Demo Abstract: Zone-level Occupancy Counting with Existing Infrastructure

Gabe Fierro, Omar Rehmane, Andrew Krioukov, David Culler
Computer Science Department
University of California, Berkeley
gt.fierro@berkeley.edu, orehmane@berkeley.edu,
krioukov@cs.berkeley.edu, culler@cs.berkeley.edu

Abstract

Through accurate and dynamic occupancy detection, building actuation systems can fine tune the targets of their actions to better fit the patterns of usage in modern buildings. We outline a method for achieving this through existing wireless infrastructure and present a demonstration of its viability in a corporate environment.

1 Introduction

Obtaining real-time counts of occupants in each room/zone within a building has been one of the holy grails of building automation. Occupancy counts can be used as an input to a wide range of controls in buildings, drastically improving energy efficiency. For example, ventilation can be set in proportion to occupancy, air conditioning can be disabled or turned down for empty or sparsely populated areas, lights can be unobtrusively turned on/off without waving at a motion detector and plug-loads in empty areas can be monitored or turned off. These techniques can make a building's energy consumption proportional to the number of people in the space, whereas today most buildings operate wastefully in just two static modes: fully occupied or unoccupied. For buildings that are never completely empty, such as labs and graduate student offices, this static regime requires that the building continue to operate at night almost the same as during the day.

The major challenge is to obtain counts of occupants cheaply, reliably, and without requiring any new actions from occupants. Previous efforts have used motion detectors (PIR) with reed switches [1, 5], infrared (IR) beams [8], video cameras [2], RFID tags, thermal imaging and CO_2 sensors [7]. All of these approaches require the deployment of new hardware, hampering scalability, and vary in their accuracy and granularity. Motion detectors can only detect the presence of people, not their number. CO_2 sensors approximate the number of people in an enclosed space, but cannot granularly count people in cubicle areas. IR beams are cheap and accurate but must be placed at all entrances and exits to an area. RFID requires occupants to carry special tags and requires large readers to be placed at all entrances.

Observing that most building occupants on campus regularly carry smartphones and other wireless devices, we explore people-counting using just existing infrastructure. Our guiding design principles are that no new actions should be required of occupants, counts are required with coarse, zone-level granularity (e.g. lighting zones as shown in Figure 1), and deployment time and hardware requirements should be minimized.

We apply a *passive localization* technique used in the wireless networking literature for rogue access point detection [3, 4] to locate and count occupants by their mobile devices. With this technique 802.11 wireless infrastructure is used to overhear packets sent by client devices and to measure the received signal strength (RSSI). Thus, occupants can be counted without a special phone application; we only require that WiFi is enabled on the phone and and automatically associated with any wireless network. On our test Android and iPhone devices this happens automatically even when the phone's screen is turned off and without the user pressing any buttons. Since only zone-level precision is required, we can effectively solve a classification problem (*i.e. which zone is a phone in?*) instead of a general localization problem. This significantly reduces error and allows us to use a localization algorithm that requires no training. Compared to prior work on indoor wireless localization, we focus on coarse-grained and completely passive localization tailored for counting occupants in buildings.

2 Infrastructure

Our prototype infrastructure utilizes wireless access points with omnidirectional MIMO antennas running open-source Linux-based firmware (OpenWRT). In a production setting, our system would be implemented on the existing building access points. In fact, Cisco already offers a form of passive localization for rogue access points that is similar to the approach we use for occupancy counting.

The campus network on which we chose to deploy is a 2.4Ghz wireless network supporting channels 1, 6, and 11. Each of the routers was set to *monitor/promiscuous* mode, and was installed with an open-source packet analyzer. All

Buildsys'12, November 6, 2012, Toronto, ON, Canada.
ACM 978-1-4503-1170-0

routers communicate with a central computer, which mitigates the collection of and computations upon packet data from each of the routers and process the results.

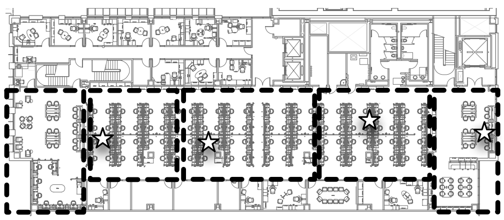

Figure 1. Location of routers shown in relation to actuation zones.

The routers cycle through each of the three wireless channels using tcpdump to capture packets in 1-second increments. Each packet is forwarded along with its 802.11 radiotap link-level header over SSH to a FIFO queue on the central computer, which parses out the RSSI and source MAC address. For each MAC address, for each time-step, for a given sample rate, each RSSI is converted to a linear scale and the resulting RSSIs are averaged per router. We can then compute the possible location (x_d, y_d) of the device possessing a given MAC address as the centroid of the known router coordinates (x_i, y_i) weighted by the average detected signal strength at that router r_i:

$$x_d, y_d = \sum_i \frac{r_i x_i, r_i y_i}{\sum_i r_i} \tag{1}$$

The location of the device is computed as the average of the 5 most closely clustered centroids of the 10 most recent centroids (using Manhattan Distance as a heuristic). In practice, this "averaging of averages" greatly reduces the influence of the deployment environment upon the consistency of the measured RSSIs. Abnormally strong or weak signals do not alter the accuracy of our estimations. There is an incurred lag of about 4 times the sample rate should the device move, but this delay is acceptable within the bounds of accurate occupancy detection. We are able to place devices within 1 cubicle of their actual location – results of a 30 minute localization test are shown in Figure 2. This level of precision is lower than that of trained localization techniques, but it is sufficient for counting people in HVAC or lighting zones which typically span 6-8 cubicles.

The passive nature of our localization infrastructure means that it requires little, if any, effort on behalf of the occupants to function. It is only necessary for a device to associate with an available network before we can start tracking it. We can obtain a session IP address for a device with a given MAC address by listening to all wireless traffic. In the absence of regular or sufficient traffic, the central computer can occasionally ping the device in question to either elicit the requisite wireless traffic or determine if the device has left the tracking area (which suggests the user has left). This approach differs from previous wireless localization work in that it does not *require* the user to ensure he/she is creating enough traffic to be localized accurately, but rather unobtrusively generates the necessary data.

This raises the issue of accounting for "detached devices," devices left connected to the network that are not actively being used and can therefore be falsely registered as occupants. The problem of detached devices can be addressed through analyzing packet and localization data. Network activity can be used as a proxy for human usage [6]. For example, high-frequency HTTP traffic indicates likely human activity. Similarly, devices that move regularly are likely to indicate human presence. Conversely, devices that are always on and in a fixed location are likely to be desktops and are filtered out.

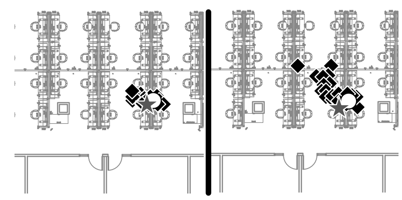

Figure 2. Increased accuracy with increased sample rates (20s, left; 10s, right). Even at times of low ambient signal noise, weighted centroid measurements can be inconsistent (black diamond), a problem remedied by averaging the 5 most closely clustered centroids (white circle). The star represents the actual device location.

3 Demonstration

We plan to demonstrate an example of deploying our occupancy detection infrastructure on a "corporate network" consisting of a few routers. Arbitrary users will be able to associate with the network using either a personal device or a provided WiFi-enabled smartphone, simulating a user entering a campus building or corporate office. Once associated with the network, a set of routers in *monitor* mode running the infrastructure and connected to a central computer will be able to track RSSIs for each device and provide occupancy counts for each of a small number of predefined zones; the counts will be displayed on a screen in real-time. Then, using a set of wireless motes, we will actuate lamps based on the occupancy counts for each area as an example application of the occupancy infrastructure.

4 References

[1] AGARWAL, Y., BALAJI, B., GUPTA, R., LYLES, J., WEI, M., AND WENG, T. Occupancy-driven energy management for smart building automation. In *BuildSys'10* (2010).

[2] ERICKSON, V. L., CARREIRA-PERPIN, M. ., AND E.CERPA, A. OBSERVE: occupancy-based system for efficient reduction of hvac energy. In *IPSN'11* (2011).

[3] FARIA, D. B., AND CHERITON, D. R. Detecting identity-based attacks in wireless networks using signalprints. In *WiSe '06* (2006).

[4] LAURENDEAU, C., AND BARBEAU, M. Centroid localization of uncooperative nodes in wireless networks using a relative span weighting method. *EURASIP 2010* (Apr. 2010), 6:1–6:10.

[5] LU, J., SOOKOOR, T., SRINIVASAN, V., GAO, G., HOLBEN, B., STANKOVIC, J., FIELD, E., AND WHITEHOUSE, K. The smart thermostat: using occupancy sensors to save energy in homes. In *SenSys* (2010).

[6] NEDEVSCHI, S., CHANDRASHEKAR, J., LIU, J., NORDMAN, B., RATNASAMY, S., AND TAFT, N. Skilled in the art of being idle: reducing energy waste in networked systems. In *NSDI'09* (2009).

[7] SAND, JAMES R. Demand-controlled ventilation using co2 sensors, 2004.

[8] SUN9. Sun9 Occupancy Counting Switch, 2012.