A Longitudinal Study of Pressure Sensing to Infer Real-World Water Usage Events in the Home

Jon Froehlich[1,*], Eric Larson[2,*], Elliot Saba[2], Tim Campbell[3], Les Atlas[2], James Fogarty[1], and Shwetak Patel[1,2]

[1] Computer Science and Engineering
[2] Electrical Engineering,
[3] Mechnical Engineering
University of Washington, Seattle
{jfroehli,eclarson,sabae,tcampbell,lesatlas, jfogarty,shwetak}@uw.edu

Abstract. We present the first longitudinal study of pressure sensing to infer *real-world* water usage events in the home (e.g., dishwasher, upstairs bathroom sink, downstairs toilet). In order to study the pressure-based approach *out in the wild*, we deployed a ground truth sensor network for five weeks in three homes and two apartments that *directly monitored* valve-level water usage by *fixtures* and *appliances*. We use this data to, first, demonstrate the practical challenges in constructing water usage activity inference algorithms and, second, to inform the design of a new probabilistic-based classification approach. Inspired by algorithms in speech recognition, our novel Bayesian approach incorporates template matching, a language model, grammar, and prior probabilities. We show that with a single pressure sensor, our probabilistic algorithm can classify real-world water usage at the fixture level with 90% accuracy and at the fixture-category level with 96% accuracy. With two pressure sensors, these accuracies increase to 94% and 98%. Finally, we show how our new approach can be trained with fewer examples than a strict template-matching approach alone.

Keywords: Water sensing, activity inference, sustainability, field deployments.

1 Introduction

Low-cost and easy-to-install methods to sense and model human activity in the home have long been a focus of UbiComp research. Because water is fundamental to many activities of human life (e.g., bathing, cooking), sensing disaggregated water usage has emerged as a particularly promising area for human activity inference in the home [6, 8, 19]. In addition, these sensing systems can play a vital role in collecting highly granular consumption information for enabling eco-feedback and sustainability applications (e.g., [7]). In previous work, we introduced HydroSense [8], a pressure-based sensing solution that disaggregates water usage at the fixture level from a single installation point. HydroSense identifies the unique pressure waves generated when

[*] The first two authors contributed equally to this work.

K. Lyons, J. Hightower, and E.M. Huang (Eds.): Pervasive 2011, LNCS 6696, pp. 50–69, 2011.

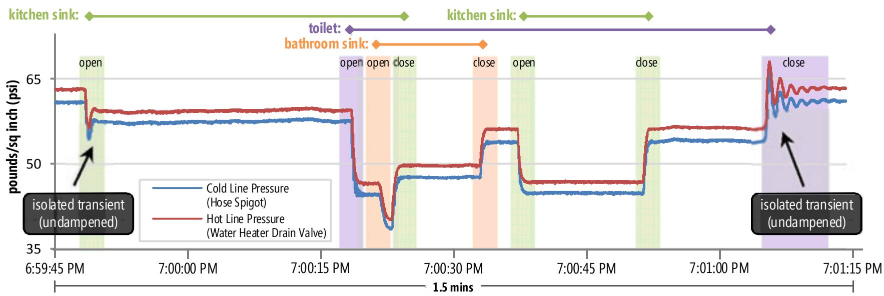


Fig. 1. A pressure stream with ground truth labels from deployment site H2. The blue line is the cold water pressure (sensed from a hose spigot) and the red line is the hot water pressure (sensed from a water heater drain valve). The pressure transients are also highlighted and colored according to fixture. Note how rapid increases and decreases in pressure correspond to opens and closes and how transient waveforms are dampened when they occur in compound.

fixtures are opened or closed. These waves propagate *throughout* a home's plumbing infrastructure, thus enabling the single-point sensing approach.

Although the original HydroSense work evaluated the pressure-based sensing approach using *staged experiments* in *controlled* home environments[8], it was unclear how well this approach would perform with *real-world* water usage. In this paper, we critically examine the feasibility of using pressure-based sensing to determine water usage activities in the home. We conduct real-world deployments in three homes and two apartments over a five-week period. In addition to installing pressure sensor sat each deployment site, we also deployed custom wireless *ground truth sensors* on individual fixtures throughout the home (*e.g.,* kitchen sink, toilet, dishwasher) to provide ground truth data on water activity events. The *ground truth sensors* were designed to track both hot *and* cold water usage at their respective fixtures. This allowed us to investigate not only whether the pressure signal could be used to infer fixture-level water activity but also whether it could be used to determine hot and/or cold water usage at each fixture. This is an important capability as water heating alone is responsible for 12.5% of residential energy consumption [17]. To our knowledge, our ground truth deployment represents the most comprehensive real-world study of hot and cold water usage in residential homes and apartments ever performed.

Over five weeks, we collected approximately 15,000 ground truth labels for the *opening* and *closing* of fixture valves (e.g., Figure 1). The scope and size of this dataset allows us to examine the practical challenges in constructing water usage activity inference algorithms and to highlight problems that any *indirect* water sensing method must address. We show, for example, that compound events (when two or more water fixtures are operating at the same time) constitute37.1% of all bathroom sink activity and nearly 20% of overall water usage activity. Such prevalence suggests that compound events should be specifically addressed and evaluated by any water disaggregation technique; however, this has rarely been the case (e.g., see [8, 9, 20]). Thus, our ground truth data serves both as a resource to inform the design of our classification algorithms as well as to evaluate their performance.

We use the ground truth labels along with the pressure stream data to design and evaluate a novel pressure-based water usage inference algorithm. Although the template matching of pressure wave transients used in our original HydroSense paper [8] worked well for controlled experiments, we show that a template-matching approach alone is insufficient for the variety of signal distortions that occur during real-world water use. For example, the speed with which a faucet handle is turned and whether an event occurs in isolation or in compound can change the shape of the pressure transient thereby rendering the naïve template matching approach inadequate. Consequently, we extend and adapt the original HydroSense algorithms to use a probabilistic model based in part on speech recognition algorithms. We show how the addition of a language model and contextual priors (e.g., fixture usage duration, and maximum flow rate) can boost classification accuracies by an average of 6% with real-world water usage data. We also show that the introduction of a language model and priors decreases the amount of training data relative to a template-based approach alone. Our current analysis provides pre-segmented pressure transients to our classification algorithm, leaving segmentation to future work. In this way, our classification results can be seen as an upper bound.

In summary, the contributions of this paper are: (1) The most comprehensive dataset of labeled real-world hot and cold water usage events ever collected in homes and apartments; (2) An analysis of our new real-world dataset to uncover challenges that any indirect sensing water disaggregation method must overcome; (3) A new probabilistic approach to water usage classification that is highly extensible and incorporates a language model, grammar, and contextual priors; (4) An evaluation showing that this new probabilistic approach performs significantly better than previous template-based methods.

2 Related Work

Automatic identification of home water usage events has largely been pursued by two non-overlapping efforts. Utilities and water resource management scientists have investigated disaggregation to inform government policy [13], plumbing codes [15], and to study the effectiveness of conservation programs [14] and low-flow fixtures [12,13]. In contrast, computing researchers have focused on human activity inference (e.g., [6, 8, 19]) and sustainability applications (e.g., [9]). We draw upon literature across both fields.

In studies by utilities and water resource management scientists, the most prevalent residential disaggregation technique is *flow-trace analysis*. Flow-trace analysis examines aggregate flow at a single *inline* water flow meter to determine the fixture *category* responsible for water usage [3]. Unlike HydroSense, flow-trace analysis only classifies at the fixture category level (i.e., it cannot determine the specific fixture or valve that was used). For example, flow-trace can determine that *a* toilet was flushed but not *which* toilet was flushed. Flow-trace analysis has been used in government- and utility- sponsored studies [3, 12, 13, 14], the largest of which included 1,188 households across North America [11]. Despite its prominence, flow-trace analysis has not been comprehensively studied. In the only known empirical investigation, Wilkes *et al.* conducted staged experiments of water usage over a five day period in one home. Flow-trace analysis correctly categorized 83% of the *isolated* water usage events at the fixture category level. When water usage overlapped (i.e., what we term *compound events*), performance dropped dramatically to 24% when two water fixtures were used in compound and 0% when three or more were used [20].

Researchers in the UbiComp and Pervasive communities have developed other water disaggregation techniques such as the Nonintrusive Autonomous Water Monitoring System [9], the original HydroSense work [8], and Sensing from the Basement [6]. In the only real-world evaluation, Fogarty *et al.,* installed microphones on water supply and sewage pipes in a single home and used temporal features in order to classify pipe noise into individual fixture usages. This work demonstrated that temporal features such as duration (e.g., a toilet flush lasts ~60 seconds) and on/off activations (e.g., a dishwasher cycles through a detectable pattern of water use) were useful in classifying water events at the fixture level. However, it also revealed the difficulty in discriminating between bathroom sink and kitchen sink uses, correctly classifying short water events (e.g., events that lasted less than 10 seconds), and correctly classifying compound events.

Our original HydroSense work was the first to show that pressure transients could be used to disaggregate water fixtures using staged experiments [8]. The experiments, however, were limited in that faucet handles were activated at approximately the same flow rate each time, and all fixtures were tested in isolation (i.e., no more than one fixture was used at a time). As we show in this paper and as could only be derived through a real-world ground truth deployment, much greater variations are common in real-world water usage. These phenomena can affect properties of the resulting pressure wave and thus the ease of classification.

3 Data Collection and Deployment

To evaluate the performance of a pressure-based approach using *real-world* data, we deployed a large ground truth water usage sensing network in three homes and two apartments. At each deployment site, we installed two pressure sensors and directly instrumented *all* water fixtures and appliances with custom wireless sensors that provided ground truth labels of water usage activity for the pressure stream. Here, we describe the ground truth data collection system and the five week study deployment.

3.1 Acquiring Ground Truth Labels in a Real-World Deployment

A key challenge in evaluating any new sensing technique is acquiring ground truth data. In the original HydroSense work [8], the team *manually* labeled the pressure stream during their staged experiments, which clearly would not work for a real-world evaluation. Thus, an automated method for labeling must be derived. An ideal labeling system would accurately detect when fixtures are turned on/off, be easy to install, work across a large variety of fixtures, and preferably provide flow and temperature information for each fixture valve. An accurate and direct approach would be to install small, wireless flow meters at each hot and cold fixture inlet (e.g., a sink would require two flow meters). Unfortunately, inline flow meters could actually distort the very phenomena we are interested in studying by impacting the

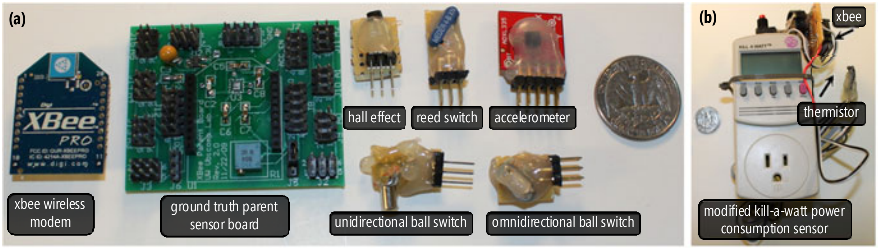


Fig. 2. The ground truth water usage sensors directly attached to (a) *fixtures* and (b) *appliances* and monitored valve *openings and closings*. This data was transmitted wirelessly in real time via the ground truth parent sensor board and aXBee wireless modem (a, left side) to a data logger.

pressure-wave signal itself. Instead, we instrumented fixtures externally, such as on faucet and toilet handles, so that we did not disturb the water stream.

We designed an array of ground truth sensors to accommodate the variety of home water fixtures: from hand operated fixtures like sinks to electromechanical appliances such as dishwashers. Even for a single fixture type, design variation affects how flow and temperature are selected and how they can be sensed. For example, some single-handle faucets move left to right for temperature and up or down for flow while dual-handle faucets select both temperature and flow by the open position of each handle.

3.2 Water Usage Activity Ground Truth Sensors

We developed seven ground truth sensors to accommodate all fixtures across our deployment sites. Each interfaced with a *parent sensor board* (wireless platform in Figure 2a, top right) to communicate water usage data in real time. At a minimum, we tracked when each valve was opened or closed and categorized temperature into *hot only*, *cold only*, and *mixed*. The parent sensor board was placed in a location protected from water and preferably next to a power outlet (5 of 33 ground truth sensor boards relied on battery power). All sensors and parent boards were weather proofed to protect against water damage. XBee Pro wireless modems (Figure 2a, top left) transmitted sensor state to a logger on a laptop installed at each deployment site. The sensor boards were configured to transmit a watchdog signal once every four minutes so failures could be quickly identified and corrected. The ground truth architecture and sensors went through several design cycles and took approximately three months to build and evaluate before being deployed in this study.

For sinks, showers, and toilets, sensors to detect handle position were affixed directly to the fixtures themselves and linked to the wireless parent board via low-voltage wires (Figure 3).We used three types of handle sensors: *reed switches* (N=34 sensors deployed), *accelerometers* (N=14), and *Hall effect sensors* (N=3). Reed switches are electrical switches that react to the presence of a magnetic field and produce binary output: on or off. They are inexpensive, robust to water exposure, and provide easily analyzable data. For toilets, we instrumented the flush handle, which only provided data on the beginning of the fill and not on the end. We discuss how this end fill information was recovered in the next section.

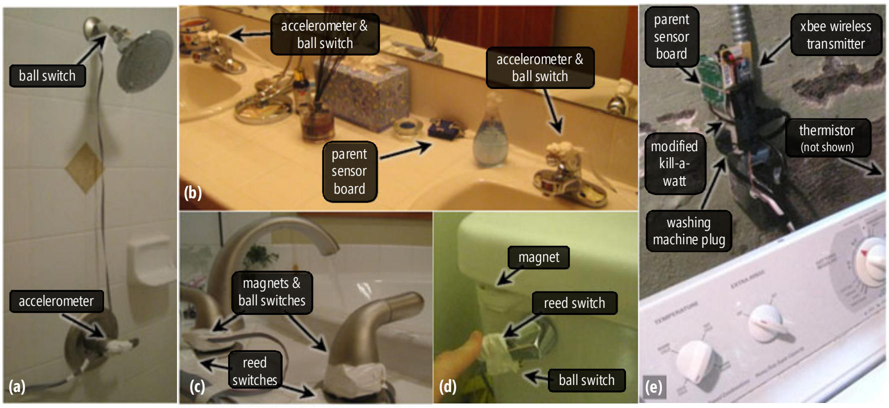


Fig. 3. A sample of instrumented fixtures from our ground truth deployments. Note how different sensors (e.g., accelerometers, reed switches) are used to accommodate the variety of fixture types.

For faucets where a single handle controls flow rate and temperature, the reed switches were insufficient. Instead, we used three-axis accelerometers (Figure 3a and 3b) to measure acceleration and interpret the handle's flow position (typically up and down movement) and temperature (typically left and right movement). Finally, we used Hall effect sensors for sensing faucets which control temperature using planar rotation but control flow through an up/down motion (i.e., where an accelerometer alone could not sense the planar motion). A Hall effect sensor provides a voltage difference representing the distance between two magnets, so we placed magnets on both sides of faucet handles and attached the Hall effect sensor to the handle itself.

Additionally, each hand-operated fixture had at least oneomni-directional *ball switch* (*N*=39) that acted as a vibration sensor and woke the parent board to read and transmit handle position sensor data. This allowed us to limit power consumption and unnecessary XBee wireless traffic.

For washing machines and dishwashers, we used three types of sensors: *power usage sensors* (*N*=7)**,** *push buttons*(*N*=2), and *thermistors* (*N*=3). Power consumption patterns were used to reconstruct when appliances used water. We could not gain access to the power outlets in two cases (deployment site A1's washing machine and H1's dishwasher), so we used push buttons and a note reminding the resident to "*please push button when turning on <appliance>*."For sites with washing machines, we also attached thermistors to the water drain pipe to measure the temperature of the previous fill cycle and infer machine settings (e.g., Hot/Cold, Warm/Cold).

3.3 Pressure Sensors and Software Tools

The above sensor network was deployed at each deployment site to provide *ground truth labels* for our pressure sensors. For our pressure sensors, we used Pace Scientific P1600s with a resolution of 0.03 psi. Each was connected to a 16-bit Texas

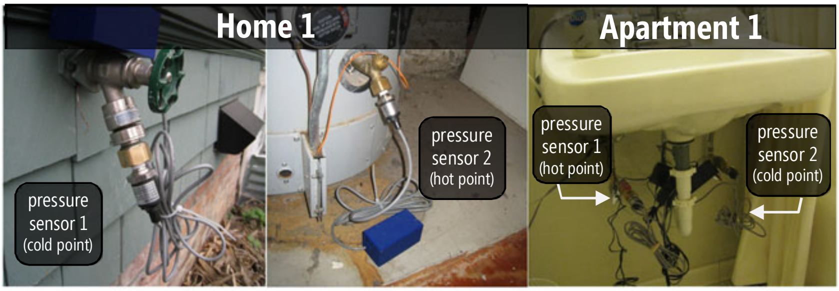


Fig. 4. Two pressure sensors were installed at each deployment site (one on a hot water access point, one on a cold) in order to study the effect of installation points on classification accuracies

Instruments ADS8344 ADC and AVR microcontroller, which interfaced with a Class 1 Bluetooth radio implementing the serial port profile with an approximate wireless range of 10m. This is the same setup as the original HydroSense study with three exceptions. First, instead of one pressure sensor, we connected *two* sensors to collect data from hot *and* cold water access points simultaneously (Figure 4). This allowed us to investigate the effect of installation point as well as the effect of two pressure streams compared to one on classification performance. Second, the original HydroSense work tested only $^3/_4$" water access points (e.g., hose spigot). We built adapters to connect to $^3/_8$" access points, which allowed us to install pressure sensors below kitchen and bathroom sinks (Figure 4, right). This was particularly important for the apartment installations, which did not have accessible $^3/_4$" access points. Finally, we used a sampling rate of 500Hz rather than 1,000Hz, as we found 500 Hz was more than sufficient to capture these pressure waves.

To communicate with the ground truth sensor network and the pressure sensors, a 2GHz Dell Inspiron 1545s laptop running Windows XP was deployed at each site. The laptops were configured with a USB XBee wireless modem and Bluetooth dongle. The laptops continuously ran a custom data logger written in C#, which received, compressed and archived data locally for backup. This was uploaded to a backend web server at 30-minute intervals. The server backend was implemented using Python and web2py. In addition to serving as a data repository, the backend automatically sent e-mail notifications when a ground truth sensor or pressure sensor was not heard from for 10 minutes or more. For analysis, we constructed a suite of tools in Matlab and C#. Because not all of the ground truth sensors provided direct labels about water usage (e.g., the power usage sensors and toilet handle sensors), we also built a custom annotation tool in C# that allowed us to quickly review the ground truth sensor streams and semi-automatically annotate the pressure stream.

3.4 Deployments

We deployed the ground truth sensor network and two pressure sensors at five sites: three houses and two apartments. Each site was a home or apartment of one of the authors. This was done because of the invasiveness of the direct sensing approach used for the ground truth data collection. There was, however, a large variation in the type, size and plumbing systems across the deployments sites (Table 1). The deployments began February 2010 and lasted for five weeks.

It took approximately two full work days per deployment site for two people to install and test the ground truth sensors. After the five-week ground truth deployments ended, we used our custom annotation tool to convert the ground truth sensor stream to labels. This was accomplished in a semi-automatic fashion—the annotation tool visualized the ground truth sensor values and the pressure streams together in a common time-series view. The ground truth sensor values could then be automatically or manually converted to labels. It took approximately 8-12 hours per week of data collected for one research assistant to convert the sensor stream to labels. These labels were then reviewed by a second research assistant for consistency, which took roughly half the time (4-6 hours per week of data).

4 Analysis of the Collected Dataset

We collected a total of 16,056 labeled events across the five deployment sites. Table 2 provides an overview. Due to ground truth sensor failures, 2.9% of this data is marked as *uncertain* and is not used in our classification experiments. Nearly 80% of the uncertainties were due to malfunctioning kitchen sink handle position sensors in H1 and H2, which were replaced within a few days of discovery. The dataset also includes *unknown* events (3.9% of our dataset), which are pressure stream transients whose origin cannot be determined because they occurred without accompanying data from the ground truth sensors. A1 has the highest proportion of unknown events (9.1%) because of water usage activity coming from other apartments. Although we do not attempt to classify uncertain or unknown events, they were not removed from the dataset and can impact classification performance when they overlap with other events. After accounting for uncertain/unknown events, we are left with 14,960 labels.

Table 1. Occupant demographics and deployment site characteristics. In A1, The toilet and shower head were replaced with low-flow equivalents ~3.5 weeks into the deployment. We discuss the effect of this change on classification performance in the results section.

	H1	**H2**	**H3**	**A1**	**A2**
# Residents	2	2	4	2	2
Gender/Age/ Profession	M/27/professor; F/29/professor	M/31/professor; F/32/office worker	4 Males/19-21/ undergrad students	M/31/grad student; F/30/post-doc	M/26/grad student; F/26/pharmacist
Fixtures/Valves	17/28	8/13	13/21	6/10 *(8/13)**	8/13
Style/Built	House/2003	House/1918	House/ 1923	Apt/1920s	Apt/2000
Size/Floors	3000 sqft/ 2 floor + basement	750 sqft/ 1 floor + basement	1200 sqft / 1 floor + basement	700 sqft/ 3rd floor of 3	750 sqft/ 6th floor of 7
Expansion Tank/ Regulator	Yes/Yes	No/No	No/No	N/A	N/A
Water Heater Tank Size/ Plumbing	50 gal/ Copper	50 gal/ PEX	50 gal/ Copper	Two 100 gal tanks/ galvanized	N/A/ PEX
Pressure Sensor Install Point Hot/Cold	Main floor bathroom sink/outdoor hose spigot	Water heater drain valve/outdoor hose spigot	Downstairs bathroom sink/outdoor hose spigot	Bathroom sink hot/cold inlet	Kitchen sink hot/cold inlet

Table 3 shows valve activity at individual fixtures by temperature state (hot, cold, mixed). We use *M.* for Master and *S.* for secondary to distinguish primary and secondary bathrooms. The *M. Bath Diverter* and *S. Bath Diverter* are for the tub/shower switch that diverts water flow from the bath to the shower and vice versa; we distinguish between a shower that is turned on straightaway and a shower that is diverted from a bath. The *Other* category includes data from only one deployment site, H1, and encompasses the *Laundry Basin* and the *Refrigerator Water Dispenser*. On average across all deployment sites, there is a nearly even proportion of cold and hot events (40.7% for cold only, 39.2% for hot, and 20% for mixed). This implies that any indirect water disaggregation sensing method, such as flow-trace analysis and HydroSense, must be equally capable of sensing usage regardless of temperature. The overall frequency of fixture usage follows a power-law distribution where the first four fixtures (*kitchen sink, master bathroom sink* and *toilet,* and *secondary bathroom sink*) account for 84.7% of the events in our dataset. For purposes of human activity inference, these fixtures are thus critically important.

Table 2. High level ground truth data collection statistics. An *event* is one occurrence of either a valve open or a valve close. Uncertain and unknowns are *not* included in the totals events row.

	H1	H2	H3	A1	A2	Totals
Days of Data	33	33	30	27	33	156
Total Events	2374	3075	4754	2499	2578	14960
Avg Events/Day	71.9	93.2	158.5	92.6	78.1	95.9
Cold Only Events	855 (36.0%)	1418 (46.1%)	1637 (34.3%)	633 (25.3%)	1657 (64.3%)	6087 (40.7%)
Hot Only Events	607 (25.6%)	1329 (43.2%)	1766 (37.5%)	1818 (72.8%)	498 (19.3%)	5870 (39.2%)
Mixed Temp Events	912 (38.4%)	328 (10.7%)	1351 (28.2%)	48 (1.9%)	423 (16.4%)	3003 (20.1%)
Isolated Events	1981 (83.5%)	2477 (80.6%)	4131 (86.9%)	1914 (76.6%)	2149 (83.4%)	12393 (82.8%)
Compound Events	393 (16.6%)	598 (19.5%)	623 (13.1%)	585 (23.4%)	429 (16.6%)	2567 (17.2%)
Transient Collisions	142 (6%)	72 (2.3%)	166 (3.5%)	219 (8.8%)	120 (4.7%)	701 (4.7%)
Uncertain Events	22 (0.9%)	175 (5.3%)	189 (3.7%)	52 (1.9%)	37 (1.4%)	467 (2.9%)
Unknown Events	52 (2.1%)	79 (2.4%)	184 (3.6%)	254 (9.1%)	85 (3.1%)	629 (3.9%)

Table 3. A breakdown of valve activity by fixture, by temperature state (hot, cold, mixed) and by compound/collisions. The *Cnt* column tabulates the number of fixtures across sites.

Fixtures	Cnt	Total	Hot	Cold	Mixed	Compound	Collision	AvgDuration
KitchenSink	5	5494 (36.7%)	2438 (44.4%)	1415 (25.8%)	1641 (29.9%)	342 (6.2%)	206 (3.7%)	22.4 secs
M.Bathroom Sink	7	3934 (26.3%)	2114 (53.7%)	1294 (32.9%)	526 (13.4%)	1459 (37.1%)	185 (4.7%)	27.2 secs
M.Bathroom Toilet	5	1886 (12.6%)	0 (0.0%)	1886 (100%)	0 (0.0%)	87 (4.6%)	117 (6.2%)	43.6 secs
S.Bathroom Sink	4	1369 (9.2%)	618 (45.1%)	637 (46.5%)	114 (8.3%)	430 (31.4%)	57 (4.2%)	30.9 secs
Washing Machine	4	430 (2.9%)	93 (21.6%)	325 (75.6%)	12 (2.8%)	12 (2.8%)	66 (15.3%)	1.6 mins
M.Bathroom Bath	5	423 (2.8%)	224 (53%)	35 (8.3%)	164 (38.8%)	87 (20.6%)	20 (4.7%)	43.4 secs
S.Bathroom Toilet	3	341 (2.3%)	0 (0.0%)	341 (100%)	0 (0.0%)	11 (3.2%)	21 (6.2%)	27.2 secs
M.Bathroom Shower	5	261 (1.7%)	55 (21.1%)	4 (1.5%)	202 (77.4%)	30 (11.5%)	10 (3.8%)	8.7 mins
Dishwasher	3	261 (1.7%)	261 (100%)	0 (0.0%)	0 (0.0%)	9 (3.4%)	6 (2.3%)	1.2 mins
M.Bath Diverter	5	228 (1.5%)	17 (7.5%)	1 (0.4%)	210 (92.1%)	92 (40.4%)	5 (2.2%)	N/A
Other	1	181 (1.2%)	28 (15.5%)	149 (82.3%)	4 (2.2%)	0 (0.0%)	4 (2.2%)	8.2 secs
S.Bathroom Bath	2	59 (0.39%)	5 (8.5%)	0 (0.0%)	54 (91.5%)	2 (3.4%)	2 (3.4%)	20.7 secs
S.Bathroom Shower	2	47 (0.31%)	11 (23.4%)	0 (0.0%)	36 (76.%)	0 (0.0%)	1 (2.1%)	9.4 mins
S.Bath Diverter	2	46 (0.31%)	6 (13%)	0 (0.0%)	40 (87%)	6 (13%)	1 (2.2%)	N/A
Totals	53	14960	5870 (39.2%)	6087 (40.7%)	3003 (20.1%)	2567 (17.2%)	701 (4.7%)	49.1 secs

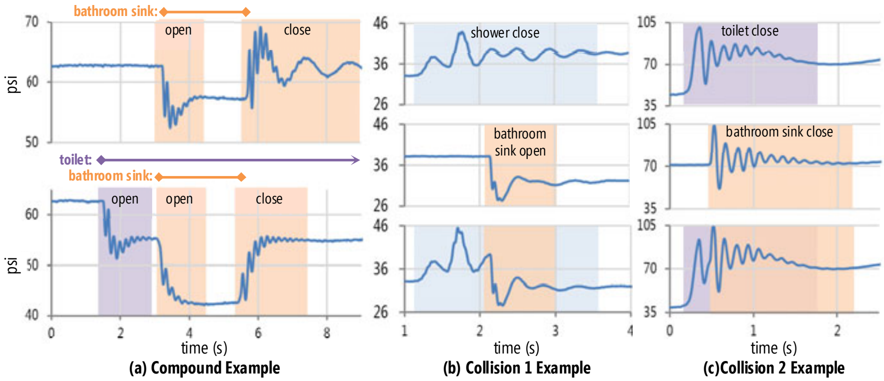


Fig. 5. (a) Bathroom sink open and close transients occurring in isolation and in compound from H2. (b) A shower close and bathroom sink close transient in isolation and colliding from A2. (c) A toilet close and a bathroom sink close transient in isolation and colliding from H3.

Although we ultimately used this data to evaluate our classification algorithms, an equally important goal was to identify potential challenges in classifying real-world water usage compared to simulated, isolated water events. A *compound valve event* is a valve event that occurs while another fixture is using water (e.g., the bathroom sink events in Figure 5a). A *collision valve event* is a valve event that occurs within *twoseconds* of one or more other valve events (Figure 5b and 5c). Previous water disaggregation sensing approaches have performed poorly in the face of compounds and collisions (e.g., [6, 20]). This is because compounds and collisions often mask or distort features used for classification. Although a collision is technically also a compound, for the purposes of our analysis we separate them to investigate the individual effect of each on classification performance. In our dataset, 17.2% of all valve events are compound while 4.7% of valve events are collisions (Table 2 and 3). The most common compound/collision events are master bathroom sink opens and closes, comprising 41.8% of all bathroom sink activity and 11% of all valve activity overall (Table 3).

With the pressure-based approach, compound valve events result in a dampening and often a severe attenuation of the high frequency component of the pressure transient. As a result, the transient signal is homogenized, making it difficult to classify. With collisions, the two colliding transient waveforms become highly distorted; although it is rarely the case that two transients occur simultaneously (more often they are offset by 200-500ms), the distortions may still render the transient unrecognizable. In Figure 5b, the shower close and bathroom sink open occur 1.1s apart. In Figure 5c, the toilet close and bathroom sink close occur 200ms apart, making it unlikely that both will be classified. For these events to be classified correctly, less emphasis may need to be placed on template matching transient signatures relative to the original HydroSense work [8]. Our new algorithm specifically addresses this issue.

5 Valve Event Classification Algorithm

To classify pressure transients as valve events, we apply a probabilistic approach using Bayesian estimation. Our particular approach is inspired by the dynamic Bayesian models used in speech recognition. Instead of recognizing *words,* we recognize *valve events*. Like many of the Bayesian approaches used in speech recognition, we incorporate a language model and grammar, which estimates the most likely *sequence of valve events* and defines permissible *valve event pairings*. This provides robustness against transient deformations that can occur during natural valve usage (e.g., brief water usage events, low-flow, and compounds).

At a high level, the classification algorithm works as follows: First, an incoming water pressure data stream is buffered and the pressure transients are segmented. This segmentation process currently uses the time series boundaries defined by the ground truth annotations but would be automated in an end-to-end system. Second, the segmented pressure transients are each compared to a library of labeled templates using a set of similarity algorithms. Third, a language model determines the likelihood of a given sequence of valve signatures and links *open* and *close* valve events into *paired tuples.* Fourth, we extract features from these paired tuples and compare them with smoothed probability distributions. For example, by pairing a *bathroom sink hot open* with a *bathroom sink hot close*, we can extract the *duration* of that event and estimate the *total flow volume* used and then obtain probabilities for those features. Finally, the probabilities from the previous three steps are multiplied together for each sequence and the sequence with the highest probability is selected.

We now formally define our Bayesian model for classifying pressure transient sequences. In eq. (1) below, let **V** denote the pressure signature template library (a vector of labeled pressure transient signatures and their transforms) and **S** denote a sequence of *unknown* segmented pressure transients. Then, using Bayes' theorem, the most likely valve sequence is defined as:

$$\hat{V} = \arg\max P(\mathbf{V} \mid \mathbf{S}) = \arg\max \frac{P(\mathbf{S} \mid \mathbf{V})P(\mathbf{V})}{P(\mathbf{S})} \tag{1}$$

The conditional probability term $P(\mathbf{S}|\mathbf{V})$ describes the outcome of the *template*-and *feature-based comparisons.* The prior probability term $P(\mathbf{V})$ describes the likelihood of the valve sequence (using bigrams) and the likelihood of each pairing in the sequence. Note that *arg max* simply returns a specific valve sequence rather than a probability estimate, thus the normalization constant $P(\mathbf{S})$ can be discarded in practice. We can expand the numerator of eq. (1) to further highlight the four major components of our approach:

$$\overbrace{\underbrace{\prod_{r=0}^{R-1} f_r(\hat{\mathbf{S}}_r \mid \hat{\mathbf{V}}_r)}_{\text{(i) templates and signal features}}}^{P(\mathbf{S}|\mathbf{V})} \overbrace{\underbrace{\prod_{n=0}^{N-1} P(v_n \mid v_{n-1})}_{\text{(ii) bigram language model}} \underbrace{\prod_{i \notin \beta} f_p(v_i)}_{\text{(iii) grammar}} \underbrace{\prod_{k=0}^{K-1} \prod_{\langle a,b \rangle \in \beta} f_k(\langle v_a, v_b \rangle)}_{\text{(iv) paired valve priors}}}^{P(\mathbf{V})} \tag{2}$$

$P(\mathbf{S}|\mathbf{V})$ is now represented by the first term in eq. (2), which describes our set of R signal transformations and comparison algorithms (where f_r is the comparison algorithm for the rth transformation). $P(\mathbf{V})$ is expanded into three terms: our bigram language model, a grammar, and water usage event priors. We describe each term in the following.

Term (i): Template- and Feature-Based Comparison: Term (i) compares the segmented unknown pressure transient s with *open* and *close* valve templates in our library. Each comparison is broken into two parts: a *signal transformation* on s to achieve $\hat{s}$ and a *similarity score* calculation between $\hat{s}$ and a corresponding transformed valve template $\hat{v}$ in our template library. We use multiple signal transformations and comparison algorithms to produce a set of similarity scores for a given valve (each transformation and score is represented by f_r in term (i)). These scores are converted into probabilities and multiplied together to form a single template-match probability between s and every valve v in the template library. This is similar to our original HydroSense work which used a hierarchical classifier to prune and classify these individual pressure transients. Unlike this past work, however, these similarity scores are incorporated into a probabilistic model.

We use eight signal transformations—four filters and a Cepstral transform of each filter. Each attempts to emphasize a unique property of the pressure transient waveform. The first two filters, a 1 Hz and a 13 Hz low-pass filter, allow us to explore the temporal shape of the transient signal. The next two filters are *derivatives* of the low-pass filtered signals, which help to uncover how resonances of the transient waveform decay over time. Specifically, we use a derivative of the 13 Hz low-pass filter and a band pass derivative of the difference between the 1 and 13 Hz low-pass filters. Finally, we apply a constant-Q Cepstral transformation on *each* of the aforementioned four transforms.

The constant Q transformation uses a filter bank with overlapping and logarithmically increasing bandwidths to break apart the frequency spectrum of the transient signal. After the filter bank, we apply a magnitude and log operation to turn multiplication of two systems in the frequency domain into addition operations. This has the effect of separating the "source" (an impulse or step into the plumbing system) from the "filter" (the physical bends and pipe lengths in the plumbing system). We then take the discrete cosine transform (DCT) of the constant-Q coefficients, which compacts harmonic structures down towards lower indices of the transform (commonly known as low-time cepstral coefficients). We truncate these coefficients (known as low-time *liftering*) before applying similarity algorithms. For more information on our constant-Q transformations, see Larson *et al.* [10].

We use two similarity algorithms over the eight signal transformations: a matched filter and a Euclidean distance measure. The matched filter is an optimal similarity measure for orthogonal signals in the presence of white noise [16]. Because our signals resemble decaying sinusoids, we can expect the above transformations to result in signals that are approximately orthogonal. The matched filter is used to compare the first four signal transformations, while the Euclidean distance measure is used for the four Cepstral transformations (given that the Cepstral space is already aligned, a matched filter type approach is unnecessary). A similar set of signal transformations and comparison algorithms were used in the original HydroSense work [8]. However, to ensure the approach works robustly with real-world data, we added the 4th signal transform above (the band pass derivative) and eliminated the *mean square error* measure because it did not improve performance.

After every $\{s,v\}$ comparison has been made, we reinterpret the similarity scores as probabilities. For the matched filter comparisons, this is trivial as the matched filter already returns a similarity score between 0 and 1. For the real Cepstral transforms, we use Euclidean distance measures d_m between each transient in **S** and template in **V**, such that $f_{EucDist}(\hat{s}|\hat{v}) = e^{-|d_m|}$ (a common interpretation of Euclidean distance as a probability in log-space [18]).

At this point in the algorithm we have an unknown transient *s* and the results of the four matched filter comparisons and the four exponential Euclidean distance comparisons (for every template in our library). To form a single template probability score, we multiply the comparisons of each template together. These scores are then grouped by valve (i.e., all "kitchen sink open hot" scores are grouped together; all "bathroom sink close cold" are grouped, etc.). We then take the *argmax* over each valve grouping to find the probability that a particular valve is the originator of *s*.

Because we now have a single probability score for each valve, we can combine these with the probability of observing valve-specific features. These features are low dimensional vectors or scalars that are pre-calculated for each valve at a deployment site. In particular, we use two features: (1) stabilized pressure drop and (2) amplitude/resonance tracking; however, other features such as *damping ratio* and *time of day used* could be explored in the future. The stabilized pressure drop can be calculated by assuming that the transient is an underlying step function with three parameters: (a) time at which the step occurs, t_0, (b) magnitude of the step, A_0, and (c) region, *T*, where the transient has many high frequency components and cannot be modeled by a step. These parameters can be solved for (in the mean square sense) using linear regression with a "don't care" region. After regression, the stabilized pressure drop is the scalar value A_0. For resonance/amplitude tracking we assume the transient can be modeled well by a four pole system and we use an auto regressive model to estimate the pole locations. Each pole represents the strongest resonances and resonance magnitude which can vary between valves.

We train probabilities for these features by calculating the pressure drop and resonance values for all templates in our library and then using Gaussian kernel density estimation (KDE) [1] to assign probability distributions to each valve in a non-parametric way. This results in a look-up table between feature observations and valve-level probability estimates. These probabilities are multiplied with the template probabilities to complete term (i). Note that when multiple pressure sensor streams are available, such as when two installation points are used, the probabilities for each stream can be multiplied together to form term (i).If we wish to use template comparisons *only*, we can simply choose the template with the highest probability. To incorporate with a language model, we use the best valve probabilities to enumerate the state space of a trellis in a bigram graphical model(where each valve is a separate state).

Term (ii): The Language Model: The *language model* assigns probabilities for possible valve sequences. This is performed using bigrams and is represented by term (ii) in eq. (2) (*N* represents the length of the sequence). Bigram analysis is commonly used in the statistical analysis of text to examine co-occurrences of words or letters. Here, our bigrams are groups of two sequential valve events; for example, *toilet* *open→bathroom sink valve hot open* comprises a single bigram. The language model consists of transition probabilities for every valve pair $\langle v_{n-1}, v_n \rangle$ and is trained by counting the number of co-occurring valve pairs in our library. These counts are smoothed using Katz smoothing, which is commonly used in speech recognition and works to assign a non-zero probability to every sequence [18]. This is important for handling transition probabilities between two valves that rarely occur in our library.

Traditionally, language models use these transition probabilities to select the optimal word (valve) sequence from all possible word (valve) sequences. We maintain an *n-best list* of sequences using Viterbi-stack decoding [2]. This allows us to dynamically reorder the most probable sequences as new valve events occur. Crucially, it also allows us to reorder based on secondary knowledge sources—particularly term (iii) and term (iv) in eq. (2).

Term (iii): The Grammar: Term (iii) describes a grammar, which is typically used to define a set of structural rules that govern the composition of sentences, phrases, and words in a given language. Here, our grammar defines the possible ways in which valve sequences can be constructed. Our grammar rules are: (1) an opening of valve v_x must be followed by a closing of valve v_x; (2) a valve's closure must be preceded by its opening; (3) and the temperature state of a valve must be consistent—e.g., a *kitchen sink hot open* event cannot be closed by a *kitchen sink cold close* event. Rather than eliminating impossible valve sequences (such as a close before an open or an open with no close), we use a *soft grammar* which applies a penalty to any valve sequence that violates a rule. In this way, sequences which contain grammatical errors but have the likeliest probabilities from the other terms can still be selected as correct. The grammar is applied to each sequence in the n-best list, resulting in a set of *paired valvetuples* β. In eq. 2, the term f_p penalizes all unpaired valves (those not in β).

These paired tuples now bind together specific valve open and close events to form a full water usage event structure. For example, given the valve event sequence $v_1 \rightarrow v_2 \rightarrow v_3 \rightarrow v_4$ where v_1*=toilet open*, v_2*=bathroom sink open,* v_3*=toilet close, and* v_4*=bathroom sink close,* our pairing algorithm might link the two toilet events into $\hat{\beta}_1 = \langle w_1 | w_3 \rangle$ and the two bathroom sink events into $\hat{\beta}_2 = \langle w_2 | w_4 \rangle$. These linkages are critically important because they allow us to compute an additional feature set (described in term (iv)) that is dependent on knowing the beginning and ending of a water usage event. We note that the language model and pairing is a novel aspect of our system. The original HydroSense had no notion of either and thus could only identify individual valve events but not the relationships between those events.

Term (iv): Paired Valve Tuple Priors: By pairing valve events, we not only have the ability to link open and close transients together but also to compute classification features, such as *water usage duration* and relative *estimates of water volume*, which are not possible without a pairing methodology. For every paired valve tuple in β, we compute K features over the entire water usage event, denoted as f_k in eq. (2). Similar to the *transient features* used in term (i), a probability density is calculated using KDE and the example water usage events in our library. For example, given a particular draw length for an unknown tuple, we can use the usage durations for all kitchen sinks in our dataset to lookup the probability that the usage event is a kitchen sink.

Once all paired prior probabilities have been multiplied together, the n-best list is reordered and the likeliest valve sequence is chosen.

We use two paired valve priors selected experimentally using one week of data from each deployment site: *usage duration*, the amount of time the given valve pair is drawing water and *flow-trace max*, an estimate of the maximum amount of flow used over the duration of the event (a feature also used in flow-trace analysis [3]).

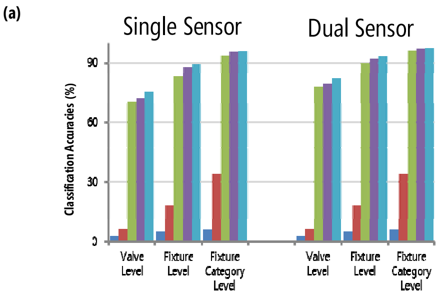
6 Analysis and Results

We compare the performance of three classification algorithms: a template classifier (term (i)); a classifier that adds a language model and grammar: *templ+LM* (terms (i, ii, iii)); and our full classifier*templ+LM+priors* (the complete eq. (2)). For baseline performance, we include *chance* and a *majority* classifier, which always selects the most likely result based purely on frequency. We were most interested in how the *templ+LM+priors* approach compares to the *template* approach. Additionally, we investigate the performance of each algorithm when using a single pressure sensor (hot or cold) versus dual pressure sensors. For the single sensor analysis, we chose the sensor (hot or cold line) that performed best. This was the cold line for all sites except for A2, where the majority of events were hot water use only.

To understand how the algorithms perform at different granularities, we conduct *valve* level, *fixture* level, and *fixture category* level classification. For valve level, the algorithm must identify the correct *fixture* responsible for the pressure transient, whether it is an *open* or a *close*, and its *temperature state* (hot, cold, or mixed). Fixture level ignores temperature state. Finally, for the fixture category level, we use the same categories as flow-trace analysis (e.g., [11]). The algorithm must correctly classify open/close events as *bath, clothes washer, dishwasher, faucet, shower* or *toilet.* Note that the same models were used to train and test all three different granularities; however, temperature errors were ignored in the case of fixture and category level.

We first focus on pre-segmented classification performance using data from a *single* pressure sensor. Figure 6 (left) displays the results of a 10-fold cross validation experiment over the full five weeks of data using the three classification algorithms and two baselines. In general, the best performing algorithm is *templ+LM+priors*, which resulted in an average overall classification accuracy of 75.5%, 89.5%, and 95.9% for valve, fixture, and fixture-category level, respectively, across the five deployment sites. The best performing home, H2, resulted in 89.4%, 94.3%, and 98.4% classification accuracies. In contrast, the worst performing home, H1, resulted in 66.6%, 79.6%, and 91.0% accuracies because of the lack of cross talk between hot and cold plumbing lines and the logarithmic pressure falloff during usage. Surprisingly, the two apartments, A1 and A2, both performed reasonably well with a single sensor: 77.3%, 89.7%, and 95% for A1 and 78.7%, 94.3% and 96.9% for A2. This is despite the pipe length distance between the hot and cold lines in an apartment being much longer than in a house.

(a)



	Single Sensor: Valve Level	Single Sensor: Fixture Level	Single Sensor: Fixture Category Level	Dual Sensor: Valve Level	Dual Sensor: Fixture Level	Dual Sensor: Fixture Category Level
chance	3.0	5.1	6.2	3.0	5.1	6.2
majority	6.5	18.5	34.2	6.5	18.5	34.2
template	70.2	83.4	93.6	78.1	89.6	96.2
template+LM	72.5	88.0	95.4	79.4	92.0	97.1
temp+LM+priors	75.5	89.5	95.9	82.4	93.5	97.7

(b)

	M. Bathrm Faucet	M. Bathrm Toilet	M. Bath	M. Bath Diverter	M. Shower	S. Bathrm Faucet(s)	S. Bathrm Toilet(s)	S. Bath	S. Bath Diverter	S. Shower(s)	Kitchen Faucet	Dishwasher	Washing Machine	Other
M. Bathrm Faucet	84.9	0.8	0.7	0.1	0.1	5.8	0.0	0.0	0.0	0.1	10.2	0.2	0.1	1.0
M. Bathrm Toilet	1.8	93.3	0.4	0.4	0.2	4.5	0.8	0.8	0.0	0.0	1.6	0.2	0.4	3.6
M. Bath	4.0	0.9	79.1	2.1	2.6	1.0	3.0	5.0	0.0	2.0	1.8	0.3	1.9	21.8
M. Bath Diverter	4.9	1.8	1.4	81.6	4.5	0.0	0.0	2.3	7.0	2.3	3.6	0.0	0.0	0.0
M. Shower	3.7	2.0	5.2	0.5	80.7	3.1	0.5	0.0	0.0	3.8	3.5	0.0	0.3	1.0
S. Bathrm Faucet(s)	7.0	0.2	0.0	0.1	0.1	86.1	0.1	0.1	0.1	0.2	5.4	0.1	0.3	0.0
S. Bathrm Toilet(s)	2.0	1.1	3.2	1.1	0.0	0.2	91.0	2.7	0.0	0.0	1.5	0.0	0.0	2.0
S. Bath(s)	0.0	6.8	4.5	0.0	0.0	0.0	2.3	81.8	0.0	4.5	0.0	0.0	0.0	0.0
S. Bath Diverter	2.3	0.0	0.0	0.0	2.3	0.0	0.0	0.0	90.9	2.3	2.3	0.0	0.0	0.0
S. Shower(s)	0.0	0.0	0.0	0.0	2.1	6.4	0.0	0.0	0.0	89.4	0.0	2.1	0.0	0.0
Kitchen Faucet	4.9	0.3	0.1	0.0	0.0	1.4	0.0	0.0	0.0	0.0	93.6	0.5	0.1	0.9
Dishwasher	4.0	0.0	0.0	0.0	0.0	0.9	0.5	0.0	0.0	0.5	11.0	84.2	0.0	0.0
Wash. Mach.	2.1	0.9	4.2	1.0	2.1	1.0	0.0	0.0	0.0	0.0	1.6	0.0	81.0	30.1
Other	0.9	0.0	0.4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.9	0.0	5.1	92.7

Fig. 6. (a) Average classification results across the five deployment sites comparing *algorithm*, *single* vs. *dual* sensor, and different granularities (*valve, fixture, fixture category*). (b) A confusion matrix that averages the confusions for fixture level *templ+LM+priors* classifications across deployment sites. Note that averaging makes it such that the percentages do not add to 100%.

To examine how events were misclassified, we calculated a confusion matrix for *templ+LM+priors*(Figure 6b), averaging the classification percentages at the fixture level across the five deployment sites. In general, classification accuracies are quite good—the most frequently used fixtures: kitchen sink, bathroom sinks, and bathroom toilets have an average classification accuracy of 90%. Confusions tend to occur within fixture categories (e.g., between sinks) and between fixtures that are situated close together with respect to plumbing layout. For example, the faucet in the secondary bathroom is misclassified as the master bathroom faucet 7% of the time while the dishwasher is misclassified as a kitchen sink 11% of the time (dishwashers are only a small distance from kitchen sinks). Recall from Table 3 that the *other* category involves data from only one home (H1) and is for the laundry basin and refrigerator water dispenser, which were classified correctly 86.1% and 98.6% of the time. However, the *washing machine* was confused as a laundry basin 30.1%, which is visible in Figure 6—this confusion can be attributed to their valve's proximity in the plumbing system.

With regards to compound and collision events, the two language model-based algorithms tend to perform better than the *templ* algorithm (Figure 7a). This is likely due to the transition probabilities of the language model and the paired valve priors in term (iv). Both reduce the weight placed on template-matching the distorted transient.

As expected, the addition of a second pressure sensor improves the overall classification accuracies for each algorithm and sensing resolution granularity: an average of 10% for valve level, 5.5% for fixture level and 2.1% for fixture category level across the three algorithms. The *templ* algorithm benefited the most from the addition of the second sensor. Similar to the single sensor, the *templ+LM+priors* algorithm performed the best with overall accuracies of: 82.4%, 93.5%, and 97.7% for valve, fixture, and fixture category levels. Because of the lack of cross talk between hot and cold pressure lines, H1 and the apartments benefited the most from the addition of a second sensor, especially for valve level classification (an increase of

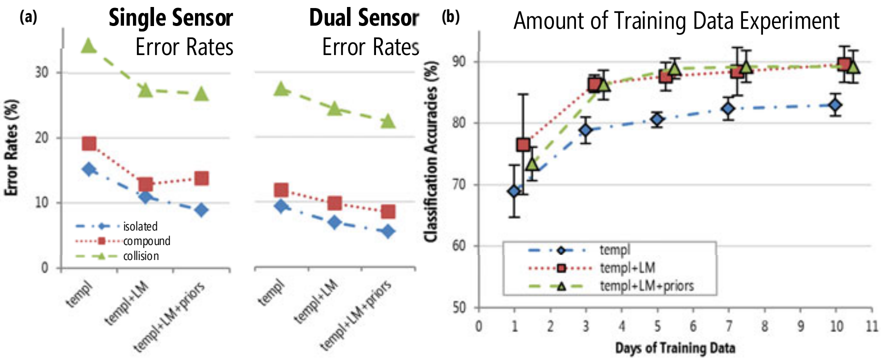


Fig. 7. (a) The error rates for fixture-level performance broken down by algorithm and whether the error occurred on an *isolated*, *compound*, or *collision* event. (b) The results of our amount of training data experiment; 1,3,5,7 and 10 days were used to test a two week period. Note that we offset the data points slightly for each algorithm to improve the readability of the graph. Error bars reflect one standard deviation above and one standard deviation below the mean.

9.5% vs. 3.1% for the other two sites). Two sensors also increase compound and collision accuracy by 5.3% and 4.4%. Finally, as noted in Table 1, the toilet and showerhead were replaced with low-flow equivalents in A1 approximately three and a half weeks into the deployment. After training on these new fixtures, we were able to correctly classify their usage despite being in the same fixture category and installed in the same location as the previous fixtures. For example, the new toilet was correctly classified 90.2% of the time and classified as the old toilet 8.2% of the time (we kept the old fixture templates in our database for all classification experiments).

To test whether *templ+LM+priors* offered a significant overall improvement over *templ* (the approach used by the original HydroSense work [8]), we conducted a three-way repeated measures ANOVA. We usedthe 10-fold classification accuracies as the dependent variable and *sensing resolution*, *number of sensors*, and *algorithm* (*templ* vs. *templ+LM+priors*) as within-subjects factors. Because we were only interested in the comparisons between the two algorithms, we report only main and interaction effects with *algorithm*. We found a significant main effect of *classification algorithm* ($F(1,4)=21.76$, $p=.010$), indicating that *templ+LM+priors* improved performance over *templ*. No interaction effects with *algorithm* were significant.

To investigate how the amount of training data impacts performance, we trained models with one, three, five, seven, and ten days of data. The amount of data is divided by days, not number of templates, as the language model requires contiguous blocks of events for training. All were then tested on 14 non-overlapping days. The results are presented in Figure 7b.Significant improvements in classification accuracy are seen with only a small number of training days. On average, *templ+LM+ priors* outperforms *templ*by 4.5%, 7.4%, 8.3%, 6.9% and 6.2%as the number of training days increases from one to ten. Note that both of the LM-based algorithms perform better throughout training though the *templ+LM* algorithm slightly outperforms *templ+LM+priors* with minimal training because it does rely on trained probability distributions for priors.

7 Discussion and Conclusion

This paper is the first to use pressure-sensing to disaggregate *real-world* water usage. Using longitudinal data collected from ground truth deployments across five residences, we showed that a single pressure sensor was sufficient to classify pressure transients with accuracies between 76% and 96% depending on granularity (i.e., valve, fixture, or fixture category). With two pressure sensors, the accuracies rose to between 82% and 98%. To achieve these results, we introduced a new type of water usage inference algorithm inspired by research in speech recognition. Unlike previous approaches [8], our algorithm is probabilistic and leverages a language model, grammar, and prior probabilities to better handle pressure transient variability and to increase robustness in the face of compound events and collisions.

Despite these advances, there are important opportunities for future work. Our current analyses used *pre-segmented* pressure transients (i.e., the start and end of waveforms are marked by the ground truth labels). Working with pre-segmented events allowed us to focus specifically on analyzing the *discriminability* and *consistency* of real-world water usage pressure transients. As such, our results demonstrate an *upper bound* of classification performance for our particular feature set and approach. Overall classification rates will likely drop once segmentation is implemented because of segmentation errors. This could be especially true for apartments which, depending on the plumbing structure, can be particularly sensitive to noise from other units in the building.

With that said, the original HydroSense work segmented staged water usage data with 100% accuracy, so segmentation of real-world data should be possible. The key challenge will be properly segmenting compound and collision events, particularly in apartments with a much noisier pressure signal. We note that our Bayesian approach is amenable to many common speech recognition detection techniques such as keyword spotting. As such, the *classification* and *segmentation* tasks could likely be combined to make the algorithm more robust to sources of ambiguity such as transient collisions. Indeed, most optimal statistical signal processing strategies become sub-optimal after separating segmentation and classification, which means the classification algorithms presented in this paper may need adjustment once incorporated with an imperfect segmentation scheme.

In terms of training, we evaluated the classification algorithms using real-world data for both training and testing. For practical end-user deployment, we might expect a small amount of *staged* training data per fixture. Future work is necessary to establish what will be the most effective staged training data for accurate classification of real-world data. For example, our current approach trains the language model and priors using data from the home where it is deployed. A more general approach could leverage usage patterns and priors (such as duration of use) across different homes, thus reducing system calibration. It may also be the case that certain fixtures, such as toilets and dishwashers, require less calibration because of more consistent transients. Furthermore, unsupervised learning approaches may allow detection of previously unknown fixtures. An interface to allow correction of misclassifications and training of the algorithm over time may also prove beneficial.

Finally, our work underscores the importance of conducting *longitudinal* evaluations *out in the wild*. Although challenging and resource-intensive, such studies are critical in providing a sound scientific basis for the sensing work that we do in the UbiComp/Pervasive communities. In our case, studying the real-world uses of water, rather than only staged experiments, uncovered crucial limitations of past approaches and allowed us to characterize general challenges for water disaggregation research.

In conclusion, this paper is the first to demonstrate that sensing pressure is a viable technique for inferring *real-world* water activity. We used *labeled* pressure stream data collected through five-week ground truth water sensor deployments across five sites to evaluate the performance of a new *probabilistic method* for inferring water usage from a single pressure sensor. To our knowledge, these ground truth deployments represent the most detailed investigation of residential hot and cold water usage ever performed.

Acknowledgements. We would like to thank: Al Dietemann, Bob Alpers, and Ray Hoffman at Seattle Public Utilities for their feedback and help; Gabe Cohn and Tien-Jui Lee for their workin designing and building the ground truth water sensors; Leah Findlater for her thoughts and feedback on early versions of this paper; and Mike Hazas for his helpful comments.

References

1. Bowman, A.W., Azzalini, A.: Applied Smoothing Techniques for Data Analysis. Oxford University Press, New York (1997)
2. Chow, Y., Schwartz, R.: The N-Best algorithm: an efficient procedure for finding top N sentence hypotheses. In: Proc. of the Workshop on Speech and Natural Language, Cape Cod, Massachusetts, October 15-18. Association for Computational Linguistics, Morristown (1989)
3. DeOreo, W.B., Heaney, J.P., Mayer, P.W.: Flow Trace Analysis to Assess Water Use. Journal of the American Water Works Association 88(1) (January 1996)
4. DeOreo, W.B., Mayer, P.W.: The End Uses of Hot Water in Single Family Homes from Flow Trace Analysis. Aquacraft, Inc., (2002)
5. Dziegielewski, B., Opitz, E., Kiefer, J., Baumann, D.: Evaluation of Urban Water Conservation Programs: A Procedures Manual. Prepared for California Urban Water Agencies by Planning and Management Consultants, Ltd., Carbondale, Illinois (February 1992)
6. Fogarty, J., Au, C., Hudson, S.E.: Sensing from the Basement: A Feasibility Study of Unobtrusive and Low-Cost Home Activity Recognition. In: Proc. of UIST 2006, pp. 91–100 (2006)
7. Froehlich, J., Findlater, L., Landay, J.: The Design of Eco-Feedback Technology. In: Proceedings of CHI 2010, Atlanta, GA, pp. 1999–2008 (2010)
8. Froehlich, J.E., Larson, E., et al.: HydroSense: infrastructure-mediated single-point sensing of whole-home water activity. In: Proc. of UbiComp 2009, Orlando, Florida, USA, pp. 235–244 (2009)
9. Kim, Y., Schmid, T., Charbiwala, Z.M., Friedman, J., Srivastava, M.B.: NAWMS: Non-Intrusive Autonomous Water Monitoring System. In: Proceedings of SenSys 2008, pp. 309–322 (2008)
10. Larson, E., et al.: Disaggregated water sensing from a single, pressure-based sensor: An extended analysis of HydroSense using staged experiments. Pervasive and Mobile Computing (in press)

11. Mayer, P.W., DeOreo, W.B., Kiefer, J., Opitz, E., Dziegielewski, B., Nelson, J.O.: Residential End Uses of Water. American Water Works Association, Denver (1999)
12. Mayer, P.W., et al.: Great Expectations—Actual Water Savings with the Latest High-Efficiency Residential Fixtures and Appliances. In: Proc. of the Water Sources Conference, Las Vegas, NV (2002)
13. Mayer, P., DeOreo, W. B., Towler, E., Lewis, D. M.: Residential Indoor Water Conservation Study: Evaluation of High Efficiency Indoor Plumbing Fixture Retrofits in Single-Family Homes in the East Bay Municipal Utility District Service Area, Prepared for EBMUD and the US EPA (July 2003)
14. Mead, N., Aravinthan, V.: Investigation of Household Water Consumption Using A Smart Metering System. Desalination and Water Treatment 11, 1–9 (2009)
15. Navigant Consulting. Water & Heating Working Group Meeting. Residential & Multifamily: Background, Outcomes & Next Steps. ACEEE Hot Water Forum, Downey, CA (March 10, 2010)
16. North, D.O.: An analysis of the factors which determine signal/noise discrimination in pulsed carrier systems. RCA Labs, Princeton (1943)
17. US Department of Energy. US Household Electricity Report, Energy Information Administration, US DoE (2001), `http://www.eia.doe.gov/emeu/reps/enduse/er01_us_tab1.html` (last accessed October 10, 2010)
18. Chen, S.F., Rosenfeld, R.: A Survey of Smoothing Techniques for Maximum Entropy Models. IEEE Transactions on Speech and Audio Processing 8(1), 37–50 (2000)
19. Tapia, E., Intille, S.S., Larson, K.: Activity Recognition in the Home Using Simple and Ubiquitous Sensors. In: Ferscha, A., Mattern, F. (eds.) PERVASIVE 2004. LNCS, vol. 3001, pp. 158–175. Springer, Heidelberg (2004)
20. Wilkes, C., Mason, A., Niang, L., Jensen, K., Hern, S.: Evaluation of the Meter-Master Data Logger and the Trace Wizard Analysis Software. Special Appendix to Quantification of Exposure-Related Water Uses for Various U.S. Subpopulations. Prepared for US EPA (December 2005)