reflections from the rest of the reflections. Specifically, we use standard techniques to separate signals along different paths (FMCW and angle of arrival), and correlate these separated signals individually with the subject's breathing signal to identify the specific signals corresponding to the person in bed. We further design a neural network model that takes this breathing filtered multipath profile, and predicts the sleep posture of the person.

A key question with such a system is how well the neural network model works with different people and in different homes. While RF reflections and the multipath effect naturally depend on the environment, one would hope that with proper design, the model would be able to transfer some of the knowledge across environments. Such a model would learn the underlying features that identify each sleep posture, and tune them to a new environment with a small amount of additional labeled data from that environment. To address this issue, we design our model to be easily transferable. Specifically, given a set of source domains *i.e.,* a number of people and their sleep postures in the training set, and a target

domain *i.e.,* a new person in his own home, the model can use a small amount of labeled data (16 minutes to one night) from the new home to optimize its performance for this new environment.

Our transfer model has three components. First, given that the calibration data from the target can be very small, particularly when we have only 20 minutes of data, we create a virtual target composed of selected data points from the existing sources. These data points are selected based on their similarity with the calibration data (the similarity is based both on labels and features as described in Sec. 5.4). Second, given that we now have a virtual target with enough labeled data, we perform domain adaptation to adapt the source models to the target domain. Finally, since not every source is amenable to domain adaptation because the data corresponding to some sources can be significantly disjoint from the target data, both in terms of subject sleeping postures and model features, we perform model selection using the calibration data[1] to pick the sources that have high accuracy on the calibration data. We treat the collection of models for these sources as an ensemble and perform majority voting over them, to infer the target sleep posture.

Our model delivers high accuracy. Specifically, our basic sleep posture model using multipath, when trained and tested on the same person and home, achieves an accuracy of 94.1%. The transfer learning model to a new person and a new home has an accuracy of 86.7% with one night of labeled data, and 83.7% with a labeled dataset comprising 8 examples, where in each example, the person lies down in one of his typical sleep postures for a duration of 2 minutes.

To summarize, this chapter makes the following contributions:

1. We present BodyCompass, the first RF-based system that provides accurate sleep posture monitoring overnight in users' own homes. It achieves high accuracy without sacrificing privacy and sleep comfort.

2. BodyCompass can transfer its model to new homes and users with very little additional training data.

[1]Note that the virtual target includes only points from the source data that are similar to the calibration data, but does not include the calibration data.

3. We implement and evaluate BodyCompass extensively in real world settings using data from 26 homes with 26 different subjects and more than 200 nights of sleep.

5.2 Related Work

Past work on sleep posture monitoring can be divided into two major categories: 1) systems with on-body sensors, and 2) non-contact monitoring systems.

(a) On-body Solutions: On-body sensors can monitor sleep postures accurately [78, 169, 20]. For example, one may attach an accelerometer to the person's chest to monitor their sleep posture. Since gravity always points downwards, the accelerometer's orientation can be calculated by combining the acceleration along three different axes [169, 20]. However this method is cumbersome and uncomfortable since the accelerometer needs to be fixed on the user's body during their sleep.

Figure 5-2: Pressure sensitive bedsheet from [83].

(b) Non-Contact Solutions: Contactless systems are more comfortable for the user compared to on-body sensors. Work in this class falls in the following categories. First, vision-based systems [6, 86, 53] deploy RGB or infra-red cameras to record videos of the user's sleep, then process those videos using convolutional neural networks to predict sleep postures. However, cameras, particularly in people's bedrooms, are privacy-intrusive. Further, the accuracy of camera systems decreases significantly in dark settings and when people are covered with a blanket or comforter [6, 86, 53].

Second, on-bed sensors cover the mattress with an array of pressure sensors [61, 83, 166, 116, 110] or RFID Tags [84, 60]. These solutions are more privacy-preserving than

camera systems. However, on-bed sensors, shown in Fig. 5-2, change the feel of the bed and thus affect the sleep comfort of the subject. Further, most of these systems are evaluated in the lab, as opposed to overnight testing in people's own homes [61, 166, 116, 110, 84, 60].

Third, a few papers have proposed the use of RF signals for monitoring sleep posture [87, 11, 85]. The approach in those papers is intrinsically different from ours; they analyze the signal power as measured by the RSSI (received signal strength indicator) [11] or the power of the frequency sub-channels extracted from the CSI (channel state information) [87, 85]. As studied in Sec. 2.9.1, they all inherently suffer from interference. That is, they have no ability to separate changes in the signal that are due to the sleeping person from those due to other sources of motion (e.g. a fan, or a person moving in a neighboring room). Such extraneous motion brings randomness and will greatly hamper the robustness of the system in the wild. As a result, all previous papers are evaluated in a single lab environment with one or two subjects consciously performing specified postures.[2] In contrast, we study the spatial pattern of reflections –i.e., the multipath – and ignore the power by re-normalizing the power distribution of each path (see Sec. 5.4). Therefore, our system can provide accurate sleep posture monitoring overnight in users' homes and can be easily transferred to new environments.

We also note that past work has demonstrated the feasibility of inferring the human skeleton using only RF reflections [178]. It might seem that one could use such models to infer the skeleton of the person lying in bed and hence their sleep posture. However, due to RF specularity, such models rely on people walking and moving around to achieve good accuracy [178]. Specifically, as described in those papers, a snapshot of RF signal reflections does not capture the full body; Any snapshot captures only a few limbs or body parts that reflect signals directly towards the radio. Hence, their neural networks rely on people moving and walking to expose different body parts in each snapshot so that the network can combine those body parts to create the human skeleton. In contrast, when the person is asleep in bed, the person is mostly static and hence there is not enough motion to allow the neural network to fill in the gaps and combine body parts across different

[2]We note that while the authors of [85] test their vital sign algorithms outside the lab, the sleep posture is only tested in the lab and in one setting.

snapshots. To deal with this challenge, our system not only takes the direct reflections towards the radio, but also all the indirect reflections due to multipath. By taking all the multi-path reflections as input, our system estimates the sleep posture accurately even when the person remains static.

Finally, BodyCompass is related to a growing body of research that focuses on passive monitoring using radio signals. Researchers have demonstrated that by carefully analyzing RF reflections off the human body, they can monitor people's location [3, 164, 162], gait [63, 158, 175], breathing [160, 172, 105], heart rate [5, 159], falls [148, 161, 156], and sleep quality and stages [179, 62, 120, 147]. Our work builds on this foundation and leverages past work on inferring the breathing signal as a sub-component in our system [172].

5.3 BodyCompass

BodyCompass is the first RF-based system that provides accurate sleep posture monitoring in the wild, *i.e.*, with subjects sleeping in their own beds in their homes, and it generalizes to new subjects and homes with minimal additional effort. It can be used by healthy individuals interested in monitoring their sleep behavior, or can be provided either to patients to help them modify their sleep posture, or to doctors to assist them in understanding disease prognosis and patient health.

BodyCompass leverages measurements from an FMCW radio equipped with an antenna array [93]. Such radios are commonly used in passive health monitoring using RF signals [172, 120, 62]. They work by transmitting a low power radio signal, and observing its reflections from the surrounding environment. The use of an antenna array combined with FMCW enables the radio to resolve RF reflections from multiple points in space. Specifically, at each instance in time, the radio outputs an array of signal values from various voxels in space, which we refer to as an *RF-snapshot*.

BodyCompass takes a sequence of RF-snapshots from an FMCW radio across a whole night, and produces the sleep postures for the night. A sleep posture is described by an angle between two normal vectors, one of the bed surface and one of the user's anterior trunk surface, as shown in Fig. 5-3a. For example, 0° represents the user facing upwards and 90°

represents the user facing rightwards. Defining sleep posture in terms of angle allows us to differentiate between a slight tilt of the trunk to the right and someone sleeping on their right side. This enables a finer granularity definition of sleep postures that encompasses and expands beyond common posture classes (supine, left side, right side, prone). A fine granularity in posture estimation is important for applications that aim to detect changes in postures, such as tracking the progression of Parkinson's patients by monitoring the frequency of their change of sleep posture.

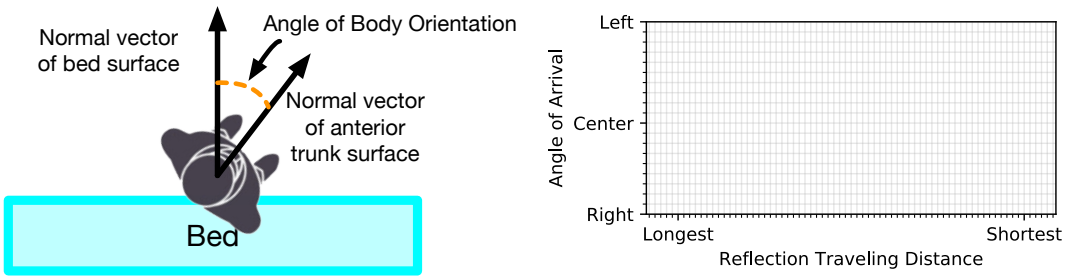
BodyCompass computes sleep postures using three components:

- A filtered multipath profile feature extractor to estimate the RF reflections that bounced off the person directly or indirectly.

- A source-specific neural network that utilizes the multipath profile features to estimate the sleep posture of a specific person in a specific home.

- A transfer learning model that adapts the source-specific models to estimate the sleep posture of a new person in a new home with minimal additional labeled data.

Below, we describe these components in detail.



(a) Angle of body orientation

(b) Coordinates of RF-snapshots

Figure 5-3: Illustration of body orientation and coordinates of RF-snapshots. In Fig. 5-3b, we represent all RF voxels in a Cartesian coordinate system. The rightmost pixels are closest to the device, and have the shortest distance. Therefore the direct-path reflections should be to the right of the indirect-path reflections since they travel the shortest path between the user and the device.

5.4 Filtered Multipath Feature Extractor

In this section, we describe how BodyCompass extracts filtered multipath features specific to a person from RF-snapshots produced by a FMCW antenna array.

An RF-snapshot consists, for each point of space (RF voxel), of the magnitude of the RF reflection from that point of space. An RF voxel is represented by two coordinates, its distance from the device, and the angle of that position relative to the normal from the device. Specifically, an RF voxel at coordinate (i, j) represents a small cube around the point at traveling distance d_j from the device, and an angle of arrival of α_i, as shown in Fig. 5-4. We divide the space into N angles, and M distances, and therefore each RF-snapshot is an $N \times M$ matrix. For better visualization, we plot voxels in a standard Cartesian coordinate system, as shown in Fig. 5-3b, instead of a polar coordinate system.

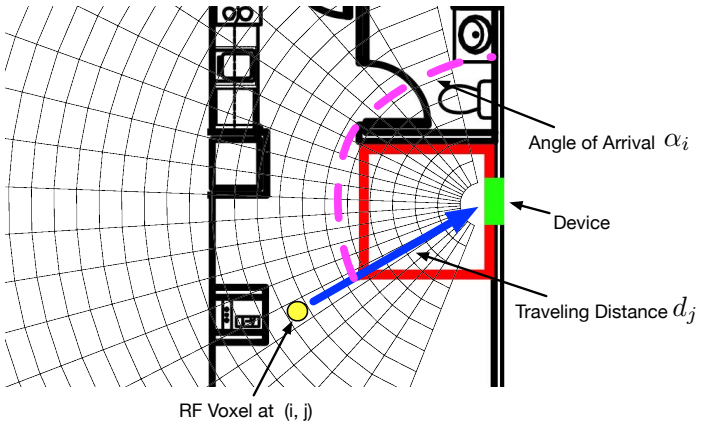


Figure 5-4: Illustrative example of an RF voxel. The green rectangle represents the radio location and the red rectangle represents the bed location. Our FMCW antenna array divides the space into small grids. The coordinates of the grid represent the distance from the radio and the angle of arrival.

5.4.1 Stable Sleep Periods

We first note that sleep postures are not independent over time, since people typically sleep in a posture for some period of time, followed by a movement, after which they settle into a different sleep posture, and so on. BodyCompass therefore first segments the night into a series of stable sleep periods. During each stable period, the orientation of the body is approximately constant, and BodyCompass extracts a single sleep posture from that period.

BodyCompass leverages prior work [172] to identify motion events from RF-snapshots, and defines the intervals between such motion events as stable sleep periods.

5.4.2 Filtered Multipath profile

Next, BodyCompass extracts the multipath profile for each stable sleep period, with the objective of learning the sleep posture from the multipath profile. Recall that the multipath profile captures the pattern of spatial reflections, *i.e.*, how the RF signals bounce around in space before they reach our radio. We represent the multipath profile of a particular stable period with the relative signal power along each path. Thus, the multipath profile of a particular stable period can be computed by taking the RF-snapshots corresponding to the stable sleep period and computing the variance in each voxel.

The multipath profile is affected by the sleep posture of the person, and is therefore informative about their orientation. For instance, when the person is supine, *i.e.*, lying flat on their back, a significant portion of the signal reflects towards the ceiling and bounces off other objects in the environment before reflecting back to the radio, and as a result the multipath profile shows significant dispersion. In contrast, when the person is sleeping on their side, the direct RF reflections will be significantly stronger than the indirect reflections, and the multipath profile will therefore show high concentration.

However, one cannot directly use the overall multipath profile in a stable sleep period to infer sleep posture. This is because such multipath profile contains reflections both from the environment and from the subject. While reflections from static reflectors (*e.g.*, walls, tables) can be removed,[3] reflections from moving objects cannot be easily disentangled, and their contributions can confound the system for two reasons. First, even within a single home, those contributions can change over time even when the sleep posture does not change, for instance, because of movements from a fan, people walking in the environment, or heating, ventilation, and air conditioning (HVAC) systems. Since these changes are not correlated with the sleep posture of the person, they will adversely affect the ability of BodyCompass to infer sleep posture. Furthermore, such reflections are highly specific to

[3]We remove static reflections by subtracting the average RF-snapshot for each stable period before computing the multipath profile [172].

each home, and incorporating them into the multipath profile will prevent BodyCompass from generalizing to new homes.

So, how does one filter out environmental contributions while still retaining the multipath contributions from the sleeping subject? Our idea is inspired by the following observation: when breathing, the chest and belly area of the human body move forward and backward. These motions will change the multipath contributions corresponding to the human body in a manner correlated with the breathing signal, while other environment related multipath contributions will not change in a manner correlated with the person's breathing. Fig. 5-5 shows an illustrative example of this point with a person sleeping facing the device, and a nearby fan.

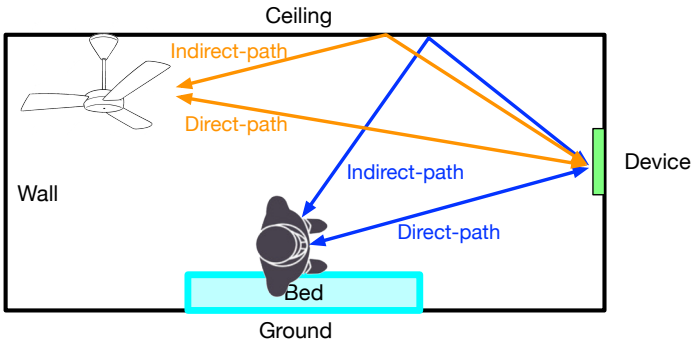


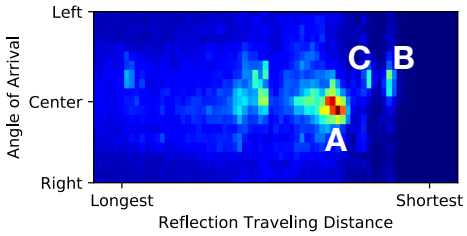
Figure 5-5: An illustrative example of signal reflections. In this case the multipath profile contains reflections from the subject (in blue) and a ceiling fan (in orange). Correct processing requires eliminating the fan reflections. Also the figure illustrates how the posture could affect the multipath. When the user is facing the device, the reflections along the direct path have the largest signal variations because the chest movements are most significant in that direction. In contrast, the signal variations along the indirect path are much smaller because the side of the body is not moving significantly.

Using breathing also allows BodyCompass to identify the orientation of the person, *i.e.*, whether the person is facing up/down when supine, and whether the person is facing towards/away from the device when on their side. This is because during breathing, only the front of the human body moves significantly, whereas the back does not, therefore breaking symmetry between the orientations, and changing the *filtered* multipath profiles in the two cases.

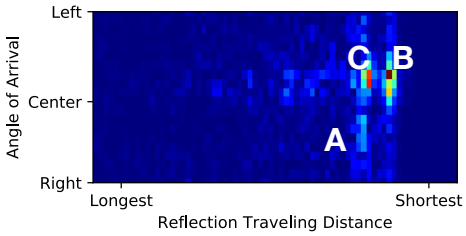
BodyCompass uses DeepBreath [172], to extract the breathing signal of the subject in bed from the RF-snapshots in the stable period. The breathing signal is a time series

that reflects the scaled chest displacement of the subject over time. Then, for each RF-voxel, BodyCompass correlates this extracted breathing signal with the time series of signal magnitudes for this RF-voxel obtained from the RF-snapshots. Specifically, for each voxel, BodyCompass computes the absolute value of the Pearson correlation coefficient between the person's breathing and the magnitude of the RF signal received from that voxel, as expressed in the sequence of RF-snapshots. This correlation provides a spatial filter that allows us to extract the voxels in the multipath profile whose signal is highly correlated with the person's breathing.

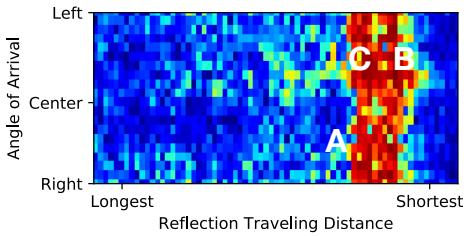
Next, BodyCompass multiplies the multipath profile with the above filter to extract the filtered multipath profile, which focuses on the signals that bounced directly or indirectly off the subject. The filtered multipath profile emphasizes pixels with a significant contribution from the subject's breathing while still retaining the relative power contributions from direct and indirect paths corresponding to that breathing.



(a) Original multipath profile



(b) Filtered multipath profile

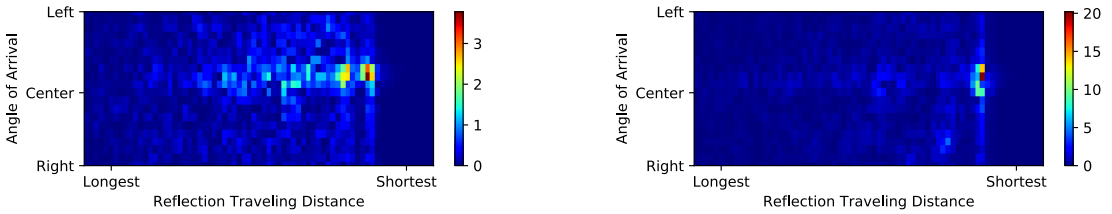


(c) Filter used to extract the multipath associated with the subject. The filter consists of the correlation coefficients with breathing

Figure 5-6: Visualization of one stable period. The value of each RF-voxel represents the corresponding attribute of the RF reflection. The visualization is color-coded, the redder the pixel, the higher the relative value of that pixel, and the bluer the pixel, the lower the relative value. Points A, B, C highlight three different kinds of reflections: environmental movement reflections, breathing reflections along direct-path, breathing reflections along indirect-path.

To help understand the process, we plot an example in Fig. 5-6. Specifically, Fig. 5-6a shows the relative power in each voxel in space in the original multipath profile. Fig. 5-6c shows the breathing filter, and Fig. 5-6b shows the filtered multipath profile. As we can see, pixel A is very bright in the original multipath profile (Fig. 5-6a), meaning that it has very a high reflection power relative to other pixels. However, since it has a very low correlation coefficient with breathing (Fig. 5-6c), it is removed in the final filtered multipath profile (Fig. 5-6b). In this case, pixel A was contributed by environmental movements from a different person walking in the environment. In contrast, while pixels B and C have lower power compared to A, they are emphasized in the filtered multipath profile since they exhibit strong correlation with the breathing signal. It is also worth noting that pixel B is actually the direct-path reflection, and pixel C is one of the indirect-path reflections.

Next we show two typical filtered multipath profiles of two different postures in Fig. 5-7. When the user is sleeping in a supine position, the filtered multipath profile shows more dispersion because the subject reflects a significant part of the signal towards the ceiling causing more indirect reflections. In contrast, when the user is facing the device most of the signal is directly reflected from the user to the device and hence the power in the filtered multipath profile is concentrated at the user's location, i.e. the direct path.



(a) The user facing up

(b) The user facing the device

Figure 5-7: Two typical examples of filtered multipath profiles of the user facing up (Fig. 5-7a) and facing towards the device (Fig. 5-7b). Compared to Fig. 5-7b, in Fig. 5-7a, we can see much higher power in further away pixels. This is because when the user is facing up, he deflects the signal towards the ceiling causing indirect reflections.

5.5 Source-Specific Sleep Posture Model

Having computed the filtered multipath profile for each stable sleep period corresponding to a source (*i.e.* a user in a specific home), BodyCompass then uses a neural network to

predict the sleep posture for that source during each stable sleep period.

Our model uses a multi-layer fully-connected neural network. We deliberately choose a fully-connected neural network instead of the commonly used convolutional neural network (CNN). CNNs are more suitable for natural images because one typically needs to compare each pixel with the pixels in its neighborhood. In contrast, to capture the multipath profile, one needs to compare pixels globally. (See Sec. 5.8.4 for an empirical comparison of the performance of a fully-connected network and a CNN on this task.)

For training the neural network, we ask subjects to wear accelerometers to collect the ground-truth angular orientation of the body. Detailed ground truth collection process is described in Sec. 5.7.2. Recall that we express the sleep position in terms of the angle specifying the trunk rotation with respect to the bed. BodyCompass averages the angular values that the accelerometer measures during a stable period to obtain the ground truth sleep posture of the subject in that period.

BodyCompass trains the neural network to predict the sleep angle associated with each filtered multipath profile. To train the network we need to compare the predicted angle with the ground truth angle. Directly comparing angles however leads to discontinuity since angles wrap around, i.e., $0°$ and $360°$ are the same angle, but simply computing their difference will yield a large loss.

Therefore, in order to ensure smoothness of the loss function, BodyCompass's model predicts complex numbers, and uses the phase of the complex number as the angle prediction. Specifically, we define *Circular Loss* as follows:

$$\mathcal{L}_c(\theta) = \mathbb{E}_{x,y\sim p(x,y)}\, arccos(\frac{Re(F(x,\theta)\cdot e^{-iy})}{|F(x,\theta)|}) \tag{5.1}$$

where x is the input feature vector (i.e., the filtered multipath profile of a stable segment), y is the ground-truth angle, $F(\cdot,\theta)$ is the model that maps a feature vector into a complex number, θ is the model parameters (the wights of the neural network), $\arccos$ denotes the arc cosine function, and $\mathbb{E}$ is the expectation.

The operand of $arccos$: $\frac{Re(F(x,\theta)\cdot e^{-iy})}{|F(x,\theta)|}$ can be interpreted as the cosine similarity between two vectors: one is our prediction $(Re(F(\cdot)), Im(F(\cdot))$ and the other one represent

the unit vector of the ground truth angle $(cos(y), sin(y))$. The similarity reaches its maximum when the predicted vector has the same angle as the ground truth (an $arccos$ of 0). And it reaches its minimum when these two vectors are diametrically opposed (an $arccos$ of π radians). This loss function solves the discontinuity problem since it computes the angle difference between our prediction and the ground truth in a differentiable way.

5.6 Transferring the Model to New Users

In the previous section, we explained how to train a model to predict a user's sleep posture accurately given abundant labeled data from the that user. However, data collection is a laborious and time-consuming task for both the user and the operator of the system. Ideally, we would like our system to perform well on new users with minimal effort.

Since the properties of RF signals (power, phase, and multipath) depend on the environment, transfer between different homes is a challenging task. In order to achieve satisfactory performance while reducing the burden on the user, we assume that only limited labeled data from a new user is available. We refer to such labeled examples (where an example is a filtered multipath profile and its correct sleep angle) as *Calibration Points*. As described in Sec. 5.8.2, the number of calibration points can be as few as 8 examples, each lasting for 2 minutes, for a total of only 16 minutes.

Given the scarcity of the calibration points, it is not practical to train a model entirely based on those points. Instead, we formulate the task as a semi-supervised domain adaptation problem: we have multiple source users, each with abundant labeled data, and a target user for which we have a few calibration points. We would like the system to achieve high accuracy on the target user given the above information.

5.6.1 Overview of the Transfer Model

Given a set of source domains i.e. a number of people and their sleep postures in the training set, and a target domain i.e. a new person in their own home, we design a model that learns from the training data of the source domains how to infer sleep posture in the target domain, with a small number of calibration points.

At a high level, our transfer model first preprocesses the training data to ensure that the probability distributions of the source domains look as close as possible to the target domain. Next, given that the amount of data from the target domain is not sufficient to train a model, we try to augment the data from the target by selecting data points from the source domains that look similar to the target data, both in terms of its feature map (*i.e.*, filtered multipath) and the corresponding posture. We use this augmented data to create a virtual target which is similar to the original target but has much more labeled data. Now we can adapt the model from each source domain to work well on the virtual target. The final prediction is then performed by majority voting over all of these adapted models.

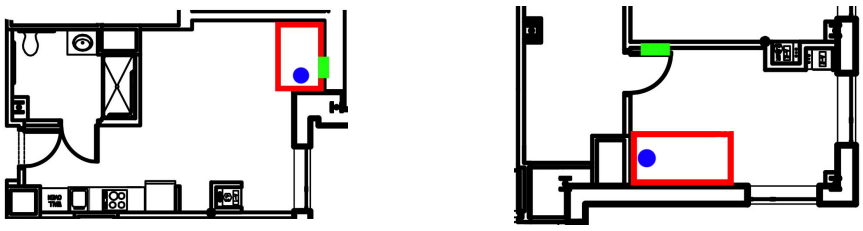
In the following sub-sections, we expand on this high-level description, providing the details of the three key components of our transfer model:

1. **Distribution Alignment**: We explicitly align the data distributions across users, whose differences are caused by different room layouts.

2. **Data Augmentation**: We generate augmented data by picking data points from source subjects that resemble the calibration points.

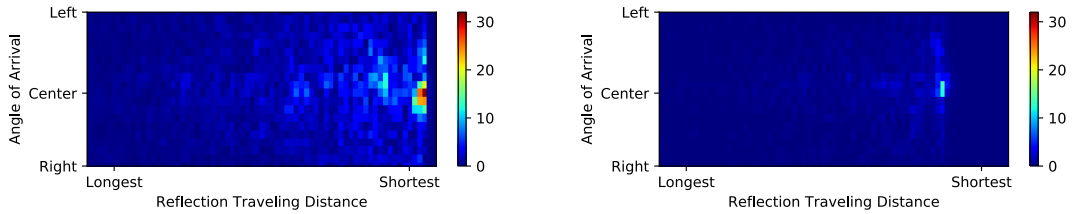
3. **Ensemble Learning**: We use Majority Voting to generate one final prediction that is robust and accurate.

5.6.2 Distribution Alignment



(a) User A's bedroom layout (b) User B's bedroom layout

Figure 5-8: Bedroom layouts of two users. Green rectangle shows the location of our device. Red rectangles shows the location of the bed and blue circle shows the position of the pillow.



(a) User A's profile when facing to the device (b) User B's profile when facing to the device

Figure 5-9: Examples showing how the bed position with respect to the radio affects the signal's strength and location. The figures show that due to differences in the position of the bed with respect to the radio, User A's direct path signal is much stronger and closer to the radio compared to User B's.

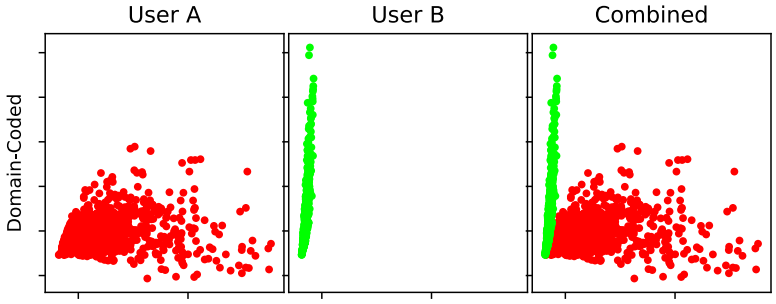
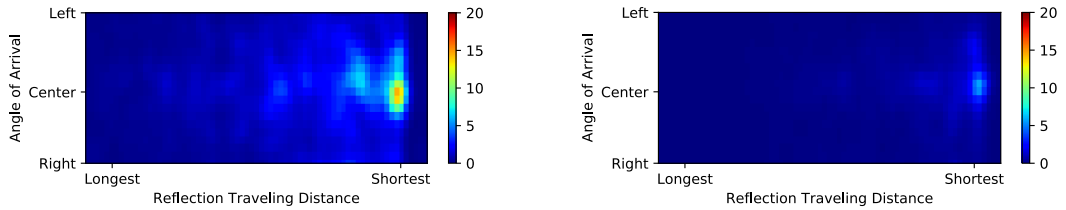


Figure 5-10: Visualization of data distribution of User A (red) and User B (green). Since the feature map (i.e., the filtered multipath profile) is high-dimensional, to visualize the data in a two-dimensional space we perform joint Principal Component Analysis (PCA) on all the feature vectors (all filtered multipath profiles) from both A and B using the same set of basis, and plot the data with respect to the two largest principle components. We plot the data of the source and target separately in the left two figures and combined in the right figure. As we can see, the distributions of two users are mismatched significantly.

Without any alignments, the model's generalization ability will be greatly hampered by the distribution shift between the source user and the target user. One major reason of this distribution shift is caused by different room layouts. For example, we have User A and User B with their floor plan visualized in Fig. 5-8. User A's bed is very close to the device (the device is right above the bed), and in comparison, User B's bed is far from the device. We show the multipath profiles of the two users in in Fig. 5-9 (a) and (b). As clear from these figures, the difference in the bed location with respect to the radio impacts the filtered multipath profile in two ways. First, the direct-path reflection of User B needs to travel a longer distance compared to User A. Thus pixels at the same location in the multipath profiles are not directly comparable. Second, the power of RF reflections decreases as the



(a) User A's profile after aligning bed location (b) User B's profile after aligning bed location

Figure 5-11: The multipath profiles in Fig. 5-9 after aligning bed locations. Now the direct path pixels of both Users A and B are at the same location.

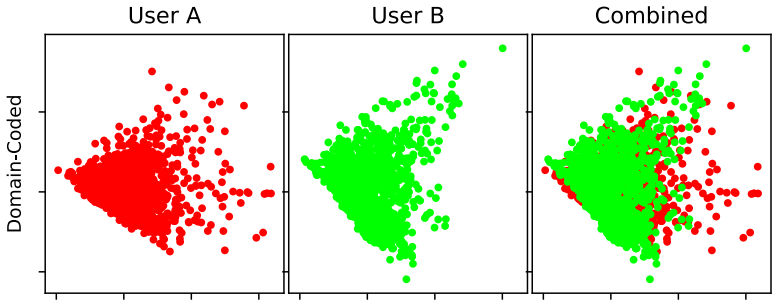


Figure 5-12: Visualization of data distribution of User A (red) and User B (green) after bed alignment and power normalization. Compared to Fig. 5-10, we can see that the two distributions are much better aligned.

traveling distance increases. Therefore the breathing powers of User B's pixels are much smaller than User A. As a result, and as shown in Fig. 5-10, the data distributions of User A and User B are significantly mismatched. Below, we explain the two methods we use to align the distributions.

(a) Aligning Bed Locations: We first align the relative location between the bed and radio across all users. While not all differences can be eliminated, this way, we ensure that all direct-path reflections have the same traveling distance. And since we cannot ask every user to move their bed, we can do this alignment virtually by reducing or increasing traveling distances of all the RF reflections in the multipath profile. This method brings us two advantages: 1) it keeps the original RF reflection pattern, and 2) we only need to know the bed location to perform this alignment.

Identifying the bed position can be done manually, however it is tedious and error-prone. Instead, we propose a robust and accurate way of measuring bed location for aligning. For each user, we do a pixel-wise summation for all of his filtered multipath profiles.

Since the direct signal path has the shortest traveling distance, usually it has the highest breathing power in the filtered multipath profiles. Therefore, the pixel with the highest sum will give us an accurate estimation of the bed location. We additionally apply a Gaussian filter with a sigma of 1 to erase small location mismatches.

Looking back to Fig. 5-8, there is another mismatch. In User A's case, the radio device is on the right-hand side of the user, and for User B, the device is on his left-hand side. This indicates that, when two users are both facing the device, they are actually facing different directions (to the right in Fig. 5-9a and to the left in Fig. 5-9b). Therefore we also align directions by flipping the angles. In all of the following discussions and results, $-90°$ (left) represents the direction facing the device (to the right for User A and to the left for User B), and $90°$ (right) represents the opposite direction.

Fig. 5-11 shows the filtered multipath profiles of Users A and B above after aligning the bed location. We can see that the direct path pixels of both A and B are now at the same location.

(b) Power Normalization: RF signals attenuate with distance. Hence, the power in a particular pixel in the filtered multipath profile depends on the path length, more so than the sleep posture, as shown in Fig. 5-11. This dependence can prevent model generalization to a new target user if the target's room layout differs from the source user. Thus we would like to eliminate this dependence. To deal with this issue, we normalize the power distribution in each pixel of the filtered multipath profile (i.e., for each data point, we subtract the mean of the distribution and divide by its standard deviation).

Data distributions for User A and B after both aligning the bed location and normalizing the power are plotted in Fig. 5-12, and they are aligned much better compared to Fig. 5-10.

5.6.3 Target Data Augmentation

Given the small number of calibration points, our information about the target user is limited. One solution is to perform data augmentation. In computer vision tasks, researchers have long been using augmentation techniques such as cropping, rotating, and horizontal flipping to help the model to capture data invariances. After those augmentations, images

are still valid images. However, this is not true for our multipath profile. For example, flipping means that the furthest pixel becomes the closest, and the longest indirect reflections becomes the direct-path reflection. Therefore such standard augmentation techniques for images will break the spatial structure of the multipath profile.

Instead, we use data points from the source users that are similar to the calibration points from the target user. Specifically, our augmentation process contains the following steps: First, we align all the data points from all the users (including the calibration points), as described in the previous section. Then given one calibration point (x_0, y_0), we first select all the points (x_i, y_i) that satisfy the condition that the angle difference between y_0 and y_i is smaller than a certain threshold (the default is 20 degrees). Then for those selected points, we further sort them based on the similarity of their multipath profiles to the calibration point as captured by the L2 distance: $\|x_i - x_0\|_2$. Finally, we pick the data points most similar to the calibration point (specifically we pick the 30 most similar source points to each calibration point). We refer to the set of augmented data points as the virtual target.

When adapting the neural network model from a particular source user to the target user, we combine the augmented data points with the labeled data from that source user to train the model and improve the model's performance on the target user. Note that we do not use the calibration points in training any of the adapted models. We hold the calibration points and use them to select the most effective adapted models as explained in the next section.

5.6.4 Majority Voting

So far, we adapted each of the source models to transfer its knowledge to the virtual target. However, some of the source users in our training set may be very different from the target user, and despite adaptation, their knowledge may not translate well to the target user. Thus we need a mechanism to detect models that are well adapted to the target and combine their predictions.

We define validation accuracy as the model's accuracy on the target's calibration points. This validation accuracy is an estimate of the adapted model's true accuracy on the target

data. Thus, we use this validation accuracy to evaluate the source's compatibility with the target. We filter out models that have bad validation accuracy (accuracy worse than 10% compared to the best model), and perform a majority vote among the models with good accuracy.

The majority vote is performed as follows: We create a histogram of the predictions where each angle degree has its own bin. Then we smooth this histogram with a Gaussian filter with a standard deviation of 20. Finally we pick the angle that has the highest value after smoothing as our final predicted angle. It is worth mentioning that, the smoothing is performed in a circular way, *i.e.*, the last bin and the first bin are connected.

5.7 Experiment Setup

5.7.1 Data Collection

All experiments with human subjects were approved by our IRB, and we have obtained informed consent from every subject. In total, we have collected 224 nights of data from 26 subjects (17 male subjects and 9 female subjects).[4] Subjects' bed sizes cover most common sizes, from twin-size (1m wide) to king-size (2m wide). Each subject sleeps alone in his/her own bedroom. For each subject, we install the radio device on the wall to the side of the bed. The distance between the bed and the radio ranges from 0 meters (right above the bed) to 4 meters (on the other side of the bedroom).

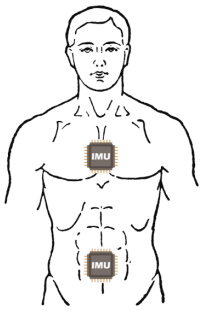


Figure 5-13: The placement of accelerometers on the subject's body. The accelerometers are used to collect the ground truth posture.

[4] After data cleaning described in Sec. 5.7.2

5.7.2 Ground Truth Collection

To collect the ground truth postures, for each night, we ask the subject to wear two accelerometers, one on the chest and one on the abdomen. Both accelerometers are fixed on the body using sport tapes to prevent sliding during sleep. In Fig. 5-13, we show the placement of accelerometers on the body.

We align both accelerometers so that the accelerometer's x's positive points towards the subject's head, and z's positive points opposite to the body. Then the angle of body orientation y can be calculated using the following equation [122]:

$$y = atan2(a_y, a_z)$$

where $atan2$ is the 2-argument arctangent function.

We use two accelerometers for mutual validation. There are many factors that can lead to poor data quality. Most commonly, sensors may fall off from the body during sleep, or the subject may fail to align the sensors correctly. By having two accelerometers, we can identify bad data points because readings from those two will no longer be equal. Such bad data points are then excluded from training and testing.

5.7.3 Radio Specification

We use a standard FMCW radio similar to the one used in past work [157, 172, 5, 176]. The radio sweeps the frequencies from $5.4\,GHz$ to $7.2\,GHz$ and transmits at sub-milliwatt power in accordance with the FCC regulations. The radio is equipped with two antenna arrays: a horizontal array and a vertical array. Each array is capable of dividing space into 20 (angle) by 71 (distance) pixels, with an angular resolution of $\sim 8^\circ$ and a distance resolution of $\sim 20cm$. In total we have $2 \times 20 \times 71$ different pixels.

5.7.4 Model Implementation

We use a fully-connected neural network with 4 layers. After each hidden layer, there is a layer of Batch Normalization [66], a layer of ReLU and a layer of Dropout. Fig. 5-14

illustrates the neural network architecture.

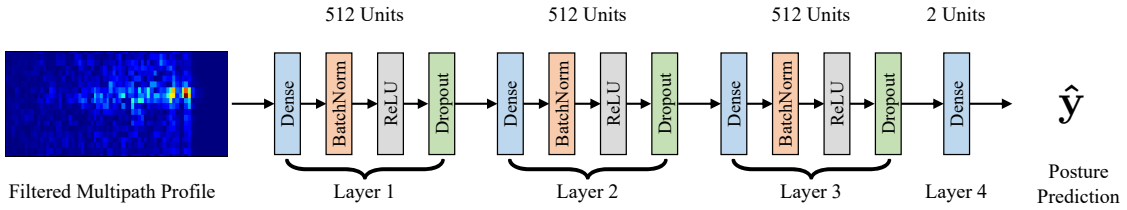


Figure 5-14: Architecture of the neural network.

5.8 Evaluation

In this section, we empirically evaluate the performance of BodyCompass.

5.8.1 Metrics

We define *Angle Error* as follows: For a single data point (stable segment) with a ground truth angle of y and a predicted angle of y', its *Angle Error* is the difference between y and y', as defined in Eq. 5.2. Angle error is always within the range from $0°$ to $180°$.

$$e(y, y') = |(y - y' + 180) \bmod 360 - 180| \tag{5.2}$$

We use following two metrics that are based on angle error to evaluate the performance of our system.

- **Average Angle Error** (Error): Average Angle Error is the weighted average of all the angle errors for all stable segments in the testing dataset, where the weight of each stable segment is set its time the duration.

- **Threshold Accuracy** (Accuracy): Since most past work computes accuracy with respect to a few key postures (supine, right side, left side, and prone), we similarly estimate accuracy as the percentage of time that the angle error between the prediction and the ground truth is smaller than $45°$. This gives us an intuitive understanding of the percentage of time we predict the direction of the user correctly.

5.8.2 Evaluation Setting

Depending on the amount of data available from the target subject, we present results under three different evaluation settings:

1. **1-Week**: If we collected enough data from the target subject, we can directly train on the target's data, without transfer learning. Under this setting, for each subject in our dataset, we report the result with leave-one-night-out cross-validation, i.e., using one night for testing and the remaining nights for training, and repeat this process until all the nights have been tested.

2. **1-Night**: In this setting we limit ourselves to only one night of labeled data from the target subject, and the rest of data from the target is unlabeled. One night is not enough for training. Therefore, we use this one night of data for our calibration points as discussed in Sec. 5.6.

3. **16-Minutes**: To further reduce the effort required from users, we present the results with only 8 labeled data points from the target subject. We assume that those 8 points cover most common positions of that user. This can be achieved by asking the user to emulate sleeping in his common sleep postures. In our evaluation, We select those calibration points from the existing sleep dataset by clustering the subject's sleep postures, and picking the center points for each cluster. If the resulting stable segment is longer than 2 minutes we use only a window of two minutes. Thus, collecting these 8 calibration points can be done in 16 minutes.

5.8.3 Evaluation of BodyCompass's Performance

To evaluate the effectiveness of BodyCompass, we compare its performance with three baselines: k-NN (k-Nearest Neighbors), Random Forest and XGBoost [21]. Note that all baselines and BodyCompass take filtered multipath profiles as input. Since the baselines do not have transferability capability like BodyCompass it is not clear how to train them when the available labeled data from the target subject is limited, i.e., in the 1-night and

Table 5.1: Evaluation results under three different settings with different methods (BodyCompass, k-NN, Random Forest (RF), XGBoost [21] (XGB)). Baseline methods are evaluated under two scenarios: All (A): trained with data from all the subjects; Target (T): trained with data from the target subject only. Note that BodyCompass significantly outperforms all three baselines under all settings.

	BodyCompass	k-NN (A)	k-NN (T)	RF (A)	RF (T)	XGB (A)	XGB (T)
Angle Error (1-week)	$15.3° \pm 4.4°$	NA	$31.3° \pm 9.7°$	NA	$33.8° \pm 13.0°$	NA	$33.8° \pm 13.3°$
Accuracy (1-week)	$94.1\% \pm 4.3\%$	NA	$77.7\% \pm 9.8\%$	NA	$75.4\% \pm 12.0\%$	NA	$75.5\% \pm 12.9\%$
Angle Error (1-night)	$25.6° \pm 6.7°$	$43.1° \pm 11.0°$	$40.6° \pm 11.0°$	$52.5° \pm 17.0°$	$45.4° \pm 15.1°$	$53.9° \pm 16.2°$	$49.2° \pm 13.1°$
Accuracy (1-night)	$86.7\% \pm 6.7\%$	$65.2\% \pm 10.5\%$	$67.8\% \pm 10.2\%$	$54.8\% \pm 14.5\%$	$62.2\% \pm 13.8\%$	$53.5\% \pm 14.2\%$	$59.9\% \pm 10.5\%$
Angle Error (16-min)	$28.3° \pm 8.7°$	$59.1° \pm 19.0°$	$60.6° \pm 19.0°$	$58.4° \pm 20.2°$	$55.0° \pm 18.9°$	$60.7° \pm 20.1°$	$65.1° \pm 13.1°$
Accuracy (16-min)	$83.7\% \pm 6.8\%$	$50.3\% \pm 14.6\%$	$46.4\% \pm 17.0\%$	$51.0\% \pm 14.9\%$	$52.2\% \pm 15.0\%$	$48.7\% \pm 15.8\%$	$42.8\% \pm 11.4\%$

16-minutes settings. Thus, for these settings, we evaluate the baselines' performance under two different training setups: 1. using data from all the available subjects; 2. using data only from the target subject.

Table 5.1 compares BodyCompass with the baselines for three different amounts of labeled data from the target subject: 1-week, 1-night, and 16-minutes. The table shows that BodyCompass significantly outperforms all three baselines under all settings. Specifically, BodyCompass and the baselines achieve their best performance when there is sufficient data from the target user. In such setting, BodyCompass's accuracy is 94%, whereas best accuracy across all baseline methods is only 77.7%.

The table also shows that when the amount of labeled data from the target subject is limited (e.g., in the 1-night setting and 16-minutes setting), the accuracies of all baselines are significantly reduced – 27.4% reduction for k-NN, 24.4% reduction for Random Forest, and 26.8% reduction for XGBoost. This is because of the natural variability in sleep postures. For example, even in the same body orientation, a slight change in arm or leg positions can cause a change in the pattern of RF reflections. All three baselines do not have the ability to handle such variability in the absence of a large amount of labeled data from the target. And since differences between different subjects are large, data from other subjects cannot help the baselines improve their robustness (e.g. under 1-night setting, data from other subjects is detrimental to the final performance). In contrast, BodyCompass aligns the distribution across different users, and performs data augmentation to battle this variability. As a result, BodyCompass can sustain high accuracy of 83.7% even with only

16 minutes of labeled data from the target user.

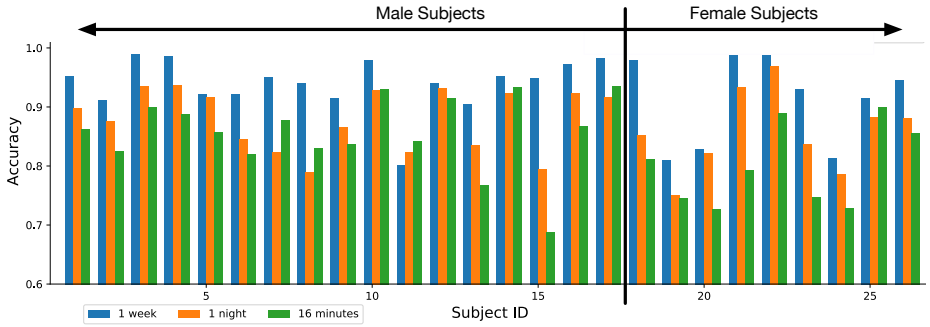


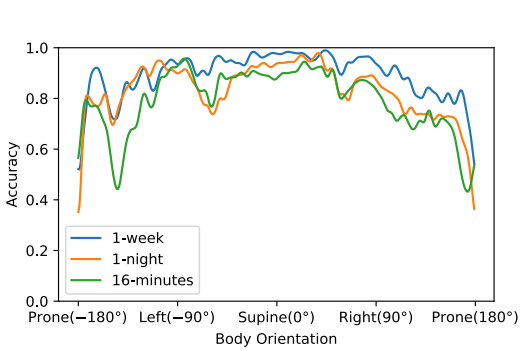
Figure 5-15: Accuracy for each of our subjects under three different test settings. Subjects are separated by their gender.

Next, we zoom in on BodyCompass and check the accuracy for each target user. In Fig. 5-15, we plot the average accuracy for each subject with our system under three different settings: 1-week, 1-night, 16-minutes. One can see that while transfer learning gives good accuracy across subjects, not all subjects have equal accuracy. Subjects who are more different from the source subjects will naturally have a lower accuracy when they have only 1-night of labeled data or 8 calibration points. We notice that, in our dataset, on average the accuracy on male subjects is higher than female subjects. Given the small number of subjects, it is not clear whether this accuracy gap is due to gender differences or is specific to the individuals in our dataset.

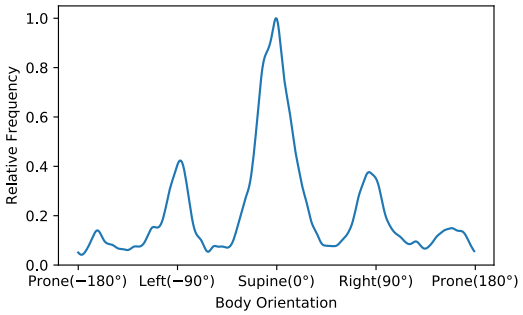
Another interesting aspect is that we see a slight increase of the accuracy of Subject #11 when moving from 1-week setting to a transfer setting like 1-night or 16-minutes. This is because while collecting data from Subject #11, the accelerometers often fell off during sleep. As a result, a significant part of each night from this subject had bad data with no accurate labels and was ignored. In this case, transfer learning can potentially have higher accuracy because it leverages labeled data from other users.

Finally, Fig. 5-16a presents the accuracy as a function of body orientation under all three settings. Our system has a high accuracy across all body orientations except for some corner cases where we lack enough training data (See the amount of labeled data for each angle in Fig. 5-16b). For example, at angle $-180°$, our system has low accuracy due to the fact that the amount of labeled data for this posture is $1/20$ amount of labeled data for the

supine case, *i.e.,* 0°.



(a) Accuracy w.r.t. Orientation



(b) Amount of labeled data as a function of orientation (normalized by the amount of data for the supine case)

Figure 5-16: Accuracy and amount of labeled data for each body orientation.

5.8.4 Evaluating the Components of BodyCompass

Table 5.2: Evaluation of the various components of BodyCompass. The table shows the accuracy under a 1-night setting for the whole system and for the system without a particular component.

	Angle Error	Threshold Accuracy
Full System	$25.6° \pm 6.7°$	$86.7\% \pm 6.7\%$
CNN instead of Fully-Connected Network	$29.5° \pm 10.1°$	$82.5\% \pm 10.0\%$
Direct Path instead of Multipath	$43.0° \pm 11.8°$	$67.7\% \pm 9.5\%$
L2 loss instead of Circular Loss	$33.1° \pm 12.4°$	$77.7\% \pm 13.2\%$
w/o Breathing Filtering	$30.0° \pm 10.2°$	$81.7\% \pm 9.3\%$
w/o Distribution Alignment	$33.4° \pm 11.9°$	$78.5\% \pm 10.8\%$
w/o Data Augmentation	$30.7° \pm 10.2°$	$81.5\% \pm 8.6\%$
w/o Majority Voting	$31.2° \pm 11.2°$	$80.5\% \pm 9.7\%$

We evaluate the contribution of each component in our system by evaluating the performance of the system without that component. Specifically, we evaluate the contribution of using multipath profiles, breathing-filtering multipath profiles, distribution alignment, data augmentation, majority voting, fully-connected neural network, and circular loss. Removing the contribution of the breathing-filtering multipath profiles, data alignment and

data augmentation components is straightforward, yet for the rest, we present the following substitutions:

1. **Substitution for the Multipath Profiles**: Instead of taking the multipath profile as input, we zoom in on the voxels from the bed area, and take only those voxels. This is equivalent to focusing on the direct path only, and ignoring any indirect paths that involve signals that bounced off the person and other objects in space.

2. **Substitution for the Fully-Connected Network**: We substitute the fully-connected neural network with a convolutional neural network (CNN) and evaluate the resulting performance. The CNN model follows the AlexNet model [74].

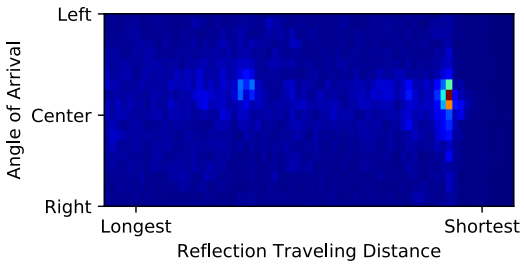
3. **Substitution for the Circular Loss**: We can directly regress the angle and use the standard L2 loss.

4. **Substitution for Majority Voting**: Instead of training on each source subject and performing majority voting, we can combine labeled data from all source subjects and train only one model.

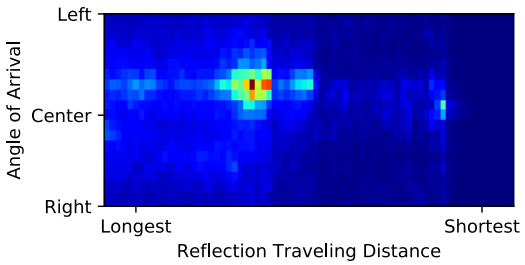
Table 5.2 provides the evaluation results for BodyCompass's components. All experiments are conducted under 1-night setting. The first row shows BodyCompass's accuracy 1-night setting when all components are active. Comparing the first row with the other rows in the table shows that each of BodyCompass's components offers a considerable improvement to the overall performance, and the removal of any component results in reduced accuracy.

5.8.5 Sensitivity Study

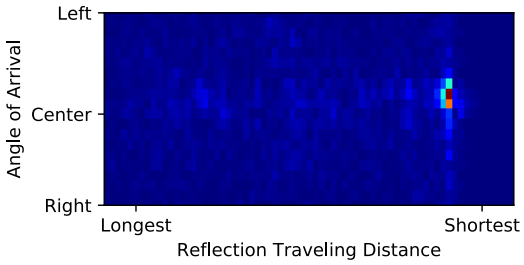
In this section, we test our system's robustness to various factors such as the presence of other people in the environment and their movements, whether the subject has shallow or deep breathing, and the exact location of the radio. We note that when collecting data in the wild, *i.e.*, in people's own homes during their natural overnight sleep, we have no control over the above factors. In fact, in all of the experiments reported in the previous sections, we leave the radio in the home of the subject for about a week to collect the data. We have no



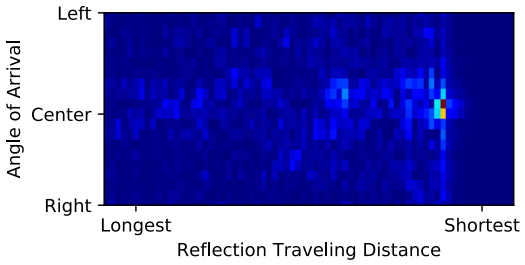
(a) Multipath Profile w/o Neighbor



(b) Multipath Profile w/ Neighbor



(c) Filtered Multipath Profile w/o Neighbor



(d) Filtered Multipath Profile w/ Neighbor

Figure 5-17: Robustness to moving neighbors. The figures in the top row show the multipath profile without and with a moving neighbor. Figures in the bottom row *filtered* multipath profile without and with a moving neighbor. The figures show that filtering the multipath profile eliminates extraneous movements from other people, hence boosts the robustness of the system.

control over when the subject goes to bed, where in the bed they sleep, whether their sleep location changes from one night to the next, how they breathe, and who else is at home and how they move while the subject is asleep. Thus, we cannot run sensitivity tests in the wild. We run these tests in a controlled environment where the subject is lying in bed but they are not asleep. In each case, the subject lies in bed at a particular body orientation, while we change the parameter we want to study, and measure BodyCompass's accuracy. The subject then changes his body orientation and we repeat the measurements while varying the parameter of interest.

Sensitivity to Movements by Other People

We evaluate our system's performance when there is a neighbor moving in an adjacent room. The subject sleeps approximately 3 meters away from the radio, whereas the neighbor is about 7 meters away from the radio. We ask the subject to lay down for 2 minutes in each of the following postures: supine, facing left, facing right, and prone. We train

the system with examples in which the subject was alone without anyone moving in the neighboring room. We use 6 examples for each sleep posture for training. During testing, we bring a second person to the adjacent room and ask them to move at will, while the subject is lying in bed. We collect 3 examples of each sleep posture for testing.

Table 5.3: Performance w/ neighbor movements.

	Angle Error	Accuracy
Full System	$12.7°$	100%
w/o Filtering	$53.6°$	58.3%

Fig. 5-17 shows that our filtering of the multipath profile makes the system robust to extraneous movements such as the presence of a neighbor. The figure plots the multipath profile as well as the filtered multipath profile. As the figure shows, the neighbor's movements have a significant impact on the unfiltered multipath profile (Fig. 5-17b), yet its impact is removed after filtering the multipath profile using the method in Sec 5.4.2 (Fig. 5-17d).

Table. 5.3 presents BodyCompass's accuracy results averaged across the test examples. It shows that our system is robust against extraneous movements by other people.

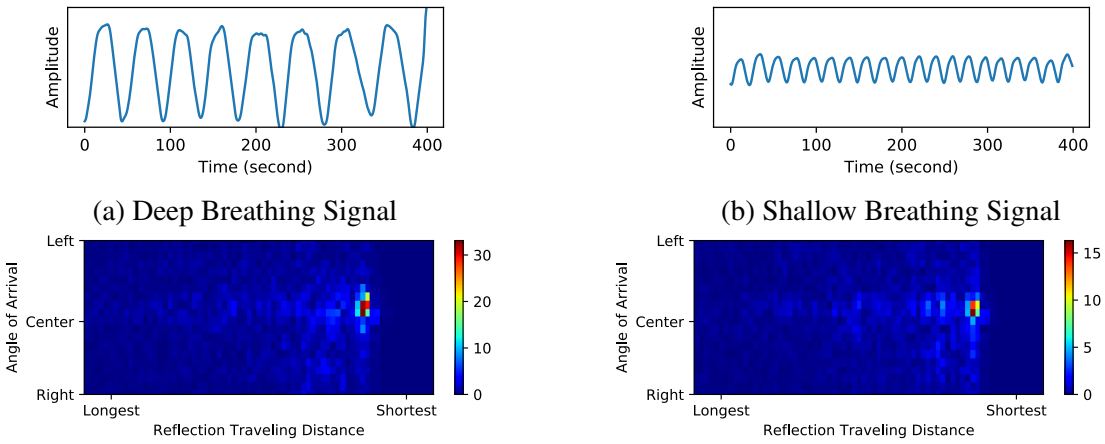
Sensitivity to Breathing Strength

Table 5.4: Performance when subjects breathe at different strengths.

Strength	Angle Error	Accuracy
Deep	$13.6° \pm 13.5°$	$97.2\% \pm 4.8\%$
Shallow	$8.6° \pm 4.8°$	$100\% \pm 0\%$

In this section, we investigate BodyCompass's robustness to variations in breathing strength, *i.e.,* to people having shallow vs. deep breathing. As in the previous section, we ask the subject to lie in bed in various sleep postures and for each posture, vary their breathing depth and rate. We repeat the test for each sleep posture, and with 3 different subjects. During testing, we ask the subject to perform 2 groups of experiments. Each

group contains 4 postures (supine, facing left, facing right, and prone) and each posture is repeated 3 times. For the first group, we ask the subject to breathe deeply and slowly, and for the second group, we ask the subject to breathe shallowly and quickly. In Fig. 5-18, we show the breathing signals as well as their corresponding multipath profiles. Table. 5.4 reports the average accuracy for different breathing strength. It shows that BodyCompass has high accuracy for both shallow and deep breathing.



(a) Deep Breathing Signal

(b) Shallow Breathing Signal

(c) Multipath Profile w/ Deep Breathing

(d) Multipath Profile w/ Shallow Breathing

Figure 5-18: Breathing signals and their corresponding multipath profiles. Although the amplitude and frequency of the two breathing signals are quite different, their multipath profiles have similar patterns.

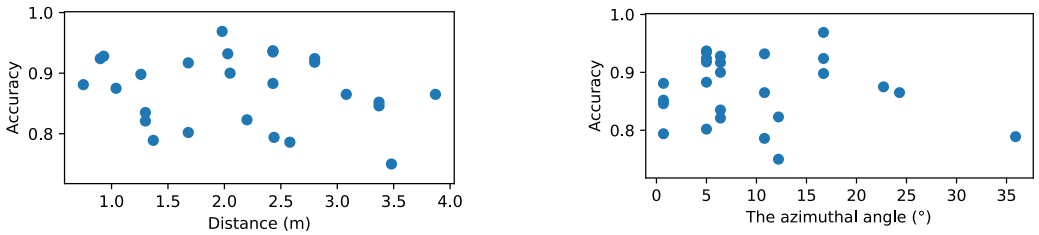
We also note that the accuracy results in these sensitivity tests are higher than the numbers presented in Table. 5.1 for testing in the wild. This is because when the subject is awake, the subject is able to accurately control his posture; In contrast, when the subject is asleep, the limbs can take various positions; Also the subjects may use pillows to support their bodies, and their use of pillows may change across days. Thus, overall there is much more variability in the wild.

Sensitivity to Device Location

Because the radio has directional antennas and the breathing signal is relatively weak, the device should face the bed to get a good SNR. However, we do not require the device to be exactly facing the chest of the person, nor do we require the device to be at a specific distance from the person.

In this section, we show that BodyCompass is fairly robust to variability in device distance and deviation from facing the chest of the person. In contrast to the previous two sensitivity tests, our in-the-wild deployments exhibit significant diversity in terms of device distance and azimuthal angle with respect to the person's chest. Specifically, in terms of distance between the chest and the device, our deployments cover a range from $0.5m$ to $4m$. In terms of the angle between the device and the person's chest, our deployments cover a range up to plus/minus $35°$, where a zero degree means that the device is facing the chest of the person.

Fig. 5-19 plots the accuracy of BodyCompass as a function of the distance to the person, and the angle to person's chest. The results in the figure show that BodyCompass works reliably for different location settings.



(a) Distance between radio and subject

(b) Azimuthal angle from device to subject's chest

Figure 5-19: Scatter plots of accuracy w.r.t. difference location settings. As two plots show, our system can accommodate a wide range of location settings.

5.8.6 Example Application: Monitoring the Frequency of Posture Changes

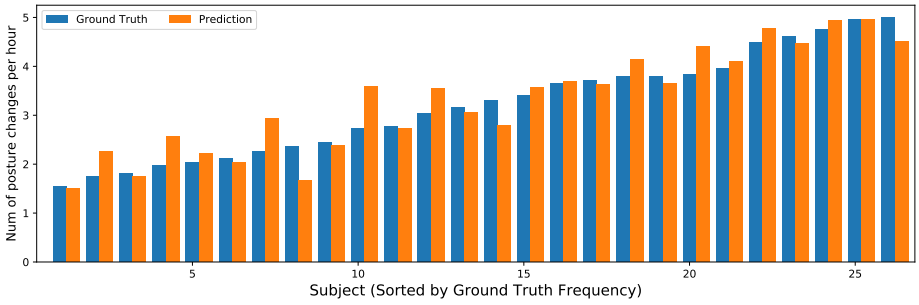


Figure 5-20: Ground truth and predictions of posture shift frequency for each subject.

The frequency of posture shift (moving from one posture to another) during sleep is an

important sleep-related metric. The literature shows that posture shift frequency is correlated with aging [132, 33] and sleep qualities [33]. Further for patients with Parkinson's disease, less frequent nocturnal turnovers reflects a deterioration in the disease condition [99].

Note that not every motion is a posture change. For example, when the user just changes his arm position, his body angle remains the same. To thoroughly evaluate our system's performance, we adopt the definition of posture change in [132]: to be quantified as a posture change, a body angle change of at least $30°$ is required.

Fig. 5-20 plots the frequency of posture shifts for all 26 subjects, ordered in ascending order. The figure compares the ground truth and our model with 1-night of training data from the target user. The figure shows that our model is capable of tracking the frequency of posture shifts accurately, with an average relative error of only 10.3%. These results indicate that our model can be used to track changes in posture shift in Parkinson's patients.

5.8.7 Failure Case Analysis

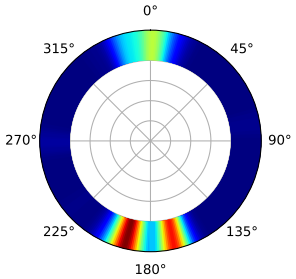


Figure 5-21: Angle histogram of Subject #15. Histogram is color-coded into the ring, as a more reddish color represent more occurrence.

Looking back at Fig. 5-15, we can see that even when the amount of labeled data is scarce, our system is still able to deliver satisfactory accuracies for most subjects. However, there are few exceptions where we can see a significant reduction in performance when the amount of labeled data is reduced. For example, for Subject #15, we have the largest reduction in performance, from 95.0% to 68.7%. To understand the reason, we plot his angle histogram in Fig. 5-21. We can see that most of his time is spent in prone position (sleep on stomach), and he never sleeps facing left or right. Recall in Sec. 5.6.2, we explicitly align the distribution to be the same. However this subject's intrinsic distribution is not

similar to any other subjects. Therefore the alignment cannot fully succeed, which causes bad transfer performance. We expect to see an increase of our system's performance on this subject if we have more subjects with similar sleep postures.

5.9 Conclusion

We present BodyCompass, a wireless system that provides accurate sleep posture monitoring in the wild. By explicitly extracting RF reflections from the user and designing appropriate machine learning algorithms, our system can accurately capture the user's posture and is able to transfer its knowledge to a new home with minimal additional data. A user study in 26 different homes with 26 subjects and more than 200 nights shows that BodyCompass is highly accurate, with an accuracy of 94% using 1 week of data from the user, and 83.7% using only 16 minutes of data. We believe that this work can serve as a practical sleep posture monitoring system, enabling easy adoption and helping doctors and patients address this unmet need.

Chapter 6

Conclusion and Discussion

In this thesis, we explore alternative solutions that allow for in-home sleep studies that are comfortable and affordable, and at the same time, deliver clinically meaningful measurements. By only relying on wireless reflections, our system do not require the user to wear any on-body sensors and he can sleep comfortably in his own bed. We developed the following systems as parts of this thesis.

For sleep parameters, we introduce **EZ-Sleep**. It involves novel algorithms that models RF-based locations as noisy measurements, and pinpoint bed entry/exit events with a Hidden Markov Model. It then applies a convolutional neural network on the wireless signals reflected from the bed to detect whether the user is asleep.

For sleep stages, we present **RF-Sleep**, which achieves a significant improvement over the state-of-the-art methods. It adapts a convolutional neural network (CNN) to extract stage-specific features from RF spectrograms, and couples them with a recurrent neural network (RNN) to capture the temporal dynamics of sleep stages. It also proposes a novel conditional adversary model, which enables the encoder at equilibrium to discard all extraneous information specific to the source domains, while retaining the relevant information for the predictive task.

For breathing monitoring, we propose **DeepBreath**, the first system that recovers the detailed respiration signals of multiple people sharing a bed or sitting with zero separation. We present a mathematical reduction that allows us to formulate the problem as blind source separation and solve it using independent component analysis.

And finally for sleep posture, we have **BodyCompass**, the first wireless system that provides accurate sleep posture monitoring in the wild. By explicitly extracting RF reflections from the user and designing appropriate machine learning algorithms, we can accurately capture the user's posture and our system is able to transfer its knowledge to a new home with minimal additional data.

Collectively, the four systems can turn the user's own bedroom into a sleep lab, where detailed sleep studies can be conducted every night with zero overhead for the subject. Our approach provides medical doctors tools to diagnose and track sleep-related problems and treatments in the patient's home.

6.1 Lessons Learned

The experience of designing and deploying RF-based deep-learning sensing systems with real users in their normal living spaces has led to the following lessons.

Labeling RF signals: Typically, deep learning requires labeled data. Yet, unlike images, audio, and text, radio signals cannot be labeled by human labelers. Further, there are no existing RF datasets that one can use for training and testing a model. Hence, RF-based sensing applications require the designer to collect a new dataset and find a way to label it. This is perhaps a more general problem that applies to data that cannot be easily interpreted by humans, like RF signals, accelerometer time series, etc. One way to label such data is to find an alternative modality that is easy to label by humans, and synchronize the radio signal with this modality. There is no single modality that works for all problems. For example, in the sleep staging application we used EEG signals which are the gold standard for scoring sleep stages; in DeepBreath, we had the user wear a breathing belt and used a matching algorithm to synchronize the breathing with the radio signal. In BodyCompass, we asked the person to wear two accelerometers on their chest and belly, which we synchronized with the radio and used to detect the ground truth body posture at different times.

Understanding RF signals: It might seem at first that one can simply feed the radio signal to a neural network and train the desired model. Unfortunately this approach does not

work. Similarly to how machine learning scientists use their intuition about images and text to design appropriate models for such modalities, a designer of an RF-based learning model needs to deeply understand the radio signal and its relationship to the task of interest. For example, knowing radio propagation and how the radio signal is modulated by all objects in the environment made us actively work to eliminate the extraneous information in the signal. This was done at multiple levels. First, a significant amount of irrelevant information was eliminated by understanding how radios signals can zoom spatially on a target. Then we eliminated the residual irrelevant information through our model design, i.e., using a conditional adversarial architecture.

Working in the Wild: Many academic papers demonstrate a proof of concept in the lab and assume that the system will generalize to work outside the lab. When we started our work on RF-sensing of breathing and sleep, we naturally conducted the first few experiments in the lab environment. However as we moved our experiments to users' homes, we discovered that even simple things that we take for granted in a lab setting become a hard problem. For example, in a sleep lab, it is clear when the user goes to bed. In contrast, at home we need to design an algorithm to discover when the user goes to bed. We also need to decide whether the user went to bed for sleep or simply to sit and work in bed. There are also many other uncertainties about the environment and the various people in it and the unknown interferers that may affect the radio signal. The robustness that is needed for working outside the lab is very different and cannot be appreciated without trying to run sensing systems in the wild with real users, as they go about their normal life.

6.2 Limitations

During the design process of our systems, we make a few assumptions about the data. Those assumptions hold for most scenarios, including data we have collected during the developments of the systems. However we do notice that those assumptions may fail under rare conditions. Some of the assumptions used by our current design are listed below.

Reliability of RF-based localization system: EZ-Sleep leverages a state-of-art RF-based

localization system, which should be able to localize the user to within 1 meter in 90% of cases. However, the error of the RF localization system will greatly increase if there are constant environmental motions near the bed, such as a rotating fan above the bed. Unless actively addressed, such motions could confuse the system and let it believe that there is always a person in the bed.

Sleep cycles: Our sleep stages progress in cycles and will enter different sleep stages in order. Our model leverages the temporal information to generate accurate estimation. However, patients with particular diseases may not follow the routine sleep cycles. For those patients, our system may not be able to achieve optimal performance.

Up to 2 persons in the bed: The identity matching algorithm assumes there are up to 2 people in the same bed. However, some couples like to sleep with their pets, and pets breath as well. Though the breathing separation module can successfully extract each individual breathing signals, the identity matching algorithm would fail because there are intrinsically more than 2 people in the same bed.

6.3 Future Directions

Beyond sleep and breathing: During a sleep study, the doctor has access to a wide range of vital signals, including electroencephalogram, electrooculography, electromyography, blood oxygen level, etc. Those vital signals are important for certain sleep diseases. Future work could consider estimating these signals from the radio waves that reflect off people's bodies. Some of these signals are more directly related to motion and can be easily extracted from radio signals with a proper choice of wavelength. Others, e.g., blood oxygen, are not directly related to motion but may be indirectly inferred from breathing which can be extracted from radio signals. The details and achievable performance are interesting topics for future work.

Chronic Disease Monitoring: Many chronic diseases are directly related to sleep. For example, research shows that depression patients usually have much longer REM. Therefore

the amount of REM sleep within one night could act as a measurement of depression severity. Some chronic diseases like Parkinson's Disease can also cause sleep fragmentation, which is reflected by a high amount of WASO. Since our systems are designed for long-term monitoring, we provide doctors tools to better understand the progression of those diseases, and potentially study the impact of medications on sleep and nocturnal breathing.

Bibliography

[1] Ossama Abdel-Hamid, Abdel-rahman Mohamed, Hui Jiang, Li Deng, Gerald Penn, and Dong Yu. Convolutional neural networks for speech recognition. *IEEE/ACM Transactions on audio, speech, and language processing*, 2014.

[2] Fadel Adib, Chen-Yu Hsu, Hongzi Mao, Dina Katabi, and Frédo Durand. Capturing the human figure through a wall. *ACM Transactions on Graphics (TOG)*, 2015.

[3] Fadel Adib, Zachary Kabelac, and Dina Katabi. Multi-person localization via {RF} body reflections. In *12th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 15)*, pages 279–292, 2015.

[4] Fadel Adib, Zachary Kabelac, Dina Katabi, and Robert C Miller. 3d tracking via body radio reflections. In *11th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 14)*, pages 317–329, 2014.

[5] Fadel Adib, Hongzi Mao, Zachary Kabelac, Dina Katabi, and Robert C Miller. Smart homes that monitor breathing and heart rate. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. ACM, 2015.

[6] Sina Akbarian, Ghazaleh Delfi, Kaiyin Zhu, Azadeh Yadollahi, and Babak Taati. Automated non-contact detection of head and body positions during sleep. *IEEE Access*, 2019.

[7] Laiali Almazaydeh, Khaled Elleithy, Miad Faezipour, and Ahmad Abushakra. Apnea detection based on respiratory signal classification. *Procedia Computer Science*, 2013.

[8] S Ancoli-Israel, R Cole, C Alessi, M Chambers, W Moorcroft, and C Pollak. The role of actigraphy in the study of sleep and circadian rhythms. american academy of sleep medicine review paper. *Sleep*, 2003.

[9] Martin Arjovsky and Léon Bottou. Towards principled methods for training generative adversarial networks. *arXiv preprint arXiv:1701.04862*, 2017.

[10] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International Conference on Machine Learning*. PMLR, 2017.

[11] Paolo Barsocchi. Position recognition to support bedsores prevention. *IEEE journal of biomedical and health informatics*, 2012.

[12] beddit. Beddit. http://www.beddit.com/, 2017.

[13] Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine learning*, 2010.

[14] Richard B Berry, Rohit Budhiraja, Daniel J Gottlieb, David Gozal, Conrad Iber, et al. Rules for scoring respiratory events in sleep: update of the 2007 aasm manual for the scoring of sleep and associated events: deliberations of the sleep apnea definitions task force of the american academy of sleep medicine. *JCSM*, 2012.

[15] Daniel J Buysse, Yu Cheng, Anne Germain, Douglas E Moul, Peter L Franzen, Mary Fletcher, and Timothy H Monk. Night-to-night sleep variability in older adults with and without chronic insomnia. *Sleep medicine*, 11(1):56–64, 2010.

[16] Mary A Carskadon, William C Dement, et al. Normal human sleep: an overview. *Principles and practice of sleep medicine*, 2005.

[17] Mary A Carskadon, William C Dement, MM Mitler, Christian Guilleminault, Vincent P Zarcone, and Rene Spiegel. Self-reports versus sleep laboratory findings in 122 drug-free subjects with complaints of chronic insomnia. *Am J Psychiatry*, 133(12):1382–1388, 1976.

[18] Mary A Carskadon and Allan Rechtschaffen. Monitoring and staging human sleep. *Principles and practice of sleep medicine*, 2000.

[19] Ashish Chaddha. Slow breathing and cardiovascular disease. *International journal of yoga*, 2015.

[20] Kang-Ming Chang and Shin-Hong Liu. Wireless portable electrocardiogram and a tri-axis accelerometer implementation and application on sleep activity monitoring. *Telemedicine and e-Health*, 17(3):177–184, 2011.

[21] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794, 2016.

[22] Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. *NIPS*, 2016.

[23] Zhenyu Chen, Mu Lin, Fanglin Chen, Nicholas D Lane, Giuseppe Cardone, Rui Wang, Tianxing Li, Yiqiang Chen, Tanzeem Choudhury, and Andrew T Campbell. Unobtrusive sleep monitoring using smartphones. In *Pervasive Computing Technologies for Healthcare (PervasiveHealth), 2013 7th International Conference on*, pages 145–152. IEEE, 2013.

[24] Eun Kyoung Choe, Bongshin Lee, Matthew Kay, Wanda Pratt, and Julie A Kientz. Sleeptight: low-burden, self-monitoring technology for capturing and reflecting on sleep behaviors. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 121–132. ACM, 2015.

[25] Seungjin Choi, Andrzej Cichocki, Hyung-Min Park, and Soo-Young Lee. Blind source separation and independent component analysis: A review. *Neural Information Processing-Letters and Reviews*, 2005.

[26] Sergei Churkin and Lesya Anishchenko. Millimeter-wave radar for vital signs monitoring. In *COMCAS*. IEEE, 2015.

[27] Jacob Cohen. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 1960.

[28] Roger J Cole, Daniel F Kripke, William Gruen, Daniel J Mullaney, and J Christian Gillin. Automatic sleep/wake identification from wrist activity. *Sleep*, 15(5):461–469, 1992.

[29] Robert Bosch Tool Corporation. Bosch laser measuring?, February 2017.

[30] Koby Crammer, Michael Kearns, and Jennifer Wortman. Learning from multiple sources. *JMLR*, 2008.

[31] Thiago LT da Silveira, Alice J Kozakevicius, and Cesar R Rodrigues. Single-channel eeg sleep stage classification based on a streamlined set of statistical features in wavelet domain. *Medical & biological engineering & computing*, 2016.

[32] Murilo Dattilo, Hanna Karen Moreira Antunes, Alessandra Medeiros, M Mônico Neto, Heitor Soares de Souza, Sergio Tufik, and MT De Mello. Sleep and muscle recovery: endocrinological and molecular basis for a new and promising hypothesis. *Medical hypotheses*, 77(2):220–222, 2011.

[33] Joseph De Koninck, Dominique Lorrain, and Pierre Gagnon. Sleep positions and position shifts in five age groups: an ontogenetic picture. *Sleep*, 15(2):143–149, 1992.

[34] Luciane de Souza, Ana Amélia Benedito-Silva, Maria Laura Nogueira Pires, Dalva Poyares, Sergio Tufik, and Helena Maria Calil. Further validation of actigraphy for sleep studies. *Sleep*, 26(1):81–85, 2003.

[35] Arnaud Delorme and Scott Makeig. Eeglab: an open source toolbox for analysis of single-trial eeg dynamics including independent component analysis. *Journal of neuroscience methods*, 2004.

[36] Gustavo Desouzart, Rui Matos, Filipe Melo, and Ernesto Filgueiras. Effects of sleeping position on back pain in physically active seniors: A controlled pilot study. *Work*, 53(2):235–240, 2016.

[37] Jeff Donahue, Philipp Krähenbühl, and Trevor Darrell. Adversarial feature learning. *ICLR*, 2017.

[38] Vincent Dumoulin, Ishmael Belghazi, Ben Poole, Alex Lamb, Martin Arjovsky, Olivier Mastropietro, and Aaron Courville. Adversarially learned inference. *ICLR*, 2017.

[39] Terence Dwyer, A-LB Ponsonby, Neville M Newman, and Laura E Gibbons. Prospective cohort study of prone sleeping position and sudden infant death syndrome. *The Lancet*, 337(8752):1244–1247, 1991.

[40] Farideh Ebrahimi, Mohammad Mikaeili, Edson Estrada, and Homer Nazeran. Automatic sleep stage classification based on eeg signals by using neural networks and wavelet packet coefficients. *IEEE EMBC*, 2008.

[41] Basura Fernando, Amaury Habrard, Marc Sebban, and Tinne Tuytelaars. Unsupervised visual domain adaptation using subspace alignment. *ICCV*, 2013.

[42] Fitbit, 2017.

[43] WW Flemons, D Buysse, S Redline, A Oack, K Strohl, J Wheatley, T Young, N Douglas, P Levy, W McNicolas, et al. Sleep-related breathing disorders in adults: recommendations for syndrome definition and measurement techniques in clinical research. *Sleep*, 1999.

[44] G David Forney. The viterbi algorithm. *Proceedings of the IEEE*, 61(3):268–278, 1973.

[45] Luay Fraiwan, Khaldon Lweesy, Natheer Khasawneh, Heinrich Wenz, and Hartmut Dickhaus. Automated sleep stage identification system based on time–frequency analysis of a single eeg channel and random forest classifier. *Computer methods and programs in biomedicine*, 2012.

[46] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. *ICML*, 2015.

[47] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *Journal of Machine Learning Research*, 2016.

[48] William F Ganong and William Ganong. *Review of medical physiology*. Appleton & Lange Norwalk, CT, 1995.

[49] C Gonzaga, A Bertolami, M Bertolami, C Amodeo, and D Calhoun. Obstructive sleep apnea, hypertension and cardiovascular diseases. *Journal of human hypertension*, 29(12):705–712, 2015.

[50] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *NIPS*, 2014.

[51] Claudia Gorecki, S José Closs, Jane Nixon, and Michelle Briggs. Patient-reported pressure ulcer pain: a mixed-methods systematic review. *Journal of pain and symptom management*, 42(3):443–459, 2011.

[52] Kristina Grifantini. How's my sleep?: Personal sleep trackers are gaining in popularity, but their accuracy is still open to debate. *IEEE pulse*, 5(5):14–18, 2014.

[53] Timo Grimm, Manuel Martinez, Andreas Benz, and Rainer Stiefelhagen. Sleep position classification from a depth camera using bed aligned maps. In *2016 23rd International Conference on Pattern Recognition (ICPR)*, pages 319–324. IEEE, 2016.

[54] Weixi Gu, Zheng Yang, Longfei Shangguan, Wei Sun, Kun Jin, and Yunhao Liu. Intelligent sleep stage mining service with smartphones. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 649–660. ACM, 2014.

[55] Reinder Haakma and R Beun. Unobtrusive sleep monitoring. In *Measuring Behavior*, volume 2012, page 122. Citeseer, 2012.

[56] Tian Hao, Guoliang Xing, and Gang Zhou. isleep: unobtrusive sleep quality monitoring using smartphones. In *Proceedings of the 11th ACM Conference on Embedded Networked Sensor Systems*, page 4. ACM, 2013.

[57] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[58] Ellen et al. Herbst. Adaptation effects to sleep studies in participants with and without chronic posttraumatic stress disorder. *Psychophysiology*, 2010.

[59] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 1997.

[60] Enamul Hoque, Robert F Dickerson, and John A Stankovic. Monitoring body positions and movements during sleep using wisps. In *Wireless Health 2010*, pages 44–53. ACM, 2010.

[61] Chi-Chun Hsia, Yu-Wei Hung, Yu-Hsien Chiu, and Chia-Hao Kang. Bayesian classification for bed posture detection based on kurtosis and skewness estimation. In *HealthCom 2008-10th International Conference on e-health Networking, Applications and Services*, pages 165–168. IEEE, 2008.

[62] Chen-Yu Hsu, Aayush Ahuja, Shichao Yue, Rumen Hristov, Zachary Kabelac, and Dina Katabi. Zero-effort in-home sleep and insomnia monitoring using radio signals. *IMWUT*, 2017.

[63] Chen-Yu Hsu, Yuchen Liu, Zachary Kabelac, Rumen Hristov, Dina Katabi, and Christine Liu. Extracting gait velocity and stride length from surrounding radio signals. In *CHI*. ACM, 2017.

[64] Aapo Hyvärinen, Juha Karhunen, and Erkki Oja. *Independent component analysis*. John Wiley & Sons, 2004.

[65] Aapo Hyvärinen and Erkki Oja. Independent component analysis: algorithms and applications. *Neural networks*, 2000.

[66] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

[67] Jawbone. `https://jawbone.com/`, 2017.

[68] Ossi Kaltiokallio, Huseyin Yigitler, Riku Jantti, and Neal Patwari. Non-invasive respiration rate monitoring using a single cots tx-rx pair. *IPSN*, 2014.

[69] Matthew Kay, Eun Kyoung Choe, Jesse Shepherd, Benjamin Greenstein, Nathaniel Watson, Sunny Consolvo, and Julie A. Kientz. Lullaby: A capture & access system for understanding the sleep environment. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, UbiComp '12, pages 226–234, New York, NY, USA, 2012. ACM.

[70] Robert Kloster and Torstein Engelskjøn. Sudden unexpected death in epilepsy (sudep): a clinical perspective and a search for risk factors. *Journal of Neurology, Neurosurgery & Psychiatry*, 67(4):439–444, 1999.

[71] Anastasi Kosmadopoulos, Charli Sargent, David Darwent, Xuan Zhou, and Gregory D Roach. Alternatives to polysomnography (psg): a validation of wrist actigraphy and a partial-psg system. *Behavior research methods*, 46(4):1032–1041, 2014.

[72] Manikanta Kotaru, Kiran Joshi, Dinesh Bharadia, and Sachin Katti. Spotfi: Decimeter level localization using wifi. In *ACM SIGCOMM Computer Communication Review*. ACM, 2015.

[73] Raghavendra Kotikalapudi. Residual networks implementation using keras-1.0 functional api, 2017.

[74] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012.

[75] Swarun Kumar, Stephanie Gil, Dina Katabi, and Daniela Rus. Accurate indoor localization with zero start-up cost. In *MobiCom*. ACM, 2014.

[76] Mark E Kunik, Kent Roundy, Connie Veazey, Julianne Souchek, Peter Richardson, Nelda P Wray, and Melinda A Stanley. Surprisingly high prevalence of anxiety and depression in chronic breathing disorders. *Chest*, 2005.

[77] J Richard Landis and Gary G Koch. The measurement of observer agreement for categorical data. *Biometrics*, 1977.

[78] Hong Ji Lee, Su Hwan Hwang, Seung Min Lee, Yong Gyu Lim, and Kwang Suk Park. Estimation of body postures on bed using unconstrained ecg measurements. *IEEE journal of biomedical and health informatics*, 17(6):985–993, 2013.

[79] Ronald Ley. Breathing and the psychology of emotion, cognition, and behavior. In *Behavioral and psychological approaches to breathing disorders*. Springer, 1994.

[80] Kenneth L Lichstein, Kristen C Stone, James Donaldson, Sidney D Nau, James P Soeffing, David Murray, Kristin W Lester, and R Neal Aguillard. Actigraphy validation with insomnia. *SLEEP-NEW YORK THEN WESTCHESTER-*, 29(2):232, 2006.

[81] Jennifer A Liebenthal, Shasha Wu, Sandra Rose, John S Ebersole, and James X Tao. Association of prone position with sudden unexpected death in epilepsy. *Neurology*, 84(7):703–709, 2015.

[82] Feng Lin, Yan Zhuang, Chen Song, Aosen Wang, Yiran Li, Changzhan Gu, Changzhi Li, and Wenyao Xu. Sleepsense: A noncontact and cost-effective sleep monitoring system. *IEEE Transactions on Biomedical Circuits and Systems*, 2016.

[83] Jason J Liu, Wenyao Xu, Ming-Chun Huang, Nabil Alshurafa, Majid Sarrafzadeh, Nitin Raut, and Behrooz Yadegar. Sleep posture analysis using a dense pressure sensitive bedsheet. *Pervasive and Mobile Computing*, 10:34–50, 2014.

[84] Jia Liu, Xingyu Chen, Shigang Chen, Xiulong Liu, Yanyan Wang, and Lijun Chen. Tagsheet: Sleeping posture recognition with an unobtrusive passive tag matrix. In *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*, pages 874–882. IEEE, 2019.

[85] Jian Liu, Yingying Chen, Yan Wang, Xu Chen, Jerry Cheng, and Jie Yang. Monitoring vital signs and postures during sleep using wifi signals. *IEEE Internet of Things Journal*, 5(3):2071–2084, 2018.

[86] Shuangjun Liu and Sarah Ostadabbas. A vision-based system for in-bed posture tracking. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1373–1382, 2017.

[87] Xuefeng Liu, Jiannong Cao, Shaojie Tang, and Jiaqi Wen. Wi-sleep: Contactless sleep monitoring via wifi signals. In *2014 IEEE Real-Time Systems Symposium*, pages 346–355. IEEE, 2014.

[88] Mingsheng Long, Yue Cao, Jianmin Wang, and Michael I Jordan. Learning transferable features with deep adaptation networks. *ICML*, 2015.

[89] Xi Long, Jie Yang, Tim Weysen, Reinder Haakma, Jérôme Foussier, Pedro Fonseca, and Ronald M Aarts. Measuring dissimilarity between respiratory effort signals based on uniform scaling for sleep staging. *Physiological measurement*, 2014.

[90] Brendan P Lucey, Jennifer S Mcleland, Cristina D Toedebusch, Jill Boyd, John C Morris, Eric C Landsness, Kelvin Yamada, and David M Holtzman. Comparison of a single-channel eeg sleep study to polysomnography. *Journal of sleep research*, 2016.

[91] National Heart Lung and Blood Institute (NHLBI). Why is sleep important?, February 2012.

[92] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *JMLR*, 2008.

[93] Bassem R Mahafza. *Radar systems analysis and design using MATLAB*. Chapman and Hall/CRC, USA, 2005.

[94] Alireza Makhzani, Jonathon Shlens, Navdeep Jaitly, Ian Goodfellow, and Brendan Frey. Adversarial autoencoders. *arXiv preprint arXiv:1511.05644*, 2015.

[95] Akshay Menon and Manoj Kumar. Influence of body position on severity of obstructive sleep apnea: a systematic review. *ISRN otolaryngology*, 2013, 2013.

[96] Luke Metz, Ben Poole, David Pfau, and Jascha Sohl-Dickstein. Unrolled generative adversarial networks. *arXiv preprint arXiv:1611.02163*, 2016.

[97] Fernand Meyer. Color image segmentation. In *Image Processing and its Applications, 1992., International Conference on*, pages 303–306. IET, 1992.

[98] Jun-Ki Min, Afsaneh Doryab, Jason Wiese, Shahriyar Amini, John Zimmerman, and Jason I Hong. Toss'n'turn: smartphone as sleep and sleep quality detector. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 477–486. ACM, 2014.

[99] Anat Mirelman, Inbar Hillel, Lynn Rochester, Silvia Del Din, Bastiaan R Bloem, Laura Avanzino, Alice Nieuwboer, Inbal Maidan, Talia Herman, Avner Thaler, et al. Tossing and turning in bed: Nocturnal movements in parkinson's disease. *Movement Disorders*, 35(6):959–968, 2020.

[100] Nikolaos Mitianoudis and Tania Stathaki. Pixel-based and region-based image fusion schemes using ica bases. *Information Fusion*, 2007.

[101] Ken Monahan, Amy Storfer-Isser, Reena Mehra, Eyal Shahar, Murray Mittleman, Jeff Rottman, Naresh Punjabi, Mark Sanders, Stuart F Quan, Helaine Resnick, et al. Triggering of nocturnal arrhythmias by sleep-disordered breathing events. *Journal of the American College of Cardiology*, 2009.

[102] Rajalakshmi Nandakumar, Shyamnath Gollakota, and Nathaniel Watson. Contactless sleep apnea detection on smartphones. In *Proceedings of the 13th Annual International Conference on Mobile Systems, Applications, and Services*. ACM, 2015.

[103] National Institute of Health. Sleep disorders. `http://www.ninds.nih.gov/disorders/brain_basics/understanding_sleep.htm#sleep_disorders`.

[104] ALISTER McKENZIE Neill, Susan Michelle Angus, Dimitar Sajkov, and RONALD DOUGLAS McEVOY. Effects of sleep posture on upper airway stability in patients with obstructive sleep apnea. *American journal of respiratory and critical care medicine*, 155(1):199–204, 1997.

[105] Phuc Nguyen, Xinyu Zhang, Ann Halbower, and Tam Vu. Continuous and fine-grained breathing volume monitoring from afar using wireless signals. In *IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications*, pages 1–9. IEEE, 2016.

[106] F Javier Nieto, Terry B Young, Bonnie K Lind, Eyal Shahar, Jonathan M Samet, Susan Redline, Ralph B D'agostino, Anne B Newman, Michael D Lebowitz, Thomas G Pickering, et al. Association of sleep-disordered breathing, sleep apnea, and hypertension in a large community-based study. *Jama*, 2000.

[107] Standards of Practice Committee of the American Academy of Sleep Medicine et al. Practice parameters for the role of actigraphy in the study of sleep and circadian rhythms: an update for 2002. *Sleep*, 2003.

[108] Maurice M Ohayon. Epidemiology of insomnia: what we know and what we still need to learn. *Sleep medicine reviews*, 6(2):97–111, 2002.

[109] Erkki Oja. Applications of independent component analysis. In *International Conference on Neural Information Processing*. Springer, 2004.

[110] Sarah Ostadabbas, Maziyar Baran Pouyan, Mehrdad Nourani, and Nasser Kehtarnavaz. In-bed posture classification and limb identification. In *2014 IEEE Biomedical Circuits and Systems Conference (BioCAS) Proceedings*, pages 133–136. IEEE, 2014.

[111] Nobuyuki Otsu. A threshold selection method from gray-level histograms. *Automatica*, 11(285-296):23–27, 1975.

[112] Kurt Peek. Estimation and compensation of frequency sweep nonlinearity in fmcw radar. Master's thesis, University of Twente, 2011.

[113] Lionel Pigou, Aäron Van Den Oord, Sander Dieleman, Mieke Van Herreweghe, and Joni Dambre. Beyond temporal pooling: Recurrence and temporal convolutions for gesture recognition in video. *IJCV*, 2015.

[114] Charles P Pollak, Warren W Tryon, Haikady Nagaraja, and Roger Dzwonczyk. How accurately does wrist actigraphy identify the states of sleep and wakefulness? *SLEEP-NEW YORK*, 2001.

[115] Djordje Popovic, Michael Khoo, and Philip Westbrook. Automatic scoring of sleep stages and cortical arousals using two electrodes on the forehead: validation in healthy adults. *Journal of sleep research*, 2014.

[116] M Baran Pouyan, Sarah Ostadabbas, Masoud Farshbaf, Rasoul Yousefi, Mehrdad Nourani, and MDM Pompeo. Continuous eight-posture classification for bed-bound patients. In *2013 6th International Conference on Biomedical Engineering and Informatics*, pages 121–126. IEEE, 2013.

[117] Pulmonology. `https://www.columbiamemorialhealth.org/pulmonology/`, 2018.

[118] Sanjay Purushotham, Wilka Carvalho, Tanachat Nilanon, and Yan Liu. Variational recurrent adversarial deep domain adaptation. *ICLR*, 2017.

[119] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.

[120] Tauhidur Rahman, Alexander T Adams, Ruth Vinisha Ravichandran, Mi Zhang, Shwetak N Patel, Julie A Kientz, and Tanzeem Choudhury. Dopplesleep: A contactless unobtrusive sleep sensing system using short-range doppler radar. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 39–50. ACM, 2015.

[121] Allan Rechtschaffen and Anthony Kales. A manual of standardized terminology, techniques and scoring system for sleep stages of human subjects. *US Government Printing Office, US Public Health Service*, 1968.

[122] Grace Gita Redhyka, Dika Setiawan, and Demi Soetraprawata. Embedded sensor fusion and moving-average filter for inertial measurement unit (imu) on the microcontroller-based stabilized platform. In *2015 International Conference on Automation, Cognitive Science, Optics, Micro Electro-Mechanical System, and Information Technology (ICACOMIT)*, pages 72–77. IEEE, 2015.

[123] Brant W Riedel, Carolyn F Winfield, and Kenneth L Lichstein. First night effect and reverse first night effect in older adults with primary insomnia: does anxiety play a role? *Sleep medicine*, 2(2):125–133, 2001.

[124] Alejandro Rodríguez-Molinero, Leire Narvaiza, Jorge Ruiz, and César Gálvez-Barrón. Normal respiratory rate and peripheral blood oxygen saturation in the elderly population. *Journal of the American Geriatrics Society*, 2013.

[125] Hiroshi Saruwatari, Satoshi Kurita, and Kazuya Takeda. Blind source separation combining frequency-domain ica and beamforming. In *ICASSP*. IEEE, 2001.

[126] Nandakumar Selvaraj. Psychological acute stress measurement using a wireless adhesive biosensor. In *EMBC*. IEEE, 2015.

[127] Purav C Shah, Eric Yudelevich, Frank Genese, Miguel Martillo, Iazsmin B Ventura, Katherine Fuhrmann, Marie Mortel, Daniel Levendowski, Charlisa D Gibson, Pius Ochieng, et al. Can disrupted sleep affect mortality in the mechanically ventilated critically ill? *A state of unrest: Sleep/SDB in the ICU and hospital*, 2016.

[128] John R Shambroom, Stephan E Fábregas, and Jack Johnstone. Validation of an automated wireless system to monitor sleep in healthy adults. *Journal of sleep research*, 2012.

[129] Mehdi Shokoueinejad, Chris Fernandez, Emily Carroll, Fa Wang, Jake Levin, Sam Rusk, Nick Glattard, Ashley Mulchrone, Xuan Zhang, Ailiang Xie, et al. Sleep apnea: a review of diagnostic sensors, algorithms, and therapies. *Physiological measurement*, 38(9):R204, 2017.

[130] Deepak Shrivastava, Syung Jung, Mohsen Saadat, Roopa Sirohi, and Keri Crewson. How to interpret the results of a sleep study. *Journal of community hospital internal medicine perspectives*, 4(5), 2014.

[131] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint*, 2014.

[132] Eivind Schjelderup Skarpsno, Paul Jarle Mork, Tom Ivar Lund Nilsen, and Andreas Holtermann. Sleep positions and nocturnal body movements based on free-living accelerometer recordings: association with demographics, lifestyle, and insomnia symptoms. *Nature and science of sleep*, 9:267, 2017.

[133] Sleepbot. `https://mysleepbot.com/`, 2017.

[134] Sleep cycle. `https://www.sleepcycle.com/`, 2017.

[135] Sleep diary. `https://www.sleepfoundation.org/sleep-diary`, 2021.

[136] Sleep profiler. `http://www.advancedbrainmonitoring.com/sleep-profiler/`, 2017.

[137] Sleep study. `https://www.sleepfoundation.org/sleep-studies`, 2021.

[138] Sleepview. `https://clevemed.com/sleepview/`, 2018.

[139] Simon Smith and John Trinder. Detecting insomnia: comparison of four self-report measures of sleep in a young adult population. *Journal of sleep research*, 10(3):229–235, 2001.

[140] Frederick Snyder, J Allan Hobson, Donald F Morrison, and Frederick Goldfrank. Changes in respiration, heart rate, and systolic blood pressure in human sleep. *Journal of Applied Physiology*, 1964.

[141] splus. S-plus, 2017.

[142] Tania Stathaki. *Image fusion: algorithms and applications*. Elsevier, 2011.

[143] C Stepnowsky, D Levendowski, D Popovic, I Ayappa, and DM Rapoport. Scoring accuracy of automated sleep staging from a bipolar electroocular recording compared to manual scoring by multiple raters. *Sleep medicine*, 14(11):1199, 2013.

[144] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. *arXiv preprint arXiv:1409.3215*, 2014.

[145] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. *CVPR*, 2015.

[146] Yu Takahashi, Tomoya Takatani, Keiichi Osako, Hiroshi Saruwatari, and Kiyohiro Shikano. Blind spatial subtraction array for speech enhancement in noisy environment. *IEEE transactions on audio, speech, and language processing*, 2009.

[147] Alexander Tataraidze, Lyudmila Korostovtseva, Lesya Anishchenko, Mikhail Bochkarev, Yurii Sviryaev, and Sergey Ivashov. Bioradiolocation-based sleep stage classification. In *2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 2839–2842. IEEE, 2016.

[148] Yonglong Tian, Guang-He Lee, Hao He, Chen-Yu Hsu, and Dina Katabi. Rf-based fall monitoring using convolutional neural networks. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 2(3):137, 2018.

[149] Wang Tianben, Daqing Zhang, Yuanqing Zheng, Tao Gu, Xingshe Zhou, and Bernadette Dorizzi. C-fmcw based contactless respiration detection using acoustic signal. In *IMWUT*, 2017.

[150] Eric Tzeng, Judy Hoffman, Trevor Darrell, and Kate Saenko. Simultaneous deep transfer across domains and tasks. In *Proceedings of the IEEE international conference on computer vision*, pages 4068–4076, 2015.

[151] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. *NIPS Workshop on Adversarial Training*, 2016.

[152] Péter Várady, Szabolcs Bongár, and Zoltán Benyó. Detection of airway obstructions and sleep apnea by analyzing the phase relation of respiration movement signals. *IEEE Transactions on Instrumentation and Measurement*, 2003.

[153] Deepak Vasisht, Swarun Kumar, and Dina Katabi. Decimeter-level localization with a single wifi access point. In *13th USENIX Symposium on Networked Systems Design and Implementation (NSDI 16)*, pages 165–178. USENIX Association, 2016.

[154] Carl Vondrick, Hamed Pirsiavash, and Antonio Torralba. Generating videos with scene dynamics. *arXiv preprint arXiv:1609.02612*, 2016.

[155] Hao Wang, Daqing Zhang, Junyi Ma, Yasha Wang, Yuxiang Wang, Dan Wu, Tao Gu, and Bing Xie. Human respiration detection with commodity wifi devices: do user location and body orientation matter? In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 25–36. ACM, 2016.

[156] Hao Wang, Daqing Zhang, Yasha Wang, Junyi Ma, Yuxiang Wang, and Shengjie Li. Rt-fall: A real-time and contactless fall detection system with commodity wifi devices. *IEEE Transactions on Mobile Computing*, 16(2):511–526, 2016.

[157] Tianben Wang, Daqing Zhang, Yuanqing Zheng, Tao Gu, Xingshe Zhou, and Bernadette Dorizzi. C-fmcw based contactless respiration detection using acoustic signal. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 1(4):170, 2018.

[158] Wei Wang, Alex X Liu, and Muhammad Shahzad. Gait recognition using wifi signals. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 363–373. ACM, 2016.

[159] Xuyu Wang, Chao Yang, and Shiwen Mao. Phasebeat: Exploiting csi phase data for vital sign monitoring with commodity wifi devices. In *2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS)*, pages 1230–1239. IEEE, 2017.

[160] Xuyu Wang, Chao Yang, and Shiwen Mao. Tensorbeat: Tensor decomposition for monitoring multiperson breathing beats with commodity wifi. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 9(1):8, 2017.

[161] Yuxi Wang, Kaishun Wu, and Lionel M Ni. Wifall: Device-free fall detection by wireless networks. *IEEE Transactions on Mobile Computing*, 16(2):581–594, 2016.

[162] Teng Wei and Xinyu Zhang. mtrack: High-precision passive tracking using millimeter wave radios. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*. ACM, 2015.

[163] L Welstein, WC Dement, D Redington, C Guilleminault, and MM Mitler. Insomnia in the san francisco bay area: a telephone survey. In *Sleep/wake disorders: natural history, epidemiology, and long-term evolution*, pages 73–85. Raven Press New York, 1983.

[164] Chenshu Wu, Zheng Yang, Zimu Zhou, Xuefeng Liu, Yunhao Liu, and Jiannong Cao. Non-invasive detection of moving and stationary human with wifi. *IEEE Journal on Selected Areas in Communications*, 33(11):2329–2342, 2015.

[165] Lulu Xie, Hongyi Kang, Qiwu Xu, Michael J Chen, Yonghong Liao, Meenakshisundaram Thiyagarajan, John O'Donnell, Daniel J Christensen, Charles Nicholson, Jeffrey J Iliff, et al. Sleep drives metabolite clearance from the adult brain. *science*, 342(6156):373–377, 2013.

[166] Xiaowei Xu, Feng Lin, Aosen Wang, Chen Song, Yu Hu, and Wenyao Xu. On-bed sleep posture recognition based on body-earth mover's distance. In *2015 IEEE Biomedical Circuits and Systems Conference (BioCAS)*, pages 1–4. IEEE, 2015.

[167] H Klar Yaggi, John Concato, Walter N Kernan, Judith H Lichtman, Lawrence M Brass, and Vahid Mohsenin. Obstructive sleep apnea as a risk factor for stroke and death. *New England Journal of Medicine*, 353(19):2034–2041, 2005.

[168] Zhicheng Yang, Parth H Pathak, Yunze Zeng, Xixi Liran, and Prasant Mohapatra. Monitoring vital signs using millimeter wave. In *Proceedings of the 17th ACM international symposium on mobile ad hoc networking and computing*, pages 211–220, 2016.

[169] Heenam Yoon, Suhwan Hwang, Dawoon Jung, Sangho Choi, Kwangmin Joo, Jaewon Choi, Yujin Lee, Do-Un Jeong, and Kwangsuk Park. Estimation of sleep posture using a patch-type accelerometer based device. In *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 4942–4945. IEEE, 2015.

[170] Terry Young, Laurel Finn, Paul E Peppard, Mariana Szklo-Coxe, Diane Austin, F Javier Nieto, Robin Stubbs, and K Mae Hla. Sleep disordered breathing and mortality: eighteen-year follow-up of the wisconsin sleep cohort. *Sleep*, 2008.

[171] Terry Young, Paul Peppard, Mari Palta, K Mae Hla, Laurel Finn, Barbara Morgan, and James Skatrud. Population-based study of sleep-disordered breathing as a risk factor for hypertension. *Archives of internal medicine*, 1997.

[172] Shichao Yue, Hao He, Hao Wang, Hariharan Rahul, and Dina Katabi. Extracting multi-person respiration from entangled rf signals. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 2018.

[173] Shichao Yue, Yuzhe Yang, Hao Wang, Hariharan Rahul, and Dina Katabi. Bodycompass: Monitoring sleep posture with wireless signals. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 2020.

[174] A Zaffaroni, L Gahan, L Collins, E O'hare, C Heneghan, C Garcia, I Fietze, and T Penzel. Automated sleep staging classification using a non-contact biomotion sensor. *Journal of Sleep Research*, 2014.

[175] Feng Zhang, Chen Chen, Beibei Wang, and KJ Ray Liu. Wispeed: A statistical electromagnetic approach for device-free indoor speed estimation. *IEEE Internet of Things Journal*, 5(3):2163–2177, 2018.

[176] Zhenyuan Zhang, Zengshan Tian, and Mu Zhou. Latern: Dynamic continuous hand gesture recognition using fmcw radar sensor. *IEEE Sensors Journal*, 18(8):3278–3289, 2018.

[177] Mingmin Zhao, Fadel Adib, and Dina Katabi. Emotion recognition using wireless signals. In *Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking*, pages 95–108, 2016.

[178] Mingmin Zhao, Tianhong Li, Mohammad Abu Alsheikh, Yonglong Tian, Hang Zhao, Antonio Torralba, and Dina Katabi. Through-wall human pose estimation using radio signals. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7356–7365, 2018.

[179] Mingmin Zhao, Shichao Yue, Dina Katabi, Tommi S Jaakkola, and Matt T Bianchi. Learning sleep stages from radio signals: A conditional adversarial architecture. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 4100–4109. JMLR. org, 2017.