# Fuzzy Logic for Unattended Ground Sensor Fusion

**by**

**Arjun Bharadwaj and Jerry M. Mendel**

**January 2006**

Signal and Image Processing Institute

**UNIVERSITY OF SOUTHERN CALIFORNIA**

Department of Electrical Engineering-Systems

3740 McClintock Avenue, Room 400

Los Angeles, CA 90089-2564 U.S.A.

# Contents


**1 Introduction** **1**

**2 Data Pre-Processing** **3**

2.1 Prototype Generation . . . . . 3

2.2 Feature Extraction . . . . . 3

2.2.1 Frequency domain $f_o$ estimation: Cepstrum . . . . . 4

2.2.2 Time domain $f_o$ estimation: Autocorrelation . . . . . 4

2.3 Distribution of Features . . . . . 5

2.4 Energy Distribution . . . . . 8

2.5 Binary Classification of Tracked vs Wheeled Vehicles . . . . . 8

2.5.1 Non-Adaptive Classification . . . . . 9

2.5.2 Adaptive Classification . . . . . 10

**3 Bayesian Classifier** **11**

**4 Brief Review of Fuzzy Logic Systems** **12**

4.1 Type-1 FLS . . . . . 13

4.1.1 Optimization . . . . . 14

4.2 Type-2 FLS . . . . . 14

4.2.1 Optimization . . . . . 16

**5 Non-Hierarchical FL-RBC** **18**

**6 Classifier Fusion** **21**

6.1 Decision Fusion using the CFI . . . . . 21

6.1.1 Fuzzy Measures . . . . . 21

6.1.2 The Choquet Fuzzy Integral (CFI) . . . . . 22

6.2 Architectures for Decision Fusion . . . . . 23

6.2.1 Internal Decision Fusion . . . . . 23

6.2.2 External Decision Fusion . . . . . 25

**7 Experiments and Results** **26**

7.1 Leave-One-Out Experiment[11] . . . . . 26

7.2 Cross-Validation Experiment[11] . . . . . 28

7.3 Fusion using the CFI . . . . . 30

**8 Conclusions** **33**

**A The Choquet and Sugeno Fuzzy Integrals: A Tutorial** **34**

A.1 Introduction . . . . . 34

A.2 Fuzzy Measures . . . . . 34

A.3 The Choquet Fuzzy Integral . . . . . 37

A.4 Generic Applications of the Choquet Fuzzy Integral . . . . . 39

A.4.1 Worker Productivity . . . . . 39

A.4.2 A Collection of Rare Books . . . . . 40

A.4.3 Multiple Judges of a Sporting Event . . . . . 41

A.5 The Sugeno Fuzzy Integral . . . . . 43

**B Derivation of Some of the Properties of Fuzzy Measures** **47**

B.1 Derivation of (A.4) . . . . . 47

B.2 Derivation of (A.6) . . . . . 48

B.3 Derivation of (A.9) . . . . . 49

B.4 Derivation of (A.14) . . . . . 49

B.5 Derivations of (A.15) and (A.16) . . . . . 50

**C Proofs of the Properties of the CFI** **51**

C.1 Property 1 . . . . . 51

C.2 Property 2 . . . . . 51

C.3 Property 3 . . . . . 52

C.4 Property 4 . . . . . 52

C.5 Property 5 . . . . . 52

C.6 Property 6 . . . . . 53

C.7 Property 7 . . . . . 53

**Bibliography** **56**

# List of Tables


2.1 Estimated errors in classification using different evaluation methods. . . . . 9

4.1 Formulas for the firing degrees. . . . . 15

5.1 The classification decision for $\mathbf{x}'$ based on $[y_1(\mathbf{x}'), y_2(\mathbf{x}')]^t$. . . . . 18

7.1 Estimate of the mean and standard deviations of the testing errors in the leave-one-out experiment using only seismic data . . . . . 28

7.2 Estimate of the mean and standard deviations of the testing errors in the cross-validation experiment using only seismic data. . . . . 30

7.3 Estimate of the mean and standard deviations of the testing errors in the cross-validation experiment using both acoustic and seismic data . . . . . 32

7.3 Scores for the participant $u$ from the five judges . . . . . 41

7.3 Scores for participant $u$ from the five judges and their corresponding weights 43

# List of Figures


2.1 Distribution of features for tracked vehicles. (a) - (d) show the features for the four Heavy Track Vehicles and (e) shows the features for the one Light Track vehicle. . . . . 6

2.2 Distribution of features for tracked vehicles. (a) and (b) show the features for two Heavy Wheel vehicles, (c) and (d) show the features for the Light Wheel vehicles. . . . . 7

2.3 Average signal energy on a log scale for LT, HT, LW and HW vehicles: (a) seismic data, and (b) acoustic data. Each plot only has four data points, one for each class of vehicles. . . . . 8

2.4 Performance of the adaptive classifier, i.e. classifier error, as a function of time. (a) re-substitution method and (b) leave-one-out method . . . . . 10

6.1 Block diagram representation of classifier fusion using the CFI. . . . . 24

6.2 Block diagram of the decision strategy of the fused classifier. . . . . 25

A.1 Plot of $h(x)$ versus $x$. Note that the $\alpha$-cut shown in this figure is the set $A_i$. This is also the $h(x_i)^{th}$-cut. The continuous curve is for purposes of illustration only, i.e., it actually consists of lines connecting discrete values of $h(x_i)$ for $x = \{x_1, ..., x_n\}$. . . . . 44

A.2 Graphical illustration of the calculation of the SFI . . . . . 45

**Abstract**

This report summarizes our research for the year Jan. 05 to Dec. 05. We have developed multi-category classifiers based on seismic data to classify heavy-tracked, light-tracked, heavy-wheeled and light-wheeled ground vehicles. We focused on data collected in the normal terrain.

We also developed fusion algorithms for type-1 and type-2 Fuzzy logic Rule-Based Classifiers (FL-RBCs) based on the Choquet Fuzzy Integral (CFI). We conducted experiments to evaluate the performance of the classifiers and to evaluate the effectiveness of seismic data for classification. We also conducted experiments to evaluate the performances of fused classifiers (both seismic and acoustic) and determine if performance could be improved. Our results show that binary classification between tracked and wheeled vehicles is effective using seismic data. However, due to the inherent unreliability of the seismic data, the performance of the classifiers based on seismic data was poor when compared to the performance of the classifiers based on acoustic data. Fusing the two classifiers also did not show any appreciable improvement in performance.

We note that FL-RBCs performed better than the Bayesian equivalent for all the experiments. This shows that FL-RBCs are better suited to handle uncertainties in the data.

# Chapter 1

# Introduction

The emissions of ground vehicles contain a wealth of information which can be used for vehicle classification, e.g., in the battlefield. The model for the emissions can be simplified to be the addition of periodic components and noise. The former accounts for the periodic movements in the engine, and the latter accounts for the propulsion process in the engine and the interactions between the vehicle and the roads. Because the operating mechanisms are different for different vehicles, it is possible to distinguish among heavy-tracked, light-tracked, heavy-wheeled and light-wheeled vehicles; however, uncertainties that arise due to the non-stationary nature of the data, different speeds and data corruption make this a challenging task.

In previous years[11], a complete Type-2 FLS theory was established with IF-THEN rules and was used to design and implement Fuzzy Logic Rule-Based Classifiers (FL-RBCs) to classify ground vehicles based only on their acoustic signatures. Binary and multi-category classifiers were developed for the normal terrain using the Acoustic-Seismic Classification/Identification Data Set (ACIDS). Some of salient items learnt from previous years are:

- The non-hierarchical T2 FL-RBC performed the best among different classifier architectures and has the least complexity of the FL-RBCs. We used this architecture during this year's study.
- Adaptive majority voting makes a significant improvement in the performance of classifiers based on acoustic data. We implemented this scheme for this year's study as well.

This year, we developed multi-category FL-RBCs for classifying heavy-tracked, heavy-wheeled, light-tracked and light-wheeled ground vehicles for the normal terrain using seismic data.

We also developed the Bayesian classifier to baseline the performance of the FL-RBCs. We investigated performance gains obtained by employing a decision fusion technique called the Choquet Fuzzy Integral (CFI) to fuse classifier outputs.

This report summarizes our research during this year and is organized as follows:

- Data pre-processing which consists of data analysis, prototype extraction, feature extraction is briefly reviewed in Chapter 2.
- In addition to the Bayesian classifier we designed and implemented the non-hierarchical FL-RBC for type-1 and type-2 fuzzy sets. The classifier designs and a brief review of fuzzy logic systems is presented in Chapters 3-5.
- The CFI as a technique for decision fusion is reviewed in Chapter 6 along with fusion architectures for implementing decision fusion.
- Experiments to evaluate performance of the classifiers based on seismic data and fused classifiers were performed. Descriptions of these experiments and the results obtained are presented in Chapter 7.
- Conclusions are drawn in Chapter 8.

# Chapter 2

# Data Pre-Processing

Pre-processing was done on the Acoustic-Seismic Classification / Identification Data Set (ACIDS) that consists of more than 270 records of acoustic and seismic data collected for nine vehicles in four different environmental conditions. We focused only on data collected for the normal terrain. The vehicle categories are: Heavy-Track, Light-Track, Heavy-Wheel and Light-Wheel.

## 2.1 Prototype Generation

The seismic data was collected by a single vertical-axis geophone and contains raw digitized seismic signatures of the different vehicles. The data was low-pass filtered at 200Hz by a fourth-order Chebyshev filter and then sampled at 512Hz. The data was corrupted by noise and acoustic signals.

A complete run lasts from under 60 seconds to over 300 seconds. The data are non-stationary and the signal to noise ratio (SNR) is constantly changing. These factors make it impossible to process an entire run as a single entity. We segmented the data into two-second blocks - prototypes. Since the SNR is highest at the closest point of approach (CPA) of the vehicle, we used this point as a frame of reference to generate prototypes. A 1024-point rectangular window with 50% overlap was slid across the data before and after the CPA to generate 90 prototypes for each run.

## 2.2 Feature Extraction

We used the amplitudes of the ***second through 12th harmonics of the fundamental frequency*** $f_o$ as the features for classification. So, one of the most important steps in the

feature extraction process is the identification of the fundamental frequency.

In our previous ARL study, we used the Harmonic Line Association (HLA) algorithm to determine the fundamental frequencies and to extract acoustic features. Because of acoustic coupling into the measured seismic data, harmonics at higher frequencies are corrupted, making the determination of the seismic $f_o$ difficult using the HLA algorithm. Instead, we considered the use of a frequency-domain estimation algorithm (Cepstral Analysis) and a time-domain algorithm (Autocorrelation method) to estimate the fundamental frequency. For reasons given in Section 2.2.2, we abandoned the use of the autocorrelation method.

### 2.2.1 Frequency domain $f_o$ estimation: Cepstrum

A reliable way of obtaining an estimate of the dominant fundamental frequency is to use the cepstrum [1], which is a Fourier analysis of the logarithmic magnitude spectrum of the signal. If the log amplitude spectrum contains many regularly spaced harmonics, then the Fourier analysis of this spectrum will show a peak corresponding to the spacing between the harmonics, i.e. the fundamental frequency. The cepstral method was developed in an attempt to make a nonlinear harmonic system more linear. Effectively we are treating the signal spectrum as another signal, and then looking for periodicity in the spectrum itself. This method works well for low fundamental frequencies.

Note that the name "cepstrum" comes from reversing the first four letters in "spectrum". The horizontal-axis of the cepstrum has units of *quefrency*, and amplitude peaks in the cepstrum (which relate to periodicities in the spectrum) are called *rahmonics*.

To obtain an estimate of the fundamental frequency from the cepstrum, we looked for a peak in the quefrency region corresponding to typical seismic fundamental frequencies. We considered the interval [1,20] Hz to correspond to typical seismic frequencies.

### 2.2.2 Time domain $f_o$ estimation: Autocorrelation

The autocorrelation function finds the "similarity" between the signal and a shifted version of itself. To detect $f_o$, we took a window of the signal, with a length at least twice as long as the longest (potentially) detectable period. In our case, this corresponded to a length of 1024 samples, given a sampling rate of 512 Hz. The autocorrelation function was then calculated for this section of the signal, and the fundamental frequency was estimated by looking for a peak in the delay interval corresponding to the typical range of seismic frequencies i.e. [1, 20] Hz [2].

This method produces peaks at sub-harmonics as well as at the fundamental frequency, and it is difficult to determine which peak corresponds to the fundamental frequency. Fur-

thermore, the method does not work well when the signal is varying too rapidly. It is also computationally more intensive than the cepstral method. Consequently, $f_o$ estimation using autocorrelation yielded frequencies that were too high. In the sequel, we use only the cepstral method for estimating the fundamental frequency.

## 2.3 Distribution of Features

Features were extracted for each prototype in a run corresponding to one of the nine vehicle types. The number of runs available for each of the nine vehicles is:

Heavy-Tracked a: 14 runs
Heavy-Tracked b: 8 runs
Heavy-Tracked c: 15 runs
Heavy-Tracked d: 8 runs
Light-Tracked a: 15 runs
Heavy-Wheeled a: 8 runs
Heavy-Wheeled b: 8 runs
Light-Wheeled a: 8 runs
Light-Wheeled b: 4 runs

For each run, we computed the mean (run-mean) and standard deviation (run-standard deviation) of the magnitudes of each feature. We represented the $i^{th}$ feature distribution ($i = 1, 2, ..., 11$) by the interval [run-mean - 2 (run-standard deviation), run-mean+2 (run-standard deviation)]. Thus, each run is represented by 11 intervals, one for each feature. These intervals are the ranges into which the feature vectors fall into with high probability. Figure 2.1 contains the distribution of the features for the tracked vehicles and Figure 2.2 contains the distribution of the features for the wheeled vehicles. Each bar in the figure denotes a complete run. There are 88 bars in each plot.

Observe from Fig. 2.1 and Fig. 2.2, that:

1. The wheeled vehicle distributions show an approximate exponential decay in the magnitude of the features as we go from feature 1 to 11.

2. The magnitudes of the features for the wheeled vehicles are considerably lower than those of the tracked vehicles.

3. The magnitudes of the features for almost all the vehicles (except HT-c) decrease as the feature dimension increases; hence, higher-harmonic seismic features may be unreliable.

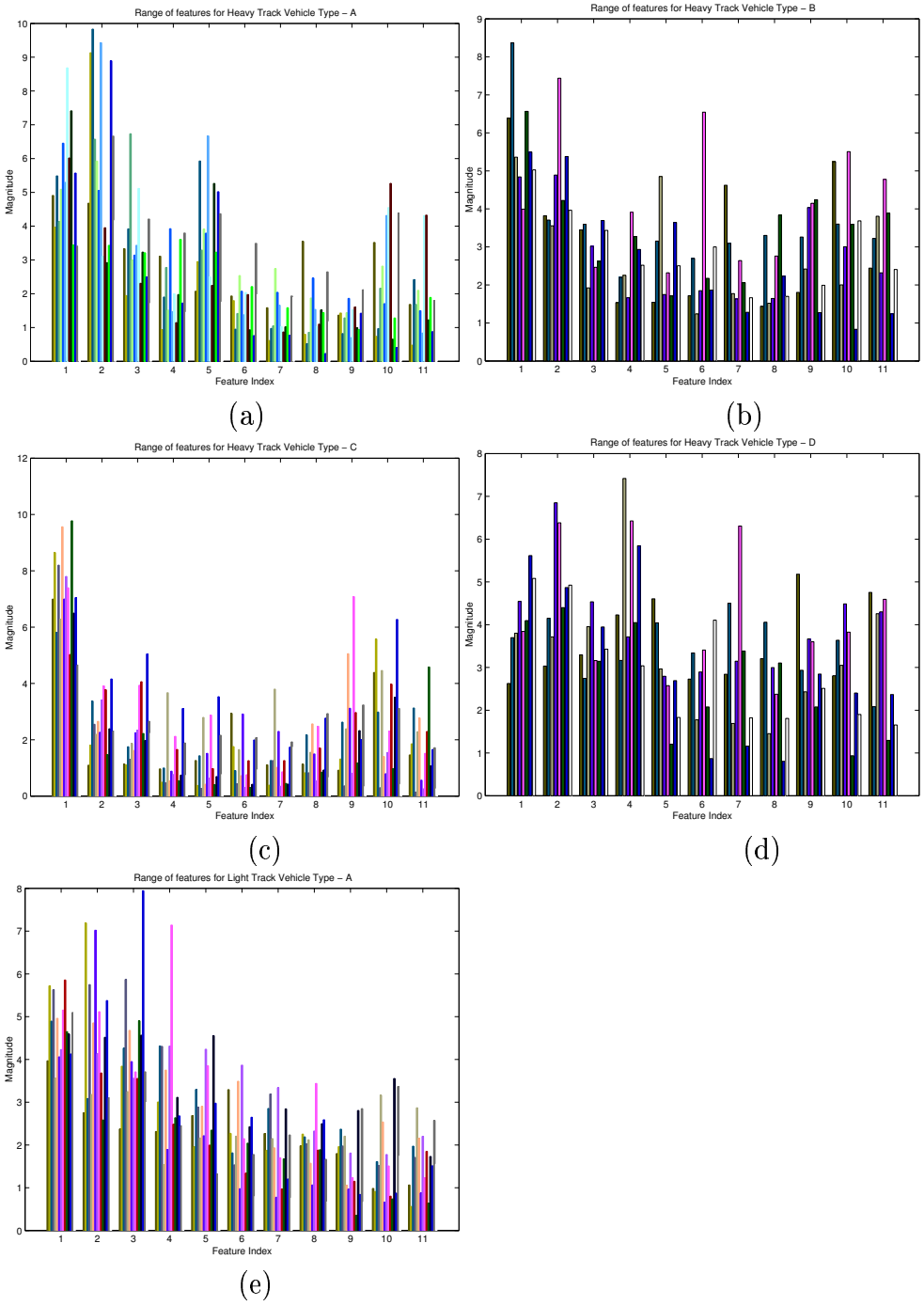



Figure 2.1: Distribution of features for tracked vehicles. (a) - (d) show the features for the four Heavy Track Vehicles and (e) shows the features for the one Light Track vehicle.

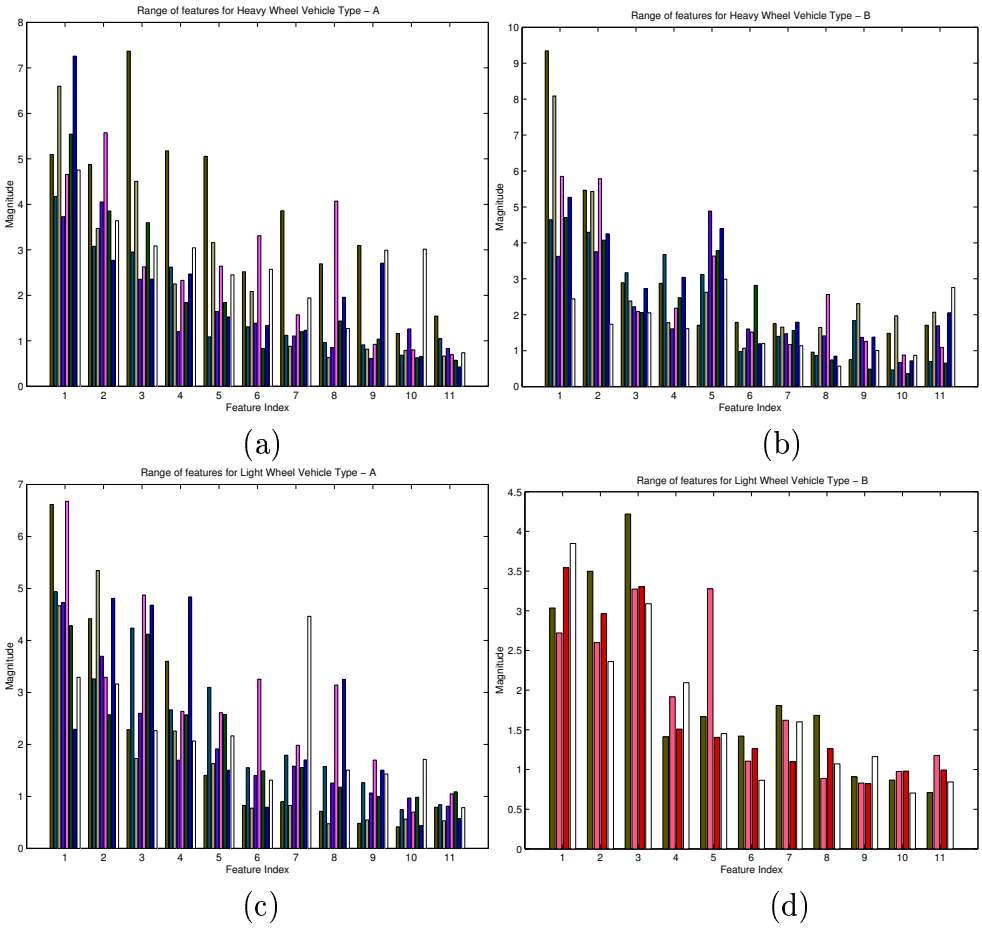
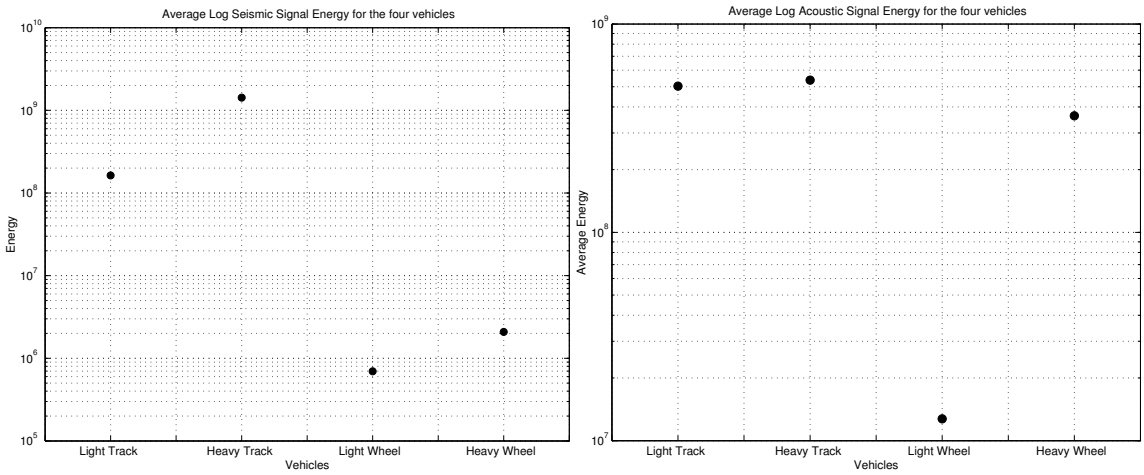



Figure 2.2: Distribution of features for tracked vehicles. (a) and (b) show the features for two Heavy Wheel vehicles, (c) and (d) show the features for the Light Wheel vehicles.




(a) (b)

Figure 2.3: Average signal energy on a log scale for LT, HT, LW and HW vehicles: (a) seismic data, and (b) acoustic data. Each plot only has four data points, one for each class of vehicles.

## 2.4 Energy Distribution

The average energy for a run is the integral of the squared amplitude of the sensor signal (acoustic or seismic) under consideration. We computed the average energy for each run and then averaged the results over all runs pertaining to a specific vehicle type. Our results are summarized in Fig. 2.3.

From Fig. 2.3, we observe that:

1. The seismic data shows appreciably lower energy levels for wheeled vehicles, which could be due to the absence of "track slap" in wheeled vehicles. Average signal energy could be used as a feature for classification, as a quick way to distinguish between wheeled and tracked vehicles. We attempt to do the same in the next section.

2. The same observation cannot be made for the acoustic data set because of the high acoustic energy levels of the HW vehicles.

## 2.5 Binary Classification of Tracked vs Wheeled Vehicles

Since we are just using the energy feature, prototypes were chosen according to their energy levels. 90 prototypes that had the highest energies for each run were chosen. The energy in each prototype was used as the single feature for classification of tracked versus wheeled vehicles, i.e. the energy in each two-second data block in a run was used as a feature. As

Table 2.1: Estimated errors in classification using different evaluation methods.

| Evaluation Method | Holdout | Cross-Validation | Leave-One-Out | Re-Substitution |
|---|---|---|---|---|
| Classifier Error | 0 | 0 | 0 | 0 |

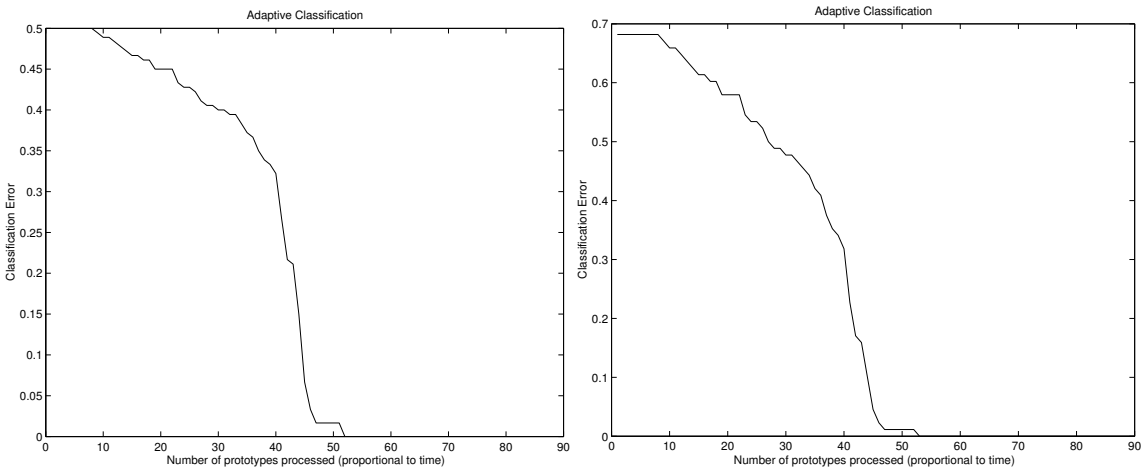
can be seen in Fig. 2.3, the energy levels for tracked vehicles differ considerably from those for wheeled vehicles.

Since we just have a one-dimensional feature set, the classification problem reduces to finding a threshold for the data set. There are a number of simple classification distribution-free algorithms that can be used for this purpose, e.g. Perceptron, Widrow-Hoff. We used the Ho-Kashyap algorithm that finds the threshold by maximizing its distance to each prototype. Details of the algorithm can be found in [3].

### 2.5.1 Non-Adaptive Classification

In non-adaptive classification, the energy levels for the prototypes in a run were summed to obtain the energy of the entire run. The energies for the tracked-vehicle runs and those for the wheeled-vehicle runs were computed and used as samples in the Ho-Kashyap classifier. The performance of the classifier was evaluated using the Holdout, cross-validation, leave-one-out, and re-substitution methods.

In the Holdout method, 10% of the samples were drawn randomly and used for testing, while the rest of the data was used for training. This was repeated a large number of times (over 1000) and the average error on all trials provided an estimate of the classifier error. In 10-fold cross-validation the data set was divided into 10 mutually exclusive subsets. Testing was done using one of the 10 subsets, whereas training was done using the remaining nine subsets. The estimated classifier error is the error averaged over all 10 possible designs. The leave-one-out method consisted of as many designs as the number of runs. In the $l$-th design, the $l$-th run was set aside for testing and the other runs were used for training. As before, the averaged error over all the designs was the estimated classifier error. In the re-substitution method the entire data set was used for both training and testing. Results for these four methods are summarized in Table 2.1. As seen from this table, perfect classification was obtained. This implies that the feature set is separable for this binary classification problem. Using only energy as a feature, we were able to classify tracked and wheeled vehicles accurately when we used entire runs.




(a) (b)

Figure 2.4: Performance of the adaptive classifier, i.e. classifier error, as a function of time. (a) re-substitution method and (b) leave-one-out method

### 2.5.2 Adaptive Classification

In many applications, we need to classify the prototypes in real time and hence may not have the luxury of processing an entire run. In such cases, we use adaptive classification. In this approach, we computed the energy levels of the prototypes and summed them up as and when they became available, after which classification was done on the available data. The performance of the adaptive classifier was evaluated using the re-substitution and holdout methods, the details of which have been described in the previous section. The Ho-Kashyap algorithm was used for training on the set of training prototypes as specified by the evaluation method, and once the threshold was obtained the test prototypes were classified adaptively. The average error for the two evaluation methods is plotted in Fig. 2.4 as a function of time (which corresponds to number of prototypes processed). Observe that perfect classification of tracked and wheeled vehicles is achieved when about half of the prototypes in a run are processed. This corresponds to just over 50 seconds in real time. This makes intuitive sense because the Closest Point of Approach (CPA) of the vehicle is reached about midway through a run and the differences in the energy levels between tracked and wheeled vehicles are most prominent at this point.

# Chapter 3

# Bayesian Classifier

The Bayesian classifier was established based on the assumptions that we made about the conditional probability density function (pdf) of each category. It has nine conditional probability models, each of which corresponds to one kind of ground vehicle, and is described by a Gaussian pdf:

$$p(\mathbf{x}|V_j) \sim N(\mathbf{x}; \mathbf{m}_j, \Sigma_j),\ j = 1, \ldots, 9, \tag{3.1}$$

where $\mathbf{x} \in R^{11}$ represents the feature vector (the magnitudes of the second through 12th harmonic components), $V_j$ represents the $j$-th kind of ground vehicle, and $\mathbf{m}_j$ and $\Sigma_j$ are the mean and covariance matrix of the multi-variate Gaussian distribution associated with $V_j$.

The mean and covariance matrix associated with $V_j$ were estimated from the prototypes of $V_j$, i.e.,

$$\mathbf{m}_j = \frac{1}{N_j} \sum_{\mathbf{x} \in V_j} \mathbf{x} \tag{3.2}$$

$$\Sigma_j = \frac{1}{N_j - 1} \sum_{\mathbf{x} \in V_j} (\mathbf{x} - \mathbf{m}_j)(\mathbf{x} - \mathbf{m}_j)^t, \tag{3.3}$$

where $N_j$ is the number of prototypes of $V_j$.

Given an unlabeled feature vector $\mathbf{x}'$ as the input, the Bayesian classifier first computes the log likelihood $L(\mathbf{x}'|V_j)$ for each kind of vehicle as:

$$L(\mathbf{x}'|V_j) = \log p(\mathbf{x}'|V_j) = -\log|\Sigma_j| - (\mathbf{x}' - \mathbf{m}_j)^t \Sigma_j^{-1} (\mathbf{x}' - \mathbf{m}_j),\ j = 1, \ldots, 9 \tag{3.4}$$

and then compares all likelihoods to determine $V_{\max}$ that is associated with the maximum likelihood. Finally, the Bayesian classifier assigns $\mathbf{x}'$ to the same category as $V_{\max}$.

# Chapter 4

# Brief Review of Fuzzy Logic Systems

We designed and implemented non-Hierarchical FL-RBCs which are based on the theory of fuzzy logic systems (FLS). It is thus pertinent to include a brief review of the theory of FLS before detailing the classifier designs.

A FLS consists of four components - fuzzifier, rule-base, fuzzy inference engine and the output processor. The *rule-base* contains rules that are extracted from expert knowledge, mathematical models or data. Each rule describes a relation from the domain $X_i \times \cdots \times X_p \subseteq R^p$ to the range $Y \in R$ and can be expressed in the following IF-THEN form:

$R^j$ : IF $x_1$ is $F_1^j$ and $x_2$ is $F_2^j$ and ... and $x_p$ is $F_p^j$, THEN $y$ is $G^j$, $j = 1, ..., M$,

where $M$ is the total number of rules, $R^j$ represents the $j$-th rule, $F_k^j$ is the antecedent associated with the $k$-th input variable $x_k$ $(k = 1, ..., p)$ and $G^j$ is the consequent associated with the output variable $y$.

Given a vector of measurements $X' \equiv \left[x'_1, ..., x'_p\right]^T$, the fuzzifier converts them into fuzzy sets $A_{x'_1}, ..., A_{x'_p}$, one for each dimension. The fuzzy inference engine then computes the firing degree for each rule. The firing degree describes the amount by which the input fuzzy sets $A_{x'_1}, ..., A_{x'_p}$ match the antecedents, $F_1^j, ..., F_p^j$. The output processor computes a crisp output, $y(X')$, by using the firing degrees and the consequents of all the rules.

In summary, through fuzzification, inference and output processing, the FLS maps the input $X'$ to an output $y(X')$ according to the rules; however, because the input fuzzy sets, antecedents and consequents can either be type-1 or type-2 fuzzy sets, the specific computations of fuzzification, inference and output processing are different for type-1 and type-2 FLS. We describe these computations in more detail in Sections 4.1 and 4.2.

## 4.1 Type-1 FLS

In a type-1 FLS, the input fuzzy sets, antecedents and consequents are all type-1 fuzzy sets, and the output processing only consists of defuzzification. For purposes of maintaining consistency with subsequent descriptions of classifiers, we describe the operations and the optimization procedures of the type-1 FLS.

**Rule Base**: In the type-1 implementation, the antecedents of each rule, $F_k^j\,(k=1,...,p)$ are modeled as type-1 fuzzy sets whose membership functions (MF) are Gaussian centered at $m_k^j$ with standard deviation $\sigma_k^j$, i.e.,

$$F_k^j : \mu_k^j(x_k) = \exp\left\{-\frac{1}{2}\left(\frac{x_k - m_k^j}{\sigma_k^j}\right)^2\right\} \equiv \phi\left(x_k; m_k^j, \sigma_k^j\right) \tag{4.1}$$

**Fuzzification**: Given the input feature vector $X' \equiv \left[x'_1,...,x'_p\right]^T$ , the system encodes $x'_k(k=1,...,p)$ as a type-1 fuzzy set $A_k$ whose MF is Gaussian centered at $x'_k$ with standard deviation $\sigma_k^j$, i.e.,

$$A_k : \mu_k(x_k) = \exp\left\{-\frac{1}{2}\left(\frac{x_k - x'_k}{\sigma_k}\right)^2\right\} \equiv \phi\left(x_k; x'_k, \sigma_k\right) \tag{4.2}$$

**Fuzzy Inference**: For the type-1 FLS, the inference engine obtains once firing degree for each rule based on the input and antecedent fuzzy sets, by using the sup-star composition [7], i.e., the firing degree of the $j$-th rule $f^j(X')\,(j=1,...,M)$ is computed [7] as

$$f^j(X') = \prod_{k=1}^{p} \sup_{x_k}\left(\mu_{A_{x'_k}}(x_k)\mu_{F_k^j}(x_k)\right) \tag{4.3}$$

$$= \prod_{k=1}^{p} \exp\left\{-\frac{\left(x'_k - m_k^j\right)^2}{2\sigma_k^2 + 2\left(\sigma_k^j\right)^2}\right\} \equiv \prod_{k=1}^{p} \phi\left(x'_k; m_k^j, \sqrt{\sigma_k^2 + \left(\sigma_k^j\right)^2}\right) \tag{4.4}$$

**Output processing**: the output processing of the type-1 implementation only consists of defuzzification which leads to a crisp output, $y(X')$, computed as [7]

$$y(X') = \frac{\sum_{j=1}^{M} f^j(x')g^j}{\sum_{j=1}^{M} f^j(x')} \tag{4.5}$$

where $f^i(x')$ and $g^j$ are the firing degree and the consequent respectively, of the $j$-th rule.

### 4.1.1 Optimization

The parameters of a type-1 FLS include the input model parameters $\sigma_k$, antecedent parameters $m_k^j$ and $\sigma_k^j$ $(k = 1, ..., p)$, and consequent parameters $g^j (j = 1, ..., M)$. there are a total of $p + M(2p + 1)$ design parameters.

We use the steepest descent algorithm to tune these parameters so as to minimize a given objective function $J$, i.e.

$$\theta_{new} = \theta_{old} - \left.\frac{\delta J}{\delta \theta}\right|_{\theta_{old}} \tag{4.6}$$

where $\theta$ is the generic symbol for all parameters to be optimized.

The partial derivative $\delta J/\delta\theta$ is computed through the partial derivative $\delta y/\delta\theta$ as

$$\frac{\delta J}{\delta \theta} = \frac{\delta J}{\delta y}\frac{\delta y}{\delta \theta}$$

where $y$ is the crisp output of type-1 FLS. The calculation of partial derivative $\delta J/\delta y$ depends on the specific form of $J$. The formulae for $\delta y/\delta\theta$ for the input, antecedent and consequent parameters of the type-1 FLS are given in [11].

## 4.2 Type-2 FLS

In the type-2 implementation of the FLS, the input fuzzy sets, antecedents and consequents are all type-2 fuzzy sets, and the output processing consists of type-reduction and defuzzification. We used interval type-2 fuzzy sets in our implementation and so we illustrate the operations and optimization procedures for them.

**Rule Base**: The antecedents of each rule $\widetilde{F}_k^j$ $(k = 1, ..., p)$ are modeled as interval type-2 fuzzy sets whose MFs are Gaussian with uncertain means, $m \in [m_{1,k}^j, m_{2,k}^j]$ and uncertain standard deviations, $\sigma \in [\sigma_{1,k}^j, \sigma_{2,k}^j]$. The lower and upper MFs (LMF and UMF) of $\widetilde{F}_k^j$, $\underline{\mu}_k^j(x_k)$ and $\overline{\mu}_k^j(x_k)$, are given as [11]

$$\underline{\mu}_k^j(x_k) = \begin{cases} \phi\left(x_k; m_{2,k}^j, \sigma_{1,k}^j\right) & \text{if } x_k \leq \left(m_{1,k}^j + m_{2,k}^j\right)/2 \\ \phi\left(x_k; m_{1,k}^j, \sigma_{1,k}^j\right) & \text{if } x_k > \left(m_{1,k}^j + m_{2,k}^j\right)/2 \end{cases} \tag{4.7}$$

$$\overline{\mu}_k^j(x_k) = \begin{cases} \phi\left(x_k; m_{1,k}^j, \sigma_{2,k}^j\right) & \text{if } x_k \leq m_{1,k}^j \\ \phi\left(x_k; m_{2,k}^j, \sigma_{2,k}^j\right) & \text{if } x_k \geq m_{2,k}^j \\ 1 & \text{otherwise} \end{cases} . \tag{4.8}$$

Table 4.1: Formulas for the firing degrees.

| $x'_k$location | $\overline{f}^j(x'_k)$ |
|---|---|
| $x'_k \leq \frac{\left(m^j_{1,k}+m^j_{2,k}\right)}{2} - \frac{\sigma^2_{1,k}\left(m^j_{2,k}-m^j_{1,k}\right)}{2\left(\sigma^j_{1,k}\right)^2}$ | $\phi\left(x'_k; m^j_{2,k}, \sqrt{\sigma^2_{1,k}+\left(\sigma^j_{1,k}\right)^2}\right)$ |
| $x'_k \geq \frac{\left(m^j_{1,k}+m^j_{2,k}\right)}{2} - \frac{\sigma^2_{1,k}\left(m^j_{2,k}-m^j_{1,k}\right)}{2\left(\sigma^j_{1,k}\right)^2}$ | $\phi\left(x'_k; m^j_{1,k}, \sqrt{\sigma^2_{1,k}+\left(\sigma^j_{1,k}\right)^2}\right)$ |
| otherwise | $\exp\left\{\frac{\left(m^j_{2,k}-m^j_{1,k}\right)^2}{8\left(\sigma^j_{1,k}\right)^2} - \frac{\left(2x'_k-m^j_{1,k}-m^j_{2,k}\right)^2}{8\left(\sigma_{1,k}\right)^2}\right\}$ |

| $x'_k$location | $\overline{f}^j(x'_k)$ |
|---|---|
| $x'_k \leq m^j_{1,k}$ | $\phi\left(x'_k; m^j_{2,k}, \sqrt{\sigma^2_{1,k}+\left(\sigma^j_{1,k}\right)^2}\right)$ |
| $x'_k \geq m^j_{2,k}$ | $\phi\left(x'_k; m^j_{2,k}, \sqrt{\sigma^2_{2,k}+\left(\sigma^j_{1,k}\right)^2}\right)$ |
| otherwise | 1 |

**Fuzzification**: Given the input feature vector $X' \equiv \left[x'_1, ..., x'_p\right]^T$, the system encodes $x'_k$ $(k = 1, ..., p)$ as an interval type-2 fuzzy set $\widetilde{A}_k$ whose MF is Gaussian centered at $x'_k$ with with uncertain standard deviation $\sigma \in [\sigma_{1,k}, \sigma_{2,k}]$. The LMF and UMF of $\widetilde{A}_k$, $\underline{\mu}_k(x_k)$ and $\overline{\mu}_k(x_k)$, are given as

$$\underline{\mu}_{\widetilde{A}_k}(x_k) = \exp\left\{-\frac{1}{2}\left(\frac{x_k - x'_k}{\sigma_{1,k}}\right)^2\right\} \equiv \phi\left(x_k; x'_k, \sigma_{1,k}\right) \tag{4.9}$$

$$\overline{\mu}_{\widetilde{A}_k}(x_k) = \exp\left\{-\frac{1}{2}\left(\frac{x_k - x'_k}{\sigma_{2,k}}\right)^2\right\} \equiv \phi\left(x_k; x'_k, \sigma_{2,k}\right) \tag{4.10}$$

**Fuzzy Inference**: In the interval type-2 implementation, the firing degree of each rule consists of two firing degrees - the lower and upper firing degrees. These are computed based on the input and antecedent fuzzy sets, by using the extended sup-star composition [7], i.e. the lower and upper firing degrees of the $j$-th rule, $\underline{f}^j(x')$ and $\overline{f}^j(x')$ are computed as

$$\underline{f}^j(x') = \prod_{k=1}^{p} \sup_{x_k} \underline{\mu}_{\widetilde{A}_{x'_k}}(x_k) \underline{\mu}^j_{\widetilde{F}_{x'_k}}(x_k) \equiv \prod_{k=1}^{p} \underline{f}^j_k(x') \tag{4.11}$$

$$\overline{f}^j(x') = \prod_{k=1}^{p} \sup_{x_k} \overline{\mu}_{\widetilde{A}_{x'_k}}(x_k) \overline{\mu}^j_{\widetilde{F}_{x'_k}}(x_k) \equiv \prod_{k=1}^{p} \overline{f}^j_k(x') \tag{4.12}$$

The formulae for computing $\underline{f}^j_k(x')$ and $\overline{f}^j_k(x')$ are given in Table 4.1.

**Output processing**: For an interval type-2 FLS, output processing consists of type reduction followed by defuzzification. Type-reduction obtains an interval output $[y_l(X'), y_r(X')]$, based on the lower and upper firing degrees as well as the consequents by using the Karnik-Mendel iterative procedures [7]. The end points of the type reduced set $y_l(X')$ and $y_r(X')$ can be expressed as [11]

$$y_l(X') = \frac{\sum_{j=1}^{M} g^j \left\{ \delta^j_l \overline{f}^j(x') + (1 - \delta^j_l) \underline{f}^j(X') \right\}}{\sum_{j=1}^{M} \left\{ \delta^j_l \overline{f}^j(x') + (1 - \delta^j_l) \underline{f}^j(X') \right\}} \tag{4.13}$$

$$y_r(X') = \frac{\sum_{j=1}^{M} g^j \left\{ \delta^j_r \overline{f}^j(x') + (1 - \delta^j_r) \underline{f}^j(X') \right\}}{\sum_{j=1}^{M} \left\{ \delta^j_r \overline{f}^j(x') + (1 - \delta^j_r) \underline{f}^j(X') \right\}} \tag{4.14}$$

where $\delta^j_l$ and $\delta^j_r$ $(j = 1, ..., M)$ are two indicator functions defined for each rule as

$$\delta^j_l = \begin{cases} 1 & \text{if } g^j \leq y^j_l(X') \\ 0 & \text{otherwise} \end{cases} \tag{4.15}$$

$$\delta^j_r = \begin{cases} 1 & \text{if } g^j \geq y^j_r(X') \\ 0 & \text{otherwise} \end{cases} \tag{4.16}$$

Defuzzification of the type-2 FLS obtains a crisp output, $y(X')$, from the type-reduced set, i.e.,

$$y(X') = \frac{1}{2} \left[ y_l(X') + y_r(X') \right] \tag{4.17}$$

### 4.2.1 Optimization

The parameters of a type-2 FLS include the input model parameters $\{\sigma_{1,k}, \sigma_{2,k}\}$, antecedent parameters $\{m^j_{1,k}, m^j_{2,k}\}$ and $\{\sigma^j_{1,k}, \sigma^j_{2,k}\}$ $(k = 1, ..., p)$, and consequent parameters $g^j (j =$

$1, ..., M$). there are a total of $2p+M(4p+1)$ design parameters. We used the steepest descent algorithm to tune these parameters. Similar to the type-1 FLS, the partial derivatives $\delta J/\delta\theta$ of the type-2 FLS are computed through the partial derivatives $\delta y/\delta\theta$. The formulae for $\delta y/\delta\theta$ for the input, antecedent and consequent parameters of the interval type-2 FLS are given in [11].

# Chapter 5

# Non-Hierarchical FL-RBC

The non-hierarchical FL-RBC as developed in [11] can be considered as a traditional FL-RBS except that its consequents and output are two-dimensional.

The rule base of the non-hierarchical FL-RBC has nine rules, one for each kind of vehicle, that are expressed as:

$$R^j\text{: IF } x_1 \text{ is } F_1^j \text{ and } \ldots \text{ and } x_{11} \text{ is } F_{11}^j\text{, THEN } y \text{ is } \left[g_1^j, g_2^j\right]^t,\ j = 1, \ldots, 9$$

where $R^j$ represents the rule for the $j$-th kind of vehicle, $F_k^j$ ($k = 1, \ldots, 11$) is the antecedent modifying the $k$-th feature variable $x_k$, and $\left[g_1^j, g_2^j\right]^t$ is the consequent modifying the output variable $y$ and is modeled as a two-dimensional vector of crisp numbers. Given an unlabeled feature vector $\mathbf{x}' \equiv [x_1', \ldots, x_{11}']^t$ (which contains crisp measurements) as the input, through fuzzification, inference and output processing, the non-hierarchical FL-RBC obtains a crisp output vector $[y_1(\mathbf{x}'), y_2(\mathbf{x}')]^t$. The non-hierarchical FL-RBC makes a final decision for $\mathbf{x}'$ based only on the signs of $[y_1(\mathbf{x}'), y_2(\mathbf{x}')]^t$ according to Table 5.1.

The non-hierarchical FL-RBC architecture can be implemented based on either type-1 or type-2 fuzzy set and fuzzy logic theories. In the type-1 non-hierarchical FL-RBC, each antecedent $F_k^j$ ($k = 1, \ldots, 11$ and $j = 1, \ldots, 9$) is modeled as a type-1 fuzzy set whose MF is Gaussian centered at $m_k^j$ with standard deviation $\sigma_k^j$, and the fuzzification process converts

Table 5.1: The classification decision for $\mathbf{x}'$ based on $[y_1(\mathbf{x}'), y_2(\mathbf{x}')]^t$.

| Decision | $y_1(\mathbf{x}')$ | $y_2(\mathbf{x}')$ |
|---|---|---|
| heavy-tracked | positive | positive |
| light-tracked | positive | negative |
| heavy-wheeled | negative | positive |
| light-wheeled | negative | negative |

each input feature measurement $x'_k$ $(k = 1, \ldots, 11)$ into a type-1 fuzzy set whose MF is Gaussian centered at $x'_k$ with standard deviation $\sigma_k$. In the type-2 non-hierarchical FL-RBC, each antecedent is modeled as an interval type-2 fuzzy set whose MF is Gaussian with uncertain mean, $m \in \left[m^j_{1,k}, m^j_{2,k}\right]$, and uncertain standard deviation, $\sigma \in \left[\sigma^j_{1,k}, \sigma^j_{2,k}\right]$, and the fuzzification process converts $x'_k$ into an interval type-2 fuzzy set whose MF is Gaussian centered at $x'_k$ with uncertain standard deviation, $\sigma \in [\sigma_{1,k}, \sigma_{2,k}]$. Accordingly, the concrete computations of fuzzification, fuzzy inference and output processing in the type-1 and type-2 non-hierarchical FLRBCs are different, the details of which can be found in Sections 4.1 and 4.2.

There are a total of 227 parameters in the type-1 non-hierarchical FL-RBC, i.e., the consequent parameters $g^j_1$ and $g^j_2$, antecedent parameters $m^j_k$ and $\sigma^j_k$, and input parameters $\sigma_k$ $(k = 1, \ldots, 11$ and $j = 1, \ldots, 9)$. Since each rule corresponds to one kind of ground vehicle, the consequent and antecedent parameters of this rule are initialized based on the label and prototypes of its corresponding vehicle. More specifically, the consequent $\left[g^j_1, g^j_2\right]^t$ of one rule is initialized as $[+1, +1]^t$ ($[+1, -1]^t$, $[-1, +1]^t$ or $[-1, -1]^t$) if its corresponding vehicle belongs to the heavy-tracked (light-tracked, heavy-wheeled or light-wheeled) category. The antecedent parameters of each rule are initialized based on the prototypes of its corresponding vehicle, i.e.,

$$m^j_k(0) = \frac{1}{N_j} \sum_{\mathbf{x} \in V_j} x_k \tag{5.1}$$

$$\sigma^j_k(0) = \sqrt{\frac{1}{N_j - 1} \sum_{\mathbf{x} \in V_j} \left[x_k - m^j_k(0)\right]^2} \tag{5.2}$$

where $N_j$ is the number of prototypes belonging to $V_j$. The input parameters, $\sigma_k$, are initialized as:

$$\sigma_k(0) = \frac{1}{M} \sum_{j=1}^{M} \sigma^j_k(0) \tag{5.3}$$

There are a total of 436 parameters in the type-2 non-hierarchical FL-RBC, i.e. the consequent parameters $g^j_1$ and $g^j_2$, antecedent parameters $\left\{m^j_{1,k}, m^j_{2,k}, \sigma^j_{1,k}, \sigma^j_{2,k}\right\}$, and input parameters $\{\sigma_{1,k}, \sigma_{2,k}\}$. These parameters of the type-2 non-hierarchical FL-RBC are initialized based on the optimal parameters of the competing fully designed type-1 non-hierarchical

FL-RBC according to the following:

$$g_1^j(0) \equiv g_1^j(\text{type-1 optimal})$$
$$g_2^j(0) \equiv g_2^j(\text{type-1 optimal}) \tag{5.4}$$
$$m_{1,k}^j(0) \equiv m_k^j(\text{type-1 optimal}) - \gamma\sigma_k^j(\text{type-1 optimal})$$
$$m_{2,k}^j(0) \equiv m_k^j(\text{type-1 optimal}) + \gamma\sigma_k^j(\text{type-1 optimal}) \tag{5.5}$$
$$\sigma_{1,k}^j(0) \equiv (1-\gamma)\sigma_k^j(\text{type-1 optimal})$$
$$\sigma_{2,k}^j(0) \equiv (1+\gamma)\sigma_k^j(\text{type-1 optimal}) \tag{5.6}$$
$$\sigma_{1,k} \equiv (1-\gamma)\sigma_k(\text{type-1 optimal})$$
$$\sigma_{2,k} \equiv (1+\gamma)\sigma_k(\text{type-1 optimal}) \tag{5.7}$$

where $\gamma \in (0,1)$. Note that in these equations their left-hand sides are the initial values of the parameters of the type-2 non-hierarchical FL-RBC, and their right-hand sides are the optimal values of the parameters of the competing type-1 non-hierarchical FL-RBC.

During the training period, all the parameters of the type-1 or type-2 non-hierarchical FL-RBC are optimized by using a steepest descent algorithm to minimize the following classification error objective function:

$$J = \sum_{\mathbf{x}\in V_{\text{train}}} \frac{1}{2}\left\{[d_1 - y_1(\mathbf{x})]^2 + [d_2 - y_2(\mathbf{x})]^2\right\} \tag{5.8}$$

where $V_{\text{train}}$ is the training set, $[d_1, d_2]^t$ is the desired classification result for $\mathbf{x}$, and is $[+1,+1]^t$ ($[+1,-1]^t$, $[-1,+1]^t$ or $[-1,-1]^t$) if $\mathbf{x}$ belongs to the heavy-tracked (light-tracked, heavy-wheeled or light-wheeled) category.

# Chapter 6

# Classifier Fusion

Wu and Mendel [11] showed that non-hierarchical FL-RBCs performed well using acoustic data to classify ground vehicles. We used seismic data to do the same. However, since seismic data is inherently more unreliable, we attempted to improve performance by developing a number of classifier designs and fusing the outputs of the classifiers. We used the Choquet Fuzzy Integral (CFI) as a tool for decision fusion. The CFI has been shown [1, 10] to achieve better classification performance than the worst component classifier. In this section we briefly review the theory behind the fusion algorithm and illustrate the fusion architectures that we employed.

## 6.1 Decision Fusion using the CFI

The fuzzy integral is a non-linear approach to combine multiple sources of uncertain information (e.g., in pattern recognition applications, where results from multiple classifiers will be combined). The CFI is a specific type of fuzzy integral which combines information from multiple sources by taking into account subjective evaluation of the worth of each of the sources. It relies on the concept of a fuzzy measure which we describe next.

### 6.1.1 Fuzzy Measures

The fuzzy integral relies on the concept of a fuzzy measure which in turn is a generalization of the concept of a probability measure. Consider a finite universal set $Y = \{y_1, \ldots, y_n\}$ and let $P(Y)$ be the power set of $Y$ . For our application $Y$ is the set of classifiers to be fused. *A fuzzy measure* over the set $X$ is the set function

$$g : P(Y) \rightarrow [0, 1] \tag{6.1}$$

such that:

1. $g(\emptyset) = 0$, $g(Y) = 1$
2. If $A, B \subset P(Y)$ and $A \subset B$, then $g(A) \leq g(B)$.

Usually $g(A)$ is interpreted as the contribution of the subset $A$ within the set $Y$. Sugeno [8] introduced the concept of $\lambda$ - fuzzy measures which specifies the importance of the union of disjoint subsets $A$ and $B$. the fuzzy measure of the union is given by

$$g_\lambda(A \cup B) = g_\lambda(A) + g_\lambda(B) + \lambda g_\lambda(A) g_\lambda(B) \ , \ \lambda > -1 \tag{6.2}$$

for all $A, B \subset Y$ with $A \cap B = \emptyset$.

Given the fuzzy densities $g_\lambda^1, g_\lambda^2, ..., g_\lambda^n$ for a set of $n$ classifiers we can find $\lambda$ by solving the equation

$$\lambda + 1 = \prod_{i=1}^{n} \left(1 + \lambda g^i\right) \tag{6.3}$$

Refer to Appendix A to see how (A.8) is obtained. Appendix A (which is a tutorial) also contains a detailed list of the properties of the fuzzy measures.

### 6.1.2 The Choquet Fuzzy Integral (CFI)

The Choquet fuzzy integral (CFI) is the integral of a measurable function $h : Y \rightarrow [0, 1]$ with respect to a fuzzy measure $g_\lambda$. For our decision fusion application, $Y$ is a set of classifiers and $h(y)$ is the soft output of the classifier (i.e., the confidence or evidence grade of the classifier) denoting that an input sample is from a particular class. In general, $Y = \{y_1, \ldots, y_n\}$ is a set of information sources and $h(y_i)$ is the confidence grade of source $i$ that a particular hypothesis is true.

Under this framework, the CFI is defined as

$$E_g(h) = \int_X h \circ g = \sum_{i=1}^{n} h(y_i) \left[g\left(A_i\right) - g\left(A_{i-1}\right)\right] \tag{6.4}$$

where $h(y_1) \geq h(y_2) \geq \cdots \geq h(y_n)$ and $h(y_{n+1}) = 0$. Set $A$ is a sequence of nested subsets starting from $A_1 = \{y_1\}$, and subsequently adding in elements $y_2, ..., y_n$, one at a time to get $A_i = \{y_1, \ ... \ , \ y_i\} \subseteq Y$. For more details on the CFI, properties, heuristic interpretations and other examples, see Appendix A.

In this form of the CFI $h(y_i)$ is a crisp real value and the ordered set $h$ is considered to be a type-1 fuzzy set. It is monotonically decreasing in the domain $[0, 1]$. When the classifiers

are constructed based on type-2 logic, then the output of the classifier $h(y_i)$ is an interval type-2 fuzzy set. We denote the end-points of this interval set as $h_{il}$ and $h_{ir}$. The extended version of the CFI is obtained when the interval type-2 fuzzy sets are fused together. In this case, the extended CFI becomes [1]

$$\int_X H \circ g = \left[ \sum_{i=1}^{n} h_{il} \left[ g\left(A_i\right) - g\left(A_{i-1}\right) \right], \sum_{i=1}^{n} h_{ir} \left[ g\left(A_i\right) - g\left(A_{i-1}\right) \right] \right] \tag{6.5}$$

## 6.2 Architectures for Decision Fusion

The CFI can be applied in different ways to the multi-category classification problem. We focus on the application of the CFI for decision fusion.

### 6.2.1 Internal Decision Fusion

During the previous years of our study, classifier architectures were developed based on acoustic data to classify ground vehicles into the four categories. These classifiers were evaluated using the leave-one-out and cross-validation methods. In each of these methods numerous classifier designs emerged, the results of which were combined using a majority vote. We propose to use the CFI to fuse the outputs of these classifiers.

Since the non-hierarchical FL-RBC has been found to be the most effective classifier architecture [11], the outputs of this classifier are fused. As mentioned in Chapter 5 the non-hierarchical FL-RBC can be implemented using type-1 or type-2 fuzzy sets. In the former case, (A.14) is used while in the latter case (6.5) is used for fusion.

Figure 6.1 shows a block diagram of this fusion architecture. To illustrate more clearly, the usage of the CFI for fusion, we consider a simple binary classification example.

#### Fusion of Binary Classifiers

Let $X = \{x_1, x_2, \ldots, x_{11}\}$ be the feature set extracted from the data. The antecedents in each of the $l\,(l = 1, ..., 9)$ rules, $F_k^l\,(k = 1, \ldots, 11)$ (one for each feature) are modeled as type-1 fuzzy sets with membership functions (MFs) $\mu_{F_k^l}(x_k)$, and the consequents $q^l$ are modeled as crisp numbers. Given an extracted feature vector $X' = [x'_1, ..., x'_{11}]$, the type-1 FL-RBC models each $x'_k$ as a type-1 fuzzy set $X_k$ and computes the firing degree $f^l(x')$ for each rule. The rules are then combined through defuzzification to obtain a crisp output $y(x')$. The decision about the feature vector $X'$ depends on the sign of $y(x')$. If $y(x')$ is positive then $x'$ is classified as a tracked vehicle, and if $y(x')$ is negative then $x'$ is classified as a wheeled vehicle.

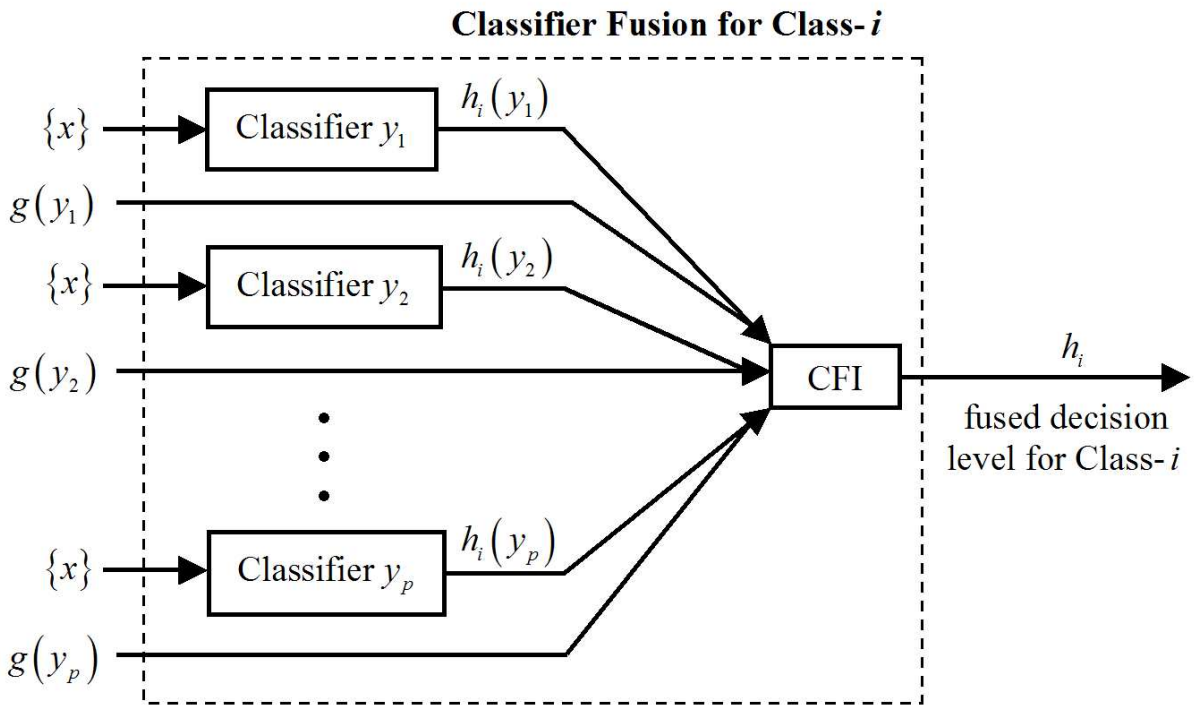



Figure 6.1: Block diagram representation of classifier fusion using the CFI.

In the interval type-2 FL-RBC, the rule structure is the same as that of the type-1 FL-RBC, but the antecedents $\widetilde{F}_k^l (k = 1, \ldots, 11)$ and the feature sets are modeled as interval type-2 fuzzy sets. The consequent $q^l$ is still modeled as a crisp number. Given an extracted feature vector $X' = [x'_1, ..., x'_{11}]$, the FLS models each $x'_k$ as an interval type-2 fuzzy set $\widetilde{X}_k$ and computed the upper and lower firing degrees, $\overline{f}^l(x')$and $\underline{f}^l(x')$, for each rule. The rules are combined through type reduction to obtain a type-reduced set $[y_l(x'), y_r(x')]$. Finally, the type-2 FL-RBC defuzzifies the type-reduced set to get a crisp output $y(x') = [y_l(x') + y_r(x')]/2$. As in the case of the type-1 FL-RBC, the decision on the extracted feature vector $X'$ depends on the sign of $y(x')$, where if $y(x')$ is positive then $x'$ is classified as a tracked vehicle, and if $y(x')$ is negative then $x'$ is classified as a wheeled vehicle.

Cross-validation is used to test the classifiers and a number of designs emerges for the type-1 and the type-2 classifiers. The parameters of all classifiers are optimized using the training data and a steepest descent algorithm. After training, the performance of the $k^{th}$ classifier is characterized by its false alarm rate (FAR), $p_k$. Given a new input feature vector $X' = [x'_1, ..., x'_{11}]$, both the type-1 and type-2 FL-RBCs generates outputs $y(X')$ for each of their classifier designs.

The CFI can be used to combine all outputs of the type-1 and type-2 classifiers with

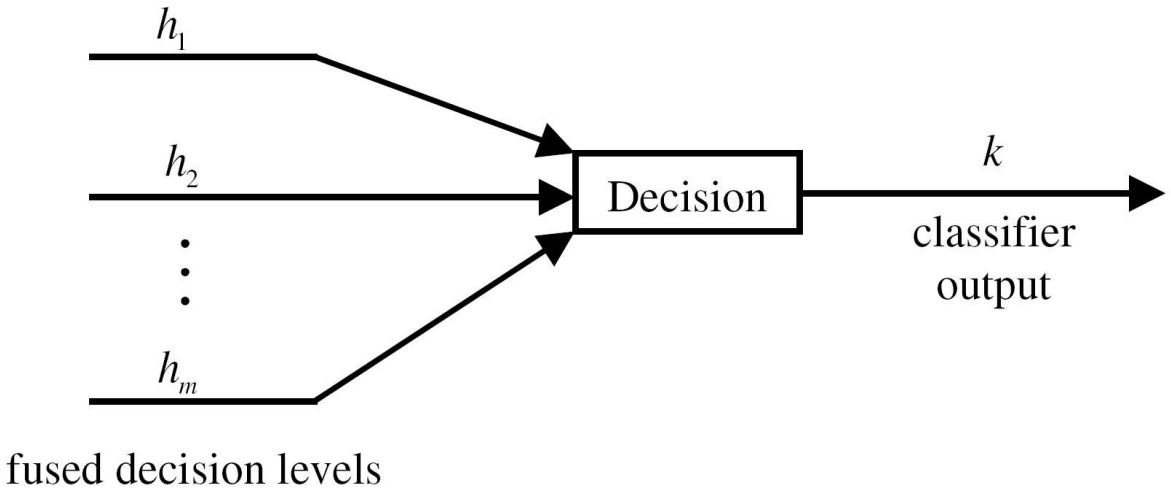



Figure 6.2: Block diagram of the decision strategy of the fused classifier.

respect to their corresponding FARs. The numerical outputs, $y(X')$, of the classifier designs are the functions $h_k(X)\,(k = 1\,, \ldots\,, n)$ for a given set of input feature vectors $X$, and the fuzzy densities $g^k$ correspond to the performance of each classifier design. If the probability of classification error using training data for design $y_k$ is $Q(y_k)$, then the fuzzy density for classifier $y_k$ is $g^k = Q(y_k)$ The outputs of the individual classifier designs are combined using the CFI as in (A.14), and a final output $y(X')$ is obtained which is thresholded to make a final decision as to whether the input feature set $X'$ corresponds to a wheeled or a tracked vehicle.

### 6.2.2 External Decision Fusion

Since we have access to both acoustic and seismic data, we can fuse classifiers based on each. We first evaluate the performance of the classifiers based on acoustic data and on seismic data and use their individual performances as the $\lambda$-fuzzy measures in the CFI. The outputs of these classifiers are fused to provide a fused classifier which is then based lined against the individual classifiers. Figure 6.2 shows a block diagram illustrating external decision fusion.

# Chapter 7

# Experiments and Results

To evaluate the Bayesian and FL-RBCs, we have performed experiments that test the classifiers based on seismic data as well as on fused classifiers. During the past few years, Wu and Mendel [11] established a testing methodology which we use here in order to remain consistent and also for effective comparison. We describe these experiments in detail in the following sections.

## 7.1 Leave-One-Out Experiment[11]

The leave-one-out experiment determines the performance of the classifiers based on the seismic data alone. It consists of 88 classifier designs (since there are 88 vehicle runs available). In the $t$-th design, the $t-$th run is left out for testing, and the remaining 87 run are used for training. The training data are used to optimize the classifier, and the testing data are used to assess the performance of the classifier. We used CPA-based prototypes for this experiment. FL-RBCs using type-1 and type-2 fuzzy logic and the Bayesian classifier were tested using this method.

So far we have designed the classifier in the way that the decision for each prototype only depends on its own features, which we call the *non-adaptive* operating mode. In previous years, we have found that even a simple *majority-voting based adaptive* operating mode can improve the classification performance [11]. In this adaptive working mode, the classification decision for each prototype depends on not only the prototype itself, but also on the decisions made on other prototypes of the same run. More specifically, if $x_1, x_2, ..., x_n$ are the prototypes of the same run and $s_1, s_2, ..., s_n$ are their corresponding non-adaptive decisions, then the adaptive decision for $x_n$ , $s_n^a$ is obtained by taking a majority vote on $s_1, s_2, ..., s_{n-1}, s_n$, i.e., $s_n^a$ is heavy-tracked (light-tracked, heavy-wheeled or light-wheeled) only if more than half of $s_1, s_2, ..., s_n$ are heavy-tracked (light-tracked, heavy-wheeled or light-wheeled). We

conducted the leave-one-out experiment for both the non-adaptive and adaptive operating modes for seismic data. An algorithm detailing the procedure is given next

```
for t =1:88 //88 designs in total
{
Leave out the prototypes of the t-th run for testing;
Use the prototypes of the remaining runs for training;
Initialize the parameters of the type-1 fuzzy rule-based classifier;
for n=1:400 //train and test for 400 epochs
 { Optimize the type-1 FL-RBC using training prototypes;
   Test the type-1 FL-RBC using the testing prototypes for the non-adaptive
operational mode;
   Save the testing error rate as e_1(t,n);
 }
Save the minimum of e_1(t,n) for n = 1,...,400 as p_1(t);
Save the optimal parameters as θ_1(t);
Test the optimal type-1 FL-RBC using the same testing prototypes for adaptive
operational mode, and        save the classification error as a_1(t);
Initialize the parameters of the type-2 FL-RBC based on θ_1(t);
for j=1:400 //train and test for 400 epochs
 { Optimize the type-2 FL-RBC using training prototypes;
   Test the type-2 FL-RBC using the testing prototypes for the non-adaptive
operational mode;
   Save the testing error rate as e_2(t,n);
 }
Save the minimum of e_2(t,n) for n = 1,...,400 as p_2(t);
Save the optimal parameters as θ_2(t) ;
Test the optimal type-2 FL-RBC using the same testing prototypes for adaptive
operational mode, and save the classification error as a_2(t);
}
Compute the mean and standard deviation of p_1(t), a_1(t), p_2(t) and a_2(t) for t =
1,...,88.
```

The results of this experiment are summarized in Table 7.1. Observe that:

- The Bayesian classifier performed worse in the adaptive operating mode. Since the adaptive mode is essentially a majority voting scheme, it is effective only when the error is under 0.5 [11].

Table 7.1: Estimate of the mean and standard deviations of the testing errors in the leave-one-out experiment using only seismic data .

| | Non-adaptive mode | | Adaptive mode | |
|---|---|---|---|---|
| **Classifier** | **Average** | **SD** | **Average** | **SD** |
| Bayesian | 0.7891 | 0.0719 | 0.8587 | 0.1164 |
| Non-hierarchical Type-1 | 0.2783 | 0.2685 | 0.2524 | 0.2378 |
| Non-hierarchical Type-2 | 0.2738 | 0.2680 | 0.2429 | 0.2184 |

- Both the FL-RBCs performed much better than the Bayesian classifier in the non-adaptive operating mode as well as in the adaptive operating mode.
- The type-2 non-hierarchical classifier performed marginally better than its type-1 equivalent.
- When compared to the results of the leave-one-out experiments for classifiers based on acoustic data [11], the classifiers based on seismic data performed significantly poorer. We conclude that seismic data is an unreliable source for feature exaction and classification.

## 7.2 Cross-Validation Experiment[11]

The training and testing data were grouped on the basis of prototypes. The prototypes were randomly divided into 10 even folds, each fold containing about 10% of prototypes. The number of designs implemented were 10. In the $l$-th design, prototypes of the $l$-th fold were set aside for testing and the prototypes of the remaining folds were used for training. We used CPA-based prototypes for this experiment. FL-RBCs using type-1 and type-2 fuzzy logic and the Bayesian classifier were tested using this method. An algorithm detailing the procedure is given next

```
Index the prototypes of all runs;
Randomly permute the index of all prototypes;
Divide the prototypes into 10 even folds;
for t =1:10 //10 designs in total
{
Leave out the prototypes of the t-th fold for testing;
Use the prototypes of the remaining folds for training;
```

```
   Initialize the parameters of the type-1 fuzzy rule-based classifier;
   for n=1:1000 //train and test for 1000 epochs
    { Optimize the type-1 FL-RBC using training prototypes;
      Test the type-1 FL-RBC using the testing prototypes for the non-adaptive
operational mode;
      Save the testing error rate as e_1(t,n);
    }
   Save the minimum of e_1(t,n) for n = 1,...,1000 as p_1(t);
   Save the optimal parameters as θ_1(t);
   Initialize the parameters of the type-2 FL-RBC based on θ_1(t);
   for j=1:1000 //train and test for 400 epochs
    { Optimize the type-2 FL-RBC using training prototypes;
      Test the type-2 FL-RBC using the testing prototypes for the non-adaptive
operational mode;
      Save the testing error rate as e_2(t,n);
    }
   Save the minimum of e_2(t,n) for n = 1,...,1000 as p_2(t);
   Save the optimal parameters as θ_2(t) ;
   }
   Compute the mean and standard deviation of p_1(t)and p_2(t) for t = 1,...,10.
```

The prototypes in this experiment were trained for 1000 epochs, so as to account for the relatively smaller number of training prototypes when compared to the leave-one-out experiment. The results of this experiment are summarized in the first column of Table 7.2. Observe that:

- The FL-RBCs performed much better than the Bayesian classifier. We conclude that when non-stationary unreliable data is used for classification, it is sub-optimal to model the data probabilistically.
- Type-2 FL-RBC performed marginally better than their type-1 counterparts.
- The results from the cross-validation experiment are in total agreement with the results from the leave-one-out experiment.

Table 7.2: Estimate of the mean and standard deviations of the testing errors in the cross-validation experiment using only seismic data.

| | Seismic data | | Fused classifier | |
|---|---|---|---|---|
| **Classifier** | **Average** | **SD** | **Average** | **SD** |
| Bayesian | 0.7248 | 0.0642 | 0.7315 | 0.054 |
| Non-hierarchical Type-1 | 0.3883 | 0.0688 | 0.3324 | 0.068 |
| Non-hierarchical Type-2 | 0.3738 | 0.0687 | 0.3629 | 0.0642 |

## 7.3 Fusion using the CFI

In this experiment we used the soft outputs of the classifiers (both Bayesian and FL-RBC). We used the log-likelihoods of the Bayesian classifier as the soft outputs to be fused. In the type-1 FL-RBC, $y_1(\mathbf{x}')$ and $y_2(\mathbf{x}')$ were considered to be the soft outputs fused using the CFI. The fused outputs were then defuzzified to get a crisp output. In the type-2 FL-RBC case, the output from the classifier after type-reduction is an interval type-2 fuzzy set; therefore, the extended CFI as described in (6.5) was used to fuse the classifier outputs.

This experiment is similar to the previous cross validation experiment. Prototypes were randomly divided into 10 even folds, each fold containing about 10% of prototypes. We implemented 10 fused classifier designs. In the $l$-th design, the $l$-th fold was set aside for testing. The cross validation experiment (as described in Section 7.2) was conducted on the 9 remaining folds.

We designed 9 non-hierarchical FL-RBCs using the 9 remaining folds. The $r$-th fold ($r = 1, ..., 9$) was left out for testing and prototypes from the remaining 8 folds were used for training. Training was done for 1000 epochs and the optimal design (least error rate) was recorded; thus, 9 such classifiers designs were stored for each $l$ ($l = 1, ..., 10$). They were then fused and tested on the $l$-th fold which was originally left out. Since $l = 1, ..., 10$, there were 10 fused classifier designs that emerged. The average error rate and the standard deviation were recorded. Since 9 classifier designs were fused for each $l$ ($l = 1, ..., 10$), a total of 90 non-hierarchical FL-RBCs were designed for this experiment. An algorithm detailing the procedure is given next:

```
Index the prototypes of all runs;
Randomly permute the index of all prototypes;
Divide the prototypes into 10 even folds;
for l =1:10 //10 fused classifier designs in total
{
```

```
      Leave out the prototypes of the l-th fold for testing;
      Use the prototypes of the remaining folds for generation classifier designs
to be fused
      for r = 1:9 // 9 FL-RBC designs for each l
      {
         Leave out the prototypes of the r-th fold for testing
        Use the prototypes of the remaining 8 folds for training;
        Initialize the parameters of the type-1 fuzzy rule-based classifier;
        for n=1:1000 //train and test for 1000 epochs
        { Optimize the type-1 FL-RBC using training prototypes;
          Test the type-1 FL-RBC using the testing prototypes for the non-adaptive
operational mode;
          Save the testing error rate as e_1(r,n);
        }
        Save the minimum of e_1(r,n) for n = 1,...,1000 as p_1(r);
        Save the optimal parameters as θ_1(r);
        Initialize the parameters of the type-2 FL-RBC based on θ_1(r);
        for j=1:1000 //train and test for 400 epochs
        { Optimize the type-2 FL-RBC using training prototypes;
          Test the type-2 FL-RBC using the testing prototypes for the non-adaptive
operational mode;
          Save the testing error rate as e_2(t,n);
        }
        Save the minimum of e_2(r,n) for n = 1,...,1000 as p_2(r);
        Save the optimal parameters as θ_2(r) ;
      }
      Use the minimum error rates p_1(r)and p_2(r) as the fuzzy densities and θ_1(r)
and θ_2(r)          as the classifier parameters for r = 1,...,9.
      Fuse the r classifiers and test on the l-th fold.
      Save the error rate as pfe_1(l) and pfe_2(l)
   }
   Compute the mean and standard deviation of pfe_1(l)and pfe_2(l) for l = 1,...,10.
```

The results of this experiment can be found in the second column in Table 7.2. Observe that:

Table 7.3: Estimate of the mean and standard deviations of the testing errors in the cross-validation experiment using both acoustic and seismic data

| | **Acoustic data** | | **Fused classifier** | |
|---|---|---|---|---|
| **Classifier** | **Average** | **SD** | **Average** | **SD** |
| Bayesian | 0.2231 | 0.0097 | 0.3541 | 0.082 |
| Non-hierarchical Type-1 | 0.1428 | 0.012 | 0.1697 | 0.106 |
| Non-hierarchical Type-2 | 0.1373 | 0.011 | 0.1642 | 0.12 |

- The performance of the Bayesian classifier is quite bad. Implementing the CFI as a fusion algorithm did not help improve its performance.
- There was only a slight improvement in performance when the CFI was used to fuse the classifiers; however, since the variance of the error rate of the classifiers is already so low, this is an expected result.
- We conclude that classification based on seismic data alone does not yield good performance. The inherent unreliability of the data, and consequently the features, is severe.

The same experiment was repeated for just the acoustic data and 10 fused classifier designs were recorded. Since the $l$-th fold was left out in the $l$-th fused classifier design, the classifiers based on the acoustic and seismic data can be fused and tested on the $l$-th fold. The first column of Table 7.3 shows the performance of the different classifier designs for the cross-validation experiment using only the *acoustic data,* while the second column of the same table shows the results of this experiment for a fused classifier that is the combination of classifiers based on both acoustic and seismic data. Observe that:

- Acoustic data is a more reliable data source for classification than acoustic plus seismic data.
- The performance of the classifiers based on acoustic data was reported in [11] and is much better than classifiers based on seismic data for all operating modes and algorithms.
- Because of the large difference in performance of the two kinds of classifiers, the fused classifier did not improve performance. We conclude that using seismic data for multi-category classification is worse than just using acoustic data.

# Chapter 8

# Conclusions

We have developed binary and multi-category classifiers based on seismic data to classify heavy-tracked, light-tracked, heavy-wheeled and light-wheeled ground vehicles and base lined their performance against a Bayesian classifier. We focused on data collected in the normal terrain.

We also developed fusion algorithms for type-1 and type-2 Fuzzy logic Rule-Based Classifiers (FL-RBCs) based on the Choquet Fuzzy Integral (CFI). We conducted leave-one-out and cross validation experiments to evaluate the performance of the classifiers and to evaluate the effectiveness of seismic data for classification. We also conducted modified cross validation experiments experiments to evaluate the performances of fused classifiers (both seismic and acoustic) and determine if performance could be improved.

Our results show that seismic data can be very effective for binary classification when energy is used as a feature. When multi-category classification was considered, the performance of the classifiers based on seismic data was poor when compared to the performance of the classifiers based on acoustic data. The performance gain was only marginal when the classifiers were tested in the adaptive mode, in the leave-one out experiment. Fusing the different classifier designs using only seismic data also did not show any appreciable improvement in performance. The performance actually deteriorated when classifiers using only seismic data and those using only acoustic data were fused. We conclude, therefore, that seismic data is a poor choice for multi-category classification of ground vehicles.

If seismic data are available, a two-stage classification process may be the optimal approach, where we first classify tracked and wheeled vehicles using seismic data and then classify heavy-tracked and light-tracked (heavy-wheeled and light-wheeled) using acoustic data.

We note that FL-RBCs performed better than the Bayesian for all the experiments conducted. This shows that FL-RBCs are better suited to handle uncertainties in the data.

# Appendix A

# The Choquet and Sugeno Fuzzy Integrals: A Tutorial

## A.1 Introduction

The fuzzy integral is a non-linear approach to combine multiple sources of uncertain information (e.g., in pattern recognition applications, where results from multiple classifiers will be combined). The function being integrated provides a confidence value for each information source for a particular hypothesis, and the integral is evaluated over the set of information sources. The Choquet (and Sugeno) fuzzy integral is a specific type of fuzzy integral which combines information from multiple sources by taking into account subjective evaluation of the worth of each of the sources.

## A.2 Fuzzy Measures

The fuzzy integral relies on the concept of a fuzzy measure which in turn is a generalization of the concept of a probability measure. Consider a finite universal set $X = \{x_1, \ldots, x_n\}$ which can be interpreted in a number of ways, e.g.

- $X$ is a set of expert judgments concerned with decision making.
- $X$ is a set of attributes or features. Each element of $X$ is used to calculate a degree of membership for an object $u \in U$ with respect to a class $w \in \Omega$.
- $X$ is a set of classifier outputs. This is different from the previous interpretation in that the classifier outputs can be represented as confidence levels for associating an object with a particular class.

Let $P(X)$ be the power set of $X$. A *fuzzy measure* over the set $X$ is the set function

$$g : P(X) \rightarrow [0, 1] \tag{A.1}$$

such that

1. $g(\emptyset) = 0$, $g(X) = 1$
2. If $A, B \subset P(X)$ and $A \subset B$, then $g(A) \leq g(B)$.

Usually $g(A)$ is viewed as the importance or power of an individual source or subset of sources ($A$) within the set $X$.

Another way of interpreting fuzzy measures is by considering the effect of the contribution of an element to a union or subset of elements. The contribution or added value of element $X_i$ in union $A$ is defined by $g(A \cup X_i) - g(A)$.

According to Sugeno [8], a fuzzy measure $g(A \cup B)$, which specifies the importance of the union of disjoint subsets $A$ and $B$, cannot be completely ascertained from the component measures $g(A)$ and $g(B)$. Consequently, he introduced $\lambda$ - fuzzy measures (also called "Sugeno measures") which satisfy the additional property, that:

$$g_\lambda(A \cup B) = g_\lambda(A) + g_\lambda(B) + \lambda g_\lambda(A) g_\lambda(B) \ , \ \lambda > -1 \tag{A.2}$$

for all $A, B \subset X$ with $A \cap B = \emptyset$. Sugeno fuzzy measures are typically denoted as $g_\lambda$; however, since they have found widespread use in applications (especially those involving fuzzy integrals), it has become common practice to denote them just as $g$. In this report, $g$ and $g_\lambda$ are used interchangeably.

The following are important properties of fuzzy measures:

1. If $\lambda = 0$, then the fuzzy measure $g_\lambda$ becomes a probability measure in that $g(A \cup B) = g(A) + g(B)$. If $\lambda < 0$, then $g_\lambda$ shows sub-additivity in that $g(A \cup B) < g(A) + g(B)$ and if $\lambda > 0$, then $g_\lambda$ shows super-additivity in that $g(A \cup B) > g(A) + g(B)$.
2. Let $X$ be a finite set of information sources $X = \{x_1, \ldots, x_n\}$ and let $g_\lambda^i = g_\lambda(\{x_i\})$. The values $g_\lambda^1, g_\lambda^2, \ldots, g_\lambda^n$ are called *fuzzy densities*[1] and represent the importance of the individual information sources.
3. Let

$$A_i = \{x_1, \ldots, x_i\} \subseteq X \tag{A.3}$$

[1] They are called "fuzzy" because $g_\lambda^1, g_\lambda^2, \ldots, g_\lambda^n$ are the values of the membership function of the fuzzy set $g$ defined on $X$.

We can then form a sequence of nested sets $A_1, \ldots, A_n$, starting from $A_1 = \{x_1\}$, and subsequently adding in elements $x_2, \ldots, x_n$, one at a time (note that $A_n = X$ and $A_0 = \emptyset$). The measure $g(A_i)$ is calculated from the following recursive formula which can be derived from (A.2) (see Section B.1):

$$g(A_i) = g(A_{i-1} \cup \{x_i\}) = g^i + g(A_{i-1}) + \lambda g^i g(A_{i-1}) \quad \text{for } 1 \leq i \leq n \tag{A.4}$$

where $g(A_1) = g^1$ and $g(A_n) = g(X)$.

4. Given the fuzzy densities for a set of sources $X$, it is important to determine the measures of the elements of the power set $P(X)$. This is essential in many applications and, as we explain next, can be done by using the $\lambda$ - fuzzy measure. Let $A_i = \{x_1, \ldots, x_i\} \subseteq X$. According to (A.4) we can write (see Section B.2)

$$g(A_n) = \sum_{j=1}^{n} g^j + \lambda \sum_{j=1}^{n-1} \sum_{k=j+1}^{n} g^j g^k + \cdots + \lambda^{n-1} g^1 \cdots g^n \tag{A.5}$$

$$g(A_n) = \left[ \prod_{x_i \in A} \left(1 + \lambda g^i\right) - 1 \right] \left(\frac{1}{\lambda}\right), \qquad \lambda \neq 0 \tag{A.6}$$

The value of $\lambda$ can then be found by solving the equation

$$g(A_n) = g(X) = 1 \tag{A.7}$$

From (A.6) and (A.7), this is equivalent to solving the following equation for $\lambda$:

$$\lambda + 1 = \prod_{i=1}^{n} \left(1 + \lambda g^i\right) \tag{A.8}$$

Hence, if we know the fuzzy densities $g^i$, $i = 1, \cdots, n$, we can construct the $\lambda$ - fuzzy measure. We first solve (A.8) for $\lambda$, and then compute the $g(A_i)$'s using (A.4).

5. For a fixed set of densities $\{g^i\}$, $0 < g^i < 1$, there exists a unique $\lambda \in (-1, \infty)$ where $\lambda \neq 0$ which satisfies (A.8).

6. Let $A_i$ be defined according to (A.3). For densities $\{g^i\}$, $0 < g^i < 1$, we have (see Section B.3)

$$0 \leq g(A_i) \leq 1 \ \forall i \tag{A.9}$$

with equality when $i = 0$ and $i = n$ , i.e., $g(A_0) = 0$ and $g(A_n) = 1$.

Estimating the individual fuzzy densities, $\{g^1, g^2, \ldots, g^n\}$, is an important problem in all applications. The behavior of fuzzy integrals (both the Choquet and the Sugeno) is heavily dependent on the choice of these fuzzy densities. In some applications it is possible to estimate these densities from training data [10]. For example, in a pattern recognition application where the output of different classifiers are fused, the densities could be the performance of the individual classifiers. Liang et al. [6] used a genetic algorithm to determine the fuzzy densities from training data.

## A.3 The Choquet Fuzzy Integral

Let $h$ be a measurable function

$$h : X \rightarrow [0, 1] \tag{A.10}$$

The Choquet fuzzy integral (CFI) defined below is the integral of $h$ with respect to a fuzzy measure $g_\lambda$. Note that in (A.10), $X$ could be a set of classifier outputs and $h(x)$ could be the soft output of the classifier (the confidence or evidence grade of the classifier) denoting that an input sample is from a particular class. In general, $X = \{x_1, \ldots, x_n\}$ is a set of information sources and $h(x_i)$ is the confidence grade of source $i$ that a particular hypothesis is true. A $\lambda$-fuzzy measure provides the importance of each subset of sources $X$ for this hypothesis evaluation.

To begin, we provide a definition of a fuzzy integral. Given a class of functions $F \subseteq \{h : X \rightarrow R\}$ and a class of fuzzy measures $m \subseteq M$, a functional

$$I : F \times m \rightarrow R \tag{A.11}$$

is a fuzzy integral [2]. Consider a specific function $h$ associated with fuzzy density $g_\lambda$. Then, we can define a fuzzy integral as

$$h, g \rightarrow I(h, g) \tag{A.12}$$

There are a number of families of fuzzy integrals in terms of the underlying fuzzy measures that have been described in the literature. We are particularly interested in the *Choquet Fuzzy Integral (CFI)* which is a nonlinear functional defined over measurable sets that combines multiple sources of uncertain information. It provides a computational scheme for aggregating information.

The CFI of $h$ over $X$ with respect to a fuzzy measure $g_\lambda$ is defined as

$$E_g(h) = \int_X h \circ g = \sum_{i=1}^{n} g(A_i) \left[ h\left(x_i\right) - h\left(x_{i+1}\right) \right] \tag{A.13}$$

where $h(x_1) \geq h(x_2) \geq \cdots \geq h(x_n)$ and $h(x_{n+1}) = 0$. Set $A$ is as defined in (A.3); i.e. $A_i = \{x_1, \ldots, x_i\} \subseteq X$, $g(A_n) = g(X)$, and $g(A_0) = 0$. The CFI can also be expressed as (see Section B.4)

$$E_g(h) = \int_X h \circ g = \sum_{i=1}^{n} h(x_i) \left[ g\left(A_i\right) - g\left(A_{i-1}\right) \right] \tag{A.14}$$

If the function $h$ is reordered such that $h(x_1) \leq h(x_2) \leq \cdots \leq h(x_n)$, then the CFI has the following form (see Section B.5) :

$$E_g(h) = \int_X h \circ g = \sum_{i=1}^{n} g\left(X - \{A_{i-1}\}\right) \left[ h\left(x_i\right) - h\left(x_{i-1}\right) \right] \tag{A.15}$$

$$= \sum_{i=1}^{n} g\left(x_i, x_{i+1}, \ldots, x_n\right) \left[ h\left(x_i\right) - h\left(x_{i-1}\right) \right] \tag{A.16}$$

All three forms of the CFI are identical. See Sections B.4 and B.5 to see how one form leads to another.

In comparison with probability theory, the CFI corresponds to the concept of expectation, and it has found extensive use in combining feature and algorithm confidence values [4].

Important properties of the CFI are (See Appendix B for their proofs):

1. The CFI is a monotonically increasing function with respect to $h(x)$ [2].

2. For all $h, g \in [0, 1]$, the range of the CFI is

$$h_{\min} \leq E_g(h) \leq h_{\max} \tag{A.17}$$

   where $h_{\min} = \min\left(h(x_1), h(x_2), \ldots, h(x_n)\right)$ and $h_{\max} = \max\left(h(x_1), h(x_2), \ldots, h(x_n)\right)$. The CFI attains its lower bound when $g_\lambda^i = 0$ for all $i$ and it attains its upper bound when $g_\lambda^i = 1$ for all $i$.

3. If $h(x_i) = c$ for all $i$, where $0 \leq c \leq 1$, then

$$\int_X h \circ g = c \tag{A.18}$$

4. If $h_1(x_i) \leq h_2(x_i)$ for all $i$, then

$$E_g(h_1) = \int_X h_1 \circ g \leq \int_X h_2 \circ g = E_g(h_2) \tag{A.19}$$

5. If $B \in X$, $C \in X$ and $B \subset C$, then

$$\int_B h \circ g \leq \int_C h \circ g \tag{A.20}$$

6. If the $\lambda$-fuzzy measure $g_\lambda$ is a probability measure, i.e. $\sum_i g_\lambda^i = 1$ and $\lambda = 0$, the CFI becomes a weighted average. In the special case where all the fuzzy density values are equal, the CFI is equivalent to the arithmetic mean. This corresponds to the case where $g_\lambda^i = 1/n$.

7. If $g_\lambda^j = 0$ for some $j$, then

$$E_g(h) = \int_X h \circ g = \sum_{i=1, i \neq j}^{n} h(x_i) \left[ g\left(A_i\right) - g\left(A_{i-1}\right) \right] \tag{A.21}$$

   This property shows that the CFI values are determined only by the input sources that have non-zero densities.

## A.4 Generic Applications of the Choquet Fuzzy Integral

The CFI aggregates the elements of the source information set $X$ according to a specified criterion, while incorporating the relative importance of each of the elements. In this section, we present the CFI for some simple "textbook" applications so as to foster a better understanding of the integral and gain insight into why it works in aggregation.

### A.4.1 Worker Productivity

Consider the example of productivity[2] in a workshop. Let $X = \{x_1, \ldots, x_n\}$ be a set of workers. Suppose that each worker $x_i$ works $h(x_i)$ hours a day from the opening hour. Without loss of generality, the function that defines the number of work hours for each worker is ordered such that $h(x_1) \leq h(x_2) \leq \cdots \leq h(x_n)$, where worker $x_1$ works the least amount of time and worker $x_n$ works the most; thus, for $i \geq 2$, $h(x_i) - h(x_{i-1}) \geq 0$.

The fuzzy measure is defined as the number of products made by the workers in one hour, with the implicit assumption that the productivity of the workers remains constant throughout the day; hence, $g(x_i)$ denotes the number of products made by worker $x_i$ in one hour, and $g$ is a measure of productivity. A group of workers $A \subset X$ produces the amount $g(A)$ in one hour. A product can be made either by one worker or by a group of workers.

[2]This example has been paraphrased from [8].

Hence, the number of products produced by 2 or more workers working together is larger than the sum of the products produced by each individual worker, if he were working alone.

Next we show that the CFI can be used to find the total number of products produced by all the workers in one work day. The working hours of all the workers are aggregated in the following way. First, the whole group $X$ with $n$ workers works $h(x_1)$ hours. Next, the group $X - \{x_1\} = \{x_2, x_3, \ldots, x_n\}$ works $h(x_2) - h(x_1)$ hours as the worker $x_1$ is no longer at work. Then, the group $X - \{x_1, x_2\} = \{x_3, x_4, \ldots, x_n\}$ works $h(x_3) - h(x_2)$ hours, and so on. Lastly, one worker $x_n$ works for $h(x_n) - h(x_{n-1})$ hours. Since group $A$ produces the amount $g(A)$ in one hour, the total number of products produced by all workers in one day can be expressed as:

$$\begin{aligned}
h(x_1)g(X) &+ [h(x_2) - h(x_1)]\, g\left(X - \{x_1\}\right) \\
&+ [h(x_3) - h(x_2)]\, g\left(X - \{x_1, x_2\}\right) + \cdots \\
&+ [h(x_n) - h(x_{n-1})]\, g\left(\{x_n\}\right) \\
&= \sum_{i=1}^{n} \left[h\left(x_i\right) - h\left(x_{i-1}\right)\right] g\left(\{x_i, x_{i+1}, \ldots, x_n\}\right) \quad \text{where } h(x_0) = 0 \\
&= \sum_{i=1}^{n} \left[h\left(x_i\right) - h\left(x_{i-1}\right)\right] g\left(X - A_{i-1}\right) \text{ where } A_i = \{x_1, \ldots, x_i\} \text{ and } A_0 = \{\emptyset\} \\
&\triangleq E_g(h) \qquad \text{(A.22)}
\end{aligned}$$

This example shows that (A.22) fits the definition of the CFI in (A.15), and demonstrates the aggregation logic behind the CFI.

### A.4.2 A Collection of Rare Books

Consider a particularly rare book[3] that comes in two volumes. The first and second volumes are denoted by $x_1$ and $x_2$, respectively. The fuzzy measure is defined as the price of the two volumes. The price of the first volume is given by $g\left(\{x_1\}\right)$, the price of the second volume by $g\left(\{x_2\}\right)$ and the price of the complete set by $g\left(\{x_1, x_2\}\right)$. The complete set is considered to be more valuable than the combination of the two volumes; hence, this fuzzy measure becomes a $\lambda-$fuzzy measure since

$$g\left(\{x_1, x_2\}\right) > g\left(\{x_1\}\right) + g\left(\{x_2\}\right) \qquad \text{(A.23)}$$

[3]This example has been paraphrased from [8].

A certain person sells $h(x_1)$ copies of the first volume and $h(x_2)$ copies of the second volume. Without loss of generality we can assume that $h(x_1) < h(x_2)$. The number of complete book sets (both volumes) sold is $h(x_1)$. The number of copies of the second volume sold separately is $h(x_2) - h(x_1)$. The total amount of money the seller gets is

$$h(x_1)\, g(\{x_1, x_2\}) + [h(x_2) - h(x_1)]\, h(x_2) \tag{A.24}$$

This expression is also similar to (A.15) and is another example of combining measurable functions with respect to densities using the CFI.

### A.4.3 Multiple Judges of a Sporting Event

A numerical example that demonstrates the calculations of the CFI, is presented next; it is adapted from [5].

Table 7.3: Scores for the participant $u$ from the five judges

| Unordered | | | Ordered | | |
|---|---|---|---|---|---|
| Judge $(x_i)$ | Score $(h(x_i))$ | Expertise $(g(x_i))$ | Judge $(x'_i)$ | Score $(h(x'_i))$ | Expertise $(g(x'_i))$ |
| 1 | 0.5 | 0.8 | 2 | 0.7 | 0.5 |
| 2 | 0.7 | 0.5 | 4 | 0.6 | 0.7 |
| 3 | 0.2 | 0.4 | 5 | 0.6 | 0.7 |
| 4 | 0.6 | 0.7 | 1 | 0.5 | 0.8 |
| 5 | 0.6 | 0.7 | 3 | 0.2 | 0.4 |

Let the set $X = \{x_1, \ldots, x_5\}$ represent five judges at a sporting event. Assume that the participant $u$ has obtained the scores shown in Table 7.3 from the $n = 5$ judges. Experts rate the judges' expertise, and their ratings are also shown in the table. A rating of 1 indicates that the judge is an expert while a rating of zero indicates a totally unknowledgeable judge (naturally a judge with a rating of zero would not be considered for aggregation). The expertise of the judges can be considered to be the densities $g_\lambda$ of the $\lambda-$fuzzy measure. The judges' scores need to be aggregated so as to determine the final score for the participant; hence, the scores become the values of the function $h$ aggregated in the CFI.

Given the fuzzy densities $g^i$ of the set $X$, $\lambda$ has to be computed using (A.8). Solving this equation with the density set $[0.8, 0.5, 0.4, 0.7, 0.7]$, we get a unique root $> -1$ which is $\lambda = -0.9943$. Next, the fuzzy measures of the power set can be determined by using (A.4).

The aggregate score is computed using the CFI by first ordering the scores and the corresponding densities such that the scores are in decreasing order. Prior to ordering, the

scores are assigned as $h(x_1) = 0.5$, $h(x_2) = 0.7$, ... , $h(x_5) = 0.6$. Let $X'$ be the ordered set so that $h(x'_1) \geq h(x'_2) \geq \cdots \geq h(x'_n)$. Then,

$$[0.5, 0.7, 0.2, 0.6, 0.6] \rightarrow [0.7, 0.6, 0.6, 0.5, 0.2] \quad \text{(scores)} \tag{A.25}$$

$$[0.8, 0.5, 0.4, 0.7, 0.7] \rightarrow [0.5, 0.7, 0.7, 0.8, 0.4] \quad \text{(densities)} \tag{A.26}$$

After ordering, the scores are assigned (see Table 7.3) as $h(x'_1) = 0.7$, $h(x'_2) = 0.6$, $\cdots$ , $h(x'_5) = 0.2$ and the corresponding densities are $g(x'_1) = 0.5$, $g(x'_2) = 0.7$ , ... , $g(x'_5) = 0.4$. The new arrangement of the judges (set $X'$) according to the re-ordering, is $X' = \{x'_1, x'_2, x'_3, x'_4, x'_5\} = \{x_2, x_4.x_5, x_1, x_3\}$. Let $A_i = \{x'_1, \ldots, x'_i\}$. The fuzzy measures for all $A_i$, $i = 1, \ldots, n$ are computed recursively according to (A.4), as:

$$\begin{aligned} g(A_1) &= 0.5 \\ g(A_2) &= 0.7 + 0.5 - 0.9943(0.7)0.5 = 0.8520 \\ g(A_3) &= 0.7 + 0.8520 - 0.9943(0.7)0.8520 = 0.9590 \\ g(A_4) &= 0.8 + 0.9590 - 0.9943(0.8)0.9590 = 0.9962 \\ g(A_5) &= 0.4 + 0.9962 - 0.9943(0.4)0.9962 = 1.0 \end{aligned} \tag{A.27}$$

Since $n = 5$, $g(A_5) = g(X) = 1.0$. The CFI can now be computed using (A.13) as:

$$\begin{aligned} E_g(h) = &(0.7 - 0.6)0.5 \\ &+(0.6 - 0.6)0.8520 \\ &+(0.6 - 0.5)0.9590 \\ &+(0.5 - 0.2)0.9962 \\ &+(0.2 - 0)1.0 \\ &= 0.64 \end{aligned} \tag{A.28}$$

According to the CFI, the aggregated score of participant $u$ is 0.64.

In this example we cannot use the arithmetic average as a tool for aggregation of the judges' scores because the sum of the densities $\sum_{i=1}^{5} g(x_i) = 3.10 > 1$. In order to compute the weighted average, we have to normalize the densities as

$$w_i = \frac{g(x_i)}{\sum_{i=1}^{5} g(x_i)} \tag{A.29}$$

The scores and the normalized weights are given in Table 7.3. The weighted average of the judges' scores can now be computed as

Table 7.3: Scores for participant $u$ from the five judges and their corresponding weights

| Judge ($x_i$) | Score ($h(x_i)$) | Normalized Expertise ($w_i$) |
|---|---|---|
| 1 | 0.5 | 0.26 |
| 2 | 0.7 | 0.16 |
| 3 | 0.2 | 0.13 |
| 4 | 0.6 | 0.225 |
| 5 | 0.6 | 0.225 |

$$W_g(h) = \sum_{i=1}^{5} h(x_i)\, w_i = 0.5(0.26) + 0.7(0.16) + 0.2(0.13) + 0.6(0.225) + 0.6(0.225) = 0.538 \tag{A.30}$$

If the normalized weights were to be considered as a $\lambda$-fuzzy measure, then $\lambda = 0$ and hence the CFI reduces to the weighted average (see Property 6 in Chapter A.3).

The aggregated result of the CFI is higher than that of the weighted average because it takes into account the increased *value* or in this case the increased expertise of two or more judges who agree on a particular score. This has occurred because of the computation of the $\lambda$-fuzzy densities for *subsets* of the set of judges $X$.

## A.5 The Sugeno Fuzzy Integral

Let $h$ be a measurable function $h : X \to [0, 1]$, one that is ordered such that $h(x_1) \geq h(x_2) \geq \cdots \geq h(x_n)$. Let $A_i = \{x_1, \ldots, x_i\} \subseteq X$ where $X = \{x_1, \ldots, x_n\}$, a finite set. The $A_i$'s form a sequence of nested sets $A_1, \ldots, A_n$, starting from $A_1 = \{x_1\}$, and then subsequently adding elements $x_2$ to $x_n$, one at a time to get $A_n = X$, as mentioned earlier in property 3 in Chapter A.2. Let $g^i = g_\lambda(\{x_i\})$ be the fuzzy densities of the set $X$. The Sugeno fuzzy integral (SFI) with respect to a fuzzy measure $g$ is given by [9]:

$$F_g(h) \;=\; \int_X h \circ g \;=\; \sup_\alpha \; \{t(\alpha, g(H_\alpha))\} \tag{A.31}$$

where $H_\alpha$ is the $\alpha$-cut of $h$ and $t$ is a $t$-norm.

The $\alpha$-cut of a fuzzy set $A$ on $U$ is the set $A_\alpha = \{u | \, u \in U, \, \mu_A(u) \geq \alpha\}$ where $\mu_A(u)$ is the membership function of $A$. In our case the $\alpha$-cut of $h$ is the set $H_\alpha = \{u | \, u \in X, \, h(u) \geq \alpha\}$. In Sugeno's original formula, the $t$-norm used was the minimum. In order to use (A.31), the $t$-norm should follow the distributive law under the supremum operator [4]. Examples

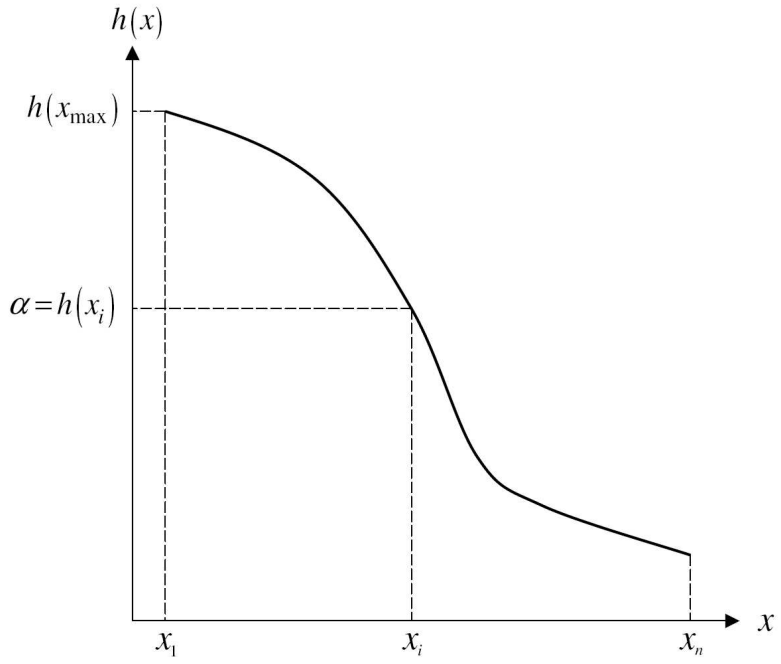



Figure A.1: Plot of $h(x)$ versus $x$. Note that the $\alpha$-cut shown in this figure is the set $A_i$. This is also the $h(x_i)^{th}$-cut. The continuous curve is for purposes of illustration only, i.e., it actually consists of lines connecting discrete values of $h(x_i)$ for $x = \{x_1, ..., x_n\}$.

of such $t$-norms are minimum, product, bounded difference, and drastic product.

Since $X$ is finite, $h$ has at most $n$ different $\alpha$-cuts ranging from $H_0 = X$ to $H_{height(H)}$. The latter only contains the element(s) that reach the maximum level of the function $h$. Since $h(x_1) \geq h(x_2) \geq \cdots \geq h(x_n)$ and $A_j = \{x_1, ..., x_j\} \subseteq X$, each $A_j \subseteq X$ is the $h(x_j)$-cut of $h$; thus, (A.31) can be expressed as

$$F_g(h) = \int_X h \circ g = \max_{j=1,...,n} \{t\,(h(x_j), g\,(A_j))\} \tag{A.32}$$

which is computationally simpler than (A.31). In Fig A.1 the $h(x_i)^{th}$-cut of $h$ is shown. This is the set $\{x_1, \ldots, x_i\}$ which we have defined to be $A_i$. In applications of the SFI to pattern recognition the $t$-norm most commonly used is the minimum; hence, the most common form of the SFI is given by

$$F_g(h) = \int_X h \circ g = \bigvee_j [h(x_j) \wedge g\,(A_j)] \tag{A.33}$$

where the values of $g(A_i)$ are determined recursively according to (A.4). Figure A.2 provides a graphical interpretation of the SFI, which attempts to find the best *consensus* between the function value and the importance attribute. This is where the SFI fundamentally differs from the CFI. While the CFI quantitatively weights the function by the *jump* in the fuzzy

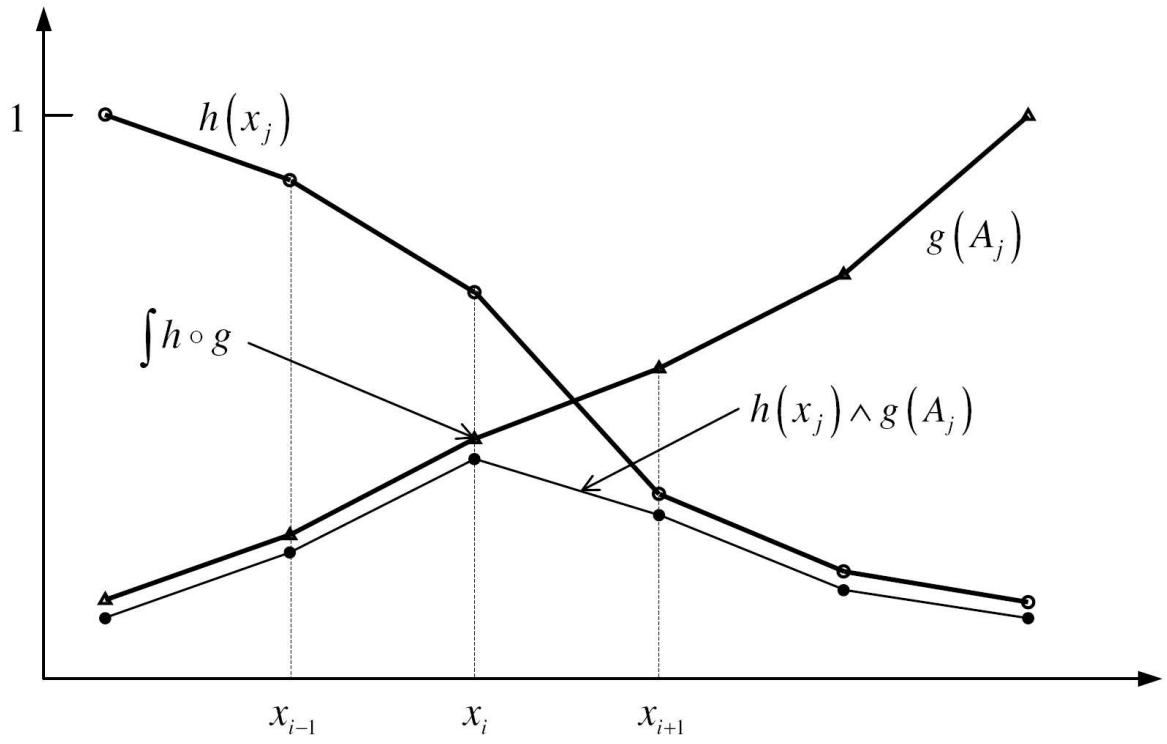



Figure A.2: Graphical illustration of the calculation of the SFI

measure, the SFI associates each function value with the corresponding importance measure.

Consider the worker-productivity problem in Section 4.A. The SFI finds the best consensus between the number of work hours and productivity whereas the CFI computes the improvement in productivity weighted by the number of hours. Clearly, the CFI makes more heuristic sense in this quantitative framework.

Next, consider the numerical example in Section 4.C where judges scored participants at a sporting event. Recall that participant $u$ has obtained the scores shown in Table 7.3 from the 5 judges; densities $g_\lambda$ of the $\lambda-$fuzzy measure are given in (A.27), and the fuzzy measures denote the expertise of the judges. The scores obtained for participant $u$ can now be aggregated as follows using the SFI so as to produce a final score. The scores are again reordered so that $h(x'_1) \geq h(x'_2) \geq \cdots \geq h(x'_n)$ where $X' = \{x'_1, x'_2.x'_3, x'_4, x'_5\} = \{x_2, x_4.x_5, x_1, x_3\}$ is the reordered set of judges as in (A.25), and the corresponding densities are also reordered as in (A.26). Using the definition of the SFI in (A.33) and the computed values of the $g(A_i)$ in (A.27) we find:

$$
\begin{aligned}
F_g(h) &= \max\left[\min\left(0.7, 0.5\right), \min\left(0.6, 0.8520\right), \cdots \right. \\
&\qquad \left. \min\left(0.6, 0.9590\right), \min\left(0.5, 0.9962\right), \min\left(0.2, 1\right)\right] \\
&= \max\left[0.5, 0.6, 0.6, 0.5, 0.2\right] \\
&= 0.6
\end{aligned}
$$

We see that the aggregated value of the scores $F_g(h)$, according to the stated fuzzy measure $g$ using the SFI is 0.6. The SFI computes the highest score that most judges agreed upon corresponding to their expertise.

In general, the CFI is used when the problem framework involves quantitative measures and the SFI is used when qualitative measures (e.g., expertise, accuracy, talent etc) define the importance attribute. An application where quantitative measures are the densities, and hence the CFI is used, is described in Section 6.C.

The Generalized Sugeno Fuzzy Integral (GSFI) is an extended version of the SFI that is formed when each $h(x_i)$ is not a real value in $[0, 1]$ but is a fuzzy number within the universal set $[0, 1]$ and was presented by Auephanwiriyakul, et al. [1]. In the case of the numerical example just presented, if the judges had scored the participants qualitatively (e.g., good, excellent), then the scores would themselves be fuzzy. In this case the GSFI would be used to aggregate the 8scores. The output of the GSFI is a type-1 fuzzy set.

# Appendix B

# Derivation of Some of the Properties of Fuzzy Measures

## B.1 Derivation of (A.4)

We know from (A.2) and (A.3) that for a $\lambda$-fuzzy measure

$$g(A_1) = g(\{x_1\}) = g^1 \tag{B.1}$$

$$g(A_2) = g(\{x_1, x_2\}) = g(\{x_1\} \cup \{x_2\}) = g^1 + g^2 + \lambda g^1 g^2 \tag{B.2}$$

$$g(A_3) = g(\{x_1, x_2, x_3\}) = g(\{x_1, x_2\} \cup \{x_3\}) = g(A_2) + g^3 + \lambda g^3 g(A_2) \tag{B.3}$$

From (B.1), (B.2), and (B.3) and extrapolating to $n$, we find

$$g(A_n) = g(A_{n-1} \cup \{x_n\}) = g^n + g(A_{n-1}) + \lambda g^n g(A_{n-1}) \tag{B.4}$$

Note that $g(A_n) = g(X) = 1$, and $g(A_0) = \emptyset = 0$ (by definition); therefore, we can state in general that

$$g(A_i) = g(A_{i-1} \cup \{x_i\}) = g^i + g(A_{i-1}) + \lambda g^i g(A_{i-1}) \quad \text{for } 1 \leq i \leq n \tag{B.5}$$

## B.2 Derivation of (A.6)

Eqn (B.2) can be written as

$$g\left(A_2\right) = \sum_{j=1}^{2} g^j + \lambda g^1 g^2 \tag{B.6}$$

Multiplying and dividing by $\lambda$ and then adding and subtracting 1 we get

$$g\left(A_2\right) = \left(\frac{1}{\lambda}\right)\left[1 + \lambda g^1 + \lambda g^2 + \lambda^2 g^1 g^2 - 1\right] \tag{B.7}$$

$$= \left(\frac{1}{\lambda}\right)\left[\left(1 + \lambda g^1\right)\left(1 + \lambda g^2\right) - 1\right] \tag{B.8}$$

$$= \left(\frac{1}{\lambda}\right)\left[\prod_{i=1}^{2}\left(1 + \lambda g^i\right) - 1\right] \tag{B.9}$$

Similarly, for $g\left(A_3\right)$, substituting (B.9) in (B.3), we see that

$$g\left(A_3\right) = \left(\frac{1}{\lambda}\right)\left[\prod_{i=1}^{2}\left(1 + \lambda g^i\right) - 1\right] + g^3 + \lambda g^3\left(\left(\frac{1}{\lambda}\right)\left[\prod_{i=1}^{2}\left(1 + \lambda g^i\right) - 1\right]\right) \tag{B.10}$$

$$= \left(\left(\frac{1}{\lambda}\right)\left[\prod_{i=1}^{2}\left(1 + \lambda g^i\right) - 1\right]\right)\left(1 + \lambda g^3\right) + g^3 \tag{B.11}$$

$$= \left(\frac{1}{\lambda}\right)\left[\prod_{i=1}^{3}\left(1 + \lambda g^i\right)\right] - \left(\frac{1}{\lambda}\right)\left(1 + \lambda g^3\right) + g^3 \tag{B.12}$$

$$= \left(\frac{1}{\lambda}\right)\left[\prod_{i=1}^{3}\left(1 + \lambda g^i\right) - 1\right] \tag{B.13}$$

For $n = k$ , we assume that

$$g\left(A_k\right) = \left(\frac{1}{\lambda}\right)\left[\prod_{i=1}^{k}\left(1 + \lambda g^i\right) - 1\right] \tag{B.14}$$

Now, for $n = k + 1$, we have (according to (B.5))

$$g\left(A_{k+1}\right) = g^{k+1} + g\left(A_k\right) + \lambda g^{k+1} g\left(A_k\right) \tag{B.15}$$

$$= \left(\frac{1}{\lambda}\right)\left[\prod_{i=1}^{k}\left(1+\lambda g^i\right)-1\right] + g^{k+1} + \lambda g^{k+1}\left(\left(\frac{1}{\lambda}\right)\left[\prod_{i=1}^{k}\left(1+\lambda g^i\right)-1\right]\right) \tag{B.16}$$

$$= \left(\left(\frac{1}{\lambda}\right)\left[\prod_{i=1}^{k}\left(1+\lambda g^i\right)-1\right]\right)\left(1+\lambda g^{k+1}\right) + g^{k+1} \tag{B.17}$$

$$= \left(\frac{1}{\lambda}\right)\left[\prod_{i=1}^{k+1}\left(1+\lambda g^i\right)-1\right] - \left(\frac{1}{\lambda}\right)\left(1+\lambda g^{k+1}\right) + g^{k+1} \tag{B.18}$$

$$= \left(\frac{1}{\lambda}\right)\left[\prod_{i=1}^{k+1}\left(1+\lambda g^i\right)-1\right] \tag{B.19}$$

From (B.9) , (B.14), (B.19) and the induction hypothesis, we conclude that

$$g\left(A_n\right) = \left[\prod_{i=1}^{n}\left(1+\lambda g^i\right)-1\right]\left(\frac{1}{\lambda}\right), \qquad \lambda \neq 0 \tag{B.20}$$

## B.3 Derivation of (A.9)

From the definition of the fuzzy measure (see (A.1)) and (A.7), we have $0 < g^i < 1$ and $g(X) = g(A_n) = 1$. From (A.4), we get

$$g(A_{i-1}) = \frac{g(A_i) - g^i}{(1+\lambda g^i)} \tag{B.21}$$

When $i = n$, (B.21) becomes

$$g(A_{n-1}) = \frac{1-g^n}{(1+\lambda g^n)} \tag{B.22}$$

Since $\lambda > -1$, $0 < g(A_{n-1}) < 1$ and consequently $0 < g(A_{n-2}) < 1$ [by substituting $i = n-1$ into (B.21)]. Similarly, $0 < g(A_i) < 1\ \forall i$. Equality is obtained when $i = 1$, i.e., $g(A_n) = 1$ [by (A.7 )].

## B.4 Derivation of (A.14)

Expand (A.13) for $k-1$ and $k$ as follows:

$$E_g(h) = \cdots + g\left(A_{k-1}\right)\left[h\left(x_{k-1}\right) - h\left(x_k\right)\right] + g\left(A_k\right)\left[h\left(x_k\right) - h\left(x_{k+1}\right)\right] + \cdots \tag{B.23}$$

Collect terms with $h(x_k)$ as the common factor, to obtain

$$E_g(h) = \cdots + h\left(x_{k-1}\right) g\left(A_{k-1}\right) + h\left(x_k\right)\left[g\left(A_k\right) - g\left(A_{k-1}\right)\right] + h\left(x_{k+1}\right)\left[-g\left(A_k\right)\right] \cdots \tag{B.24}$$

which when summed leads to (A.14).

## B.5 Derivations of (A.15) and (A.16)

Let $Z = \{z_1, \ldots, z_n\}$ such that $z_1 = x_n, z_2 = x_{n-1}, \ldots, z_n = x_1$. Because $h(x_1) \leq h(x_2) \leq \cdots \leq h(x_n)$, it follows that $h(z_1) \geq h(z_2) \geq \cdots \geq h(z_n)$. This one-to-one correspondence between $X$ and $Z$ can be specified by the relation $x_i = z_{n-i+1}$ . Replacing $x$ by $z$ in (A.16), we find

$$E_g(h) = \sum_{i=1}^{n} g\left(\{z_{n-i+1}, z_{n-i}, \ldots, z_1\}\right)\left[h\left(z_{n-i+1}\right) - h\left(z_{n-i+2}\right)\right] \tag{B.25}$$

Letting $j \equiv n - i + 1$, and $B_j \equiv \{z_1, \ldots, z_j\}$, (B.25) can be expressed as:

$$E_g(h) = \sum_{j=1}^{n} g\left(\{z_j, z_{j-1}, \ldots, z_1\}\right)\left[h\left(z_j\right) - h\left(z_{j+1}\right)\right] \tag{B.26}$$

$$= \sum_{j=1}^{n} g\left(B_j\right)\left[h\left(z_j\right) - h\left(z_{j+1}\right)\right] \tag{B.27}$$

which is in exactly the form of (A.13). This proves that (A.16) is a form that is equivalent to the CFI in (A.13). Finally (A.16) is equivalent to (A.15) because $X = \{x_1, x_2, \ldots, x_n\}$ and $A_i = \{x_1, x_2, \ldots, x_i\}$ and so, $g\left(X - \{A_{i-1}\}\right) = g\left(x_i, x_{i+1}, \ldots, x_n\right)$.

# Appendix C

# Proofs of the Properties of the CFI

## C.1 Property 1

Let us take the partial derivative of CFI in (A.14) with respect to $h(x_i)$ for any $i$, i.e.

$$\frac{\partial E_g(h)}{\partial h(x_i)} = g(A_i) - g(A_{i-1}) \tag{C.1}$$

Using (A.4) to expand $g(A_i)$, (C.1) can be expressed as

$$\frac{\partial E_g(h)}{\partial h(x_i)} = g^i + \lambda g^i g(A_{i-1}) = g^i \left(1 + \lambda g\left(A_{i-1}\right)\right) \tag{C.2}$$

Because $g^i > 0$ , $0 \leq g\left(A_i\right) \leq 1$ and $\lambda \geq -1$, the expression $g^i \left(1 + \lambda g\left(A_{i-1}\right)\right) \geq 0$, which proves that the CFI is a monotonically increasing function with respect to $h(x)$.

## C.2 Property 2

When $g^i = 1$ for all $i = 1, 2, \ldots, n$, then $g(A_i) = 1$ for all $i = 1, 2, \ldots, n$ (from (A.7) and (A.4)); therefore, the CFI becomes (using (A.14))

$$E_g(h) = \sum_{i=1}^{n} h\left(x_i\right)\left[g\left(A_i\right) - g\left(A_{i-1}\right)\right] = h\left(x_1\right) \tag{C.3}$$

since $g(A_0) = 0$. Because $h(x_1) \geq h(x_2) \geq \cdots \geq h(x_n)$, it follows that

$$h(x_1) = \max\left(h(x_1), h(x_2), \ldots, h(x_n)\right) \tag{C.4}$$

When $g^i = 0$ for all $i$, then, $g(A_n) = 1$, $g(A_i) = 0$ for all $i \neq n$ (from (A.7) and (A.4)). The CFI then becomes (using (A.14))

$$E_g(h) = \sum_{i=1}^{n} h\left(x_i\right)\left[g\left(A_i\right) - g\left(A_{i-1}\right)\right] = h\left(x_n\right) \tag{C.5}$$

Because $h(x_1) \geq h(x_2) \geq \cdots \geq h(x_n)$, it follows that

$$h(x_n) = \min\left(h(x_1), h(x_2), \ldots, h(x_n)\right) \tag{C.6}$$

When $0 \leq g^i \leq 1$ for all $i$, then $g(A_n) = 1$, and $0 \leq g(A_i) \leq 1$ for all $i$. Since the maximum value of $h(x_1)$ is obtained only when all the density values are one, in all other cases $E_g(h) \leq x_{\max}$. Similarly, $x_{\min} \leq E_g$ since the minimum value is obtained only when all the densities equal zero.

## C.3 Property 3

Using (A.13), $g(A_n) = 1, \text{and}\, h(x_{n+1}) \equiv 0$, we have

$$E_g(h) = \sum_{i=1}^{n} g\left(A_i\right)\left[h\left(x_i\right) - h\left(x_{i+1}\right)\right] = g\left(A_n\right) h\left(x_n\right) = h\left(x_n\right) = c \tag{C.7}$$

since all the other terms vanish because the $h$ terms cancel as $i$ ranges from 1 to $n$.

## C.4 Property 4

In (A.14), let $a_i = g\left(A_i\right) - g\left(A_{i-1}\right)$ for all $i$. In (A.19) $g$ is the same for both $h_1$ and $h_2$; hence,

$$\int_X h_1 \circ g = E_g(h_1) = \sum_{i=1}^{n} h_1\left(x_i\right)\left[g\left(A_i\right) - g\left(A_{i-1}\right)\right] = h_1\left(x_1\right) a_1 + h_1\left(x_2\right) a_2 + \cdots + h_1\left(x_n\right) a_n$$

$$\leq h_2\left(x_1\right) a_1 + h_2\left(x_2\right) a_2 + \cdots + h_2\left(x_n\right) a_n = \int_X h_2 \circ g \tag{C.8}$$

since $h_1(x_i) \leq h_2(x_i)$ for all $i$.

## C.5 Property 5

Let $g^i(i = 1, \ldots n)$ be the fuzzy densities of the universal set $X$, $B = \{x_1, \ldots, x_l\} \subset X$, $C = \{x_1, \ldots, x_m\}$ and $l < m \leq n$. Then using (A.13), we see that

$$\int_C h \circ g = \sum_{i=1}^{m} g\left(A_i\right)\left[h\left(x_i\right) - h\left(x_{i+1}\right)\right]$$

$$= \sum_{i=1}^{l} g\left(A_i\right)\left[h\left(x_i\right) - h\left(x_{i+1}\right)\right] + \sum_{i=l+1}^{m} g\left(A_i\right)\left[h\left(x_i\right) - h\left(x_{i+1}\right)\right]$$

$$= \int_B h \circ g + \sum_{i=l+1}^{m} g\left(A_i\right)\left[h\left(x_i\right) - h\left(x_{i+1}\right)\right] \geq \int_B h \circ g \tag{C.9}$$

because $g(A_i) \geq 0$ and $h\left(x_i\right) - h\left(x_{i+1}\right) \geq 0$.

## C.6 Property 6

Given that

$$g(A_n) = 1 = \sum_{j=1}^{n} g^j + f(\lambda) \tag{C.10}$$

where $f(\lambda)$ is a function of $\lambda$ as in (A.5), and $\sum_{j=1}^{n} g^j = 1$, we see that $\lambda = 0$ for this equation to be satisfied. In this case, the $g(A_i)$'s become additive measures, i.e., $g(A_i) = \sum_{j=1}^{i} g^j$, and the CFI in (A.13) simplifies to (refer to (A.5)):

$$E_g(h) = \sum_{i=1}^{n} h\left(x_i\right)\left[g\left(A_i\right) - g\left(A_{i-1}\right)\right] = \sum_{i=1}^{n} h\left(x_i\right)\left[\sum_{j=1}^{i} g^j - \sum_{j=1}^{i-1} g^j\right] \tag{C.11}$$

$$= \sum_{i=1}^{n} h\left(x_i\right) g^i \tag{C.12}$$

which is a weighted average. In the specific case when the densities all equal $1/n$, (C.12) simplifies further to:

$$E_g(h) = \frac{1}{n} \sum_{i=1}^{n} h\left(x_i\right) \tag{C.13}$$

which is the arithmetic mean.

## C.7 Property 7

If $g^j = 0$, then according to (A.4)

$$g(A_j) = g^j + g(A_{j-1}) + \lambda g^j g(A_{j-1}) = g(A_{j-1}) \tag{C.14}$$

In this case, (A.14) becomes

$$E_g(h) = \sum_{i=1}^{n} h(x_i) \left[ g\left(A_i\right) - g\left(A_{i-1}\right) \right] = \sum_{i=1, i \neq j}^{n} h(x_i) \left[ g\left(A_i\right) - g\left(A_{i-1}\right) \right] \tag{C.15}$$

since the $g(A_i)$ terms cancel out when $i = j$.

# Acknowledgements

The effort reported on was sponsored by the Department of Army Research Office, Project W911 NF-05-02-0008. The content of the information does not necessarily reflect the position or policy of the federal government, and no official endorsement should be inferred.

# Bibliography


[1] S. Auephanwiriyakul, J. M. Keller, and P. D. Gader, "Generalized Choquet fuzzy integral fusion," *Information Fusion*, vol. 3, no. 1, pp. 69–85, 2002.

[2] J.-H. Chiang, "Choquet fuzzy integral-based hierarchical networks for decision analysis," *IEEE Trans on Fuzzy Systems*, vol. 7, pp. 63–71, 1999.

[3] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd ed. New York: John Wiley & Sons, Inc., 2001.

[4] J. Keller, P. Gader, H. Tahani, J. H. Chiang, and M. Mohamed, "Advances in fuzzy integration for pattern recognition," *Fuzzy Sets and Systems*, vol. 65, pp. 273–283, 1994.

[5] L. I. Kuncheva, *Fuzzy Classifier Design.* New York: Physica-Verlag, 2000.

[6] J. Liang, W. Yang, and X. Cai, "Decision fusion using the fuzzy integral method," *Proc of SPIE Conf. Sensor Fusion: Architecture, Algorithms, and Applications IV*, pp. 468–475, 2000.

[7] J. M. Mendel, *Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions.* Upper Saddle River, NJ: Prentice Hall, 2001.

[8] T. Murofushi and M. Sugeno, "An interpretation of fuzzy measures and the Choquet integral as an integral with respect to a fuzzy measure," *Fuzzy Sets and Systems*, vol. 29, pp. 201–227, 1989.

[9] M. Sugeno, "Fuzzy measures and fuzzy integrals - a survey," *Fuzzy Automata and Decision Processes*, pp. 89–102, 1977.

[10] H. Tahani and J. Keller, "Information fusion in computer vision using the fuzzy integral," *IEEE Trans on Systems, Man and Cybernetics*, vol. 20, pp. 733–741, 1990.

[11] H. Wu and J. M. Mendel, "Multi-category classification of ground vehicles based on the acoustic data using fuzzy logic rule-based classifiers," Signal and Image Processing Institute, University of Southern California, Tech. Rep. 360, 2003.