## Abstract

Real-time location systems (RTLS) are commonly used for identifying the location of an asset inside of a room. These RTLS applications are called indoor positioning systems. In this paper, we investigate the effect of removing an access point on the accuracy of an indoor positioning system developed by Nolan and Temple Lang (1). We also improve upon their location determination methods by implementing a weighted *k*-nearest neighbors model for position identification. Results show an improvement in location prediction when considering a different access point previously unused in the authors’ analysis. Additionally, we find greater success using a weighted *k*-nearest neighbors approach for determining location.

## Introduction

Tracking an Amazon package online utilizes a real-time location system to communicate critical location information to consumers. With a simple button press on their phone, users can track their packages instantly. Real-time location systems are also a component in inventory management and logistics. Companies such as Dell Computer pioneered inventory management systems utilizing real-time location systems (Ojo-Osagie 2).

Real-time location systems (RTLS) automatically track and identify the location of people or objects wirelessly. A classic RTLS example is location tracking of objects in a room. This RTLS application is called an indoor positioning system. Hardware such as bluetooth tags or cell phones are used as mobile beacons to communicate with strategically placed access points around the room. These beacons can be attached to people, other devices, and even robots. A beacon produces a signal that is read by the access points around the room. Each signal provides data used to determine the location of the mobile device within the room. Software applications utilizing triangulation, trilateration or a combination of location determination algorithms actively translate this location data and produce usable interfaces for people to identify the location of a beacon.

Applications for real-time location systems are vast and extend beyond a simple enclosed space. Advances in wireless technologies and proliferation of tracking tags have made real-time location systems ubiquitous in manufacturing, inventory management, and navigation (Winick 3). Identifying the current location of a package is a popular use of RTLS technology. Tracking previous location history and tracing locations to predict future locations are also common tasks. Additionally, active tags can provide critical data on the temperature of a package or the blood sugar level of a patient along with location information.

In this paper, we improve upon previous analysis by Nolan and Temple Lang (1) using indoor wireless signal strength data provided by Mannheim University. These data are generated from a single mobile device at 166 locations on one floor of a multi-story building. Eight orientation angles are considered at each location and 110 readings are taken for each (x,y) location, angle combination. For each reading, seven access points provide signal strength data to the mobile device.

In their analysis, Nolan and Temple Lang implement an indoor positioning system using signal strength data and a mean-based *k*-nearest neighbors model. The model is used to predict the location of a mobile device using previously unseen signal strength data. The authors drop an access point from their training data set and use six of the seven access points for their *k*-nearest neighbors model.

We investigate the accuracy of the authors’ *k*-nearest neighbors model utilizing the access point previously not considered. We also extend their mean-based *k*-nearest neighbors approach by implementing a weighted *k*-nearest neighbors model for predicting the location of previously unseen signal strength data.

## Literature Review

Various reader and tag communication protcols are used depending on the technology employed with real-time location systems. For example, radio frequency identification (RFID) often employs ISO 18000, a universal protocol for different RFID frequencies (ISO 4). On the other hand, infrared line-of-sight systems often use affordable hardware and a protocol stack for short range communication (Barker, et al 5). These and other protocols along with associated hardware allow for location data to be passed through objects and provide for precise indoor location identification.

The techniques used to analyze the position of the tags within each system can also vary. Communication protocols and associated data transmission methods impact analysis tasks for determining the location of a particular target. Though we utilize *k*-nearest neighbors in this paper, other location prediction methods frequently depend on wireless tag communication protocols (Boulos, et al 6).

Bayesian methods have been implemented for location prediction (Madigan, et al 7). Location information is not necessary for training given prior knowledge about the nature of Wi-Fi signals. Thus, query times for prediction were reduced and location prediction proved much faster than traditional *k*-nearest neighbors methods.

To address inaccuracies and rough calibrations associated with common wireless location modeling, weighted least squares techniques have been developed to improve localization results (Tarrío, et al 8). These techniques outperform standard propogation channel modeling for multiple wireless network types. They also result in very limited overhead compared to standard methods associated with channel modeling and produce more accurate location results.

Real-time location systems are applied in inventory management, livestock maintenance, and personnel tracking. Healthcare has continued to develop ways to utilize real-time location systems to limit cross-contamination. At the Toronto University Health Network, a real-time location system allows the university to locate personnel and equipment entering a contaminated area. The real-time location system sends an alert if the person or equipment hasn’t been decontaminated (Swedberg 9). Another example is a patent filed for tracking livestock that would assist ranchers and farmers in obtaining location information on their herds. This system utilizes the distance value transmitted from the reader and *k*-nearest neighbor analysis to determine location (Chung 10). In the defense industry, contractors assist in moving products from one location to another in their industrial processes to increase safety, efficiency and reduce human overhead (Hidalgo 11).

Supporting hardware and communication protocols will continue to advance as businesses continue using real-time location systems to improve fulfillment requests with robotics and in warehouse distribution (Ding 12). An array of location determination algorithms currently exist. However, these algorithms are often dependent upon the scope and infrastructure deployment of the real-time location system. Consequently, there is currently no one-size-fits-all approach when it comes localization or location determination.

## Methods

In this paper, we investigate the impact of Nolan and Temple Lang’s decision to exclude an access point from their indoor positioning system. We also implement a weighted *k*-nearest neighbors model and compare it to the mean-based *k*-nearest neighbors model they developed. Prior to discussing methods for accomplishing our two analysis objectives, we provide salient findings about the raw data and the transformations taken to prepare the data for analysis.

The raw data is split into a training data set and a test data set. We focus on the training data set for analysis purposes. The training data set contains 151,392 unique lines detailing the (x,y) position and orientation of a mobile device. At each position and orientation combination, 110 signal strength readings are obtained between the mobile device and seven access points. In addition to signal strength readings, variables such as the POSIX time of the reading and the physical MAC address for each access point are also captured.

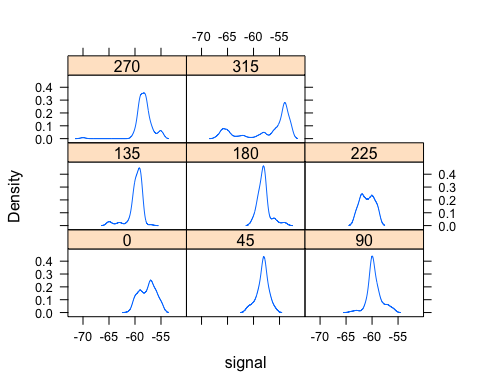
After excluding extraneous access points, normalizing orientation values, and translating time variables, a data set for analysis is created. The format of this data set can be seen in Table 1.

Table 1: Formatted Dataset for Analysis

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| time | posX | posY | orientation | mac | signal | rawTime | angle |
| 2006-02-11 07:31:58 | 0 | 0 | 0 | 00:14:bf:b1:97:8a | -38 | 1.139643e+12 | 0 |
| 2006-02-11 07:31:58 | 0 | 0 | 0 | 00:14:bf:b1:97:90 | -56 | 1.139643e+12 | 0 |
| 2006-02-11 07:31:58 | 0 | 0 | 0 | 00:0f:a3:39:e1:c0 | -53 | 1.139643e+12 | 0 |
| 2006-02-11 07:31:58 | 0 | 0 | 0 | 00:14:bf:b1:97:8d | -65 | 1.139643e+12 | 0 |
| 2006-02-11 07:31:58 | 0 | 0 | 0 | 00:14:bf:b1:97:81 | -65 | 1.139643e+12 | 0 |

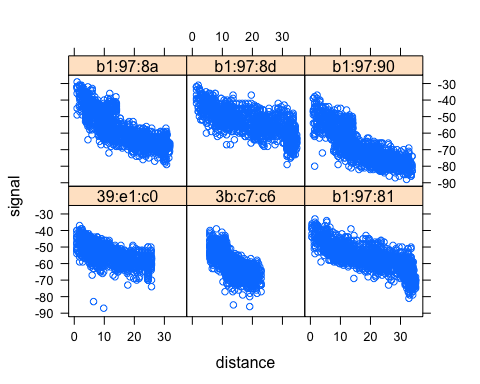
Key exploratory data analysis results help to form assumptions and methods for an indoor positioning system using a *k*-nearest neighbors model. Non-normal, multi-modal signal distributions are found when considering signal strength and the angle of the mobile at each (x,y) location reading. This indicates angle has an effect on signal strength. Figure 2 displays this analysis finding for one access point at a single (x,y) position.

Figure 1: Angle Based Density Plots for 39:e1:c0



The transformed data set is subsequently aggregated by (x,y) position, angle, and MAC address. Additional analysis using the aggregated data set shows higher variability for stronger signal strengths. Also, the relationship between signal and distance shows that signal strength weakens as distance increases. This negative relationship is expected and can be seen for each access point in Figure 2.

Figure 2: Relationship Between Signal and Distance



To prepare the training and test data sets for use in a *k*-nearest neighbors model, we aggregate records based on (x,y) position. Signal strengths from each access point for each position are mean-aggregated, forming a data set where each record is unique based on (x,y) position. Each position is associated with a vector of six signal strength values. A concise sample of this format including three access points can be seen in Table 2.

Table 2: Formatted Dataset for KNN Use

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| posXY | posX | posY | 00:0f:a3:39:e1:c0 | 00:14:bf:3b:c7:c6 | 00:14:bf:b1:97:81 |
| 0-0 | 0 | 0 | -53.47602 | -66.18943 | -64.53751 |
| 0-1 | 0 | 1 | -52.90871 | -66.17718 | -65.91699 |
| 0-10 | 0 | 10 | -55.25828 | -65.03161 | -66.36639 |
| 0-11 | 0 | 11 | -54.16614 | -67.93829 | -68.84551 |
| 0-12 | 0 | 12 | -54.45000 | -68.17850 | -70.83332 |

Given previous analysis results, we assess the impact of Nolan and Temple Lang’s decision to exclude a MAC address on the accuracy of their *k*-nearest neighbors model. We also implement a weighted *k*-nearest neighbors model to predict location inside of the building.

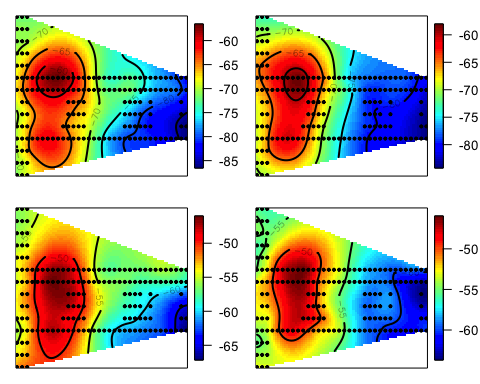
### Analysis of Extraneous MAC Address

Nolan and Temple Lang use six of seven access points presented in the training data set. They exclude an access point with a MAC address ending in 39:dd:cd. This particular access point resides at roughly the same location as another access point (39:e1:c0) on the building floor.

Prior experience with triangulation of position based on sighting lines in land navigation leads to a general finding that additional measurement points will reduce the area of uncertainty for a specific position. Therefore, it could be hypothesized a similar effect can be achieved for an indoor positioning system that uses signal strength of known-position access points to locate an unknown receiver’s physical location.

Visual analysis of the relative signal strength of the two access points reveals a higher degree of certainty for the access point with a MAC address ending in 39:e1:c0. In the visualization below, the top two plots show signal strength for the retained MAC Address access point ending in 39:e1:c0 from the angles 90 degrees and 135 degrees. The bottom plots repeat the exercise, including the access point ending in 39:dd:cd previously removed from the analysis by Nolan and Temple Lang.

Figure 3: Signal Strength for Two Similar Access Points



To test the decision made by Nolan and Temple Lang, we compare the cross-validation results of *k*-nearest neighbor models with three different training data sets. Formally, we compare the root mean square error between predicted location and actual location.

*Exclusion of the Access Point with MAC Address Ending in 39:dd:cd*

Model results are established via cross-validation considering 11 folds and 15 locations sampled randomly for each fold. We establish a random seed to ensure consistent results for each training data set considered.

Repeating Nolan and Temple Lang’s analysis by excluding the access point with a physical address ending in 39:dd:cd yields an optimal root mean square error of 1038.5 using six neighbors. The worst root mean square error of 1515.0 was produced by a *k*-nearest neighbors model considering one neighbor.

*Exclusion of the Access Point with MAC Address Ending in 39:e1:c0*

Next, we build another model excluding the access point with a physical network location ending in 39:e1:c0. Instead, we include the access point 39:dd:cd previously not considered by Nolan and Temple Lang. Cross-validating our *k*-nearest neighbors model utilizing the access point previously not considered results in a minimum error of 935.6 given seven neighbors.

Reviewing the previous signal strength visualization, we hypothesize that the abrupt drop-off in signal strength offers a more discriminatory set of data to model against than an access point with very slowly decaying signal strength relative to distance. We investigate a mean-based *k*-nearest neighbors model utilizing both access points next.

*Inclusion of Both Access Points*

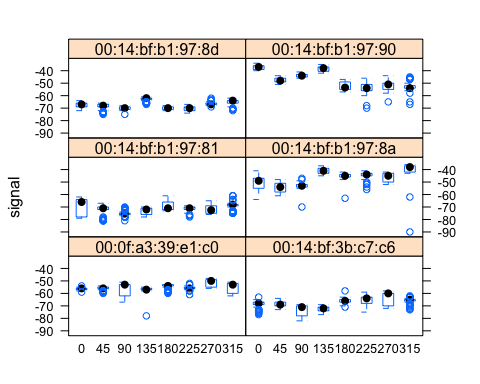
Including both MAC addresses results in a minimum root mean square error of 1100.0 using four neighbors. This result implies a model considering both access points underperforms relative to models including only one of the two access points. We formally explore comparisons in the results section.

### Weighted K-Nearest Neighbors Implementation

Nolan and Temple Lang implement a *k*-nearest neighbors model utilizing the mean of *k* known neighbors in the training data to predict the position of previously unseen records. These records contain signal strengths from six access points on the building floor.

Prior to model fitting, it is determined that the orientation of a reading has an impact on signal strength. For instance, when considering an x,y position of (2,12), MAC address ending in b1:97:90 shows signal strength variability differences based on the orientation considered.

Figure 4: Influence of Angle on Signal Strength for (x,y) Position (2,12)



The authors identify an angle parameter as part of their *k*-nearest neighbors model. This parameter allows the model to account for multiple angles when considering training set records for predicting the location of new observations. While we do not explore the angle parameter in our analysis, we note that three angles are considered when building the training data used for the *k*-nearest neighbors model. Thus, for every new prediction of location, a training set is created using three orientations. To maintain consistency, we also preserve three orientations in the implementation of a weighted *k*-nearest neighbors model. This parameter value reduces bias and allows for multiple angles to influence the mean signal strength used for each observation in the training dataset.

Implementing a weighted *k*-nearest neighbors model requires weighting each training observation associated with the new location we wish to predict. New location in this task means a previously unseen vector of six signal strengths. Training observations closer to the new observation are weighted heavily, while training observations further away have smaller weights. The distance between any new observation and observations in the training data is calculated with Euclidean distance:

*Equation 1: Euclidean Distance*

Signal strengths (*S*) are considered for all six access points when calculating distance. This results in a distance calculation for each training observation given each new observation.

These distances are sorted in ascending order. Weights are calculated based on the choice of *k*, or the number of observations in the training set to consider when predicting the location for the new observation. Thus, the *k* neighbors are training observations taken from the sorted list of distances. In the example of five neighbors, we utilize the five closest training observations based on Euclidean distance to predict the location of a new observation. Each of the five training observations used for prediction are weighted according to Equation 2.

*Equation 2: Nearest Neighbors Weighting*

For each of the *k* training observations, the numerator is represented as one over the distance between the training observation and the new observation. The denominator is the sum of each numerator for the *k* observations we are considering. The first neighbor (k=1) is the closest to the new observation, therefore the weight associated with the first neighbor is always the largest.

Given previous weighting logic, we are able to form a vector of individual weights for each observation in the training dataset used for the *k*-nearest neighbors model. These weights are subsequently multiplied by the x and y locations of each *k* neighbor and summed to predict a weighted (x,y) position for the new observation. To ensure proper predictions, only the weights for each *k* neighbor is considered. All other neighbors assume a weight of zero. Given a new observation, the weight vector can be seen for three neighbors in Table 3.

Table 3: Weight Vectors for k=3

|  |  |  |
| --- | --- | --- |
| posX | posY | wts |
| 1 | 5 | 0.3831149 |
| 1 | 2 | 0.3221486 |
| 0 | 4 | 0.2947365 |
| 2 | 1 | 0.0000000 |

In summary, each new observation we wish to predict is compared to the training data set given the new observation’s orientation. This is the “lazy learning” for which *k*-nearest neighbors is known. A model is never fit to the data, we simply utilize the existing training data for prediction. A unique weight vector is formed based on the number of neighbors *k* and position predictions are obtained by multiplying and summing these weights for each training observation’s (x,y) position.

Nolan and Temple Lang’s mean-based method considers the mean of neighbors for prediction. This method can lead to skewness in predictions as averages are not resistant to outliers. Weighting neighbors, on the other hand, partially corrects for the skewness seen using averages.

*Nearest Neighbor Method Comparisons*

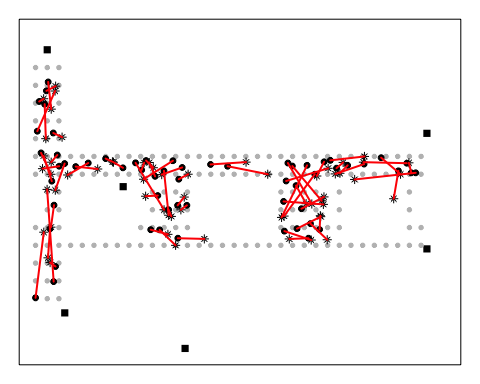
We formally compare our weighted *k*-nearest neighbors method to Nolan and Temple Lang’s mean-based method. We cross-validate using 11 folds, each with 15 unique (x,y) positions. We consider the same training data sets used in the MAC analysis in the previous section.

We first consider a training data set containing the access point with a MAC address ending in 39:e1:c0 and excluding the access point with a MAC address ending in 39:dd:cd. This is the same training data Nolan and Temple Lang used for their model. It was previously seen that this mean-based *k*-nearest neighbors model resulted in an optimal root mean square error of 1038.5 using six neighbors.

When a weighted implementation is considered, the optimal *k*-nearest neighbors model produces a root mean square error of 1015.19 using six neighbors. This error is lower than the mean-based *k*-nearest neighbors model, indicating weighting outperforms the mean-based approach for the original training data.

Using the weighted implementation and six neighbors, errors are visualized as lines running from actual positions in the test data set to predicted positions from the *k*-nearest neighbor model. The asterisks represent predictions, black circles display the location of new observations from the test data set, and gray circles represent the positions of each observation in the training data. The model struggles with predicting locations in the southwest corner of the room. However, other predictions are close to actual observations from the test data set.

Figure 5: Floor Level Errors Using Weighted KNN with 6 Neighbors



We also analyze the dataset including the access point with a MAC address ending in 39:dd:cd and excluding the access point with a MAC address ending in 39:e1:c0. This data set matches the second data set analyzed in the previous section. Model performance was greatest using this data set and considering an mean-based *k*-nearest neighbors model.

Cross-validation of a weighted *k*-nearest neighbors model on the adjusted training data is executed using 11 folds and 15 unique (x,y) positions. The optimal model uses six nearest neighbors and results in a root mean square error of 908.38. This result is better than both the mean-based *k*-nearest neighbors model as well as the weighted model considering the alternate access point.

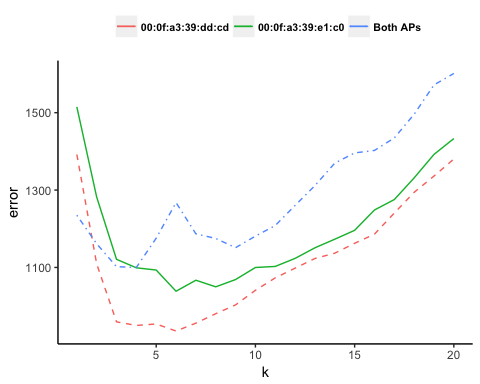
For completeness, both access points are considered in a weighted *k*-nearest neighbors model. After cross-validation, minimum root mean square error of 1051.74 is achieved with four neighbors. Similar to the mean-based model, including both access points in the weighted model results in the highest error.

## Results

### MAC Address Conclusions

Given the learning curves displayed in Figure 4, including the access point with a MAC address ending in 39:dd:cd results in the lowest error. This result implies Nolan and Temple Lang may have discarded the incorrect access point. However, we utilize a random sample with a set seed to develop our cross-validation folds. Given the relative closeness of error in the displayed results and randomness involved in developing the cross-validation folds, the decision to *exclude* 39:dd:cd could easily become the superior option. This was the case when considering other cross-validation folds and can be confirmed by an eager reader.

Figure 6: Learning Curves for Each Scenario Given K Neighbors



Nolan and Temple Lang’s decision to use the MAC address ending in 39:e1:c0 and exclude the access point at 39:dd:cd in their location analysis may have been predicated on knowledge of the hardware of the two devices and some difference in the performance. The signal strength visualization seems to lend credence to this idea. Visualizing a larger area of higher signal strength offered by dd:cd might suggest a lack of sensitivity between measurement of signal strength at different position if the strength of the signal decays more slowly or operates generally at a higher strength.

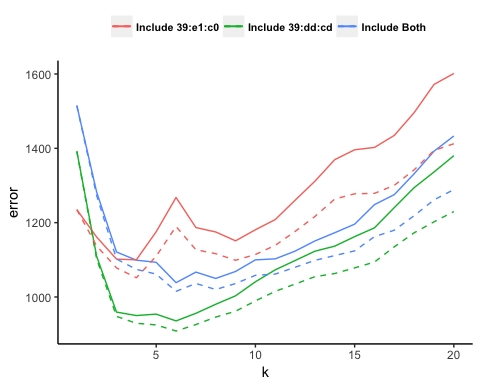
The failure of the inclusion of both MAC Addresses to improve error versus the inclusion of only the address of the access point ending in 39:e1:c0 might suggest that the access points need to be dispersed, and generally lower powered if any gain in accuracy is to be gained from adding access points.

This might offer another line of investigation to deterimine the optimal coverage model for a desired confidence level given a position. While it could be assumed that more access points will generally improve the performance of the location system, the placement of each access point is very important in managing the system. If this access point was placed elsewhere in the environment, the results would likely have been different.

### Weighted *k*-nearest Neighbors Conclusions

To compare the two *k*-nearest neighbors model implementations for each training data set scenario, we provide a learning curve for each scenario in Figure 5.

Figure 7: Learning Curves for Each Scenario Given K Neighbors



The dotted lines represent error results from weighted *k*-nearest neighbor models for each training set scenario analyzed. Solid lines represent mean-based *k*-nearest neighbor models. It is easily noticeable that the weighted implementations outperform for each scenario, especially when considering a larger number of neighbors.

This result makes logical sense, as the weighted method places minimal emphasis on neighbors further away. The average method considers each neighbor equally and is severely affected by skewness. Therefore, one outlier neighbor can influence the predicted position drastically in an mean-based *k*-nearest neighbors model.

In general, we consider the weighted *k*-nearest neighbors method more efficient. Especially when considering observations that are not tightly clustered, as in our example.

## Conclusions and Future Work

We have shown the results of two location determination algorithms using different data sets. For our case, there were multiple factors that changed the performance of our indoor positioning system. Within the limited example of a single floor of a multi-story building, we saw a difference in accuracy through the inclusion or exclusion of an overlapping access point. Further, different location determination algorithms offered different results. Exploration of the visualization of several access points showed obstructions changed the decay of signal strength. When developing an indoor positioning system, it is critical to assess environmental surroundings, hardware and location determination algorithms. These three factors are unique for every indoor positioning system. A general framework should not be applied if optimal performance is expected.

The ever-increasing availability of public Wi-Fi has helped RTLS expand its reach. Wireless devices are built using similar manufacturing standards. However, these devices have a vast array of functionality. Therefore, it is important to understand the location accuracy differences that exist between tracking a user with a tablet, a phone, or a laptop. Further, environmental surroundings greatly impact the ability of locations to be predicted. Building materials or objects in the line of sight of a mobile tag can affect the accuracy of a signal and subsequently lead to mistargeting a location. Real-world environments vary drastically. While our case used a single mobile device, a real-world application may need to provide a location for across many different types of devices.

Real-time location systems can result in privacy concerns given the high level of precision offered with modern location determination technology. When a device is used to track a piece of equipment, one could surmise that the user of the equipment will generally be in possession of the device. This location then serves to track to the user throughout their work day. A recent patent issued to Amazon is for a wrist-wearable device that tracks a warehouse employee’s hands in real-time (Ong 13). Employees assume that the device is used as a measure undertaken to reduce errors in the picking, processing and shipping of customer orders. However, the employee can also view the device as an overlord, creating a running case for justifiable termination based on comparative productivity against other workers. Employees would likely not see an equal access to this data for their own cause. A company implementing RTLS employee trackers would likely protect the data against subpoena by binding arbitration agreements in the case of an on-the-job injury. Encryption may offer an additional measure of security and privacy protection. Yi, et al. propose to utilize the Paillier public-key cryptosystem during the query of the location given a *k*-nearest neighbors model. This provides the end-user with both location and enhances data privacy (Yi, et al 14).

Real-time location services in retail could serve to reduce the intrusiveness of market-research activities conducted in stores. Firms, including VideoMining, have long used in-store video surveillance to monitor shoppers’ activities in store. While most consumers would generally expect that their activities in public are subject to observation, few shoppers might consider the existence of firms that will review footage and analyze the behavior of individual shoppers from the moment they enter the store. This analysis includes classification of the shopper based on their observed and perceived demographics (Hudson 15). Use of a RTLS tag on the shopping cart or basket could bring more anonymity to the process and eliminate the use of the video footage in this activity.

Recent advances in RTLS usage have been seen in the healthcare industry. Patient Room 2020 is a project led by the American Medical Association that uses RTLS to reduce inefficiencies (Bridges 16). The software tools help the administrators cut costs without compromising patient care or safety. The positioning system tracks a caregivers’ position to check on wait times and response times to the patients. RTLS can help identify efficiency gaps and workflow bottlenecks. An administrator can easily identify the closest caregiver to a patient room which clearly has an impact on operational efficiencies. However, given HIPAA regulations, data created by a real-time location system should be heavily scrutinized. It is important to promote benefits of RTLS and provide transparency to how the employer or system owner will use the data, as well as how the employer will safeguard and restrict the use or release of the data.

While real-time data collection is far from new, the public has little understanding of data collection processes. One could argue that the public’s understanding of real-time location services will have a similarly long learning curve. As noted already, many areas of concern for privacy and data security already exist. An ethical framework is required for RTLS data as with all data collection practices. This framework should be based on the foundation of transparency to ensure efficient public understanding of data collection and retention policies.

## 

## References

1. Nolan, D., Temple Lang, D. DATA SCIENCE IN R: A Case Studies Approach to Computational Reasoning and Problem Solving. CRC PRESS, 2017.
2. Ojo-Osagie, O. “Dells Just In Time Inventory Management System.” Academia.edu, www.academia.edu/23256794/Dells\_Just\_In\_Time\_Inventory\_Management\_system.
3. Winick, E. “Amazon’s Investment in Robots Is Eliminating Human Jobs.” MIT Technology Review, 4 Dec. 2017, www.technologyreview.com/the-download/609672/amazons-investment-in-robots-is-eliminating-human-jobs/.
4. ISO Committee. “Arameters for Air Interface Communications at 860 MHz to 960 MHz General.” International Organization for Standardization, 15 Jan. 2018, www.iso.org/standard/59644.html.
5. Barker, P., et al. “Performance modelling of the IrDA infrared wireless communications protocol stack.” International Journal of Communication Systems, 2000. <https://aetos.it.teithe.gr/~vitsas/publications/IntJCommSys_Peter.pdf>
6. Boulos, Maged N Kamel, and Geoff Berry. “Real-Time Locating Systems (RTLS) in Healthcare: a Condensed Primer.” International Journal of Health Geographics, BioMed Central, 28 June 2012, www.ncbi.nlm.nih.gov/pmc/articles/PMC3408320/.
7. Madigan, D. et al. “Location Estimation in Wireless Networks: A Bayesian Approach.” Rutgers University and Avaya Labs, 2006. <http://dimacs.rutgers.edu/Research/MMS/PAPERS/wireless.ps>.
8. Tarrío, P., et al. “Weighted Least Squares Techniques for Improved Received Signal Strength Based Localization.” Sensors, 2011. <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3231493/>
9. Swedberg, C. “Toronto General Hospital Uses RTLS to Reduce Infection Transmission.” RFID Journal, 28 Feb. 2012, www.rfidjournal.com/articles/view?9266.
10. Chung, Won. “RTLS and Livestock Management System.” Google Patents, Google, 10 Sept. 2010, patents.google.com/patent/US20120065483A1/en.
11. Hidalgo, Jeff. “Raytheon Missile Facility Modernized by Advanced Technologies.” Raytheon Company: Technology Today , 2015, www.raytheon.com/news/technology\_today/2015\_i1/facilitymod.html.
12. Ding, Bin, et al. “Application of RTLS in Warehouse Management Based on RFID and Wi-Fi.” IEEE Xplore, 12 Oct. 2008, ieeexplore.ieee.org/abstract/document/4679157/.
13. Ong, Thuy. “Amazon Patents Wristbands That Track Warehouse Employees’ Hands in Real Time.” The Verge, The Verge, 1 Feb. 2018, www.theverge.com/2018/2/1/16958918/amazon-patents-trackable-wristband-warehouse-employees.
14. Yi, Xun; Paulet, Russel; Bertino, Elisa; Varadharajan, Vijay. “Practical *k*-nearest Neighbor Queries with Location Privacy” (2014). Cyver Center Publications. Paper 592. Retrieved from <http://dx.doi.org/10.1109/ICDE.2014.6816688>.
15. Hudson, Steve. “How Retailers Can Extract Value From Location Data.” RFID Journal, 11 Jan. 2016, www.rfidjournal.com/articles/view?13922.
16. Bridges, Holly. “Patient Room 2020 Is on Its Way. More Calming, Connected Care.” Innovatemedtec, 19 May 2015, innovatemedtec.com/content/patient-room-2020-is-on-its-way-more-calming-connected-care.