

# Engineering Realization Algorithm for Multi-target Identity Recognition

Xuecheng ZHANG<sup>1,\*</sup>, Li JIAN<sup>2</sup> and Jingsheng HU<sup>1</sup>

<sup>1</sup>Military Representative Office in Shanghai; Shanghai, 201109, China

<sup>2</sup>Shanghai Electro-Mechanical Engineering Institute; Shanghai, 201109, China

xc-zhang@outlook.com

**Keywords:** Multiple targets, Identity recognition, Similarity threshold.

**Abstract.** The simulation training is popular for the low cost and high effectiveness. Aerial targets are simulated instead of using real aircraft in a missile simulation training system. The multi-target identity recognition should be done to ensure that the detected target by the search radar simulator is the same one generated by the simulation target source module. The position and speed of two types of targets are compared to calculate the deviations. Then the similarity threshold is used to judge the tracks of the same target. Simulation results indicate that the proposed algorithm could satisfy the requirements of engineering application.

## Introduction

Today's weapon systems become much more complex and expensive with the highly technical advances and modern warfare requirements. They require operators to spend much more training time on mastering the knowledge, skills, and expertise. The simulation training provides an efficient way of promoting operators skills and proficiency at a lower cost. US army views the simulation training as the main training mode in the 21st century, which continues to expand around the world because of the advantages of low-cost and high-effectiveness [1]. An air defense missile simulation training system is built to train initial neophytes efficiently in the air defense warfare without using real aircraft and missile. Aerial targets are generated by the simulation target source module in the system. The search radar simulator begins to locate aerial targets and form their tracks when it receives these simulated points. In order to judge whether or not the detected target is the same as the simulated one, the evaluation module should recognise the identities of many targets from the simulated target source and the search radar simulator.

The paper presents an engineering realization algorithm for the multi-target identity recognition. The sequential estimation of similarity is used to recognise identities of multiple targets from the two simulation sources. It is used to construct the simulated aerial targets for training experience. Simulation results show that the proposed algorithm has a relatively higher degree of real-time and accuracy. Therefore, implementation of the proposed multi-target identity recognition algorithm could provide aerial targets for the missile simulation training without using real aircraft or missile.

## Design of the Recognition Algorithm

### Recognition principles

The goal of multi-target identity recognition is to determine whether or not the target tracked by the search radar simulator is the same as the one sent from the simulation target source module. Therefore, the similarities of flight speed, altitude, distance, azimuth are used to characterize the identities of many targets from the two generating sources. First, the deviations of targets from the two sources are calculated. Second, the deviations are sequenced. Third, the similarity of two targets with the minimum deviation adds an incremental count while the others cut down a count. Finally, the two targets could be regarded as the same one if their similarity reaches the settled threshold value.

## Recognition procedure

Fig.1 shows the procedure of the proposed multi-target identity recognition algorithm. The recognition procedure has seven steps as following:

Step 1 Obtaining the target data information from sensor 1. The data information includes target lot number  $i$ , position, speed and time message. The target data are put into structured data array and updated at regular intervals. The total number of targets from sensor 1 should not exceed  $m$ .

Step 2 Obtaining the target data information from the other sensor 2. The target information also includes target lot number  $j$ , position, speed and time message. And the total number of targets from sensor 2 should not exceed  $n$ .

Step 3 Aligning time-base and transforming coordinate. The  $j$ th target is first aligned to the same time-base as the  $i$ th target. Then the coordinate of the  $j$ th target is transformed based on the same observation point. The deviation array of speed and position between the  $j$ th target and the  $i$ th target could be calculated. The deviation is regarded as the identity value.

Step 4 Calculating identities of the  $j$ th target with the total  $m$  targets from sensor 1.

Step 5 Sequencing identities and selecting the target  $i_1$  with the minimum deviation.

Step 6 Calculating similarity. The similarity of the  $j$ th target and target  $i_1$  adds an incremental count while the others cut down a count till zero.

Step 7 Evaluating identity. The  $j$ th target is as same as target  $i_1$  if their similarity is greater than a threshold value.

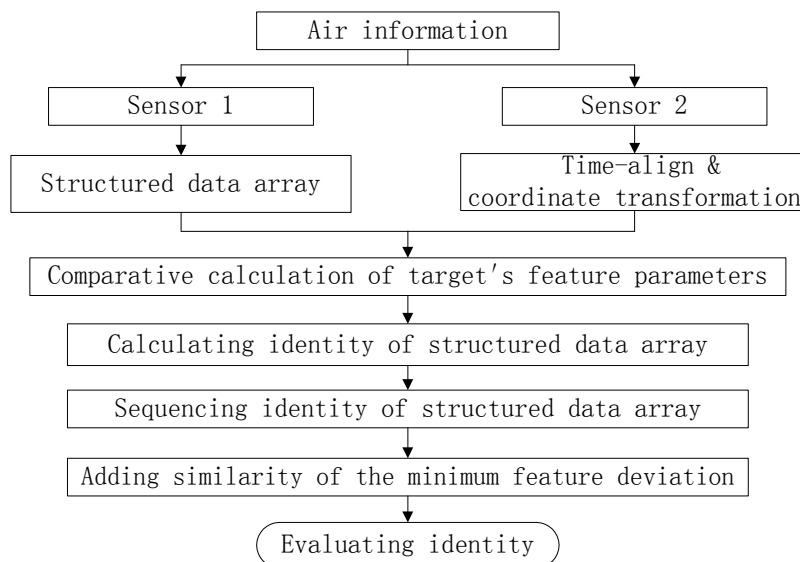


Fig.1 Sketch of multi-target identity recognition procedure

## Realization of the Algorithm

### Executive means

This section will describe the realization of the identity recognition algorithm in detail. The target information is stored in the structured data array. The main characteristics of an aerial target are location and speed [2] because the target is regarded as a particle. Sensor 1 denotes the simulation target source module. Therefore, the aerial target could be determined by six parameters classified as location coordinate  $L_c < x, y, z >$  and velocity  $V < v_x, v_y, v_z >$  in the Cartesian coordinate system. The number of array items equals to the total target number  $m$  from sensor 1. And each item of the structured data array includes target lot number  $i$ , location coordinate  $L_c$ , velocity  $V$ , corresponding time  $T_i$ , the identities  $I_{ji}$  and similarities  $S_{ji}$  compared with  $n$  targets from sensor 2, and the matching target lot number from sensor 2. The structured data array is updated in real time when sensor 1 receives the air information.

Sensor 2 denotes the search radar simulator. It begins to establish target trajectories when it receives the target information from sensor 1. That is, it begins to estimate the intention of target movement and predict the coming position as the real radar would do. Then the data in the radar's coordinate system could be obtained, which include target lot number  $j$ , location information (distance  $R$ , azimuth  $A$  and pitch angle  $E$ ), velocity  $\langle v_x, v_y, v_z \rangle$ , and corresponding time  $T_2$ . In order to calculate the similarity, the location parameters  $\langle R, A, E \rangle$  should be transformed from the polar coordinate system to the Cartesian coordinate system as  $\langle x_{ts}, y_{ts}, z_{ts} \rangle$ . The Cartesian coordinate could be calculated from the radar polar one as follows:

$$\begin{pmatrix} x_{ts} \\ y_{ts} \\ z_{ts} \end{pmatrix} = \begin{pmatrix} R \cos A \cos E \\ R \sin E \\ R \sin A \cos E \end{pmatrix} \quad (1)$$

Sensors have their own coordinate systems to observe targets. In order to calculate the similarity, the compared targets should have the same benchmarks. Therefore, the  $j$ th target information data from sensor 2 are treated using the same reference time as the  $i$ th target from sensor 1. The target's position from sensor 2 has not updated now. But position could be estimated as linear movement because the updating frequency of target information is so high that the error could be ignored. Assuming that sensor 2 updates data at time  $t_k$ , then the target's position from sensor 1 at time  $t_k$  could be predicted according to previous position and the temporal velocity.

$$\begin{pmatrix} x(t_k) \\ y(t_k) \\ z(t_k) \end{pmatrix} = \begin{pmatrix} x(t_{k-1}) + v_x(t_k - t_{k-1}) \\ y(t_{k-1}) + v_y(t_k - t_{k-1}) \\ z(t_{k-1}) + v_z(t_k - t_{k-1}) \end{pmatrix} \quad (2)$$

And the earth-centered earth-fixed (ECEF) coordinate system is chosen to as the common coordinate system for coordinate transformation of two targets. The sensor's position is known and usually written as  $\langle \Phi, \Theta, H \rangle$  for latitude, longitude and height respectively. Then the sensor's coordinate  $\langle x_{se}, y_{se}, z_{se} \rangle$  in the ECEF system could be calculated [3] from the following equation:

$$\begin{pmatrix} x_{se} \\ y_{se} \\ z_{se} \end{pmatrix} = \begin{pmatrix} (H + R_e) \cos \Phi \cos \Theta \\ (H + R_e) \cos \Phi \sin \Theta \\ [H + (1 - e^2) R_e] \sin \Phi \end{pmatrix} \quad (3)$$

Where  $e$  is the first eccentricity of ellipsoid ( $e=0.08181919$ ) and  $R_e$  is the prime vertical radius of curvature ( $R_e=6,378,137.0/(1-e^2 \sin^2 \Phi)^{1/2}$ ).

The target position in the sensor's Cartesian coordinate system could be transformed into the ECEF one. Thus the two types of targets are in the same coordinate system through this orthogonal transformation. Let  $\langle x_{ts}, y_{ts}, z_{ts} \rangle$  be the target Cartesian coordinate obtained from eq.1, the target coordinate in the ECEF coordinate system could be given by [4]:

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} x_{se} \\ y_{se} \\ z_{se} \end{pmatrix} + \begin{pmatrix} -\sin \Phi & -\cos \Phi \sin \Theta & \cos \Phi \cos \Theta \\ \cos \Phi & -\sin \Phi \sin \Theta & \sin \Phi \cos \Theta \\ 0 & \cos \Theta & \sin \Theta \end{pmatrix} \times \begin{pmatrix} x_{ts} \\ y_{ts} \\ z_{ts} \end{pmatrix} \quad (4)$$

Two targets from different sources could be then compared in the same ECEF coordinate system by synthesizing eq.3 and eq.4. Therefore, the difference of position between the  $j$ th target and the  $i$ th target could be expressed as:

$$\begin{pmatrix} x'_{ji} \\ y'_{ji} \\ z'_{ji} \end{pmatrix} = \begin{pmatrix} x_j \\ y_j \\ z_j \end{pmatrix} - \begin{pmatrix} x_i \\ y_i \\ z_i \end{pmatrix} \quad (5)$$

The calculating method of speed deviation  $\langle v'_x, v'_y, v'_z \rangle$  is similar. Six parameters' deviations of position and speed are thus obtained. The identity  $I_{ji}$  of the  $j$ th target and  $i$ th target is consequentially determined by summing the deviations' absolute value:

$$I_{ji} = |x'_{ji}| + |y'_{ji}| + |z'_{ji}| + |v'_{xji}| + |v'_{yji}| + |v'_{zji}| \quad (6)$$

The sensor 1 generates  $m$  aerial targets. And  $m$  identity values could be calculated sequentially. Then a search is executed over all the  $m$  values to find the minimum of identity .

$$\Delta I_j = \min\{I_{ji}, i=1, 2, \dots, m\} \quad (7)$$

Then the corresponding target lot number  $i1$  with the minimum identity value could be found. And the similarity of the  $j$ th target with  $m$  targets is made by:

$$S_{ji} = \begin{cases} S_{ji} + 1 & \text{if } i = i1 \\ S_{ji} - 1 & \text{others} \end{cases} \quad (8)$$

It is noted that the similarity in eq.8 has minimum limit zero. The track information of the  $j$ th target is compared with the  $m$  targets from sensor 1 one by one. The  $j$ th target and the  $i$ th target are regarded as the same target when their similarity value reaches the settled maximum limit. The data from sensor 1 should be updated more frequently than those from sensor 2. All the targets could be recognised through  $n$  traverses by using the above procedures.

### Programming code

The algorithm is programmed by VC++ language running with the Windows XP operating system. The structured data array for the judging matrix is formed. The array is designed as

```
structure S_target_judge
{
    BYTE count[255], fixpi;
    double x, y, z, vx, vy, vz, absadd, r, a, e;
};
S_target_judge s_judge[255];
```

The program compares one target from sensor 2 with those  $m$  targets from sensor 1. A processing calculation includes  $n$  time-aligns,  $n$  coordinate transformings,  $n$  identities and  $n$  sequences. To save space, a part of the code for identity recognition is shown below.

```
for (int j=0; j<maxpi; j++) //traversing all the targets from sensor 2
{
    double addtmp=0;
    double dx=fabs(x-s_judge[j].x); // position at x axis
    ...
    double dvx=fabs(vx-s_judge[j].vx); //velocity at x axis
    ...
    addtmp=dx+dy+dz+dvx+dvy+dvz;//the identity value of the jth target
    s_judge[j].absadd=addtmp;
    if (minaddpos==254)
    {
        minaddpos=j;
    }else
```

```
    if (s_judge[minaddpos].absadd>s_judge[j].absadd)
        minaddpos=j;
}
```

### **Application result**

The proposed method is used in a missile training system. The search radar simulator tracks the simulated targets from the simulation target source module. The types of targets include slow, fast, intersection and apart from each other. The identity recognition result could be obtained after three sampling cycle times for most cases. The data comparison proves that the proposed algorithm to be effective in the multi-target identity recognition.

### **Summary**

Deviations of position and speed indicate the difference of two targets. Therefore, this proposed algorithm adopts similarity threshold to achieve the identity of the same target from the search radar simulator and the simulation target source module. It has low computational complexity with the stable and accurate output results. It has been used in a certain missile simulation training system to provide simulated aerial targets. Simulation experiences show that this algorithm is effective for the engineering application. There are somethings to enhance the algorithm in the future, for example, the dimension of the attitude could be added to increase the target's authenticity. And more examines should be conducted to evaluate the resultant effect on the performance in the missile simulation training system.

### **Acknowledgement**

This research was financially supported by the GAD key program (Project Number:ZLA14065).

### **References**

- [1] J. E. Summers, Simulation-based military training: an engineering approach to better addressing competing environmental, fiscal, and security concerns. *J. Wash. Acad. Sci.* 98(2012)9-29.
- [2] Z.J. Hu, H. Leung, Statistical performance analysis of track initiation techniques. *IEEE Trans. Sign. Proc.* 45(1997)445-456.
- [3] J. Zhu, Conversion of Earth-centered Earth-fixed coordinates to geodetic coordinates. *IEEE Trans. Aer. Elec. Syst.* 30(1994)957-961.
- [4] Yakov D. Shirman, *Computer Simulation of Aerial Target Radar Scattering, Recognition, Detection, and Tracking*, Artech House, 2002.
- [5] H.J. Liu, L.N. Wang, Z. Li. Study on target location and measurement data association of passive detection system based on direction of arrival, *Comp. Eng.&Des.* 28(2007)657-660.
- [6] V.G. Keuk, Sequential track extraction. *IEEE Trans. Aer. Elec. Syst.* 34(1998)1135-1148.