The Second International Workshop on Man-Machine Symbiotic Systems, November 23-24, 2004, Kyoto, Japan, pp. 241-250.

TempIO: Inside/Outside Classification with Temperature

John Krumm

Microsoft Research
Microsoft Corporation
One Microsoft Way
Redmond, WA USA
e-mail: jckrumm@microsoft.com

Ramaswamy Hariharan

School of Information and Computer Science
University of California, Irvine
Irvine, CA USA
e-mail: rharihar@ics.uci.edu

Keywords: temperature, context, inside, outside, building, ubiquitous computing

Abstract

TempIO is a method to classify a device as either being inside or outside based on its ambient temperature. It takes advantage of the fact that inside temperatures are normally controlled to within a range comfortable for people, while outside temperatures fluctuate with the weather. Inside/outside classification could be used to automatically turn off a GPS receiver when inside, for automatically adding metadata to digital photos, and for higher-level context inference. TempIO works by measuring the ambient temperature and looking up the current outside temperature via a network. We derive a Bayes-based classification rule based on probability distributions of inside, outside, and measured temperatures. Based on test data from five U.S. cities, TempIO classifies correctly 81% of the time when using a web service for outside temperatures and almost 91% of the time when using an ousider thermometer.

1. Introduction

As computing moves off the desktop into the hands of mobile users, it is becoming more important for mobile devices to be aware of the user's context. Important pieces of context include the user's location, activities, nearby people and devices, and mode of transportation. This knowledge can in turn be used by mobile devices to display reminders, to configure themselves for use with other devices, and to behave in a way that is appropriate for the surrounding environment.

One important piece of context concerns whether or not the user is outside. This can be used to help infer the user's location (*e.g.* in a building) and her mode of transportation (*e.g.* in a bus or car). Inside/outside can also be used to turn off a GPS receive when inside, because GPS does not generally work inside. It is also a useful piece of metadata for digital photos, potentially serving as a way to filter photos in a search.

One way to make an inside/outside determination would be to use a digital map of building footprints along with knowledge of the user's location. However, for most buildings such a map does not exist. Also, location data is not necessarily available, especially inside where GPS fails. Another inside/outside feature is light intensity, but this fails at night.

We have developed TempIO, a working technique for inside/outside classification that exploits the fact that inside environments are normally temperature-controlled. If the mobile device can measure the ambient temperature, and if it knows the current outside temperature, it can infer whether or not it is outside. The outside temperature comes from a database of worldwide, outside temperatures that we

Figure 1: We use this RS-232 thermometer to measure ambient temperature.

maintain based on hourly updates from the American National Oceanic and Atmospheric Administration's (NOAA's) National Weather Service (NWS). If the device's ambient temperature is within the range of normal inside temperatures, and if the outside temperature is significantly different, then there is a high probability that the device is inside. If, on the other hand, the device's ambient temperature is closer to the local outside temperature, then the device is more likely outside.

One attractive characteristic of this technique is the simplicity of the required sensing. We measure temperature with a small, off-the-shelf, RS-232 thermometer that draws its power from the mobile device, as shown in Figure 1.

Looking up the outside temperature requires that the device have a rough idea of its own location. But, since temperatures vary only slowly with location, the location estimate needn't be accurate. For instance, our system can work with locations given in terms of postal codes.

Clearly our technique will be only guessing if the inside and outside temperatures are very close to each other. By reasoning mathematically about the temperature distributions, our technique gives a probability of being inside, which reflects the uncertainty caused by similar inside and outside temperatures. Also, despite this potential ambiguity, we found that our technique is correct about 81% of the time based on tests with weather data from five U.S. cities.

The certainty of our inside/outside inferences are strongly related to the certainty of three different temperature distributions:

1. Measured ambient temperature from the device
2. Expected inside temperature
3. Outside temperature interpolated from weather stations

The next section explains how we derived these three different temperature distributions. After that, we show how we combined these distributions mathematically to create a probability estimate of being inside or outside. We then describe our accuracy tests.

2. Temperature Distributions

TempIO's inside/outside inference is a function of three different temperatures: the measured ambient temperature, the outside temperature, and the inside temperature. All three are described by probability distributions that are used to compute the probability of being inside. This section describes how we derived the three probability distributions. The probabilistic inference described here takes a closed form if these three probability distributions are Gaussian, so we endeavor to model the distributions as such as long at it appears reasonable.

2.1. Measured Temperature Distribution

For measuring temperature, we us a TempTrax™ RS232 thermometer. It has an advertised accuracy of ±0.28 ^oC over a range of -28.9 ^oC to 48.9 ^oC (-20 ^oF to 120 ^oF). Because the manufacturer could not clarify the meaning of this accuracy figure, we reasoned that this uniform distribution over ±0.28 ^oC is actually a Gaussian with the same variance, as shown in Figure 2. The variance of a uniform distribution over $[a,b]$ is

$$\sigma^2 = \int_a^b x^2 /(b-a)\, dx = (b-a)^2 /12 \qquad (1)$$

From our thermometer's accuracy specification, $(a,b) = (-0.28\ ^oC, 0.28\ ^oC)$, giving $\sigma_m = 0.162$. Thus the distribution of actual temperatures t_a is $t_a \sim N\{t_m, \sigma_m^2\}$, where t_m is the temperature measured with the mobile device, and $N\{\mu, \sigma^2\}$ represents a normal distribution.

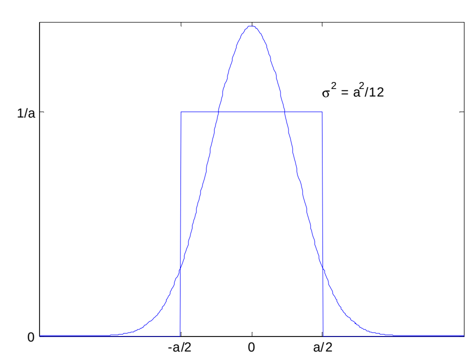

Figure 2: A uniform and Gaussian distribution with the same mean and variance.

2.2. Expected Inside Temperature

The inside temperature of a building of interest could easily be measured with an inside thermometer connected to a network and used as part of our system. This requires extra infrastructure, however, so we chose to depend instead on the fact that building temperatures normally vary over only a small range.

Buildings are usually temperature-controlled for the comfort of their occupants, with obvious exceptions for saunas, wine cellars, *etc*. In lieu of temperature data from a large sample of buildings, we use ISO standard 7730 that limits temperatures of commercial buildings to 20 - 24 ^oC in winter and 23 - 26 ^oC in summer. We have a temperature range $(a,b) = (20\ ^oC, 26\ ^oC)$ that we model as a normal distribution with a variance from Equation (1) and a mean that splits the range. This gives $\mu_{in} = 23$ and $\sigma_{in} = 1.732$, with inside temperature t_{in} distributed as $t_{in} \sim N\{\mu_{in}, \sigma_{in}^2\}$.

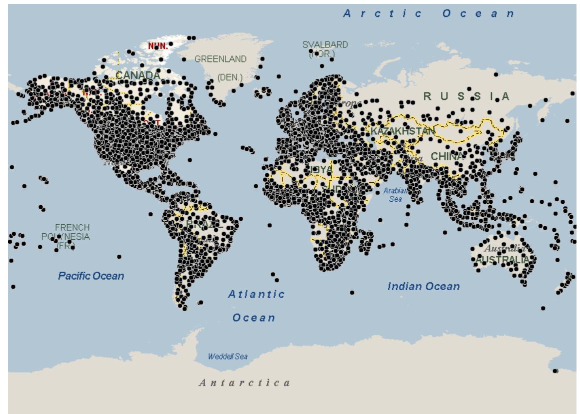

Figure 3: Black dots indicate the locations of 6510 weather stations that we use to compute outside temperature at a given location.

2.3. Outside Temperature

One way to get the local outside temperature would be to equip areas of interest with networked-connected thermometers. For instance, if a nursing home wanted to monitor if any of its residents left the building, it could use a thermometer installed immediately outside. For this project, however, we instead chose to exploit thermometers which are already in place, meaning that we do not depend on any new infrastructure.

Our outside temperatures come from 6510 weather stations located around the world, shown in Figure 3. Hourly updates from these stations is gathered by the American NOAA's National Weather Service and made available as METAR reports[1]. Our server maintains the latest data from each of these stations by hourly downloading the latest METAR summary file.

As part of the inside/outside inference, the user must specify his/her location in order to compute the outside temperature. As temperature varies only slowly as a function of location, the measured location does not need to be very accurate. It is sufficient to give the last known (latitude, longitude) from a GPS receiver or, in the U.S., the postal code which we convert to a (latitude, longitude) via our web service that accesses a database of postal codes and their (latitude,longitude)'s.

Given a (latitude, longitude), we use interpolation to compute the local temperature. This problem has been studied previously, and we base our choice of interpolation scheme on the work of Collins and Bolstad[2], who compared interpolation methods. They found that "optimal inverse distance weighting" to perform well. This technique interpolates temperature at a point of interest as a weighted average of all the known temperatures. The weights are the reciprocals of the distances between the known points and the point of interest, raised to some power that is computed by experiment. Mathematically, the outside temperature t_{out}^* is computed as

$$t_{out}^* = \frac{\sum_{i=1}^{n} t_i / d_i^r}{\sum_{i=1}^{n} 1 / d_i^r} \qquad (2)$$

Where t_i is the temperature reported from the i^{th} weather station, d_i is the distance between the point of interest and the i^{th} weather station, n is the number of weather stations, and r is the experimentally determined optimal exponent. We can compute d_i because the METAR reports give the (latitude, longitude) of each weather station.

We computed the best r based on 24 consecutive hours of temperature data from all the weather stations, excluding the inevitable missing reports from some stations. For each hour time slice, we used a leave-one-out procedure to estimate the interpolation error. Leaving out one weather station, we used all the others to estimate its temperature using Equation (2). Taking each station and each hour in turn, we computed an rms interpolation

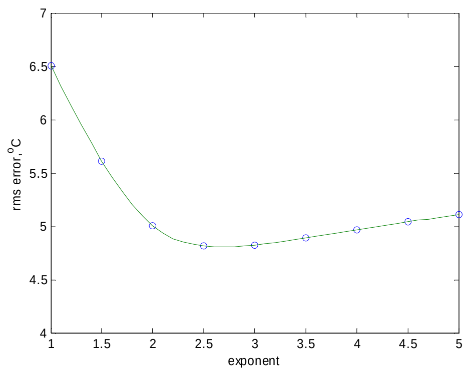

Figure 4: RMS temperature error as a function of weighting exponent gives an optimal exponent of 2.65.

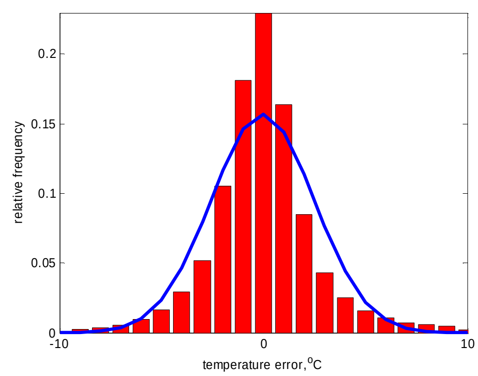

Figure 5: Error distribution for estimating outside temperature.

error. By exhaustively searching through different values of r, we found the minimum rms exponent was $r = 2.65$, as shown in Figure 4.

To estimate the error distribution of outside temperature, we employed the same leave-one-out procedure as above and created a histogram of errors shown in Figure 5. Before computing error statistics, we eliminated errors above 10 ^{o}C and below -10 ^{o}C, which amounted to about 3% of the data. This gave a better-fitting Gaussian, also shown in Figure 5. Since this error distribution has a mean of approximately zero (actually -0.4 ^{o}C), and a standard deviation of $\sigma_{out} = 2.545$, we will model the distribution of interpolated outside temperatures as

$$t_{out} \sim N\left\{t_{out}^{*}, \sigma_{out}^{2}\right\} \quad (3)$$

where t_{out}^{*} is the interpolated temperature for the given location.

We have created two web services to facilitate access to our interpolated outside temperatures. The first, mentioned above, converts U.S. postal codes into (latitude, longitude). The second takes a (latitude, longitude) and returns the outside temperature based on the interpolation in Equation (2).

3. Probabilistic Inference

Based on the three temperature distributions above, our aim is to derive an equation giving the probability of being inside based on the measured ambient temperature and interpolated outside temperature. Starting with Bayes' rule, we have the probability of being inside given the measured ambient temperature t_m:

$$p(\text{in}|t_m) = \frac{p(t_m|\text{in})p(\text{in})}{p(t_m|\text{in})p(\text{in}) + p(t_m|\text{out})p(\text{out})} \quad (4)$$

For lack of any prior assumptions, we will assume the prior probabilities $p(\text{in}) = p(\text{out}) = 0.5$.

3.1. Measured Temperature Conditioned On Inside

The first state conditional probability in Equation (4) is $p(t_m|\text{in})$, which is the probability of the measured temperature t_m given that the device is inside. This is a function of the actual ambient temperature, t_a, which we do not know. We can introduce the joint conditional probability distribution $p(t_m, t_a|\text{in})$ and integrate out the actual temperature to compute the probability we need:

$$
\begin{aligned}
p(t_m|\text{in}) &= \int_{-\infty}^{\infty} p(t_m, t_a|\text{in})dt_a \\
&= \int_{-\infty}^{\infty} p(t_a|\text{in})p(t_m|t_a, \text{in})dt_a
\end{aligned}
\quad (5)
$$

The distribution $p(t_a|\text{in})$, reduces to the normal distribution governing inside temperatures:

$$
\begin{aligned}
p(t_a|\text{in}) &= p(t_a) \\
&= N\{t_a; \mu_{in}, \sigma_{in}^2\}
\end{aligned}
\quad (6)
$$

where $N\{x;\mu,\sigma^2\}$ is the Gaussian density function:

$$
N\{x;\mu,\sigma^2\} = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2} \quad (7)
$$

The distribution $p(t_m|t_a, \text{in})$ represents the accuracy of our thermometer, and reduces as follows:

$$
\begin{aligned}
p(t_m|t_a, \text{in}) &= p(t_m|t_a) \\
&= N\{t_a; t_m, \sigma_m^2\}
\end{aligned}
\quad (8)
$$

Continuing from Equation (5) using the resultant normals from Equations (6) and (8), we have the closed form

$$
\begin{aligned}
p(t_m|\text{in}) &= \int_{-\infty}^{\infty} p(t_m, t_a|\text{in})dt_a \\
&= \int_{-\infty}^{\infty} p(t_a|\text{in})p(t_m|t_a, \text{in})dt_a \\
&= \int_{-\infty}^{\infty} N\{t_a; \mu_{in}, \sigma_{in}^2\}N\{t_a; t_m, \sigma_m^2\}dt_a \\
&= N\{\mu_{in}; t_m, \sigma_{in}^2 + \sigma_m^2\}
\end{aligned}
\quad (9)
$$

The last step comes from the identity[3]

$$
\begin{aligned}
&\int_{-\infty}^{\infty} N\{x;\mu_1,\sigma_1^2\}N\{x;\mu_2,\sigma_2^2\}dx \\
&\quad = N\{\mu_1;\mu_2,\sigma_1^2+\sigma_2^2\}
\end{aligned}
\quad (10)
$$

Equation (9) is intuitively satisfying in that the maximum of $p(t_m|\text{in})$ occurs at the mean inside temperature μ_{in}. The function broadens and falls with increases in the uncertainty of the inside temperature (σ_{in}^2) and the uncertainty of the measured temperature (σ_m^2).

3.2. Measured Temperature Conditioned On Outside

The other conditional probability from Equation (4) is $p(t_m|\text{out})$, which is the probability of the measured temperature given that the device is outside. Proceeding as above, we derive a closed form:

$$
\begin{aligned}
p(t_m|\text{out}) &= \int_{-\infty}^{\infty} p(t_m, t_a|\text{out})\,dt_a \\
&= \int_{-\infty}^{\infty} p(t_a|\text{out})\,p(t_m|t_a, \text{out})\,dt_a \\
&= \int_{-\infty}^{\infty} N\{t_a; t_{out}^*, \sigma_{out}^2\}\,N\{t_a; t_m, \sigma_m^2\}\,dt_a \\
&= N\{t_{out}^*; t_m, \sigma_{out}^2 + \sigma_m^2\}
\end{aligned} \qquad (11)
$$

3.3. Inside/Outside Probability *vs.* Measured Temperature

Substituting Equations (9) and (11) into (4) gives a closed form for the probability of being inside given a measured temperature and an interpolated outside temperature:

$$
p(\text{in}|t_m) = \frac{N\{\mu_{in}; t_m, \sigma_{in}^2 + \sigma_m^2\}}{N\{\mu_{in}; t_m, \sigma_{in}^2 + \sigma_m^2\} + N\{t_{out}^*; t_m, \sigma_{out}^2 + \sigma_m^2\}} \qquad (12)
$$

And $p(\text{out}|t_m) = 1 - p(\text{in}|t_m)$. This is a closed form solution for computing the probability of being inside or outside based on these parameters, all in ^{o}C:

t_m	Temperature measured on mobile device
$\sigma_m = 0.162$	Standard deviation of measured temperature
$\mu_{in} = 23$	Mean of expected inside temperature
$\sigma_{in} = 1.732$	Standard deviation of expected inside temperature
t_{out}^*	Outside temperature interpolated from weather stations
$\sigma_{out} = 2.545$	Standard deviation of outside temperature

To demonstrate Equation (12), we simulate two different people are using the technique, one inside and one outside. The ambient inside temperature is $t_{in} = \mu_{in} = 23$, and the person inside measures it as exactly this value. We look at the behavior of the equation as the outside temperature varies from -20 ^{o}C (-4 ^{o}F) to 40 ^{o}C (104 ^{o}F),

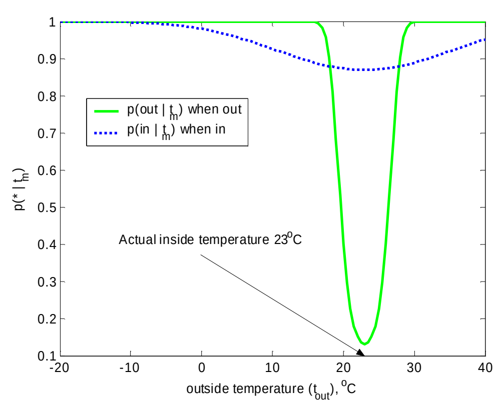

Figure 6: The solid line shows the computed probability of an outside person being outside as a function of outside temperature. The dotted line shows the computed probability of an inside person being inside.

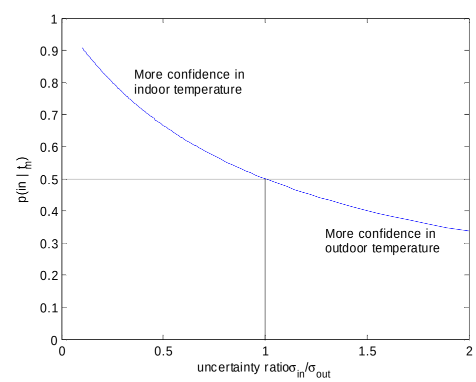

Figure 7: When the inside and outside temperatures are the same, probability favors the category whose temperature measurement has the lowest uncertainty.

which the person outside measures exactly. Figure 6 shows the results. The solid line shows the computed probability of the outside person being outside as the outside temperature changes. The probability remains high as long as the outside temperature is different enough from the inside temperature. As we would expect, the probability of being outside drops when the inside and outside temperatures are similar. The dotted line shows the computed probability of the inside person being inside. This probability also drops when the inside and outside temperatures are similar.

The simulation above confirms our intuition that the probabilities rise and fall as we expect. The amount that they rise and fall is a function of the temperature uncertainties. The probability of the inside person being inside never drops below 0.5, even when the inside and outside temperatures are equal. This is because the uncertainty of the outside temperature ($\sigma_{out} = 2.545$) is larger than the uncertainty of the inside temperature ($\sigma_{in} = 1.732$). Qualitatively, when the measured temperature is close to the expected inside temperature, the probability computation attributes more weight to the inside hypothesis, because the actual outside temperature can deviate more from the measured temperature than the inside temperature can. This is illustrated in Figure 7, which simulates an outside and inside temperature both equal to the mean inside temperature $\mu_{in} = 23$. When the ratio $\sigma_{in} / \sigma_{out}$ is low, confidence in the inside temperature is higher, which biases the probability toward being inside. At $\sigma_{in} = \sigma_{out}$, the probability of being inside is 0.5. When $\sigma_{in} / \sigma_{out}$ grows beyond 1.0, the probability of being inside drops below 0.5.

4. Demonstration Implementation

While our technique is more likely to be useful as a background process in a larger context inference system, we have also implemented a version to run it as a standalone program for demonstration purposes.

Figure 8 shows our technique exposed in a demonstration program. In the upper left the user gives his or her current location in the form of (latitude, longitude). Optionally, the user can input a U.S. zipcode which is converted to a (latitude, longitude) via one of our web services.

In the middle left box, the user clicks to call our outside temperature service based on the (latitude, longitude) given above. In the lower left box, the user clicks to get an ambient temperature measurement from a connected temperature sensor such as the one we use shown in Figure 1.

On the right, clicking "Infer" invokes the computation of Equation (12) and the drawing of the bars indicating the probabilities of inside and outside.

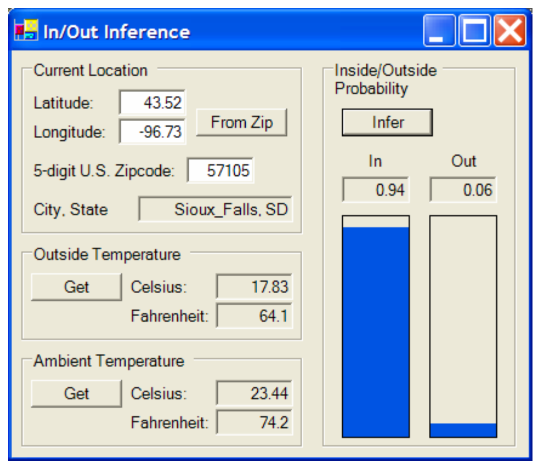

Figure 8: Our in/out inference technique is demonstrated with this program. The resulting probabilities are shown as blue bars.

5. Accuracy Tests

We tested our technique on temperature data downloaded from the American "National Virtual Data System" operated by NOAA[4]. This source gives temperature data recorded from over 700 U.S. weather stations. We used all the hourly data from the year 2003 for weather stations based in five U.S. cities.

Clearly our technique will be more accurate at distinguishing inside from outside when there is a larger temperature difference between the two states. In order to select cities conducive to a range of expected performance, we looked at candidate cities' yearly "heating degree days" (HDD) and "cooling degree days" (CDD) to assess how extreme their temperatures are. These quantities are based on the difference between a day's average temperature and a comfortable base temperature, which was 65°F for the data from [4]. If the average temperature is below the base temperature, the absolute difference is added to HDD, implying that a certain amount of inside heating would be required. These are summed for a year to give HDD, likewise for CDD when the average temperature is above the base temperature. Thus, a city with a high HDD is generally colder, and a high CDD is generally warmer. Of the U.S. cities in the data, we picked for testing Barrow, Alaska (highest HDD), Key West, Florida (highest CDD), San Diego, California (minimum sum of HDD and CDD), and Atlantic City, New Jersey (HDD and CDD were both close to the medians of all the cities). We also chose Seattle for testing, as this is our home city. We would expect our technique to work best where HDD or CDD are high, and not as well where HDD and CDD are both low, because our algorithm depends on the temperature inside buildings to be artificially controlled.

We used the hourly temperature data as the basis for computing the outside temperature in our evaluations. In our system, the outside temperature comes from our weather service, which has a Gaussian distribution shown in Figure 5 and up to an hour lag because our weather station download runs once per hour. We simulated both these effects in our evaluation, lagging *all* the outside temperatures by one hour and adding Gaussian noise with our measured $\sigma_{out} = 2.545$. Both the lag and the added noise tend to reduce the reported accuracy of our method, but we included them to make the test a more realistic representation of the inaccuracies inherent in accessing worldwide outside temperatures.

In evaluating the inference accuracy outside, we set the measured ambient temperature equal to the current (non-lagged) outside temperature and added Gaussian noise with $\sigma_m = 0.162$ representing the uncertainty of our thermometer.

In evaluating the inference accuracy inside, we assumed the building's heating and cooling policy was to let the inside temperature match the outside temperature when the outside temperature was within the ISO recommended range of $[20^o C, 26^o C]$ ($[68^o F, 78.8^o F]$), assuming that no heating or cooling would be used in this range. If the outside temperature exceeded this range, we clamped the inside temperature to whichever end of the ISO range was nearest the outside temperature. To this inside temperature we added Gaussian noise with $\sigma_m = 0.162$ to get the measured temperature from the thermometer.

The numerical results are summarized in Table 1. On average, each city had about 8715 temperature test points. The best performing city was Barrow, AK, which is unsurprising given its cold climate. Here the overall inference accuracy was 100%. Atlantic City, NJ, representing a typical American city, resulted in a correct inference 82.4% of the time for both inside and outside. Key West, FL, with the maximum CDD, gave the poorest inference performance with only 63.4% correct. We expected better, given the necessity of that city's inside cooling. However, despite the high CDD, this city has a mean temperature of 25.66 ^{o}C, which is within the ISO range. Barrow, which performed perfectly, has a mean temperature of -10.43 ^{o}C. The overall classification accuracy for the five cities tested was 81.0%. These results are shown graphically in Figure 9.

For our first study, our outside temperature measurements were simulated as coming from our temperature web service. This is inherently inaccurate due to interpolation and temporal lag. If instead outside temperature came from a nearby, networked thermometer, then performance improves significantly. We simulated this by reducing the outside temperature lag to zero and by reducing the outside temperature uncertainty to be the same as our measured uncertainty, *i.e.* $\sigma_{out} = \sigma_m = 0.162$. As shown in the right half of Table 1, the overall classification accuracy improves to 90.6% from 81.0% for the previous case.

One advantage of our probabilistic formulation is that the technique accurately reports its own confidence in the classification. Table 1 shows the mean of the computed $p(\text{in}|t_m)$ and $p(\text{out}|t_m)$ for each test city, and these fractions closely match the actual fractions of correct classifications. For instance, the inside test for Atlantic City, NJ worked correctly 79.3% of the time, and the mean value of $p(\text{in}|t_m)$ was 0.782. The shaded boxes in the table show that the average correct classification rate using the temperature web service was 81.0%, while the average classification probability was 0.816. For the outside thermometer case, the numbers are 90.6% and 0.905, respectively. Figure 10 shows a plot of the mean computed probabilities for $p(\text{in}|t_m)$ and $p(\text{out}|t_m)$ as a function of the actual correct classification fractions for each city. The points fall near a diagonal from (0,0) to (1,1), meaning that they are approximately equal. This realistic self-assessment of confidence is important for other reasoning modules that might depend on ours.

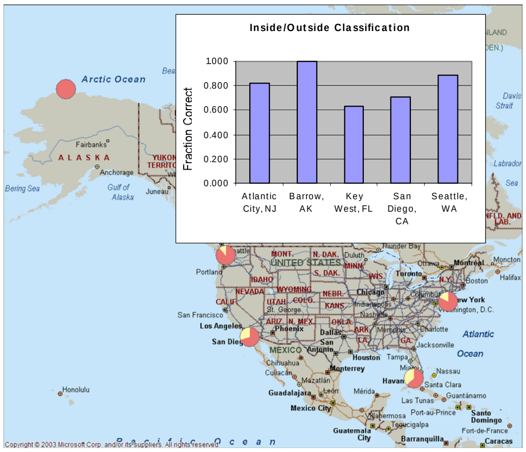

Figure 9: Classification accuracy varies from city to city. Pie chart on map shows fraction of correct classifications.

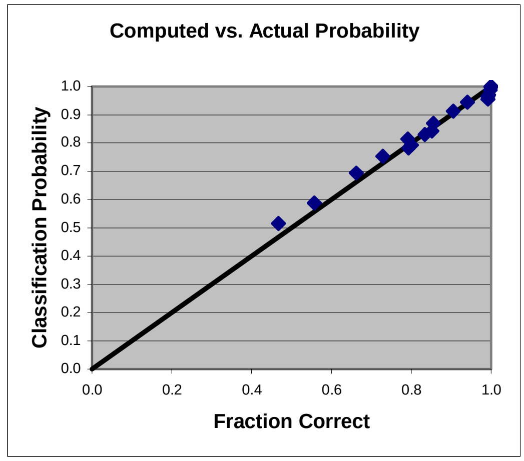

Figure 10: TempIO's fraction of correct classifications are accurately predicted by its own computed probabilities.

6. Conclusions

TempIO is an effective method for detecting if a device is inside or outside based on temperature. It requires only a digital thermometer, a network connection, and a rough estimate of the device's (latitude, longitude). Using a web service to find the outside temperature, TempIO is about 81% accurate in classification based on weather data from five U.S. cities. Using a nearby outside thermometer, the classification accuracy grows to about 91%. By using Bayes rule, TempIO's classification probabilities closely match its actual accuracy, meaning that it faithfully assesses its own confidence in its results.

We envision TempIO to be useful for turning off a GPS receiver when inside, for adding metadata to digital photos, and as a component of higher-level context inference for ubiquitous computing.

			With Outside Weather Service						With Outside Thermometer					
			Fraction Correct			Computed Probability			Fraction Correct			Computed Probability		
City	HDD	CDD	In	Out	Mean	p(in)	p(out)	Mean	In	Out	Mean	p(in)	p(out)	Mean
Atlantic City, NJ	5113	935	0.793	0.855	0.824	0.782	0.870	0.826	0.791	0.994	0.893	0.814	0.970	0.892
Barrow, AK	19873	0	0.999	1.000	1.000	0.998	1.000	0.999	1.000	1.000	1.000	1.000	1.000	1.000
Key West, FL	62	4830	0.467	0.800	0.634	0.515	0.793	0.654	0.662	0.992	0.827	0.694	0.955	0.825
San Diego, CA	1063	866	0.557	0.851	0.704	0.588	0.842	0.715	0.728	0.992	0.860	0.754	0.965	0.860
Seattle, WA	4797	173	0.834	0.940	0.887	0.830	0.944	0.887	0.905	0.998	0.952	0.912	0.987	0.950
All			0.730	0.889	0.810	0.743	0.890	0.816	0.817	0.995	0.906	0.835	0.975	0.905

Table 1: Performance of inside/outside classification in five test cities. HDD and CDD are annual "heating degree days" and "cooling degree days", respectively, averaged over 1971-2000. The columns under "With Outside Weather Service" use interpolated temperatures from our web service for the outside temperature. The columns under "With Outside Thermometer" assume outside temperature came from a nearby thermometer. The "In" and "Out" columns show the fraction of correct classifications, and the "p(in)" and "p(out)" columns show the mean computed probabilities from (12). The overall classification accuracy using the weather service is 0.810, while using the outside thermometer gives a classification accuracy of 0.906.

References

1. NOAA, METAR Data Access, *http://weather.noaa.gov/weather/metar.shtml*.
2. Fred C. Collins, J. and P.V. Bolstad. A Comparison of Spatial Interpolation Techniques in Temperature Estimation. in Third International Conference/Workshop on Integrating GIS and Environmental Modeling. 1996. Santa Fe, New Mexico, USA.
3. Williams, J.L., Gaussian Mixture Reduction for Tracking Multiple Maneuvering Targets in Clutter. 2003, Air Force Institute of Technology: Wright-Patterson Air Force Base, Ohio.
4. NOAA, Local Climatological Data Publication, *http://nndc.noaa.gov/?http://ols.nndc.noaa.gov/plolstore/plsql/olstore.prodspecific?prodnum=C00128-PUB-S0001*.