FixtureFinder: Discovering the Existence of Electrical and Water Fixtures

Vijay Srinivasan*
Samsung Research America
75 W. Plumeria Dr
San Jose, CA, USA
v.srinivasan@sisa.samsung.com

John Stankovic and Kamin Whitehouse
Dept of Computer Science
University of Virginia
Charlottesville, VA, USA
{stankovic, whitehouse}@cs.virginia.edu

ABSTRACT

The monitoring of electrical and water fixtures in the home is being applied for a variety of "smart home" applications, such as recognizing activities of daily living (ADLs) or conserving energy or water usage. Fixture monitoring techniques generally fall into two categories: fixture recognition and fixture disaggregation. However, existing techniques require users to explicitly identify each individual fixture, either by placing a sensor on it or by manually creating training data for it. In this paper, we present a new *fixture discovery* system that automatically infers the existence of electrical and water fixtures in the home. We call the system *FixtureFinder*. The basic idea is to use *data fusion* between the smart meters and other sensors or infrastructure already in the home, such as the home security or automation system, and to find repeating patterns in the fused data stream. To evaluate FixtureFinder, we deployed the system into 4 different homes for 7-10 days of data collection. Our results show that FixtureFinder is able to identify and differentiate major light and water fixtures in less than 10 days, including multiple copies of light bulbs and sinks that have identical power/water profiles.

Categories and Subject Descriptors

C.3 [**Special-Purpose and Application-Based Systems**]: Real-time and Embedded Systems

Keywords

Fixture Discovery, Smart Homes, Data Fusion, Smart Meters, Disaggregation

1. INTRODUCTION

Over the past several decades, several new technologies have emerged to monitor the use of electrical and water fixtures in the home. This information is being applied for a variety of "smart home" applications, such as recognizing activities of daily living (ADLs) or conserving energy or water usage.

Fixture monitoring techniques generally fall into two categories. The first category includes *fixture recognition* techniques, that identify when a particular fixture is turned on or off. For example, the ElectriSense [10] and HydroSense [7] systems attach a sensor to a wall socket or hose bib to monitor high frequency noise in the voltage and water pressure, respectively. The user trains the system by manually turning fixtures on or off so that the system can learn each fixture's noise profile. Then, the system automatically recognizes those fixtures each time they are used in the future. The second category includes *fixture disaggregation* techniques, that identify how much energy or water is used by each individual fixture. For example, the Viridiscope [17] and NAWMS [16] systems attach a sensor to each electrical or water fixture to recognize when they are used, and also use a smart meter on the electrical or water mains to monitor aggregate energy/water usage in the entire house. Based on the assumption that the total energy/water usage of the home is equal to the sum of energy/water usage of individual fixtures, these systems learn the quantity of energy or water used by the individual fixtures. However, all of these techniques have one key limitation: they require an initialization phase where the user must first identify the individual fixtures, either by manually creating training data for each fixture, or by placing a sensor on each fixture.

We present a new *fixture discovery* system that automatically infers the existence of electrical and water fixtures in the home. Many major appliances are distinctive enough to be discovered through the smart meter data alone, due to their periodicity (e.g. refrigerators and space heaters [3]) or a predictable progression through multiple modes of operation (e.g. dishwashers and washing machines [22]). However, small, simple fixtures tend to be much less distinctive and can easily be buried in the noise of other, simultaneous fixtures. Furthermore, power/water meter data alone cannot distinguish between multiple, identical fixtures in a home such as 60W light bulbs or bathroom sinks. To address this problem, we use *data fusion* between the smart meters and other sensors or infrastructure already in the home, such as the home security or home automation system. Then, we search for repeating patterns in this fused data stream to uncover the existence of small fixtures, and to differentiate between multiple, identical fixtures. For example, sink activity may be highly correlated with motion sensor data from the bathroom and the home water meter data; light

*This work was performed while the author was at the University of Virginia.

IPSN'13, April 8–11, 2013, Philadelphia, Pennsylvania, USA.

fixture activity in the living room may be correlated with the home power meter data and light sensor data from the living room. The basic insight behind our approach is that every fixture has a distinctive profile in the home, even if it is not distinctive in the smart meter or ambient sensor data alone. By fusing multiple data streams, we are able to uncover *multi-modal fixture profiles* that are not apparent in the smart meter or ambient sensor data alone.

In this paper, we present *FixtureFinder*, a prototype of a complete fixture discovery system. Our current prototype focuses on light fixtures, sinks, and toilets. These fixtures are among the most difficult to discover because of their non-distinctive power/water profiles, and because numerous copies of each are likely to be found in most homes. Thus, our demonstration on these fixtures provides a proof-of-concept for the basic principles presented in this paper. Our prototype performs data fusion of smart meters with typical security and home automation sensors, since such sensors are already deployed in over 32 million homes in the US [21]. By combining commercial home sensors with the smart power and water meters that are already widely deployed by utility companies [1], FixtureFinder can enable fixture recognition and disaggregation across millions of homes, without requiring an initialization phase where a person manually identifies the individual fixtures.

FixtureFinder uses two aspects of the motion sensors typically found in home security and automation systems: the detection of infrared activity and the detection of ambient light levels (many motion detectors contain sensors in the visible light spectrum for calibration purposes). Using this data, FixtureFinder performs four steps to recognize light and water fixtures: (1) it detects all rising edges and falling edges in the four data streams: power, water, motion, and light; (2) it fuses the data streams by identifying pairs of co-temporal edges in different data streams; (3) it uses clustering algorithms to recognize repeating patterns of multi-modal pairs, and uses Bayesian matching to select only those rising/falling edges that have a matching falling/rising edge with the same multi-modal profile; (4) it clusters all discovered ON-OFF events based on their multi-modal profile to discover the unique fixtures in a home. Unlike fixture recognition, the goal of fixture discovery is not to recognize every ON/OFF event but rather to select only those events that are very likely not to be caused by spurious noise. Once a set of such events is recognized, it can be used to create a training set for existing fixture recognition or disaggregation systems such as ElectriSense and Viridiscope. In this paper, we show how the principles and algorithms behind FixtureFinder can also be applied for fixture recognition and disaggregation. To our knowledge, this is the first system that can automatically discover the presence of small, simple fixtures; that can perform fixture recognition; and that can disaggregate power and water usage, all in a completely unsupervised fashion.

To evaluate FixtureFinder, we deployed between 25-40 sensors in 4 different homes for 7-10 days of data collection. The sensors included a whole-house power meter and water meter, a motion and light sensor in every room, and ground-truth sensors on light and water fixtures in the home. Using this data, we demonstrate that FixtureFinder is able to discover all water fixtures and 37 out of 41 light fixtures monitored in less than 10 days, including multiple copies of light bulbs and sinks that have identical power/water profiles. The 4 light fixtures that were not detected were specialized task lighting such as under-cabinet lights that were not heavily used. FixtureFinder was able to recognize fixture usage events and disaggregate water and electrical usage with about 90% accuracy.

2. RELATED WORK

Fixture monitoring generally falls into two categories: fixture recognition and fixture disaggregation. Some existing systems in these two categories are built upon principles that could in theory also be used for fixture discovery, but to date this potential has not been fully explored for any existing system. Below, we describe examples of both types of systems, touching on their potential to discover fixtures where appropriate.

2.1 Fixture Recognition

Fixture recognition systems identify when a particular fixture is turned on or off. Perhaps the most well-known approach for fixture recognition is *non-intrusive appliance load monitoring (NIALM)* [11], which can be used to recognize the usage of electrical fixtures in the home based on their power signatures. These signatures can be extracted using only a single power meter, which is already available in many homes today [1]. Some appliances have a unique profile of real and reactive power, while other appliances such as washing machines and dishwashers exhibit characteristic electrical patterns over time. When a home contains multiple similar appliances, however, NIALM techniques cannot identify *which* appliance is being used. Furthermore, due to low power state transitions from complex appliances such as the television or the HVAC system, NIALM techniques are not effective for small fixtures such as electric lights that exhibit constant, low power values [11]. In theory, NIALM signatures can be used to automatically discover major appliances [22]. However, this approach would require a large database with the electrical signatures of every appliance ever manufactured. Despite over two decades of discussion, no such database has been created and, despite some recent home energy data sets [2, 18], to our knowledge there is no current effort to create one. Furthermore, the number of manufactured appliances is enormous, and comparing a noisy electrical signal against a large space of electrical signatures is likely to produce spurious detections of appliances that do not exist. Further investigation into the potential of this approach is needed.

Approaches similar to NIALM have also been developed to recognize water fixtures. For example, one system uses flow signatures, such as flow rate, flow duration and, in the case of washing machines and dishwashers, patterns of flow to identify types of fixtures and appliances [20]. Another system uses patterns in the presence or absence of flow in both water pipes and drain pipes [6], as detected by microphones installed in the basement. These systems achieve high accuracy in recognizing high consumption appliances, but low accuracy in differentiating between different instances of the same fixture category such as different instances of identical sinks, toilets, or showers in the same home.

Recently, new solutions have been developed to recognize fixtures based on noise profiles in the power or water lines. For example, two systems use an easy-to-install, plug-in sensor that leverages unique high frequency EMI(Electromagnetic Interference) signals on the power line to recognize electrical fixtures. The first system called Flick-of-a-switch [24] can recognize mechanically switched appliances while the second system called ElectriSense [10] can detect fixtures that use switched-mode power supplies (SMPS) such as low-voltage electronics or CFL bulbs. Similarly, HydroSense [8] samples a water pressure sensor at 500Hz from anywhere in the piping system, such as a hose bib outside the home. The system recognizes water fixtures based on the "water hammer" signature caused when a fixture is turned on or off. These three approaches can differentiate between multiple, identical fixtures such as light bulbs, sinks, and toilets. However, they all require users to manually train the system by labeling ON and OFF events for every single fixture. If fixtures are added or moved throughout the house, training must be performed again. The ElectriSense signatures may be persistent across houses, and the authors suggest that a large database of such signatures would be sufficient for fixture discovery. However, this assertion must still be fully explored, and is subject to many of the same challenges as the NIALM approach discussed above.

2.2 Fixture Disaggregation

Fixture disaggregation systems identify how much energy or water is used by each individual fixture. The simplest approach to measure power and water usage at individual fixtures is to place a sensor on each one [13]. This approach is sometimes called *direct sensing*, and requires sensors that can directly be integrated into pipes or electrical wiring, which can be expensive to install in terms of both hardware and installation time. Alternatively, *indirect sensing* approaches fuse data from smart meters with sensors placed on or near each appliance. For example, Viridiscope [17] uses one specialized sensor node per appliance to measure light, acoustic, and/or magnetic field changes, and tries to correlate the intensity of the indirect measurement with the amount of power used by the appliance. This is performed through a global calibration process based on the assumption that all fixtures are instrumented, and that the whole house power demand is equal to the sum of the individual fixtures. Viridiscope makes a firm assumption that each fixture being monitored is paired with a different sensor: if any sensor's magnetic field, acoustic, or light values are strongly influenced by more than one fixture, the system would treat them as a single fixture and would produce an incorrect calibration function. FixtureFinder provides the mathematical machinery (in the form of Bayesian clustering and matching algorithms) to help Viridiscope dissociate two or more fixtures detected by the same sensor, and to combine a single fixture that is being detected by multiple sensors.

The principles underlying Viridiscope have also been demonstrated for water systems, where accelerometers were used on the pipes to measure vibration and a centralized water meter was used on the water mains [16]. Several other, similar techniques have been proposed for power or water disaggregation [9, 14, 26]. However, these techniques all require a single sensor per fixture, and sometimes even additional sensors internal to the power and water infrastructure. The goal of FixtureFinder is to automatically discover fixtures based on sensors and infrastructure that is already present in the home, such as security systems or home automation sensors, that were not deployed with the explicit purpose of fixture disaggregation.

FixtureFinder builds on earlier results in water disaggregation by the authors, that were published at a workshop [27]. This paper focuses on the fixture discovery aspects of that work, and generalizes the solution to include both electrical and water fixtures.

In theory, fixture recognition and fixture disaggregation techniques could be used in combination. For example, the user could install an ElectriSense sensor to measure noise on the electrical lines. After a manual training process, the ElectriSense system could recognize when each appliance is used, and the Viridiscope system could use that information to disaggregate their individual energy usage levels. FixtureFinder is also expected to work in cooperation with systems like these, in order to avoid the need to explicitly add sensors or manually create training data for each fixture.

3. APPROACH OVERVIEW

The goal of the FixtureFinder algorithm is to combine smart meters with in-home sensors to form a fused data stream, and to discover frequently repeating patterns within that stream. For example, it will detect when a 5 liter/minute water flow repeatedly co-occurs with activity in a particular motion sensor. We call these patterns multi-modal fixture profiles, because they represent the signature of a fixture's usage as viewed by multiple sensor types simultaneously. FixtureFinder's mathematical machinery is based on two underlying insights: 1) the usage of a fixture often has a repeating signature in multiple different sensor streams simultaneously, and 2) the ON and OFF events of a fixture come in pairs. Additional states other than ON and OFF could be incorporated into Step III without loss of generality, but FixtureFinder does not yet address multi-state fixtures. The FixtureFinder algorithm has four Steps. In Step I, it uses edge detection to compute a sequence of timestamped *rising and falling edges* in each data stream. In Step II, data streams are fused by finding events in multiple streams that frequently co-occur in time, and combining them to creating *edge pairs*. This fusion step eliminates spurious edges that exist in only one stream, but are not observed in the other data streams as expected. In Step III, the edge pairs are matched in rising/falling sequences called *usage events*, and all edge pairs that do not successfully match are discarded. This matching process eliminates additional spurious edges that do not correspond to a fixture ON or OFF event. In Step IV, the usage events are clustered into groups that have similar multi-modal profiles. These clusters represent the fixtures that have been discovered.

Algorithm 1 shows the mathematical formalism behind the four main steps of the FixtureFinder algorithm. The variables shown in bold indicate the output of each step. The final output of Step IV is thus the **Fixture Set**. The inputs to the algorithm are two sensor streams S^i, S^j from sources i and j respectively; however, Steps II, III, and IV could be extended in the future to handle more than two sensor streams. We expand on algorithm 1 below and explain the intuition behind each step in detail.

Step I: Event Detection: For every sensor i in the system that produces a time series $S^i = s^i_1, s^i_2, s^i_3, ..., s^i_t$, the first step performed by FixtureFinder is to apply an edge detection algorithm on S^i to produce a set of **Edges** E^i. Lines 1:2 in algorithm 1 show Step I. Each edge $e^i = (m^i, t^i) \in E^i$ is an ordered pair, where m^i is the magnitude of the edge and t^i is the timestamp of the edge. If $m^i > 0$ then e^i is

Algorithm 1 Steps in the FixtureFinder algorithm

Inputs: Sensor streams S^i, S^j

Event Detection:

1: **Edges** $E^i = [M^i \ T^i] \leftarrow$ EdgeDetect(S^i)
2: **Edges** $E^j = [M^j \ T^j] \leftarrow$ EdgeDetect(S^j)

Data Fusion:

3: **Edge Pairs** $P^{i,j} = \{\}$
4: **for all** edge pairs (e^i, e^j), where $(e^i = (m^i, t^i) \in E^i) \wedge (e^j = (m^j, t^j) \in E^j)$ **do**
5:    **if** $(|t^i - t^j| < T \wedge (m^i * m^j) > 0)$ **then**
6:       Add (e^i, e^j) to $P^{i,j}$
7:    **end if**
8: **end for**

Matching:

9: Edge Cluster $C^i \leftarrow$ Cluster(M^i)
10: Edge Cluster $C^j \leftarrow$ Cluster(M^j)
11: **for all** edge pairs $p = (e^i, e^j) \in P^{i,j}$ **do**
12:    Edge pair probability $P(p) = \sum_{c^i \in C^i} \sum_{c^j \in C^j} P(e^i|c^i) * P(e^j|c^j) * P(c^i|c^j) * P(c^j)$
13: **end for**
14: **for all** (p_x, p_y), where $(p_x = (e^i_x, e^j_x) \in P^{i,j}) \wedge (p_y = (e^i_y, e^j_y) \in P^{i,j})$ **do**
15:    Pair match probability $P(p_x, p_y) = (\sum_{c^i \in C^i} P(e^i_x|c^i) * P(-e^i_y|c^i)) * (\sum_{c^j \in C^j} P(e^j_x|c^j) * P(-e^j_y|c^j))$
16:    Pair match weight $W(p_x, p_y) = -log(P(p_x) * P(p_y) * P(p_x, p_y))$
17: **end for**
18: **Usage Events** $F^{i,j} = [E^i_{on} \ E^i_{off} \ E^j_{on} \ E^j_{off}] \leftarrow$ Min Cost Bipartite Matching(W)

Fixture Discovery:

19: **Fixture set** $\leftarrow$ Cluster($[M^i_{on} \ M^j_{on}]$)
20: **return Fixture Set**

a *rising edge*, otherwise it is a *falling edge*. Every type of sensor will produce a stream with a different structure, and will therefore require a different type of edge detection algorithm with parameters set appropriately by the user. In sections 3.1 and 3.2, we discuss the particular edge detection approaches used for each sensor type.

Step II: Data Fusion: FixtureFinder creates a set of **Edge Pairs** $P^{i,j}$ by combining every pair of edges that are generated by the two sensor modalities, that are both rising or falling edges, and that co-occur in time. Lines 3:8 in algorithm 1 show Step II of the FixtureFinder algorithm. More specifically, for every pair of edges $e^i = (m^i, t^i)$ and $e^j = (m^j,^j)$ from the two streams S^i and S^j respectively, FixtureFinder creates a new *edge pair* $p = (e^i, e^j)$ if m^i and m^j have the same sign (i.e. both rising or both falling) and $|t^i - t^j| < T$, where T is a time windowing parameter that defines how close a pair of edges must be. Any edge that does not co-occur with an edge from another data stream is not used to create an edge pair, and is therefore not considered further.

The value of T depends on the sampling rate of the sensors, and possibly on any time synchronization errors between the sensors. Due to a noisy data stream S^j or simultaneous human activity in the home, we frequently pair a single edge e^i from one stream S^i to multiple noisy edges e^j from S^j that occur within time window T; in Step III, we eliminate noisy edge pairs by only retaining frequently occurring edge pairs that can be successfully matched to edge pairs of the opposite polarity.

Step III: Matching: Once all edge pairs are created, FixtureFinder matches rising edge pairs p_x with falling edge pairs p_y in order to create multi-modal fixture **Usage Events** $F^{i,j}$. Lines 9:18 in algorithm 1 show Step III of the FixtureFinder algorithm. First (in lines 9:17), we compute a weight function $W(p_x, p_y)$ between any two edge pairs p_x and p_y from set $P^{i,j}$; the weight function is designed to be very low if the two edge pairs are highly likely to be from a single fixture's ON-OFF event. A min-cost, bipartite matching algorithm [19] is then used to match edge pairs based on W, and each edge pair can only be matched once; any unmatched edge pairs are thus eliminated as noise.

We compute the weight function $W(p_x, p_y)$ (line 16) for the matching algorithm based on two probability functions: (i) the probabilities $P(p_x)$ and $P(p_y)$ that the individual edge pairs p_x and p_y are created as a result of frequent fixture usage and not as a result of noise, and (ii) the probability $P(p_x, p_y)$ that both edge pairs are from the same fixture's ON-OFF event. To compute the edge pair probabilities (i) and (ii), we first cluster the edges E^i from each sensor i based only on their magnitudes M^i to a set of clusters C^i, as seen in lines 9:10. We use a soft clustering algorithm to obtain the probabilities $P(e^i|c^i)$ that any edge $e^i \in E^i$ belongs to any cluster $c^i \in C^i$. The intuition is that each fixture generates a unique edge cluster combination (c^i, c^j) in two data streams.

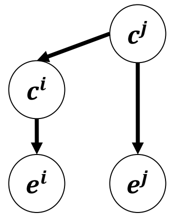


Figure 1: Bayesian network used to compute $P(e^i, e^j)$ for edges e^i and e^j from sensors i and j respectively.

Probability (i) is then computed using the equation in line 13, which follows from our simple Bayesian network formulation shown in figure 1. Observed edges e^i and e^j are dependent on hidden clusters c^i and c^j respectively; clusters c^i and c^j co-occur whenever the underlying fixture generating these two clusters of events is triggered. In line 13, $P(c^j)$ is the fraction of events in cluster c^j relative to other clusters, and $P(c^i|c^j)$ is the fraction of events in c^j that are paired with events in c^i. In line 15, we compute probability (ii) as the probability that both edge pairs are from the same edge cluster creating the rising edge pair.

Before the matching algorithm is executed in line 18, certain matches are eliminated by setting their weight to zero, including

- any match where the rising edge occurs after the falling edge.
- any match where, before the falling edge occurs, the total power or water usage drops below the magnitude of the rising edge.

This last condition is designed to avoid matches where, for example, it appears that a sink is turned on and off 2 minutes apart, but during that interval, the total water flow actually dropped to zero. After the matching process is complete, any unmatched edge pair is not included in a usage event and is therefore no longer considered. The set of matched fixture usage events $F^{i,j}$ thus consists of multi-modal rising (E^i_{on}, E^j_{on}) and falling edges (E^i_{off}, E^j_{off}) from both sensor streams S^i and S^j.

Step IV: Fixture Discovery: Once all usage events $F^{i,j}$ are created, FixtureFinder clusters them based on the multi-modal profile, i.e. the magnitudes of rising edge values from the two fused data streams ($[M^i_{on}\ M^j_{on}]$). Step 19 shows the fixture discovery step; every cluster produced represents a discovered fixture, and the usage events associated with that cluster represent instances when that fixture was used. It is important to note that the fixtures discovered in Step IV represent only those fixtures with a multi-modal profile in sensor streams S^i and S^j, such as the whole house power meter and the light sensor in the living room; to discover fixtures in other rooms such as the kitchen, we run the fixture finder algorithm on other pairs of sensor streams, such as the whole house power meter and a light sensor from the kitchen.

3.1 Case Study: Light Fixture Discovery

In the first case study, FixtureFinder combines the whole-house smart power meter with ambient light sensors to discover light fixtures, infer their nominal wattage values and usage times. We considered other sensor pairings such as the smart power meter and ambient motion sensors, but found that the false positive rate in this pairing was too high to accurately identify light fixture events; in the future, we plan to include more than two sensor streams including motion and other sensors to improve accuracy.

We assume that each room or area in the home has one light sensor. When we observe light edges in multiple light sensors simultaneously with the same polarity, we only retain the light edge with maximum intensity, since that edge is most likely from the sensor in the same room as the light fixture. Finally, we independently apply the FixtureFinder algorithm 1 on each light sensor paired with the whole-house power meter to discover the individual light fixtures in each room.

Figure 2 shows an example of this process using real data traces from a whole house power meter and a bedroom light sensor, on typical day from 6AM to 4PM. The top box shows the two true light fixture ON-OFF events observed during the time period shown. The power meter data (right side) and bedroom light sensor data (left side) are both used to generate edges. On the left side, we show the light edges created by a simple window-based edge detection algorithm and also the effect of applying additional filters (explained in detail below) to eliminate spurious edges caused due to human movement or natural light changes. In both the light sensor and power meter data, a large number of spurious edges are difficult to differentiate from the true fixture usage events shown on top. By fusing the two data streams and looking for matched ON/OFF events, two true light usage events are discovered (top).

To perform Step I, we choose edge detection algorithms based on the characteristics of the power and light data. Certain major appliances such as HVAC are very salient in the power trace (right side of figure), but small fixtures such as light bulbs are buried in the noise. Furthermore, many low power noisy events occur within 2-3 seconds of each other while a few light fixtures take more than a second for a full edge transition. These two cases can be difficult to differentiate due to the 1Hz sampling frequency of the TED power meter. For this reason, we must use a very aggressive edge detection algorithm that finds all edges besides very low intensity edges (noise), or edges that rise too slowly (most likely because they are aggregates of smaller edges). Specifically, we apply a custom sliding window technique that detects all power edges with at least a minimum intensity dP and a maximum time window bound $maxwinP$ set to 5 seconds.

We use the same sliding window technique to find edges in the ambient light data. Most changes due to natural lighting result in gradual changes, and are eliminated using our window-based edge detection algorithm which only looks for high intensity edges within a short time window. As seen on the left side of the figure, the window-based edge detection still produces a very large amount of false edges due to shadows and partial cloudiness. We use two key insights to differentiate artificial lighting from the noise natural lighting:

- Lights are not turned on and off very quickly.
- Lights are not turned on and off repeatedly for long periods of time.

Thus, any highly frequent or very rapid or very gradual edges are filtered.

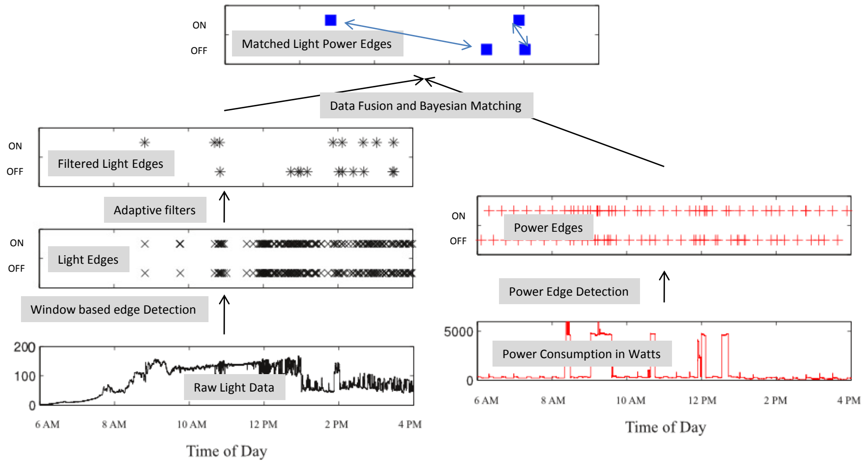


Figure 2: These data traces are from a bedroom light sensor and whole house power meter from 6AM to 4PM in House 2. FixtureFinder eliminates false positive light and power edges by performing data fusion and matching.

In Step II, we chose a parameter value $T = 12 seconds$ for the data fusion to account for time synchronization errors and delays in observing the power edges in the off-the-shelf power meter (caused due to transmission delays on the noisy power line infrastructure).

In Step III we chose Quality Threshold Clustering [12] as our clustering algorithm to generate the edge clusters from each sensor stream. The advantage of Quality threshold clustering is that it allows us to control the *maximum difference* between any two edges in a given edge cluster; this allows us to use prior knowledge about the typical error distribution of light fixture edges in a home. Based on empirical experiments, instead of using a fixed maximum difference, we modified the quality threshold algorithm to use a maximum *relative difference* (difference between two edges as a proportion of the smaller of the two edges); we used a maximum relative difference of 0.25 in our deployments. Because Quality Threshold clustering is a hard clustering approach, we estimate cluster membership probabilities of edges $p(e^i|c^i)$ by fitting a normal distribution to each edge cluster as an approximation. In Step IV, we again use the Quality Threshold Clustering algorithm with a maximum relative difference of 0.25 to generate our final set of fixtures from the multi-modal fixture usage events.

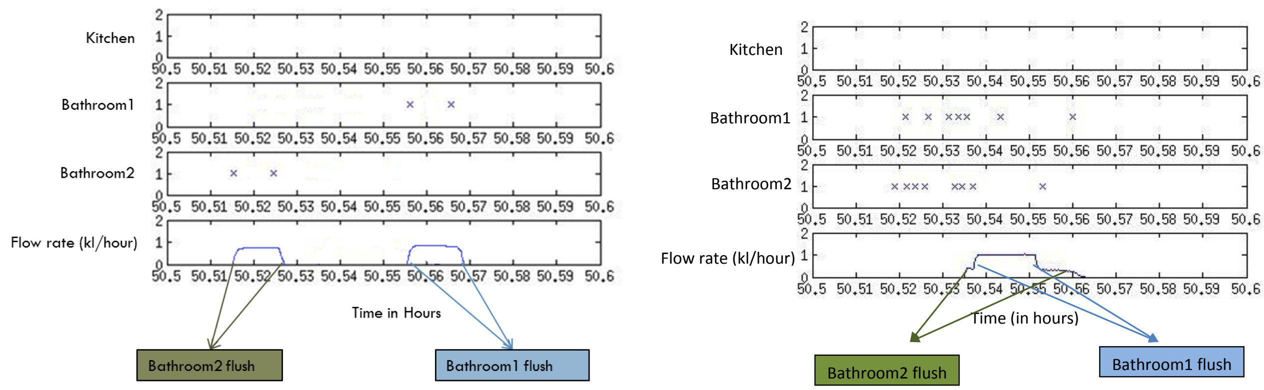
3.2 Case Study: Water Fixture Discovery

In the second case study, FixtureFinder combines the smart water meter with motion sensors to discover water fixtures, and to infer their water flow and usage times. We did not consider other pairings such as the water meter with the light sensor, since in some homes, light fixtures may only be used a small fraction of the time that water fixtures are used; in the future, we plan to include these additional sensor streams to improve accuracy.

Figure 3(a) shows two examples of simultaneous flush events in different bathrooms of house 1 and 2, respectively, plotted together with the motion sensor data from the both bathrooms and the kitchen in each home. House 1 has two identical toilets, but FixtureFinder is able to use the motion sensor signatures to differentiate the two flow events as originating from different bathrooms. House 2 has different models of toilet, each with different flow rates of approximately 0.3 kl/hour and 0.6kl/hour. Thus, even when they occur at essentially the same time, as shown in the figure, it can also differentiate them based on the difference in their flow rates. Notice that FixtureFinder must still use the motion sensor signatures to differentiate these flow events from, for example, a dishwasher fill cycle. This is particularly true because the flow rates change when the events co-occur, due to limited water flow. In the case of simultaneous flush events in House 1 where the flow rates are identical, FixtureFinder would not necessarily be able to associate the events with the correct fixture.

We assume a single motion sensor in each room containing a water fixture. Our off-the-shelf X10 motion sensors represent a challenge since the sensors already filter and aggregate the passive infrared data and send binary event messages whenever motion is detected. In contrast to light fixture discovery, we consider all the ambient motion sensors as a single multi-dimensional sensor stream to be paired with the water meter data stream using the FixtureFinder algorithm 1; the reason is that water fixture events may generate motion signatures spanning multiple binary motion sensors.

In Step I, for the water meter, we use the Canny edge detection algorithm [4]. To perform Step I on the motion sensor stream, we simply generate a multi-dimensional distance vector D corresponding to each water edge, that contains temporal distances between each water edge and the closest binary motion event from each motion sensor considered. In Steps III and IV we use the Quality Threshold Clustering [12] algorithm with a maximum relative difference of 0.25 to generate edge clusters from the water flow data. For the motion sensor data, we simply define a fixed set of clusters R corresponding the rooms in which the motion sensors are deployed. Given a room $r \in R$ with a motion sensor, and a water edge cluster c, we can evaluate the Bayesian network shown in Figure 1 after we compute $P(D|r)$ and $P(r|c)$; we do not discuss these probability computations here due to



(a) House 1: Double Flush

(b) House 2: Double Flush

Figure 3: These data traces show aggregate water flow and motion sensor data for two simultaneous flush events, in different homes. FixtureFinder can use both the motion sensor data and the flow rates to distinguish between two otherwise similar fixtures, even when they are used simultaneously.

space constraints, but more details on this Bayesian network evaluation can be found in our earlier work [27]. Thus, each fixture computed in Step IV represent a fixture with a unique water flow signature and a unique room assignment; given that the various fixtures in a single room typically have unique water flow signatures, our FixtureFinder approach is likely to find the set of all water fixtures in the home.

Additionally, we use simple heuristics to label low flow matched events (less than 0.3 kl/hour) as sinks and higher flow matched events as toilet flushes. As we increase the number of types of fixtures, more sophisticated classification schemes for fixture type will be necessary, and we expect to leverage established work in this area [22].

4. EXPERIMENTAL SETUP

To evaluate the FixtureFinder system, we deploy sensors in four multi-resident homes for 10 days each. All four homes had multiple residents, multiple bathrooms, and a wide array of light and water fixtures. Details of the deployments are summarized in Table 1. In contrast to many existing fixture studies that use bench top testing or controlled testing in homes, we performed our evaluation *in-situ*: the data traces were collected over the course of multiple days while people lived normal lives in the home. Naturally, other fixtures and appliances aside from those being measured were also used during the experiment period. This in-situ evaluation setup ensures that FixtureFinder is able to operate even in the presence of real-world noise and signals commonly present in households. For example, the homes operated complex appliances such as dishwashers and washing machines, as well as central HVAC, space heaters, microwaves, and in some cases plasma televisions. Plasma televisions present a particularly challenging noise problem because, unlike LCD screens, the power usage of a plasma television changes with the brightness of the scene, creating high-magnitude and rapidly-changing noise patterns on the electrical lines. Most of the rooms in the homes had windows, which introduced a light variability in the light readings that typically far exceeded any changes in light value caused by artificial lights. This reveals one major difference with the Viridiscope system, which intentionally put the light sensors very close to the light and used a simple threshold value to detect whether the lights were turned on or off. In contrast, FixtureFinder is opportunistically using light sensors that were deployed to detect ambient light levels, which includes natural lighting and can include multiple light fixtures. Our in-situ study also captured the natural simultaneous usage of water fixtures, one artifact of which is a change in flow rate: when the toilet is flushed and the sink is turned on, the aggregate water flow is typically lower than the sum of the two fixtures individually. This effect is often more extreme in homes with older piping infrastructure that has limited water flow, and is commonly known to cause burning hot showers whenever a toilet is flushed. Finally, our in-situ deployments also encountered a diverse range of light fixtures including incandescent bulbs, CFLs, and halogen lights, and both dimmable and non-dimmable switches with different light intensities and wattages, as seen in figure 6. Due to all of these complex and natural sources of noise, variety, and interference, *in-situ* testing is an important part of our empirical validation. To our knowledge, this is the first study to perform in-situ evaluation of a fixture monitoring system across a range of multiple, diverse homes. This testing was enabled by the specific sensing and ground truth system that we developed, as described below, which can be considered one of the contributions of this paper.

Sensors: To execute FixtureFinder in these homes, we deployed a single water flow sensor on the water mains and a single power meter on the power mains. To measure aggregate water flow, we used the Shenitech Ultrasonic water flow meter that clamps on to the outside of the water mains pipe. It uses the Doppler effect to measure the velocity and resulting flow of water through the pipeline. The flow meter reports instantaneous water flow (in cubic meters per hour) at a frequency of 2Hz using the home's Wi-Fi connection to transmit data. We expect that utility water flow meters being deployed in a large scale in homes will have a similar setup. To measure aggregate power usage, we used the The Energy Detective (TED) 5000 power meter, which uses a clamp-on ammeter that measures total current drawn by the home appliances. The power meter reports instantaneous power (both real and reactive) at a rate of approximately 1Hz. Figure 4.a shows the installation of the smart power

Home#	Type	#Residents	#Rooms	#Lights	#Sinks	#Toilets
1	3 Story house	3	8	12	-	-
2	3 Bedroom student housing	3	6	6	-	-
3	1 Bedroom condo	2	8	9	3	2
4	2 Story house	4	9	14	3	2

Table 1: Our deployments involved four homes with multiple residents, and a variety of lights and water fixtures. Due to cost limitations, smart water meters were only installed in two of the four homes.

Figure 4: Our deployments included 25-40 sensors per home, including (a) a smart power and water meter (b) a motion and light sensor in every room. For experimental ground truth, we also deployed (c) Z-wave light switches, Z-wave plug load meters, and Z-wave contact switches on the water fixtures.

and water meter in one of our home deployments. Because the Shenitech water meters are much more expensive than the TED power meters ($2000 vs. $200 each), we deployed a water meter in only two of the homes, and water meter data was collected for only 7 days in each home. An example of a home deployment is illustrated in Figure 5.

In addition to the smart meters, FixtureFinder requires data from other sensors or infrastructure in the home, such as a security or home automation system. Off-the-shelf motion sensors for security and automation typically measure both motion and light. Since our test homes did not have a pre-existing home automation system, we deployed one motion sensor and light sensor per room to emulate a typical home automation system. Although one sensor per room may be redundant for some home security systems, we believe that smart homes of the future will contain at least one occupancy sensor per room to support diverse applications such as medical activity monitoring or home energy.

In our deployments, we used off the shelf X10 motion sensors inside rooms to detect occupancy, as shown in Figure 4.a. These sensors are inexpensive ($5 each) and can be installed with double-sided tape. The X10 motion sensors send a binary ON message whenever motion is seen with a minimal damping interval of 7 seconds between ON messages. In general, we installed one motion sensor per room in a prominent location with good visibility over the entire room, if possible. In Home 2, one of the motion sensors in a bathroom malfunctioned during our week long deployment, so we used a motion sensor in an adjacent bedroom with a partial view of the bathroom in our analysis. Because of the challenges of accessing ambient light sensors in existing motion sensors, we instead used the cheap Hamamatsu photo diode connected to a telosb mote [25], as shown in Figure 4.a, sampling at approximately 2Hz; in the future, we expect more open, commercial sensors that allow users to access the raw light sensor data similar to our deployments. Our light sensors were installed near the locations where a motion sensor would be installed. Across our four homes, we used one sensor per room in all but 3 rooms, where two sensors were required to achieve coverage of the user's living space.

Ground Truth: To measure ground truth, we instrumented all of the light and water fixtures in each house. To instrument the light switches, we replaced all existing switches in each home with wireless ZWave smart switches, that transmit a wireless message whenever the light switch is turned on or off. For plug-in lamps, we installed a wireless ZWave smart plug load meter that measures the power consumption of the appliance plugged into it. We assume that any non-zero power consumption indicates that the lamp is switched on. For sinks and toilets, we installed Z-Wave door/window sensors (i.e. magnetic reed switches) on the faucet and flush handles. Figure 4.c shows examples of each of the 4 different types of Z-Wave installations used for ground truth. Due to cost and deployment constraints, the ground truth sensors were not installed on all fixtures and appliances that were used during our in-situ study; in particular, showers, and faucets connected to dishwashers

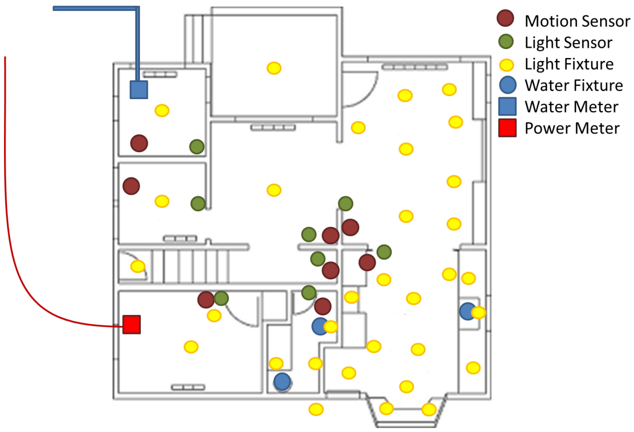


Figure 5: This diagram illustrates one home, including the locations of the meters, sensors, and fixtures. Many of the light fixtures are not used during the deployment period, and are not considered for our analysis of FixtureFinder.

and washing machines, were not instrumented. Our evaluation on water fixtures is thus limited to sinks and toilets. In total, we observed 41 light fixtures being used across the 4 homes over 10 days. We obtained a total of 775 ON-OFF pairs from all the light fixtures; the number of usage events attributed to each light fixture is shown in figure 6. We observed 10 water fixtures across 2 of our test homes over the 7 day deployment period. The 10 water fixtures were used for a total of 424 times; the distribution of usage events across the 10 fixtures is shown in figure 11.

5. RESULTS

For each house, we executed the FixtureFinder system on the data traces from the power meter, water meter, motion sensors and light sensors. The current version of FixtureFinder extracts correlations between the power meter and the light sensors, and between the water meter and the motion sensors. We plan to extend the system to also extract light-water and motion-power correlations, which is only expected to improve performance. The main result of this system is generated by Step IV in the FixtureFinder system, which produces a list of clusters, each with a multi-modal fixture profile. FixtureFinder discovered the top 90% of energy consuming light fixtures in each home, reporting the nominal wattage of each light fixture within $\pm 5W$. The multi-modal profile of each fixture is shown in Figure 6, and includes the nominal wattage as measured by the power meter, the lighting intensity as measured by the light sensors, the name of the light sensor that detected it (typically, a room name), and the number of usage events (ON-OFF pairs) originating from each fixture. FixtureFinder also discovered the sinks and toilets in each house. The multi-modal profile of the water fixtures were generally the same for all sinks and toilets, and are not shown due to lack of space. We observed several anecdotal instances of water fixtures such as the shower or the sprinkler system being discovered by FixtureFinder, but we currently limit our evaluation to only those fixtures monitored by our ground truth sensors. In the future, we intend to expand our evaluation to a larger set of fixtures and appliances.

From the the number of ON-OFF event pairs for each light fixture shown in Figure 6, we observe that the FixtureFinder system is able to discover light fixtures with a wide range of usage counts, ranging from as few as 5-10 events to as high as 50-100 events; this variability in usage counts suggests that FixtureFinder takes very few usage events to stabilize and identify individual fixtures, and stability is not significantly affected as more usage events occur over time. Of the 41 light fixtures that we instrumented, FixtureFinder was not able to discover 4 light fixtures that consumed very little energy, either because they were rarely used or because they had a very low wattage, e.g. LED bulbs. These fixtures include two in House #1 and three in House #4. All four of the undiscovered lights were *task lighting* fixtures, including under-cabinet lighting, coffee-bar lighting, and exit/entrance door lighting; all four fixtures were located in the kitchen, where task lighting is most common in homes. The fixtures were rarely used (1-3 times), had very low wattage, and were not easily detected by ambient light sensors because of their task-oriented nature.

5.1 Bootstrapping a Training Set

In addition to discovering the fixtures, FixtureFinder simultaneously produces a set of usage events that it associates with each fixture, where each event includes ON/OFF times and power or water flow measurements. These events can be used as an automatically created training set for supervised learning systems such as ElectriSense or HydroSense. For example, if FixtureFinder provides 100 usage events that are associated with the bathroom sink, HydroSense can use the ON/OFF times associated with those usage events to learn features of the pressure waves caused by that sink, so that it can recognize it again in the future. Thus, FixtureFinder can serve to bootstrap a training set, effectively converting HydroSense from a supervised learning system into an unsupervised system.

Usage events are associated with each fixture during Step IV of the FixtureFinder algorithm, when they are clustered together based on their multi-modal profile to initially provide support for a fixture's existence. However, not all of these usage events are necessarily caused by the same fixture; some may have been caused by other fixtures or spurious noise, and accidentally had the same multi-modal profile. The degree to which these clusters correctly separate usage events from different fixtures dictates the degree to which FixtureFinder will be useful as an unsupervised bootstrapping technique. We measure this clustering accuracy in terms of two metrics:

Precision: the number of events correctly associated with a fixture, divided by the total number of events associated with that fixture.

Recall is the number of events correctly associated with a fixture, divided by the total number of events generated by that fixture.

For the purposes of bootstrapping a training set, we care more about precision: we do not want spurious events incorrectly associated with a fixture, because they will cause errors in the training set. Recall is not as important: we don't need all usage events of a fixture, only enough to have a sufficiently large training set.

We analyze the trade off between precision and recall by

House 1

ID	# of times used	Room	Power (W)	Light value
1	10	MasterBed	135	23
2	4	Livingroom	35	193
3	13	Livingroom	40	29
4	9	Kitchen	90	20
5	13	MasterBath	20	36
6	6	MidBathroom	10	62
7	8	BottomBath	40	41
8	7	MasterBath	42	0
9	2	BottomBath	40	27
10	2	Kitchen	5	-
11	3	TopRoom	80	923
12	1	Kitchen	95	-

House 2

ID	# of times used	Room	Power (W)	Light value
1	13	Livingroom	185	366
2	20	Bedroom1	90	55
3	15	Kitchen	120	71
4	25	Bedroom2	35	50
5	4	Bedroom3	50	414
6	123	Bathroom	50	260

House 3

ID	# of times used	Room	Power (W)	Light value
1	9	Livingroom	95	155
2	29	Livingroom	115	59
3	71	Bathroom1	220	1187
4	91	Bedroom	95	70
5	53	Kitchen	110	181
6	30	Officeroom	90	55
7	29	Bathroom2	305	1157
8	6	Diningroom	200	182
9	12	Livingroom	55	239
10	-	Diningroom	100	19

House 4

ID	# of times used	Room	Power (W)	Light value
1	11	Frontroom	55	311
2	10	Basement	325	353
3	4	Kitchen	250	60
4	53	Bathroom	395	1028
5	12	Kitchen	280	125
6	12	Livingroom	80	50
7	10	Dining Room	200	110
8	32	Bedroom	95	69
9	4	Bathroom1	95	649
10	6	Nursery	55	129
11	6	Bedroom	60	32
12	2	Kitchen	70	-
13	3	Kitchen	30	-
14	2	Kitchen	110	-

Figure 6: FixtureFinder discovered 37 of the 41 light fixtures that we instrumented, and produces a multi-model profile for each (wattage + light intensity). Four low-power, infrequently used, and specialized task lighting fixtures (shown in gray) in the kitchen were not discovered. There was one false alarm fixture, namely fixture #10 in House 3 (shown in black). The light fixture numbers here match the light fixture numbers in figure 10.

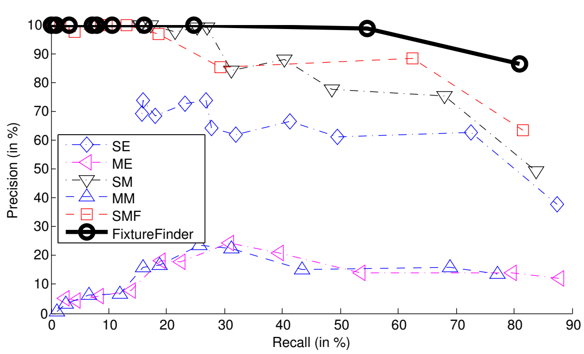


Figure 7: FixtureFinder is able to achieve 98% precision, with recall values of 55%, in order to bootstrap training data sets. Variants of the system illustrate the limited value of edge detection, matching, or fusion alone.

varying the threshold used for edge detection in the Step I of the algorithm: higher thresholds result in fewer event detections (and thus higher precision), while lower thresholds result in more event detections (and thus higher recall). For the sake of brevity, we only present the results for light fixture discovery here, but water fixture discovery presents a similar underlying trade off. The dark, black line in Figure 7 corresponding to FixtureFinder shows the recall/precision that is achieved when varying the edge detection thresholds. In this figure, we show the recall/precision over all light fixtures in the four homes considered. This figure shows that precision at or near 98% can be achieved, for recall values close to 55%. Thus, we conclude that FixtureFinder can produce training data sets with few if any errors. Recall levels of 55% merely indicate that the system will take longer to produce that training set.

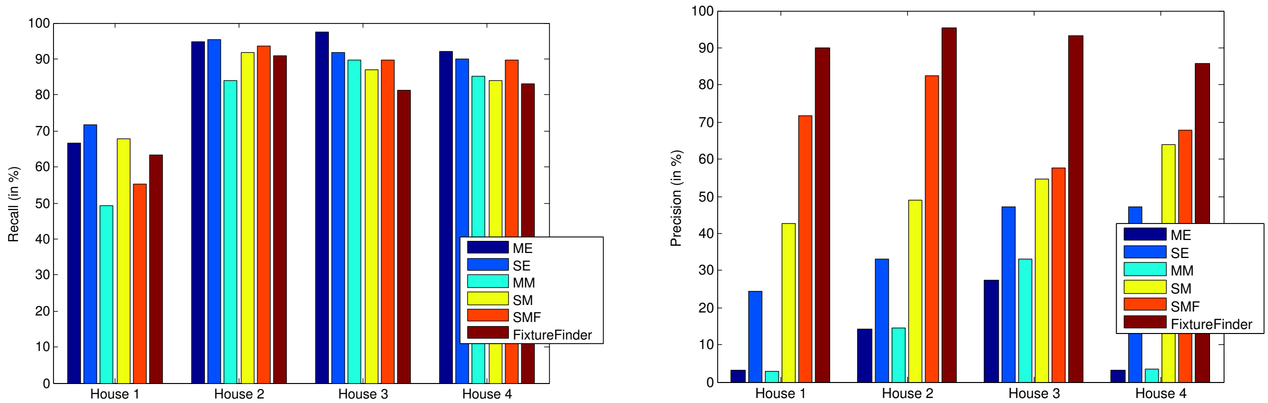
5.2 Analysis

In this section, we explore each of the components of the FixtureFinder algorithm in order to explain the degree to which they contribute to its overall performance. Specifically, we examine (i) edge detection, (ii) matching, and (iii) fusion by creating five variants of FixtureFinder that infer the existence of individual light fixture events based on the following criteria:

- **SE** – Sensor Edge Only: distinct changes in sensor readings only, using our edge detection algorithms
- **ME** – Meter Edge Only: distinct changes in smart meter readings only, using our edge detection algorithms
- **SM** – Sensor Matching: distinct changes in sensor readings that have corresponding ON/OFF matches, using our edge detection and matching algorithms
- **MM** – Meter Matching: distinct changes in power meter readings that have corresponding ON/OFF matches, using our edge detection and matching algorithms
- **SMF** – Sensor/Meter Fusion: changes in sensor/meter readings that occur co-temporally, by applying Steps I and II of the FixtureFinder algorithm

Figure 7 shows the recall/precision across four homes that is achieved by each of these algorithms when varying the edge detection thresholds; we show the results only for light fixtures for the sake of brevity. The results show that smart meter data alone (ME) are not sufficient to identify fixtures with anything greater than 25% precision. This is consistent with previous studies of power meter data [11], due to the power of light fixtures being small compared to noise from other, simultaneous fixtures. Similarly, sensor data alone (SE) is not sufficient because, e.g. artificial lights have low light output compared to natural variations in sunlight and the effects of person movement and other shadows.

When ON/OFF matching is used, many spurious edges are eliminated which allows higher precision for a given recall value. Because of the very large number of edges in



(a) Recall

(b) Precision

Figure 8: FixtureFinder achieves consistently higher precision than variants of the system in all 4 houses. Each house shows variability in the performance of the variant schemes depending on the particular electrical and ambient light noise present in that house.

smart meter data, matching produces little benefit (MM). Matching improves the discovery precision using sensor data (SM) to be above 99%, but only up to recall values of less than 28%. Fusion of the sensor and smart meter data further eliminates edges in either data stream that do not have a co-occurring event in the other data stream (SMF), but because of the very large number of edges in the smart meter data stream, this approach does not perform as well as FixtureFinder. When we combine the fusion, matching, and Bayesian clustering algorithms that eliminate all but frequently occurring sensor/meter pairs, we create the FixtureFinder algorithm that is able maintain precision at 98% for recall values up to 55%.

Figure 8 shows a snapshot of the recall and precision of the five component schemes and FixtureFinder in our four test homes; the snapshot is shown at the same parameter settings used to achieve greater than 80% recall and precision for FixtureFinder in Figure 7. We observe from Figure 8 that the FixtureFinder approach trades off a small reduction in recall for a significant increase in precision. In all four homes, by applying data fusion, matching, and Bayesian clustering, FixtureFinder achieves a precision ranging from 80-90% with only a negligible reduction recall. We observe in general that there is variability in the precision achieved by the 5 component schemes implemented. House 1 had the most noisy light environment due to wall-sized windows on most floors; thus, we observe the worst performance among all four homes from schemes SE and SM that use the light sensor data. Interestingly, House 2, which had the least electrical noise, achieves close to 80% precision only through data fusion with ambient light sensor data (SMF). Houses 3 and 4 had significant noise from both light sensors and the power meter, and thus require the aggressive FixtureFinder algorithm to remove false positive fixture events.

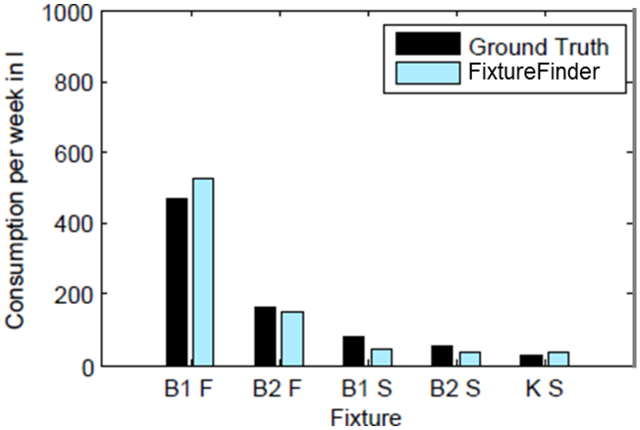
In general, we observed no difference in light intensity changes, power meter changes, water flow changes, light-power correlation, and motion-water correlations between day and night. For other pairings, such as light-water, we might expect significant differences in correlation between day and night, as some bathroom or kitchen lights may be used only at night; thus, in the future, time of day may need to be incorporated into the multi-modal sensor clusters that we produce, to improve accuracy. Also, we observed that 15.8% of the light fixture events occur within 10 seconds of each other, and 32.3% of all light fixture events occur within 20 seconds of each other. These co-occurrences make the data fusion and matching steps more challenging, by introducing numerous candidate choices for fusion and matching (in addition to noisy edges inherent in the system); in spite of these co-occurrences, FixtureFinder is able to accurately identify individual fixture events in the home with high accuracy, by combining the matching, data fusion, and Bayesian clustering steps.

6. RECOGNITION AND DISAGGREGATION

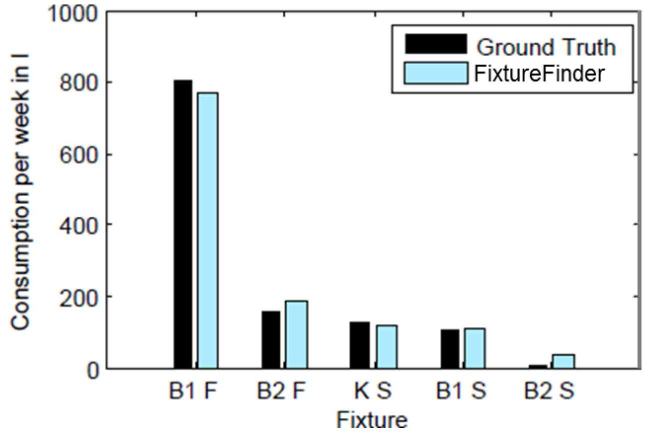
In this section, we demonstrate that FixtureFinder can also be used to perform fixture recognition and disaggregation, allowing smart power and water meter data to be used to infer activities and/or disaggregated energy usage merely by piggybacking on other sensors and infrastructure already in the home.

To perform fixture discovery, FixtureFinder must be configured to achieve high precision, possibly sacrificing recall (98% and 55% respectively, according to our results in Section 5). To perform fixture recognition, it must be configured to achieve more balanced precision and recall values. Figure 7 shows that the system can also achieve an operating point with about 86% precision and 81% recall. In other words, about 14% of the usage events associated with a fixture are actually caused by another fixture, while 19% of the events actually caused by the fixture are not associated with it.

To perform fixture disaggregation, FixtureFinder leverages the multi-modal profile of each fixture event that was created during fixture recognition, which includes the power or water consumption levels of that event. FixtureFinder estimates the total power/water usage for each event by summing the usage levels of all events assigned to each fixture. Figures 9 and 10 illustrate both the actual and the estimated energy/water usage for each of the water and light fixtures in



(a) Home 3



(b) Home 4

Figure 9: The water consumption estimated by FixtureFinder for each water fixture in the two homes with water meters closely matches the ground truth values for the most high flow fixtures, generally achieving 85-90% accuracy (B stands for bathroom, K for kitchen, S for sink, and F for flush).

the four test homes. Results show that FixtureFinder computes the water usage with 80-90% average accuracy across all water fixtures, and the energy usage with 91% average accuracy for the light fixtures consuming 90% of the home's lighting energy.

FixtureFinder's disaggregation results are surprisingly high given its recognition accuracy. The reason is that fixture recognition is most likely to make assignment errors on events with low power/water usage, and is most likely to assign the event to another fixture with similar average power/water usage. The errors often cancel out and do not significantly affect the overall energy/water estimates. For example, Figure 11 shows the confusion matrices for water fixture classification in the two homes that contained a smart water meter (element $[x,y]$ in the confusion matrix indicates the number of times an event from fixture x was associated with fixture y). Most misclassifications that did occur were due to simultaneous occupancy in multiple rooms from fixtures with similar flow signatures, such as high confusion between sink usage in the kitchen and bathroom2 in Home 3. In Home 4, there is about 7% confusion between the two flush fixtures. These misclassifications cause limited degradation in water consumption accuracy because they are infrequent, typically between fixtures with similar flow rates, and are roughly symmetric across the diagonal. Therefore, the recognition errors often cancel each other out in the disaggregation results.

7. LIMITATIONS

This paper presents a proof-of-concept of the FixtureFinder principles: that fixtures can be discovered and differentiated based on multi-modal profiles, even if a home contains multiple identical fixtures, and even if the fixtures are too simple and basic to be discovered based on smart power or water meter data alone. The results presented in this paper are not intended to represent a complete exploration of this concept, and the current version only explores a small subset of sensors and fixtures. In future work, we plan to study a more complete set of fixtures, including both major appliances and smaller appliances. We expect to use NIALM-like approaches to discover major appliance, but that FixtureFinder will be needed to narrow the set of appliance candidates and perhaps by identifying the number of appliances in the home. We expect that ElectriSense-like approaches can be used to discover very small, low-power and battery powered devices, such as cell phone chargers, electric toothbrushes, and music players. The current results do not yet demonstrate how this approach will scale to large numbers of low-power and battery powered devices.

The results presented in this paper indicate that FixtureFinder was unable to detect 4 of the 41 light fixtures due to their task-specific lighting capabilities that were not detected by ambient light sensors. Also, we require at least one sensor per room in our current deployments and evaluation. In future work, we expect to address this problem by exploring the use of a broader range of sensors and data streams. Previous work has shown the feasibility and usefulness of harvesting information from home infrastructure such as the home router, air pressure, and gas lines [5,15,23], and we will build on that work to also perform fixture discovery. As the variety of home sensors and automation devices proliferate, and as devices in the home increasingly become wireless and connected, the number of devices that must be automatically discovered will continue to grow, and the amount of information available to discover them will also grow.

The current implementation of FixtureFinder is limited to simple fixtures with ON/OFF events, but Step III could be extended to appliances with a wider variety of states without loss of generality. In the future, we expect FixtureFinder to leverage established work in this area to address a broader range of fixtures [9]. Also, due to sensor sampling rate limitations, if a set of light fixtures is switched on and off simultaneously (within 1 second of each other) within a single room in a consistent manner, the system will treat them as a single fixture; the use of additional data sources from the home may allow FixtureFinder to disambiguate these multi-fixture scenarios. Finally, in the current FixtureFinder implementation, we use fixed time thresholds for data fusion under the assumption that fixture usage simultaneously affects different sensor streams; an interesting direction for

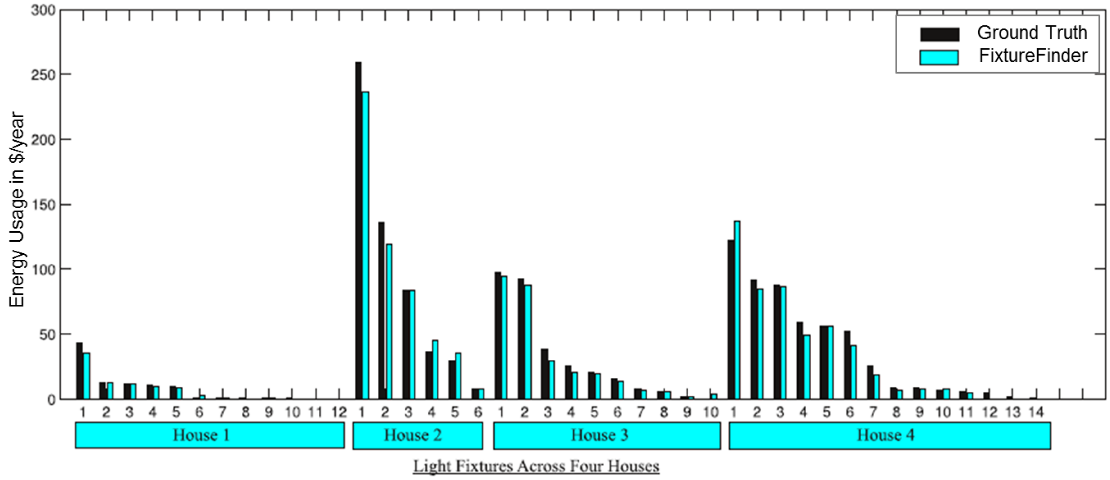


Figure 10: FixtureFinder accurately reports the energy usage of the top energy-consuming light fixtures in each home (as measured in dollars), achieving 90% accuracy for the 90% of appliances consuming the most energy. Importantly, the ordering of the light fixtures based on estimated energy usage is correct. This is a promising next step for FixtureFinder.

Fixtures	# of usage events	K S	B1 S	B1 F	B2 S	B2 F
K S	17	8	0	0	8	1
B1 S	22	1	19	0	2	0
B1 F	50	0	1	49	0	0
B2 S	22	3	1	0	16	2
B2 F	15	0	0	0	1	14

(a) Home 3

Fixtures	# of usage events	K S	B1 S	B1 F	B2 S	B2 F
K S	94	81	10	0	3	0
B1 S	90	7	78	0	5	0
B1 F	91	1	0	85	0	5
B2 S	7	0	1	0	6	0
B2 F	16	0	0	2	1	13

(b) Home 4

Figure 11: FixtureFinder accurately classifies water flow events for most of the monitored water fixtures across the two homes as seen in the fixture level confusion matrices (B stands for bathroom, K for kitchen, S for sink, and F for flush). Confusion between fixtures of the same type occurs due to simultaneous occupancy of different rooms, but has limited effect on water flow estimates. Confusion between fixtures of different types of fixtures is less common because the flow rates are distinctive.

future work is to automatically learn these temporal correlations between sensor streams over different time-scales.

8. CONCLUSIONS

In this paper, we present the FixtureFinder system that automatically infers the existence of electrical and water fixtures in the home. It uses data fusion between the smart meters and other sensors or infrastructure already in the home, such as the home security system, and searches for repeating patterns in the fused data stream. Unlike fixture recognition systems, Fixture Find does not try to recognize every ON/OFF event. Instead, it tries to select only those events that are very likely not to be caused by spurious noise. Once a set of such events is recognized, it can be used to create a training set for existing fixture recognition or disaggregation systems such as ElectriSense and Viridiscope. To our knowledge, this is the first system that can automatically discover the presence of small, simple fixtures.

We evaluated FixtureFinder by deploying between 25-40 sensors into 4 different homes for 7-10 days of data collection. Our results indicate that FixtureFinder is able to identify and differentiate major light and water fixtures in less than 10 days, including multiple copies of light bulbs and sinks that have identical power/water profiles. It can also produce clean training data sets to be used by other algorithms that require supervised training. In effect, FixtureFinder can be used to bootstrap other fixture recognition and disaggregation techniques at low cost by piggybacking on data from other sensors and infrastructure, such as home security or automation systems. In the future, the techniques described in this paper can be made more general and more effective by combining with other sensors and infrastructure, such as cell phones, home routers, gas meters, and the millions of home automation devices being sold today.

Acknowledgment

This work was supported, in part, by NSF grants CNS-1017363 and EFRI-1038271.

9. REFERENCES

[1] Pike research smart grid deployment tracker. http://www.pikeresearch.com/research/smart-grid-deployment-tracker-3q10.

[2] S. Barker, A. Mishra, D. Irwin, E. Cecchet, P. Shenoy, and J. Albrecht. Smart*: An open data set and tools for enabling research in sustainable homes. *The 1st KDD Workshop on Data Mining Applications in Sustainability (SustKDD)*, 2011.

[3] C. Beckel, W. Kleiminger, T. Staake, and S. Santini. Improving device-level electricity consumption breakdowns in private households using on/off events. *SIGBED Rev.*, 9(3), 2012.

[4] J. Canny. A computational approach to edge detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, (6):679–698, 1986.

[5] G. Cohn, S. Gupta, J. Froehlich, E. Larson, and S. N. Patel. Gassense: appliance-level, single-point sensing of gas activity in the home. In *Proceedings of the 8th international conference on Pervasive Computing*, Pervasive'10, 2010.

[6] J. Fogarty, C. Au, and S. Hudson. Sensing from the basement: a feasibility study of unobtrusive and low-cost home activity recognition. In *Proceedings of the 19th annual ACM symposium on User interface software and technology*, pages 91–100. ACM, 2006.

[7] J. Froehlich, E. Larson, T. Campbell, C. Haggerty, J. Fogarty, and S. Patel. HydroSense: infrastructure-mediated single-point sensing of whole-home water activity. In *Proceedings of the 11th international conference on Ubiquitous computing*, pages 235–244. ACM, 2009.

[8] J. Froehlich, E. Larson, T. Campbell, C. Haggerty, J. Fogarty, and S. Patel. Hydrosense: infrastructure-mediated single-point sensing of whole-home water activity. In *Proc. UbiComp*, volume 9, pages 235–244. Citeseer, 2009.

[9] H. Goncalves, A. Ocneanu, and M. Berges. Unsupervised disaggregation of appliances using aggregated consumption data. *The 1st KDD Workshop on Data Mining Applications in Sustainability (SustKDD)*, 2011.

[10] S. Gupta, M. Reynolds, and S. Patel. Electrisense: Single-point sensing using emi for electrical event detection and classification in the home. In *Proceedings of Ubicomp*, 2010.

[11] G. Hart. Nonintrusive appliance load monitoring. *Proceedings of the IEEE*, 80(12):1870–1891, 1992.

[12] L. Heyer, S. Kruglyak, and S. Yooseph. Exploring expression data: identification and analysis of coexpressed genes. *Genome research*, 9(11):1106, 1999.

[13] X. Jiang, S. Dawson-Haggerty, P. Dutta, and D. Culler. Design and implementation of a high-fidelity ac metering network. In *Proceedings of IPSN*, 2009.

[14] D. Jung and A. Savvides. Estimating building consumption breakdowns using on/off state sensing and incremental sub-meter deployment. In *SenSys*, pages 225–238, 2010.

[15] Y. Kim, R. Balani, H. Zhao, and M. B. Srivastava. Granger causality analysis on ip traffic and circuit-level energy monitoring. In *Proceedings of the 2nd ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Building*, BuildSys '10, pages 43–48, 2010.

[16] Y. Kim, T. Schmid, Z. Charbiwala, J. Friedman, and M. Srivastava. Nawms: nonintrusive autonomous water monitoring system. In *Proceedings of the 6th ACM conference on Embedded network sensor systems*, pages 309–322. ACM, 2008.

[17] Y. Kim, T. Schmid, Z. Charbiwala, and M. Srivastava. Viridiscope: design and implementation of a fine grained power monitoring system for homes. In *Proceedings of Ubicomp*, 2009.

[18] J. Z. Kolter and M. J. Johnson. Redd: A public data set for energy disaggregation research. *The 1st KDD Workshop on Data Mining Applications in Sustainability (SustKDD)*, 2011.

[19] H. Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955.

[20] P. Mayer and W. DeOreo. *Residential end uses of water*. American Water Works Association, 1999.

[21] Parks Associates Research and Analysis for Digital Living. Home security system forecasts: 2005 and beyond, November 2005.

[22] O. Parson, S. Ghosh, M. Weal, and A. Rogers. Non-intrusive load monitoring using prior models of general appliance types. *AAAi*, 2012.

[23] S. Patel, M. Reynolds, and G. Abowd. Detecting human movement by differential air pressure sensing in hvac system ductwork: An exploration in infrastructure mediated sensing. In *Pervasive Computing*, Lecture Notes in Computer Science, 2008.

[24] S. Patel, T. Robertson, J. Kientz, M. Reynolds, and G. Abowd. At the flick of a switch: Detecting and classifying unique electrical events on the residential power line. *Lecture Notes in Computer Science*, 4717:271, 2007.

[25] J. Polastre, R. Szewczyk, and D. Culler. Telos: enabling ultra-low power wireless research. In *Proceedings of IPSN*, 2005.

[26] A. Rowe, M. Berges, and R. Rajkumar. Contactless sensing of appliance state transitions through variations in electromagnetic fields. In *Proceedings of Buildsys*, 2010.

[27] V. Srinivasan, J. Stankovic, and K. Whitehouse. Watersense: Water flow disaggregation using motion sensors. In *The 3rd ACM Workshop On Embedded Sensing Systems For Energy-Efficiency In Buildings (BuildSys), in conjunction with ACM SenSys*, 2011.