A Wireless Spectrum Analyzer in Your Pocket

Article · February 2015

DOI: 10.1145/2699343.2699353

Tan Zhang
National University of Defense Technology
91 PUBLICATIONS **880** CITATIONS

SEE PROFILE

Ashish Patro
University of Wisconsin–Madison
12 PUBLICATIONS **228** CITATIONS

SEE PROFILE

A Wireless Spectrum Analyzer in Your Pocket

Tan Zhang[#], Ashish Patro[#], Ning Leng[$], Suman Banerjee[#]
Department of Computer Sciences[#], Department of Statistics[$]
University of Wisconsin-Madison, USA
{tzhang, patro, suman}@cs.wisc.edu[#], leng@stat.wisc.edu[$]

ABSTRACT

We propose Snoopy, a system that can translate one's mobile phone or tablet into a low-cost, yet effective RF spectrum analyzer. Since typical spectrum analyzers are specialized hardware that is both expensive to acquire and cumbersome to carry around, they are rarely available for quick-and-easy spectrum sensing while on the go. To address this challenge, Snoopy augments popular mobile devices with a small attachable hardware unit (RF frequency translators) that can provide a reasonable view of the wireless spectrum across different frequency bands. It achieves this by leveraging the spectral scan functionality available in certain 802.11 NICs (e.g., the Atheros 9280 family of chipsets), which provides an unique lens towards the WiFi spectrum (2.4GHz). Through the use of suitable frequency translation, such a view can be flexibly shifted to other spectrum bands. Although such a construction might not match the precision and accuracy of the most sophisticated but expensive spectrum analyzers, we show that by leveraging some carefully designed spectral features, Snoopy can achieve decent accuracy in the TV whitespace band (512 – 698MHz) – it can detect primary signals up to - 90dBm with an average error rate of <15%, while achieving a median error of <4dB in estimating the power of these signals. These promising results suggest that Snoopy is an intriguing option in bringing the ability of spectrum sensing to the masses, thereby truly enabling crowd-sourcing options in this domain.

Categories and Subject Descriptors

C.2.1 [**Network Architecture and Design**]: Wireless communication; C.4 [**Performance of Systems**]: Measurement techniques

Keywords

Spectrum sensing; Crowdsourcing; Smartphone; TV whitespaces

1. INTRODUCTION

Smartphones and tablets are among the most common devices that are carried around by individuals today. As the number of such devices exceed the count of the worldwide human population, a service that is enabled in these devices can truly achieve global reach. This has led to mushrooming of many data-aggregation services implemented on them that take advantage of its scale and penetration. In this paper, we consider the possibility of implementing a low-cost RF spectrum sensing capability on this common platform, thereby unlocking significant new opportunities for citizen-contributed analytics of the RF spectrum.

Current RF spectrum sensing approaches: The ability to efficiently sense and infer spectrum occupancy is an important need in the evolving debate of spectrum availability. Many use cases exist today. Regulatory bodies across the world, such as FCC in the US and Ofcom in the UK, are exploring new policy regimes where dynamic spectrum sharing will play an increasingly central role. To ensure such decisions are grounded in reality, these organizations need widespread data to determine the efficacy of spectrum utilization. Similarly, if one is curious to learn whether prohibited wireless transmitters are operating in a certain region (for example, cellphones not in the airplane mode in a flying aircraft or in a sensitive facility), spectrum sensing can provide the answer. In particular, any kind of spectrum transmission policy enforcement rely on effective spectrum sensing at its core.

RF spectrum sensing has traditionally been done following a "cathedral" model, by using sophisticated and specialized hardware, e.g., high-end spectrum analyzers. This type of hardware is both expensive and cumbersome, and can only be operated by a few experts. Using such hardware, a few concerted efforts from researchers have led to significant spectrum measurements. Notable among them are the efforts of SharedSpectrum [14], the Spectrum Observatory project [8], and SpecNet [3]. While these measurement efforts can provide high quality and actionable data, the cost of hardware and training effort for operating these systems have limited their deployment to a selected few sites and locations.

Proposed approach in Snoopy: In this work, we envision an alternative solution where the capability of RF spectrum sensing is embedded in some popular off-the-shelf mobile devices, thus bringing spectrum sensing to the "bazaar" and opening it up to the masses. Such a solution can truly enable the creation of a widespread real-time spectrum observatory. In particular, we propose a measurement system called *Snoopy* (Spectrum knowledge out of pocket), which can turn those off-the-shelf smartphone devices into spectrum sensors to effectively perform spectrum monitoring functions.

The advantage of this bazaar-style spectrum sensing is obvious — many more users can easily deploy this service, thus allowing a greater reach of spectrum sensing activities. To realize this goal, our proposed solution - Snoopy utilizes emerging features that are available in many common WiFi NICs to collect energy samples for each WiFi subcarrier at a fine timescale (often referred to as WiFi spectrum scan). Drivers capable of exposing such spectral

HotMobile'15, February 12–13, 2015, Santa Fe, New Mexico, USA.
Copyright © 2015 ACM 978-1-4503-3391-7/15/02 ...$15.00.
http://dx.doi.org/10.1145/2699343.2699353.

data exists for the Intel 5300 cards [5]. In addition, we have recently released an analogous driver for the Atheros 92xx and 93xx chipsets [19]. By attaching a RF frequency translator to this type of WiFi NICs, it is possible to turn this WiFi spectral scan feature into a wide-band spectrum sensing functionality (the exact bands that can be scanned depends on the precise frequency translator used). For our initial prototype, we use a slightly expensive RF frequency translator from prior work [13, 16]. Nevertheless, in discussion with various hardware vendors, it appears that relatively low-cost versions ($<$ \$50 [9] and likely far cheaper) are possible, especially if the focus is purely for receiving spectrum data, and not for transmitting.

While extolling the virtues of Snoopy, we hasten to add that such low cost hardware is not likely to achieve the same level of accuracy compared to those sophisticated spectrum analyzers. High-end spectrum analyzers like ThinkRF analyzer [17] usually have a much higher sampling rate (125MHz), thereby achieving a finer frequency granularity ($<$ 1 KHz). Compared to the spectrum granularity of WiFi spectral scan in Snoopy ($\sim$ 312.5 KHz), the former can potentially achieve greater accuracy for detecting signal activity when combined with some well-known feature based detection algorithms [3, 11, 15]. Nevertheless, we show that some carefully designed statistics on the low-resolution spectrum can indeed be helpful in improving detection accuracy. These statistical features are widely present in many types of signals (e.g., TV, microphone, WiFi, WiMax) and thus generally applicable to feature detection techniques. In addition, we believe that the affordability of such hardware along with the simplicity of the mobile platform's software can make it a compelling sensing device, at least for researchers and hobbyists, thus leading to its wide adoption as a "quick-and-dirty" tool. This makes combining data from many diverse sources possible to ultimately achieve a significantly higher measurement accuracy.

Before delving into the design of Snoopy, we would like to point out that one recent effort [2] has also proposed the use of mobile devices for RF spectrum sensing. The solution is based on a different architecture that uses a RTL-SDR hardware (essentially a TV-dongle) to directly capture measurements at specific parts of a frequency band. While retaining many benefits of Snoopy, we believe that the granularity of WiFi spectral data available from common WiFi NICs are finer and richer than those available from common TV dongles, ultimately enabling Snoopy to achieve higher accuracy in estimating signal power and detecting different types of signals from spectrum measurements (e.g., TV broadcasts and microphones in the TV band in our experiment). Furthermore, a frequency translator can flexibly sense a wider band (30 MHz - 7.5GHz) than RTL-SDR (52 MHz - 2.2 GHz). Nevertheless, Snoopy suffers from a current limitation that it only works with WiFi radios based on above chipsets, which could either be built into smartphone devices or attached externally through their MicroUSB port. Unfortunately, none of them have exposed an external RF connector to interface with our frequency translator. Thus, in our proof-of-the-concept implementation, we use an OpenWrt router hosting a Atheros 9280 card to serve as mobile devices, which is connected to our translator via a RF cable (Figure 1).

Key Contributions: We had to address a number of technical challenges that form the key contributions of this work:

- We have designed and implemented a mobile device based spectrum sensing system – Snoopy. Snoopy leverages the spectrum sensing capabilities of off-the-shelf WiFi cards (Atheros chipsets 9280, 9271, etc.), while attaching them with a frequency translator to achieve a wide sensing range from 30 MHz - 7.5 GHz (§2).

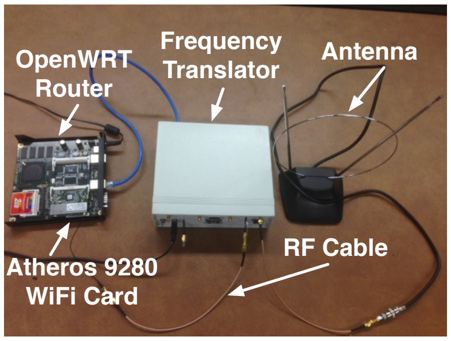


Figure 1: Snoopy sensing prototype.

- We have developed specific techniques that take the sub-carrier spectrum scan data from the WiFi cards to detect signals at low-power (up to -90dBm). Our experiments on the UHF television band show that Snoopy can detect primary signals (i.e., TV and microphone) at a wide range of power with an error rate of $<$15%. In addition, it can measure the power of TV channels with an error of $<$ 4dB in most cases (§3).
- We have ported the software module of Snoopy to an Android application that is readily available for Nexus Android tablets and phones to perform spectrum sensing in the 2.4 GHz WiFi band [20]. We have also released an open-source version of the WiFi spectral scan feature for the Android platform that can be utilized by the community [19]. We are currently working on the last hurdle of building a compatible hardware connector for our frequency translator to connect to off-the-shelf mobile devices, which will be discussed in §5.

2. SNOOPY DESIGN

In this section, we first highlight the major challenges in designing a low-cost mobile sensing platform. We then present Snoopy and its key techniques to overcome these challenges.

Design challenges: We have identified three major challenges in designing a mobile sensing platform for satisfying various aspects of performance requirement, i.e., delay, frequency range, and detection sensitivity.

- *Low-latency:* Snoopy aims to perform real-time spectrum analysis with a low latency on the order of milliseconds. This rules out many software based FFT options.
- *Frequency range:* Snoopy should be flexible enough to monitor a wide frequency range spanning UHF, cellular, ISM bands.
- *Detection sensitivity:* Prior work Airshark [12] focused on interference scenarios and is capable of detecting both WiFi and Non-WiFi interference signals at a relatively high power (> -80 dBm). In contrast, Snoopy aims to determine spectrum availability, thus having to use the same WiFi scanner in Airshark to detect primary signals (e.g., TV and microphones in the TV band) at a much lower power (up to -90dBm).

Snoopy consists of three major components — a) a frequency translator that translates incoming signals from any frequency band to the 2.4GHz WiFi band, b) a WiFi based mobile device that uses its built-in WiFi radio to collect spectrum samples (FFTs) of the translated signal, c) a software module that performs an enhanced signal detection algorithm on the spectrum scan data to detect different types of signals and measure their power. Figure 1 shows our sensing prototype, which uses a OpenWRT based router controlling an Atheros AR9280 card as the mobile device to conduct spectrum sensing.

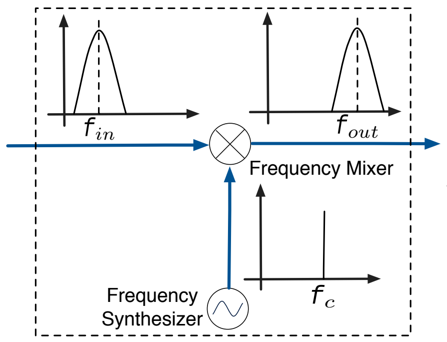


Figure 2: Simplified diagram of a RF frequency translator.

Operation	Latency
Capturing 20MHz spectrum by a WiFi radio	120us
Switching frequency bands by a translator	1ms
Signal detection for each TV channel by a router	10ms

Table 1: Approximate latency of different operations in Snoopy.

Signal Types	Spectrum Features
TV v.s. Noise	Power, Fourier Transformation coefficients
Microphone v.s. Noise	Power, 75th quartile FFT, Fourier Transformation coefficients
TV v.s. Microphone	75th quartile FFT

Table 2: Features used by Snoopy to classify different types of signals in the TV band.

Frequency translator: Figure 2 shows a simplified diagram to illustrate the operation of a RF frequency translator. A translator leverages two major hardware components for frequency translation, i.e., frequency synthesizer and frequency mixer. The frequency synthesizer is able to generate a carrier signal (sine waves) at any specified frequency f_c. This signal is taken by the frequency mixer, which combines it with the input RF signal at f_{in} to generate an output signal at a center frequency of f_{out}. Here $f_{out} = f_{in} + f_c$. By tuning f_c, we can convert signals from any target band to the WiFi band (f_{out} = 2.4 GHz), thereby allowing the WiFi radio of a mobile device to collect spectrum data of these signals after frequency conversion [1].

In our current implementation, we use the *Wide Band Digital Radio* (WDR) to perform the frequency translation function. This platform has been used in prior work [13, 16] to translate 2.4 GHz WiFi signals to the 600 MHz television band to enable TV whitespace communications. Similar platforms have been reported in prior work such as WhiteFi [4]. There are two unique advantages of our translator platform. First, it can translate signals from a wide frequency range of 30MHz – 7.5GHz, thus allowing Snoopy to monitor a variety of licensed and unlicensed bands including land mobile, radio navigation, television, cellular, satellite, and so on. In addition, its frequency synthesizer incurs very low latency (≈1ms) in switching across frequencies (f_c), thus allowing Snoopy to fast sweep across a wide range of spectrum. The WDR uses an Ethernet connection to receive configuration information about f_c, and leverages RF cables to send and receive wireless signals.

Spectrum sensing on WiFi cards: After translating a target frequency band to the WiFi band, Snoopy leverages the off-the-shelf WiFi radio of mobile devices to generate its spectrum. Our current prototype is built on Atheros AR9280 AGN cards and leverages the driver module developed in prior work Airshark [12] to extract spectrum samples. Each spectrum sample comprises the power of 64 sub-carriers (FFTs) over a 20MHz WiFi channel, with each FFT representing the power over a 312.5KHz band (20MHz/64). Since these FFT samples are generated by the WiFi hardware, it incurs a very low latency (120us) for producing a 20MHz spectrum. Nevertheless, we find it can take much longer time (≈20ms) to switch to a different WiFi channel. Given the much lower latency (1ms) of the translator in switching frequencies, we decide to fixed the operating channel of the WiFi card (at 2437MHz), but only changing the carrier frequency f_c of the translator. Table 1 summarizes the latency of different operations performed by Snoopy.

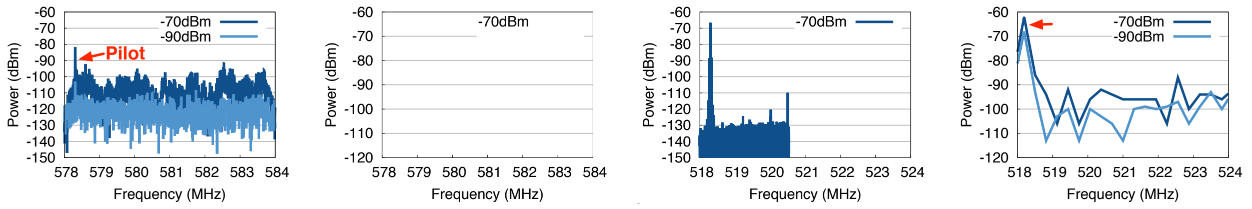
Using WiFi cards to determine TV whitespaces: Prior work such as Airshark [12] has used the aforementioned WiFi radio capabilities to detect non-WiFi activity. In this paper, we motivate the use of these capabilities for determining spectrum availability, with TV whitespaces as an example. Compared to detecting non-WiFi activity, this task is much more challenging due to the need to detect weak primary signals.

[1]A typical RF translator can convert signals in both directions, even though we use it for reception alone.

To demonstrate this challenge, we used Snoopy and a high-end spectrum analyzer [17] to capture the spectrum of a digital TV signal and a microphone signal at two different powers. We will give more details about this spectrum analyzer in Section 3. Figure 3 shows that the spectra captured by these devices have a similar shape for each signal type. Nevertheless, the spectrum collected by Snoopy is much more coarse-grained, with a noisy and fluctuating power distribution in most of its FFT bins. This is because its wider FFT bins can aggregate more noise and the frequency translator also introduces non-negligible distortion. This jagged spectrum leads to far less distinguishable peak features (i.e., TV pilot and microphone tones), which in turn cause higher errors of state-of-the-art feature detection algorithms [7, 11, 15] (Section 3). To improve the accuracy of primary signal detection, we have developed an enhanced feature detection algorithm that leverages some statistical feature inherent in the low resolution spectrum as described next.

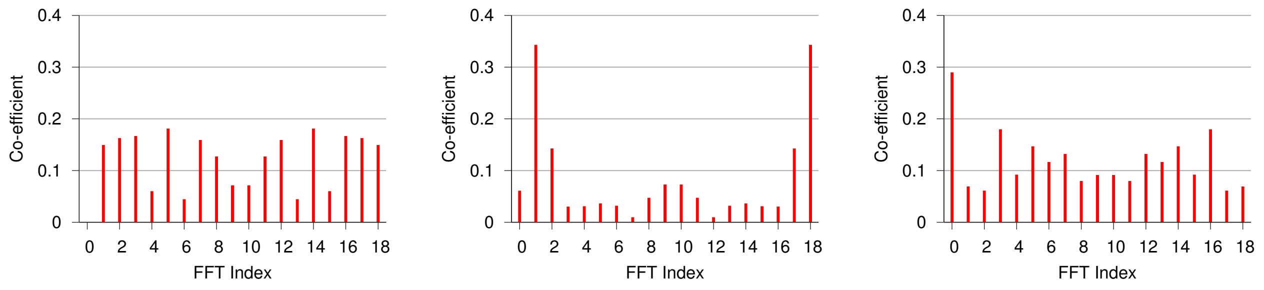
Signal detection based on statistical spectrum features: Our proposed detection algorithm leverages two statistics of the low-resolution spectrum as additional features to improve detection accuracy. These statistics are 1) 75th quartile of FFTs, 2) coefficients of Discrete Fourier Transformation performed on the collected spectrum (FFT over FFT). These features are motivated by the distinct power distribution in the spectrum of primary signals from that of noise. Specifically, Feature 1 is based on the fact that a microphone signal has most of its power concentrated on its audio tones (2 – 3 FFT bins), whereas a TV signal and noise have a relatively even power distribution across the entire 6MHz TV channel (Figure 3(b) 3(d)). Thus, the 75th quartile of FFTs for a microphone signal is much *higher* than TV and noise, thus serving an effective feature to detect microphone signals.

The motivation behind Feature 2 is that the envelop of a spectrum can be represented by the sum of sinusoid waves at different frequencies, by *performing Discrete Fourier Transformation again*. The coefficients of Fourier Transformation are the weights used to combine different sinusoid waves to reconstruct the original spectral shape. Thus, the distribution of these coefficients can comprehensively capture all the shape related features, including the bandwidth and peak features used in prior approaches [7, 11, 15], along with many subtle ones such as the notch in the center of a WiFi spectrum. To make best of this generalized shape feature, we leverage some state-of-the-art classifiers to assign different weights on



(a) TV-Analyzer (b) TV-Snoopy (c) Microphone-Analyzer (d) Microphone-Snoopy

Figure 3: Spectrum of primary signals captured at different powers by a high-end spectrum analyzer (at 0.238KHz resolution) and Snoopy (at 312.5KHz resolution). The left 2 graphs show the TV spectrum and the right 2 graphs show the microphone spectrum.



(a) TV (b) Microphone (c) Noise

Figure 4: Coefficients of Discrete Fourier Transformation applied to the spectrum of different types of signals.

these coefficients, thereby relying more on those distinct features for signal classification.

To demonstrate the efficacy of this statistical feature, Figure 4 shows the co-efficient values after applying Fourier Transformation on the spectrum of different signals captured by Snoopy as shown in Figure 3(b) 3(d). For each spectrum, we have normalized the spectral FFTs to their median value before applying Fourier Transformation to make these coefficients invariant to the signal power. The x axis shows the index of coefficients and a higher order index indicates a sine wave at higher frequency. We note the distribution of co-coefficients differ for each type of signals. The microphone signal has a large high-order coefficient due to its narrow tone. In contrast, the noise has a larger zero-order coefficient because it has a random fluctuating pattern. Finally, a TV signal has similar coefficients because of its trapezoid spectrum with jagged envelop. These distinct distributions suggest that the coefficients of Fourier Transformation applied to the spectrum envelop can indeed be a useful feature for classifying different types of signals.

Putting it all together, our detection procedure proceeds as follows. a) We start by using Airshark module to collect spectrum from 20 MHz spectral blocks in the UHF band. The collected spectrum is divided according to 6MHz TV channels. b) We then scale these FFTs by their median value to gather spectral features that are invariant of signal power. c) The 75th quartile of the scaled FFTs is calculated. d) We also apply Fourier Transformation on these FFTs to obtain their transformation coefficients. d) We feed these spectral statistics along with channel power to a classifier for determining the type of signals in each TV channel (i.e., TV, MIC). We have experimented with two state-of-the-art classifiers, i.e., support vector machine (SVM) and multinomial logistic regression. Their performance is compared in Section 3. Table 2 summarizes the useful features for classifying different types of signals.

Implementation: Since existing smartphones do not expose any RF connector of their WiFi radios, our proof-of-the-concept implementation uses a Alix3D2 router board equipped with an Atheros 9280 AGN card as the mobile device. The WiFi card is connected to a WDR radio via a RF cable as shown in Figure 1. The router uses an Ethernet connection to configure the WDR to measure a specific frequency band. It receives spectrum samples from the WiFi card to detect primary signals (e.g., TV and microphone) while measuring their power. The entire sensing procedure is implemented in ≈1000 lines of Python code. It incurs a low latency of about 10ms for processing each 6MHz TV channel. We have also released a smartphone based platform [20] that can provide spectrum sensing capabilities for the WiFi band.

3. EVALUATION

In this section, we evaluate the performance of Snoopy in performing two popular spectrum sensing functions in the UHF television band — a) detecting primary signals, b) measuring channel power. We perform a head-to-head comparison between Snoopy and a high-end spectrum analyzer to understand the performance limitation of our platform. Overall, we find Snoopy has a reasonably low error rate of <15% in detecting different types of primary signals at up to -90dBm, which is <10% higher than the spectrum analyzer. In addition, it achieves a median error of $<$ 4dB in measuring the power for most of the TV channels.

Dataset: We collected two datasets of spectrum measurements from the Snoopy platform and a high-end spectrum analyzer from ThinkRF Inc [17]. The ThinkRF analyzer has a 8MHz capture bandwidth with 32768 FFTs, thus producing spectrums at a very fine-grained resolution of 238Hz (see Figure 3(a) 3(c)). This device was used in prior work [15] to accurately detect TV signals at up to -114dBm. Thus, we used this analyzer to establish ground-truth results by connecting both devices to a single antenna through a RF splitter. We have measured 8 UHF channels that are distributed across the entire UHF band. Using the ThinkRF analyzer, we have detected TV signals in 4 channels and microphone signals in 2 other

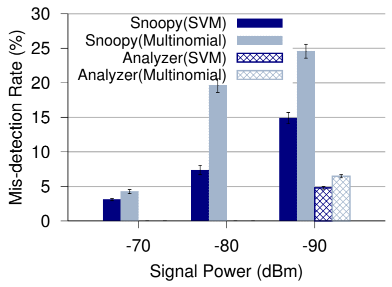


Figure 5: Error rate in detecting primary signals by Snoopy and the ThinkRF analyzer. Error bars show standard deviation.

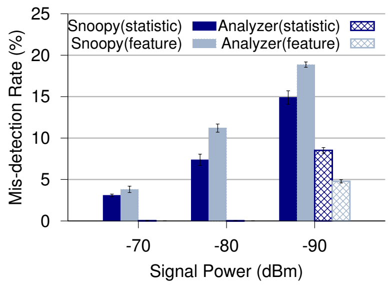


Figure 6: Error rate in detecting primary signals using different spectral features. Error bars show standard deviation.

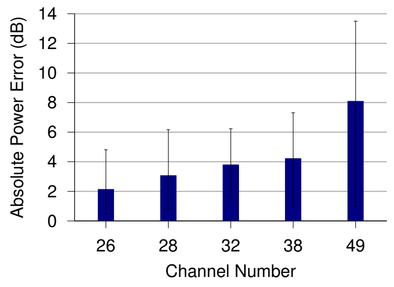


Figure 7: Median error in measuring the power of TV channels by Snoopy. Error bars show 10th and 90th quartile error.

Detected / **Ground truth**	**TV**	**Microphone**	**Noise**
TV	97.7%	1.3%	1%
Microphone	8.4%	91.5%	0.1%
Noise	12.3%	0.5%	87.2%

Table 3: Accuracy of Snoopy in detecting different types of signals with a SVM classifier.

channels. We then used a RF attenuator to attenuate these primary signals (TV and MICs) for constructing spectrum traces at a wide range of power (from -90dBm to -50dBm at 10dB step).

Methodology: To evaluate the accuracy of detecting primary signals, we applied 5-fold cross validation on each dataset by using 80% spectrum samples to train different signal classifiers, and tested them on the remaining data. To quantify the performance of measuring channel power, we compared the power readings that are calculated from spectrum measurements collected by Snoopy and ThinkRF.

Metrics: We applied two metrics to evaluate Snoopy. The mis-detection rate is the ratio of the number of mis-detected spectrum measurements divided by the total number of measurements. The absolute power error is the difference in power readings between Snoopy and the ThinkRF analyzer.

Accuracy in detecting primary signals with two platforms: We start by comparing the performance of Snoopy and the ThinkRF analyzer for detecting two types of primary signals in the UHF band (i.e., TV and microphone). Figure 5 shows the error rates of detecting primary signals at various powers by the two platforms using different classifiers (i.e., support vector machine and multinomial logistic regression). We first observe that when using a SVM classifier, Snoopy can achieve low error rates of 3 – 15% for detecting primary signals at up to -90dBm. These error rates are only 3 – 10% higher than that of the commercial spectrum analyzer using the same classifier. We also note that SVM classifier performs constantly better than multinomial regression classifier for both platforms at all the signal powers. This is because multinomial regression assumes a linear trend between the classification probability and the signal power, which can be violated by these measurements. Thus, we choose SVM classifier in our final implementation. The error rates increase at a lower signal power because the weak signals have a less distinct spectral features from noise spikes. On the other hand, we have found very low error rates (<0.1%) of both platforms in detecting strong primary signals at above -70dBm. We omit this result for the sake of brevity.

Table 3 shows the error rate of Snoopy in classifying different types of signals from its entire dataset using a SVM classifier – Snoopy (SVM). We observe that most of the errors come from misclassification between TV signals and noise because they both have a flat spectrum shape that renders the feature of 75th quartile of FFT to be less effective. In addition, the higher noise floor and distortion introduced by the Snoopy platform can lead to a jagged TV spectrum that has similar Fourier Transform coefficients to noise. We will discuss possible approaches to enhancing detection accuracy in Section 5.

Accuracy in detecting primary signals based on different features: We next quantify the performance gain of incorporating statistical features into signal detection. Figure 6 shows the mis-detection rates of the two platforms based on peak features alone and the additional use of statistical spectral features using a SVM classifier. These peak features have been used in several state-of-the-art spectrum sensing systems [11,15]. We observe that for spectrum collected by Snoopy, statistical features can reduce the error rate by 1 – 5% at different signal powers. Our further analysis shows that the 75-quartile of FFTs is effective in detecting microphone signals, while the Fourier Transform coefficients are beneficial for distinguishing both types of primary signals from noise. For the ThinkRF analyzer, however, the statistical features can introduce slightly higher errors. The reason is that the peak features in its high-resolution spectrum are clear enough for signal detection (Figure 3), and the noise fluctuation can sometimes disturb spectral statistics introducing unnecessary errors. Thus, our proposed statistical features, while beneficial for classifying coarse-grained spectrum, should be replaced with simple peak features for analyzing high-resolution spectrum. Dynamically adapting these features based on the spectral resolution is part of our future work.

Accuracy in measuring channel power: We next evaluate the accuracy of Snoopy in measuring the power of different TV channels. This function is useful for estimating the quality of an operation channel and debugging interference related issues. Figure 7 shows the median error of Snoopy in measuring the power of 4 channels with TV signals and 1 channel with noise (28). We note a low median error of <4dB and a 90th quartile error of <8dB for most of the channels except channel 49. The higher error for channel 49 comes from the significant *distortion* introduced by the frequency translator in the upper UHF band. We find this uneven frequency response is quite common for wide-band frequency translators. Fortunately, we observe all the power offsets have a low variation of < 6dB (error bar). This allows us to use the median error to effectively calibrate these power readings in Snoopy. Such a median error can be obtained by collecting a few measurements in each channel through an one-time calibration effort .

4. RELATED WORK

Spectrum utilization: Several measurement studies have identified spectrum under-utilization in different frequency bands. Authors in [18] have monitored a wide spectrum range (20MHz-6GHz) from three countries and reported that 54% of spectrum is never used and 26% is partially in use. Other work [10,21] has found substantial spectrum resources in TV whitespaces, and proposed different spectrum database designs to better predict vacant TV spectrum. Nevertheless, most of the prior work is based on measurements at a very few locations. V-Scope [15] is a recent measurement system that leverages spectrum sensors mounted on public vehicles to collect wide-area measurements for enhancing TV whitespace databases. Snoopy can significantly extend the measurement coverage of these prior systems by bringing spectrum sensing to the masses of mobile devices.

Sensing platforms: Prior measurement systems [3, 11, 15] use commercial spectrum analyzers to collect measurements, while extracting spectral features to detect different types of primary signals. Unlike these systems, Snoopy can potentially transform off-the-shelf smartphones to spectrum sensors by using a frequency translator to extend the sensing capability of their WiFi radios. While similar translation devices have been reported in prior systems [4,11], they are primarily used to enable TV whitespaces communications by translating signals of commercial WiFi radios.

A recent system [2] has explored the similar concept of spectrum sensing on smartphones by connecting them with a TV dongle (RTL-SDR). While the two systems share the same goal, their approaches differ significantly. Snoopy relies on the WiFi radio of smartphone devices to collect spectrum measurements, and leverages a frequency translator to extend its sensing range. Such an approach has a much higher sampling rate (40MHz) in the WiFi radio compared to the RTL-SDR (2.4MHz), thus providing a much wider instantaneous bandwidth (40MHz) for detecting wide-band signals. The statistical spectral features exploited by Snoopy can also be beneficial to RTL-SDR for signal detection. Nevertheless, Snoopy is not readily applicable to existing smartphone devices, which do not expose a RF connector from their WiFi radio to connect to our current translator hardware. Despite these tradeoffs, we believe the two platforms can potentially be parecSDR for b3(sp4(concm)-37ranstoge-294(Otans)-202ans)-2677f mobans de differ7(y)-qui(measure)-29a

[illegible]