Qualitative Tracking of Objects in a Smart Home

A Passive RFID Approach Based on Decision Trees

Frédéric Bergeron
Université de Sherbrooke
Sherbrooke, Quebec, Canada
frederic.bergeron2
@usherbrooke.ca

Kevin Bouchard
Université de Sherbrooke
Sherbrooke, Quebec, Canada
kevin.bouchard
@usherbrooke.ca

Sylvain Giroux
Université de Sherbrooke
Sherbrooke, Quebec, Canada
sylvain.giroux
@usherbrooke.ca

Sébastien Gaboury
Université du Québec à
Chicoutimi
Chicoutimi, Quebec, Canada
sebastien.gaboury
@uqac.ca

Bruno Bouchard
Université du Québec à
Chicoutimi
Chicoutimi, Quebec, Canada
bruno.bouchard
@uqac.ca

ABSTRACT

This paper presents a novel Indoor Tracking System (IT-S) based on passive radio-frequency identification (RFID) technology. The new ITS exploits decision trees built from one dataset per room of a smart home. The datasets are built using a bottle equipped with four class 3 RFID tags and by dividing each room into qualitative zones. The paper discusses how to exploit positioning from decision trees to implement real-time tracking. The long term goal of this ITS is to extract qualitative spatial information to improve recognition of daily living activities' granularity. The results obtained are very encouraging as the average accuracy of the trajectories recognized is over 75%.

CCS Concepts

•Applied computing → Life and medical sciences; Health informatics;

Keywords

Smart home, Tracking, RFID, Decision Trees

1. INTRODUCTION

Technology for assistance is a vast and developing topic of research capturing the attention of researchers in computer science, engineering and in other fields [8]. The increasing attention is mainly due the well-known problem of world population ageing and its many consequences [11]. It is predicted that health care costs will continue to grow at a faster rate than the economy for the next few decades [15]. Assistive technologies could help to limit the impact of ageing by decreasing the resources needed for treatment and support. Toward that goal, many research laboratories around the world are working on assistive smart home [13] as a way to increase the autonomy of ageing people. Smart home is a generic term that can be used to designate a home enhanced with simple technologies to increase the comfort of the resident or reduce energy footprints [9]. In the case of assistive smart homes, the technologies exploited generally cover a wide range of sensors, actuators and effectors.

PETRA '16 Corfou Island, Greece

ISBN 978-1-4503-4337-4/16/06. . . $15.00

DOI: http://dx.doi.org/10.1145/2910674.2910700

Despite the progress that was achieved by the community over the last two decades, there are still many challenges to be overcome in order to successfully deliver assistive smart homes to the public. One of these challenges is the fundamental recognition (and prediction) of the inhabitant's ongoing Activity of Daily Living (ADL) [4]. There are plenty of algorithms and methods designed for that purpose. The main limitation of the literature is the low granularity of the recognizable ADLs. To palliate this issue, more information is required on the state of the environment in which the ADL is realized. Many avenues are explored. One is to define precise models of ADL in a large logical library [5], however, it requires extensive human labour and such a library does not evolve easily. It can also be addressed by exploiting video cameras that enable to capture much more information. Also, vision sensors and wearable [6] are often considered more invasive than ambient technologies [7].

Many researchers are turning toward the exploitation of passive radio-frequency identification (RFID) technology to acquire spatial information on the environment. To do so, daily life objects in the smart home can be equipped with tags in order to be located by antennas distributed into the environment. The problem of developing passive RFID Indoor Tracking System (ITS) is also well documented in the literature [16]. Existing methods are generally designed for robots tracking or industrial contexts which are not adapted to the tracking of objects inside a busy smart home [14]. Additionally, most of them depend on mathematical models that must be precisely configured and tweaked [3]. In our context, it is more important to have an accurate and simple to implement system with less precision. Indeed, Cartesian positions is not more informative than knowing it is current-

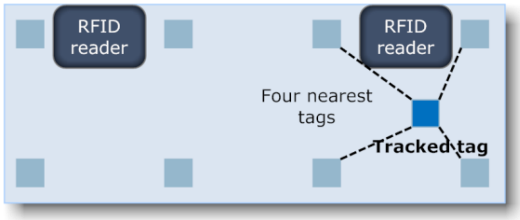

Figure 1: Reference tag method

ly on the kitchen counter for example.

In this paper, we present a new Indoor Tracking System (ITS) capable of tracking daily life objects in real time in a smart home from passive RFID tags. The system only aims to track objects. The ITS must be simple to implement yet scalable to new smart home infrastructure. This system is created with Decision Trees (DTs). There are many advantages to using DTs for tracking. First, the training process is very simple to reproduce. It is not necessary to be an expert to achieve it. Second, the method can be adapted to new RFID infrastructure easily by simply repeating the learning phase. Third, the tracking corresponds mostly to travelling a tree which is very fast.

For the sake of repeatability and to contribute further to the research community, all datasets collected for the experiments presented in this paper are available online on Dr. Bouchard website: www.Kevin-Bouchard.com.

2. RELATED WORK

In this section, we present the main families of approaches to passive RFID positioning and tracking. Most algorithms that can be found in the literature are based on the reference tags principle first exploited by the LANDMARC system [12]. The basic idea, shown on Figure 1, is to exploit the Received Signal Strength Indication (RSSI) of nearby tags fixed at known positions to adjust the RSSI of the tracked tags. The method can be improved with various statistical filters [1]. LANDMARC based systems work very well in general, but they need to be used on a two-dimensional plane. They are therefore not adapted to a smart home, which encompasses furniture and where most of the tracking is not performed at floor level. They also mainly use active RFID tags. These tags allow stronger RSSI and a longer range than passive tags [12] and thus help the system to achieve a good performance.

The second family of algorithms is based on trilateration [3] and triangulation principles [10]. Trilateration uses RSSI and converts it to distance from antennas to draw imaginary circles. The position is the intersection point between three circles. Triangulation cannot be performed with all RFID systems as it requires the capability to calculate the angle of arrival and not all RFID readers collect them. It is with the angles from three antennas that an intersection point can be found.

Finally, the last family exploits data mining and other learning algorithms. However, work on this family is scarce and mainly explores other wireless technologies. For example, Yim et al. [17] exploited wireless local area network access points to build a decision tree during the off-line phase in order to determine the user's location. They have shown that their technique is simpler to implement and perform better than the classical fingerprinting methods. Our hypothesis is that it should also perform well with RFID technology.

3. METHODOLOGY

As mentioned in the introduction, our goal is to create an Indoor Tracking System (ITS) for objects. Such a system needs to know the position of a tracked object at all times. There are many positioning systems that we could have used, but we wanted our system to be simple to implement yet scalable. Therefore, we chose a system based on classical data mining algorithms. The rest of this section gives details about our positioning system.

3.1 Indoor positioning system

To begin with our methodology, we first need to describe the most important part of an ITS: the positioning system. In a previous experiment [2], we built a qualitative indoor positioning system (IPS) in Java using random forests as they resulted in the best overall accuracy for fast training, an important criterion in our context. Therefore, for the ITS, we use a random forest of 250 random trees built with the data from the first experiment. This IPS has an accuracy ranging from 95% to 99% depending on the room of our smart home. The accuracy is computed using 10-fold cross-validation on more than 47 thousands readings. The qualitative tracking relies upon named zones of a certain dimension. These dimensions range from 40cm x 40cm in the kitchen to 75cm x 75cm in the hall, the living room and the dining room. Zones in the bedroom and the bathroom have dimensions of 60cm x 60cm. The counter of the bathroom has a higher precision with dimensions of 30cm x 30cm. Those sizes were chosen to make sure that most ADLs would not occur within only one zone.

When we built the IPS, the antennas were set to emit at intervals of 750ms and we recorded fifty readings per zone. The IPS uses twenty antennas disposed strategically to cover all of the apartment used in this study. Figure 2 shows a map of the DOMUS apartment. Antennas are indicated by crosses. The map also shows all the qualitative zones used by the IPS and the tracking algorithm. The movement of the tracked object is a directed sequence of zones. There are six rooms in the apartment. Each room is associated with a different random forest trained with a different dataset. The IPS uses six different datasets containing fifty classified readings from each zone. We consider that, at all times, we know what room the object is in. We can get this information from other sensors in the apartment, like motion detectors. Therefore, there is no need to combine the random forests.

3.2 Tracked object

Once we have an accurate positioning system, we need an object to track. The same object used to create the IPS was used again: a reusable plastic bottle. Four class 3 passive RFID tags were installed on it, each one facing a different direction. This way, we ensure that there is always a tag almost directly facing an antenna, thus increasing the stability of the Received Signal Strength Indication (RSSI). Then, the four resulting readings are programmatically merged into a single one containing the maximal non zero value for each antenna. Therefore, a reading from the bottle consists of a vector of twenty integers representing the highest value

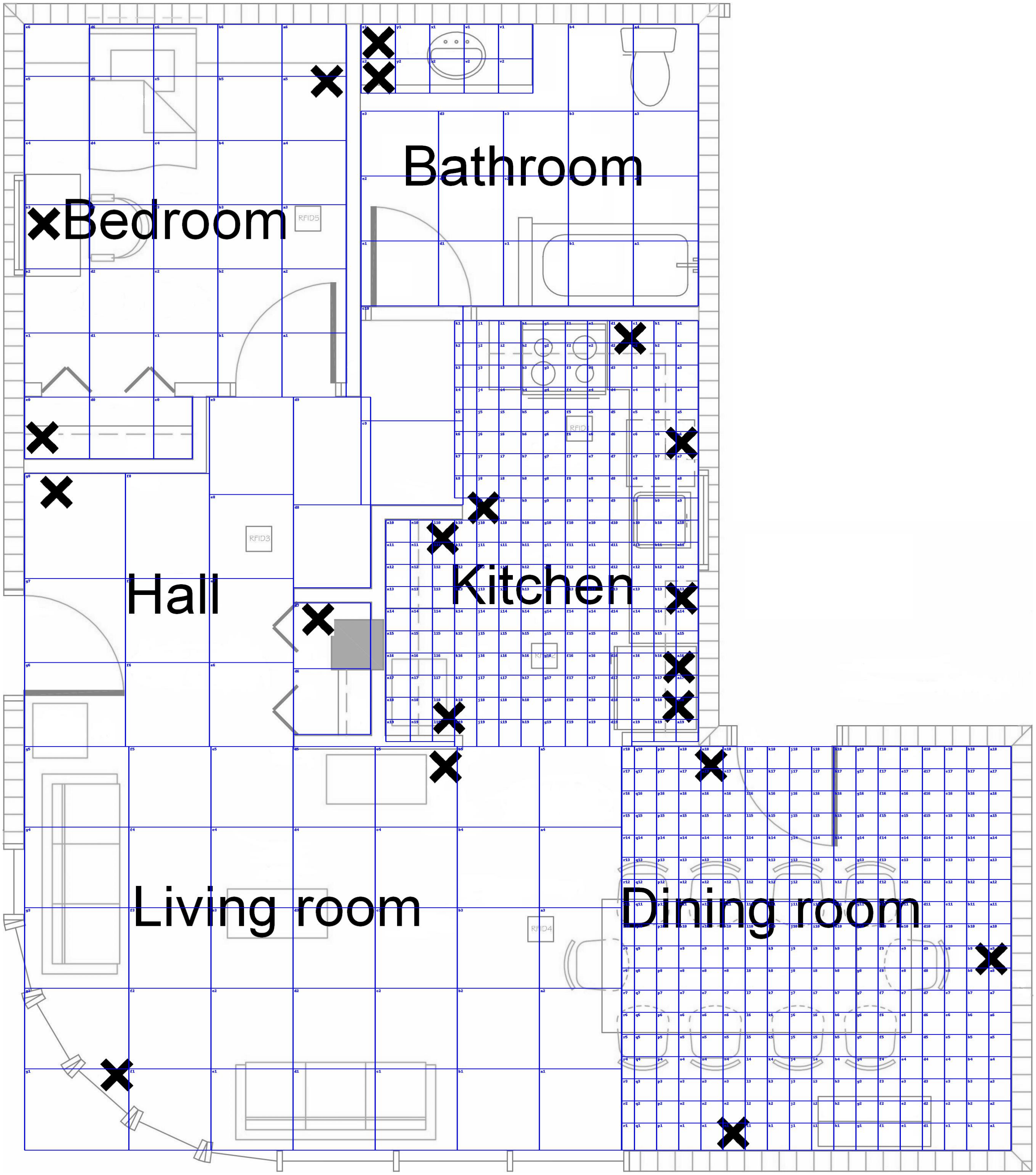

Figure 2: Map of the DOMUS home. The RFID antennas are marked by a X. The grid represents the qualitative zones

Figure 3: Experimental setting in the smart home

collected by tags during a certain interval of time. When building the IPS, this certain interval was 750ms. However, this is way too slow for a real time tracking system. The interval was reduced to the minimal value achievable with our passive RFID system of 20ms. Given the shape of our zones, this allowed us to get many readings for each zone when moving at normal speed.

3.3 Collecting a dataset

In the previous subsections, we explained how the IPS works. The next and final step of the physical part of our experiment was then to gather data associated with paths we would like to track. To do this, we created four plausible paths for each room (i.e.: trajectories that could be part of a normal activity of daily living). We tried to cover most of the daily living surface. Paths also have a varying length, matching the activity they tend to mimic. While some paths may overlap others, they are all distinct and no path is a subpath of a longer one.

Figure 3 shows some paths in the room while members of the laboratory are taking measurements with the special object. As you can see, paths are indicated by a black tape. When taking measurements, the member places the bottle over the tape and moves at slow speed following the tape, making sure the bottle stays over it. We repeated this process ten times for each path. In total, this first practical experiment gave us 240 paths to evaluate to accuracy of the ITS. We took special care to hold the bottle at the same height as the antennas because the random forests were built using data collected at this height. To distribute the impact of human interference with the RFID signal, we hold the bottle in a different way at each repetition. Sometimes we would be behind the bottle, while sometimes we would be in front of it or on the side. This way, we do not always interfere with the same antenna. Moreover, we changed the angle of the bottle in our hands to make sure it would not always be the same tag facing the same antenna.

4. EXPERIMENTATION AND RESULTS

An ADL recognition algorithm like the one we planned to design needs a real-time tracking system. In the previous section, we presented how we gathered the data required to build this system. In this section, we first present how we evaluated the accuracy of our tracking system, then we present the filters we used in order to increase the accuracy.

4.1 Accuracy evaluation

There are many ways to evaluate the accuracy of a tracking system. We chose to only compare the practical sequence of crossed zones versus the theoretical sequence, the path we draw on the floor. Here again, there were many possible metrics. The four we selected are as follows.

4.1.1 Targeted Zones Found

This first metric (TZF) is purely statistical. It examines the overall coverage of the path. Let $\mathbf{S}$ be the set containing the theoretical zones and $\mathbf{P}$ the set of the zones found by the tracking system. Then, the number of targeted zones is simply the cardinality of the intersection of those two bags. To compare sequences between them, we divide this number by the cardinality of $\mathbf{S}$.

4.1.2 Sequential Targeted Zones Found

This second metric (STZF) looks like the first one, but with a temporal factor. It counts the number of practical zones that are found in the theoretical order. If the theoretical sequence says that **b1** comes after **a1**, then the measure will only count **b1** if the zone **a1** has already been seen. The measure does not require **a1** and **b1** to be directly one after the other, only that they are in the right order. Once again, we divide the number of zones correctly identified by the cardinality of $\mathbf{S}$ to compare sequences between them. None of our paths use the same zone twice, so this metric does not have to take repetition into account.

4.1.3 Levenshtein distance

The Levenshtein distance (L dist), also referred as the edit distance, is a well-known measure to compare sequences. It consists of counting the number of insertion, deletion and replacement to transform a string into another one. It is a measure of dissimilarity often used in text processing or in bioinformatics to compare DNA sequences. Here, we use it to find the difference between the practical sequence and the theoretical one. We use an insertion and deletion cost of one and a replacement cost of two. As you will see, this measure is not very useful with our data because the length difference between the practical and the theoretical sequences is too important. All distances are in the same range, no matter the length of the path or the room.

4.1.4 Euclidean distance

The last metric we used is the Euclidean distance (E dist). But, to use it, we have to change how we consider our data because otherwise we would not know to what theoretical zone we should compare a given practical zone. The aim of the pre-treatment we need to apply is to make sure both sequences are of the same length. To do so, we take the raw data from the antennas for the whole path and we divide it in $|\mathbf{S}|$ bags. We then average each bag to get a single vector that we can ask our IPS to classify. We compute the Euclidean distance between this averaged zone and the reference zone and sum all the distances together to find the distance between the theoretical sequence and the practical readings. Finally, we divide the sum by $|\mathbf{S}|$ to get the average distance of the zones.

Table 1 shows the results we obtained with those four metrics on the six rooms. As we can see, we do not even reach 60% of zone recognition on average, and barely 35% if we consider the order. As for the other metrics, they

Table 1: Raw results

Metric	TZF	STZF	L dist	E dist
Bedroom	0,409	0,230	74,275	2,744
Bathroom	0,610	0,291	76,500	4,501
Hall	0,426	0,245	85,300	5,451
Kitchen	0,845	0,707	79,350	2,338
Living room	0,596	0,321	100,950	2,291
Dining room	0,544	0,374	41,800	2,762
Average	**0,572**	**0,361**	**76,363**	**3,348**

Table 2: Effect of the moving average in the kitchen

Size	Metric			
N	TZF	STZF	L dist	E dist
5	0.814	0.656	81.225	3.880
10	0.772	0.652	66.150	4.099
20	0.745	0.575	49.025	4.479
30	0.687	0.518	40.85	4.411
40	0.673	0.496	34.125	4.573

show that the distance between the sequences is reasonable. The Euclidean distance shows that we do not miss by a substantial difference for most rooms. In the kitchen, the system only misses by an average of 80cm, which is not critical in most cases when doing ADL recognition as it is still precise enough to distinguish the sink from the oven or one counter from another.

4.2 Filters

The accuracy we got with the raw data indicated to us that there was a lot of room for improvement using post-treatment. While looking at a real-time map of the tracked object, we found that the readings were not stable, the object often *teleporting* far of its true position. So, we developed four filters with the goal of stabilising the readings and the predicted zones.

4.2.1 Simple Moving Average

The first filter we implemented is the simple moving average. The idea is to replace each new reading by the average of the readings of N previous ones, were N is chosen with regards to the number of readings we get for each zone when moving or, to simply put, the non-null readings. This simple moving average greatly helped to stabilise our data by significantly reducing the effect of aberrant data. We tried various values for N and the results are shown in Table 2. We can see that all metrics except the edit distance decrease when N increases, indicating that we were using readings that were too old. The only metric that gets better as N increases is the edit distance, showing that the simple moving average tends to create shorter sequences. Table 3 shows the results for all rooms when using an N of 5. Except for the kitchen, this filter improved the accuracy everywhere.

4.2.2 Weighted Moving Average

In a second trial, we extended the moving average to its general case. Instead of giving each reading the same weight, this version allow us to give more or less importance to the readings regarding their rank in the average. It can be expressed as:

Table 3: Results for the moving average

Metric	TZF	STZF	L dist	E dist
Bedroom	0,45	0,251	73,475	2,572
Bathroom	0,686	0,323	85,175	4,097
Hall	0,43	0,246	87,250	6,155
Kitchen	0,814	0,656	81,225	3,880
Living room	0,642	0,364	109,775	2,524
Dining room	0,572	0,391	49,825	3,319
Average	**0,599**	**0,372**	**81,121**	**3,758**

$$\bar{R} = \frac{\sum_{n=0}^{N-1} C_n * R_{N-n}}{\sum_{n=0}^{N-1} C_n} \quad (1)$$

where $\bar{R}$ is the weighted average, R_{N-n} is the N^{th} reading and C_n is the C^{th} weight coefficient. Note that we consider N_0 as being the current reading. We also have to divide by the sum of all coefficients to make sure they sum to one. We tried four different distributions for the weighting. Those distributions imply the following functions: the logarithm function, the linear function, the exponential function with base 2, and finally, the factorial function.

Logarithmic distribution: The first weighting distribution is related to the natural logarithm function. This distribution is defined as follows:

$$C_n = \frac{1}{\ln(n+2)} \quad (2)$$

We use $n+2$ to avoid the part of the logarithm function that is negative and rapidly growing. This way, we get a coefficient that is slowly decreasing, giving less and less weight to the older results. Table 4 clearly shows that the accuracy increases with N, while still staying under the accuracy of the raw data and therefore, also under the accuracy of the simple moving average.

Linear distribution: A second weight function is a simple decreasing linear function. The idea is to decrease the importance of older readings in a linear way, as in:

$$C_n = 1 - 0.1n \quad (3)$$

An interesting fact about this function is that the tenth value is nullified. Like the log function, the linear function increases accuracy as **n** grows bigger. This is one of the weight functions that works best with our data, yet not enough to offer an increase in accuracy over the raw data.

Exponential distribution: The third weighting distribution is related to the exponential function, with a base 2. Mathematically, it is:

$$C_n = \frac{1}{2^{a\,n}} \quad (4)$$

In simple words, it means that every older reading weighs half of the preceding one when $a = 1$. We tried many value for the modifier a. With $a = 1$ the results were terrible. However, with $a = 1/8$ or $a = {}^1/_{16}$ the results were much better.

Factorial distribution The last weighting distribution is related to the factorial. The factorial function is the product of all positive integers less or equal to a given positive integer. The mathematical expression of our coefficient is:

$$C_n = \frac{1}{a\,n!} \quad (5)$$

Table 4: Effect of N and the weight function with the moving average in the kitchen

Parameters		Metric			
N	Weight function	TZF	STZF	L dist	E dist
5	Log	0,367	0,288	62,125	5,696
10	Log	0,489	0,374	72,975	5,510
20	Log	0,593	0,430	71,375	5,458
30	Log	0,565	0,385	62,725	5,487
40	Log	0,552	0,400	56,3	5,431
5	Linear	0,417	0,315	66,3	5,403
10	Linear	0,533	0,393	74,425	5,364
20	Linear	0,763	0,648	108,475	5,549
30	Linear	0,783	0,600	112,225	5,415
40	Linear	0,789	0,565	83,575	5,778
5	Exp a=1/8	0,426	0,329	67,65	5,438
10	Exp a=1/8	0,666	0,498	84,375	5,277
20	Exp a=1/8	0,685	0,493	76,475	5,323
10	Exp a=1/16	0,661	0,499	85,475	5,275
20	Exp a=1/16	0,733	0,580	74,525	5,271
10	Fact a=1/4	0,623	0,455	92,225	5,837
10	Fact a=1/8	0,821	0,677	113,200	5,237
10	Fact a=1/16	0,886	0,776	117,700	5,532
10	Fact a=1/32	0,875	0,756	120,125	4,949
10	Fact a=1/128	0,878	0,720	117,375	4,441

Where $n!$ is the factorial of n. Since the factorial rapidly increase, the length of the moving average (**N**) has no impact on the result. We therefore only use N=10 and tweak a instead.

Of the four weighting distributions we tried, the linear one offered the best performance without tweaking. However, with tweaking we were able to obtain good results on all metrics and even higher accuracy by tweaking the factorial distribution. The factorial distribution seems to outperform the SMA indicating that giving slightly less weight to older readings is a good idea. We suppose that better results could be obtained with a more sophisticated tuning of any of the distributions. We saw that the accuracy decreased as N increased in the SMA. Here, it is generally the opposite, with the accuracy increasing with N in most distributions.

4.2.3 Blocking neighbour

The blocking neighbour filter prevents *teleportation* by only allowing movement between neighbouring zones. For example, let us say the object is in B3. Then, the filter will reject any zone that is not a direct neighbour, namely A2, A3, A4, B2, B4, C2, C3 and C4. The main issue with such a filter is when it misses a valid movement. If, for any reason, the object is now in B6 and the collected readings did not predict to both B4 and B5, then the filter might be forever stuck thinking the object is in B3 and thus, reject every new zone. To prevent that, we added a simple counter that counts all successive rejection by the filter. When a certain limit is reached, the filter sets its current zone to the predicted zone. Table 5 shows the results of this filter alone in all rooms. The results are generally worse than without this filter, except for the edit distance. Indeed, by rejecting numerous zones, the practical sequence is shorter than before and therefore less *delete* operation are needed to equalize the sequences.

Table 5: Results for the blocking filter alone

Metric	TZF	STZF	L dist	E dist
Bedroom	0,254	0,170	30,400	3,501
Bathroom	0,498	0,239	31,575	5,462
Hall	0,337	0,193	33,425	3,438
Kitchen	0,730	0,616	36,825	3,140
Living room	0,480	0,259	39,225	2,741
Dining room	0,467	0,342	22,200	2,939
Average	**0,461**	**0,303**	**32,275**	**3,537**

Table 6: Results for the limiting filter alone

Metric	TZF	STZF	L dist	E dist
Bedroom	0,797	0,653	124,775	2,297
Bathroom	0,886	0,660	236,350	5,732
Hall	0,814	0,711	64,800	1,218
Kitchen	0,963	0,931	146,300	2,171
Living room	0,913	0,753	207,250	2,143
Dining room	0,835	0,730	91,900	2,329
Average	**0,868**	**0,740**	**145,229**	**2,648**

4.2.4 Limiting neighbour

The last filter we integrated to our ITS is a less restrictive version of the blocking neighbour. Instead of rejecting movement between zones that are not neighbouring each other, we limit the movement to only one zone at the time. Again, let us say the object is in B3. The newly predicted zone by the random forest is G1. While the blocking filter would only reject G1 and say the object is still in B3, this new filter will say the object is now in C2. Table 6 shows how greatly this filter increases the accuracy of our system. Globally, it increases the accuracy by 30% when compared to the raw data for all rooms. We reach a 74% of sequential accuracy in average. Results would be even greater if not for the bedroom and the bathroom where results are still low. However, this is consistent with the fact that these room are covered by less antennas than the others. Still, the distance metrics remain high, indicating that there are many offset readings that increase the distances. The effect of this filter can be seen on Figure 4. The opacity of the zones increase given the number of times they are visited. The left side of the figure shows what it looks like when no filters are applied. The right side shows the effect of the limiting filter. It is obvious that the limiting filter concentrates the zones around the correct path.

4.3 Combining filters

We previously said that it would not make sense to use the first two or the last two filters together, but nothing prevents us from combining the moving averages with the zone filters. As the blocking neighbour does not perform well, we will only present the results when combining with the limiting neighbour. In Table 7, we see that the combination of the simple moving average (n=5) with the limiting neighbour does not offer any significant gain over the filter alone. It might be better in certain rooms, but it is worse in others and so the overall gain is null. We can still see that the edit distance is reduced by the moving average. In Table 8, we have the results of combining the same limiting filter with the weighted moving average. As it was the best fit, we used the factorial weight function (n=10, a=1/16). Surprisingly,

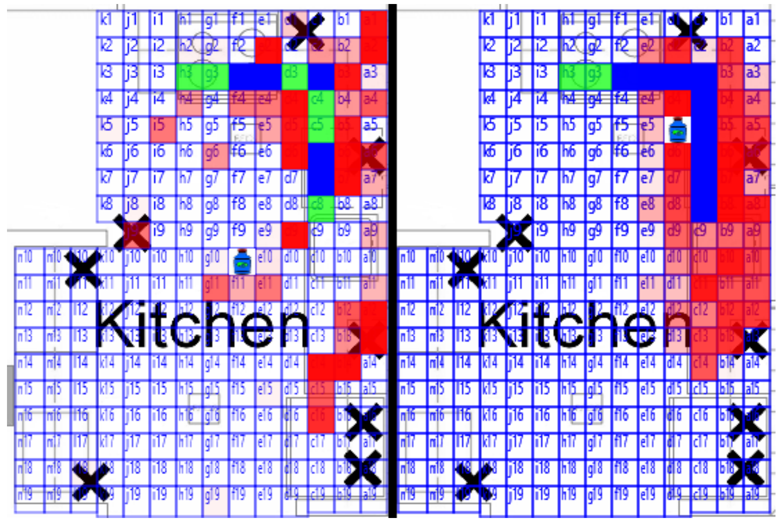

Figure 4: Map of the kitchen with raw data versus the limiting filter. The green squares show the expected path and they turn blue when correctly identified by the algorithm. Red squares are wrongly identified zones.

Table 7: Limiting filter with the moving average

Metric	TZF	STZF	L dist	E dist
Bedroom	0,820	0,650	159,175	2,689
Bathroom	0,946	0,827	275,650	4,547
Hall	0,762	0,514	204,975	4,886
Kitchen	0,970	0,937	153,000	3,312
Living room	0,926	0,754	205,925	2,368
Dining room	0,843	0,734	99,325	2,989
Average	**0,878**	**0,736**	**183,008**	**3,465**

Table 8: Limiting filter with the weighted moving average using factorial

Metric	TZF	STZF	L dist	E dist
Bedroom	0,895	0,704	195,425	4,561
Bathroom	0,963	0,773	316,025	4,721
Hall	0,746	0,594	291,525	4,937
Kitchen	0,944	0,926	188,150	3,518
Living room	0,923	0,814	235,000	2,226
Dining room	0,826	0,723	112,050	2,295
Average	**0,883**	**0,756**	**223,029**	**3,710**

it produces the same results that we see in Table 7, strongly suggesting the limiting filter is the key to our improvements.

4.4 Effect of sampling speed

We said earlier that our IPS was trained with the antennas set at 750ms between readings and that we reduced it to 20ms for this experiment. Such a change could have greatly affected the accuracy of our IPS. To make sure it did not, we conducted a simple experiment. We designed a special path over the kitchen counter and we crossed it with the water bottle two times, one at normal speed with antennas set at 20ms and one at very slow speed with antennas set at 750ms. Our simple experiment indicated that the accuracy was better at 20ms when no filters were used. However, this is certainly because we were still moving too fast when at 750ms between readings, reducing the chance for our random forest to predict the right zone at least one time. Still, the difference in accuracy was small. Further experiments should be conducted on this subject.

5. DISCUSSION

In the previous sections, we saw how we used our indoor positioning system to create an indoor tracking system. These two systems will ultimately play a crucial role to increase the granularity and the accuracy of the ADL recognition in a smart home. However, we saw in section 4 that tracking is not accurate with only the raw data. Even when the object is not moving, it is *teleporting* everywhere, moving all around the room. When we added the moving average filter, it mostly stopped teleporting, stabilising to the correct zone or the neighbouring ones. Then, we tried to use a weighted moving average to see if the motionless object would gain total immobility. It did not, because there were too many incorrect predictions by the IPS. Indeed, most of the tuning tests were made in the kitchen, where the IPS had an accuracy of about 95%. So, one out of twenty times a bad reading happens and the older readings cannot compensate for it given their deceasing weight. Moreover, all versions of the moving average became less accurate when we started moving the object.

On the other hand, filters that affect the liberty of movement of the object have better potential. The blocking filter kills almost all movement for the motionless object, as it was designed to do, but is too aggressive. If it missed the first valid movement it would never be able to move after that. The fail limit we described in section 4 allowed us to reduce the effects of this constraint, but it was still too aggressive to catch all valid movements. By only allowing moving of one zone at a time, the limiting filter was better adapted. It could never get stuck like the blocking filter. It allowed us to move to valid zones that were never reached by the R-FID readings. It also produced plausible paths, with many valuable sequences in them. It would take many consecutive bad readings to make this filter diverge from its correct path, reducing their effect even more than the moving average. Computationally, the filters works quickly. It only needs compute some subtractions or additions, thus being of complexity O(1). Figure 4 shows how greatly it concentrated all *predicted* zones to the near path zones. This will be of great help to the final ADL recognition system.

Overall, our new ITS is easy to implement and sufficiently precise for our purpose. The main factor decreasing the tracking performance is human interferences. Indeed, the IPS was built from datasets built without any human in the smart home. When a human stands between the object and an antenna, the corresponding signal intensity is affected. This is clearly the biggest challenge for our system. It particularly affects DTs since the decision leading to a zone or another does not take the neighbourhood into account. For example, if the signal of an antenna decrease and that antenna was the most discriminating for the classification of this zone, the tree might explore a branch that is irrelevant and predict a zone far away from the intended one. Since we cannot simulate all possible ways a human can interfere, the IPS cannot be altered to take this into account.

6. CONCLUSION

In conclusion, we presented a system capable of indoor tracking with more than 75% of the exact path recognized. This system is based on a qualitative indoor positioning system that uses zones of varying sizes. While the accuracy may not seem high, the system also recognizes nearby zones to the path, which is sufficient for our final goal of building a complete system for the recognition of ADLs using passive RFID. Therefore, we can say that our method is working, even if human interference was worse than expected.

The main disadvantage of using decision trees instead of a classical localization method such as trilateration, for example, is that an error causes an unpredictable move, an unwanted *teleportation*. In our future work, our team will consider using larger size zones for the IPS and the ITS. The team is currently investigating the exploitation of qualitative ITS for the problem of activity recognition and results should be published within few months. Additionally, the team will create a tweaking software that will automatically find the best modifiers for all distributions.

7. ACKNOWLEDGEMENT

We wants to thank INTER for their financial support.

8. REFERENCES

[1] A. Bekkali, H. Sanson, and M. Matsumoto. Rfid indoor positioning based on probabilistic rfid map and kalman filtering. In *Wireless and Mobile Computing, Networking and Communications, 2007. WiMOB 2007. Third IEEE International Conference on*, pages 21–21. IEEE, 2007.

[2] F. Bergeron, K. Bouchard, S. Gaboury, S. Giroux, and B. Bouchard. Indoor positioning system for smart homes based on decision trees and passive rfid. In *Pacific Asia Knowledge Discovery and Data Mining Conference (PAKDD), 2016 Conference on*. PAKDD, In press.

[3] K. Bouchard, D. Fortin-Simard, S. Gaboury, B. Bouchard, and A. Bouzouane. Accurate trilateration for passive rfid localization in smart homes. *International Journal of Wireless Information Networks*, 21(1):32–47, 2013.

[4] L. Chen, J. Hoey, C. D. Nugent, D. J. Cook, and Z. Yu. Sensor-based activity recognition. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 42(6):790–808, 2012.

[5] L. Chen, C. D. Nugent, and H. Wang. A knowledge-driven approach to activity recognition in smart homes. *Knowledge and Data Engineering, IEEE Transactions on*, 24(6):961–974, 2012.

[6] S. Chernbumroong, S. Cang, A. Atkins, and H. Yu. Elderly activities recognition and classification for applications in assisted living. *Expert Systems with Applications*, 40(5):1662–1674, 2013.

[7] G. Demiris, B. K. Hensel, M. Skubic, and M. Rantz. Senior residentsâĂŹ perceived need of and preferences for âĂIJsmart homeâĂİ sensor technologies. *International journal of technology assessment in health care*, 24(01):120–124, 2008.

[8] E. Frank Lopresti, A. Mihailidis, and N. Kirsch. Assistive technology for cognitive rehabilitation: State of the art. *Neuropsychological rehabilitation*, 14(1-2):5–39, 2004.

[9] J. Han, C.-S. Choi, W.-K. Park, I. Lee, and S.-H. Kim. Smart home energy management system including renewable energy based on zigbee and plc. *Consumer Electronics, IEEE Transactions on*, 60(2):198–202, 2014.

[10] X. Liu, J. Peng, and T. Liu. A novel indoor localization system based on passive rfid technology. In *Electronic and Mechanical Engineering and Information Technology (EMEIT), 2011 International Conference on*, pages 4285–4288. IEEE, 2011.

[11] U. Nations. World population ageing 2013. *Department of Economic and Social Affairs PD*, 2013.

[12] L. M. Ni, Y. Liu, Y. C. Lau, and A. P. Patil. Landmarc: indoor location sensing using active rfid. *Wireless networks*, 10(6):701–710, 2004.

[13] P. Rashidi and D. J. Cook. Keeping the resident in the loop: Adapting the smart home to the user. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, 39(5):949–959, 2009.

[14] C. Roehrig, A. Heller, D. Hess, and F. Kuenemund. Global localization and position tracking of automatic guided vehicles using passive rfid technology. In *ISR/Robotik 2014; 41st International Symposium on Robotics; Proceedings of*, pages 1–8, June 2014.

[15] A. M. Sisko, S. P. Keehan, G. A. Cuckler, A. J. Madison, S. D. Smith, C. J. Wolfe, D. A. Stone, J. M. Lizonitz, and J. A. Poisal. National health expenditure projections, 2013–23: faster growth expected with expanded coverage and improving economy. *Health Affairs*, 33(10):1841–1850, 2014.

[16] L. Yang, J. Cao, W. Zhu, and S. Tang. A hybrid method for achieving high accuracy and efficiency in object tracking using passive rfid. In *Pervasive Computing and Communications (PerCom), 2012 IEEE International Conference on*, pages 109–115. IEEE, 2012.

[17] J. Yim. Introducing a decision tree-based indoor positioning technique. *Expert Systems with Applications*, 34(2):1296–1302, 2008.