

Community Model - A New Data Fusion Filter Paradigm

Boudhayan Bhattacharya
Dept. of Computer Application
Sabita Devi Education Trust – Brainware Group of
Institutions
Kolkata, India
mailforboudhayan@gmail.com

Banani Saha
Dept. of Computer Science & Engineering
University of Calcutta
Kolkata, India
Bsaha_29@yahoo.com

Abstract—Data fusion is an important area in different fields of applications as it provides a detailed analysis of the data store which helps in analytics and decision support system. Among the numerous Data Fusion models available in literature, the six pioneer pieces of work in terms of Data Fusion Models are – Waterfall Model, Multisensor Integration Fusion Model, Behavioral Knowledge-based Data Fusion Model, Omnibus Model, Dasarathy Model and JDL Model. In this paper, we have given a comparative study of the above models based on basic features like signalling overhead, power consumption, fault tolerance, survivability and robustness facilities subsequently. Here, a new improved model, named Community Model has been proposed for better throughput in terms of Data Transmission Time (DTT). This model can be applied to any number of filter levels irrespective of application areas.

Keywords—Community Model; Master Fusion Filter; Reference Sensor; Data Transmission Time; Fusion Domain; Situation Filtration; Impairment minimization; Information Fusion; State Fusion;

[2], pattern recognition and radar tracking [3], mine detection [4] and other military applications [5], remote sensing [6], traffic control [7],[8], aerospace systems [9], law enforcement [10], medicine, finance, meteorology [11], and geo-science. There are different structural models available in the literature[11]. Naturally the models are of prime importance as the structure defines the working principle in terms of the practical implication. It was found that all the proposed models try to incorporate a particular practical scenario and based on that decides the working principle. Here, a new functional model namely Community Model, has been proposed to extend the concept of fusion in different environments and it has been found that it yields better result in terms of transmission time with respect to the other models. This paper is organized as follows - Section 2 deals with the classical data fusion models and their comparison. Section 3 deals with the Community model and deduces a formula to calculate the DTT and introduces an algorithm for community model. Section 4 deals with the comparative analysis based on the DTT.

I. INTRODUCTION

With more and more information sources available via inexpensive network connections, either over the Internet or in company intranets, the desire to access all these sources through a consistent interface has been the driving force behind much research in the field of data fusion. The advantages of information integration systems is that the user of such a system obtains a complete yet concise overview of all existing data without needing to access all data sources separately. The applications of data fusion are used in different areas namely maintenance engineering [1], robotics

II. CLASSICAL DATA FUSION PARADIGMS

The literature provides many models to compose data fusion. It was found that the Waterfall Model, Multisensor Integration Fusion Model, Behavioral Knowledge-based Data Fusion Model, Omnibus Model, Dasarathy Model and JDL Model gives better result in different application specific areas.

A. *Waterfall Model* [11][14]

In waterfall model, it is observed that data flow operates from the data level to the decision making level. The continuous updation of sensor system is carried out on a steady basis with feedback information that arrives from the decision-making module. The feedback element provides advice to the multi-sensor system on re-configuration and data gathering aspects.

The three levels of representation in the waterfall model are:

1) *Level 1*

Transformation of raw data is carried out to provide the requisite information about the environment. For this to be achieved, the models of sensors are required. The models could be based on either physical laws or experimental analysis.

2) *Level 2*

This level comprises of feature extraction and thereafter the fusion of those features. This is done to obtain a symbolic level of inference about the data. The main objective here is to minimize the data content and maximize the amount of information delivered thereby. The output of this particular level is a list of estimates with the corresponding probabilities.

3) *Level 3*

This level maps objects to the corresponding events. On the basis of information that has been gathered, the available databases and libraries, and the human interaction, the possible routes of action are assembled.

B. *Multi-Sensor Integration Fusion Model* [11][15]

In Multi-Sensor Integration Fusion Model, data from the different sources was collected and combined in a hierarchical manner within the embedded fusion centres. Using this model, a clear distinction was made between multi-sensor integration and multi -sensor fusion. Multi-sensor integration is all about the usage of multiple sensor information in order to assist in a particular task, whereas multi-sensor fusion refers to any arbitrary stage of the integration process where the actual combination of data takes place.

Firstly data is collected at the sensor level which is then transferred to the fusion centres. The fusion process is thereafter carried out in the fusion centres which in turn could be done in a hierarchical and sequential manner. An information system is incorporated in the fusion process and is highly useful in the sense that it contains the relevant databases and libraries.

C. *Behavioral Knowledge-based Data Fusion Model* [11]

This framework comprises of a few basic stages which necessarily needs to be computed before the output is established. According to this framework, firstly a feature vector is extracted from the raw data. The feature vector is associated to the defined features, thereafter fusion is carried out at the data analysis levels and at the sensor attribute. Finally, a set of behavioral rules are extracted as a final representation of the fused output. This model describes the three levels of representation.

In the lowest level, corresponding to each sensor, there exists a vector space with measured parameters and coordinate dimensions. The second level attaches labels to the appropriate or relevant features that are in turn extracted from these vectors. Lastly, the third level comprises of a set of formalisms which relates or maps the feature vectors to the events.

This model is employed in the fields like human detection and identification like security control, monitoring, etc.

D. *Omnibus Data Fusion Model* [11][14]

Omnibus data Model, is a hybrid of the three models namely Boyd loop, Dasarathy, and Waterfall models. These modules address the variety of tasks in data fusion and it's functional objectives. The Boyd loop is an iterative process with four elements (namely observe, orient, decide and act) which operates in a closed loop.

E. *Dasarathy Model* [11][16]

In Dasarathy model, the three main levels of abstraction during the data fusion process are – Decisions, Features and Data. The Fusion might occur both within these levels and as a means of transformations between them.

F. *JDL Model* [11][14]

In this model, the information sources which are used for data fusion can very well include both local and distributed sensors (ones that are physically linked to the other platforms), or environmental data, a priori data, and human guidance or inferences. The JDL data fusion process contains the six different levels of processing.

G. *Comparison of Data Fusion Models*

This section deals with the comparison of various existing data fusion models based on different parameters like power consumption, fault tolerance etc.

TABLE I. COMPARISON OF SIX SIGNIFICANT DATA FUSION MODELS

Model/ Paradigm	Advantage /Contribution	Drawback/ Shortcomings	Level Traversability			Fusion Overhead
			Level 1	Level 2	Level N	
Waterfall Model	Reduce signaling overhead and power consumption	Frequent inter domain movement increase latency.	√	√	√	Low under restricted levels.
Multisensor Integration Fusion Model	Provides fault tolerance, survivability and robustness.	Multiple registrations increase signaling overhead.	√	√	√	High
Behavioural Knowledge-based Data Fusion Model	Provides survivability based on knowledge-based wisdom.	Multiple registrations increase signaling overhead.	√	√	√	High
Omnibus Model	Signals lifetime can be adjusted according to mobility pattern to reduce signaling overhead.	Frequent inter domain movement leads to the normal binding update and same signaling cost.	√	√	√	Low under restricted levels.
Dasarathy Model	Controlled monitoring can be done based on information integration.	Incurs extra cost for control monitoring	√	√	√	Low
JDL Model	Buffering of packets in each stage during data transmission.	Extra registration requires for prefix notification.	√	√	√	High

III. THE PROPOSED PARADIGM: COMMUNITY MODEL

The data fusion models discussed in the previous section are application specific and minimize the data transmission time in signaling overhead, power consumption, fault tolerance, survivability and robustness facilities. Our intention is to develop a new model which will be applicable in different application areas like radar signal processing, image processing, data mining, decision support system etc. The proposed Paradigm, Community Model (Fig 1) is aimed at developing a model which can be used irrespective of environments. Its different levels can ease out the burden from Master Fusion Filter by comparing the fused information using the reference sensor at each level. Here each sensor has multiple levels. These multiple levels were used for further refinement of the collected data from the sensors. As different filtering algorithms are used and executed at different filter levels, the fine-tuning of the data increases in manifold. As the reference sensor is feeding the data directly at each level, it is easier to compare the fused data with its actual value. At each level, all parallel filters compare their individual sensor data with a common reference data and calculates local system states. Then these local estimates are sent to the parallel filters of the next level which repeats the same process as the parallel filters in the previous level and it continues up to the Nth level of filter. After crossing all the levels, the local estimates from the last level are finally fused in master filter to get global estimates. Parallel filters have a common state vector as they all share common reference system. When the signals arrive to the filters at various levels, they are processed and transmitted.

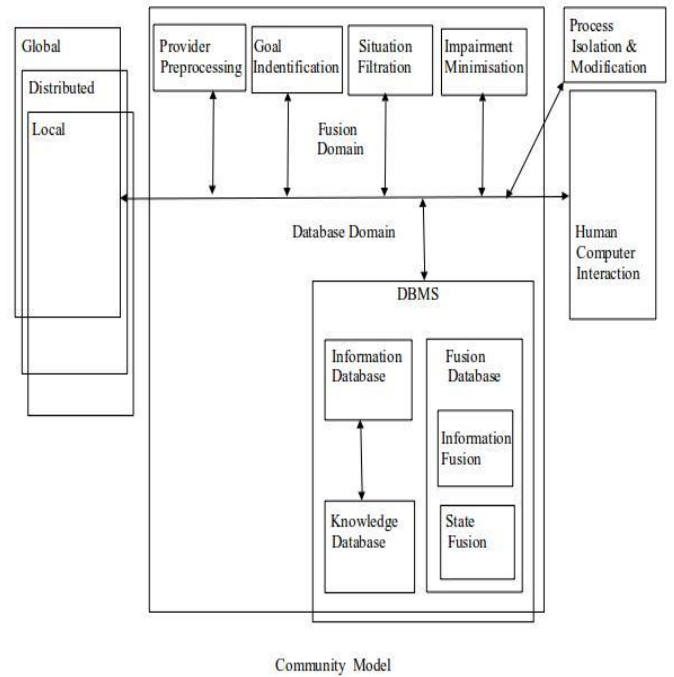


Figure 1 Community Model

A filter can process only one signal at a time. If signals arrive at a rate faster than the filter can process them (such as in a burst transmission) the filter puts them into the queue (also called the buffer) until it is transmitted. Here the transmission time is the composition of the time difference between each filter levels up to the maximum level as well as the final transmission to the Master Fusion Filter. During network congestion, transmission time can be considered infinite when the signal is dropped. The retransmission of such signals causes significant overall delay because all forms of delay will be incurred more than once. The proposed model and its detailed process flow are shown in fig 1 and 2 respectively.

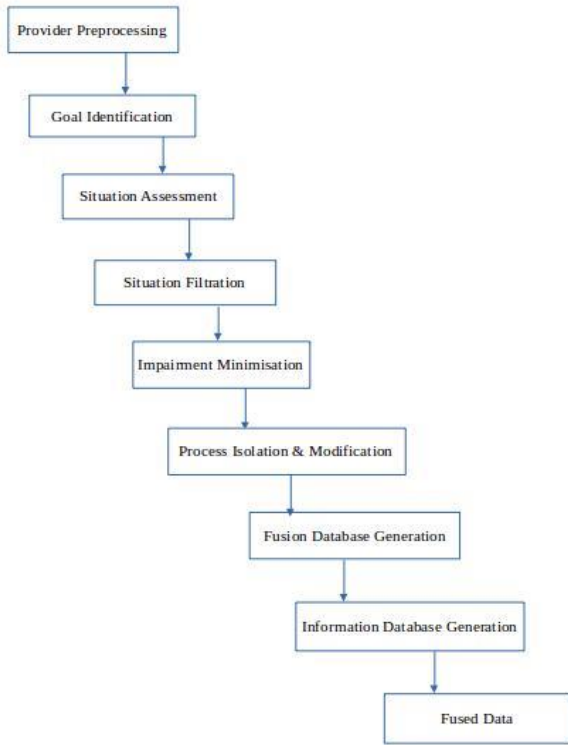


Figure 2 Community Model Flow-Graph

Related Terms

To compare different models, we require to find the DTT. This section discusses about the terms related to the formulation of DTT:

- (i) Perimeter of a data fusion filter means the area through which the data is passed.
- (ii) Data fusion density is a data fusion operator based on averaging that is weighted by the density of each particular data sample.
- (iii) Flattening is a measure of the compression of an object where the individual entries are collated to form the actual object.

(iv) Scale factor stability is a measure of how good the gyro output is with respect to rotation input. It actually deals with the difference between the different angular inputs for a specific data set.

(v) Alignment represents data points that lie on a relatively straight path.

(vi) Random noise is a noise characterized by a large number of overlapping transient disturbances occurring at random.

The uncertainty in the bias of a parameter consists of the uncertainty due to drift, the uncertainty due to random variations and due to uncertainty in testing.

Mathematical Formulation

This section deals with the formulation of DTT in terms of different parameters such as perimeter of a data fusion filter, data fusion density, flattening, scale factor stability, Alignment, random noise, bias uncertainty etc. which have been discussed in the previous section.

Let N = Total number of local filters, P_m = Perimeter of master fusion filter,

Hence, Perimeter of each local filter(P_n) will be -

$$P_n = N\sqrt{(P_m/N)^2} \quad (1)$$

If we consider, ρ = Data fusion density per LF, v = Average velocity of the light then the Crossing rate of different LF levels (R_n) is,

$$R_n = (\rho v P_n / \pi) \quad (2)$$

Considering, Flattening of the LF surface(F), the modified crossing rate is -

$$R_{nF} = (\rho v P_n / \pi) / F \quad (3)$$

Incorporating scale factor stability(S), equation (3) becomes, $R_{nFS} = S(\rho v P_n / \pi) / F$ (4)

Introducing random noise(N_R), the formulation of crossing rate will be -

$$R_{nFSR} = S(\rho v P_n / \pi) / F + N * N_R \quad (5)$$

Considering the alignment consideration as A , we get

$$R_{nFSRA} = 1/A [S(\rho v P_n / \pi) / F + N * N_R] \quad (6)$$

Taking the bias uncertainty(H) into account, we have

$$R_{nFSRAH} = 1/A [S(H * \rho v P_n / \pi) / F + N * N_R] \quad (7)$$

Considering the latency time as L_T we get,

$$R_{nFSRHL} = L_T * 1/A [S(H * \rho v P_n / \pi) / F + N * N_R] \quad (8)$$

So the transmission time upto N th level will be the summation of the crossing rate of all the individual levels. And hence the DTT (T_R) is -

$$T_R = \sum_1^N L_T * 1/A[S(H*\rho v P_n/\pi)/F + N*N_R] \quad (9)$$

Component Modules used in Community Model

The community model structure can be represented as a collection of different procedures following the model shown in figure 1 viz. Provider Preprocessing (PP), Goal Identification (GI), Situation Filtration (SF) and Impairment Minimization (IM). Here we are illustrating the different components of the fusion domain of figure 1 as the functions in pseudocode assuming that the database fusion domain information is available. Initially we call the procedures GI and SF and then we are initializing the data transmission time (T_R) as zero. In the next step we are taking the input values. Here we have three basic filtering criteria - local, distributed and global. If we choose the local level then the transmission of data is no longer required. But if we choose global or distributed state then we have to verify whether the filtered data matches with that of the reference sensor or not. If the data does not match, the values are updated using the reference sensor otherwise the values are taken as it is. After that the data transmission is calculated as in (9).

For the sake of simplicity we have modularized the community model procedure with the following components –

1) Provider Preprocessing (PP)

The procedure PP actually deals with the initialization of different parameters with specified values available in the literature [17], so that after each stage of filtering, they can be verified and if the value does not match with the required value then it is replaced. Here the data are inputted to the Reference Sensor (RS) so that they can be used to verify the data with the data that are available after each stage of filtration. It actually consists of a single stage which only takes the inputs against the standard parameters. Generically, the input parameters are defined through the $RS_{(i)}$. This (i) signifies the following –

A = Alignment consideration
S = Scale factor stability
H = Bias uncertainty
 ρ = Data fusion density per local filter
V = Average velocity of the light
P = Perimeter of local filter
N = Number of filter level

Thus the explicit input parameters are $RS_A, RS_S, RS_H, RS_\rho, RS_V, RS_P, RS_N$.

2) Goal Identification (GI)

The GI procedure is one of the most difficult steps related to the construction of the community model. Here the goal is to be determined based on the current scenario of the application area. Initially we have taken our goal to determine Data Transmission Time (DTT), though this goal may be

changed or appended based on the input parameters and objectives. This algorithm describes the procedure to calculate the data transmission time. Initially we input the standard parameter values [17] to the reference sensor so that they can be verified at a later stage. Then the DTT is initialized to zero. Then the values are inputted to the values to the procedure IM. If it is found that the data matches with the data already available with the reference sensor then it is calculated and if it does not match with the reference sensor values then they are updated and the total transmission time is calculated till the Nth level of filter is reached. Here T_R denotes the total data transmission time.

Procedure: GI

T_R = Data transmission time

D = Data for master fusion filter

δ = Impairment for Data

RS_i = Reference Sensor considering different parameters $RS_{(i)}$ like A= Alignment consideration, S= Scale factor stability, H=Bias uncertainty, ρ = Data fusion density per local filter, V= Average velocity of light, P = Perimeter of master fusion filter, N= Number of filter levels

1. Input $RS_A, RS_S, RS_H, RS_\rho, RS_V, RS_P, RS_N$
2. set $T_R \leftarrow 0$
3. input $L_T, A, S, H, \rho, V, P_n, P, N, N_R$
4. call IM
5. if ($\delta \neq \emptyset$) || D !=RS
6. update $A \leftarrow RS_A, S \leftarrow RS_S, H \leftarrow RS_H, \rho \leftarrow RS_\rho, V \leftarrow RS_V, P \leftarrow RS_P, N \leftarrow RS_N$
7. endif
8. $i \leftarrow 1$
8. while ($i \leq N$)
9. $T_R = T_R + L_T * 1/A[S(H*\rho v P_n/\pi)/F + N*N_R]$
10. $i \leftarrow i + 1$
11. end while
12. end procedure

3) Situation Filtration (SF) [11][14][18]

Here we use the Boyd Loop as the standard procedure. The following values from [18] are reiterated for completeness. For network synchronization among filter levels, Kuramoto's order parameter is applied separately to the each filter in the following way -

$$r(t)e^{i\Psi(t)} = 1/N \sum_1^N X_i(t).$$

The sum is taken for the N levels of filter. The values of r close to 1 over time interval indicate high synchronization. When the system is highly synchronized, the collective phase of the system or the centroid of the points for each oscillator on the unit circle are denoted by Ψ . This angle is calculated

$$\text{by } \Psi = \text{ArcTan} \left[\frac{\sum_1^N \sin \theta_i}{\sum_1^N \cos \theta_i} \right].$$

The synchronization parameters F1 and F2 are the perimeter of the first and second level filter respectively. Thus F1 is used for the first level and F2 is used for second level. The

adversarial performance is measured directly from the Kuramoto's order parameter -

$$\Psi(t) = \Psi_{F1}(t) - \Psi_{F2}(t)$$

Procedure: SF

1. Input $RS_A, RS_S, RS_H, RS_\rho, RS_V, RS_P, RS_N, \Psi$
2. call GI
3. if $(\delta \neq \emptyset) \parallel D \neq RS$
4. $\Psi(t) = \Psi_{F1}(t) - \Psi_{F2}(t)$
5. endif

4) *Impairment Minimisation (IM)*

Here, we consider impairment minimization function (δ), is a composition of three different parameters viz. attenuation (δ_A), distortion (δ_D) and noise (δ_N). If we consider the local level of filtration, the concept of impairment does not arise as the data are not traversed through filter levels. However, in case of global and distributed levels of filtration, the impairment does come into picture through the following formulas [19] -

$$\delta_A = \Psi.l.f$$

$$\delta_D = A.x(t-T)$$

$$\delta_N = E_b/N_0$$

If the signals are sufficiently strong then the attenuation (δ_A) minimizes, similarly if the frequency components of a signal arrive at the receiver end at the same time then the distortion (δ_D) is minimized and if there is no interfering signal available then the noise will be zero.

Procedure: IM

1. Input $RS_A, RS_S, RS_H, RS_\rho, RS_V, RS_P, RS_N, \Psi$
2. if $(\delta \neq \emptyset) \parallel D \neq RS$
3. $\delta_A = \Psi.l.f$
4. $\delta_D = A.x(t-T)$
5. $\delta_N = E_b/N_0$
6. endif

Moreover after each stage of filtration, the transmission time is saved to the database for future reference.

Algorithm for the Community Model

In this section the main algorithm of community model has been provided with the component modules – PP, GI, SF and IM discussed in the previous section.

1. Input $RS_A, RS_S, RS_H, RS_\rho, RS_V, RS_P, RS_N$
2. call GI
3. call SF (F1, F2)
4. set $T_R \leftarrow 0$
5. input $L_T, A, S, H, \rho, V, P_n, P, N, N_R$
6. if state ==Local
7. set $N \leftarrow 1$
8. end if

9. if $((\delta \neq \emptyset) \parallel D \neq RS)$
10. update $A \leftarrow RS_A, S \leftarrow RS_S, H \leftarrow RS_H, \rho \leftarrow RS_\rho, V \leftarrow RS_V, P \leftarrow RS_P, N \leftarrow RS_N$
11. endif
12. $i \leftarrow 1$
13. while $(i \leq N)$
14. $T_R = T_R + IM(A, S, H, \rho, V, P, N)$
15. insert singly linked list with header
16. memoisation (h)
17. $i \leftarrow i + 1$
18. end while
19. end procedure

In line 16, the function called memoization keeps track of already computed values and avoids re-computation

IV PERFORMANCE ANALYSIS

Transmission Time is one of the difficult steps in data fusion filtering. It consists of identifying and correlating noisy measurements, the genesis of which are unidentified because of several inescapable situations. The key models used in this fields are either deterministic (based on Classical Hypothesis), or probabilistic models (based on Bayesian Hypothesis) . The values taken here as the standard parameters are bias uncertainty = 01/10/40, scale factor stability = 100-500, alignment = 200, random noise =01/05/13, flattening = 1.298.257223563, latency time = 10 [17]. We found that our formulation yields better result than that of Waterfall Model, Multisensor Integration Fusion Model, Behavioural Knowledge-based Data Fusion Model, Omnibus Model, Dasarathy Model and JDL Model. The result is shown in figure 3 using MATLAB.

The DTT of each of these models has been calculated with the standard values using the respective formula and the result for level 1 are reflected in the following table -

TABLE II. DTT VALUES

Model	Data Transmission Time
Waterfall Model	511
Multisensor Integration Fusion Model	459
Behavioural Knowledge - based Data Fusion Model	436
Omnibus Model	441
Dasarathy Model	443
JDL Model	439
Community Model	389

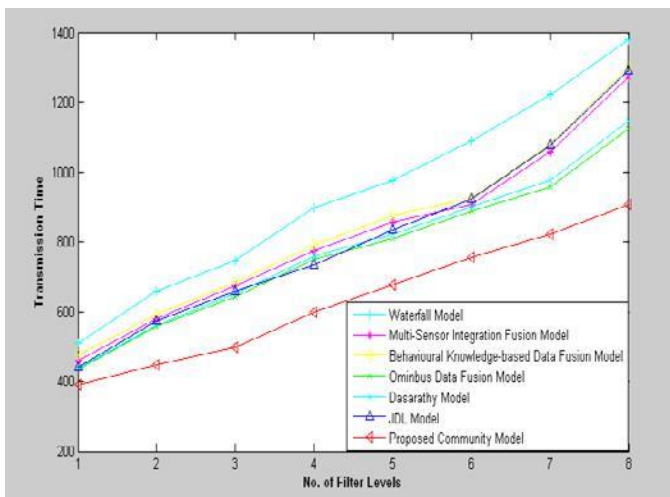


Fig 3 Comparison of DTT

V. CONCLUSION

This paper analyses signaling overhead, power consumption, fault tolerance, survivability and robustness facilities among different models and compares the community model with respect to Transmission Time. The Community Model tries to reduce the Transmission Time to improve the performance of Data Fusion in information processing. More over the transmitted data is much more accurate as they are compared with the reference sensor in each stage. The model gives better result in terms of data transmission time and we are currently studying the aspects related to transmission overhead.

References

- [1] I. Edwards, X. E. Gross, D. W. Lowden, and P. Strachan, "Fusion of NDT Data," *British Journal of Non-Destructive Testing*, Vol. 35 No.12 pp. 710-713, 1993.
- [2] I. Ayari and J. P. Haton, "A Framework for Multi-Sensor Data Fusion," *Proceedings of IEEE Symposium on Emerging Technologies and Factory Automation*, 2 pp. 51-59, 1995.
- [3] R. J. Linn and D. L. Hall, "A Survey of Data Fusion Systems," *Proceedings of SPIE Conference on Data Structure and Target Classification*, pp. 13-36, 1991.
- [4] F. Cremer, E. den Breejes and S Klamer, "Sensor Fusion for Anti-personnel Land Mines Detection," *Proceedings. of 3rd Eurofusion Conference*, pp. 63-70, October 1998.
- [5] C. J. Harris, A. Bailey and T. J. Dodd, "Multi-sensor Data Fusion in Defence and Aerospace," *Aeronautical Journal* 102 (1015) pp. 229-244, May 1998.
- [6] L. Bruzzone, D. Fernandez and G. Vernazza "Data Fusion Experience: From Industrial Visual Inspection to Space Remote-sensing Application," *Proceedings of Academic and Industrial Cooperation in Space research*, pp. 147-151, November 4-6, 1998.
- [7] L. Nigay and J. Coutaz "A Generic Platform for Addressing the Multimodal Challenge," *Proceedings of ACM-CHI'95*, pp. 98-105, May 1995.
- [8] D. J. Sentinella and A. G. Raines, "Real time data fusion(road traffic strategic control)," *IEE Colloquium on Strategic Control of Inter-Urban Road Networks (Digest No. 1997/055)*, pp. 5/1-5/3, March 1997.
- [9] J. Schoess and G. Castore, "A Distributed Sensor Paradigm for Advanced Aerospace Systems," *Proceedings of SPIE Sensor Fusion*, Vol. No. 0931, pp. 74-86, August 1988.
- [10] L. F. Pau, "Behavioral knowledge in sensor/data fusion systems," *Journal of Robotic Systems*, Vol. 7 Issue 3, pp. 295-308, June 1990.
- [11] J. Esteban, A. Starr, R. Willetts, P. Hannah, P. Bryanston-Cross "A Review of Data Fusion Models and Paradigms: Towards Engineering Guidelines," *Journal Neural Computing and Applications Volume 14 Issue 4*, pp. 273-281, December 2005.
- [12] M. D. Bedworth and J. C. O'Brien, "The Omnibus Model: A New Model of Data Fusion?," *Aerospace and Electronic Systems Magazine*, IEEE, Vol. 15 Issue 4, pp. 30-36, April 2000.
- [13] R. Johansson "Information Acquisition in Data Fusion Systems," TRITA-NA-0328 Licentiate Thesis Royal Institute of Technology Department of Numerical Analysis and Computer Science, Stockholm, Sweden, November 2003.
- [14] W. Elmenreich, "A Review on System Architectures for Sensor Fusion Applications," *Lecture Notes in Computer Science*, Springer, Vol. 4761, pp. 547-559, 2007.
- [15] R. C. Luo, C. C. Chang and C. C. Lai "Multisensor Fusion and Integration: Theories, Applications, and its Perspectives" *IEEE Sensors Journal*, Vol. 11 No. 12, pp. 3122-3138, December 2011.
- [16] E. F. Nakamura, A. A. F. Loureiro and A. C. Frery, "Information Fusion for Wireless Sensor Networks: Methods, Models, and Classifications," *ACM Computing Surveys*, Vol. 39 No. 3, August 2007.
- [17] C. R. Spitzer (Edited), "The Avionics Handbook," CRC Press LLC, 2001.
- [18] A. Kalloniatis, "On the 'Boyd-Kuramoto Model': Emergence in a Mathematical Model for Adversary C2 Systems", 17th ICCRTS, The Mason Inn in Fairfax, VA, June 19-21, 2012.
- [19] Y. Huang, J. P. Heritage, and B. Mukherjee "Connection Provisioning With Transmission Impairment Consideration in Optical WDM Networks With High-Speed Channels" *IEEE Journal of Lightwave Technology*, Vol. 23 No. 3, March 2005.