Real-time Gesture Recognition Using RFID Technology

Parvin Asadzadeh, Lars Kulik and Egemen Tanin

NICTA, Department of Computer Science and Software Engineering, University of Melbourne, Melbourne, Australia
{parvina, lars, egemen}@csse.unimelb.edu.au

Abstract. This paper presents a real-time gesture recognition technique based on RFID technology. Inexpensive and unintrusive passive RFID tags can be easily attached to or interweaved into user clothes. The tag readings in an RFID-enabled environment can then be used to recognize the user gestures in order to enable intuitive human-computer interaction. People can interact with large public displays without the need to carry a dedicated device, which can improve interactive advertisement in public places. In this paper, multiple hypotheses tracking is used to track the motion patterns of passive RFID tags. Despite the reading uncertainties inherent in passive RFID technology, the experiments show that the presented online gesture recognition technique has an accuracy of up to 96%.

1 Introduction

Large digital displays have become a well-established medium for advertising in public places. However, the efficacy of outdoor advertising can be improved by providing intuitive ways for users to interact. Interfaces that utilize the users' gestures as they are standing in front of a public display offer an intuitive and natural way of interaction. Gesture recognition techniques in human-computer interaction can be broadly divided into vision-based and device-based techniques. Murthy and Jadon [13] give a comprehensive review of vision-based gesture recognition techniques. Their challenges are cluttered backgrounds and varying illuminations, especially in public places. Moreover, recording user movements using video is resource intensive. Device-based hand gesture recognition techniques, on the other hand, typically use customized equipment such as gloves with embedded sensors [10, 15–18], mobile devices such as NFC-enabled mobile phones [8, 6] or accelerometer-enabled devices such as the Wiimote [18] to measure user movements. Glove-based techniques are relatively intrusive while the less intrusive devices are not readily available in public places.

In this paper, we propose the use of passive Radio Frequency IDentification (RFID) to provide hand gesture-based human-computer interaction. RFID is an effective automatic identification technology that allows for easy proximity sensing of tagged objects. Objects tagged with small inexpensive and unintrusive passive RFID tags can be sensed from a few centimeters up to several meters. Passive RFID tags operate without a battery and it is possible to tag large collections of objects with multiple tags. All RFID tags contain unique identification numbers along with other data to easily identify tagged objects.

In a previous work [2], we employed multiple hypotheses tracking to track the motion patterns of RFID tags in an offline gesture recognition technique. A combined tag consisting of multiple subtags was used to increase the readout reliability of the RFID readers. In this paper, we design and evaluate an online gesture recognition technique, which is capable of real-time gesture recognition with an accuracy comparable to the offline gesture recognizer. Detailed experiments are conducted to compare and evaluate both techniques. Further, Levenshtein distance [11] is used to find the closest matching gesture to a tag track. It is shown that the *online* technique is capable of real-time gesture recognition with an accuracy of up to 96%, without requiring any learning or training. We also show our findings for independent gesture recognition of two users.

2 RFID Localization

There are a number of location-sensing techniques based on *active* RFID technology that measure the received signal strength (RSS) to estimate a tag's location [5]. However, the use of RSS in tag localization is more accurate for active tags since they carry a power source and hence, have more stable performance within crowded environments.

Because of the various error sources in passive RFID systems, reliable operation as the tag moves in the environment is inherently difficult and presents a significant challenge. To localize *passive* RFID tags, some researchers use angulation technique to estimate the direction of arrival of a tag signal [14, 21]. Furthermore, Wilson *et al.* [20] use the percentage of positive tag reads as an indication of distance and Chawla *et al.* [7] infer a tag's position based on the relative power level that is necessary for a reader to detect the tag.

A number of passive RFID-based location-sensing systems use only the presence information from RFID readers to localize a tag. RFID readers can only sense the presence of a tag within their detection fields, providing proximity information of the tag, but they cannot directly determine the tag's distance to the reader. However, one positive detection of a tag greatly reduces its possible locations, since it indicates that the tag is in the reader's detection field. To estimate the whereabouts of the tag more precisely, the tag readings from a mobile RFID reader from different vantage points [12] or the output of several stationary readers [4, 9] can be combined.

Existing passive RFID-based location-sensing techniques mainly focus on localization of stationary tagged objects [4, 7, 9, 14, 20, 21]. A few of the proposed methods also try to localize and track moving objects [14, 20, 21] under particular conditions. However, none is capable of accurate online tracking of arbitrarily moving tags.

3 RFID-based Gesture Recognition System

We have built an experimental system using the SkyeModule M9 UHF reader from SkyeTek [1] (Figure 1(a)) with a maximum read range of approximately two and a half meters, and their linear broadband UHF antennas (Figure 1(b)). We chose ISO 18000-6C tags (Figure 1(c)) since they are small and compatible with the employed readers.

The tag-antenna orientation determines if the tag receives enough energy to be detected by the antenna. To increase the readout reliability of a tag when it is close enough

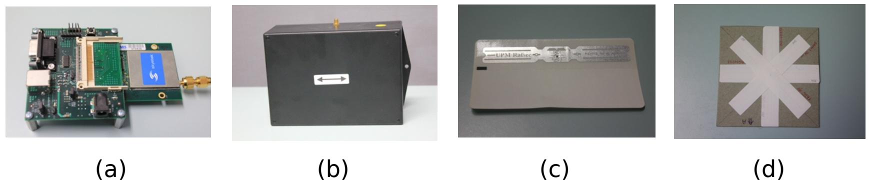
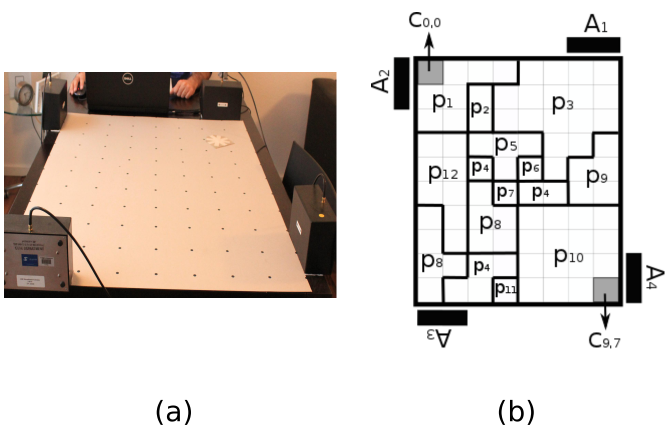


Fig. 1. (a) reader (b) antenna (c) single tag (d) combined tag

to an antenna, we use a combined tag instead of a single tag. As shown in Figure 1(d), our combined tag consists of four individual co-located tags, where each tag is rotated 45 degrees to its neighbor tag. All single tags, or *subtags*, in a combined tag have different identifiers but they are combined to represent one *super* tag. We monitor a 80cm by 100cm rectangular area on a desk (Figure 2(a)). The rectangular monitored area is divided into eighty equally-sized square cells $C_{0,0},\ldots,C_{9,7}$ (Figure 2(b)) with the side equal to the width of the super tag, which is 10cm. The reader works in inventory mode, which runs an anti-collision protocol to read many tags simultaneously. It is connected to four antennas ($A_1 - A_4$) via a multiplexer, which are placed just outside the monitored area (Figure 2(a)). Time slicing is used to avoid an interference between the antennas. The four antennas are sequentially energized, which in turn return the tag identifiers in their detection fields. The RFID readings are then sent to the gesture recognizer on a laptop via a USB connection.



partition	code
p_1	0110
p_2	0100
p_3	1100
p_4	1011
p_5	1110
p_6	1101
p_7	1010
p_8	0011
p_9	1000
p_{10}	1001
p_{11}	0001
p_{12}	0010

(c)

Fig. 2. The monitored area

3.1 Space Partitioning

RFID readers can only sense the presence of a tag within their detection fields, but they cannot directly determine the tag's distance to an antenna. To estimate the whereabouts of the tag more precisely, the output of several stationary readers or antennas can be combined. In this case, the monitored area is divided into multiple partitions so that each partition is in detection fields of a particular set of antennas. We refer to the technique of using stationary RFID antennas to partition a space as *space partitioning*.

Figure 2(b) shows the partitioning of our monitored area into twelve partitions by $A_1 - A_4$ with overlapping fields, at a given time. We differentiate the created partitions by their assigned codes. A 4-bit binary code, $BC[C_k] : b_1.b_2.b_3.b_4$, is assigned to each cell C_k, b_j is set if any subtag of a supertag in C_k is within the detection field of A_j. Figure 2(c) shows the binary code assigned to each partition. Partition p_1, for example, is assigned a binary code of 0110, since it is only in detection fields of A_2 and A_3.

Because of false negative and positive readings, there are always unavoidable uncertainties about the presence of RFID tags. This is the biggest challenge in designing a passive RFID-based system, especially when antennas are in close proximity. To cope with uncertainties in RFID readings, instead of assigning a fixed code to each cell of the monitored area, each cell C_k is assigned a sequence of possible codes, BC_i, and their associated codes $W(C_k, BC_i)$: $\{(BC_1, W(C_k, BC_1)), \ldots, (BC_p, W(C_k, BC_p))\}; p_{max} = 2^n$, where n is the number of antennas. The weight assigned to each code shows how probable the occurrence of that particular code is.

3.2 RFID-based Gesture Recognition

Whenever a user draws a gesture by moving a tag on the monitored area, a sequence of RFID readings is generated. We use multiple hypotheses tracking approach [3] to track a tag, which generates a set of hypotheses to account for all possible tracks of the tag based on the received RFID readings. The key principle of this approach is that the track update decisions are deferred until more RFID readings are received. A flow diagram of both offline and online gesture recognizers is shown in Figure 3.

Offline Gesture Recognition On the receipt of new data (the k^{th} set of RFID readings), BC_k code is generated in CODEGEN (code generator), as explained in Section 3.1. The INI (initiator) process creates a hypotheses tree once BC_0 is received, which includes as its children the cells C_i with $W(C_i, BC_0) > 0$.

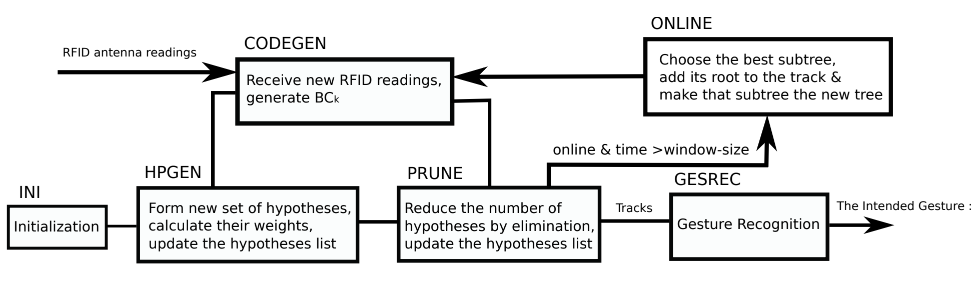


Fig. 3. The gesture recognizer architecture

On the receipt of new codes BC_k at time k, HPGEN (hypotheses generator) expands each hypothesis into a set of new hypotheses by considering all possible new locations of the tag, which are determined by considering the possible movements of the tag. We assume that the tag is either moving horizontally or vertically, as shown in Figure 4(a). The antennas' reading speed is high enough to ensure that the tag does not move more than one cell away between any two consecutive readings. Furthermore, Figures 4(b)-(c) show two illegal local movements, both assuming that the tag is always moving forward. Moreover, it can be assumed that the tag does not remain in the same cell for long – the speed of the tag is greater than a threshold.

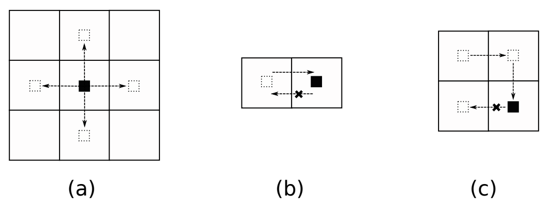


Fig. 4. (a) Possible local moves (b)-(c) Illegal local moves

Figure 5 shows a sample hypotheses tree after three sets of readings are received. Each branch of the hypotheses tree represents a possible track of the tag and nodes of the tree are the cells the tag has traversed. A hypotheses list is also created that contains all possible current locations of the tag along with the corresponding track weight, SW_i, which is the sum of the weights of all cells contained in that track. The track weight is later used to assess track validation as well as track selection.

To eliminate unlikely hypotheses, PRUNE (pruning) uses a *weight-based* pruning method. In weight-based pruning, the tracks are evaluated based on their weight and tracks that are unlikely to reach a minimum weight requirement are removed from the hypotheses tree. This allows us to use a threshold to reject a track rather than picking the nearest matched track. The tree expansion process continues until the end of the gesture. After the last validation phase, the most likely tracks are the ones with $SW > \beta \times SW_{max}$ ($\beta = 0.90$, in our tests). For each likely track, the GESREC (gesture recognizer) finds the gesture that best matches that track, using the Levenshtein ditance, explained later in this section. The output gesture is the one with maximum probability of occurrence.

Online Gesture Recognition The complete hypotheses tree for every gesture grows exponentially as more readings are processed. Thus, there is a clear potential explosion in the number of possible tracks (hypotheses) that our system can generate. Therefore, the offline algorithm is not suitable for real-time gesture recognition. In the online version of our algorithm, the complete hypotheses tree is created during the first *window-size* time steps. The window-size is best to be set to the largest diameter of the partitions to make sure that sufficient evidence is gathered before the tag track is updated.

At any time step afterwards, the hypotheses tree is replaced with one of its subtrees (Figure 5) and the root of the new hypotheses tree is appended to the tag track. Both the number of children of a subtree as well as the sum weights of its tracks are used to compare the subtrees of a hypotheses tree. Section 4 discusses the performance of the online gesture recognizer and shows that it can recognize gestures in real-time with an accuracy comparable to the offline gesture recognition technique.

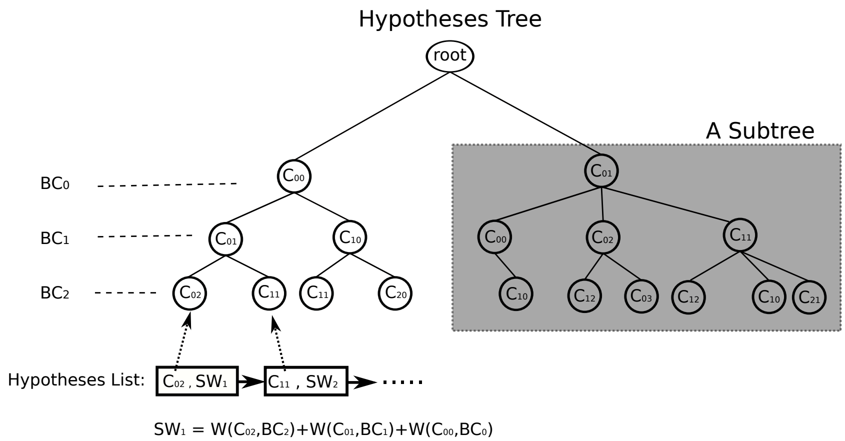


Fig. 5. A hypotheses tree

The Matching Algorithm At the end of both offline and online recognition techniques, the GESREC (gesture recognizer) is provided by a set of the most probable tracks or the sequences of cells traversed by the tag. The Levenshtein distance [11] is used to find the closest matching gesture to each track. The Levenshtein distance is a measure of the similarity between two strings. It is the number of deletions, insertions, or substitutions required to transform one string into the other string.

In order to use Levenshtein ditance for comparing each track and every candidate gesture in our alphabet, it is necessary to represent both track and each candidate gesture as strings. Each track is a sequence of traversed cells and thus, can be converted to a sequence of directional moves. As shown in Figure 4(a) the tag is assumed to move in only four directions of up, down, right, and left. Consequently, a track is transformed to a string consisting of *u*, *d*, *r*, and *l*, representing moving up, down, right and left, respectively. To find the nearest gesture to the tag track, each candidate gesture is first converted to the possible strings of the same size. For example, to calculate the distance of an eight-cells long track to an *up-right* gesture, the possible strings of the candidate gesture are *uuurrrrr*, *uuuurrrr* and *uuuuurrr*, assuming that each movement element is greater than two. The gesture matching to the track is the one with the minimal distance to the track and the output gesture is the one which matches to the highest number of tracks. The gesture is unrecognizable if more than one gesture matches the track or the minimal distance is higher than a threshold. Figures 6 shows two sample tracks along with their nearest matched gesture.

4 Gesture Recognition Experiments

This section presents the test results of both offline and online gesture recognition techniques. It is shown that the online gesture recognition algorithm (Section 3.2) provides real-time recognition of gestures, with a comparable recognition rate to the offline algorithm. In all experiments, the antennas are working in different time slots to avoid

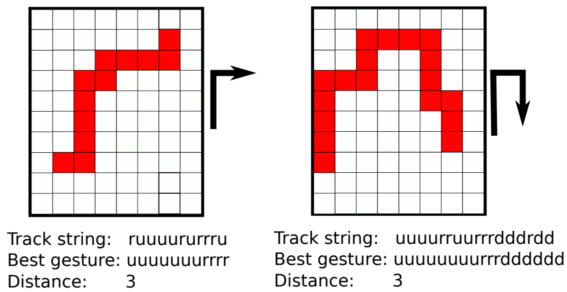


Fig. 6. Matcher

interference. One reading cycle varies between 530 and 1328 *milli-seconds*, depending on the number of tags in the field of each antenna. Moreover, both algorithms were run on the same machine, a 2.16 GHz Intel Core 2 Duo laptop with 2 GB of RAM.

The Gestures We collected quantitative data to evaluate the performance of both offline and online gesture recognition techniques. Figure 7 shows our tested alphabet of gestures. We tested 16 single gestures of $G1$ to $G16$, consisting of up to 4 *gesture elements – same direction movements*. Users performed gestures by moving the supertag on a desk. Gestures had different sizes and were performed in different parts of the monitored area. Gesture $G5$, for example, consists of two elements. It was performed when the user moved the tag from bottom to up and then left to right, while facing A_1, anywhere within the monitored area.

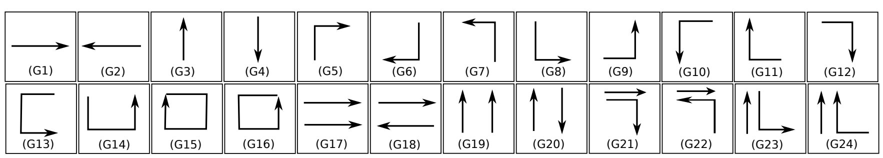


Fig. 7. Example gestures

The accuracy of the proposed gesture recognition techniques depends on the size of the partitions relative to the gestures. The created partitions must be small enough to ensure that every *gesture element* crosses more than one partition. Otherwise, without further information, inferring the direction of a tag's movement is not possible. In our partitioned area (Figure 2(b)), the maximum diameter of partitions is five cells. The shortest length of the gesture elements is, therefore, set to six cells in all experiments. A total of 640 samples were collected: 40 samples of each gesture G_1 to G_{16}. Each gesture is performed in two sample sets of different sizes. The first 20 samples of each gesture consist of gesture elements of the minimum size of six, while the second 20 samples of each gesture consist of slightly longer gestures - with movement elements of seven up to eight cells.

The first and second sample sets of gestures G_1 to G_4 crossed 6 and 8 cells and were performed in an average of 5456 and 7204 *ms*, respectively. The first and second samples sets of gestures G_5 to G_{12}, which consist of two elements each, crossed an average of 11 and 13 cells and were performed in an average of 10023 and 11831 *ms*, respectively. The first and second samples sets of gestures G_{13} and G_{14}, which consist of three elements each, crossed 16 and 19 cells and were performed in an average of 14419 and 16988 *ms*, respectively. The first and second sample sets of gestures G_{15} and G_{16}, which consist of four elements each, crossed an average of 21 and 25 cells and were performed in an average of 19187 and 22812 *ms*, respectively. Overall, the gestures were performed with an average speed of 0.11 *m/s*.

The M9 SkyeTek reader is relatively slow with a tag interrogation rate of 25 tags per second. In fact, the passive tag read rate can be up to 500 tags per second [19], which is 20 times faster than M9 SkyeTek reader. Therefore, a commercial system is likely to be able to recognize much faster gesture movements.

Offline Test Results The size of the hypotheses tree and consequently, the run-time of the offline gesture recognition algorithm grows exponentially with respect to the size of the gestures. Therefore, the offline gesture recognition technique is unable of real-time recognition of gestures. In fact, in our used machine, it failed to recognize gestures of longer than 13-cells long in less than a second. Therefore, the offline recognition algorithm results are presented for the gestures shorter than 13 cells. The average rate of correctly recognized gestures of all 480 (40 samples of each gesture G_1 to G_{12}) gestures was 94%. However, the second sample set of gestures – slightly longer gestures – were recognized with an accuracy of 98%, which was 7% higher than the recognition rate of the first sample set of gestures, which was 91%. Table 1 shows the number of correctly recognized samples (CR_{off}) for each gesture G_1 to G_{12}. A recognition error occurred when the system was unable to match the track to a unique gesture, or it yielded a gesture other than the drawn one. In the latter case, the output gesture (OG_{off}) is shown on the third row of Table 1.

Online Test Results The online gesture recognition technique was able to recognize all gestures in real-time – in an average of 50 *ms* in our experiments. This is due to the fact that the depth of the hypotheses tree does not grow with respect to the length of the gestures (Section 3.2). Overall, the average rate of correctly recognized gestures of all 640 (40 of each gesture G_1 to G_{16}) gestures was 92%. However, the second sample set of gestures – slightly longer gestures – were recognized with an accuracy of 96%, which was 8% higher than the recognition rate of the first sample set of gestures, which was 88%. CR_{on} and OG_{on} values are shown in the fourth and fifth rows of Table 1.

Gesture	G_1	G_2	G_3	G_4	G_5	G_6	G_7	G_8	G_9	G_{10}	G_{11}	G_{12}	G_{13}	G_{14}	G_{15}	G_{16}
CR_{off}	39	39	32	39	40	38	38	36	39	40	37	36	-	-	-	-
OG_{off}	G_4	G_6	$G_{0,1}$	G_7	-	G_3	$G_{1,2}$	$G_{3,5,11}$	G_4	-	$G_{1,2}$	$G_{0,3,7}$	-	-	-	-
CR_{on}	39	39	29	38	36	38	37	35	40	36	32	37	36	40	39	39
OG_{on}	G_{10}	G_5	$G_{0,1,10}$	$G_{7,11}$	$G_{0,2,13}$	G_3	$G_{1,2}$	$G_{0,3}$	-	$G_{3,5}$	$G_{1,2,6,9}$	$G_{0,3,7}$	$G_{7,9}$	-	G_{10}	G_{10}

Table 1. Recognition results

Online vs. Offline Test Results Both offline and online recognizers were applied on the same sample sets of gestures G_1 to G_{12}. The tested sample gestures were recognized with an accuracy of 94% and 91% with offline and online recognizers, respectively. However, the average computation time of the online recognizer was 50 *ms* in average, which was up to 25 times faster than the offline gesture recognizer on the same machine. The higher recognition rate in offline recognizer is due to the fact that it creates the complete hypotheses tree before likely tag tracks are chosen (Section 3.2). However, the offline gesture recognition technique is unable of real-time gesture recognition.

Double Gestures Test Results Since the tag IDs are transferred to the RFID readers at detection, tracking of several tags can be done independently and hence simultaneously. To demonstrate that we can get comparable results when more gestures are to be recognized, two users performed two simultaneous gestures. An alphabet of eight double gestures, G_{17} to G_{24}, as shown in Figure 7, were tested. A total of 160 double samples were collected: 20 samples of each double gesture G_{17} to G_{24}. Similar to single gestures, all double gestures were performed with an average speed of 0.11 *m/s*. They were also performed in different parts of the monitored area and with different relative distance to each other. The average rate of correctly recognized gestures of all 160 double gestures (20 sample of each) using the offline gesture recognition technique was 91%. The slight decrease in the recognition rate is because of the increased tag interference.

5 Discussion and Conclusion

In this paper, we presented the design and evaluation of a real-time hand gesture recognition technique based on RFID, which can be used to develop intuitive interfaces for pervasive applications such as interactive advertisements. We proposed the use of multiple hypotheses tracking to track the motion patterns of passive RFID tags and hence, the hand gestures. Our online gesture recognition technique was able to recognize gestures in real-time with up to 96% recognition accuracy.

Due to the low-cost of passive RFID tags and the fact that they can easily be attached to a user's clothes, we believe that passive RFID technology form a promising solution for unintrusive gesture-based interaction. In an RFID-enabled environment, users can interact with displays without the need for an auxiliary device, such as mobile phones or customized devices. Since RFID antennas can sense the passive tags in their fields up to a few meters, they provide remote user-display interaction, unlike touch-based screens. Furthermore, since a user's identity as well as the data stored in their tags are known to the system, advertisements can be tailored to the informational need of a user who is in the display's vicinity. Moreover, the system can easily distinguish various users and support simultaneous interaction as well as interaction among users.

On the other hand, the positioning accuracy of the proposed technique is limited by the size of the created partitions, and hence, the range and the number of the used antennas. Consequently, while the proposed technique is effective for macro-scale interactions, it might be less useful in applications that require very fine-grained manipulations. However, an RFID-based interaction technique can be combined with other techniques such as accelerometers-based techniques to make the recognition of finer-grained gestures possible. Such combined techniques still have the advantages of RFID-based techniques, in particular personalized services due to the known identity of a user and support of multiple users.

Acknowledgements
This work was partly funded by National ICT Australia (NICTA).

References

1. SkyeTek. http://www.skyetek.com/
2. Asadzadeh, P., Kulik, L., Tanin, E.: Gesture recognition using RFID technology. Personal and Ubiquitous Computing 16, 225–234 (2012)
3. Blackman, S.: Multiple hypothesis tracking for multiple target tracking. IEEE Aerospace and Electronic Systems Magazine 19, 5–18 (2004)
4. Bouet, M., Pujolle, G.: 3-D localization schemes of RFID tags with static and mobile readers. In: International networking conference on adhoc and sensor networks, wireless networks, next generation internet. pp. 112–123 (2008)
5. Bouet, M., dos Santos, A.L.: RFID tags: Positioning principles and localization techniques. Wireless Days pp. 1–5 (2008)
6. Broll, G., Reithmeier, W., Holleis, P., Wagner, M.: Design and evaluation of techniques for mobile interaction with dynamic NFC-displays. In: International conference on Tangible, embedded, and embodied interaction. pp. 205–212 (2011)
7. Chawla, K., Robins, G., Zhang, L.: Object localization using RFID. In: IEEE International Symposium on Wireless Pervasive Computing (ISWPC). pp. 301 – 306 (2010)
8. Hardy, R., Rukzio, E.: Touch & interact: touch-based interaction of mobile phones with displays. In: MobileHCI. pp. 245–254 (2008)
9. Hinske, S.: Determining the position and orientation of multi-tagged objects using RFID technology. In: PerComW. pp. 377–381 (2007)
10. Kim, J., He, J., Lyons, K., Starner, T.: The gesture watch: A wireless contact-free gesture based wrist interface. In: ISWC. pp. 15–22 (2007)
11. Kruskal, J.: An overview of sequence comparison: Time warps, string edits, and macromolecules. SIAM review pp. 201–237 (1983)
12. Liu, X., Corner, M.D., Shenoy, P.: Ferret: RFID localization for pervasive multimedia. In: International Conference on Ubiquitous Computing. pp. 422–440 (2006)
13. Murthy, G., Jadon, R.: A review of vision based hand gestures recognition. International Journal of Information Technology and Knowledge Management 2, 405–410 (2009)
14. Nikitin, P.V., Martinez, R., Ramamurthy, S., Leland, H., Spiess, G., Rao, K.V.S.: Phase based spatial identification of UHF RFID tags. IEEE RFID pp. 102–109 (2010)
15. Rahman, A.M., Hossain, M., Parra, J., Saddik, A.E.: Motion-path based gesture interaction with smart home services. In: MM. pp. 761–764 (2009)
16. Rekimoto, J.: Gesturewrist and gesturepad: Unobtrusive wearable interaction devices. In: ISWC. pp. 21–27 (2001)
17. Starner, T., Auxier, J., Ashbrook, D., Gandy, M.: The gesture pendant: A self-illuminating, wearable, infrared computer vision system for home automation control and medical monitoring. In: ISWC. pp. 87–94 (2000)
18. Vlaming, L., Smit, J., Isenberg, T.: Presenting using two-handed interaction in open space. In: TABLETOP. pp. 29–32 (2008)
19. Want, R.: An Introduction to RFID Technology. Pervasive Computing 5, 25–33 (2006)
20. Wilson, P., Prashanth, D., Aghajan, H.: Utilizing RFID signaling scheme for localization of stationary objects and speed estimation of mobile objects. In: IEEE International Conference on RFID. pp. 94–99 (2007)
21. Zhang, Y., Amin, M.G., Kaushik, S.: Localization and tracking of passive RFID tags based on direction estimation. International Journal of Antennas and Propagation 2007, 1–9 (2007)