# Fuzzy Methodologies for Multi-Sensor Information Fusion with Applications to Precision PNT

James A. Crowder and John N. Carbone,
Raytheon Intelligence, Information, and Services, Aurora, CO, USA
Raytheon Intelligence, Information, and Services, Garland, TX, USA

***Abstract***— Providing processing for intelligence operations requires collecting, sorting, and fusing data from a variety of sources to produce coherent and correlated intelligence "information" (Multi-Sensor Data Fusion). Multi-sensor data fusion is an evolving technology, concerning the problem of how to fuse data from multiple sensors in order to make a more accurate assessment of a given situation or environment. Applications of data fusion cross a wide spectrum, including automatic target detection and tracking, battlefield surveillance, remote sensing, etc. They are usually time-critical, cover a large geographical area, and require reliable delivery of accurate information for their completion. One of the big problems with multi-sensor fusion is the Level 0 processing: A new approach to fusion is the joint Mutual Information between the features and the class labels. It can be shown that Mutual Information minimizes the lower bound of the classification error. However, according to Shannon's definition this is computationally expensive.

Evaluation of the joint Mutual Information of a number of variables is plausible through histograms, but only for a few variables. If we look toward a different definition of Mutual Information we find a different result. Using Renyi's entropy instead of Shannon's, combined with Parzen density estimation, leads to expression of Mutual Information with significant computational savings. Here, we will extended Renyi's method for Mutual Information to multiple continuous variables and discrete class labels to learn linear dimension-reducing linear feature transforms for data fusion and parameter estimation utilizing competing parameter measures.

***Keywords—Fuzzy Logic, Mutual Information, Information Fusion***

## 1. Introduction: Multi-sensor fusion

Multi-sensor data fusion/integration is an evolving technology, concerning the problem of how to fuse data from multiple sensors in order to make a more accurate measurement of the environment. Applications of data fusion cross a wide spectrum, including automatic target detection and tracking, battlefield surveillance, remote sensing, etc. They are usually time-critical, cover a large geographical area, and require reliable delivery of accurate information for their completion [9]. According to the Office of Naval Research:

*"Sensor integration is concerned with the synergistic use of multiple sources of information. In warfare, no one piece of information can be accepted as complete truth. The combination of information from every possible source is of primary importance."*

Sensor fusion/integration is divided into three classes: complimentary sensors, competitive sensors, and cooperative sensors [8]:

**Complimentary sensors** do not depend on each other directly but can be merged to form a more complete picture of the environment, for example, a set of radar stations covering non-overlapping geographic regions. Complementary fusion is easily implemented since no conflicting information is present.

**Competitive sensors** each provide equivalent information about the environment. A typical competitive sensing configuration is a form of N-modular redundancy. For example, a configuration with three identical radar units can tolerate the failure of one unit. This is a general problem that is challenging, since it involves interpreting conflicting readings.

**Cooperative sensors** work together to drive information that neither sensor alone could provide. An example of cooperative sensing would be using two video cameras in stereo for 3D vision. This type of fusion is dependent on details of the physical devices involved and cannot be approached as a general problem.

Here we attack the problem of real-time, distributed, competitive sensor fusion for time-critical sensor readings. Figures 1 and 2 depict sensor fusion scenarios for this study. It is assumed that each sensor platform has some local intelligence and memory [3]. We also assume that every sensor has limited accuracy and that a limited number of readings may be arbitrarily faulty, each $m_i$ uses possibly different logic to deduce the position, velocity, and parametric measurements of the object under surveillance." Once the information is transmitted to the central processing system, fusion and situational awareness software are used to provide an overall picture of the battlefield to the war fighter. Figure 3 below illustrates a block diagram of an overall fusion and situational awareness processing system. One of the big problems with multi-sensor fusion is the Level 0 processing shown below. This involves putting the various sensors into a classification system where the sensors can be evaluated against each other. i.e., how to eliminate the differences between the sensors so the information content of each can be fused into intelligent information with error bounds consistent with the various information sources [13].

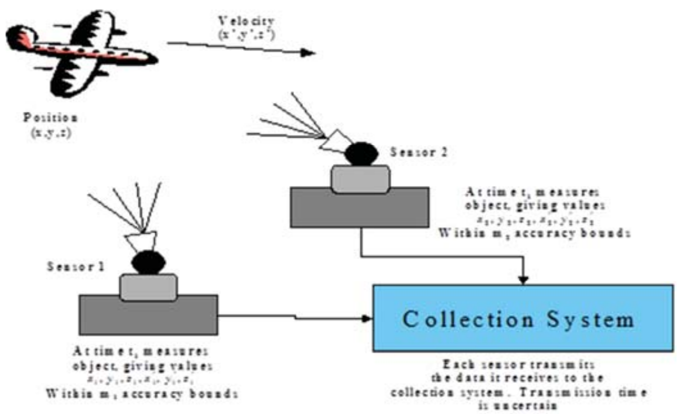



*Figure 1 - Competitive Ground Sensors*

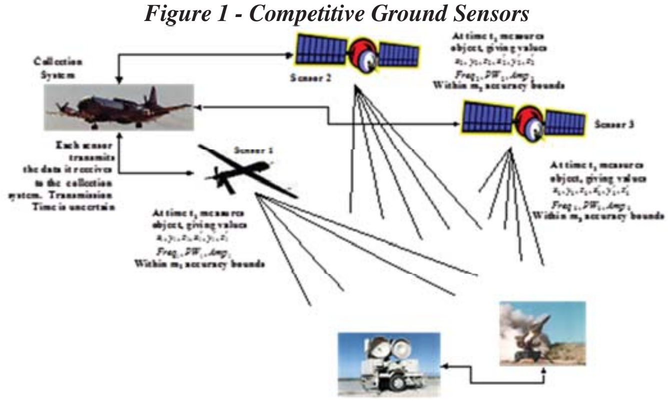



*Figure 2 - Competitive Air and Space Sensors*

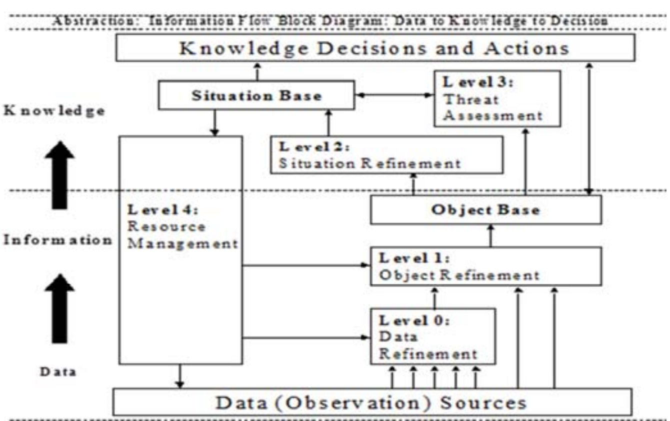



*Figure 3 - Data/information flow for situational awareness*

The process of data fusion requires a large number of disciplines including signal and image processing, control theory, database design, networks, data standards, as well as human computer interface. Military research applicable to data fusion is in the areas of Intelligence Surveillance and Reconnaissance (IRS) sensors, Command and Control (C2), Communications (C), and Computers, which collectively make up C4ISR. This paper is concerned with Level 0 (Data Refinement) for Competitive Sensors; how to provide a normalization environment to fuse various sensor information so that the overall intelligence and situational awareness processing system can ingest, process, and report on large volumes of disparate intelligence information.

## 2. JOINT ENTROPY

Since the information coming into a given processing system may have from 1-to-n inputs at any given time, and each sensor that provides input will have random errors associated with any given measurement. The data streams coming into the processing systems may be seen as systems of random data, or each data stream represents a random variable to the system. We will start with a random pair of variables, ***(X,Y)***. Another way of thinking of this is as a vector of random variables.

**Definition 1**: If ***X*** and ***Y*** are jointly distributed according to ***p(X, Y)***, then the Joint Entropy ***H(X, Y)*** is (**EQ 1**):

$$H(X,Y) = -\sum_{x\in X}\sum_{y\in Y} p(x,y)\log\ p(x,y)$$

*or*

$$H(X,Y) = -E\ \log\ p(x,y)$$

**Definition 2**: if ***(X, Y) ~ p(x, y)***, then the conditional entropy ***H(Y|X)*** is (**EQ 2**):

$$H(Y\mid X) = -\sum_{x\in X}\sum_{y\in Y} p(x,y)\log p(y\mid x) = -E_{p(x,y)}\log p(y\mid x)$$

This can also be written in the following equivalent ways:

$$H(Y\mid X) = -\sum_{x\in X} p(x)\sum_{y\in Y} p(y\mid x)\log p(y\mid x) =$$

$$-\sum_{x\in X} p(x) H(Y\mid X = x)$$

**Theorem 1**: (Chain Rule)

$$H(X\mid Y) = H(X) + H(Y\mid X)$$

The uncertainty (entropy) about both ***X*** and ***Y*** is equal to the uncertainty (entropy) we have about ***X***, plus whatever we have about ***Y***, given that we know X. This can also be done in the following streamlined manner: Write (**EQ 3** Joint Entropy)

$$\log p(X,Y) = \log p(X) + \log p(Y\mid X)$$

and take the expectation of both sides. We can also have a joint entropy with a conditioning on it, as how below:

**Corollary 1**:

$$H(X,Y\mid Z) = H(X\mid Z) + H(Y\mid X,Z)$$

### *2.1 Relative Entorpy and Mutual Informaiton*

Suppose there is a random variable with true distribution *p*. Then (as we will see) we could represent that random variable with a code that has average length *H*(*p*). However, due to incomplete information we do not know *p*; instead we assume that the distribution of the random variable is *q*. Then (as we will see) the code would need more bits to represent the random variable. The difference in the number of bits is denoted as *D*(*p*|q). The quantity *D*(*p*|q) comes up often enough that it has a name: it is known as the **relative entropy**.

**Definition 3**:The relative entropy or **Kullback-Leibler** distance between two probability mass functions ***p(x)*** and ***q(x)*** is defined as (**EQ 4**):

$$D(p \parallel q) = \sum_{x \in X} p(x) \log \frac{p(x)}{q(x)} = E_p \log \frac{p(X)}{q(X)}$$

It should be noted that this is **not** symmetric, and that ***q*** (second argument) appears only in the denominator.

Another important concept is Mutual Information [2]. This describes how much information on random variable tells about another one. This is, perhaps, the central idea in information theory. When we look at the output of a sensor, we see the information as a random variable. What we want to know is what was sent, and the only information we have is what cam out of the sensor. Or, we have two sensors that are each random variables, both providing information about the same thing. We know that the information should be the same, but it is not because of a host of errors that the sensors and the transport medium introduced into the measurements. What we want is to extract the exact information from the sensor readings. Or, in other words, we want to find the mutual information between the sensor two different sensor readings.

**Definition 4**: Let ***X*** and ***Y*** be random variables with joint distribution ***p(X,Y)*** and marginal distributions ***p(x)*** and ***p(y)***. The **Mutual Information *I(X;Y)*** is the relative entropy between the joint distribution and the product distribution (**EQ 5**):

$$I(X;Y) = D(p(x,y) \parallel p(x)p(y))$$

$$= \sum_x \sum_y p(x,y) \log \frac{p(x,y)}{p(x)p(y)}$$

Note that if ***X*** and ***Y*** are truly independent then ***p(x,y) = p(x)p(y)*** so ***I(X;Y) = 0***. However, if they are sensing the same thing, then they should not truly be independent. An important interpretation from Mutual Information comes from the following theorem:

**Theorem 2**:

$$I(X;Y) = H(X) - H(X \mid Y)$$

The interpretation of this is that the information that ***Y*** tells us about ***X*** is the reduction in uncertainty about ***X*** due to the knowledge of ***Y***. The information ***X*** tells about ***Y*** is the uncertainty in ***X*** plus the uncertainty about ***Y*** minus the uncertainty in both ***X*** and ***Y***. We can summarize a bunch of statements about entropy as follows [10]:

$$I(X;Y) = H(X) - H(X \mid Y)$$
$$I(X;Y) = H(Y) - H(Y \mid X)$$
$$I(X;Y) = H(X) + H(Y) - H(X,Y)$$
$$I(X;Y) = I(Y;X)$$
$$I(X;X) = H(X)$$

## 3. Fuzzy Data Normalization and fusion

Figure 4 represents a processing flow for intelligence information. The process involves two main layers, the deductive process and the investigative process. The deductive process goes after assembling information that has been previously known while the inductive process (data mining) looks for patterns and associations that have not been seen before. The model illustrated in Figure 3-1 is the deductive process used to detect previously known patterns in many sources of data by searching for specific information signatures and templates in data streams to understand the state of the intelligence knowledge [16]. As the systems continues to evolve in complexity, the number of objects, situations, threats, sensors and data streams dramatically increase, presenting a very complex challenge for advanced fusion system designers. In order to keep the system "on-top" of its data environment is to have data mining operations going on in the background at all times, finding new associations and evolving the templates and information correlations [6].

Data Mining is an off-line knowledge creating process where large sets of previously collected data is filtered, transformed, and organized into information sets. This information is used to discover hidden but previously undetected intrusion patterns. Data mining is called knowledge/pattern discovery and is distinguished from the data fusion process by two important characteristics, inference method and temporal perspective [7]. Data fusion uses known templates and pattern recognition. Data mining processes search for hidden patterns based on previously undetected intrusions to help develop new detection templates. In addition, data fusion focuses on the current state of information and knowledge; data mining focuses on new or hidden patterns in old data to create previously unknown knowledge, illustrated in Figure 4 [11].

In both data mining and data fusion, feature selection or feature transforms are important aspects of any system. Optimal feature selection coupled with pattern recognition leads to a combinatorial problem since all combinations of available features must be evaluated before deciding how to fuse the information available.

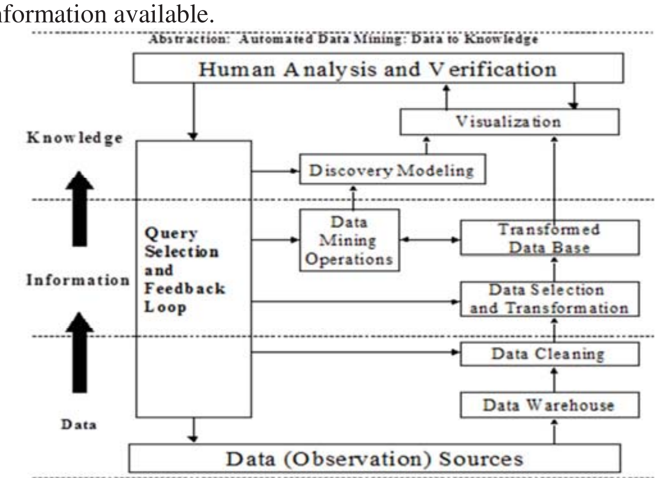



*Figure 4 - Data/Information Flow for Data Mining Operations*

Another such criterion is the joint ***Mutual Information*** between the features and the class labels. It can be shown that Mutual Information minimizes the lower bound of the classification error. However, according to Shannon's definition this is computationally expensive. Evaluation of the joint Mutual Information of a number of variables is plausible through histograms, but only for a few variables. If we look toward a different definition of Mutual Information we find a different result. Using Renyi's entropy instead of Shannon's, combined with Parzen density estimation, leads to expression of Mutual Information with significant computational savings. As a part of this study, we extended Renyi's method for Mutual Information to multiple continuous variables and discrete class labels to learn linear dimension-reducing linear feature transforms for data fusion and parameter estimation utilizing competing parameter measures [5].

We applied Renyi's entropy-based Mutual Information measure to create fuzzy membership functions that can be used to rapidly asses the Mutual Information content between multiple measurements of a given parameter from different sensors [1]. We introduce the Mutual Information measure based on Renyi's entropy, and describe its application to Fuzzy Membership Functions that were used transform multiple parameter measures and error estimates into a single parameter and error bound estimate for the parameter.

### *3.1 Shannon's Definition of Mutual Informaiton*

We denote labeled samples of continuous-valued random variable ***Y*** as pairs $\{y_i, c_i\}$, where $y_i \in R^d$, and class labels are samples of a discrete-valued random variable *C*, $c_i \in \{1,2,\ldots,N_c\}, i \in [1, N]$. If we draw one sample of ***Y*** at random, the entropy or uncertainty of the class label, making use of Shannon's definition, is defined in terms of class prior probabilities (**EQ 6 – Shannon's Entropy Theory**) [4]:

$$H(C) = -\sum_c P(c)\log(P(c))$$

After having observed the feature vector y, our uncertainty of the class identity is the conditional entropy is (**EQ 7**):

$$H(C) = \int_y p(y)\left(\sum_c p(c/y)\log(p(c/y))\right)dy$$

The amount by which the class uncertainty is reduced after having observed the feature vector *y* is called the Mutual Information, $I(C,Y) = H(C) - H(C/Y)$, which can be written as (**EQ 8**):

$$I(C/Y) = \sum_c \int_y p(c,y)\log\frac{p(c,y)}{P(c)p(c)}dy$$

Mutual information also measures independence between two or more variables, in this case between ***C*** and *Y*. It equals zero when $p(c,y) = P(C)p(y)$, i.e., when the joint density of ***C*** and ***Y*** factors (the condition for independence). Mutual Information can thus be also viewed as the divergence between the joint densities of the variables, and the product of the marginal densities. Connection between Mutual Information and Data Fusion is given by Fanno's inequality.

This result, originally from digital communications, determines a lower bound to the probability of error when estimating a discrete random variable ***C*** from another variable *Y* (**EQ 9**):

$$\Pr(c \neq \hat{c}) \geq \frac{H(C/Y)-1}{\log(N_c)} = \frac{H(C)-I(C/Y)-1}{\log(N_c)}$$

where $\hat{c}$ is the estimate of ***C*** after observing a sample of *Y*, which can be a scalar or multivariate. Thus the lower bound on error probability is minimized when Mutual Information between ***C*** and ***Y*** is maximized, or, finding such features achieves the lowers possible bound to the error of the classifier. Whether this bound can be reached or not, depends of the goodness of the classifier.

### *3.2 A Definition Based on Renyi's Entropy*

Instead of Shannon's entropy, we apply Renyi's quadratic entropy because of its computational advantages. For a continuous variable ***Y***, Renyi's quadratic entropy is defined as (**EQ 10**):

$$H_R(Y) = -\log\int_y p(y)^2 dy$$

It turns out that Renyi's measure, combined with the Parzen density estimation method using Gaussian kernels, provides significant computational savings, because a convolution of two Gaussians is still a Gaussian.

If the density $p(y)$ is estimated as a sum of symmetric Gaussians, each centered at a sample $y_i$ as:

$$p(y) = \frac{1}{N}\sum_{i=1}^{N} G(y - y_i, \sigma I)$$

then it follows that the integral above equals (**EQ 11**):

$$\int_y p(y)^2 dy =$$

$$= \frac{1}{N^2}\int_y\left(\sum_{k=1}^{N}\sum_{j=1}^{N} G(y - y_k, \sigma I)G(y - y_j, \sigma)\right)dy$$

$$= \frac{1}{N^2}\sum_{k=1}^{N}\sum_{j=1}^{N} G(y_k - y_j, 2\sigma I)$$

Thus, Renyi's quadratic entropy can be computed as a sum of local interactions as defined by the kernel, over all pairs of samples. In order to make use of this convenient property, we make use of fuzzy membership functions and the natural way they demonstrate local interactions to find a function which maximized Mutual Information among sensor measurements.

### *3.3 A Maximizing Mutual Information*

In any real-time system, data arrives at the input as a random variable, since it cannot be known a' priori what data may or may not be received at any given time. This is particularly true of the type of system radar environment represent. In fact, each measurement is a random variable *X*, with a σ bound determined by the system dynamics. Since each of the sensors acts as a random variable we are looking to maximize Mutual Information in order to find the normalization that produces the best overall result.

If we assume each sensor measurement to be a random variable $\boldsymbol{x}$, with its associated s bound, we can form a fuzzy membership distribution function around each measurement using the measurement as the mean (best guess with the information given) and the given error bound as the membership function bounds.In order to maximize Mutual Information across the measurements, we need to find the random variable $\boldsymbol{x_i}$, which minimizes the distance between random variables. We accomplish this by mapping the each measurement value onto each of the fuzzy membership functions [12]. We compute fuzzy membership curves for each sensor, based on measurement and □□ populate each with measurements from all sensors. Figure 5 illustrates 12 sensor measurements on the same system, each reporting a slightly different RF. Each system would have its own measurement error, as shown in Figure 6.

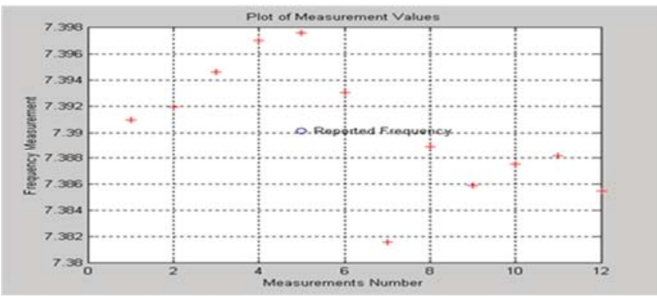



***Figure 5 - Example of Multiple Sensor Measures of the Same Frequency***

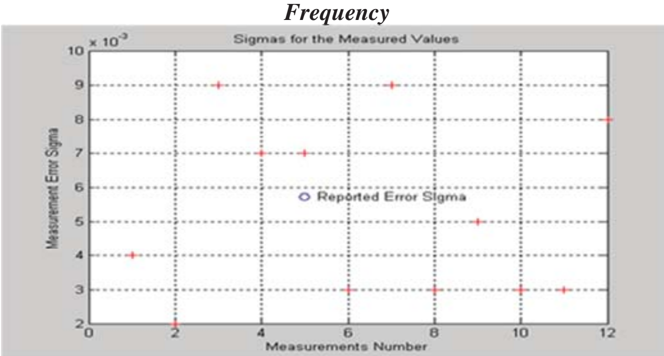



***Figure 6 - Examples of Multiple Sensors, each with their own Error Bounds***

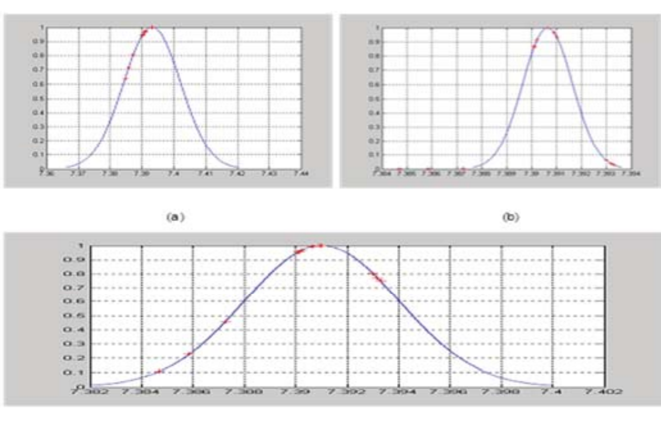



***Figure 6 - Examples of Fuzzy Membership Functions for Sensor Measurements, Populated with all the Sensor Measurements***

For each sensor, a fuzzy membership normalization function is formed and then each sensor measurement is mapped onto each membership normalization function (**EQ 12**):

$$Y_i = E_{j=1}^{n}\left( e^{\frac{-(M_i - M_j)^2}{2*\sigma_i^2}} \right)$$

Figure 7 a-c illustrates this process

Once all of the curves have been populated, we compute the mean fuzzy membership value for each curve (**EQ 14**):

$$Y_i = E_{j=1}^{n}\left( e^{\frac{-(M_i - M_j)^2}{2*\sigma_i^2}} \right)$$

$$Y_{\max} = \max_{i=1,n} (Y_i)$$

The normalization function with the highest mean membership represents the normalization mapping with the highest Mutual Information and is therefore given the highest weighting in determining the measurement value to report. The weighting factors are then determined for rolling up the measurements and error bounds into a single parametric estimation (EQ 15):

$$W_i = Y_i \Big/ \sum_{j=1}^{n} Y_j$$

where the $\boldsymbol{W_i}$*s* are the weighting factors. Figure 7 illustrates the process.

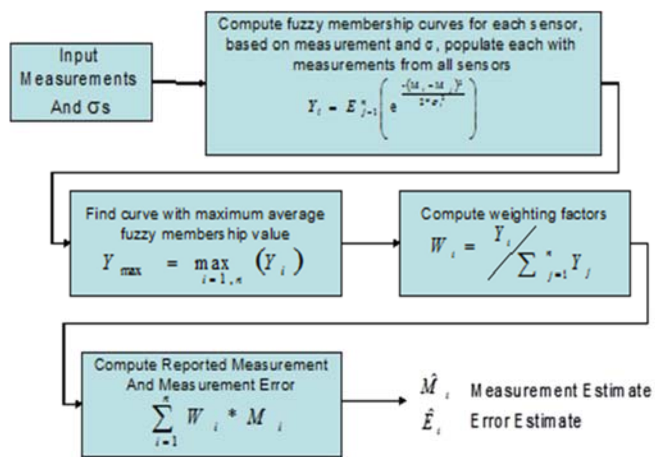



***Figure 7 - Weighted Fuzzy Parameter Estimation Process***

## 4. Conclusions and Discussion

Renyi's theory of information is extremely important in intelligence work, much more so than its use in cryptography would indicate. The theory can be applied by intelligence agencies to keep classified information secret, and to discover as much information as possible about an adversary. His fundamental theorems lead us to believe it is much more difficult to keep secrets than it might first appear. In general it is not possible to stop the leakage of classified information, only to slow it.

Furthermore, the more people that have access to the information, and the more those people have to work with and belabor that information, the greater the redundancy of that information becomes. It is extremely hard to contain the flow of information that has such a high redundancy. This inevitable leakage of classified information is due to the psychological fact that what people know does influence their behavior somewhat, however subtle that influence might be.

The premier example of the application of information theory to covert signaling is the design of the Global Positioning System signal encoding. The system uses a pseudorandom encoding that places the radio signal below the noise floor. Thus, an unsuspecting radio listener would *not even be aware that there was a signal present*, as it would be drowned out by atmospheric and antenna noise [15]. However, if one integrates the signal over long periods of time, using the "secret" (but known to the listener) pseudorandom sequence; one can eventually detect a signal, and then discern modulations of that signal. In GPS, the C/A signal has been publicly disclosed to be a 1023-bit sequence, but the pseudorandom sequence used in the P(Y) signal remains a secret. The same technique can be used to transmit and receive covert intelligence from short-range, extremely low power systems, without the enemy even being aware of the existence of a radio signal. We believe the use of Renyi's information theory might enhance these capabilities over the original work by Shannon [14].

The discussion below illustrates one possible use of Renyi's Information theory and the use of Fuzzy Filters to implement this theory for asynchronous data fusion for PNT estimations for Unmanned Air and Underwater vehicles (UAVs and AUVs). The fuzzy estimator performs asynchronous data fusion of all sensor measurements based on their relative confidence levels, and then nonlinearly combines the fused information with the INS estimates via fuzzy implementation of Renyi's mutual information theory. The basis and implementation of the estimator is described, and navigation results are presented based on the fuzzy estimator. We believe a fuzzy normalization procedure similar to the one outlined here provides the best automated, and dynamic way to roll-up sensor measurements into a single reported measurement and error bound for intelligence reporting and analysis. Section B presents the results of a number of tests for this process.

### *4.1 Fuzzy Data Fusion for UAV/AUV PNT Estimates*

Figure 15 illustrates the results of utilizing the Stochastic Derivative algorithms on the pulse environments described above. The Higher-Order moments of the 1st 8 Stochastic Derivatives were generated and plotted for each of the signal environments. As can be seen from 15, even 15 pulses from a pseudorandom-driven, multiply-agile radar signal caused a jump in the Stochastic Derivative moments. And 300 pulses out of 10,000 cased a major jump in the Higher Order Stochastic Derivative moments. Clearly the algorithms can detect the presence on non-stochastic signals in the environment.

With rapid progress in COTS sensors and electronics technology, miniaturized Autonomous Underwater Vehicles (AUV) and Unmanned Air Vehicles (UAV) have reached an acceptable level of maturity and reliability that can be capitalized on their use for commercial and military applications. Examples include spatio-temporal surveys during clandestine oceanographic and air-surveillance environments. Without requiring any tethering support, the dynamic stability and data sampling quality can be much improved. In addition, multiple small AUVs or UAVs can be deployed simultaneously to traverse in different regions without necessitating one-to-one support contact and allows higher data sampling efficiency. To truly characterize four-dimensional ocean or air dynamics autonomously, high precision underwater and air navigation is a technically challenging issue because vehicle localization must be done onboard. While differential GPS sensor technology provides a good solution for surface navigation, underwater and complex air navigation still remains a challenging problem especially for autonomous vehicles. The main goal here is to present a novel fuzzy data fusion that collates different and independent asynchronous position sensors together, and nonlinearly gain schedules them with the onboard INS system. One important objective here is to evaluate the possible use of the fuzzy implementation of mutual information theory for autonomous navigation and look at the effectiveness of the fuzzy sensor-based fusion approach with respect to steady-state and convergence performance of bias estimation.

### *4.2 Fuzzy PNT Fusion Architecture*

Figure 8 shows a general architecture for setting up a navigational. In this architecture, there are arbiters created for position sensors, attitude sensors and motion sensors. This set up is desirable because many existing AUVs and UAVs incorporate multiple sensors performing same functions, and it is thus beneficial to fuse all information to obtain the best navigation estimates. In cases of sensor failure, these arbiters will reconfigure in order to complete time-critical missions. Inputs to the position sensor arbiter are absolute position measurements, which can be based on (D)GPS and various forms of baseline sensors. These measurements are particularly valuable because of their drift-free properties over a longer time scale, as compared to the dead-reckoning position estimate. However, over a shorter time scale, the DGPS measurements introduce undesirable position error for compensation. It is thus important to carefully sample these so that the signal-to-noise ratio is maximized. In addition, measurements from all sensors are generally unavailable at every time sampling instant, and a strategy is thus needed to combine these asynchronous measurements before routing the result to the position estimator. Since the number of sensors may not be constant one data point to the next, the real-time fuzzy fusion techniques discussed here are applicable to this problem.

To combine the position measurements, each sensor is assigned a confidence value that characterizes its expected variance in error about the true value. For an example, typical DGPS can have 1-5m range error (σ) together with horizontal dilution of precision (HDOP) uncertainty due to satellite geometry. The total position error introduced in terms of root-mean-square value is HDOP *σ. It should be noted that while the range error is generally hard to quantify, the HDOP profile can be readily obtained from any receiver, and thus it should be accounted for in the position estimator otherwise position error might be compromised. Once a suitable time sampling interval is chosen, the output of the position sensor arbiter is given as

$X = \sum Z_i C_i$ where X is the arbiter output, $Z_i$ is the *ith* sensor output, and $C_i$ is its confidence value which has accounted for both the range error and baseline geometry of satellites or sonar beacons.

Note that a constraint of $\sum_{i=1}^{N} C_i \equiv 1$ is imposed on the confidence values and $C_i$ is then loosely interpreted as the probability that the ith sensor is correct.

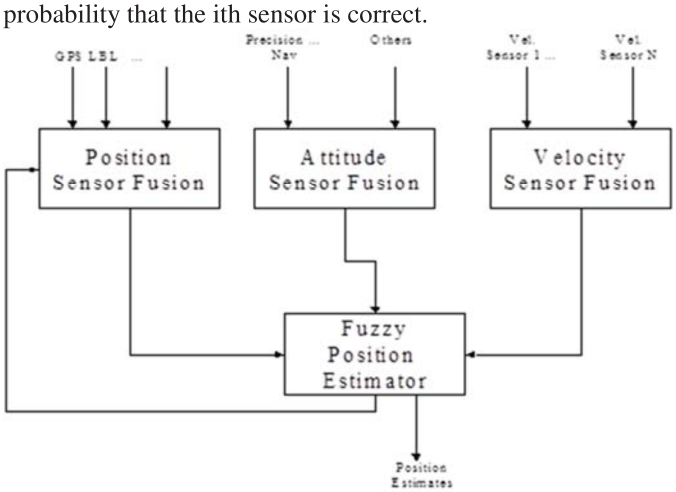



***Figure 8 - High-Level Fusion Architecture for PNT Estimation***

The use of the fuzzy fusion algorithm discussed here provides a practical estimation algorithm that is not computational intensive but yet provides theoretically sound approach in performing data fusion. Figure 9 below illustrates the GPS X-Y error evaluation for the test.

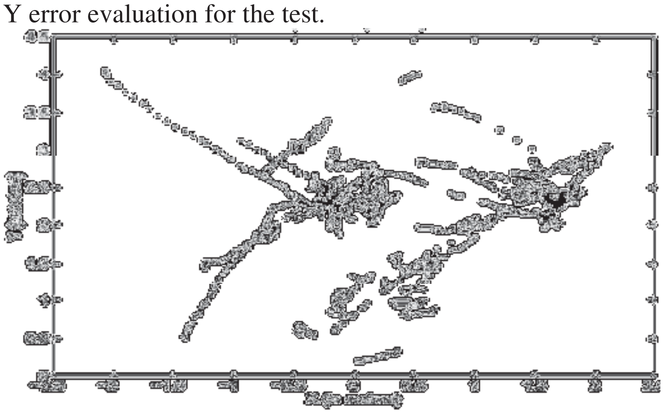


***Figure 9 – GPS Error Evaluation***

Figure 10 compares the INS way-point navigation performance based on 1) doppler returns (dash line) and 2) GPS + doppler (solid line). In these figures, `X' and `O' represent a differential and regular GPS fix respectively, and in both missions the AUV was started at the origin. During these missions, the AUV was underwater most of the time, and commanded to surface during some specified cornerings in order to obtain fixes, and thus position drift due to Doppler and attitude sensors can be easily observed, as compared to the DGPS fixes as the only source of reference. At the end of the last eastward leg in Figure B3, there was a significant discrepancy between the position estimator and DGPS measurement (approximately 1.5% error based on 50 meters after 3300 meters transect). Figure B3 presents results of a 3-hr mission covering 15km transect. Sporadic fixes can be seen in the figure which corresponds to the AUV surfacing maneuvers. Among these fixes, maximum discrepancy between the position estimator and GPS fixes was found at location [-50 east, 150 north], and also it can be seen that the position estimator responds less to these GPS fixes, but much more to the DGPS fixes obtained immediately afterwards. By observation, the discrepancy was approximately 100m since the last update (6 legs of transect away _ 3300m), and thus the error was approximately 3%. It should be noted that 100m is within the limit of the GPS error deviation, and the result suggests that accurate navigation does not necessarily require frequent surfacing. The results here demonstrate the usefulness of the fuzzy – mutual information algorithms for use in real-time sensor fusion. We believe further investigation is required to determine the complete usefulness of this approach. In Figure 10, the solid line represents the fuzzy position estimator output and the dashed line represents the dead-reckoned output. 'X' and 'O' represent differential and regular GPS fix, respectively.

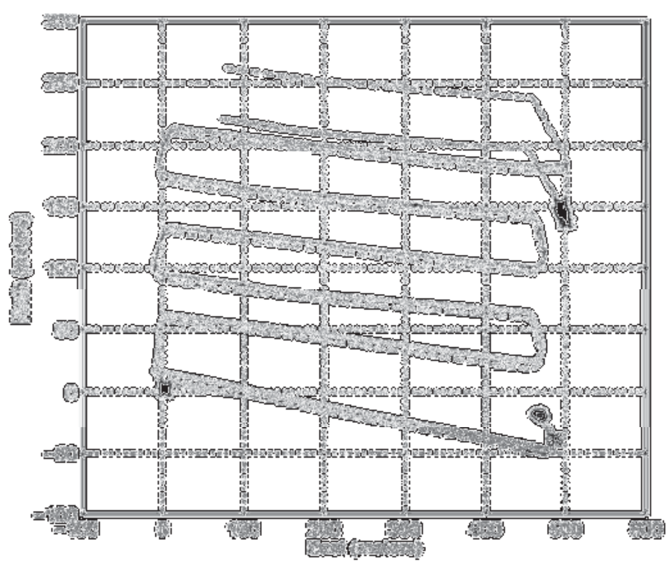
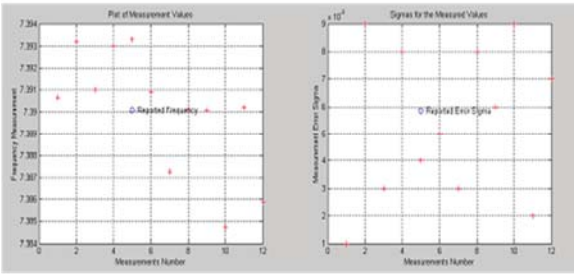
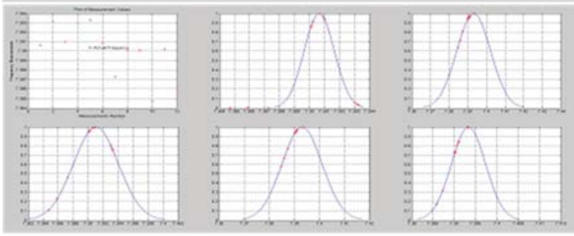
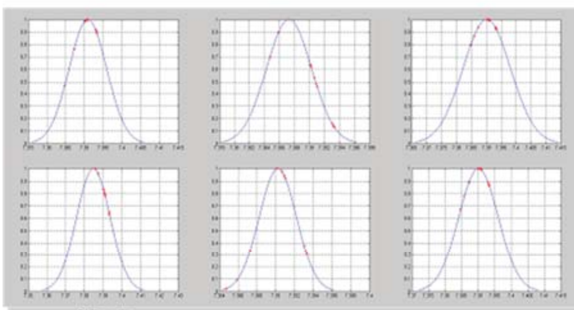


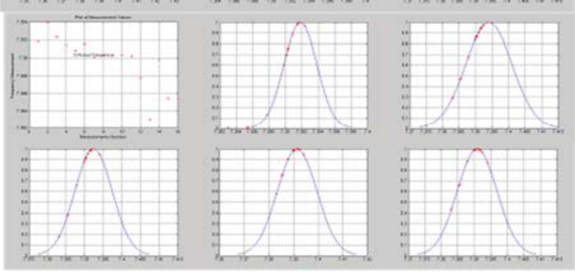
***Figure 10 – Fuzzy PNT Estimation and GPS Outputs***

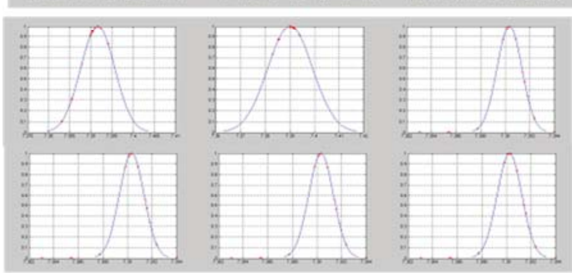
### *4.3 Results for the Mutual Information Fusion Process*

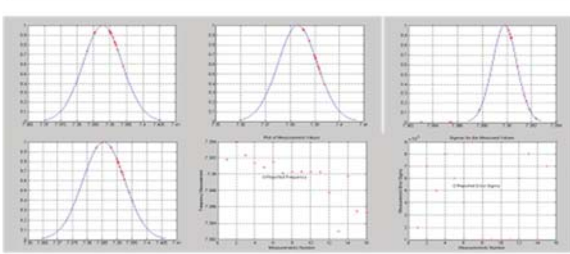


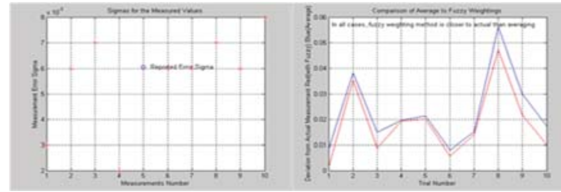












## 5. REFERENCES


1. Bonarini, A. 1997. Anytime Learning and Adaptation of Structured Fuzzy Behaviors. Adaptive Behavior Volume 5. Cambridge MA: The MIT Press.
2. Brillouin, L. 2004. Science and information theory. Dover.
3. Crowder, J. 2002. Derivation of Fuzzy Classification Rules from Multidimensional Data. NSA Technical Paper CON_0013_2002_001.
4. Crowder, J. A. 2002. Machine Learning: Intuition (Concept Learning) in Hybrid Neural Systems, NSA Technical Paper-CON-SP-0014-2002-06 Fort Meade, MD.
5. Crowder, J. 2004. Multi-Sensor Fusion Utilizing Dynamic Entropy and Fuzzy Systems. Proceedings of the Processing Systems Technology Conference, Tucson, AZ.
6. Crowder, J. 2005. Cognitive Systems for Data Fusion. Proceedings of the 2005 PSTN Processing Technology Conference, Ft. Wayne, Indiana.
7. Crowder, J. A. 2010a. The Continuously Recombinant Genetic, Neural Fiber Network. Proceedings of the AIAA Infotech@Aerospace-2010, Atlanta, GA.
8. Crowder, J. A., 2010c. Flexible Object Architectures for Hybrid Neural Processing Systems. Proceedings of the 11th International Conference on Artificial Intelligence, Las Vegas, NV.
9. J. Llinas, C. Bowman, G. Rogova, A. Steinberg, E. Waltz, and F. White. 2004. Revisiting the jdl data fusion model ii.
10. Crowder, J. A. and Crowder, J. A., and Carbone, J. N. 2011a. Recombinant Knowledge Relativity Threads for Contextual Knowledge Storage. Proceedings of the 12th International Conference on Artificial Intelligence, Las Vegas, NV.
11. Carbone, J. N. and Crowder, J. 2011b. Transdisciplinary Synthesis and Cognition Frameworks. Proceedings of the Society for Design and Process Science Conference 2011, Jeju Island, South Korea.
12. Kosko, G.. 2002. Fuzzy Cognitive Maps. International Journal of Man-Machine Studies,
13. Michalski, R. 1993. Inferential Theory of Learning as a Conceptual Basis for Multi-Strategy Learning. Machine Learning, 11:111-151.
14. Liao, S., He, J., and Tang, T. 2004. A framework for context information management. Journal of Information Science, vol. 30, pp. 528-539.
15. Ramamurti, V., and Ghosh, J. 1996. Advances in using Hierarchical Mixture of Experts for Signal Classification. Proceedings of ICASSP.
16. Stylios, C. and Groumpos, P. 2000. Fuzzy Cognitive Maps in Modeling Supervisory Control Systems. Journal of Intelligent & Fuzzy Systems 8, 2:83-98.