EnergySniffer: Home Energy Monitoring System using Smart Phones

Mostafa Uddin
Department of Computer Science
Old Dominion University
muddin@cs.odu.edu

Tamer Nadeem
Department of Computer Science
Old Dominion University
nadeem@cs.odu.edu

***Abstract*—Tracking energy consumption for individual operating machines (e.g., home appliance) is a prerequisite for making energy conservation and management efficient. In order to meet the requirement for monitoring home energy consumption, several industries and researchers came up with different solutions. Unfortunately, all these solutions require invasive and expensive installation of sensor devices. Furthermore, many of these solutions can't measure the energy consumption of individual machines. In this paper, we propose and evaluate the feasibility of using smart phones in machines's energy monitoring system. We call our system EnergySniffer in which it exploits various sensors, such as magnetic sensor, light, microphone, temperature, camera, WiFi, in smart phones to build a multi sensing framework. This framework is used to build a unique fingerprint profile for each individual machine. As a proof of concept, we develop a simple sensing framework prototype that utilizes only the microphone sensor on the phone. We call this framework a sound sensing framework. Experimental evaluation on sound sensing framework demonstrate the feasibility of continuously identifying and monitoring individual machine in real-time.**

I. Introduction

Home energy consumption is a great concern to our current time. Several studies [2], [3] have shown that detailed energy monitoring system at houses can built awareness among householders. Researchers and industries have developed such real time energy monitoring systems for household users [4], [5], [6], [7], [8], [9], [10]. However, these systems require additional setup cost and have their own drawbacks. For example devices like TED [6] monitors the whole house energy consumption but does not provide energy consumption for each individual machine. On the other hand, Watts Up [7] can provide energy consumption for each machine but it requires additional inline installation between AC plug and outlets. In this paper, we propose a simple and flexible energy monitoring system using smart phones. We call our system EnergySniffer in which it exploits various sensors, such as magnetic sensor, light, microphone, temperature, camera, WiFi, in smart phones to detect and monitor operating machines in its vicinity. In the rest of this paper, we use the term "machine" to refer to any type of machine at home including home appliances, computing machines, non-computing machines etc. The advantages of EnergySniffer system can be summarized as follow: First, it monitors energy consumption for each individual machine. Second, it has very low overhead and also no new hardware is needed to install or maintain. Third, very flexible in updating software and deploying new services using the application updating feature of the smart phones' application markets.

Using the sensors in smart phones to monitor the energy consumption by machines is an eccentric way to approach the problem. Our final objective is to fuse the data from multiple sensors in phone to build a multi sensing framework to generate a unique fingerprint profile for each machine. Later, we apply a machine learning method using fingerprint profiles to recognize and monitor operating machines. In addition to that, this system will also communicate with the Energy Profile of the identified machine to finally calculate the actual energy consumption.

In this paper, we develop a simple sensing framework prototype, called sound sensing framework that utilizes only the microphone sensor of the smart phone. As an initial step of the sound sensing framework, we collect a raw sound data for each individual machine. This sound data is used to build a sound profile to detect and identify the machine later. In our experiments, we evaluate our sound sensing framework in detecting and identifying operating machines in real time. We conclude with a discussion on several implementation challenges as well as the current on going work to efficiently monitor home energy consumption.

We summarize the contributions of this paper as follow:

- Introduce a cheap and flexible system based on multi sensing framework to monitor energy consumption of each individual machine.
- Implement a sound sensing framework in smart phone as a prototype that uses the microphone sensor of the phone to detect the running machines in real-time.
- Evaluate our prototype sound sensing framework in real environments, which shows the feasibility of our system.

The rest of the paper is organized as follows. In section 2, we explain the main components of our "EnergySniffer" system. In the following section, we describe the implementation. We describe the experimental evaluation of the sound sensing framework in section 4. In section 5, we describe limitations and ongoing work with some possible applications. Finally, we describe the related works and conclude in section 6 and section 7 respectively.

II. "Energy Sniffer" System

In this section, we describe the two main components of the EnergySniffer System. The first component is the Energy Profile component that consists of a pre-installed database containing the energy consumption profiles of the individual machines. User can create a new energy profile for a machine that does not exists in the Energy Profile database. The second is the multi Sensing Framework that consists of both Offline Learning and Online Detection phases. Offline Learning is a collection of modules to build a unique fingerprint profile for each individual machine through offline training. The system, through a machine learning algorithm, will utilize the pre-built fingerprint profiles to detect and identify the current operating machines within the Online Detection phase. Once the system detect a machine, it uses the corresponding energy profile from the Energy Profile database to track the energy consumption of this machine. Figure 1 shows the workflow of the EnergySniffer system. In the figure 1, Multi Sensing Framework consists of offline learning system, online detection system and the fingerprint profiles of the machines.

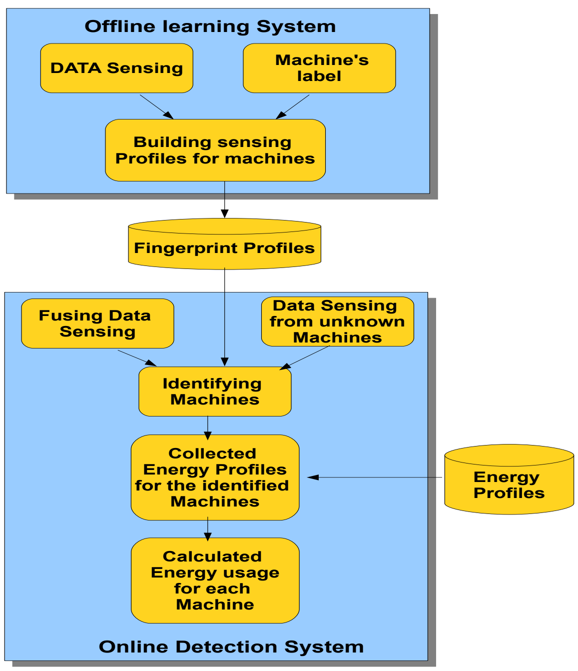

Fig. 1. Workflow of EnergySniffer System

A. *Energy Profile*

Energy Profile is a database of list of machines with their corresponding energy consumption profiles. Since this list is expected to be large when we have many machines, we chose to maintain the energy profile database as a web service instead of storing it locally in phone's storage. When a phone detect and identify a machine, it contacts the web services and downloads the corresponding energy profile of the machine. Furthermore, user has the option to upload a new machine profile to the Energy Profile database. Initially, this database is built from the information provided by the manufacturer. For example, in our experiments we use a microwave oven, which has a energy profile that is collected from the corresponding manufacturer web site [11].

```
<microwave oven>
 <brand>Emerson</brand>
 <model_number>MW8784</model_number>
 <capacity unit="cubic feet">0.7</oven>
  ...
 <energy consumption="W/hr">1050</energy>
</microwave oven>
```

In the Online Detection phase, once the phone matches the collected sensing data with the fingerprint profile of a machine, then its corresponding energy profile from Energy profile database is used to track the energy consumption of the machine. Note that, a machine might have different modes of operation in which each mode of operation will have its corresponding energy consumption profile.

B. *Multi Sensing Framework*

Multi Sensing Framework has two main components: the offline learning component that is responsible to build the fingerprint profile for each individual machine from the sensing data, the online detection component that uses these fingerprint profiles to detect and monitor operating machines. IIn building the fingerprint profiles, the characteristics of each machine (e.g., sound characteristics, light characteristics, etc.) are collected and analyzed to identify the unique features and corresponding sensors that could be used to detect and identify the machine. Then, for each machine, the sensing profile is built from the collected sensing data of the identified sensors. These sensing data in collective way represent the features that construct the sensing profile of the machine. For example, major electrical machines at our home show some distinguish characteristics. Such observations help us to utilize the sensors to identify the machine. In other words, the multi sensing framework builds a fingerprint profile for each machine using different sensors reading. Sensors such as RF, magnetic, light sensor, temperature, sound etc. can be utilized and fused together to detect an operating machine [12], [10], [9]. In a fingerprint, some sensor data might not be relevant for some machines. For example, vacuum machine has no light sensing data. On the other hand, some machines might have multiple sensing types in building their profiles. For example, microwave oven have both sound and radio frequency data that could be sensed and utilized in detecting it. Fusing the multiple sensing data of each machine reduces the ambiguity of identifying the operating machine [13]. Moreover, multiple sensing data of a machine can also be utilized to identify the multiple operational modes of that machine [9].

In summary, building a multi sensing framework has several challenges. First, while creating a fingerprint profile for a machine from the sensing data, we need to study, how the sensing data behaves under different operational modes of the machine. Inferring from that, we need to select the sensing data that are the most sensitive to the characteristics of the machine. Second, how to fuse the selected sensing data to build the profile of a machine for better detection process, is another challenge. Finally, most of the sensing data are highly prone to the environment noise, which make it hard to build a robust fingerprint profile of a machine.

III. SOUND SENSING FRAMEWORK

In this section, we discuss the Sound sensing framework that utilizes only the microphone sensor of the smart phone. In this framework, initially we build a sound profile for each individual machine, then we apply a offline training to build the fingerprint profile of the machine. The fingerprint profile represents an acoustic model for each individual machine that is build from the collected audio data from microphone sensor of the smart phone. For creating these models, initially, the system will collect the raw audio samples at lower sampling rate for a fixed period of time. The feature extraction subcomponent of the system uses this sampling data to extract the relevant features. In the next step, collected features are used in training for building a probabilistic model for each machine. In general, this model represents the fingerprint profile of the machine. In this implementation, we use a supervised learning technique for building the acoustic models.

A. Acoustic Feature Extraction

This component extracts the acoustic features from the raw audio data through a feature extraction procedure. This feature extraction procedure has three goals. First, the collected features should reflect the distinctive acoustic nature among the machines. Second, the effect of any transient background noise on the features should be minimized. Finally, the contextual orientation and the position of the smart phone needs to be considered while extracting these features. It is well known that most of the machines at home contains a motor that shows a distinctive characteristic between the frequency band 0Hz to 1KHz [10]. However, for our feature collection procedure we have considered the frequency range from 0Hz to 5500Hz, which is half of the sampling rate. In speech and audio processing, MFCC [14], [15] is the mostly used feature for audio classification. Therefore, in our system we use the MFCC features for models generation and classification.

Most of the machines have ambient sounds, which show more consistent characteristic compared to human voice sound. Therefore, in framing the sampling audio data, we use the larger frame length compared to the usual frame length of 20ms-40ms. We consider a fixed size of raw audio data as a Frame. In feature extraction, we applied the Hamming window, Fast Fourier Transform (FFT), triangular filter bank and Inverse FFT (IFFT) consecutively over the collected raw audio data. In our empirical experiments, we see that machines at home work in the lower frequency range between 0Hz and 2000Hz. Moreover, most of the machines show distinguish characteristic in lower frequency range. Therefore, in designing the triangular filter bank, we use more filters with high weight values at lower frequency. Finally, we apply IFFT to filter bank output to get the final features in cepstrum features, where cepstrum is the inverse Fourier transform of the log-magnitude Fourier spectrum.

B. Model generation

In offline training phase, we use the extracted features from a frame of raw sound data to generate the probabilistic model for each machine. In model generation, we use a supervised machine learning algorithm to generate the multivariate Gaussian model for each machine. Each model is represented by a multivariate Gaussian function $\mathcal{N}(\vec{x},\vec{\mu},\Sigma)$ with mean $\vec{\mu}$ and variance Σ parameters. For simplicity, Σ is considered as a symmetric matrix. In other words, we assume that all the acoustic features of a frame are independent. However, $\vec{\mu}$ and Σ are calculated from the collected features using the Maximum-Likelihood(ML) algorithm as follow:

$$\begin{aligned} \vec{\mu}_{ML} &= \frac{1}{N}\sum_{i=1}^{N} x_i \\ \Sigma &= \frac{1}{1-N}\sum_{i=1}^{N}(x_i - \vec{\mu}_{ML})^2 \end{aligned}$$

Usually, a machine has several modes of operation. For instance, In our experimental example, the washing machine has multiple states of operation such as water-filling and washing states. Obviously, each machine consumes different energy for each different operational mode [9]. Fortunately, each operation mode has its own unique sound. In order to reflect the different acoustic natures of a machine corresponding to the different operational modes, we use two types of probabilistic models for each machine. One type represents the acoustic model of the machine , while the other type represents the acoustic model of an operational mode of the machine.

In this paper, we refer to the first model as the state independent model. The *state independent* model is built from the collected features of all the operating modes of the machine. The other model is called the *state dependent* model. The state dependent model is generated from the collected features of a particular operating mode of the machine. We use these two model types in our machine recognition component described next.

C. Machine Recognition

In machine recognition, we use the collected features, $\vec{f}$ (feature vector) from a frame of testing sound data to calculate the likelihood values from the Multivariate Gaussian Model of each machine. We label a frame to the machine, which shows maximum likelihood values for the feature vector of that frame as follows,

$$F^l = \arg\max_m \mathcal{N}(\vec{f}, \vec{\mu_m}, \Sigma_m). \quad (1)$$

where F^l is the frame label, $\vec{f}$ is the feature vector of a frame and $\vec{\mu_m}$,Σ_m are the mean and variance of the Multivariate Gaussian Model of the machine m. Finally, we label the window, which consists of multiple number of sequential non-overlapping frames. The maximum occurrence of certain frame's label in a window is the label for a window. In other words, in a window, if M is the machine that was detected maximum number of times as a frame label, then that window will be labeled as M.

$$\begin{aligned} w &= [F_1, F_2, ..., F_n] \\ w^l &= \arg\max_m K(F^l = m) \end{aligned}$$

where w is a window consists of n sequential frames from F_1 to F_n. w^l is the label of the window and $K(F^l = m)$ is a function that provides the number of frame that is labeled by machine m.

Within the recognition phase, our approach initially identifies the machine using state independent model of the machine. After identifying the machine, we use the state dependent models of a machine to recognize the different operational modes of that machine. In the experimental section in this paper, we evaluate our approach in regard of identifying multiple operation mode of a machine.

D. Implementation

In our prototype implementation, we implement the sound sensing framework on Android phones. The implemented framework prototype is responsible to generate the acoustic models for the different machines, as well as recognizing the operating machines from the recorded raw audio data. Our prototype implementation is about 900 lines of Java code using in android development platform. Furthermore, we deploy our prototype in Nexus S phones, which has 1GHz A8 processor with 16GB flash memory. In our prototype, we implement both the feature extraction and the machine recognition in the Android platform. The complete prototype application is 72Kb in binaries.

In the feature extraction component, we implement the Fast Fourier Transform (FFT), Hamming window, Inverse FFT and weighted filter bank on the Android phones. We record the raw data using the 11025Hz sampling rate with 8 bit PCM encoding through the Android phone audio SDK API [16]. In forward FFT, we use 256 frequency bins while we use 32 frequency bins for the reverse FFT. Moreover, we consider 1024 samples (=92-93ms)of raw sound data as a Frame and 10 Frames as a Window (=1s). In case of a weighted filter bank, we use 20 linear filters in frequency band of 0 - 1000 Hz and 12 logarithmic filter of 1000-5500Hz frequency band. The reason behind this large number of filters at lower frequency is that, most of the machines show distinctive characteristic at lower frequency. In implementation, we continuously collected PCM formatted audio data from the microphone and put the data in a buffer of 1024 samples (1 frame). After that, a thread is called to make further signal processing on the buffered data. Finally we get cepstrum features from sampling raw sound data. These features are used for further classification.

In generating the model, we collected 2 seconds of data (features) and then apply Maximum-Likelihood algorithm to generate Multivariate Gaussian Distribution model for the machine. In the smart phone implementation, we assume that each machine has one mode. Even if a machine has multiple operational modes, we label each operational mode as a different machine. Moreover, during experiments, we build the probabilistic model of the machines in different days than the testing day, in order to make time invariant evaluation.

In the prototype implementation, we use equal prior bayesian classifier for classifying and labeling each frame. After labeling the frames, we label each window based on the maximum occurred label in he frames of the window. We also measure the power consumption of our prototype application and compare it with the case when the phone is idle and not detecting machines. In the figure 2, we see our prototype application consumes on avg. 1200mW of energy that is less then the average energy consumption while receiving WiFi data(1700mW). However, most of the energy consumption of our prototype application take places due to the continuous high computation of FFT and IFFT. Such energy consumption can be reduced by increasing the time window size of FFT. As a future work we like to focus on how to reduce the energy consumption of our prototype application.

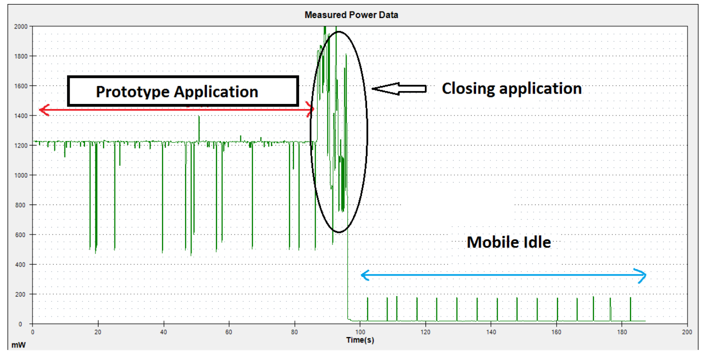

Fig. 2. Mosoon Power monitoring output while our prototype application is operating and when the mobile is idle.

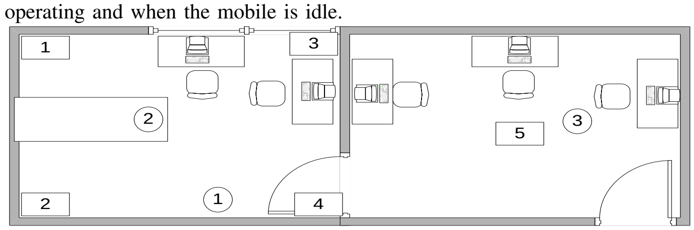

Fig. 3. Circle 1, 2, 3 shows the position of a microwave, a fan and a vacuum cleaner respectively at our lab space. Square 1, 2, 3, 4 and 5 show the position where we have detected and identified the current operating machine using our prototype application.

IV. Experiments and Evaluation

In this section, we describe three sets of experiments to evaluate the feasibility of identifying and tracking the operating of machines using our implementation of the sound sensing framework that is described in the previous section. In the first set of experiments, we evaluate how correctly we can identify the frames of a window. In the second set, we conduct a real-time evaluation of our implemented system by continuously tracking the machine operation for a certain period of time. Finally, in the third set of experiments, we evaluate the accuracy in detecting the different operational modes of an operating machine.

First Set of Experiments: In this experiment, we build a unique fingerprint profile for three different machines; a microwave, a table fan and a vacuum cleaner. In building these fingerprint profiles (i.e, acoustic models), we collected the acoustic raw data in a close vicinity to the machine. Figure 3 shows the layout of our lab space and also the position of each machine in the lab. As shown, our lab consists of two adjacent rooms each of size 20 feet by 10 feet. In the experiment, we run a single machine at a time and use our implemented system in Nexus S phone to detect and identify the operating machine at five different locations for two orientation(horizontal and vertical) of the phone. During running our experiments, we keep the door between the two rooms of our lab open. In this set of experiments, we collect at each testing position using our prototype application a number of acoustic frames corresponding to the operating machine in a 2 sec window. For each frame, our application labels this frame with the corresponding detected machine. Table I shows the percentage of the detection accuracy at each position for each different machine. The shown percentages are averaged over four different runs for the four different orientations of the phone.

In table I, ”none” is a sound profile that we use when there is no operating machine. From table I we see that, while

Machine	Position	microwave	fan	vacuum	none
microwave oven	1	**64.06%**	23.43%	3.12%	9.37%
	2	**71.87%**	9.37%	0.00%	18.75%
	3	**65.62%**	28.12%	0.00%	6.25%
	4	**60.93%**	20.31%	0.00%	18.75%
	5	**53.12%**	12.5%	0.00%	34.37%
Fan	1	32.81%	**67.19%**	0.00%	0.00%
	2	42.19%	**54.68%**	0.00%	3.12%
	3	43.75%	**56.25%**	0.00%	0.00%
	4	32.81%	**57.81%**	0.00%	9.37%
	5	**48.44%**	29.69%	0.00%	21.87%
Vacuum	1	3.12%	0.00%	**96.88%**	0.00%
	2	0.00%	0.00%	**100.00%**	0.00%
	3	15.66%	0.00%	**84.34%**	0.00%
	4	0.00%	0.00%	**100.00%**	0.00%
	5	0.00%	0.00%	**100.00%**	0.00%

TABLE I
RESULT FROM FIRST EXPERIMENT.

the microwave machine is operating, our prototype application manages to label most of the frames at all the five positions correctly to the microwave machine. In case of the fan, our system labels most of the corresponding frames correctly at all the position except position 5. The distance from the fan to position 5 is relatively larger than to the other positions. Moreover, the wall between position 5 and the fan hinders the sound of the fan. Therefore, creating multiple acoustic fingerprints of a machine at multiple different distances will be an interesting direction to explore. On the other hand, in case of the vacuum cleaner is operating, our application identifies almost all the frames correctly as shown in Table I. This is because the sound intensity of the vacuum cleaner is relatively much higher than the microwave and the fan.

Second Set of Experiments : In the setup of these experiments, we use the same set of machines(microwave, fan and vacuum cleaner). We use the same setup shown in Figure 3 with one exception in which we placed the Android phone on the center of the table shown at the left of the west room. During this experiment, we run our prototype application in Nexus S phone for 105 minutes. The prototype application continuously senses surrounding sound to identify any operating machine in real-time and write down the label of the detected machine for each window (1second) in a file. In this experiment, our application consumes 15% of battery charges in 105 minutes.

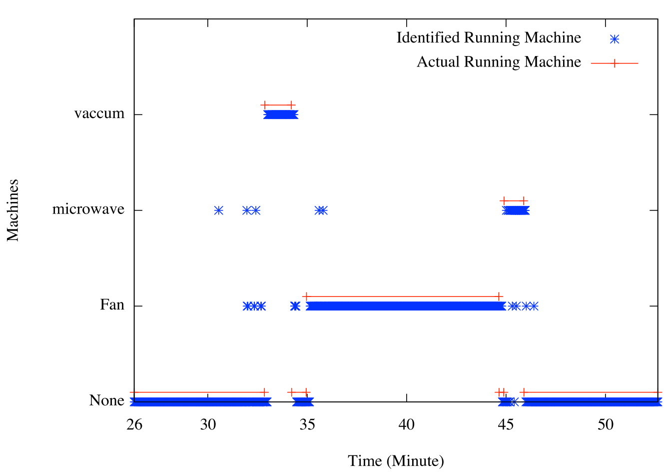

Fig. 4. The recognized machines by our prototype and the actual operating machines during the 25 min period

During the 105 minutes we run the microwave 4 times, the fan 3 times and the vacuum cleaner 2 times. In this setting, no two operating machines overlap. Without loss of generality and for visualization purpose, we show our experiments results for only the time period between 25 minute and 50 minute in Figure 4. In the figure, it is noticeable that some of the assigned labels are incorrect and considered as outliers. For example, while the fan is operating some of the labels are detected as microwave, and vice versa. Fortunately, the experiments show that these outliers happen in separately. Therefore, these outliers could be easily removed using the smoothing techniques.

Third Set of Experiments: In this experimental setup, we use two machines, the fan and a dishwasher. We use two operational modes for the dishwasher (water filling and washing) and three modes for the fan (slow speed, medium speed and fast speed). First, we create two state independent models for the two machines, the fan and the dishwasher. Second, we create three state dependent models for the three individual operational modes of the fan, and the two state dependent models for the two individual operational modes of the dishwasher. Finally, we test our models with audio data collected at different days. Initially, we identify the machine using the state independent model, and after that the operational mode is recognized using the state dependent model. Table II shows the number ofthe labeled frames for the 100 frames collected for each operational mode of the two machines.

Machine,Mode	Fan1	Fan2	Fan3	DF	DW
Fan slow speed (Fan1)	59	8	9	9	15
Fan medium speed (Fan2)	19	65	6	8	2
Fan fast speed(Fan3)	9	25	59	3	4
Dishwasher waterfilling (DF)	0	1	1	96	2
Dishwasher washing (DW)	3	0	4	4	89

TABLE II
CONFUSION MATRIX FOR IDENTIFYING DIFFERENT OPERATIONAL MODES FOR THE MACHINES FAN AND DISHWASHER.

V. CHALLENGES AND FUTURE WORK

Our evaluation of detecting a single operating machine using the sound sensing frame work is promising. However, the real world problem of energy monitoring is far more challenging. Some of the key challenges include the detection of multiple operating machines at the same time, the accurate identification schemes that is invariant to the environmental noise, and the ability to detect the operating machines at different positions. In order to make these challenges more addressable, we could assume that we know the layout and the positions of the machines as well as the smart phone.

Nowadays, a smart phone has potential number of sensors that can have a lot of implications in our real life. However, in our study we found that some sensors are limited in their functionality. For example, we observed that the magnetic sensor chip in Nexus S phone uses a very narrow bandwidth low pass filter on the magnetic sensor reading. The reason behind this behavior is that the magnetic sensor is less sensitive to high frequency changing in the magnetic field. Moreover, the microphone sensor at different devices and platforms shows different sensitivity to the sound. In extending our work, we like to understand more about the limitation, sensitivity and characteristic of different types of sensors in different smart phones in order to create a suitable fingerprint for the machines.

Some machines have several operational modes. In each operational mode the machine consumes different amount of energy [9]. It is a challenging task to detect the different operational modes of a machine to have higher granularity of energy consumptions. Intuitively, fusing different sensing profiles of the machines help to identify different operational modes of the machine. For example, in the case of a refrigerator machine, when the compressor is on, its sound intensity gets more prominent in comparison to the state when the compressor is off [9]. In our implemented sound sensing framework we use a naive probabilistic model to detect the machines. Our simple probabilistic model increases the number of misclassification with the increase in the number of machines. In future work, we will investigate how to build a more accurate and sophisticate model to detect and identify the machines more accurately.

In summary, our on going work on EnergySniffer project is based on the above challenges and presumption include: (1) extensive experiment on using smart phones location in addition with layout information of the machines, to detect multiple machines, (2) leveraging multiple smart phones with wireless communication for further evaluation of our system, (3) interfacing additional or external sensors with the smart phone to create sophisticated fingerprints for the machine.

A. *Possible Applications*

The machines detection and monitoring has implication for context-aware application, home automation, energy monitoring, machine health monitoring, human activity detection, etc. For example, a machine detection system can let the user know about the machines that are operating in result user's vicinity. The system can warn the user if a machine is operating while it should not be or vice versa. Moreover, the machine monitoring system can also be applied to detect the machine's malfunction nature.

VI. Related Works

Non-Intrusive Load Monitoring(NILM) is one of the state of art work in monitoring home energy consumption. NILM is based on the idea that, each individual operating machine generates a distinctive signature on the power distribution system of the building. In [13], the authors use several additional environmental sensors like light intensity, temperature, acceleration and sound level with the NILM system to enhance the signature of the appliances. In their work, they relate the power distribution event with the environmental sensing data to extract the relevant appliance-related information from the sensors. Similarly, in [4] a single point sensor is attached with the power distribution system at home to detect any electric events. In [9], ViridiScope is a power monitoring system for individual appliances at home, which uses magnetic, acoustic and light sensor to compute the consuming energy of the appliances. The VirdiScope system collects the sensing reading by putting sensor devices near to the appliance. On the other hand, EnergySniffer system collect the sensing data using smart phone sensors. In [12], the authors use radio frequency to identify non-wifi devices like, microwave oven, video camera, cordless phone etc. Inferring from the article [12], radio frequency can be a potential way to identify a operating machine.

Microphone sensor of a smart phone is becoming popular in current research activity [18][19][20]. In [18], SoundSense provides a general purpose sound sensing framework for resource limited smart phones. The main design goal of SoundSense is to maintain a scalable architecture of the system to detect large number of sound events for individual users. In our sound sensing framework, we use the sound profile as a fingerprint to detect the machines. In the article [19], the authors use the acoustic background sound as a fingerprint to identify a room or indoor location. SurroundSense [20] is an another mobile application to detect the logical location of the user using smart phone sensors. In [20], the authors use the sound in addition with other sensor to create a fingerprint for different logical location. In our best knowledge, EnergySniffer is the first proposed smart phone sensing application for monitoring home energy consumption. Some existing system like TinyEARS [10] provides a individual machine level power consumption details using acoustic signature of the machine. TinyEARS uses audio sensors in each room to identify the currently operating machines. In addition to that, it connects with the power meters at home to detect the real-time changes in power usage to make final correlation about which machine is using how much electric energy. Although,TinyEARS provides a solution for monitoring the energy consumption of the home appliances, but it need to setup audio sensor in each room and to manage those sensors to recognize the machines.

VII. Conclusions

In this paper, we propose the EnergySniffer system, that uses the smart phones sensing capability to monitor the energy consumption of the machines at home . The main purpose of our system is to provide a simple and flexible home energy monitoring system. Compared to other existing home energy monitoring system, our system is simple to deploy and easily adaptable to any new machine. Finally, we implement a sound sensing framework in the smart phones as a prototype, which only uses the microphone sensor to monitor the energy consumption of the machines at home. Experimental evaluation of our prototype system shows a good accuracy of detecting and monitoring machines in real environment, which indicates the potential of EnergySniffer system.

References

[1] S. Keshav and C. Rosenberg, "How internet concepts and technologies can help green and smarten the electrical grid," in *Green Networking 2010*, August 2010, new Delhi, India.

[2] T. Hargreaves, M. Nye, and J. Burgess, "Making energy visible: A qualitative field study of how householders interact with feedback from smart energy monitoring," *Journal of Energy Policy*, vol. 38, pp. 6111–6119, October 2010.

[3] S. Darby, "The effectiveness of feedback on energy consumption: a review for defra of the literature on metering, billing and direct displays," http://www.eci.ox.ac.uk/research/energy/electric-metering.php, 2002.

[4] S. N. Patel, T. Robertson, J. A. Kientz, M. S. Reynolds, and G. D. Abowd, "At the flick of a switch: Detecting and classifying unique electrical events on the residential power line," in *Ubicomp*, 2007, pp. 271–288.

[5] "Energy monitoring technologies inc." 2011, http://www.energymonitor.com.

[6] "Ted: The energy detective web site." 2011, http://www.theenergydetective.com.

[7] "Watts up? pro web site." 2011, http://www.wattsupmeter.com/.

[8] X. Jiang, S. Dawson-Haggerty, P.Dutta, and D. Culler, "Design and implementation of a high-fidelity ac metering network," in *ACM IPSN/SPOTS 2009*, 2009.

[9] Y. Kim, T. Schmid, Z. M. Charbiwala, , and M. B. Srivastava, "Viridiscope: Design and implementation of a fine grained power monitoring system for homes," in *Ubicomp 2009*, Sep 30- Oct 3 2009, orlando, Florida, USA.

[10] Z. C. Taysi, M. A. Guvensan, and T. Melodia, "Tinyears: Spying on house appliances with audio sensor nodes," in *Buildsys 2010*, November 2 2010, zurich, Switzerland.
[11] "Emerson radio corporation web site." 2011, http://www.emersonradio.com.
[12] S. Rayanchu, A. Patro, and S. Banerjee, "Airshark: Detecting non-wifi rf devices using commodity wifi hardware," in *IMC'11*, November 2-4 2011, berlin, Germany.
[13] M. Berges, L. Soibelman, and H. S. Matthews, "Leveraging data from environmental sensor to enhance electrical load disaggregation algorithms," in *Proceeding of the 13th International Conference on Computing in Civil and Building Engineering*, 2010, nottingham, UK.
[14] F. Zheng, G. Zhang, and Z. Song, "Comparison of different implementation of mfcc," *Journal of Computer science and Technology*, vol. 16, pp. 582–589, 2002.
[15] J. Wang, Jhing-Fa, and Y. S. Weng, "Chip design of mfcc extraction for speech recognition," *Integr. VLSI*, vol. 32, pp. 111–131, 2002.
[16] "Android sdk web site." 2011, http://developer.android.com/sdk/index.html.
[17] O. Ardakanian, S. Keshav, and C. Rosenberg, "Markovian models for home electricity consumption," in *GreenNet'11*, August 19 2011, toronto, Ontario, Canada.
[18] H. Lu, W. Pan, N. D. Lane, T. Choudhury, and A. T. Campbell, "Soundsense: Scalable sound sensing for people-centric application on mobile phones," in *MobiSys'09*, June 22-25 2009.
[19] S. P. tarzia, P. A. Dinda, R. P. Dick, and G. Memik, "Indoor localization without infrastructure using the acoustic background spectrum," in *MobiSys'11*, June 28-Jul 1 2011, bethesda, Maryland, USA.
[20] M. Azizyan, I. Constandache, and R. R. Choudhury, "Surroundsense: Mobile phone localization via ambient fingerprint," in *MobiCom'09*, September 20-25 2009, beijing, China.