HotMobile 2012 Poster: MachineSense: Detecting and Monitoring Active Machines using Smart Phone

Mostafa Uddin
muddin@cs.odu.edu

Tamer Nadeem
nadeem@cs.odu.edu

CS Department, Old Dominion University, Norfolk, VA, USA

I. Introduction

Detecting and tracking individual running machine at home has variety of implication at context-aware application for home automation, energy monitoring, machine health monitoring, human activity detection, etc. Researchers have came up with various ideas of detecting machines or machine related events based on their interest of problems. Unfortunately, all these solutions require invasive and expensive installation of sensor devices. In order to address these problems, we propose a simple and flexible machine monitoring system using smart phones. We call our system MachineSense in which it exploits various sensors in smart phones to build a unique fingerprint profile for each individual machine. In building fingerprint profile, each machine's characteristics are analyzed to identify the sensors (such as magnetic sensor, light, microphone, temperature, camera, RF etc.) that could be utilized to collect sensing data. These sensing data in collective way represent a fingerprint profile of the machine. Later, we apply a machine learning method using fingerprint profile for recognizing running machine. In this poster, we refer to all kind of home appliances, computing machines and non-computing machines as ”machine”.

II. Sound sensing framework

Towards proof of concept, we implement a Sound sensing framework that utilizes only the microphone sensor of the phone. In this framework, we use only sound profile for each machine as a fingerprint profile. This sound profile represents an acoustic model for each individual machine. This model is build from the collected acoustic features by applying audio signal processing over the collected raw audio data. For creating these models, initially, the system generate MFCC features from the raw audio samples for a fixed period of frame. 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 prototype implementation, we use equal prior bayesian classifier for classifying and labeling each frame. After labeling the frame, we label each window based on maximum occurred label for frames in a window. In our implementation, we consider 1024 samples (=92-93ms) of audio data as a Frame and 10 Frames as a Window (=1s).

II.A. Acoustic Feature Extraction

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 collecting raw audio data we use 11025Hz sampling rate and 8 bit PCM encoding in Android phone Nexus S. In our observation, we saw that, machines at home generate sound in lower frequency range from 0Hz to 1000Hz. Moreover, most of the machine shows distinguish characteristic in lower frequency range. Therefore, in designing the filter bank, we use more high number of filters at lower frequency.

II.B. Model generation

In model generation, we use 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 parameters mean, $\vec{\mu}$ and variance Σ. For simplicity Σ is considered as symmetric matrix. In other words, we assume that all acoustic features of a frame are independent. However, $\vec{\mu}$ and Σ are calculated from collected features using Maximum-Likelihood(ML) algorithm.

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

II.C. Machine Recognition

Initially, we collect features from each frame. Afterward, we use the features from a frame, $\vec{f}$ (feature vector) to calculate the likelihood value for each machine's Multivariate Gaussian Model. The model of a machine, that provides the maximum likelihood value for a frame represents its label.

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

where F^l is the frame label, $\vec{f}$ is the feature vector of a frame and $\vec{\mu_m}$,Σ_m are mean and variance of a machine m. The maximum occurrence of certain frame's label in a window is the ultimate output label for a window. In other words, in a window, if M is the machine model that occurred maximum number of time as a frame label, then that window will be labeled as M.

$$w = [F_1, F_2, ..., F_n]$$
$$w^l = \arg\max_m K(F^l = m)$$

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

II.D. Implementation and Evaluation

In our prototype experiment, we create sound profiles for microwave oven, table fan, vacuum cleaner. After that we put all three machines in a single room and run our prototype application in Nexus S phone for 105 minutes. The prototype application continuously sense surrounding sound to identify any running machine in real-time and write down the label of identifying machine for each window (1second) in a file. During105 minutes we run the microwave oven 4 times, fan 3 times and vacuum cleaner 2 times. None of the machine runs at the same time. in the figure 1, we show the result of our prototype application running from 25 minutes to 50 minutes for better view. In the figure 1, it is noticeable that, some of the labels of window out lies from the actual label. However, these outliers happen discretely, which can be removed using further smoothing technique over the output of window label. In the figure 1, "none" is a sound profile that we build when none of the machine is running.

III. Challenges and Ongoing Work

Now-a-days, smart phones have potential sensors that can have several implications in our real life. However, in our study we found out that some sensors may be very limited in functionality. For Example, we observe that the magnetic sensor chip in Nexus S phone use a very low pass filtering technique that generates only the DC component of the signal. As a result, magnetic sensor reading is less sensitive to high frequency changes in magnetic field reading. This characteristic make magnetic sensor in smart phone less useful to be utilized in detecting running machines. Also, the microphone sensor at different devices and platforms show different sensitivity to acoustic data reading. In continuation to our work, we like to understand more about the limitation, sensitivity and characteristic of different sensors in different smart phones for creating suitable fingerprint for the machine.

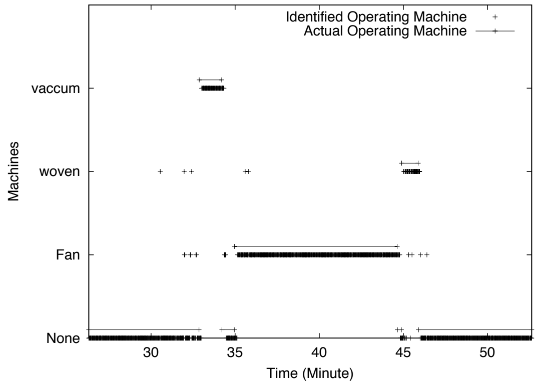

Figure 1: The machines recognized by our prototype system during 25 minutes.

Our evaluation on detecting a single running machine based on sound is promising, however real world problem of detecting machine is far more challenging. Detecting multiple machines at a time, identifying machines regardless of un-relevant background sound effect, recognizing running machines from different positions are some key challenges. In order to make these challenges more addressable, we could make some presumptions such as knowing the layout and positions of the machines as well as the smart phone.

In summary, our ongoing work on MachineSense project based on the above challenges and presumptions 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.