

SunSpot: Exposing the Location of Anonymous Solar-powered Homes

Dong Chen, Srinivasan Iyengar, David Irwin, and Prashant Shenoy
University of Massachusetts Amherst

ABSTRACT

Homeowners are increasingly deploying grid-tied solar systems due to the rapid decline in solar module prices. The energy produced by these solar-powered homes is monitored by utilities and third parties using networked energy meters, which record and transmit energy data at fine-grained intervals. Such energy data is considered anonymous if it is not associated with identifying account information, e.g., a name and address. Thus, energy data from these “anonymous” homes is often not handled securely: it is routinely transmitted over the Internet in plaintext, stored unencrypted in the cloud, shared with third-party energy analytics companies, and even made publicly available over the Internet. Extensive prior work has shown that energy consumption data is vulnerable to multiple attacks, which analyze it to reveal a range of sensitive private information about occupant activities. However, these attacks are useless without knowledge of a home’s location.

Our key insight is that solar energy data is not anonymous: since every location on Earth has a unique solar signature, it embeds detailed location information. To explore the severity and extent of this privacy threat, we design SunSpot to localize “anonymous” solar-powered homes using their solar energy data. We evaluate SunSpot on publicly-available energy data from 14 homes with rooftop solar. We find that SunSpot is able to localize a solar-powered home to a small region of interest that is near the smallest possible area given the energy data resolution, e.g., within a $\sim 500\text{m}$ and $\sim 28\text{km}$ radius for per-second and per-minute resolution, respectively. SunSpot then identifies solar-powered homes within this region using crowd-sourced image processing of satellite data before applying additional filters to identify a specific home.

CCS Concepts

•Computing methodologies \rightarrow Model development and analysis; Model verification and validation;

Keywords

Solar Modeling, Net Metering, Privacy

1. INTRODUCTION

The number of solar-powered homes is rapidly increasing due to a steep decline in solar module prices. To illustrate, the cost of solar energy in $\$/\text{W}$ dropped an estimated 50% from 2008 to 2013 [2],

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

BuildSys '16, November 16 - 17, 2016, Palo Alto, CA, USA

© 2016 Copyright held by the owner/author(s). Publication rights licensed to ACM. ISBN 978-1-4503-4264-3/16/11...\$15.00

DOI: <http://dx.doi.org/10.1145/2993422.2993573>

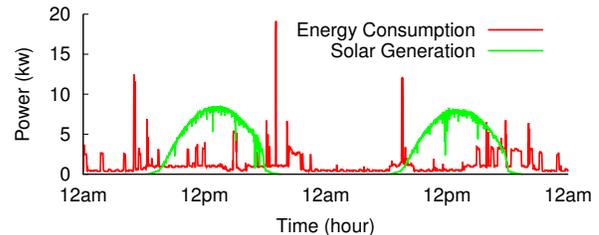


Figure 1: Example data from solar-powered home that is making its 1Hz solar generation and energy usage publicly available on the the Internet under the assumption of anonymity.

resulting in a 418% increase in solar energy capacity in the United States over the same period [8]. These trends are accelerating due to the Swanson effect [22], which observes that solar module prices tend to decrease 20% for every doubling in manufacturing volume. Solar power prices have now fallen below retail electricity rates in many areas, further increasing the incentive to install solar modules [27]. Importantly, utilities and third-parties monitor the energy produced by solar-powered homes using networked energy meters, which record and transmit energy data at fine-grained intervals.

Such energy data is generally considered anonymous if it is not associated with identifying account information, e.g., a name and address. Thus, energy data from these “anonymous” solar-powered homes is often not treated as sensitive: instead, it is routinely transmitted over the Internet in plaintext, stored unencrypted in the cloud, shared with third-party energy analytics companies, and even made publicly available. For example, Figure 1 shows a screenshot of 1Hz energy data an anonymous solar-powered home has made publicly available on the Internet via a networked energy meter, such as the TED, eGauge, BrulTech, or Enphase Envoy.

These meters connect to the Internet and upload energy data to the cloud in real time, where it is then stored to enable queries on archival data. Solar installers typically add networked meters to enable homeowners to monitor energy generation and consumption via web dashboards or smartphone applications. For simplicity, in many cases as in Figure 1, accessing the data does not require a password, as there is an assumption the data is anonymous and cannot be associated with a particular home. The example in Figure 1 is from one of the 28,000 anonymous homes we have found uploading solar generation and energy consumption data to the public Internet. While the example makes the data publicly available for simplicity, similar data is also being intentionally gathered and released by various research institutions to support energy analytics research. As above, these datasets often include detailed solar and energy usage data from thousands of volunteer anonymous homes.

While users may choose to not install (or securely configure) the meters above, they are *forced* to allow utilities to monitor their energy usage. In addition, to receive reimbursements for solar energy generation, some states also require users to upload their utility energy data to an external database managed by a third party [19]. This energy data is becoming increasingly detailed, as utilities employ “smart” meters that record energy usage at fine-grained intervals.

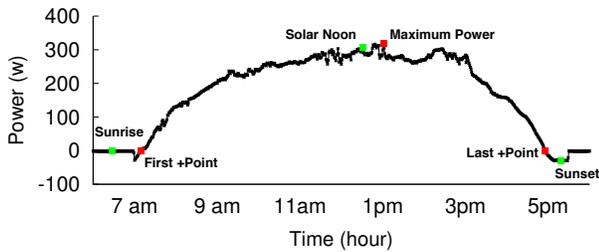


Figure 2: The start, stop, and peak of solar generation (red) approximates the time of sunrise, sunset, and solar noon (green).

Current smart meters monitor energy usage on the order of minutes [7] with next-generation meters expected to monitor on the order of seconds [24]. In the U.S., utilities have deployed >50 million smart meters [10], and are rapidly accumulating smart meter data, which they may permanently archive for later analysis.

A plethora of startups have now arisen to analyze these vast archives of utility energy data, ostensibly to make energy-efficiency recommendations [3, 16, 18]. Prior research has demonstrated the ability to learn a variety of insights into private user behavior by analyzing their energy data [15]. For example, energy data indirectly leaks occupancy [5, 12], which may reveal whether a home’s occupants: i) include a stay-at-home spouse, ii) keep regular working hours and daily routines, iii) frequently go on vacation, or iv) regularly eat out for dinner. Energy data can also reveal load power signatures—changes in power unique to a device—for specific appliance brands and models. These behavioral insights and appliance details are valuable to companies in profiling homes and directing advertising campaigns, and may also be exploited by tech-savvy criminals. Thus, some contend that energy data will eventually be worth more than the energy consumed to generate it [17].

Users and utilities commonly provide energy data to the energy analytics companies above under the assumption the data is anonymous. In many cases, users do not realize their energy data leaks side-channel information. Utilities typically anonymize any energy data they share with third-parties by removing account names and addresses, as suggested by the U.S. Department of Energy’s recently released Voluntary Code of Conduct (VCC) for managing user energy data [26]. Importantly, the VCC *does not* require user consent to release anonymized energy data with names and addresses stripped. Consent is likely not required because the energy analytics above do not reveal location, which prevents third-parties from associating private behavior above with a specific home.

Our key insight is that solar energy data is not anonymous: since every location on Earth has a unique solar signature, e.g., a unique sunrise, sunset, and solar noon time, it embeds detailed location information. While there is substantial prior work on estimating solar energy output based on a home’s location, we know of no work that does the reverse—estimating the location based on solar output. The localization threat means home energy data that includes solar generation is never anonymous.

As one example of this threat, an attacker could determine when to burglarize the anonymous home in Figure 1 by first determining its occupancy pattern from its consumption data (in red) using existing techniques [5, 12], and then analyzing its solar signature (in green) to determine the home’s location. As a result, users and utilities should treat such data as highly sensitive by, in particular, not making it publicly available on the Internet or releasing it to third-parties without user consent. To explore the severity and extent of this privacy threat, we design SunSpot, a system for localizing an anonymous solar-powered home by analyzing its solar energy data. Exposing and evaluating this threat is critically important in informing evolving policies by DOE and others for managing

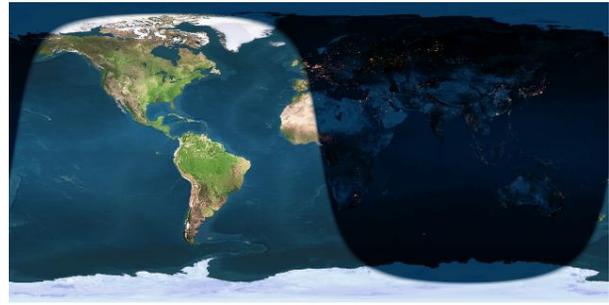


Figure 3: Sunlight map of the Earth.

anonymous energy data, and in emphasizing to users and utilities the need to securely handle energy datasets that include solar generation. In doing so, this paper makes the following contributions.

Localization Challenges. We highlight numerous challenges to localization from solar energy data, as a solar module is a highly imprecise sensor for tracking the sun. Solar energy data is affected by numerous unknown variables, including a home’s local climate, e.g., frequency of cloud cover and temperature variations, physical characteristics, e.g., tilt/orientation, topography, shading from nearby structures, etc., and properties of the electrical system, e.g., variations in grid voltage, choice of wiring and inverter(s), etc.

SunSpot Design. We design SunSpot, which localizes a solar-powered home to a small region by exploiting multiple insights derived from the regularity in the Earth’s orbit. We leverage crowd-sourced image processing on publicly-available satellite data to identify potential homes in the area with visible solar modules. SunSpot significantly reduces the search area by filtering out areas without man-made structures, and may then apply additional filters, e.g., by matching solar output to module size or local weather patterns, to hone in on a specific solar-powered home.

Implementation and Evaluation. We implement and evaluate SunSpot on publicly-available energy data at both per-second and per-minute resolution from 14 solar-powered homes. We find that SunSpot localizes a solar-powered home to near the smallest possible region given the energy data resolution, e.g., within a $\sim 500\text{m}$ and $\sim 28\text{km}$ radius for per-second and per-minute resolution, respectively. SunSpot then leverages Amazon’s Mechanical Turk at a cost of $\$13.60/\text{km}^2$ to identify a specific home, after reducing the search space by filtering out regions without man-made structures, which eliminates on average >97% of the search area in the U.S.

2. LOCALIZATION CHALLENGES

SunSpot assumes an anonymous solar-powered home at an unknown location equipped with a networked energy meter that monitors energy generation over time. Given this solar energy data, SunSpot’s objective is to infer a location—a latitude and longitude—where the data originates. Note that we focus exclusively on localizing the source of solar energy data, and not “net meter” data, which is the sum of a home’s solar generation and energy consumption. Energy analytics companies have already developed solar disaggregation techniques, which analyze net meter data to separate solar data from consumption data [14], and are actively applying them to utility smart meter data [3, 11]. While our techniques may be used in conjunction with solar disaggregation to localize based on net meter data, we consider solar disaggregation a separate research problem that is out of scope. In addition, as discussed in Section 1, there are already thousands of solar-powered homes, including the home in Figure 1, that are separately exposing their solar generation and their energy consumption data.

The basic principle for localizing a solar-powered home from its

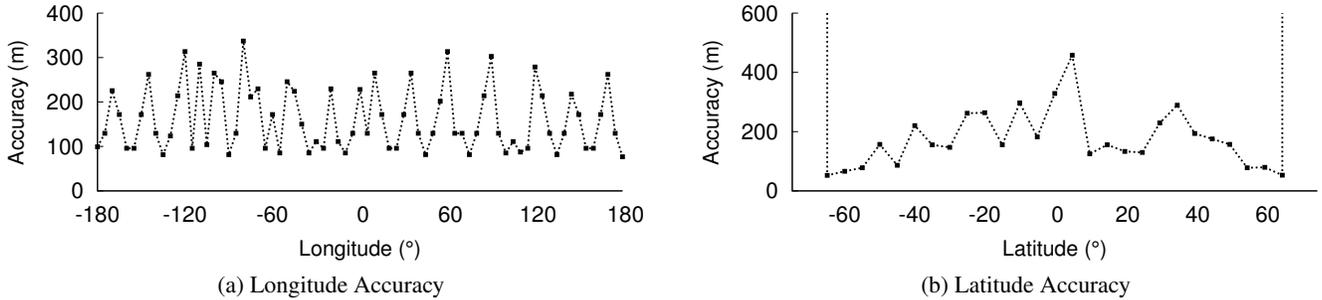


Figure 4: We accurately derive location from sunrise and sunset times using existing online APIs that perform the reverse operation.

solar energy data is straightforward. On a clear sunny data, solar generation data reveals a location’s unique *solar signature*, which derives from the sun’s position in the sky at a particular location and time and determines the amount of solar radiation that strikes the Earth. In particular, a location’s unique solar signature dictates a unique time of sunrise, sunset, and solar noon (see Figure 2), which correspond to the times when a solar system’s generation starts, stops, and peaks each day, respectively. SunSpot leverages this information to infer the location where solar energy data originates.

2.1 Deriving Location from the Sun

Given the sun’s importance to life on Earth, astronomers can derive its movements with incredible precision. For example, the PSA algorithm [4] provides the sun’s position, i.e., its azimuth and elevation angles, in the sky to within 0.0083° at any location, given its latitude and longitude, at any time of the year. Open-source code and online APIs are available that implement the sunrise/sunset algorithm, which provides precise sunrise and sunset times (to the second) given a location’s latitude and longitude [20, 21].

Interestingly, while technically feasible, there are no commonly available open-source libraries or online APIs that perform the reverse operation, by computing a location from the sunrise and sunset times. Unfortunately, the PSA and the sunrise/sunset algorithm above are not reversible, since they both use trigonometric functions at multiple stages that are not one-to-one, i.e., their inverse yields multiple solutions. Instead, the algorithms for deriving location from sunrise/sunset events are much more obscure, as they are typically only used for celestial navigation of ships without electronic navigation [25]. Unlike the open-source code and online APIs above, these localization algorithms widely published in textbooks do not compensate for the slight irregularities in the Earth’s shape and orbit that are required for high precision.

However, as a prerequisite to localizing solar energy data, SunSpot requires a precise algorithm for determining a location based on its sunrise and sunset times. Due to the issues above, rather than implement and refine published algorithms, we develop an approach that uses available APIs, which only work in the opposite direction by computing sunrise/sunset time given a latitude and longitude, as tools to conduct a binary search for a location. Note that in the paper we use UTC time to eliminate time zone issues.

Deriving Latitude. To determine a location’s latitude, we observe that all locations at the same latitude have the same daylength, i.e., the duration between sunrise and sunset, on each day. We also observe that, the daylength gets shorter the further north the latitude in the fall/winter, and gets longer the further north the latitude in the spring/summer. To illustrate, Figure 3 shows a sunlight map of the earth in the northern hemisphere’s winter, where daylength becomes shorter moving from south to north. We leverage this insight to conduct a binary search to find a latitude that yields our desired daylength, given a sunrise and sunset time. That is, we pick any

longitude value and then compute the daylength using the online APIs for -90° , 0° , and 90° latitude. We then select the region, either $[-90^\circ, 0^\circ]$ or $[0^\circ, 90^\circ]$, that includes the desired daylength. We then compute the daylength for the mid-point of that interval, and repeat the process. We terminate the search when the latitude computed at the next step does not significantly change.

Deriving Longitude. We perform a similar procedure to compute a location’s longitude. Longitude is uniquely determined by the time of solar noon, when the sun is at its highest point in the sky, which is always the mid-point between the sunrise and sunset times. In this case, we pick any latitude and then compute solar noon using the online APIs for both 0° and $\pm 180^\circ$ longitude. We then select the region, either $[0^\circ, -180^\circ]$ or $[0^\circ, 180^\circ]$, that includes our desired solar noon. As above, we compute solar noon for the mid-point of the selected region, either 90° or -90° , and repeat the process until the longitude computed at the next step does not change.

Note that, by searching based on daylength and solar noon, the two procedures above are independent of each other. That is, computing the longitude does not depend on knowing the latitude or vice versa. We evaluate our approach across the full range of latitudes and longitudes above using existing online APIs [21], as shown in Figure 4. Figure 4(a) shows that our derived longitude is always within 400m of the actual location’s longitude. Longitude accuracy is a function of the latitude, such that higher latitudes enable higher accuracy at the same data resolution. The Earth’s rotation speed decreases by the cosine of the latitude, such that the speed at X° latitude is $465 \times \cos X^\circ$ m/s. As a result, the maximum precision possible at the equator with second-level data is 465m, and the accuracy possible with minute-level data is 27.9km. Figure 5 plots the maximum longitude precision possible for second- and minute-level resolution data across all latitudes. To make both lines visible, we plot second-level data based on the left y-axis and minute-level data based on the right y-axis.

Similarly, Figure 4(b) shows that our computed latitude is always less than 500m from the actual location. The abrupt increases at $\pm 66.56^\circ$ ¹ indicate regions near the poles where the sun does not rise or set. We ran this experiment on data from June 21st, 2015 (the summer solstice) where the half of the Earth lit by the sun is maximally misaligned with the poles. Note that the solstices represent the days that yield the most accurate results for latitude. Latitude accuracy changes over the course of the year, as shown in Figure 6. Since, on the equinoxes (September 22nd and March 20th), every location experiences 12 hours of daylight, it is impossible to distinguish a location’s latitude from daylength.

2.2 Challenges to using Solar Energy Data

Our approach above derives a location from a known sunrise and sunset time—or equivalently the daylength and solar noon—on a particular day. A naïve approach to compute location from solar en-

¹This latitude is equal to 90° minus the Earth’s tilt of 23.44° .

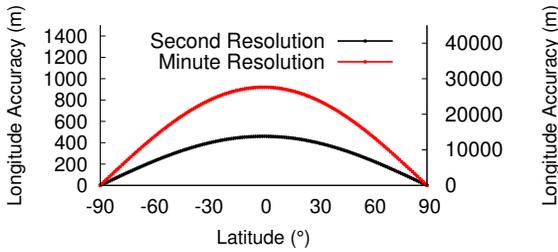


Figure 5: The precision possible for computing longitude from solar noon is a function of the latitude and the data resolution.

ergy data is to simply use the times for the first and last positive solar generation of the day as the sunrise and the sunset times, respectively. SunSpot can use these times to directly estimate daylength and solar noon, and then provide these estimates as input into the algorithm above. However, this approach is inaccurate—on the order of hundreds to thousands of kilometers—because a solar system is a highly imprecise sensor for numerous reasons. For example, even a few minutes of inaccuracy in solar noon can yield massive errors, as each minute of error translates to roughly 27.9km error in longitude, as mentioned above. Figure 2 shows that even on a seemingly ideal day, sunrise, sunset, and solar noon often do not precisely align with the start, stop, and peak of solar generation, respectively. Below we describe the reasons for this error.

Atmospheric Conditions. Solar output depends on changing environmental conditions, namely solar irradiance. For a flat stationary solar deployment, the maximum solar irradiance (in W/m^2) is proportional to the sun’s position in the sky. However, atmospheric effects alter the maximum solar irradiance. These effects include not only the presence of visible clouds, but also other conditions, including humidity, rain, dust, snow, pollution levels, etc. As a result, on a cloudy day, the start and stop of solar generation may be tens of minutes after and before sunrise and sunset, respectively.

Generation Inefficiency. Solar modules are not 100% efficient at converting solar radiation to power, but instead range in efficiency from 15-25%. Due to this inefficiency, even under ideal conditions with no clouds, the start and stop of solar generation each day will not precisely align with sunrise and sunset, as shown in Figure 2. Solar module efficiency also decreases as the temperature increases. Thus, the lag in detecting the first positive generation after sunrise (and the last positive generation before sunset) varies with temperature. Temperatures may vary significantly over the day (from morning to evening) and year (from winter to summer).

Shading from Nearby Objects. Sunrise and sunset times are derived assuming no topographical effects, i.e., the location and its surroundings are at sea level. This is only true in the middle of the ocean. In reality, the surrounding landscape dictates the horizon. For example, in a valley, the sun will rise from behind the mountains later and set behind them earlier than the official sunrise and sunset times. The opposite will occur at the top of a mountain. Solar deployments, especially on rooftops, are also often obstructed by nearby buildings and trees. The impact of these effects is not consistent, but will vary over time, e.g., when trees lose their leaves.

Physical Properties. The physical properties of a module, namely its tilt and orientation, also affect energy generation. The power output of a stationary deployment oriented toward the equator, e.g., south in the northern hemisphere and north in the southern hemisphere, is proportional to solar irradiance, which is a function of the sun’s position in the sky. However, many deployments are not perfectly oriented toward the equator, and may also be tilted to varying degrees. The equation below computes solar output as a function of the sun’s position in the sky, and modules’ tilt and orientation.

$$S_p = S_i [\cos(\alpha) \sin(\beta) \cos(\psi - \Theta) + \sin(\alpha) \cos(\beta)] \quad (1)$$

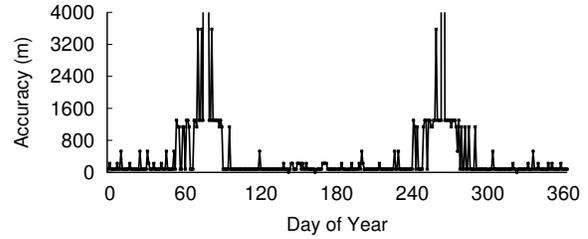


Figure 6: The accuracy of deriving latitude from daylength varies over the year and is least accurate near the equinoxes.

Here, S_i is the intensity of solar radiation that strikes a flat module, while S_p is the amount of solar radiation that strikes an actual module, given the module’s azimuth and tilt angles (ψ and β , respectively), as well as the sun’s azimuth and elevation angles (Θ and α , respectively). Figure 7 graphically depicts how the orientation affects the output of a module (in the northern hemisphere). An ideal module oriented south ($\psi = 180^\circ$) will experience its maximum production at solar noon (when the sun is at its highest point in the sky, maximizing solar radiation). In contrast, an ideal module with more of an eastward orientation will shift the generation curve earlier, such that its maximum production is earlier than solar noon. The more eastwardly the orientation, the earlier the maximum production point and the earlier the day’s first and last generation times. A westward orientation has the opposite effect.

Unlike changes in the orientation, changes in the tilt do not affect either the time of maximum generation or the time of first or last positive generation. However, they do reduce the magnitude of the maximum generation and, thus, result in a more gradual rise and fall of the generation curve. Note that the relationships above dictate the output for a solar system where all modules are tilted and oriented in the same way. If there are multiple modules with different tilts and orientations their output is the sum of each module’s individual production based on its own tilt and orientation (assuming each module has its own microinverter, as discussed below).

Electrical Characteristics. A solar deployment’s electrical system also affects its output. For example, the output of modules wired in series is dictated by the individual module generating the least current. Each individual module’s output is also dictated by its IV curve, which defines the amount of current (and power) a module generates at different voltages. The IV curve of a multi-module deployment connected to a single inverter is a complex function of the IV curves of all the modules and how they are wired, e.g., in series or parallel. As with module efficiency, the shape of the IV curve also varies with temperature and solar irradiance. Inverters actively vary their operating voltage to search for the maximum power point on this complex aggregate IV curve as conditions change, which may lead to periods of operation below the maximum power point. Rather than connect multiple modules to a single inverter, deployments may also attach a microinverter to each module. In this case, each microinverter independently optimizes its module’s maximum power point. Thus, the same system using microinverters will generate a different energy profile than when using a single inverter.

Meter Accuracy. Ultimately, solar energy data derives from meters that sense its generation. These meters have varying levels of accuracy, typically ranging from 0.5% to 2%, depending on whether they are certified as utility- or consumer-grade. Energy meters are typically placed in front of the inverter and measure AC power. As a result, the power they record is a function of, not only the current generated by the modules, but also the grid’s voltage. While RMS grid voltage in the U.S. is 120V, it may vary by $\pm 5\%$ based on current standards. Thus, recorded power generation will also vary in proportion to these voltage fluctuations.

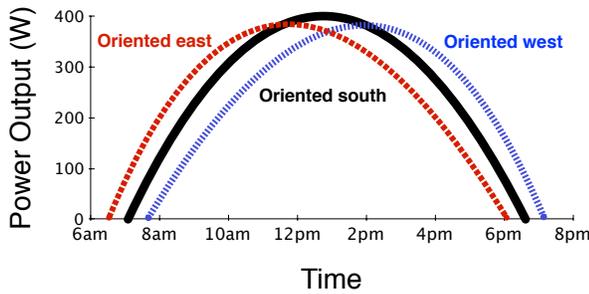


Figure 7: Depiction of how solar generation changes based on solar module orientation (in the northern hemisphere).

2.3 Summary

The inaccuracy in solar energy data caused by the effects above varies across locations. For example, the power generated by a deployment in Southern California (which has few temperature variations and cloudy days) that has few obstructions and all modules oriented towards the equator (with the same tilt) will more closely reflect the sun’s path than a deployment in a location with a highly variable climate, many obstructions, multiple modules with different orientations and tilts, unstable grid voltage, etc. In essence, the more efficient a solar deployment is at generating power, the closer it tracks the sun’s position in the sky, and the more susceptible it is to localization. This general principle—that the more energy-efficient a system, the more vulnerable it is to leaking information via energy data—has been observed in other contexts [5, 12].

3. SUNSPOT DESIGN

The effects from the previous section are often significant—even for the most efficient deployment—and impossible to accurately model without knowing details of a solar installation, e.g., its location, tilt/orientation, wiring, etc. Thus, accurate localization from single day’s solar data (or even a few days or weeks) is challenging, and impossible if the time period is near the equinox (since all locations have a similar daylength near the equinox). However, since utilities, third parties, and current networked energy meters have archives of solar energy data, as discussed in Section 1, SunSpot leverages data over multiple days to mitigate inaccuracy from any single day’s data. Note that Sunspot *does not require many months of data*, and can operate on even a few weeks of data, as long it includes some clear sunny days. However, as we discuss, for inferring latitude, SunSpot does require data from a separate set of days in the fall/winter and the spring/summer. Of course, in general, data over a longer period increases both accuracy and confidence.

Similar to the approach in Section 2, SunSpot works by first inferring a location’s longitude and latitude separately. SunSpot uses this inferred location to identify a region of interest, since solar energy data alone is not accurate enough to precisely identify a home’s location. After identifying a region of interest, SunSpot then uses image processing on publicly-available satellite data within the region to identify candidate homes with visible solar systems. Finally, SunSpot applies filters to further prune this set of homes. Figure 8 depicts SunSpot’s pipeline of operations.

3.1 Identifying a Region of Interest

Inferring Longitude. To infer longitude, SunSpot leverages the time of solar noon. Solar noon is the wall clock time on any given day where the sun is at its highest point in the sky at a specific location. The clock time of noon, e.g., 12pm, often differs from solar noon, as clock times are based on the local time zone, where a large region within the same time zone has the same wall clock time. In contrast, each location within that time zone has a different solar

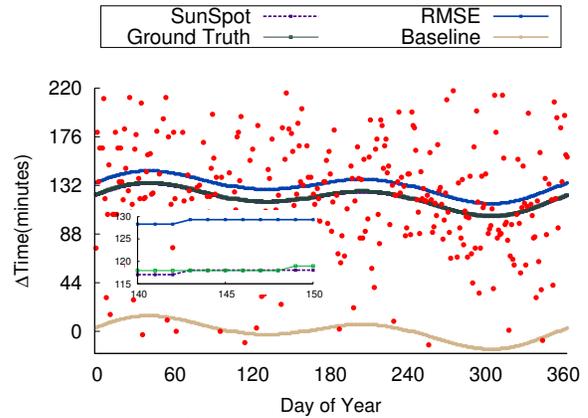


Figure 9: SunSpot finds the time where the EoT curve overlaps the most maximum solar generation points over the year. The zoomed-in inset shows SunSpot curve nearly precisely overlaps the ground truth EoT curve. The y-axis is the change in solar noon time from January 1st.

noon time, depending on when the sun rises to its highest point, i.e., nearest zenith, at that specific location. Further, as the Earth orbits the sun, the time of solar noon for each location changes gradually over the course of the year. *SunSpot leverages the fact that the day-to-day changes in solar noon across a year are consistent at every location on Earth.* In particular, the ~31 minutes of movement in solar noon are the same at every location and dictated by the Equation of Time (EoT), which is imprinted on sundials to reconcile the difference between apparent solar time (which tracks the actual movement of the sun each day) and mean solar time (which tracks an “average” sun where noon is always 24 hours apart).

Figure 9 shows the EoT (the bottom line) at 0° latitude over the course of a year, where the y-axis is the change in solar noon time assuming we set solar noon time on January 1st to zero. For an efficient solar system (oriented towards the equator) on a sunny day, *solar noon should correspond to the time of maximum generation.* Thus, the change in the time of maximum generation should precisely track the EoT, regardless of a deployment’s location. Note that using solar noon should mitigate the effect of shading from obstructions, as only the most inefficient deployments would be shaded at solar noon. Of course, due to the other effects in the previous section, the maximum time of generation does deviate significantly from that predicted by the EoT over the year. Figure 9 includes a scatterplot of the time of maximum energy generation (with energy data at one minute resolution) for a representative home. The scatterplot shows that there are numerous and significant deviations in the time of maximum generation across the year.

Since the EoT is the same for every location on Earth, SunSpot knows the shape of the EoT curve it must “fit” to the data: it need only shift up and down the y-axis to determine where to best place it. In the figure, we use 0° latitude as the baseline EoT. To “fit” the EoT to the data, SunSpot assumes that on ideal (sunny) days the time of maximum generation should often track the EoT, while on non-ideal (cloudy) days the time of maximum generation will be random, e.g., it might be before or after solar noon depending on the weather. Given this assumption, we place the EoT curve at the spot on the graph where it overlaps the most data points within some tolerance, e.g., ±1 minute. In Figure 9, the placed curve (in dark violet) nearly precisely overlaps the actual ground truth EoT curve for the location—the bottom of the zoomed-in inset shows the two overlapping curves. We tried various other methods for placing the EoT curve, such as placing it to minimize the root mean squared error, or RMSE (in blue), but found that this and

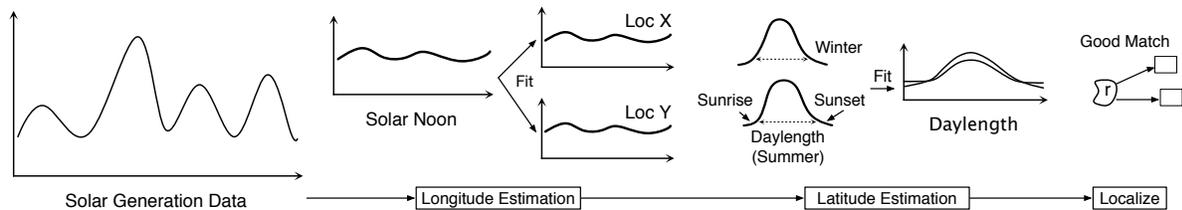


Figure 8: Pipeline of operations SunSpot uses to infer the location of a solar array from its energy data.

similar approaches were not as accurate. This likely occurs because using RMSE assumes the magnitude of the deviations above and below solar noon are the same. However, the local climate may cause the magnitude of these deviations to be biased towards the morning or afternoon. For example, frequent fog in the mornings might increase the probability of the time of maximum generation often occurring much later than solar noon (as in the figure).

After placing the EoT on the graph, SunSpot then infers longitude by taking solar noon time for any day on the EoT and applying our algorithm from Section 2 to compute the longitude. Note that solar noon time for any day on the same EoT curve will yield exactly the same longitude. In experimenting with the basic approach above, we found that the time of maximum generation often deviates from solar noon on many sunny days that appear ideal. This likely occurs due to small variations in the various factors listed in Section 2.2, such as slight variations in grid voltage. As a result, we extend this approach by using the top k times of maximum generation. That is, we sort the data points (for any resolution) by their energy generation and plot the top k data points on the graph. Figure 10 shows how longitude accuracy changes based on the value of k for 1Hz data. We typically use $k = 3$, as we have empirically found that it performs best on a large set of solar deployments.

Note that our basic approach above assumes a deployment with a single set of modules oriented towards the equator. Since most deployments strive for efficiency, they are typically oriented towards the equator, which mitigates the impact of our orientation assumption in practice. However, we can extend the basic approach to account for tilts and orientations, as described below.

Based on the PSA algorithm [4] and Equation 1, we can compute a modified EoT that tracks the movement in solar radiation incident on a module of different orientations. The PSA algorithm gives the sun’s position in the sky at any time for any location, and Equation 1 computes the expected solar generation given the sun’s position and a deployment’s tilt and orientation. The PSA algorithm requires a latitude and a longitude: we use the latitude we infer below (which is derived independently of the longitude) and we choose any longitude, since the EoT and our modified EoT will have the same shape at any location. We then compute multiple modified EoT curves for many different orientations, e.g., every 5° from 0° to 180° , and place them based on the procedure above. We choose the curve (and orientation) with the most overlapping points as above, and use Equation 1 to compute the difference between the point of maximum generation and the real solar noon for a module with that orientation. After inferring solar noon (on any day), as above, we infer longitude by taking this solar noon and using our algorithm from Section 2 to compute a longitude.

While it may be possible to adjust for other factors that contribute to inaccuracy, e.g., multiple modules with different tilts and orientations, etc., we leave these optimizations as future work.

Inferring Latitude. To infer latitude, we observe that the length of a day—the time from sunrise to sunset—varies with latitude. For example, in the summer, the daylength gets longer as we go from the equator to the north pole and shorter as we go from the equator

to the south pole. The situation is reversed in the winter. Thus, latitude is a function of the daylength—the length of the day at a location and how the daylength changes over the course of the year depends on its latitude. To compute the daylength, we must estimate the sunrise and sunset time. SunSpot estimates sunrise and sunset by simply taking the first and last positive points of generation in the day, respectively. However, as discussed earlier, this approach will *always* result in a significantly shorter daylength than the actual daylength. We have found the difference to be on the order of tens of minutes for sunrise and sunset, resulting in latitude errors that approach 1000 kilometers.

SunSpot mitigates this error by leveraging the insight above: *namely, in the fall/winter, daylength becomes shorter moving north, and in the spring/summer, daylength becomes shorter moving south.* As a result, using the approach above, in the fall/winter, SunSpot will always infer a location north of the actual location, and in the spring/summer, SunSpot will always infer a location south of the actual location. SunSpot splits the difference between these two errors by computing latitude separately for each half of the year, and then averaging them. This approach is surprisingly accurate, reducing latitude errors from near 1000km to less than 20km for 1Hz energy data. The accuracy improvement derives, in part, from the technique’s ability to mitigate the impact of orientation, shading from structures, etc., as these characteristics affect the inaccuracy in the fall/winter and spring/summer in a similar way. Thus, when averaging, the effects largely cancel each other out.

The approach above requires some energy data from the two different halves of the year. We generally use a few months worth of data to mitigate the inaccuracy of data from any single day. Since daylength is a function of latitude, we can derive a daylength curve that dictates the daylength over a year. Of course, unlike the EoT, the shape of the curve is dependent on the latitude, requiring SunSpot to find the latitude curve that best “fits” the data. In addition, just as above, the curve that minimizes the root mean squared error is inaccurate, as it is “pulled down” by many short daylengths due to cloudy days. Instead, SunSpot defines the best fit as the daylength curve that represents the tightest upper bound on the data. While the data point that defines this tightest bound represents the most ideal day of the year (in that it is the longest day we record relative to the daylength curve), it will still always be shorter than the actual daylength, since a solar deployment cannot generate power until strictly after the sun has risen (and will stop generating strictly before the sun has set). Thus, the tightest bound will never define a longer daylength than the actual one. We find this upper bound separately for data in the spring/summer and fall/winter.

Figure 11 shows three daylength curves for an example home, representing the upper-bound on the spring/summer data and the fall/winter data, as well as the daylength curve associated with the average of the two derived latitudes. In computing the tightest bound, we adjust for outliers due to sensing errors by removing the data point that defines the tightest bound of the daylength curve, and then find the next tightest bound. We then compare the distance between these two latitudes, and if it is less than a threshold

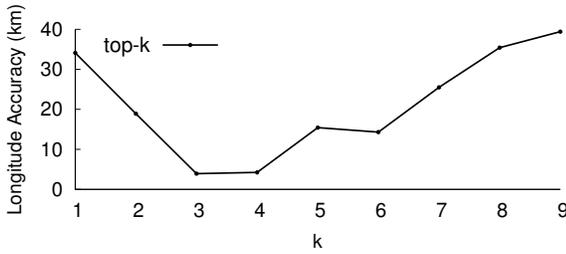


Figure 10: Longitude accuracy depends on the top k points of generation we include in the scatterplot when fitting the EoT.

distance n we stop, but if not we iterate again. We continue until the latitude does not change significantly. This approach ensures that at least two points over the year define near the same daylength curve. The figure shows how far apart the tightest bounds of the daylength curve are in each half of the year. The difference typically translates to near 1000km. However, as we show, when averaging the two, SunSpot achieves a location near the ground truth.

3.2 Localizing a Home

The latitude and longitude define only a region of interest, and are not accurate enough, even with 1Hz resolution energy data, to identify a specific address. SunSpot uses another method to localize a home within the region, as described below. We define a region of interest as being a radius r around the inferred location.

Identifying Candidate Homes. To identify candidate homes, we observe that solar modules are clearly visible from publicly available satellite imagery. Figure 12 shows a representative photo of a rooftop solar deployment. While it is likely possible to identify candidate homes using image recognition, given the consistent and distinctive appearance of solar modules, SunSpot takes advantage of crowd-sourced image recognition on Mechanical Turk, which provides a programmatic interface to hiring people to perform routine tasks, such as image processing. In this case, SunSpot submits tasks to identify whether a solar module appears within the image.

To reduce costs, we also leverage Google Maps’ landscape API that colors areas with and without man-made structures differently, allowing us to filter out forests, deserts, bodies of water, etc. Since $>97\%$ of land area in the U.S. does not have man-made structures, this optimization significantly reduces the search area, although it becomes less effective the more urban the region. Figure 13 shows an example of Manhattan, where man-made structures are colored black and other areas are colored green (land) or blue (water). The figure shows the API is precise at distinguishing areas without man-made structures, as streets, central park, and shoreline are not black. We provide images to the OpenCV image processing library to filter out images with very little black color.

Filtering Sites. There may be many candidate solar homes identified within the region of interest. There are numerous ways to filter this list of candidate homes. Some examples include: computing the area covered by a solar system to estimate its maximum output, and then filtering out homes that deviate from the anonymous solar data; observing panel properties, such as the orientation or the presence of obstructions, and checking if those properties manifest themselves in the data; or comparing how well drops in energy generation align with clouds moving over each candidate home.

Of course, the filtering above may not be able to prune the list of candidate homes to only a single one. Solar energy data inherently provides k -anonymity, where k represents the number of nearby homes with similar solar deployments [23]. For example, an Hawaiian neighborhood where nearly every home has solar is less vulnerable to precise localization, despite the sunny weather,

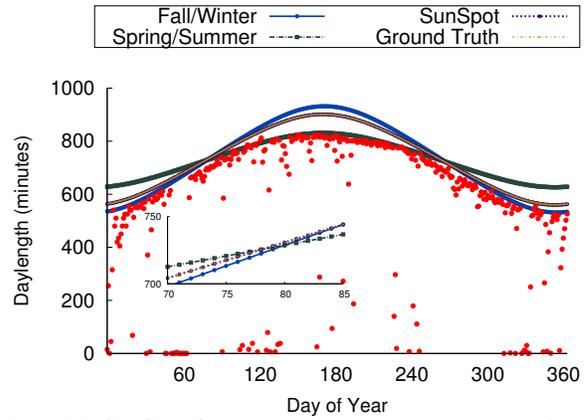


Figure 11: SunSpot finds the daylength curve that provides the tightest upper bound on the data in each half of the year. The zoomed-in inset shows SunSpot curve nearly precisely overlaps the ground truth daylength curve.

compared to a home in the Southeast where few homes have solar.

3.3 Preserving Privacy

There are many possible ways to preserve the privacy of solar energy data. We discuss a few below, but, due to space constraints, only focus on localization in this paper and leave a full treatment of privacy preservation to future work. Simple data transformations that shift all datapoints forward or backwards in time would reduce the accuracy of longitude estimates. Utilities could apply these data transformations (or even remove time labels altogether) before releasing data to third parties. However, there may be legitimate reasons for third parties to know the absolute time of generation. Consumers could apply such transformations themselves by shifting their consumption using batteries. Such shifting would require significantly less battery capacity than is required to prevent Non-Intrusive Load Monitoring [13, 28], since consumers can significantly reduce longitudinal accuracy by shifting perceived generation by only a few minutes. Consumers might also be able to employ background load scheduling to introduce noise at the start, stop, and peak of solar generation. While our approach relies on these three key generation points, more sophisticated approaches may be possible that leverage the entire generation profile for localization. Thus, provably masking latitude poses a greater challenge, since it potentially requires modifying the entire profile.

4. IMPLEMENTATION

We implemented SunSpot in `python` using widely available open-source code that computes a location from its sunrise and sunset time²; SunSpot could also leverage any of a number of online APIs [21]. Our current implementation determines the region of interest as described in Section 3, but does not implement the adjustments to account for different orientations (from Section 3.1). After defining the region of interest, our implementation then processes satellite data to filter areas without man-made structures, divides it into many small images, and submits them to Amazon’s Mechanical Turk to detect candidate homes. Our integration with Mechanical Turk downloads satellite imagery from Google Maps, which has a maximum zoom of 20 that corresponds to a width of $\sim 70\text{m}$ in the northern U.S. (but increases to $\sim 100\text{m}$ near the equator). In some highly rural areas, Google Maps either has much lower resolution or is not available. For these areas, higher resolution satellite imagery, which is available for purchase, may be necessary. The to-

²See <https://github.com/mikereedell/sunrisesunsetlib-java> and <https://github.com/rconradharris/pysunset/>



Figure 12: Rooftop solar is identifiable from satellite imagery.

tal number of images at a zoom-level of 20 (640x640 pixel) within a 1km² radius is 276. We use the Google Maps API to generate the equivalent images with areas with man-made objects black and other areas a different color, and then use OpenCV to automatically remove any images that have more than 5% of their area covered in black pixels. We currently do not apply the additional filters from Section 3.2. Thus, our results are conservative, as applying these techniques would only improve SunSpot’s accuracy.

5. EXPERIMENTAL EVALUATION

We evaluate our results on publicly-available energy data from 14 solar-powered homes at known locations with visible solar modules in the northern hemisphere. We use per-second solar energy data for three of the homes and per-minute resolution data for the remaining 11 homes. For each home, we have between 6 months and a year’s worth of data. Since our initial prototype does not account for irregular module tilts or orientations (as discussed in Section 3.1), we focus on homes with mostly south-facing orientations that maximize solar generation. Our evaluation quantifies the localization accuracy for solar deployments per-second and per-minute data resolution. We then evaluate the cost and accuracy of crowd-sourced image processing on Mechanical Turk.

5.1 Localization Accuracy

Figure 14 shows distance error when localizing the region of interest for the three homes with per-second resolution solar energy data. The latitude error is the north-south accuracy, while the longitude error is the east-west accuracy. We then compute the combined distance error as the hypotenuse of the right triangle formed by the latitude and longitude error. The combined area represents the minimum radius required to include the home in the search area. The figure shows that with per-second energy data the inaccuracy ranges from 10km to 20km. Interestingly, for Homes A and B the error in latitude dominates the total error, while for Home C the error in longitude dominates the total error. We believe Home C’s higher longitude error is largely due to its orientation, which deviates the most from south-facing (and our current implementation does not take into account when determining the longitude). The underlying reason for the difference in latitude error is more difficult to determine, as averaging the spring/summer and fall/winter cancel out some, but not all, of the effects of a solar system’s irregularities. This may be due to different conditions in each half of the year, such as the presence of nearby trees, which might provide shade in the summer but not in the winter.

Similarly, Figure 16 shows results for 11 homes with per-minute resolution data. We sort the homes based on their error in total distance from the ground truth location in Figure 16(a), and again report both the latitude and longitude error in (b) and (c). The red line at 27.9km indicates the baseline precision (at the equator) with minute-level data.³ Overall, the minimum error is 10km with

³Note that the actual precision varies by the cosine of the latitude,



Figure 13: Map of Manhattan using Google Maps API that colors areas that include man-made structures black.

six homes having an error near or below the baseline precision. The average error is 62km (or near 2× the baseline precision), the largest error is 160km (or near 5× the baseline precision). Again, the larger errors are due to less efficient deployments with orientations that deviate more from south-facing. Since we define the region of interest based on a radius r from the home, the region of interest on average is within $2^2 = 4\times$ the smallest possible region given the minute-level resolution of the data. Interestingly, in this case, the average latitude error is less than the average longitude error, despite the fact that our daylength estimates are much more inaccurate than our solar noon estimates (due to the start and stop of generation not aligning with the sunrise and sunset times).

Our results above demonstrate that 1Hz resolution significantly improves accuracy. To illustrate this, Figure 15 shows this scatterplot for five months of data for Home A, along with the inferred EoTs when using minute-level and second-level data over this period. As the graph shows, the time difference between the best fit with minute-level data and the best-fit with second-level data is 22 seconds, which corresponds to an error of ~ 41 km. In contrast, the error between the ground truth longitude and the longitude inferred by the second-level data is only 1.75km (or 4× the baseline precision for second-level data).

SunSpot’s longitude accuracy also depends on tuning the k parameter and the tolerance parameter from Section 3.1. Figure 17 plots the accuracy of inferring longitude as a function of the tolerance parameter for $k=3$ using Homes A, B, and C with 1Hz resolution data. Recall that the tolerance parameter is the amount of time above and below the EoT curve we are fitting, such that we count datapoints within this range as overlapping the EoT curve. The graph shows that as we increase the threshold to near 50 seconds the localization for all three homes becomes more accurate. Interestingly, the best tolerance for second-level data is near one minute, which is similar to the one minute tolerance we also used for the minute-level data. Since our dataset includes a wide range of homes from different locations, these results suggest the magnitude of these parameters do not vary significantly across deployments.

Finally, while our current implementation does not adjust for inefficient orientations that deviate significantly from south-facing, we have performed an initial evaluation of this effect. Home C is unique in that it has two separate solar arrays—one south-facing and one east-facing—with two separate inverters and meters. While the data in Figure 14 is from the south-facing solar array and has a longitude error of 18km, the longitude error for the east-facing array is 50km. By quantifying the effect of orientation on solar generation under the same weather conditions, this initial result indicates the potential of the optimizations that adjust for orientation. We plan to explore this as part of future work.

and these homes are located across a range of latitudes.

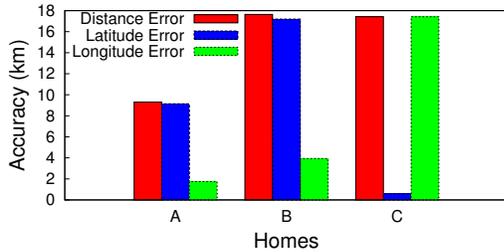


Figure 14: Sunspot accuracy when identifying efficient solar deployments using per-second resolution solar energy data.

5.2 Image Processing

To preserve privacy, we did not conduct any real searches on Amazon’s Mechanical Turk, but rather ran microbenchmarks to evaluate the accuracy and cost of identifying anonymous solar-powered homes. Here, we took a random urban area with 2km radius (or 12.6km²). We chose this small area both to limit costs and to enable manual verification of all the solar-powered homes in the area by observing each image. We divided this area into 3481 satellite images from Google Earth, which has a maximum zoom of ~65-70m at 640x640 resolution. Since we chose an urban location, a much higher percentage (82%) of these images contained man-made structures compared to the U.S. average, yielding a total of 2847 images. We manually checked these images for solar-powered homes and found 28 total.

We then submitted the 2487 images as “categorization” tasks on Mechanical Turk. We selected master-level workers for 5% extra cost to ensure high accuracy.⁴ Amazon allows a maximum redundancy of two workers per task, so we issued a total of $2487 \times 2 = 5694$ images for categorization into two categories: i) yes, solar modules do exist or ii) no, solar modules do not exist. Each task had a reward of \$0.02 and Amazon charges an additional \$0.02 per task. Thus, the overall cost of the experiment was \$170.82, or \$13.60/km². Of the 5594 images, 99% were categorized in <30 minutes with the average time per task equal to 42 seconds. The workers agreed on 26 images, and were thus correct in identifying all but two deployments, yielding a 93% accuracy.

Since we chose a relatively small area (to minimize experiment costs and perform manual verification) in an urban setting, we were only able to filter out 18% of the search area. However, generally the larger the search radius, the higher the percentage of land area that can be filtered out. For example, in this experiment, if we had chosen a 10km radius near the search radius for Home A, only 60% of the images contained man-made structures (38085 out of 62845 total images). For non-urban areas and larger areas, we expect to be able to filter out an even higher percentage of the images. Thus, even for the relatively large search areas, it is possible to filter out a high percentage of the actual land area when searching. These costs may be reduced using computational image processing, rather than people, recognizing solar modules. Given the uniformity in solar module appearance, automated recognition is likely possible.

6. RELATED WORK

There is significant prior work on estimating solar production for a specific location, which solar installers routinely use to give users an estimate of their potential benefits from solar. There is also significant prior work on analyzing building energy consumption data to infer individual appliance energy usage [29], i.e., Non-Intrusive Load Monitoring (NILM) and user behavioral patterns, such as occupancy [5, 12]. NILM researchers have also looked at inferring the generation profile of solar power by treating it as another load

⁴Master-level workers have an accuracy >90%.

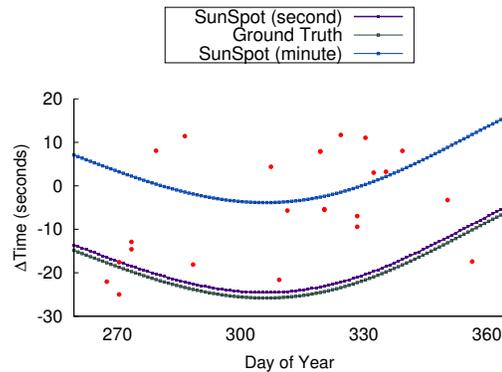


Figure 15: The difference in error in placing the EoT when using minute-level and second-level data.

(with negative consumption) and disaggregating it [14]. Such solar disaggregation is now included in commercial offerings from third party analytics startups, which actively use it on data from a large number of utilities [3, 11]. As part of future work, we plan to extend SunSpot to localize a home based on net meter data by first disaggregating the solar data.

Security researchers have recognized that the energy analytics above represent significant privacy threats [13, 28, 15, 6]. However, this prior work focuses on using chemical or thermal energy storage, e.g., batteries and water heaters, to mask the changes in energy usage that analytic techniques use to infer behavior. The threat is that utilities can associate behavior learned from energy data with account information, e.g., names and addresses. The threat SunSpot exposes is different, as it reveals that data most people believe is anonymous is actually not anonymous.

SunSpot is also related to prior work on modeling a solar deployment’s generation based its various characteristics, e.g., the weather, tilt/orientation, etc. These models are largely used for predicting solar generation in the near-term future based on weather forecasts. Large solar farms may develop detailed models that incorporate specific characteristics of the deployment, e.g., type of panels, tilt/orientation, wiring, etc., and data from co-located irradiance sensors [1]. SunSpot differs from this work in that it does not predict solar output, but instead estimates the location of a solar-powered home based on its output. Prior work also applies machine learning techniques on empirical solar data to develop “black box” models that do not require such deployment-specific details [9]. SunSpot is similar to this work in that it also operates on anonymous solar energy data, although for localization and not prediction. However, SunSpot could potentially improve its accuracy by incorporating information about a solar deployment’s characteristics learned via such models, such as tilt and orientation.

7. CONCLUSION

We design SunSpot to localize anonymous solar energy data and expose its threat to privacy. SunSpot extracts the location information inherently embedded in solar data to localize a solar deployment to a small region of interest. The system then uses crowd-sourced image processing to identify a small set of potential solar deployments within the region. We evaluate SunSpot on publicly-available energy data from 14 homes with rooftop solar, and show that its accuracy ranges from 10km to 20km for 1Hz data. For per-minute resolution, SunSpot identifies a region of interest that is within $4\times$ on average the size of the smallest possible region given the data resolution. SunSpot is then able to narrow this region to a set of candidate sites with visible solar panels using crowd-sourced image processing of publicly-available satellite data on Amazon’s Mechanical Turk at a cost of \$13.60/km². SunSpot’s motivates a

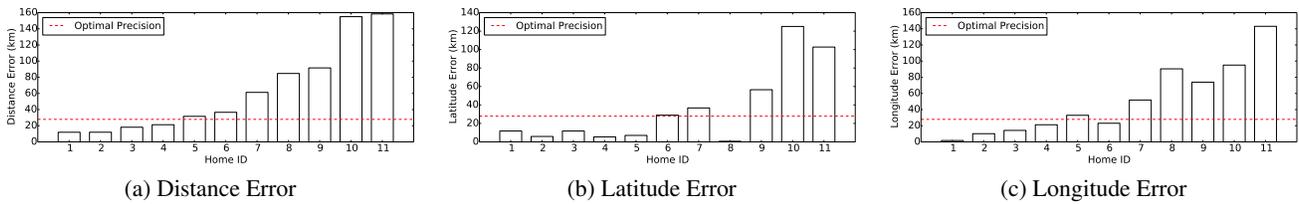


Figure 16: Sunspot accuracy when identifying efficient solar deployments using minute resolution solar energy data. The red line depicts the baseline precision possible (27.9km) with minute-level data.

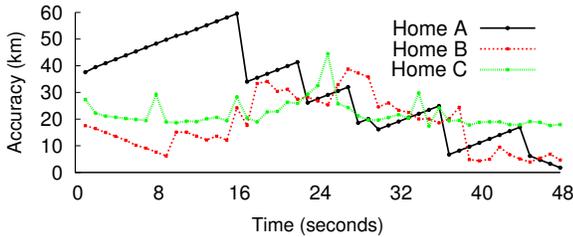


Figure 17: Homes A, B, and C show similar trends when tuning SunSpot's tolerance threshold when computing longitude.

reconsideration of what energy data is classified as “anonymous,” as current regulations, such as the DOE’s Voluntary Code of Conduct for handling energy data, only consider energy data without associated account information to be anonymous. In contrast, our work shows that energy data itself can reveal location.

We plan to implement the optimization that accounts for module orientation to see if we can improve the accuracy of modules with orientations that do not point towards the equator. Even so, there may be deployments that are inherently inaccurate due to conditions outside SunSpot’s control, such as wide variations in grid voltage that might occur in regions with unstable grids. Even if we cannot accurately localize a deployment, it may be possible to estimate localization accuracy. For example, the amount of deviation from the Equation of Time when inferring longitude might indirectly reveal the level of inaccuracy. Thus, we plan to study different methods for assessing the probability that our localization is accurate. We also plan to investigate various approaches for preserving the privacy of solar data, as this paper focuses solely on localization. Finally, we plan to apply existing techniques for solar disaggregation on net metered data to understand the potential to accurately localize “anonymous” net metered data that includes solar generation.

Acknowledgements. This research is supported by NSF grants IIP-1534080, CNS-1405826, CNS-1253063, CNS-1505422, and the Massachusetts Department of Energy Resources.

8. REFERENCES

- [1] S. Achleitner, A. Kamthe, T. Liu, and A. Cerpa. SIPs: Solar Irradiance Prediction System. In *IPSN*, April 2014.
- [2] G. Barbose, S. Weaver, and N. Darghouth. Tracking the Sun VII: An Historical Summary of the Installed Price of Photovoltaics in the United States from 1998 to 2013. Technical report, Lawrence Berkeley National Laboratory, September 2014.
- [3] Bidgely. <http://bidgely.com>, May 2015.
- [4] M. Blanco-Muriel, D. Alarcon-Padilla, T. Lopez-Moratalla, and M. Lara-Coira. Computing the Solar Vector. *Solar Energy*, 70(5):431–441, 2001.
- [5] D. Chen, Sean Barker, A. Subbaswamy, D. Irwin, and P. Shenoy. Non-Intrusive Occupancy Monitoring using Smart Meters. In *BuildSys*, November 2013.
- [6] D. Chen, D. Irwin, P. Shenoy, and J. Albrecht. Combined Heat and Privacy: Preventing Occupancy Detection from Smart Meters. In *PerCom*, March 2014.
- [7] U.S. Energy Information Administration, Frequently Asked Questions, How Many Smart Meters are Installed in the U.S. and who has them? <http://www.eia.gov/tools/faqs/faq.cfm?id=108&t=3>, 2011.
- [8] U.S. Energy Information Administration, Electricity Monthly Update. <http://www.eia.gov/electricity/monthly/update/>, January 2016.
- [9] S. Iyengar, N. Sharma, D. Irwin, P. Shenoy, and K. Ramamkritham. SolarCast - A Cloud-based Black Box Solar Predictor for Smart Homes. In *BuildSys*, 2014.
- [10] J. St. John. 50 Million U.S. Smart Meters and Counting. In *GreenTech Grid*, September 16th 2014.
- [11] J. St. John. Bidgely Thinks Algorithms Can Replace Hardware to Capture the Impact of Rooftop Solar. <http://www.greentechmedia.com/articles/read/bidgely-launches-solar-disaggregation-for-the-home>, July 8th 2014.
- [12] W. Kleiminger, C. Beckel, T. Staake, and S. Santini. Occupancy Detection from Electricity Consumption Data. In *BuildSys*, November 2013.
- [13] S. McLaughlin, P. McDaniel, and W. Aiello. Protecting Consumer Privacy from Electric Load Monitoring. In *CCS*, October 2011.
- [14] R. Mohan, T. Cheng, A. Gupta, and Y. He. Solar Energy Disaggregation using Whole-house Consumption Signals. In *NILM Workshop*, June 2014.
- [15] A. Molina-Markham, P. Shenoy, K. Fu, E. Cecchet, and D. Irwin. Private Memoirs of a Smart Meter. In *BuildSys*, November 2010.
- [16] Pecan Street. <http://www.pecanstreet.org/>.
- [17] D. Perera. Smart Grid Powers Up Privacy Worries. in *Politico*. <http://www.politico.com/story/2015/01/energy-electricity-data-use-113901>, January 1st 2015.
- [18] PlottWatt. <https://plotwatt.com/>, May 2015.
- [19] Massachusetts CEC, Production Tracking System. <http://www.masscec.com/production-tracking-system>, January 2016.
- [20] Sunrise/Sunset Algorithm. http://williams.best.vwh.net/sunrise_sunset_algorithm.htm, January 2016.
- [21] Sunset and Sunrise Times API. <http://sunrise-sunset.org/>, January 2016.
- [22] R. Swanson. A Vision for Crystalline Silicon Photovoltaics. *Progress in Photovoltaics: Research and Applications*, 14(5), August 2006.
- [23] L. Sweeney. k-Anonymity: A Model for Protecting Privacy. *Journal on Uncertainty, Fuzziness, and Knowledge-based Systems*, 10(5), 2002.
- [24] Smart Meter Implementation Programme. https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/42737/1480-design-requirement-annex.pdf.
- [25] H. Umland. *A Short Guide to Celestial Navigation*. 1997.
- [26] Voluntary Code of Conduct (VCC). Technical report, U.S. Department of Energy, January 12 2015.
- [27] B. Wallace and Z. Shahan. Solar and Wind Power Prices Often Lower Than Fossil Fuel Power Prices, in *CleanTechnica*, April 13th 2015.
- [28] W. Yang, N. Li, Y. Qi, W. Qardaji, S. McLaughlin, and P. McDaniel. Minimizing Private Data Disclosures in the Smart Grid. In *CCS*, October 2012.
- [29] M. Zeifman and K. Roth. Nonintrusive Appliance Load Monitoring: Review and Outlook. *IEEE Transactions on Consumer Electronics*, 57(1), February 2011.