MULTI RADAR DATA FUSION

Submitted by:

Waqar Khalid
01-241172-032

Master in Software Engineering

Department of Software Engineering
Bahria University Islamabad, Pakistan

MULTI RADAR DATA FUSION

Submitted by:

Waqar Khalid
01-241172-032

Supervised by:

Dr. Ahmad Ali

Master of Science (Software Engineering)

A thesis submitted in partial fulfillment of the requirements for the degree of Master of Science (Software Engineering) at Bahria University Islamabad

Department of Software Engineering
Bahria University Islamabad, Pakistan

Author’s Declaration

I, Waqar Khalid hereby declare that the contents of my thesis “**Multi Radar Data Fusion**” is based on my own research and it has not been previously published or presented anywhere in partial fulfillment of the degree or diploma in the world. If my statement is found to be incorrect, the University has the right to cancel my MS degree.

(Student Signature)

Abstract

The objective of our research is to design a multi radar data fusion algorithm which accepts the input from multiple disparate radars and fuse them to generate a precise and reliable picture for air traffic controllers. Data fusion is a technique used for efficiently combining the relevant information coming from multiple sources. The purpose is to achieve estimates and assure better results than those attainable by depending on a single source.

Multi radar data fusion can be separated into two sub-problems: 1) State estimation 2) Data Association. Our proposed algorithm efficiently uses gating and data association techniques to solve the multi radar data fusion problem. Uninterrupted updates of the radars are the most crucial part of the multi radar data fusion. Therefore, Kalman filter is used for the prediction of the target if updates are not received from the radars. When the number of radar increases, the amount of information and data association complexity also increases. This results in confusion, because single target is observed multiple times. This phenomenon is known as clutter. Air traffic controlling is a critical job. Hence, if clutter is observed, it affects the controlling and decision making process for the operator.

The proposed algorithm is evaluated by using real world data of domestic and international flights. Different operational scenarios are discussed with air traffic controllers and three live scenarios are selected for the testing of the algorithm. The results of the proposed method are compared with the data of Automatic Dependent Surveillance Broadcast (ADS–B). It is a modern tracking system in which aircraft uses satellite navigation system to decide its current position and regularly transmits to its ground bases controlling station. The air traffic controller uses ADS-B receivers as a replacement of the secondary radars because no interrogation signal is needed from the ground.

The results of the proposed algorithm are compared with the ADS-B data as a true reference because of its precision and reliability.

Acknowledgements

All the admirations to Allah, Who is Rehman and Raheem, the most beneficent, the most merciful, as He enabled me to continue this research and fulfill the requirements of my degree.

First of all I want to thank my supervisor Dr. Ahmed Ali for his valuable time, support and motivation towards my thesis. Whenever I needed his advice he always appreciated and welcomed my ideas. His presence and optimism have provided an invaluable influence on my career and outlook for the future. I consider it my good fortune to have an opportunity to work with such a wonderful person.

I also want to thank all the faculty members and staff of the Department of Software engineering department in Bahria University, Islamabad who have encouraged me throughout the degree.

I would like to thank all my friends and especially my classmates for all the solicitous and mind motivating discussions we had. I have enjoyed their company so much during my stay at Bahria University Islamabad.

Here I would also want to thank my family members who always prayed and motivated me for my success. Above all, I am very grateful to Allah Almighty for helping me in every aspect of this thesis and paving my way forward.

This thesis is dedicated to my family and supervisor

Table of Contents

Abstract .... **iv**

Acknowledgements .... **v**

Table of Contents .... **vii**

List of Tables .... **ix**

List of Figures .... **x**

List of Abbreviations .... **xi**

1 Introduction **1**

1.1 Motivation 2

1.1.1 Ocean Surveillance Systems 2

1.1.2 Air-to-Air and Surface-to-Air Defense 2

1.1.3 Battlefield intelligence 3

1.1.4 Threat Assessment 3

1.2 Challenges in Multi Radar Data Fusion 3

1.2.1 Conflicting Output 3

1.2.2 Features Processing Framework 3

1.2.3 Cost Efficiency 4

1.2.4 Computational Power 4

1.3 Problem Statement 4

1.4 Research Objective 5

1.5 Research Methodology 6

1.5.1 Literature Review 6

1.5.2 Proposed Algorithm 6

1.5.3 Results of Proposed Algorithm 6

1.5.4 Discussions and Conclusions 6

1.6 Contribution 6

1.7 Document Organization 7

2 Literature Review **8**

2.1 Data Fusion 8

2.2 Target Tracking 8

2.3 Estimation of Target 9

2.3.1 Kalman Filter 9
2.3.2 Interactive Multiple Model 12
2.3.3 Particle Filters 12
2.4 Gating 13
2.5 Data association algorithms 14
2.6 Data Fusion 18
2.6.1 Data level 18
2.6.2 Feature level 19
2.6.3 Decision level 20
3 Proposed Algorithm 22
3.1 Description of Proposed Algorithm 22
3.1.1 Track Management 23
3.1.2 Track Prediction 24
3.1.3 Gating and Association 24
3.1.4 Fusion 27
4 Results 29
4.1 Data Set 29
4.2 Scenario-1 30
4.3 Scenario-2 32
4.4 Scenario-3 34
4.5 Quantitative Analysis 36
5 Discussion and Conclusions 38
5.1 Hardware and Software Requirements 39
5.2 Future Work 40
References 41

List of Tables

Table 4.1: Parameters Summarized of Live Data Set 29

Table 4.2: Quantitative Analysis Summary for Scenario1, 2 and 3.............................37

Table 5.1: Summarized Requirements of the Software 39

Table 5.2: Summarized Requirements of the Hardware..40

Table 5.3: Fusion Processing Time for Scenario 1, 2, 3 40

List of Figures

Figure 1.1: Tactical Picture of Multi Radar Data Fusion..5

Figure 1.2: Objective of Multi Radar data Fusion 5

Figure 2.1: Perception of Multi Radar Data Fusion..8

Figure 2.2: A general Target Tracking Diagram 9

Figure 2.3: A Gating Example, of Elliptic Gates...14

Figure 2.4: Examples of nearest neighbor and Global nearest neighbor 15

Figure 2.5: Data Level Fusion Data block Diagram..19

Figure 2.6: Feature Level Fusion Data block Diagram...20

Figure 2.7: Decision Level Fusion Data block Diagram 21

Figure 3.1: Proposed Multi Radar Data Fusion Algorithm...22

Figure 3.2: Proposed Fusion Algorithm Flow Diagram 28

Figure 4.1: Radar-A, Live and Predicated Updates of Scenario-1 30

Figure 4.2: Radar-B, Live and Predicated Updates of Scenario-1...............................31

Figure 4.3: Fused Graph of Scenario-1...31

Figure 4.4: Radar-A, Live and Predicated Updates of Scenario-2 32

Figure 4.5: Radar-B, Live and Predicated Updates of Scenario-2...............................33

Figure 4.6: Fused Graph of Scenario-2...33

Figure 4.7: Radar-A, Live and Predicated Updates of Scenario-3 34

Figure 4.8: Radar-B, Live and Predicated Updates of Scenario-3...............................35

Figure 4.9: Radar-C, Live and Predicated Updates of Scenario-3...............................35

Figure 4.10: Fused Graph of Scenario-3...36

List of Abbreviations

MSDF	Multi Radar Data Fusion
ATC	Air Traffic Controller
TA	Threat Assessment
FLIR	Forward-Looking Infrared
DGPS	Differential Global Positioning System
ADS-B	Automatic Dependent Surveillance-Broadcast
GNSS	Global Navigation Satellite System
PF	Particle Filter
EKF	Extended Kalman Filter
UKF	Unscented Kalman Filter
GPS	Global Positioning System
KF	Kalman Filter
AEW	Airborne Early Warning
HMM	Hidden Markov Model
SAR	Synthetic Aperture Radar
ELINT	Electronic Intelligence
ESM	Electronic Support Measures
IFF	Information Foe And Friend
SVF	State Vector Fusion
MS	Measurement Fusion
GF	Gain Fusion
MSE	Mean Square Error
IMM	Interactive Multiple Model
MLAT	Multilateration
WAM	Wide-Area Multilateration
PDA	Probabilistic Data Association
SOP	Standard Operating Procedures

Chapter-1

Introduction

The first introductory chapter presents the contextual background of the problem, motivation towards the topic along with framing the problem description and the purpose of this thesis. Besides, the chapter also explains the merits and de- merits of the multi radar data fusion.

With the passage of time, human and animals learnt the art of survival by using their nervous system. It consists of five God gifted sensors, like eyes are for sight, touch is to feel, hear is for audition, smell is for olfaction and taste is for gustatory. Human and animals use the combination of these sensors to perform any action. For example, to assess the taste of a food, sight, smell and taste are used in combination, similarly hearing and vision is used by the animals to predict any danger. Hence, the multi radar data fusion is efficiently used by the humans and the animals to assess their environment and to forecast any threat. So, multi radar data fusion is not a novel idea, although it’s a natural factor in human and the animals.

With the emergence of new active and passive radar suites, which includes radar, electronic intelligence (ELINT), sonar, infrared, synthetic aperture radar (SAR), artificial intelligence, statistical estimation, digital signal processing, control theory approaches and advanced processing techniques have made the multi radar data fusion possible in real time.

In the recent years, substantial efforts have been made by the researchers to focus on multi-radar data fusion for both civil and military applications. Data fusion is an approach, in which multiple disparate radars integrate their target information to get more accurate and reliable air picture than could be attained through single radar. Multi radar data fusion for military applications includes [1]: air-to-air defense systems, surveillance and target detection system, unmanned systems, battlefield intelligence and sub-marine warfare system. And non-military applications include [1]: weather monitoring system, natural disaster management system and remote sensing. There are multiple diverse approaches available in the literature which has derived from pattern recognition, deep learning, machine learning and statistical estimation.

The determination of this research is to provide a solution for multi radar system for air traffic controllers. When a single physical object is detected by multiple radars, the measures obtained from multiple radars can be combined in order to get high level of accuracy, which is called as multi radar data fusion.

1.1 Motivation

Radars are employed in any surveillance system as the primary mean to monitor the external or internal environment. Single radar employed systems have limited capabilities and shortcomings for providing consistent information about the external environment. Accurate and timely management of information is vital for successful decision-making in any critical system. Air traffic controlling is one of the crucial and important military application in which hundreds of human lives depend upon air traffic controllers. Therefore, any misleading information can put hundreds on human in danger. The multi radar data fusion algorithm integrates the information coming from different radars and fuses them in order to generate precise and reliable air situation picture for air traffic controllers. This helps the operators in making fast and efficient decisions in their daily operations.

The military applications which motivated me for the research of multi radar data fusion are:-

1.1.1 Ocean Surveillance Systems

Ocean surveillance systems are used to detect and monitor ocean based objects. For examples in naval operations, ocean surveillance systems are used to guide sub-marines and underwater unmanned system [2]. Sensing element suites will embody measuring system, sonar, electronic intelligence (ELINT), active and passive radar, synthetic aperture radar and infrared. These sensing systems help the marine force to cover larger areas for performing their surveillance duties efficiently.

1.1.2 Air-to-Air and Surface-to-Air Defense

Air-to-Air and Surface-to-Air defense systems are very critical for the air space management of any country. Air defense systems consist of different active and passive radars includes information about friend and foe (IFF) radars, electronic support measures (ESM), electro optic image radars. Multi radar data fusion is very critical for rapid and effective decision making, route planning, target prioritization and assignment of ground and air targets [1].

1.1.3 Battlefield intelligence

Battlefield intelligence systems are used for the identification of any probable target or point of interest on the ground [1]. For example, embrace the position of the ground forces, the location of their vital points and target identification. Radars includes Passive electronic support measures, airborne surveillance radars, ground based static radars, autonomous air vehicles and infrared radars. Multi radar data fusion is used to provide key information to support threat assessment and situational awareness for the decision makers.

1.1.4 Threat Assessment

Threat assessment (TA) concentrates on the information which helps the decision makers to assess the current situation and take corrective actions [2]. The assessment of the situation on the bases of information gathered from different radars helps to estimate enemy capabilities and provides the opportunity to take corrective actions against enemy.

1.2 Challenges in Multi Radar Data Fusion

1.2.1 Conflicting Output

In a multi radar scenario, similar observations may be understood with entirely different meaning, which leads to a big loss. Especially in critical systems like air traffic controlling system [3]. The probability of false and cluttered amount of data is still comparatively high in modern radars. Consequently, it is indispensable to give a priority to the technology, which is more reliable.

1.2.2 Features Processing Framework

In most of the multi radar environment, data of every radar is evaluated individually, and the results of the evaluation are combined as a last phase in order to get the desired results. However, there are some approaches in which raw data of individual radar is transferred to the central fusion module without any preprocessing. On the other hand, in particular scenarios data can be processed in conjunction and then fused with the data of the other radars. Resultantly, it causes an impediment in the fusion process and consequently increases the processing time.

1.2.3 Cost Efficiency

Multi radar data fusion helps to get better reliability and performance of the detection system. In a multi radar environment, where efficiency and reliability is increased it also increases the budget of the overall system [3]. In any design and development system budget forecasting is a crucial part, which needs to be assessed at its early stage. Therefore, with any flexible configuration of the radar, cost factor should be assessed at its early stage to get desired performance of the system.

1.2.4 Computational Power

Multiple radar results in additional data obtained from various radars, which enhance the overall computing cost of the system [3]. The problem can be solved through the division of data analysis into various phases, including feature extraction, data preprocessing, and filtering of the data. Each specific processing type can be carried out using a separate module with a processing center, where the final decision is made.

1.3 Problem Statement

Air traffic controlling is a crucial and demanding task in the world, because it demands split-second decisions in order to save hundreds of lives. In today's world, there is variety of disparate radar available, which leads to an overwhelming amount of information and due to the availability of this vast information, single target in the air is detected multiple times to users. The graphical representation of the scenario is depicted in the first part of the Figure 1.1 in which each target is detected multiple times. Eventually this increases the workload and reduces the efficiency of the air traffic controller for effective decision making and airspace management.

Multi radar data fusion is a viable mean to increase the capability and performance of the air traffic controller. The purpose of this research is to design a multi radar data fusion algorithm, which integrate the information coming from different radars and fuse them in order to produce clear and accurate air situation picture as depicted in the second part of the Figure 1.1, thereby reducing air traffic controller workload and enables them to take fast and efficient decision making in daily ATC operations [4].

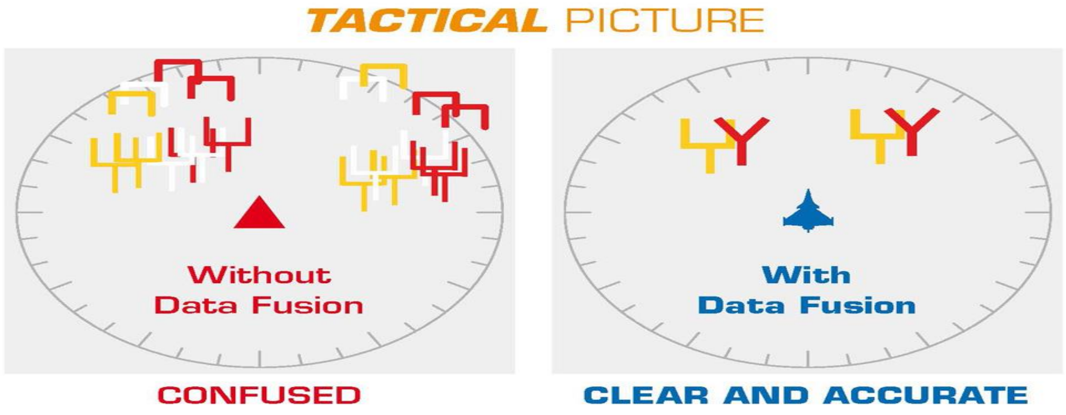

Figure 1.1: Tactical Picture of Multi Radar Data Fusion

1.4 Research Objective

Our research objective is to produce accurate and reliable air picture in the multi radar environment, so that, air traffic controller can take effective, efficient and timely making decision for airspace management. The Figure 1.2 depicts the ultimate goal of our research, in which we can witness that multiple radars have successfully fused the targets.

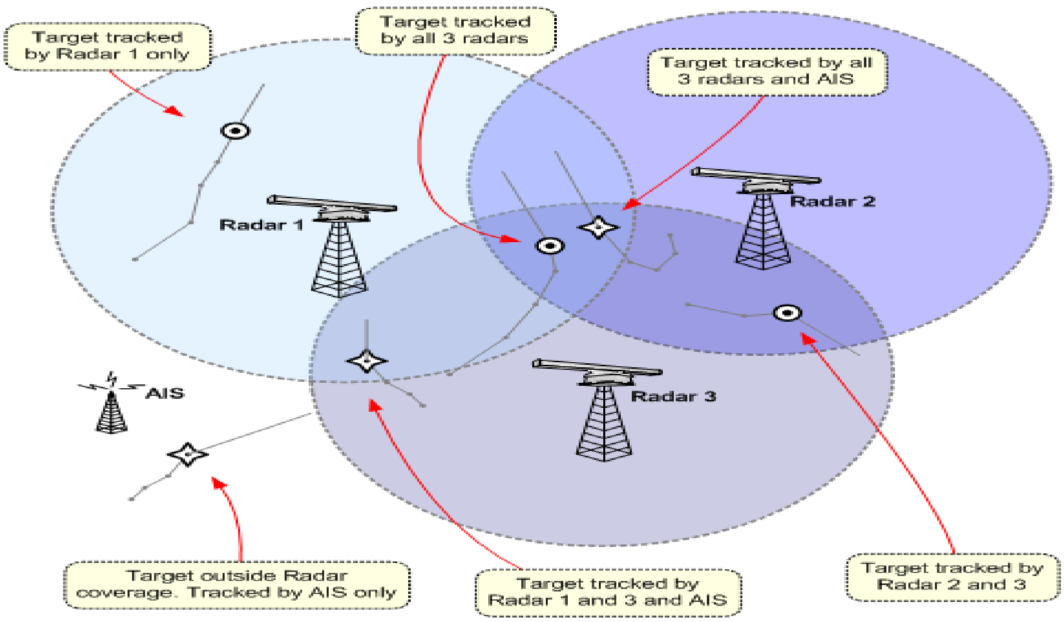

Figure 1.2: Objective of Multi Radar data Fusion

To achieve our research objective, we designed a multi radar data fusion algorithm, which accepts the inputs coming from multiple disparate radar and fuse them to produce precise and clear air picture. Our designed algorithm enhances the capability of the air traffic controller for effective and efficient decision making process.

1.5 Research Methodology

A step-by-step analysis approach followed in the research is mentioned below:

1.5.1 Literature Review

In the first step we go through the available research. The main purpose of the review is to get a deep understanding of the various approaches available in the research. Literature review helps to understand the way researchers solve the problem of Multi radar data fusion.

1.5.2 Proposed Algorithm

In the second step, we formulate the Multi Radar Data Fusion Engine algorithm which combined the input from multiple radars and fuse them into one single target. Our ultimate goal was to provide an effective and efficient solution to air traffic controllers for their accurate and timely decision making and reduce their workload.

1.5.3 Results of Proposed Algorithm

In the third step, the formulated MSDF algorithm is implemented in MATLAB and evaluated on different scenarios. Different scenarios were discussed with air traffic controllers through an interview and these scenarios were evaluated.

Real world data sets of different scenarios were taken, after, a detail discussion with the air traffic controller. The results of the selected scenarios were evaluated with reference to ADS-B data.

1.5.4 Discussions and Conclusions

Finally, a comprehensive discussion about the results attained in the previous chapter is carried out. The conclusions of the algorithm are based on the evaluation of the results through the implementation of the scenarios. Some recommendations are also made for future work.

1.6 Contribution

Our contribution towards this research is mentioned below:

- Designing of multi radar data fusion algorithm.
- Implementation of algorithm in MATLAB.
- Results are evaluated on Real time data of domestic and International flights.

- Real time data of ADS-B is used for the true comparison of the results.

1.7 Document Organization

The structure of the thesis is as follows:

- Chapter 2 provides the detailed overview of the literature.
- Chapter 3 depicts the complete details of the formulated multi radar data fusion algorithm.
- Chapter 4 represents the implementation of the multi radar data fusion algorithm in MATLAB and delivers evaluation of the algorithm and the statistical analysis of results.
- Chapter 5 discusses the discussions and conclusion of the research and details of the contributions.

Chapter-2

Literature Review

This chapter enlightens the background of multi radar data fusion under the real-time restrictions. In the first step general introduction to the data fusion is described. After introduction tracking, association and fusion are presented, followed by a detailed review of different research papers.

2.1 Data Fusion

Data fusion is a process of integrating multiple sources information to generate more accurate, reliable and precise data about a target or an object. MSDF is the synthesis of many traditional fields and new engineering areas and their implementation [5]. A general diagram for multi radar data fusion is shown in Figure 2.1.

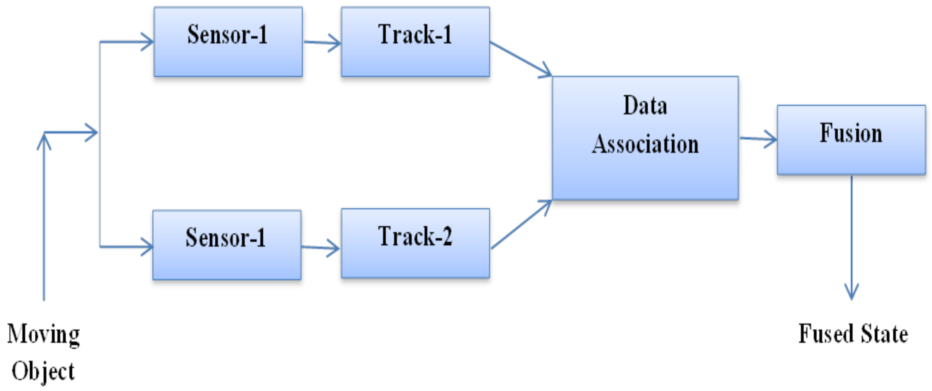

Figure 2.1: Perception of Multi Radar Data Fusion

The data extracted from single radar is not precise and reliable. It takes more time to convert into useful information for the user. When single radar information is compared with multi radar information, we found that multi radar data fusion is more precise, accurate and covers a larger geometrical area [5].

Some fundamental modules which are necessary for multi radar data fusion are discussed in detail in this chapter.

2.2 Target Tracking

The way toward evaluating direction, location and other attributes of one or more objects over time is eluded to target tracking. The concept of target tracking was initially developed in military application where the target items were jet fighters or missiles. Generally, a tracking framework comprises of a set of functions for updating tracked objects, which are largely referred to as targets. The function includes data association, measurement update, track handling and prediction. A general target tracking framework is shown in Figure 2.1. Normally the measurement updates 'Z_k' and the predictions '$\boldsymbol{x}_{k|k}$' as filters in the mention below framework [6]. An algorithm should incorporate the probability that all readings are not from an actual target, but are the result of noisy radars. This noise is often referred to as clutter.

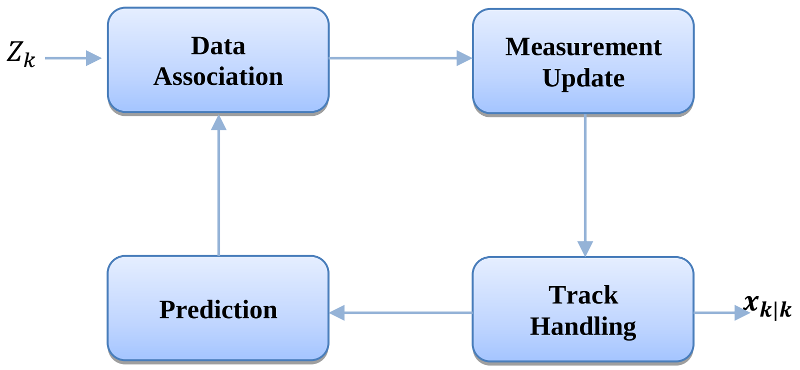

Figure 2.2: A general Target Tracking Diagram

2.3 Estimation of Target

This portion introduces the issue of kinematic state estimation, which typically relates to filtering and predicting an object's position and velocity. Single and multi-radar tracking algorithm depend on the predication at discrete time 'k-1', where 'k' denotes the object next position in time step [7]. Hence we need a sort of filter which can predict the next position of the object by utilizing the prior and current data. The most frequently used filters for the said purpose is Kalman filter. It is a linear estimator which is normally used for the solution of the estimation problems. The Kalman filter reduces the mean squared error if measurement noise and the target dynamics are correctly modeled.

2.3.1 Kalman Filter

The Kalman filter infers that the fundamental dynamics of the target and the measurement procedures can be deduced to be Gaussian. Mean and covariance can entirely parameterize Gaussian distribution, which add to the convenience of the Kalman filter [7]. First, it presumes that the target dynamic procedures are modeled in discrete time as:

$$x_k = Fx_{k-1} + v_{k-1} \quad \textbf{(2.1)}$$

Whereas'x_k' is the n-dimensional state vector at time k that includes the estimated amounts and 'F' is transition matrix and 'v_{k-1}' is zero, which depicts Gaussian process noise with expected identified covariance Q_k. Measurements are in form of linear combinations of the system state variables, but are also corrupted by white noise 'w_k' with covariance 'R_k'[8]:

$$y_k = H_k x_k + W_k \quad \textbf{(2.2)}$$

'y_k'is an M-dimensional measurement vector and H_k is an M×N matrix. As per target model from equation (2.1) and the measurement model from equation (2.2). The iterative process for the calculation of Kalman filter equations are completed in two steps. The first predicated step is.

$$x_{k/k-1} = Fx_{k-1/k-1} \quad \textbf{(2.3)}$$

$$P_{k/k-1} = FP_{k-1/k-1}F^T + Q_k \quad \textbf{(2.4)}$$

The projected state '$X_{k/k-1}$' in equation (2.3) is used as an estimate, where the tracked target must be in time 'k'. '$P_{k/k-1}$' In equation (2.4) provides an appropriate measure of the estimation accuracy. The covariance matrix is additionally utilized in the algorithms of data association discussed in subsequent sections [7]. The performance of the tracking algorithm will decline if the predictions are uncertain. The next step of the iterative algorithm is the measurement update. The projected state is amended with the new measurement 'y_k'.

$$\hat{x}_{k/k-1} = \hat{x}_{k/k-1} + K_k(y_k - \hat{y}_{k/k-1}) \quad \textbf{(2.5)}$$

Where'$\hat{y}_{k/k-1}$' is calculated as

$$\hat{y}_{k/k-1} = H_k \hat{x}_{k/k-1} \quad \textbf{(2.6)}$$

'K_k'Indicates the Kalman gain:

$$K_k = P_{k/k-1} / H_k^T S_{k/k-1}^{-1} \quad \textbf{(2.7)}$$

'S^k'Is the residual covariance matrix:

$$S_{k/k-1} = H_k P_{k/k-1} H_k^T + R \quad \textbf{(2.8)}$$

Lastly, the covariance can be simplified as:

$$P_{k/k-1} = P_{k/k-1} - K_k S_{k/k-1} K_k^T \quad \textbf{(2.9)}$$

The author evaluated the performance of three nonlinear motion targets through fixed gain and Kalman filter for nonlinear motion such as particle filter [9]. The fusion structure of the algorithm is based on the state vector fusion (SVF) and measurement fusion (MS). Author implemented the algorithm in MATLAB and results are evaluated by RMS and RSSP [9]. The simulation result shows that linear rustic filters are the basics for the state estimation to nonlinear filters. Analysis shows that each estimate has its own advantages and disadvantages depending on the motion of the target linear or non-linear. As per results particle filter gives the better estimation for a complete nonlinear.

In another research Kalman filter based hybrid framework [10] is proposed the fusion of Automatic Dependent, Surveillance-Broadcast (ADS-B) and MLAT to achieve high precision and accuracy for ATC (Air Traffic Controller). In the proposed hybrid design the ADS-B data are fused firstly, and then the output of ADS-B is fused with MLAT to obtain the results. Both the radars can detect the exact position of the approaching target. ADS-B is equipped with GPS which gives velocity, speed and heading of the approaching target [10]. In the first step parameters which are extracted from ADS-B and parameters taken from GPS are fused. In the next step the result of fusion taken from step one is further fused with MLAT data to get the final fused data.

Evaluation of the hybrid fusion framework includes indoor and outdoor testing. In the testing precision and accuracy of the target position was focused and it was observed that after fusion of ADS-B and MLAT error of the target position is reduced by 75%. Experiment results are witness that the proposed framework has good accuracy and performance.

Three data fusion algorithms based on Kalman filter namely [5], Measurement fusion (MS), Gain fusion (GF) and State Vector Fusion (SVF) have been implemented in MATLAB. Their performance has been evaluated through Percentage Fit Error (PFE), Mean Square Error (MSE) and Mean Absolute Error (MAE) results shows that the State Vector Fusion algorithm performed well as compare to other 2 fusion algorithms.

As per models which are described by Blackman and Y. Bar-Shalom and Li [11]. Many complicated frameworks tend to be nonlinear in general. When solving the state estimation problem, non-linear models are often administered which are not pertinent in the linear Kalman filter.

2.3.2 Interactive Multiple Model

During the lifetime of maneuvering targets, they display different motion modes [7]. The KF family mentioned above needs information about the underlying state transition model [7]. The estimation outcome would be imprecise, if the incorrect filter from KF family is used. In this case, one can run multiple filters in parallel and the target motion is presumed to be in one of the 'n' possible modes, in each filter. This technique is known as an Interactive Multiple Model (IMM) [12].The IMM estimates the status of a dynamic system with different system transition, which can change the status one from another.

2.3.3 Particle Filters

Particle Filters are sub-set of Monte Carlo algorithms used for solving the filtering problems. It uses the samples called as particles for the representation of the posterior density of given observations. Particle Filter [7] provides a method for the generation of the sample from the desired density without requiring the assumptions about the state density. State of each particle$\{x_p^i(k)|i = 1,\ldots,N_p\}$ can be predicted through the equation:-

$$x_p^i(k+1) = f(x_p^i(k), v(k)) \tag{2.10}$$

The measurement likelihood function is used to calculate the posterior probability of the sample is as shown in the equation below:

$$w^i(k+1) = \frac{w^i(k)p(z(k+1) \mid x_p^i(k+1))}{\sum_{j=1}^{N_p} w^j(k)p(z(k+1) \mid x_p^j(k+1))} \tag{2.11}$$

Resample 'N_p' particles of equal weight $\left\{x_p^i(k)\middle|i = 1,\ldots,N_p\right\}$ from the weight

$$\{(x_p^i(k), w^i(k)) \mid i = 1,\ldots, N_p\} \tag{2.12}$$

Lastly, the estimate $\hat{x}(k)$ can be calculated through posterior density as:-

$$\hat{x}(k) = \frac{1}{N_p}\sum_{i=1}^{N_p} x_p^i(k) \tag{2.13}$$

Interacting Multiple Model (IMM) filter [12] can also be used for the estimation and fusion of the data. In this paper authors have discussed the data fusion in Air traffic management. As per International Civil Aviation Organization (ICAO) standards, Communications, Navigation, and Surveillance/Air Traffic Management (CNS/ATM) used different radars such as Ground-Based Augmentation System (GBAS), Automatic Dependent, Surveillance-Broadcast (ADS-B), Multilateration (MLAT) and wide-area Multilateration (WAM) for surveillance. Radar fusion method with GBAS, ADS-B, MLAT, and WAM data to Interacting Multiple Model (IMM) filter is proposed. In the proposed fusion algorithm covariance and estimates of radar measurements are taken with the IMM filter [12] and results are given to the main filter. After predication the estimates of each radar data are obtained and combined. The performance of the algorithm is evaluated on three different scenarios (Accelerated motion, uniform motion, and rotational motion) of air traffic control. The result of the simulation shows that the performance is improved by 40.93% as compared with available measurements of the radars. The limitation of the research is the poor estimates of aircraft location on some occasions which may further be improved through particle filter.

2.4 Gating

As information affiliation is complicated procedure, the data association mostly has strong computational requirements. According to Y. Bar-Shalom and Li [11], 10 targets were observed by tracking system at the last time step, and 80 new targets are received from the radar at the current time step.

The readings can be obtained from clutter, new objects or from objects that are tracked previously [11]. It will still take approximately two billion years to calculate all possible hypothesis association, if $P_d = 1$ and the hardware used has the ability to calculate one possible million associations per second. The complexity and data association time will be efficiently reduced by introducing a gate for each track. Example is as shown in Figure 2.2. Gates are created around the predicted measurements‘ $\hat{y}^1_{k/k-1}$ ’ and ‘ $\hat{y}^2_{k/k-1}$ ’ of earlier seen tracks at last time ‘ $k-1$ ’. All readings within the gating region are then considered contenders in the algorithm of data association while a separate logic called track management is required to make decisions regarding reading outside the gating regions [6]. This is used to exclude extremely unlikely readings to the current track as a potential new measurement. Among many different gating techniques, using Euclidean or absolute distance is the simplest.

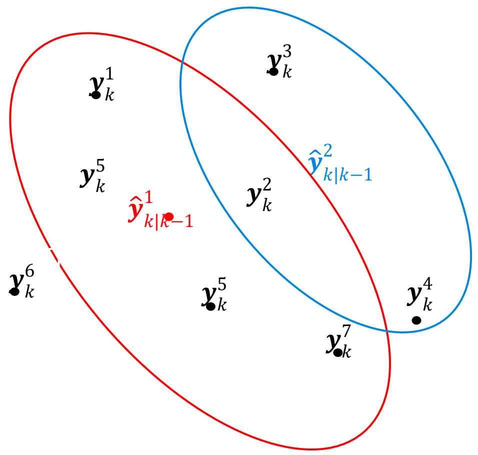

Figure 2.3: A Gating Example, of Elliptic Gates

The measurement is a valid contender for current track association, if the calculated distance is lower than the selected threshold, based on the likelihood of real measurement falling in the gating region. Problematically, however, shown in Figure 2.3, there could be multiple readings within the gates and few measurements can appear in multiple gates, thus gating does not completely solve the track updating problem. Data association algorithm decides that which of the measurements truly belongs to the tracked objects [6]. If multiple tracks are updated using the same measurement, tracks may unify after a while if this is not taken into account in the data association algorithm.

2.5 Data association algorithms

After the gating process is completed, the easiest technique for data association is to allow each track use the readings with the smallest statistical distance as shown in equation (2.14).

$$d(p,q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + .. + (p_n - q_n)^2} \qquad \textbf{(2.14)}$$

This is called the Nearest-neighbor (NN) algorithm [6] and is regarded as the fastest data association algorithm. Locally, NN minimize the association distance independently for each track. This leads to this track being able to share readings, i.e. if the reading is found within their gates, one measurement is used for updating various tracks. This could lead to combination of tracking as stated earlier. A better alternative is to simultaneously consider all tracks; with the limitation that only one track can be connected with a measurement. This may lead to a global optimization problem.

$$\min_{k} \sum_{i=1}^{N_{k-1}} (d^2)^{i,j} + \ln\sqrt{|S_k^i|} \quad \{\forall\, j : (d^2)^{i,j} \angle G\} \qquad \textbf{(2.15)}$$

Whereas ‘k’ Ɛ (0, 1, 2……N), is global associative vector and carries the measurement ‘j’ assigned to track ‘i’ with the limitation that each informational of the measurement can be used for one time only. The term in equation (2.15) ‘$\ln\sqrt{|S_k^i|}$’ is included just to ensure that high quality track measurements should be safeguarded in comparison with poor track qualities. The mentioned optimization problem can be solved through the Global Nearest-Neighbor (GNN) algorithm [6], once the optimization problem is solved. GNN ensures that a measurement is not shared and that the global statistical distance is kept to a minimum. But the complexity and computational load improves compared to the NN algorithm, as this is an optimization problem. A comparison can be shown below between the association outcome from the NN and the GNN [6].

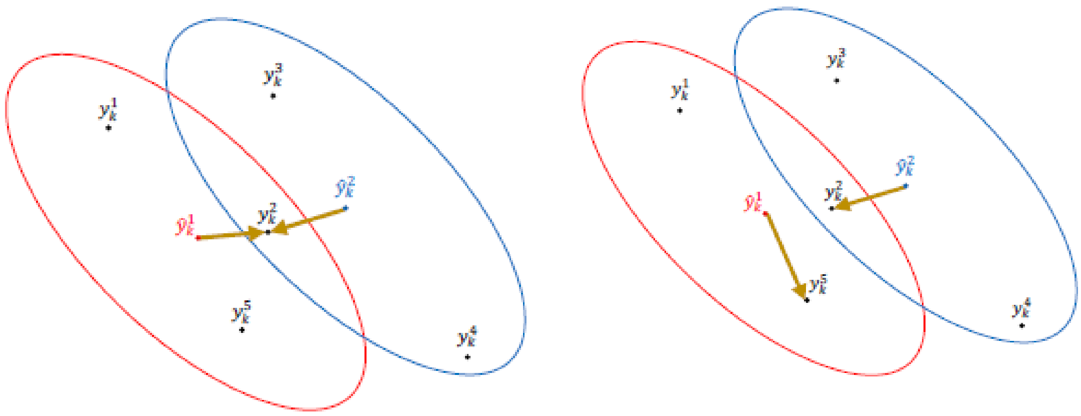

Figure 2.4: Examples of nearest neighbor and Global nearest neighbor

Even though the GNN finds the global optimum, many writers still criticize it. This is because if the measurements within the gating region arise from the real object or clutter, no consideration is given. In an extremely cluttered environment, instead of the object's real measurements, it is feasible that the hypothesis is modeled on clutter. In the cluttered environment, there is always a possibility that target is created on the cluttered instead of the true target, therefore, there may be a chance of measurement inside the gated region, and this approach is known as all-neighbors. The approach is based on the Bayesian probability, where cluttered is shown as a passion method, Noise and the density ‘β’ is implicit as Gaussian [7]. The first algorithm known as Probabilistic Data Association (PDA) using this was presented by Yaakov Bar

Shalom and Tse [13]. The probability of detection of a single established target is defined as ‘ P_D ’ and ‘ P_G ’ fall inside the gating region with ‘N’ observations. There is ‘N+1’ assumption that can be made against a single target. Let’s assume that ‘ H_0 ’ is the scenario where not even a single measurement pertains to the detected target, then ‘ H_0 ’ according to [14].

$$p^{i01} = \beta^{N}(1 - P_D P_G) \tag{2.16}$$

The remaining assumptions H_j (j = 1, 2, and 3….N), the proportional probability of at least one measurement represents:-

$$p^{ij\iota} = \beta^{N-1} P_D P_G \frac{e^{-(d^2)^{i,j/2}}}{P_G (2\pi)^{M/2}\sqrt{S_k^i}} \tag{2.17}$$

Where, ‘M’ represents the dimension. The probability can be calculated by the equation

$$p^{ij\iota} = \frac{p^{ij\iota}}{\sum_{i=0}^{N} p^{ij\iota}} \tag{2.18}$$

After the calculation of the equation (2.16) to (2.18), the assumption is combined and the method of combination is given in [13].

In HMM algorithm [15] likelihood is calculated on the bases of sequence of observation which is called as similarity metrics, which is further used for association of the targets.

$$a_{ij} = p[q_t = S_j \mid q_{t-1} = S_i] \quad \text{where}\, a_{ij} \leq 1w \tag{2.19}$$

$$\sum_{j=1}^{N} a_{ij} = 1 \tag{2.20}$$

The Hidden Markov Model (HMM) is an association method which is used to arrest transition probabilities through the state space training [15]. The proposed method has benefits of balance between tracking performance and the computational complexities. Therefore, with less computational complexity, HMM achieves better performance. HMM evaluates and compares with other data association techniques of multi object tracking, the evaluation of the results shows that HMM has much better performance than NNSF approach.

There are a number of distinct solutions to the issue of assignment; however the Munkres algorithm has traditionally been used to fix this issue. The Jonker-

Volgenant- Castanon (JVC) [16] or the Auction algorithm is dramatically quicker than the Munkres algorithm [8] when comparing distinct algorithms to fix the assignment problem. In sparse matrices the Auction algorithm is regarded quicker, but in dense matrices JVC is preferred. A dense or sparse matrix is the number of tracks within the gating threshold. If there is a small gating threshold, the Auction algorithm is favored.

The Auction algorithm was proposed in 1979 [17] for the classical assignment problem. The purpose of the algorithm was to give the solution of the problem by using parallelism, but this algorithm also performed very fast and efficiently to solve the serial and network flow problem. Following work extended the Auction algorithm to other linear network flow problems. The author also proposed shortest path and max flow algorithms for the solution to solve minimum cost problem. All mentioned algorithms were derived from the original Auction algorithm which was proposed in 1979, but our focus will be on basic Auction algorithm for solving assignment problem.

In the classical assignment problem [17] there are 'n' object and 'n' persons, and we have to assign one object to one person. There is a benefit a_{ij} for the equivalent person 'i' with object 'j' and we indent to maximize the advantage by assigning persons to objects. We have a set 'A' of pair (i, j) that can be accorded. We denote each person by A(i) from the collection of objects matching with 'i'.

$$\mathbf{A(i) = \{j \mid (i, j) \in A\}} \qquad \textbf{(2.21)}$$

B(j) is used for the representation of individual object from the set of person that matches object 'j'.

$$\mathbf{B(j) = \{i \mid (i, j) \in A\}} \qquad \textbf{(2.22)}$$

In Auction algorithm assignment means a set which is denoted by 'S' of Person to object pairs (i, j) in which each object 'j' and person 'i' is involved in one pair from set 'S'. if the number of pairs are 'n' in set 'S' then each person is assigned with a different object, we can say that set 'S' is feasible, otherwise set 'S' is infeasible. We assume that assignment is feasible [in person to object pairs $(1, j_1), (2, j_2), \ldots\ldots\ldots\ldots\ldots (n, j_n)$ from A, where all objects $\{ j_1, j_2, j_3, \ldots\ldots\ldots\ldots j_n \}$ are diverse], which is optimum in a sense that its maximum total benefits are $\sum_{i=1}^{n} a_{ij}$.

The assignment problem is imperative in many practical frameworks. The most evident ones are resource allocation problems, for example, personnel to jobs assignment, machines to tasks, and others. There may be scenarios where the assignment problem seems to be a sub-problem in numerous approaches for solving more complex issues; for example, track to track assignment or data association problems.

2.6 Data Fusion

Data fusion is the method of integrating various data sources in order to generate information that is more coherent, precise and helpful than any individual data source. Based on the processing stage at which fusion occurs, data fusion procedures are often classified as low, intermediate or high. Low-level fusion of data combines several raw data sources to generate new raw data [18]. Fused data is expected to be more informative and synthetic than the initial inputs.

Human beings are Data Fusion's prime instance. As human beings, we depend strongly on our senses like our Voice, Vision, Taste, Smell and Physical Movement. A mixture of all these senses combines on a regular basis to assist us does most, if not all, of our daily life [18]. That is a prime instance of data fusion in itself. To make sure it is edible or not, we depend on a fusion of smelling, tasting and touching food. Likewise, we depend on our vision and our capacity to hear and regulate our body's motion to walk or drive and do most of our life's duties. The Brain conducts the process of fusion in all these instances and checks what we have to do next. Our brain depends on a merger of information from the above-mentioned senses.

The level of data fusion can also be described based on the type of information used to supply in the fusion algorithm. More exactly, radar fusion can be conducted using raw data from distinct sources, extrapolated characteristics or even single node decisions.

2.6.1 Data level

Data level (or early) fusion is intended to fuse raw data from various sources and represent the smallest level of abstraction fusion method. In many areas of implementation, it is the most prevalent radar fusion method. Usually, data level fusion algorithms strive to combine various homogeneous radar data sources to obtain precise and synthetic measurements [3]. An example of data level fusion block diagram is depicted in Figure 2.5.

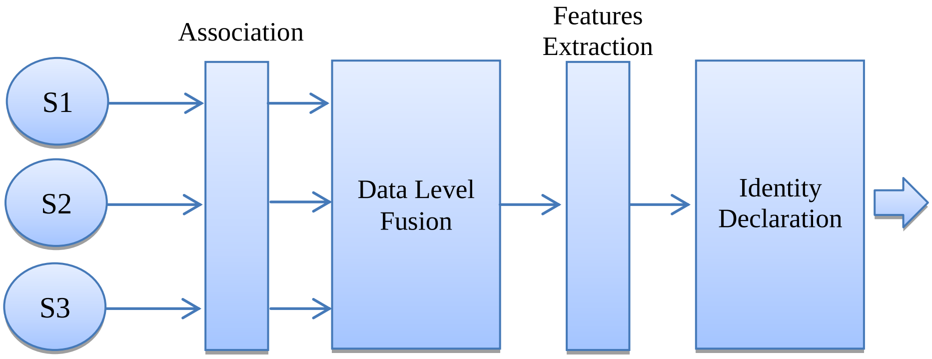

Figure 2.5: Data Level Fusion Data block Diagram

Data compression is a significant consideration when using portable devices, as gathering raw data from various sources produces enormous information spaces that could specify a problem for portable devices in terms of memory or communication bandwidth. Fusion of information at the data level tends to produce large input spaces that slow the process of decision-making. Furthermore, fusion of data level often cannot manage incomplete readings. If one of the modalities of radar becomes ineffective owing to malfunctions, breakdown or other reasons, the entire system may result in inconsistent results.

2.6.2 Feature level

The features represent data that each sensing node is calculated on board. To feed the fusion algorithm, these characteristics are then sent to a fusion node. With regard to data level fusion, this method produces lower information spaces, and this is better with respect to computational load. Obviously, it is essential to select correctly the characteristics on which classification processes are defined: the selection of the most effective characteristics set should be a key element of method design. Using selection algorithms for features that correctly detect correlated characteristics and subsets of features increases precision of recognition, but generally big training sets are needed to identify the most important subset of features. The data block diagram for feature level fusion is as shown in Figure 2.6.

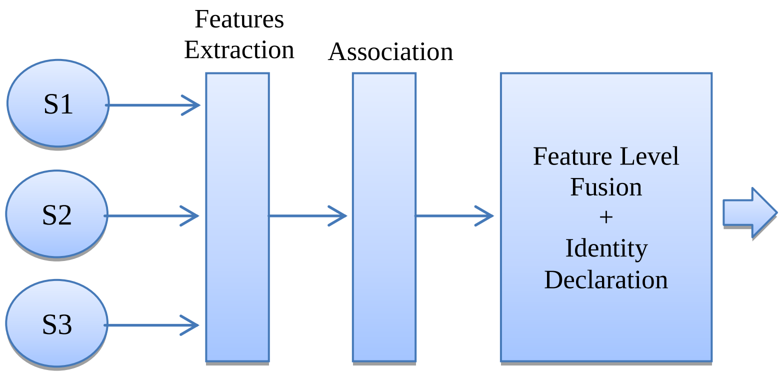

Figure 2.6: Feature Level Fusion Data block Diagram

With reference to the navigation and obstacle, avoidance of autonomous vehicle through multi radar data fusion can be done [19]. The author has improved the efficiency of virtual force field (VFF) navigation and obstacle avoidance algorithm with Back propagation fuzzy neural network.

According to the paper, a global planning path algorithm is used for the navigation of the vehicle, when any obstacle is detected through 12 radars [19], which are installed at the front, back and both sides of the vehicle, then using VFF algorithm steering angle of the vehicle is calculated and vehicle adjust its direction and the global path algorithm is updated based on the current calculation.

For the verification of the algorithm road environment is simulated, which consists of anti U obstacle, non-trap obstacle, emergency environment and U type obstacles. Simulation experiment is divided into 2 parts, hardware and software. Radars transmit the data to the simulation platform which calculates the path through Back propagation neural network data fusion method and VFF obstacle avoidance algorithm [19]. Finally, commands are given to the vehicle to avoid obstacles. The result shows that VFF algorithm based on neural network have increases the performance of autonomous vehicles. Resultantly, it can reduce economic losses and road accidents which can ultimately result into improved road environment.

2.6.3 Decision level

Decision level (late) fusion is the method for choosing a hypothesis from a collection of hypotheses produced by various nodes (generally weaker) decisions [3]. It is the abstraction of the highest level and uses the information that has already been developed through the processing of preliminary data or feature level. In decision fusion, the primary objective is to use a meta-level classifier while node information is pre-processed by extracting characteristics from them. In classification, a recognition activity is typically used for decision level radar fusion, and the two most popular methods are majority vote and Naive-Bayes. Benefits from the fusion of decision level involve communication bandwidth and enhanced precision of decision. It also enables multiple radars to be combined. The data block diagram for Decision level fusion is as represented in Figure 2.6.

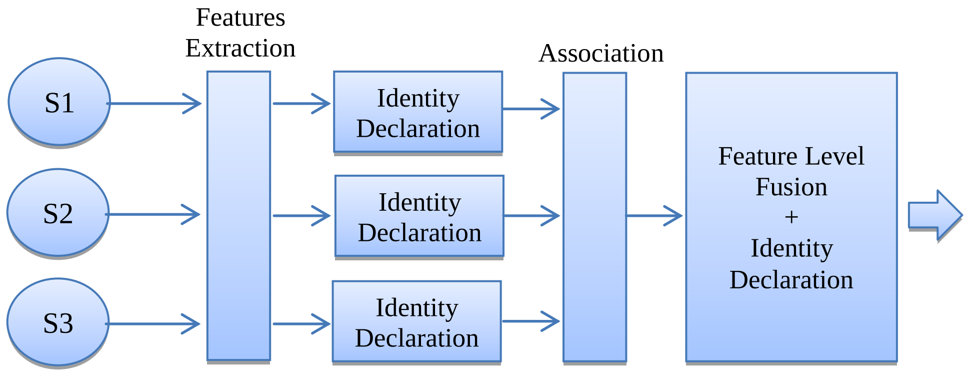

Figure 2.7: Decision Level Fusion Data block Diagram

In this paper author has highlighted an active area research that is neuromorphic radars [20]. The paper is divided into three major parts. In the first section different preprocessing methods of the spiking data from the event base radar for the use of deep network are discussed. The main purpose of preprocessing the input data from event-based radar is to produce, frame based representation which is used for the training of deep network. The deep network consists of Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNN). The combination of CNN and RNN deep network is used to train both audio and visual inputs using Python and keras software's [20]. In the last step Multimodal fusion with deep neural network takes visual feature and audio features as input and produces the results which are combinations of both the audio and visual features.

Two standard data sets MNIST digit recognition for visual data and TIDIGITS for audio data are used for the training and performance evaluation of the deep network. From TIDIGITS data set, 2464 digits are used for the training and 2486 digits for the testing of the deep neural network [20]. Similarly from MNIST data set 60,000 handwritten digits are used for the training and 10,000 digits for the evaluation of the deep network. Finally the result shows that 98.31% accuracy is achieved in classification as in comparison with other trained deep network.

Chapter 3

Proposed Algorithm

This chapter explains the design and overall structure of the proposed algorithm of multi radar data fusion. In the first step some concern related to design in the context of implementation is highlighted, which followed by a detailed discussion on the proposed Algorithm.

3.1 Description of Proposed Algorithm

The objective of this research is to design an effective and efficient multi radar data fusion algorithm that could combine the input of multiple disparate radars into a single positive target. Our goal is to provide an accurate and a reliable air picture, thereby, decreasing controller's load and enabling fast reaction, in order to handle various situations occurring in the air space.

The design of the proposed algorithm is based on the modular architecture, which provides the flexibility for the extension of more radar. A block diagram of the proposed multi radar data fusion algorithm is presented in Figure 3.1.

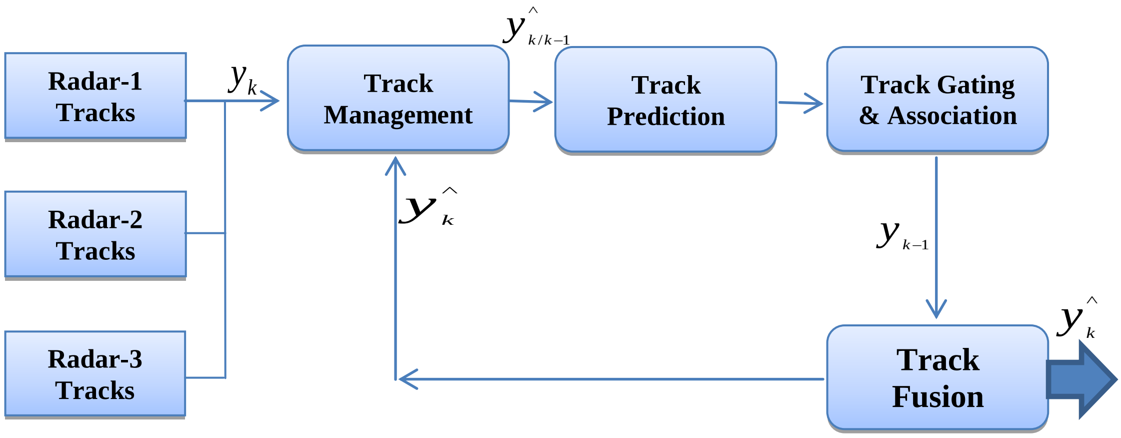

Figure 3.1: Proposed Multi Radar Data Fusion Algorithm

In the proposed architecture initially three disparate types of radars are used for the multi radar data fusion. Radar is uniquely identified by its ID, for example, radar-1 is denoted by 'R-1', radar-2 is denoted by 'R-2' and radar-3 is denoted by 'R-3'. Different radars have different scan time, which are configurable according to the user and operational requirement. At the end of each scan, radar sends its detected measurements to the central node called track management module over the network.

3.1.1 Track Management

The track management module is responsible for the tracking and management of all the tracks, which are received from radars. When the track is received, first of all measurement parameters such as latitude, longitude, speed, height, heading, age, track number and call sign are extracted. The value of X- velocity and Y-Velocity are calculated on the bases of heading and speed of each track.

The tracking process produces the tracks which are active in the environment and detected by the radars, problem arises, when a target is no longer in the environment, then we required an efficient tracking algorithm, which maintains the record of every periodic update of each track and decides for how long time, the track should be kept since the last update of the track is not received.

Each track contains a unique identification of its source radar, time stamp and following measurement parameters:-

- Longitudinal distance to the object is denoted by $\{x_{R_1}, x_{R_2}, x_{R_3}\}$
- Lateral distance to the object is denoted by $\{\mathrm{y}_{R_1}, \mathrm{y}_{R_2}, \mathrm{y}_{R_3}\}$
- The relative longitudinal velocity of the object is denoted by $\{\mathrm{y}_{v_1}, \mathrm{y}_{v_2}, \mathrm{y}_{v_3}\}$
- The relative lateral velocity of the object is denoted by $\{x_{v_1}, x_{v_2}, x_{v_3}\}$
- A unique identification, reported for each track by each radar is denoted by $\{\mathrm{Id}_{R_1}, \mathrm{Id}_{R_2}, \mathrm{Id}_{R_3}\}$
- A Speed reported for each track by each radar is denoted by $\{S_{R_1}, S_{R_2}, S_{R_3}\}$
- The Height of the track reported by radar is denoted by $\{\mathrm{H}_{R_1}, \mathrm{H}_{R_2}, \mathrm{H}_{R_3}\}$
- The Height reported by the target itself is denoted by $\{\mathrm{H}_{cR_1}, \mathrm{H}_{cR_2}, \mathrm{H}_{cR_3}\}$
- An age of the track reported by radar is denoted by $\{\mathrm{T}_{QR_1}, \mathrm{T}_{QR_2}, \mathrm{T}_{QR_3}\}$
- A Heading reported for each track by each radar is denoted by $\{\mathrm{h}_{R_1}, \mathrm{h}_{R_2}, \mathrm{h}_{R_3}\}$
- A Call Sign reported for each track by each radar is denoted by $\{\mathrm{C}_{R_1}, \mathrm{C}_{R_2}, \mathrm{C}_{R_3}\}$

Each radar has its own built in tracker unit, which is responsible for the production of tracks from its detected raw data. We assume that noise factor of the radar is well managed by the radar built in tracking unit. When the measurements are received from the radar, it is assumed, that the targets which are rendered to the tracking module are positive and no more than one target is generated by the radar tracking unit. With this assumption only positive targets are forwarded to the central track management without any multiple targets or clutter.

3.1.2 Track Prediction

Track prediction refers to the Kinematic estimation of the target, which typically predicts the position of the target, if live updates are not received from the radar. Multi radar data fusion depends upon the prediction at time ‘k-1’ and next position of the target at time ‘k’. Therefore, we used the Kalman filter, which uses the current and previous measurements to estimate the next position of the target. Kalman filter shown in section (2.2.1) is frequently used for the estimation of the state for a target.

Live updates are always dependent upon the scan time of each radar, which may vary according to the user and operational requirement. In multi sensor data fusion, we have to define a fixed interval of time, after which all available measurements are transferred to the fusion module. In our proposed fusion algorithm, we have taken an interval of five second for the fusion of all available updates of the targets. If the live update of any target is not received within five second, then the next position of the target is predicted by Kalman filter and estimated update is transferred to the fusion module for further processing.

3.1.3 Gating and Association

In Gating and the association module of the proposed algorithm, some important parameters such as ‘x’ velocity and ‘y’ velocity are calculated against each track.
Equation 3.1 and equation 3.2 is used for the calculation ‘x’ velocity, which is denoted by ‘v_x’ and ‘y’ velocity which is denoted by ‘v_y’. Track features such as speed and heading is used for the calculation of said parameters.

$$\mathrm{v_x} = \mathrm{v}\cos\theta \quad \textbf{(3.1)}$$

$$\mathrm{v_y} = \mathrm{v}\sin\theta \quad \textbf{(3.2)}$$

Where ‘v’ is the velocity and can be calculated with the help of speed and time

$$\mathrm{v} = \mathrm{S} / \mathrm{T} \quad \textbf{(3.3)}$$

Above calculated parameters and the Latitude, Longitude of the reference point, are used to calculate the ‘X’ estimate of each track. Assignment of the parameters in the column matrix against each track is depicted in column matrix 3.4.

$$X_{Est} = \begin{bmatrix} G_{Lat} \\ v_x \\ G_{Long} \\ v_y \\ H_c \\ 0 \end{bmatrix} \tag{3.4}$$

In the above column vector ‘G_{Lat}’ and ‘$G_{\log}$’ represents the Global ‘x and ‘y’ position which we have taken as center of the earth, whereas,‘v_X’and ‘v_y’ denotes x-velocity and y-velocity whereas H_C denotes the Height of the target. After the calculation of the X-estimate, value of the D-square is calculated with the help of the mention below equation 3.5 against each track:

$$\begin{aligned} C = [\{\alpha * X_{Est'}(1,1) * X_{Est'}(1,1) * \alpha\} + \{\alpha * X_{Est'}(3,1) * X_{Est'}(3,1) * \alpha\} \\ + \{X_{Est'}(4,1) * X_{Est'}(4,1)\} + \{\beta * X_{Est'}(5,1) * X_{Est'}(5,1) * \beta\} + (D * D)] \end{aligned} \tag{3.5}$$

In equation 3.5 **‘C’** represents the D-Square value of a track, ‘α’ represents the normalizing factor, ‘β’ represents another normalizing factor, whereas, **‘**D**’** represents the distance between the two tracks with respect to the radius of the earth. Distance is calculated with the help of equation (3.8). Haversine formula [21] is used to calculate the distance between two points that is the shortest distance between two points over the surface of the earth.

$$a = \sin^2(\Delta\phi / 2) + \cos\phi_1 * \cos\phi_2 * \sin^2(\Delta\lambda / 2) \tag{3.6}$$

$$c = 2 * a\tan 2(\sqrt{a}, \sqrt{(1-a)}) \tag{3.7}$$

$$D = R * c \tag{3.8}$$

In equation 3.6, ‘φ’ represents the latitude, ‘λ’ is the longitude, ‘c’ depicts the angular distance in radians, ‘a’ describes the square of half the chord length between the points and ‘R’ represents the mean radius of the earth, which is equal to 6,371 Km [21]. If the value of D-Square lies inside the gated region or less than the threshold value, then the track is considered a valid candidate for data association algorithm otherwise target is simply rejected and marked as an invalid candidate for current cycle of the data assignment process. Once it’s decided which are valid candidates for the assignment process, all those valid targets are further transferred to the Auction algorithm [17] for track to track association process.

Let's consider a process for determining the assignment and a price vector for each track. We will call this process as an Auction algorithm. This algorithm proceeds in repetitions and creates a sequence of assignment and price vector for each track. At the commencement of each repetition,

$$a_{ij_i} - p_{j_i} = \max_{j \in A(i)} \{a_{ij} - p_j\} \quad \textbf{(3.9)}$$

The above equation 3.9 'a_{ij_i}' represents the benefits, 'p_{j_i}' represents the price and (i,j) represents the pair of tracks which are from two different radars. If equation 3.9 satisfies all the pairs (i, j) of the assignment, then it means that tracks are assigned to the best suited corresponding tracks and algorithm terminates. Else a subset of unassigned tracks which is denoted by 'I', are selected and the following calculations are executed.

Let consider this subset of unassigned tracks and each track i $\in$ I, finds a track 'j' which offers maximum value that is

$$j_i \in \arg \max_{j \in A(i)} \{a_{ij} - p_j\} \quad \textbf{(3.10)}$$

And calculates a bidding increment

$$\gamma_i = v_i - w_i \quad \textbf{(3.11)}$$

In equation 3.11 'γ_i' represents a bidding increment. The value of 'γ_i' becomes '0' when maximum value for the bidder track 'i' is offered by more than one track. Resultantly, it may happen that a smaller number of equally desirable tracks are contested by more than one track, without increasing their prices. It creates a never ending cycle. To break such type of cycle perturbation is introduced, in which every bid raises the price by a minimum positive increment called as epsilon '$\in$' which is a positive scalar value,

$$\gamma_i = v_i - w_i + \in \quad \textbf{(3.12)}$$

Where 'v_i' represents the best value

$$v_i = \max_{j \in A(i)} \{a_{ij} - p_j\} \quad \textbf{(3.13)}$$

And 'w_i' represents the second best value

$$w_i = \max_{j \in A(i), j \neq j} \{a_{ij} - p_j\} \quad \textbf{(3.14)}$$

When each track 'j' which is nominated to be the best track for track of the nonempty sub-set 'p_j' decides the highest bidder

$$i_j = \arg\max_{i \in p(j)} \gamma_i \tag{3.15}$$

Highest bidding increment raises the prices of the tracks '$\max_{i \in p(j)} \gamma_i$' and assigned to next the highest bidder 'i_j' and the assignment to track 'j', done at the start of the repetition cycle (if any) is removed. This process continues until all tracks are assigned with best suited tracks.

3.1.4 Fusion

The Auction algorithm returns the best match tracks to the fusion module. In our proposed algorithm we have used a feature level data fusion; therefore, we don't need to de-correlate the already fused tracks on the start of the next fusion cycle. The fusion process does not contain the information, whether the upcoming updates of each track are already fused or not, therefore, track list is re-formed at each time stamp. The data flow diagram of the proposed fusion algorithm is depicted in Figure 3.2.

The Auction algorithm returns the best match track, which means that both associated targets represent the single target in the air and due to multi radar environment single target is detected multiple times. Now our algorithm decides which track is going to be the master and another is to be a slave. Only mater tracks are transferred to the user interface for their situational awareness and slave target will be invisible. When two or more tracks are fused with each other than it's difficult to distinguish between master and slave. In our proposed algorithm, track age as a key feature of each track which decides between the master and the slave.

Age is the one of the most important feature of the track, which is calculated on the bases of detection per scan by the radar tracking unit, if the target is detected in every scan of the radar and there is no miss in any scan, then the target will have a higher value of the age. Therefore, the track having a higher age value is marked as master, and with less age value is marked as a slave. Master track is transferred to the user interface for user surveillance and decision making process, whereas, slave will be invisible for the user.

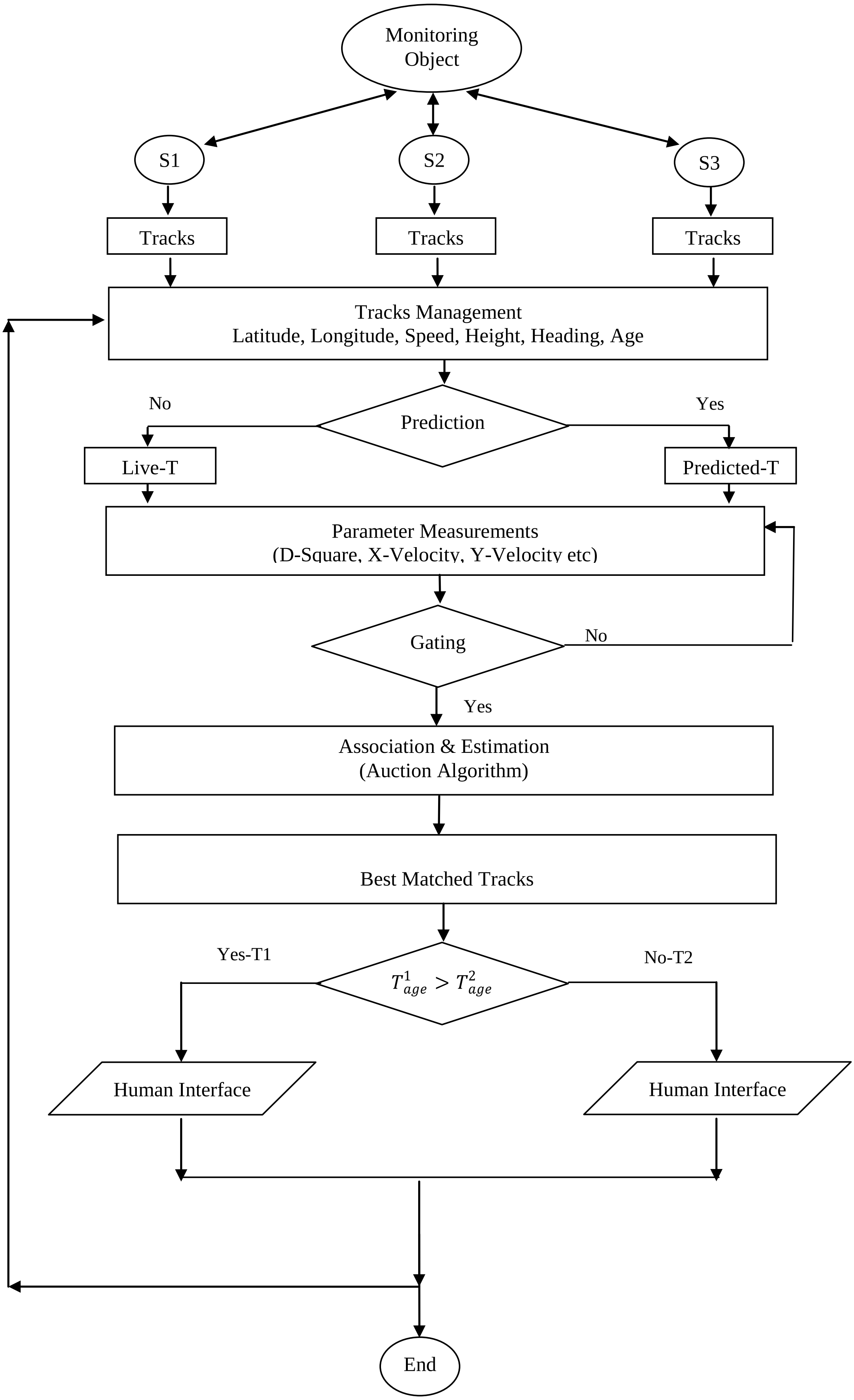

Figure 3.2: Proposed Fusion Algorithm Flow Diagram

Chapter 4

Results

This chapter presents the practical implementation of the proposed Multi Radar Data Fusion algorithm presented in section 3.1. To demonstrate the capability of the proposed algorithm, multi radar data fusion algorithm is implemented in MATLAB. After a detailed discussion with the Air Traffic controller, real world data and live scenarios are selected for the evaluation of the algorithm.

4.1 Data Set

The accuracy of the multi radar data fusion algorithm is tested by using real world data of domestic and international flights. Detailed discussion was done with air traffic controller about their, SOPs and daily traffic handling methods. Different scenarios were discussed in detail, and three live scenarios were selected as a test data for the evaluation of the algorithm. Live data of the commercial flights were used to evaluate the accuracy of the algorithm. Summarized form of the live scenarios is stated in the Table 4.1.

Table 4.1: Parameters Summarized of Live Data Set

S.NO	Flight ID	Source	Destination	Flight Type	Dated	Features
1.	PK301	Islamabad	Karachi	Domestic	30/08/2019	Latitude, Longitude, Speed, Height, Heading, Call Sign, Age
2.	AB612	Peshawar	Sharja	International	28/08/2019	
3.	B788	Islamabad	London	International	30/08/2019	

In the first scenario, we selected the data of a domestic flight, which was planned from Islamabad to Karachi with a flight ID '**PK301'.**

In the second scenario, we selected the data of an international flight, which was planned from Peshawar to Sharja with flight ID **'AB612'.**

In the third scenario, we selected the data of an international flight, which was planned from Islamabad to London with flight ID '**B788'.**

The results of our proposed algorithm were compared with the Automatic dependent surveillance-broadcast (ADS-B), which is an air surveillance low cost system of the next generation that will replace the conventional radar. ADS-B is completely redefining the concept of navigation, Surveillance and communication in air traffic management. With the help of ADS-B air traffic controller and pilots can control the aircraft with more precision and accuracy over the surface of earth than ever before. It is more precise and reliable in its position because aircraft equipped with ADS-B entirely depends upon the GNSS.

4.2 Scenario-1

In the first scenario we have taken the data of an International flight, which was planned from Peshawar to Sharja. After its take-off from Peshawar, the flight was identified through its call sign **'AB612'.** The flight was detected by two different radars A and B.

Red circles in the first portion of the Figure 4.1 shows the live updates of the target detected by the Radar-A. Whereas, the second portion of the graph represents the live as well as predicted updates of the same target. The predicted updates are represented by the pink circle in the graph.

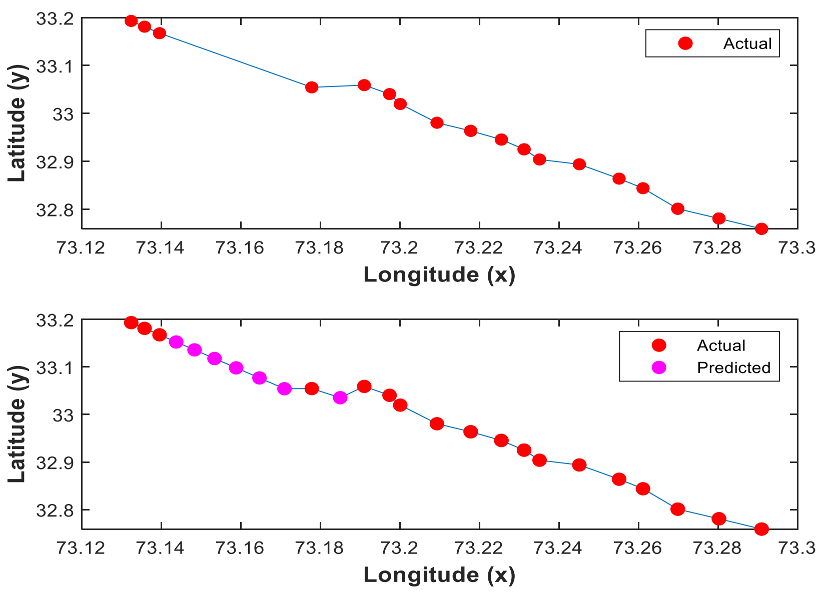

Figure 4.1: Radar-A, Live and Predicated Updates of Scenario-1

Blue circles in the first portion of the Figure 4.2 show the live updates of the same target detected by the Radar-B. Whereas, the second portion of the graph represents the live as well as predicted updates of the target. The predicted updates are represented by the pink circle in the graph.

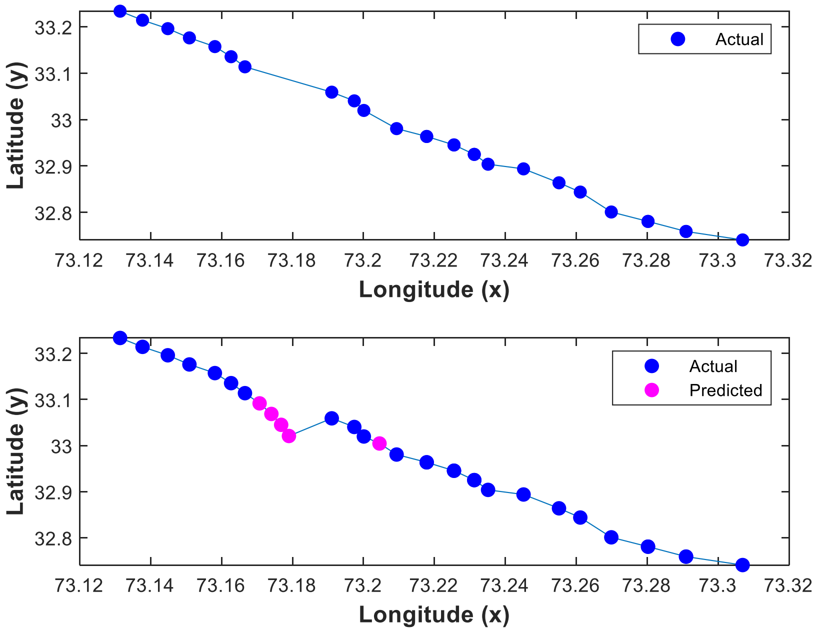

Figure 4.2: Radar-B, Live and Predicated Updates of Scenario-1

When the target is detected and reported by radar-A and radar-B, then features mentioned in section 3.1 are extracted and procedure mentioned in section 3.1.1, 3.1.2, 3.1.3 and section 3.1.4 is applied. The fused updates of radar A and B are depicted in Figure 4.3.

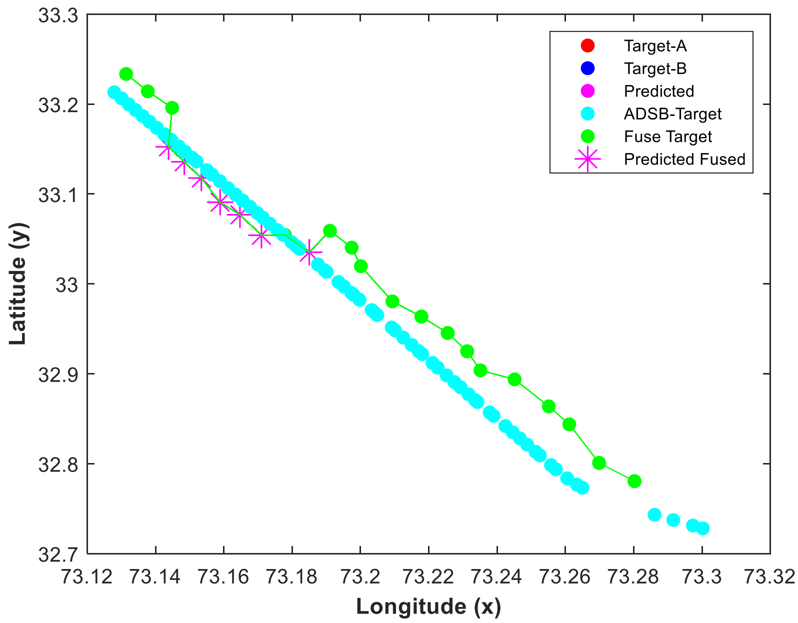

Figure 4.3: Fused Graph of Scenario-1

The fused updates of radar ‘A’ and radar ‘B’ are depicted in the green color. The

ADS-B data is used for the true reference of the target, which depicts the actual position of the aircraft. The same target received by the ADS-B receiver is depicted in cyan colour.

4.3 Scenario-2

In the second scenario, we have taken the data of a domestic flight which was planned from Islamabad to Karachi. After its take-off from Islamabad, the flight was identified through its call-sign **'PK301'.** The flight was detected by two different radars A and B.

Red circles in the first portion of the Figure 4.4 shows the live updates of the target detected by the Radar-A. Whereas, the second portion of the graph represents the live as well as predicted updates of the same target. The predicted updates are represented by the pink circle in the graph.

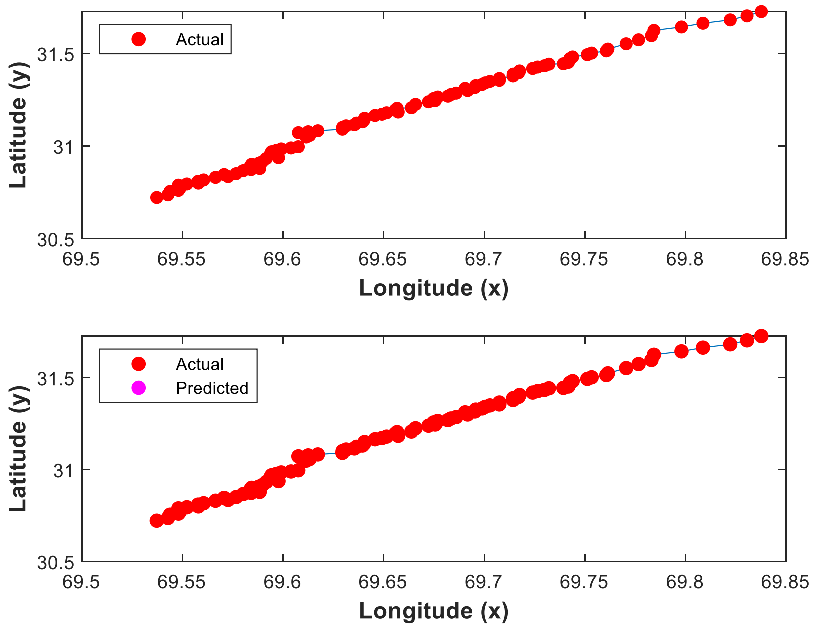

Figure 4.4: Radar-A, Live and Predicated Updates of Scenario-2

Blue circles in the first portion of the Figure 4.5; show the live updates of the same target detected by the Radar-B. Whereas, the second portion of the graph represents the live as well as predicted updates of the same target. The predicted updates are represented by the pink circle in the graph.

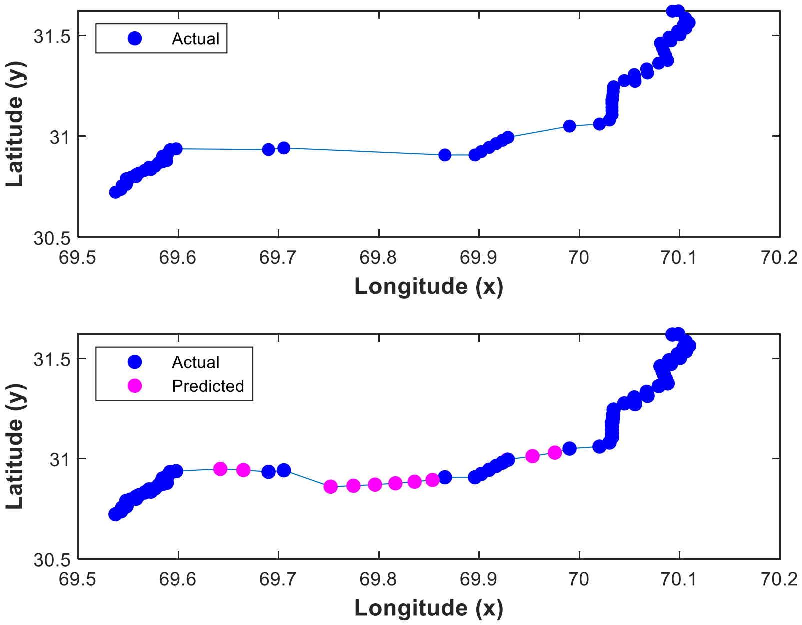

Figure 4.5: Radar-B, Live and Predicated Updates of Scenario-2

When the target is detected and reported by radar-A and radar-B, then features mentioned in section 3.1 are extracted and procedure mention in section 3.1.1, 3.1.2, 3.1.3 and section 3.1.4 is applied. The fused updates of radar A and B are depicted in Figure 4.6.

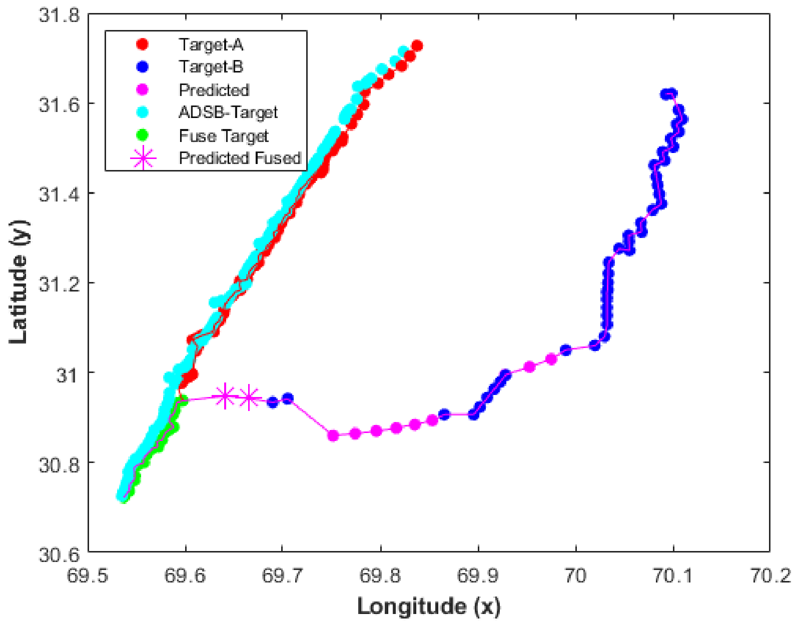

Figure 4.6: Fused Graph of Scenario-2

The fused updates of both the radars 'A' and radar 'B' are depicted in green color. The ADS-B data is used for the true reference of the target, which depicts the actual position of the aircraft. The same target received by the ADS-B receiver is depicted in cyan colour.

4.4 Scenario-3

In scenario-3, we selected the data of an International flight, which was planned from Islamabad to London. The flight was identified through its call-sign '**B788**'. In this scenario the flight was detected by three different radars, 'A', 'B' and 'C'.

Red circles in the first portion of the Figure 4.7, shows the live updates of the target detected by the Radar-A. Whereas, the second portion of the graph represents the live as well as predicted updates of the same target. The predicted updates are represented by the pink circle in the graph.

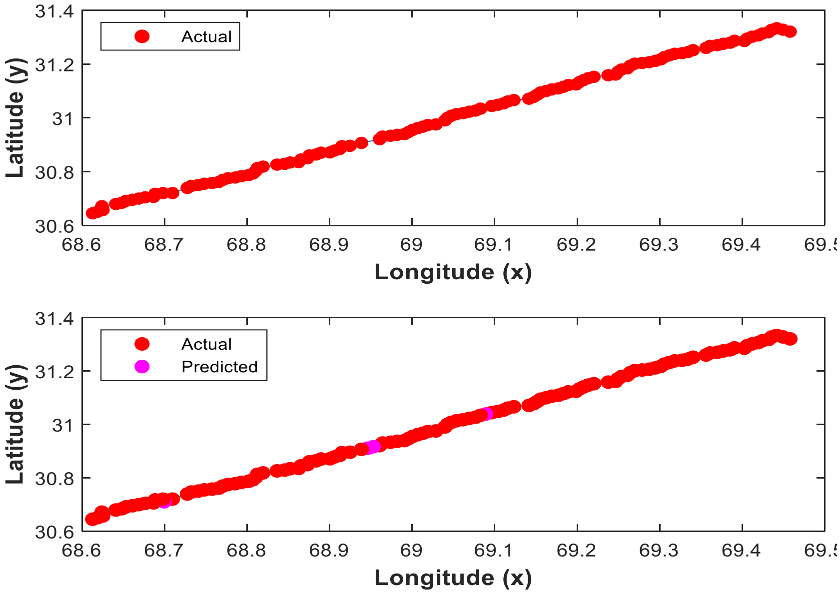

Figure 4.7: Radar-A, Live and Predicated Updates of Scenario-3

Blue circles in the first portion of the Figure 4.8 show the live updates of the same target detected by the Radar-B. Whereas, the second portion of the graph represents the live as well as predicted updates of the target. The predicted updates are represented by the pink circle in the graph.

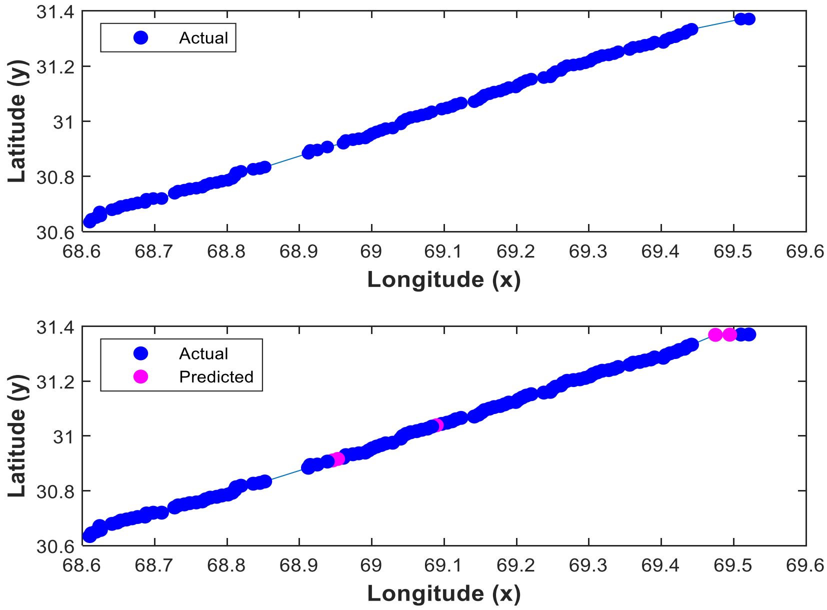

Figure 4.8: Radar-B, Live and Predicated Updates of Scenario-3

Black circles in the first portion of the Figure 4.9 shows the live updates of the same target detected by the Radar-C. Whereas, the second portion of the graph represents the live as well as predicted updates of the target. The predicted updates are represented by the pink circle in the graph.

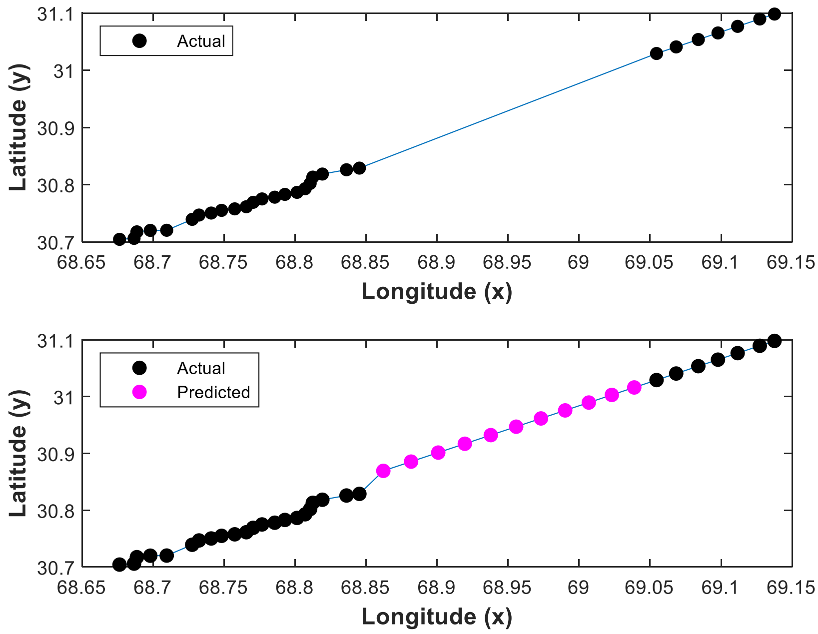

Figure 4.9: Radar-C, Live and Predicated Updates of Scenario-3

When the target is detected and reported by radar-A, radar-B and radar-C, then features mentioned in section 3.1 are extracted and Procedure mention in section 3.1.1, 3.1.2, 3.1.3 and section 3.1.4 is applied. The fused updates of radar ‘A’, ‘B’ and ‘C’ are depicted in Figure 4.6.

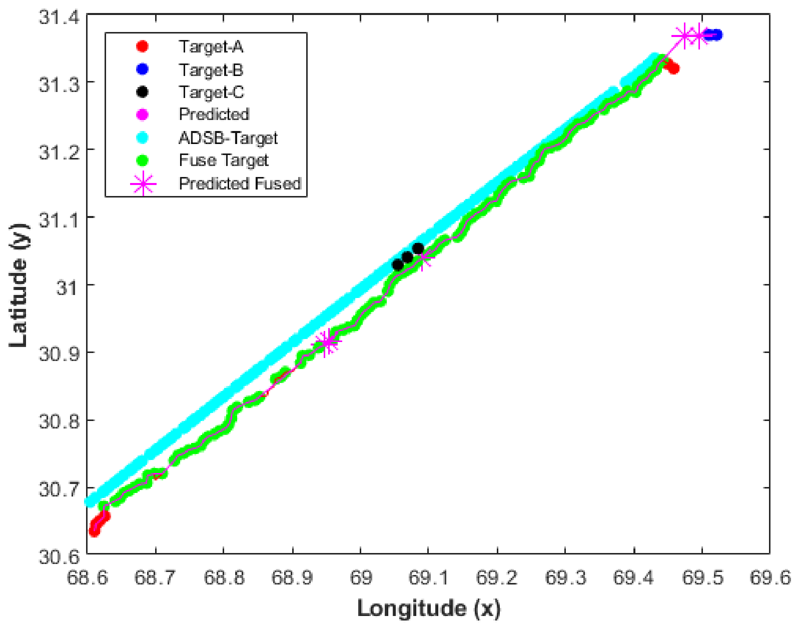

Figure 4.10: Fused Graph of Scenario-3

The fused updates of both the radar-A, radar-B and radar-C are depicted in green color. The ADS-B data is used for the true reference of the target which depicts the actual position of the aircraft. The same target received by the ADS-B receiver is depicted in cyan colour.

4.5 Quantitative Analysis

Quantitative analysis of scenario-1, scenario-2 and scenario-3 were carried out with the help of root mean square (RMS). The mathematical form of the RMS is depicted in equation 4.1.

$$\mathrm{RMS} = \sqrt{\frac{1}{S}\sum_{k=1}^{S}(y_{est}(k) - y_{ref}(k))^2} \qquad \textbf{(4.1)}$$

In the above equation ‘ y_{est} ’ denotes the fused value, ‘ y_{ref} ’ denotes the ADS-B value, and ‘S’ denotes the total number of updates of the target [22].

In scenario-1, the first RMS value is calculated for all the updates of radar-A against the respective updates of the ADS-B data. The same procedure is repeated to calculate the RMS value for all the updates of radar-B. Secondly, The RMS value is calculated for all the fused updates against the respective updates of the ADS-B data. The RMS values for radar-A, radar-B and fused data are shown in Table 4.2.

The same procedure is used for scenario-2 and scenario-3 for the calculation of RMS values for radar-A, radar-B and fused data, in scenario-3, data of three radars is used for the fusion of the target, therefore, The RMS value of radar-C is calculated by applying the same procedure. The RMS values of scenario-1, scenario-2 and scenario-3 are depicted in the Table 4.2.

Table 4.2: Quantitative Analysis Summary for Scenario1, 2 and 3

Scenario No	Radar-A	Radar-B	Radar-C	Fused Updates
1	1.1204	1.1517	-	1.1204
2	1.2448	3.6069		1.2448
3	1.2148	1.1516	1.1619	1.1516

Chapter 5

Discussion and Conclusions

This chapter presents a comprehensive discussion about the results attained in the previous chapter. The conclusions of the algorithm are based on the evaluation of the results through the implementation of different live scenarios. Required hardware software details are discussed and some recommendations are also made for future work.

The main objective of our research was to develop an algorithm that produces an accurate fusion of multiple disparate set of distributed radars, which results in a precise and reliable air picture. When coverage volume is increased by increasing the number of radars then, tracks could be formed, that remain unbroken over a larger area, which increases the accuracy of the target and its measurement parameters. This objective has successfully been achieved through our proposed algorithm.

Our multi radar data fusion algorithm was divided into different subsystems, each performs a specific task. Subsystems of our proposed algorithm are: track management, track prediction, track gating and association and track fusion respectively. The modular approach of the proposed algorithm enables us for partial replacement of these subsystems if required in future for enhancement in the algorithm. The tracking module is responsible for the management of all tracks received from different radars. Prediction module uses a Kalman filter for the prediction of the targets, if updates are not received from radars. The gating is done with the Mahalanobis distance, which is a common way of reducing the number of possible candidates for association process. The association process for finding a best match is done through the use of an Auction algorithm.

The proposed fusion algorithm was developed using MATLAB. Three different scenarios were selected after detailed discussion with air traffic controller. The Real world data of the selected scenarios were used to demonstrate the capability of the proposed algorithm. The results were evaluated in comparison with the ADS-B for the truth reference of the target as depicted in Figure (4.3), (4.6), and (4.10).

It has been observed in the industry, that the simulated scenarios are generated by the researchers to check the accuracy and performance of their algorithms. The exact evaluation of the results demands the true reference of the target, which is a difficult task to get until each target is equipped with the ADS-B interrogation system and data is openly available for the comparison.

In our research, we have achieved the goal of getting real world data. Live data of domestic and international flights is used for the implementation of the algorithm. The results of the implementation are evaluated through ADS-B data which is a modern tracking system in which aircraft uses satellite navigation system to decide its current position and regularly transmits to its ground bases controlling station.

When comparison is made between our fusion result and the ADS-B data, we may find some minor differences in the updates of the target as depicted in Figure (4.3), (4.6) and (4.10), which may be due to inherent issues of the radars, for example, the difference in the scan time of the radars, Inaccuracies may exist on the radar due to their type, their location where radar is installed, atmospheric condition and the age of the radar.

5.1 Hardware and Software Requirements

For multi radar data fusion algorithm, we need to correctly identify the requirements of hardware and software before the design and implementation of the proposed algorithm. In 1995 Y. Bar-Shalom and Li [13] claim that without considering the effect of hardware and software limitations, the researcher takes a big risk for the design of an algorithm. For some reason, if design and implementation is possible, then, future enhancements or requirements could not be possible for the designer to accommodate, if the proper and complete step by step systems design approach is not followed. In section 1.2, we discussed the challenges of multi radar data fusion. When the number of radar increases, it increases the amount of information. Resultantly, it increases the cost and hardware requirement. The hardware and software, which are used for the implementation of the proposed algorithm is mentioned in Table 5.1.

Table 5.1: Summarized Requirements of the Software

Software	Description
Operating System	Microsoft Windows 10 Pro
Programming Tool	MATLAB, R2018

Table 5.2: Summarized Requirements of the Hardware

Hardware	Description
Processor	Intel (R) Core (TM) i7-4710MQ CPU 2.50 GHZ, 2.49 GHZ
Memory	16.0 GB
Hard Disk	1 TB,
Other	Standard computer peripherals

Scenario -1, which carries total '71' updates of each radar takes '10.543241' seconds to complete the fusion process. Scenario 2, which contains total '110', updates of each radar takes '26.744750' seconds to complete the fusion process. Similarly, Scenario 3, which contains total '120', updates of each radar takes '29.978604' seconds to complete the fusion process. Table 5.3 represents the fusion processing time of each scenario with no of radars and it's no of updates.

Table 5.3: Fusion Processing Time for Scenario 1, 2, 3

Scenario	No of Radars	No of Updates	Fusion processing time
1	2	71	10.543241
2	2	110	26.744750
3	3	120	29.978604

5.2 Future Work

As concluded, the results of our proposed algorithm are very encouraging, we recommend that the algorithm should be implemented in any low level language like 'C' or 'C++' and must be used operationally by the air traffic controllers for their routine operations. We hope, our proposed algorithm will enhance the efficiency and reliability of the air traffic operations.

Obviously, our suggested fusion technique requires a refinement and enhancement to enhance efficiency by reducing the mistake of estimation. In order to prevent error and noise in the measurements of radars proper filtering mechanism may be added. The Efficiency of the algorithm can be improved by introducing a state of the art user defined tracking and prediction module instead of relying on radar built in tracking unit.

References

[1] M. E. Liggins, D. L. Hall, and J. Llinas, *Handbook of Multisensor Data Fusion: Theory and Practice*. 2009.

[2] Y. Bar-Shalom and W. D. Blair, “Multitarget-Multisensor Tracking: Applications and Advances,” *Norwood, MA, Artech House, Inc., 2000*. 2000.

[3] G. Koshmak, A. Loutfi, and M. Linden, “Challenges and issues in multisensor fusion approach for fall detection: Review paper,” *Journal of Sensors*. 2016.

[4] Eloise, “Data Fusion Tactical Picture.” [Online]. Available: http://www.f-16.net/forum/viewtopic.php?f=36&t=28397.

[5] R. Anitha, S. Renuka, and A. Abudhahir, “Multi sensor data fusion algorithms for target tracking using multiple measurements,” *2013 IEEE Int. Conf. Comput. Intell. Comput. Res. IEEE ICCIC 2013*, pp. 2–5, 2013.

[6] H. Lundin, “Using non-kinematic information to reduce the complexity of data association,” 2016.

[7] B. K. HABTEMARIAM, “Effective Data Association Algorithms for Multitarget Tracking,” 2014.

[8] H. W. Kuhn, “The Hungarian method for the assignment problem,” in *50 Years of Integer Programming 1958-2008: From the Early Years to the State-of-the-Art*, 2010.

[9] D. S. R. Kondru and M. Celenk, “Predictive airborne target tracking using all-terrain fusion based mobile surveillance system,” *2018 52nd Annu. Conf. Inf. Sci. Syst. CISS 2018*, pp. 1–6, 2018.

[10] Y. Lu, R. S. Huang, and Z. L. Xu, “Multi-sensor data fusion based on ADS-B and MLAT in approach,” *Appl. Mech. Mater.*, vol. 602–605, pp. 2491–2494, 2014.

[11] Y. Bar-Shalom and X. R. Li, “Multitarget-Multisensor Tracking: Principles and Techniques.,” *IEEE Control Systems*. 1996.

[12] T. Cho, C. Lee, and S. Choi, “Multi-sensor fusion with interacting multiple model filter for improved aircraft position accuracy,” *Sensors (Switzerland)*, vol. 13, no. 4, pp. 4122–4137, 2013.

[13] Y. Bar-Shalom and E. Tse, “Tracking in a cluttered environment with probabilistic data association,” *Automatica*, 1975.

[14] R. Blackman, S. S. and Popoli, *Design and analysis of modern tracking systems*. 1999.

[15] N. A. Zaher, A. M. Aziz, and H. H. Ghouz, “A data association approach for multitarget tracking based on a Hidden Markov Model,” *ISPACS 2013 - 2013 Int. Symp. Intell. Signal Process. Commun. Syst.*, no. November, pp. 136–141, 2013.

[16] D. B. Malkoff, “Evaluation of the Jonker-Volgenant-Castanon (JVC) assignment algorithm for track association,” in *Signal Processing, Sensor Fusion, and Target Recognition VI*, 1997.

[17] D. P. Bertsekas, “Auction Algorithms for Network Flow Problems,” *Lids*, no. Ddm, pp. 1–54, 1992.

[18] “Data fusion,” *wikipedia*. 2019.

[19] M. Han, “Application of artificial intelligence detection system based on multi-sensor data fusion,” *Int. J. Online Eng.*, vol. 14, no. 6, pp. 31–43, 2018.

[20] D. Neil and S. C. Liu, “Effective sensor fusion with event-based sensors and deep network architectures,” in *Proceedings - IEEE International Symposium on Circuits*

and Systems, 2016.

[21] “Calculate distance, bearing and more between Latitude/Longitude points.” [Online]. Available: https://www.movable-type.co.uk/scripts/latlong.html.

[22] A. M. Sabatini and V. Genovese, “A Sensor Fusion Method for Tracking Vertical Velocity and Height Based on Inertial and Barometric Altimeter Measurements,” pp. 13324–13347, 2014.

[23] T. Johnsen, B. Hafskjold, S. Fagerlund, T. E. Strand, and A. Jensen, “Data Fusion for Improved Air Picture Generation in Air Defence Systems,” *Meet. Proc. RTO-MP-SCI-143, Pap. 13*, no. 2005, pp. 1–14, 2005.

[24] M. A. Richards, *Fundamental of Radar Signal Processing*. 2014.

[25] Stimson, *Introduction to Airborne Radar*. 1998.

[26] M. I. Skolnik, *Introduction To Radar Systems*. 1981.

[27] J. Han, “Multi-Sensor Data Fusion for Travel Time Estimation,” no. March, 2012.

[28] M. Theses, M. Reports, and T. Pinar, “Feature and Decision Level Fusion Using Multiple Kernel Learning and Fuzzy Integrals,” 2017.

[29] H. Jia, “Data Fusion Methodologies for Multisensor Aircraft Navigation Systems College of Aeronautics PhD Thesis PhD Thesis.”

[30] B. Olivier, G. Pierre, H. Nicolas, O. Loïc, T. Olivier, and T. Philippe, “Multi Sensor Data Fusion Architectures for Air Traffic Control Applications,” no. February, 2009.

[31] A. Berg and A. Käll, “Track-to-track Fusion for Multi-target Tracking Using Asynchronous and Delayed Data Master’s thesis in Systems, Control and Mechatronics,” 2017.

[32] L. Danielsson, *Tracking and radar sensor modelling for automotive safety systems*. 2010.

[33] C. Engineering, “DEVELOPMENT AND IMPLEMENTATION OF IMAGE FUSION ALGORITHMS BASED ON WAVELETS DEVELOPMENT AND IMPLEMENTATION OF IMAGE FUSION ALGORITHMS BASED ON,” no. May, 2013.