Sept. 2014. Vol. 5. No. 04 ISSN2305-8269
International Journal of Engineering and Applied Sciences

www.eaas-journal.org

IMAGE-BASED CLUSTERING ALGORITHM IN HIGH PRECISION REAL TIME LOCATION SYSTEM

[1]JONG-RYUL YANG, [2]MOON-KYOUNG KANG, [3]JIMYUNG KANG

[1, 3]Senior Researcher, Korea Electrotechnology Research Institute, Ansan-city, South Korea
[2]Researcher, Korea Electrotechnology Research Institute, Ansan-city, South Korea
E-mail: [1]net10226@keri.re.kr, [2]mkkang@keri.re.kr, [2]jmkang@keri.re.kr

ABSTRACT

An image-based clustering algorithm is presented for high-precision real time location system (RTLS). In RTLS, it is necessary to make each cluster consisting of two or more objects when these objects are close enough to meet a specific distance threshold for the minimum period of time. These clusters can show the user of the specific object. The proposed algorithm is based on comparisons between the images generated from the tracking data in a pre-defined area. The processing time in the RTLS using the proposed algorithm can decrease with the almost same detection performance, compared with the previous distance-based algorithm.

Keywords: ***Clustering, Image-based Algorithm, Impulse Radio Ultra-wideband (IR-UWB), Real Time Location System (RTLS), Tracking Objects***

1. INTRODUCTION

Real time location system (RTLS) provides positioning, tracking, and monitoring services for human beings or objects in real time [1]. RTLS is classified as both indoor and outdoor system depending on its application. The most promising indoor RTLS service is hospital resource management. RTLS for the medical application can be applied to user detection, medical device, equipment and medicine usage monitor, and patient position tracking [2]. The technology based on impulse radio UWB (IR-UWB) signals is the most suitable solution because it can differentiate objects with high resolution even if the distance is less than several-centimeter. It operates many tags attached to objects simultaneously [3].

RTLS can show the object user information by clustering the user and the object. The cluster can be determined in RTLS when two or more objects are positioned and tracked together with less than a specific distance threshold value determined by the minimum distance resolution of the system, for a period of time. The general algorithm for clustering uses a single or multiple data file, including all locations of the objects in the system during the operating time. However, there is a limited scope to shorten the processing time because all the locations are loaded on the same system and each position is compared with other positions step by step. The other algorithms in [4, 5] use a database (DB) for RTLS and reduce the processing time by loading the selective locations in chronological order. Unfortunately, they require large computing powers for positioning comparison due to large location data files in the algorithm. The cost and space also increase dramatically due to the DB usage.

In this paper, we present an effective cluster algorithm among objects in RTLS tracking the objects locations by comparing images. The processing time decreases because data is compared all at once. In addition, the location data size does not affect the processing time. The performance of the algorithm is demonstrated in the RTLS based on IR-UWB signals.

2. IMAGE-BASED CLUSTERING ALGORITHM

The proposed algorithm is based on image processing. Image files are generated from the object location data moving for a period of the time

Sept. 2014. Vol. 5. No. 04 ISSN2305-8269
International Journal of Engineering and Applied Sciences

www.eaas-journal.org

defined as the unit time t_{unit}. The object location data in the fixed area is expressed in a single image with tracking time. The image in the proposed algorithm shows the object traces, and it is individually generated as the number of objects and t_{unit}. The algorithm selects and clusters two files generated from different objects at the same time. The clustering degree is presented to two parameters, which are *Image Similarity* S_{img} and *Image Dissimilarity* D_{img} respectively and can be expressed as

$$S_{img}(A,B)[\%] = \frac{A \cap B}{\max(A,B)} \times 100 \qquad (1)$$

$$D_{img}(A,B)[\%] = 1 \quad S_{img}(A,B) \qquad (2)$$

where *A* and *B* are matrixes of binary image files, *A*∩*B* is a function of intersection operation, and *max(A, B)* returns the larger number comparing the number of binary 1 in *A* and *B*. Binary 1 in the matrix means the object location in the area. S_{img} can be briefly expressed as the pixel number ratio with the same value to the overall number of pixels.

The algorithm source code is shown in Fig. 1. It starts with the following input parameters: *Date*, *Time*, and *Reference object*. *Date* and *Time* are start and end time to check whether clusters are formed or not. *Reference object* is optional when we want to know which objects are placed around the special object. Using *Reference object* in the algorithm, the processing time decreases with the fixed object in the clustering process.

The next step is to select image files which agree with initial parameters. Then, the selected files make the blurred images using *Filter Pixel* (*FP*). The blurred images are determined by dividing *FP* into the number of *pixels per distance* (*PPD*). *FP* presents the matrix size in the blur method [6]. The blur method is necessary to reduce data errors generated by converting the location data to the image. The errors are caused by the different scales between the image with discrete pixel points and the location data with high range resolution. In the next procedure, the algorithm computes S_{img} compared with *the threshold similarity* S_{TH}. When S_{img} is larger than S_{TH}, the algorithm stores the clustering time rate in t_{unit}. The final step is to determine whether two blurred images *A'* and *B'* are a cluster or not by using the minimum time period T_{TH}. The clustering time rate should be longer than T_{TH}. These steps are repeated for the number of objects, *Date*, *Time*, and *Reference object*.

In the algorithms in [4, 5], S_{img} is calculated using the raw location data of each object for all times,

```
Clustering(Date, Time, Reference_object) {
    for( All images of all objects) {
        A is selected image file;
        for(All images of all objects) {
            B is selected image file;
            if(A and B are agree with Date, Time and
    Reference_object) {
                [A',B'] = blur method(FP);
                Similarity = S_img(A',B');
                if(Similarity > S_TH) {
                    Time = Time + Similarity × t_unit;
                }
            }
        }
        if (Time > T_TH) {
            Cluster between A and B;
        }
        else {
            No Cluster between A and B;
        }
    }
}
```

Figure 1 : Source code of the proposed image-based clustering algorithm

and the amount of data in the sequence should be the same among the data. When data is lost, an interpolation process is essentially added within the procedure of the algorithms [7]. The additional process is not mandatory in the proposed algorithm because the overall resemblance between images is used with negligible location data loss.

3. EVALUATION OF THE ALGORITHM IN RTLS

Two different experiments were performed to evaluate the proposed algorithm. The first one is to compare the clustering results of the proposed algorithm with those of the reference algorithms in [4, 5]. The second one is to compare the algorithm processing time with those of the reference algorithms based on the size of t_{unit}. The image files used in the experiments are obtained from raw location data using the RTLS based on IR-UWB signals. The details of system configurations and specifications of the RTLS are shown in [8]. Two moving objects are considered in the experiments. These objects are moved within the 20 m × 10 m area. Locations are estimated using the RTLS with a 10 Hz update rate. The RTLS clustering server consists of an Intel core i5 processor with a speed of 3.3-GHz, 8 GB of RAM, and Windows 7 64-bit OS. The range resolution of the RTLS is 30 cm, as determined by the bandwidth of the transmitted signal [8]. The file image resolution is set to 200 × 100 pixels with each pixel size representing 1-cm.

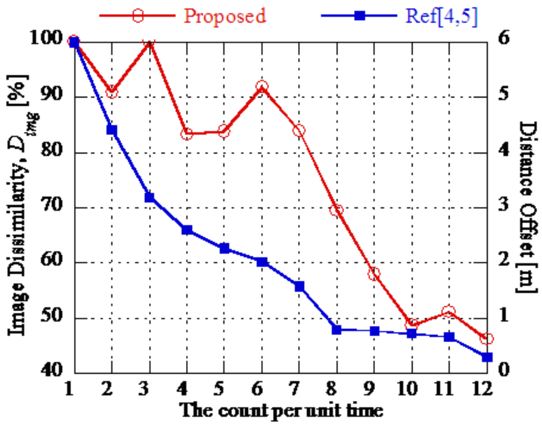

Figure 2 : Image Dissimilarity (D_{img}) and distance difference depending on the count per unit time. The correlation coefficient is 0.82.

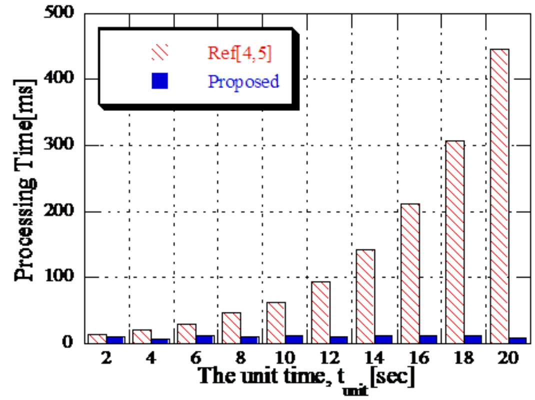

Figure 3 : Processing time depending on times of the unit time.

Figure 2 shows a comparison of *Image Dissimilarity* D_{img} with an average distance difference between two objects by using the reference algorithms. The abscissa axis of Fig. 2 is the count per t_{unit}, which is 5 sec, and the overall time is 60 sec. The interrelationship between two different graphs can be described by the correlation coefficient or the correct cluster rate (CCR) [9]. CCR is 100% when the two data are exactly the same. CCR between two graphs shown in Fig. 2 is 82%. It means that D_{img} from the proposed algorithm is related to distance difference.

It is assumed that the maximum distance difference among objects is 1-m in the room when several objects are moved together or the person brings some objects. For that reason, the threshold distance for clustering objects is set to 1-m in the RTLS. With the threshold distance, D_{img} in the proposed algorithm should be selected from 70% to 80% as shown in Fig. 2. D_{img} can be selected fewer than 70% when the threshold distance is set to less than 0.8 m, but the distance difference is negligible in the RTLS because of the range resolution of the RTLS and the image resolution of the image files.

It is essential to decrease the processing time in the algorithm of the RTLS because there are many objects in the system operating environment. Figure 3 shows the processing time of each algorithm. While the time of the reference algorithms abruptly increases depending on the size of t_{unit}, the time of the proposed algorithm is almost constant. The unit time t_{unit} does not have effects on the processing time of the proposed algorithm because the algorithm is only function of the image resolution and the additional process, such as the interpolation, is not necessary for the algorithm. The results show the proposed algorithm can effectively cluster objects with the appropriate value of D_{img} in the RTLS regardless of the number of objects or the amount of data increase.

4. CONCLUSION

The efficient algorithm is proposed for clustering objects in the IR-UWB RTLS. The algorithm is based on the comparison method using images of traces, which are generated from the object location for each unit period of time. The proposed algorithm performance is demonstrated in the RTLS such that the processing time can be decreased with the same performance of the existing well-known clustering algorithm.

REFRENCES

[1] Y. Gu, A. Lo, and I. Niemgeers, “A survey of indoor positioning system for wireless personal networks”, *IEEE Communication Surveys Tutorials*, vol. 11, pp. 13-32, 2009.

[2] D. Mohapatra, and S. B. Suma, “Survey of location based wireless services”, *Proc. Int. Conf. Personal Wireless Communication*, pp. 358-362, 2005.

[3] S. Gezici, T. Zhi, G. B. Giannakis, H. Kobayashi, A. F. Molisch, H. V. Poor, and Z. Sahinglu, “Localization via ultra-wideband radios: a look at positioning aspects for future sensor networks”, *IEEE Signal Processing Magazine*, vol. 22, pp. 70-84, 2005.

[4] Z. Fu, W. Hu, and T. Tan, “Similarity based vehicle trajectory clustering and anomaly detection”, *Proc. Int. Conf. Image Processing*, vol. 2, pp. 602-605, 2005.

[5] F. I. Bashir, A. A. Khokhar, and D. Schonfeld, “Segmented trajectory based indexing and retrieval of video data”, *Proc. Int. Conf. Image Process*, vol. 2, pp. 623-626, 2003.

ISSN2305-8269
International Journal of Engineering and Applied Sciences

www.eaas-journal.org

[6] R. Crane, "A Simplified Approach to Image Processing: Classical and Modern Techniques in C", *Prentice Hall*, 1996.

[7] R. W. Schafer, and L. R. Rabiner, "A digital signal processing approach to interpolation", *Proc. IEEE*, vol. 61, pp. 692-702, 1973.

[8] J. Kang, M.-K. Kang, S. Lee, Y.-J. Park, and K. Kim, "Characteristics of the Threshold-based IR-UWB Positioning System", *Proc. IEEE 72nd Vehicular Technology Conf. Fall*, pp. 1-5, 2010.

[9] R. A. Fisher, "Frequency distribution of the values of the correlation coefficient in samples from an indefinitely large population", *Biometrika*, vol. 10, pp. 507-521, 1915.