

BLUETOOTH BASED INDOOR LOCALIZATION USING TRIPLET EMBEDDINGS

Karel Mundnich, Benjamin Girault, and Shrikanth Narayanan

Signal Analysis and Interpretation Lab, University of Southern California, Los Angeles, 90089 CA

ABSTRACT

We propose a novel algorithm for indoor localization using triplet embeddings through Bluetooth connectivity streams obtained in very noisy settings with irregular sampling schemes using environmental sensors distributed ad hoc inside buildings. We pose the problem as a matrix completion problem, where a single row and column is added to the (noisy) distances matrix of sensors. Since this is an underdetermined problem, we use information from connectivity between the sender and the trackers to find the missing distances with the help of triplet comparisons. We test our algorithm in a busy hospital setting, where locations such as patient rooms, intensive care units and nursing stations have been equipped with Bluetooth trackers, that are capable of sending Bluetooth packets as well. We assign the sender role to three different trackers, and estimate their locations. We achieve a mean error of 4.01m in indoor localization for three different senders.

Index Terms— Bluetooth, indoor localization, internet of things, triplet embeddings

1. INTRODUCTION

The internet of things (IoT) has opened multiple sensing possibilities in places and settings where sensing would otherwise be impractical or impossible. Unlike specialized hardware, many of these devices may be easily and more cheaply deployed often in an ad hoc fashion depending on various practical constraints. However, these devices impose challenges due to the hardware characteristics, such as lossy communication channels or noisy measurements. Therefore, new algorithms need to be developed to process the data and obtain meaningful information.

One of the new opportunities that has arisen comes from Bluetooth-enabled devices. It is currently possible to install Bluetooth trackers that obtain information about the vicinity of devices with respect to these sensors, primarily for context-aware proximity based services. These sensors can connect to available WiFi networks to send the data to a remote server to process the information. However it is a well-known problem that obtaining location information from Bluetooth is a challenging problem (cf. Table III of [1]).

We are interested in the problem of localization where a user carries a Bluetooth-enabled device sending packets in a building equipped with n Bluetooth receivers. In this paper, locations of

Bluetooth receivers, and their environment are not constrained. We only require that these receivers record the signal strength from the target devices (i.e., the Received Signal Strength Indicator (RSSI), in dBm) over time.

Various approaches have been proposed to obtain location information from Bluetooth signals. In [2], the authors study a similar scenario, but design the location of receivers such that they can create equal hexagonal Voronoi connectivity regions, similarly to cell-phone towers. This setup has two disadvantages: First, it is not always possible to choose the receiver layout (due to space, power, or connectivity constraints); second, the signal is attenuated by occlusions, so the RSSI values become very noisy. In other words, even if we are able to choose receiver locations with homogeneous Voronoi connectivity regions, the boundaries of these regions would not necessarily correspond to a given signal strength due to these occlusions (a reasonable assumption for cell phone towers). In [3], the authors use two different diffraction models to estimate the distance between a sender and a receiver using RSSI. They present a framework to estimate the parameters of the model, and test their findings in a setting where a sender device is located at different distances of the receivers by using trilateration and a signal coverage density model (SCDM). However, the authors do not inform if the experiments were held in a building in use where diffraction patterns are inherently dynamic due to human activity. In [4], the authors propose an indoor localization and tracking system using Apple's iBeacon protocol with a mix of Kalman filters and Particle Filters. However, testing was performed inside a single room, such that all senders and receivers are in line of sight, i.e., free of any occlusion, static or dynamic. Moreover, sampling rate is constant, which we do not assume to have. This allows simpler models for Bayesian filtering and accurate models for distance based on RSSI, but they are unlikely to hold in a time-varying setting.

In this paper, we propose the use of a model-free algorithm based on triplet comparisons for indoor localization in a realistic setting where the time-varying environment produces diffraction and reflections of the Bluetooth signal. We use RSSI *differences* instead of the measured values to quantify relations between relative distances of a sender to its receivers. In turn, these differences allow to define triplets (i, j, k) of wireless devices each time device j can be assumed closer to i than k . *Triplet embeddings* [5] can then be leveraged to infer the localization of a given device with respect to all other devices. We test our algorithm in a highly dynamic indoor hospital environment.

2. PROBLEM DESCRIPTION

We want to recover the approximate location $\mathbf{x}_s$ of a sender device s in a low-dimensional Euclidean space $\mathbb{R}^d$ (typically $d = 2$ or 3) from its Bluetooth connectivity or RSSI as received by n receivers in a natural indoor environment with human traffic flow. We only have RSSI values for each received Bluetooth packet, obtained in

The research is based upon work supported by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), via IARPA Contract No 2017-17042800005. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the ODNI, IARPA, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon.

this very noisy and time-varying environment.

We assume that we know the location of all n receivers in Euclidean space, as described by the columns of $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_n] \in \mathbb{R}^{d \times n}$. We want to find:

$$\mathbf{Z} = [\mathbf{x}_1, \dots, \mathbf{x}_n, \mathbf{x}_s] = [\mathbf{X}, \mathbf{x}_s], \quad (1)$$

so that the sender s is embedded in the same space as the n receivers.

We model the problem as a Euclidean matrix completion problem, where $\mathbf{D}_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\|_2$ are known for $i, j = 1 \dots, n$ and unknown if $i = s$ or $j = s$. Therefore, we want to complete a Euclidean distance matrix¹ that has a missing row and column:

$$\mathbf{D} = \begin{bmatrix} \mathbf{D}_{11} & \dots & \mathbf{D}_{1n} & ? \\ \vdots & \ddots & \vdots & \vdots \\ \mathbf{D}_{dn} & \dots & \mathbf{D}_{nn} & ? \\ ? & \dots & ? & \mathbf{D}_{ss} \end{bmatrix}. \quad (2)$$

This problem is underdetermined without more information.

2.1. Comparing distances vs. comparing RSSIs

Let $\mathbf{r} \in \mathbb{N}^n$ denote the connectivity vector containing the RSSI values for a given received packet, where $\mathbf{r}_i$ is the RSSI seen by receiver i . Note that $\mathbf{r}_i = 0$ if no packet was received by receiver i .

We use the connectivity information to lift the underdetermined structure of the matrix completion problem. We make the following assumption to use the ordinal relations of RSSI values against distances:

Assumption 1 *Higher RSSIs imply smaller distances between sender and receiver:*

$$\mathbf{r}_j > \mathbf{r}_k \Rightarrow \mathbf{D}_{sj} < \mathbf{D}_{sk}, \quad \forall j \neq k, \quad (3)$$

with some probability $p > p_0$.

This assumption is reasonable because large scale fading models relate RSSI and distances in a decreasing fashion, i.e., RSSI values are a decreasing function of the distance between sender and receiver. For example, the RSSI (in dBm) values can be related in a simple model to the distance:

$$\mathbf{r}_j = -10b \log_{10}(\mathbf{D}_{sj}) + A, \quad (4)$$

for some reference A in dBm that depends on the receiver sensibility, and b a path loss exponent typically equal to 4 for indoor environments [1, 7].

The probability p_0 accounts for small scale fading effects, which introduce noise in the obtained RSSI values.

3. ALGORITHMS

We now introduce triplet embeddings in a noisy setting, and present an algorithm to leverage the missing distances and RSSI information to find $\mathbf{D}$ through noisy triplet comparisons.

¹For a gentle introduction to Euclidean distance matrix properties, please refer to [6].

3.1. Background: Triplet embeddings

Let us consider items $\{1, \dots, m\}$ that we want to represent through points $\mathbf{z}_1, \dots, \mathbf{z}_m \in \mathbb{R}^d$, respectively. Let $\mathcal{T} = \{(i, j, k) : 1 \leq i \neq j \neq k \leq m, j < k\}$ be the set of all unique triplets (i, j, k) . We want to recover the embedding $\mathbf{Z} \in \mathbb{R}^{d \times m}$ from triplet comparisons of the form $\mathbf{D}_{ij} < \mathbf{D}_{ik}$, where $(i, j, k) \in \mathcal{T}$. Let $\mathcal{S}$ be the set of noisy *observed* triplets:

$$\mathcal{S} = \{(i, j, k) \in \mathcal{T}\} \subset \mathcal{T}. \quad (5)$$

Following the model proposed in [5], let us consider that we observe an independent random variable y_t , for each $t = (i, j, k) \in \mathcal{S}$:

$$y_t = \begin{cases} -1 & \text{w.p. } f(\mathbf{D}_{ij} - \mathbf{D}_{ik}) \\ +1 & \text{w.p. } 1 - f(\mathbf{D}_{ij} - \mathbf{D}_{ik}). \end{cases} \quad (6)$$

where $f : \mathbb{R} \rightarrow [0, 1]$ is a link function (for example, the logistic function). Therefore, the probability of correctly labeling a triplet comparison is a function of the distances between items. In particular, f allows to model the fact that smaller differences are harder to compare, with noise actually increasing the probability of a wrongly labelled triplet when distances are similar. We use $\mathcal{S}$, $\{y_t : t \in \mathcal{S}\}$, and a function f to estimate an embedding $\mathbf{Z} \in \mathbb{R}^{d \times m}$. In [5], the authors show that it is possible to recover the Euclidean distance matrix $\mathbf{D}$ from $\mathcal{S}$ under a maximum likelihood framework, when triplets in $\mathcal{S}$ are independently and uniformly drawn from $\mathcal{T}$.

We can pose the optimization problem that computes the estimate $\hat{\mathbf{G}}$ of the Gram matrix [6] $\mathbf{G} = \mathbf{Z}^\top \mathbf{Z}$ by minimizing the empirical risk [5]:

$$\hat{R}_{\mathcal{S}}(\mathbf{G}) = \frac{1}{|\mathcal{S}|} \sum_{t \in \mathcal{S}} \ell(y_t, \langle \mathbf{L}_t, \mathbf{G} \rangle), \quad \text{with } \mathbf{L}_t = \begin{matrix} & i & j & k \\ \begin{matrix} i \\ j \\ k \end{matrix} & \begin{bmatrix} 0 & -1 & 1 \\ -1 & 1 & 0 \\ 1 & 0 & -1 \end{bmatrix} \end{matrix}, \quad (7)$$

and zeros everywhere else in $\mathbf{L}_t$, such that $\langle \mathbf{L}_t, \mathbf{G} \rangle = \|\mathbf{z}_i - \mathbf{z}_j\|_2^2 - \|\mathbf{z}_i - \mathbf{z}_k\|_2^2$. The realizations of the random variables y_t control the direction of our decision:

$$\|\mathbf{z}_i - \mathbf{z}_j\|_2 \stackrel{?}{\leq} \|\mathbf{z}_i - \mathbf{z}_k\|_2 \quad (8)$$

for each $t = (i, j, k)$ with some error probability defined in (6). In this setting, the loss function ℓ is induced by the link function f in the noise model. We can recover $\hat{\mathbf{Z}}$ from $\hat{\mathbf{G}}$ (the solution to (7)) using an eigenvalue decomposition.

The embedding that we find is unique up to translations, rotations and scaling factors since the distances $\|\hat{\mathbf{z}}_i - \hat{\mathbf{z}}_j\|_2$ and $\|\hat{\mathbf{z}}_i - \hat{\mathbf{z}}_k\|_2$ are invariant to the first two transformations and scale invariance is dealt during the minimization of $\hat{R}_{\mathcal{S}}(\mathbf{G})$ [5]. This is not a problem in our case, since we know the exact locations of the receivers (see Sec. 3.3).

3.2. Generating the set of observed triplets

We now present one of the main contributions of the paper. To estimate the matrix $\mathbf{Z}$ in (1), we use both $\mathbf{D}$ and $\mathbf{r}$ to build a set of observed triplets $\mathcal{S}$. All the triplets related to $\mathbf{D}$ are straightforward to build, since we have all the noiseless distances $\mathbf{D}_{ij}$ and $\mathbf{D}_{ik}$ for n receivers. For the remaining triplets involving the sender, our algorithm leverages the construction of the triplets between the n receivers and the sender s through the information in $\mathbf{r}$, and using Assumption 1. Therefore, we assume that the RSSI values in $\mathbf{r}$ induce a triplet comparison between items (s, j, k) , for all $j, k = 1 \dots, n$, and $j \neq k$. We describe the procedure in Algorithm 1.

Algorithm 1: Generate set of observed triplets $\mathcal{S}$ from distances and connectivity

Input: Distances $D \in \mathbb{R}^{n \times n}$; Sensor distances
Input: Connectivity $r \in \mathbb{R}^n$; RSSIs
Input: Non-decreasing function $g : \mathbb{R}_+ \rightarrow \mathbb{R}$
Result: Set of observed triplets $\mathcal{S}, \mathbf{y}$

```

1  $\mathcal{S} = \{\}$ ;
2 for  $k \leftarrow 1$  to  $n, j \leftarrow 1$  to  $k-1$  do
    //  $i$  is a receiver: use the distance
3     for  $i \leftarrow 1$  to  $n$  and  $i \neq j, i \neq k$  do
4         if  $D[i, j] < D[i, k]$  then
5              $\mathcal{S} = \mathcal{S} \cup \{(i, j, k)\}$ ;
6              $y_{(i, j, k)} = +1$ 
7         else if  $D[i, j] > D[i, k]$  then
8              $\mathcal{S} = \mathcal{S} \cup \{(i, j, k)\}$ ;
9              $y_{(i, j, k)} = -1$ 
    //  $s$  is the sender: use the RSSI
10    if  $g(r_j) > g(r_k)$  then
11         $\mathcal{S} = \mathcal{S} \cup \{(s, j, k)\}$ ;
12         $y_{(s, j, k)} = +1$ 
13    else if  $g(r_j) < g(r_k)$  then
14         $\mathcal{S} = \mathcal{S} \cup \{(s, j, k)\}$ ;
15         $y_{(s, j, k)} = -1$ 
16 return  $\mathcal{S}, \mathbf{y}$ ;
```

Algorithm 1 takes as input a non-decreasing function $g : \mathbb{R}_+ \rightarrow \mathbb{R}$, to include prior knowledge into the comparisons involving RSSI values. For example, Bluetooth receivers usually have segments of the RSSI values in which they are very trustworthy (when sender and receiver are very close). Using g we can also remove receivers that received a packet far away with low RSSI, as this is a very noisy phenomenon. An example of a function g is shown in Figure 1.

3.3. Finding the location

To find the location $\mathbf{x}_s$ of the sender device, we minimize (7) to obtain $\hat{\mathbf{Z}}$ [5, 8]. Recall that $\hat{\mathbf{Z}} = [\hat{\mathbf{X}}, \hat{\mathbf{x}}_s]$, as in (1). We use Procrustes [9] to find the best *affine* transformation Π mapping $\hat{\mathbf{X}}$ to $\mathbf{X}$ (in the mean-squared error sense) [6, 9], and apply Π to $\hat{\mathbf{x}}_s$ to obtain the final position. The steps are shown in Algorithm 2. If $\Pi(\hat{\mathbf{x}}_s)$ does not lie in the convex hull of the set of receivers, we assume that we cannot localize the sender with the current information.

4. EXPERIMENTS

4.1. Context

As part of the "TILES: Tracking Individual Performance with Sensors" study to examine the physiological, environmental, and behavioral variables affecting job performance and employee wellness, we

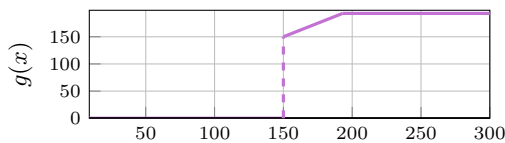


Fig. 1. Example function $g_2(x)$ for Algorithm 1.

Algorithm 2: Estimate sender position from sensor positions and RSSIs

Input: $\mathbf{X} \in \mathbb{R}^{d \times n}$: Known receiver positions
Input: $\mathbf{r} \in \mathbb{N}^n$: RSSIs
Result: $\Pi(\hat{\mathbf{x}}_s)$: Estimated sender position

```

1  $D = \text{distances}(\mathbf{X})$ ;
    // Obtain triplets according to Algorithm 1
    // and compute the embedding
2  $\mathcal{S}, \mathbf{y} = \text{triplets}(D, \mathbf{r}, g)$ ;
3  $\hat{\mathbf{Z}} = \text{embedding}(\mathcal{S}, \mathbf{y}, f)$ ;
4  $\hat{\mathbf{Z}} = [\hat{\mathbf{X}}, \hat{\mathbf{x}}_s]$ 
    // Find the best affine transformation for
    // the receiver positions.
5  $\Pi = \text{Procrustes}(\mathbf{X}, \hat{\mathbf{X}})$ ;
    // Apply affine transformation and return
6 if  $\Pi(\hat{\mathbf{x}}_s) \in \text{conv}(\mathbf{X})$  then
7     return  $\Pi(\hat{\mathbf{x}}_s)$ ;
8 else
9     return null;
```

installed Owl-in-Ones (owls for short, model reelyActive RA-R436 [10]) in a large Los Angeles-based critical care hospital. These Bluetooth transceivers were installed across nursing units covering patient rooms, nursing stations, and various rooms of interest including intensive care units and patient rooms, as shown in Figure 2. The owls are capable of sending Bluetooth packets [11], which we use for the validation of the algorithm. The owls receive signals and send them to a server for preprocessing using UDP. This server aggregates the packets in time and uses preprocessing heuristics such that the RSSI between adjacent events are averaged [11]. These events are accessible through an API, to which we have access. Each event contains the receiver identifiers, corresponding RSSIs, and a timestamp.

4.2. Validation of the proposed approach

We set up our experiments such that an owl takes the role of a sender at a time. This is, we assume that we have a network of $n-1$ receivers, and the n^{th} receiver takes the role of a sender device. We use the known positions $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_{n-1}]$ and the event including the vector $\mathbf{r}$ with RSSI information to generate the *observed* triplets $\mathcal{S}$, according to Algorithm 1. We assume that the noise is defined by

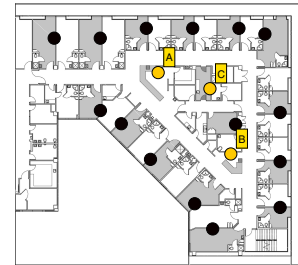


Fig. 2. Owl installation for a nursing unit. The black circles show owl positions, placed in rooms shaded in light gray. The yellow circles show owls that we use as sender devices (one at a time) to validate our algorithm.

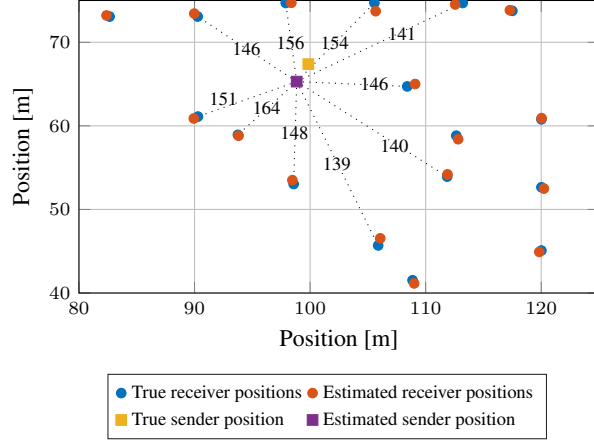


Fig. 3. Positioning example using $g_1(x)$. Error is 2.320m from true location. Dotted lines show connectivity, numbers show RSSI values. Sender is A from Figure 2.

the logistic function:

$$f(D_{ij} - D_{ik}) = \frac{1}{1 + \exp(D_{ij} - D_{ik})}, \quad (9)$$

which induces the logistic loss ℓ in the empirical risk (7) [5]. This loss function is equivalent to the cost used in stochastic triplet embeddings [8], with $\sigma^2 = 1/2$.

We preprocess the data such that we eliminate any entries of the events that have RSSI values greater than 193 and smaller than 136, which is the owl sensibility range. It is possible to receive in the server these RSSI values due to channel noise. We also use two clipping functions for g in Algorithm 1:

$$g_i(x) = \begin{cases} 0, & \text{if } x < \text{RSSI}_i^{\min}, \\ x, & \text{if } \text{RSSI}_i^{\min} \leq x < \text{RSSI}_i^{\max}, \\ \text{RSSI}_i^{\max}, & \text{if } x \geq \text{RSSI}_i^{\max}, \end{cases} \quad (10)$$

with $\text{RSSI}_1^{\min} = 136$, $\text{RSSI}_2^{\min} = 150$, and $\text{RSSI}_i^{\max} = 193$, for $i = 1, 2$. We use these clipping functions to filter out received packets with low RSSIs, that are typically very noisy. Note that given our preprocessing, $g_1(x)$ is equal to the identity in the sensibility range, while $g_2(x)$ clips the values below and RSSI of 150.

We evaluate the error for each event using:

$$\text{Error} = \|\mathbf{x}_s - \Pi(\hat{\mathbf{x}}_s)\|_2, \quad (11)$$

for each received packet. We iterate over three receivers (A, B, and C from Figure 2), so that we have different locations from which to test our approach. We discard any events for which there is only one receiver with RSSI greater than 0, and the estimated positions that are outside of the convex hull generated by the receivers.

4.3. Validation results

Table 1 shows the average error for three different sender locations highlighted on Fig. 2. The average is over the number of packets for which localization was possible. The average error is not always smaller for the clipping function $g_2(x)$ in comparison to $g_1(x)$, but the variance of the error diminishes. We can also see this in for owl A in Figure 4. From this figure, we conclude that the information below RSSI is very noisy, but it is still usable in our setting.

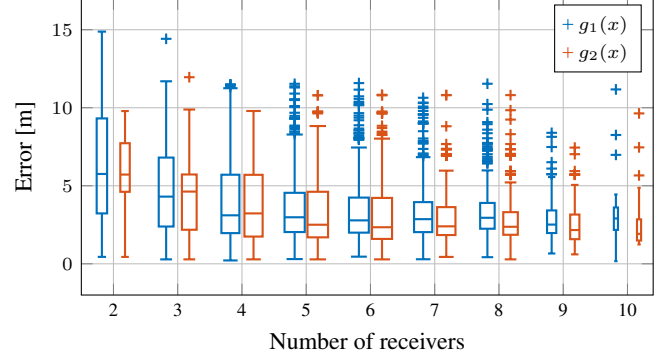


Fig. 4. Box plot of the error as a function of number of receivers for a sent packet for two clipping function g_1 and g_2 . The sender is device A from Figures 2 and 3.

Table 1. Localization results for three different owls used as senders. Mean over approximately 3200 packets sent for each device.

Owl	Error [m] for $g_1(x)$	Error [m] for $g_2(x)$
A	3.767 ± 2.446	3.358 ± 2.118
B	5.456 ± 2.595	5.779 ± 2.306
C	3.244 ± 2.114	4.149 ± 1.117

Figure 3 shows an example of localization from connectivity and the RSSI information. We observe in this figure that the connectivity with a receiver is sometimes occluded by the environment, even though the receivers might be in range. We also observe certain directions of preference for the strength of the signal, which is expected in a real-world setting with constant and time-varying occlusions.

Our proposed algorithm has a good average performance. However, there are many outliers that increase the error. These outliers are mostly caused by receivers that should have received a package (albeit with a low RSSI), but did not. This forces the algorithm to push the sender away from all receivers, in order to minimize the loss related the triplets generate by the lack of connectivity.

5. CONCLUSIONS

This paper presents a novel indoor localization algorithm using triplet embeddings. We use RSSI information in the triplet comparisons as a proxy to compare distances between sender and receivers, and use this information in an optimization framework to find an embedding in 2-dimensional space.

We provide real-world testing of the algorithm in a large dynamic hospital setting, where disconnections, occlusions and fading are time-varying. In this real scenario, we estimate three known positions for senders, and achieve mean squared errors of 3.36m, 5.78m, and 3.24m, respectively for three sender devices with fixed positions. These results are valid in a real world indoor hospital setting based on Bluetooth connectivity.

For future work, we will compare our algorithms with the state of the art, an explore using ideas from robust estimation. We are also interested in integrating time dependency in our algorithm, which we believe can further help estimate locations in noisy conditions.

The code and data is available at <https://github.com/kmundnic/ICASSP2019/>.

6. REFERENCES

- [1] Faheem Zafari, Athanasios Gkelias, and Kin Leung, "A survey of indoor localization systems and technologies," *arXiv preprint arXiv:1709.01015*, 2017.
- [2] Francisco J Gonzalez-Castano and Jaime García-Reinoso, "Bluetooth location networks," in *Global Telecommunications Conference, 2002. GLOBECOM'02. IEEE*. IEEE, 2002, vol. 1, pp. 233–237.
- [3] Javier JM Diaz, Rodrigo de A Maues, Rodrigo B Soares, Eduardo F Nakamura, and Carlos MS Figueiredo, "Bluepass: An indoor bluetooth-based localization system for mobile applications," in *Computers and Communications (ISCC), 2010 IEEE Symposium on*. IEEE, 2010, pp. 778–783.
- [4] Faheem Zafari, Ioannis Papapanagiotou, Michael Devetsikiotis, and Thomas Hacker, "An ibeacon based proximity and indoor localization system," *arXiv preprint arXiv:1703.07876*, 2017.
- [5] Lalit Jain, Kevin G Jamieson, and Rob Nowak, "Finite sample prediction and recovery bounds for ordinal embedding," in *Advances In Neural Information Processing Systems*, 2016, pp. 2711–2719.
- [6] Ivan Dokmanic, Reza Parhizkar, Juri Ranieri, and Martin Vetterli, "Euclidean distance matrices: essential theory, algorithms, and applications," *IEEE Signal Processing Magazine*, vol. 32, no. 6, pp. 12–30, 2015.
- [7] Praveen Kumar, Lohith Reddy, and Shirshu Varma, "Distance measurement and error estimation scheme for rssi based localization in wireless sensor networks," in *Wireless Communication and Sensor Networks (WCSN), 2009 Fifth IEEE Conference on*. IEEE, 2009, pp. 1–4.
- [8] Laurens Van Der Maaten and Kilian Weinberger, "Stochastic triplet embedding," in *Machine Learning for Signal Processing (MLSP), 2012 IEEE International Workshop on*. IEEE, 2012, pp. 1–6.
- [9] Peter Hans Schoenemann, *A solution of the orthogonal Procrustes problem with applications to orthogonal and oblique rotation*, Ph.D. thesis, University of Illinois at Urbana-Champaign, 1964.
- [10] Jeffrey Dungen, "reelyActive Owl-in-One starter kit," <https://shop.reelyactive.com/products/owl-in-one-ble>, 2018, [Online; accessed 1-October-2018].
- [11] Jeffrey Dungen, "Pareto server packet aggregation," personal communication.