Master of Technology

# Computational Intelligence II

# Fuzzy Decision Making

Dr. TIAN Jing
Institute of Systems Science,
National University of Singapore
Email: tianjing@nus.edu.sg

# Multiple objective decision making

- Single objective decision making: such as minimizing cost, maximizing profit, and minimizing run time.
- Multiple objective decision making
  - » Design a new computer, and simultaneously minimize cost, maximize CPU, maximize random access memory (RAM), and maximize reliability.
  - » Suppose cost is the most important of our objectives and the other three (CPU, RAM, and reliability) carry lesser but equal weight when compared with cost.
- Two primary issues in multi-objective decision making
  - » Acquire meaningful information regarding the satisfaction of the objectives by the various choices (alternatives).
  - » Rank or weight the relative importance of each of the objectives.

# Fuzzy decision making

- Considering a situation of single or multi criteria decision-making with uncertainty, where
  - » activities and observations are ill defined, i.e.: differ from each other only vaguely
  - » objectives and constraints are fuzzy in nature
- Such situation can be formulated as fuzzy decision making problems
  - » It is required to satisfy (optimize) the fuzzy objectives as well as the fuzzy constraints, and
    - ♦ the decision is also fuzzy

# Fuzzy decision making

- A "decision" in a fuzzy environment is defined as the selection of options

  > achieving the fuzzy objectives and meeting fuzzy constraints at the same time

- The main advantage of the fuzzy formulation compared to the crisp formulation is that
  - » the decision maker is not forced to give a precise formulation, for the sake of mathematical reasons, even though he or she might be able or willing to describe the problem in fuzzy terms.

# Fuzzy decision making

- Fuzzy goals (objectives)
- Fuzzy constraints
- Fuzzy aggregation on objective functions
- Fuzzy aggregation on relation constraints
- Fuzzy decision, can be viewed as the intersection of fuzzy objectives and fuzzy constraints.

Representation of decision problem → Fuzzy set evaluation of decision alternatives → Selection of optimal alternative

# Fuzzy decision making: Formulation

- A set of options notated using $X_{op} = \{x_1, \ldots, x_n\}$
- A fuzzy objective is a fuzzy set $G$ defined in $X_{op}$

$$\mu_G : \quad X_{op} \rightarrow [0, 1]$$

- A fuzzy constraint is a fuzzy set $C$ also defined in $X_{op}$
- A fuzzy decision $D$ is a fuzzy set created as a result of intersection of fuzzy objective and fuzzy constraint:

$$D = G \cap C$$

while $\mu_D(x) = T\{\mu_G(x), \mu_C(x)\}$ for each $x \in X_{op}$ (*T* is T-norm)

- A maximization decision is the option $x^* \in X_{op}$ such that

$$\mu_D(x^*) = \max_{x \in X_{op}} \mu_D(x)$$

# Fuzzy decision making: Bellman-Zadeh model

- 'Decision' is defined as the confluence of goals and constraints
  - Given $n$ fuzzy goals (objectives) $\widetilde{G}_i(i = 1,2,\cdots,n)$ and $m$ fuzzy constraints $\widetilde{C}_j(j = 1,2,\cdots,m)$
  - Then $\widetilde{G}_i(i = 1,2,\cdots,n)$ and $\widetilde{C}_j(j = 1,2,\cdots,m)$ are combined to form a "decision", a fuzzy set $\widetilde{D}$ resulting from the intersection of fuzzy goals and fuzzy constraints as $\widetilde{D} = \widetilde{G_1} \cap \cdots \cap \widetilde{G_n} \cap \widetilde{C_1} \cap \cdots \cap \widetilde{C_m}$ , and $\mu_{\widetilde{D}} = \left(\mu_{\widetilde{G_1}} \times \cdots \times \mu_{\widetilde{G_n}}\right) \times \left(\mu_{\widetilde{C_1}} \times \cdots \times \mu_{\widetilde{C_m}}\right)$ , where $\times$ is logical "*and*" and represented by $min$ operator.

# Other useful operators

- The "fuzzy and" operator is defined as:

$$\mu_{\tilde{A}_i, and}(x) = \gamma \cdot \min_i \{\mu_i(x)\} + (1-\gamma) \cdot \sum_{i=1}^{m} (\mu_i(x)) / m, \quad x \in X, \quad 0 \leq \gamma \leq 1$$

- The "compensatory and" operator is defined as:

$$\mu_{\tilde{A}_i, comp}(x) = \left( \prod_{i=1}^{m} \mu_i(x) \right)^{(1-\gamma)} \left( 1 - \prod_{i=1}^{m} (1 - \mu_i(x)) \right)^{\gamma}, \quad x \in X, \ 0 \leq \gamma \leq 1$$

- The "convex combination of min and max" operator

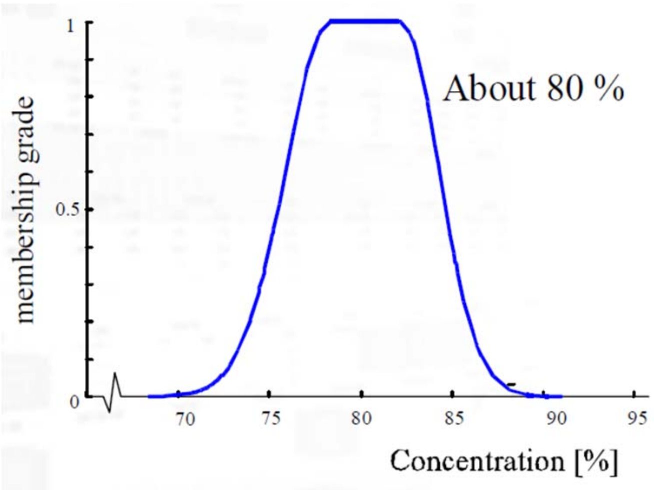
$$\mu_{\tilde{A}_i, \min - \max}(x) = \gamma \cdot \min_i \{\mu_i(x)\} + (1-\gamma) \cdot \max_i \{\mu_i(x)\} \quad x \in X, \quad 0 \leq \gamma \leq 1$$

where $\gamma$ is chosen arbitrarily; $\widetilde{A_i}$ are $m$ fuzzy sets with membership functions $\mu_i(x)$.

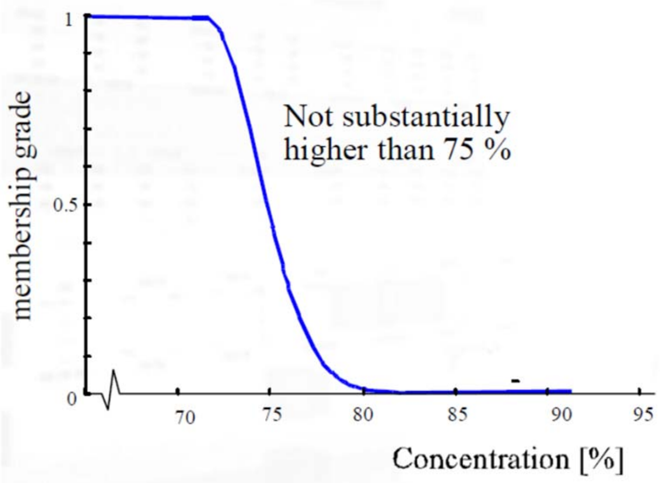
*Note: these are for reference only, not discussed in class

# A toy example

**Goal**: "Product concentration should be *about* 80%".



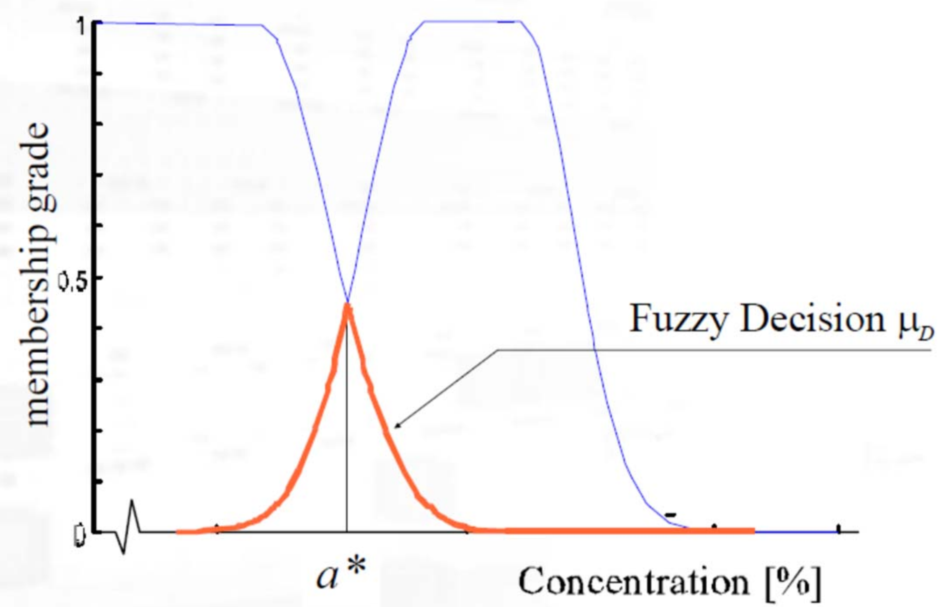

**Constraint**: "Product concentration should be *not substantially higher* than 75%".




Maximizing decision using **min**:

**Decision**

$$a^* = \arg\max \mu_D$$



# Example 1

- Example: college selection
  - » A talented student applied to several colleagues and after exams, he was accepted to 4 of the colleges that make up the set of options $X_{\mathrm{op}} = \{x_1, x_2, x_3, x_4\}$.
- Goal: $G$ = "to study at a good college"
- Constraints:

$C_1$ = "not far from the place of residence",
$C_2$ = "international exchange program",
$C_3$ = "good technical facilities",
$C_4$ = "high odds to find good job".

# Example 1

- Assume the fuzzy goal and fuzzy constraints are given as

$$G = \frac{0.75}{x_1} + \frac{1}{x_2} + \frac{0.25}{x_3} + \frac{0.5}{x_4}$$

$$C_1 = \frac{0.8}{x_1} + \frac{0.9}{x_2} + \frac{0.4}{x_3} + \frac{0.5}{x_4}, \quad C_2 = \frac{0.2}{x_1} + \frac{0.2}{x_2} + \frac{0.9}{x_3} + \frac{0.6}{x_4}$$

$$C_3 = \frac{0.5}{x_1} + \frac{0.3}{x_2} + \frac{0.6}{x_3} + \frac{0.7}{x_4}, \quad C_4 = \frac{0.6}{x_1} + \frac{0.4}{x_2} + \frac{0.7}{x_3} + \frac{0.7}{x_4}$$

  - By $D = G \cap C_1 \cap C_2 \cap C_3 \cap C_4$

    we obtain the fuzzy decision (using $min$ as T-norm):

$$D = \frac{0.2}{x_1} + \frac{0.2}{x_2} + \frac{0.25}{x_3} + \frac{0.5}{x_4}$$

Decision: college $x_4$ with greatest membership degree

# Example 2

- Example: Job selection
- Given four different jobs *a, b, c*, and *d*. The annual salaries of the four jobs are given by

  $a$ = 30,000

  $b$ = 25,000

  $c$ = 20,000

  $d$ = 15,000
- Goal: *high* salary
- Constraint: *interesting* and *close* driving distance

# Example 2

- The fuzzy goal $G$ of a high salary is defined on the universal set $X$ of salaries by the membership function

$$\mu_G(x) = \begin{cases} 0 & \text{for } x < 13{,}000 \\ 1 - 0.00125\,(x/1000 - 40)^2 & \text{for } 13{,}000 \leq x \leq 40{,}000 \\ 1 & \text{for } x > 40{,}000 \end{cases}$$

Substituting four options a, b, c, d into above membership function, we get the corresponding goal $G'$ induced on the set of alternative jobs as $G' = 0.875/a + 0.7/b + 0.5/c + 0.2/d$

# Example 2

- The first constraint of interesting is represented by the fuzzy set $C_1$ defined on the universal set of alternative jobs:

$$C_1 = 0.4/a + 0.6/b + 0.8/c + 0.6/d$$

- The second constraint of close driving distance is represented by the fuzzy set $C_2$ defined on the same universal set of alternative jobs:

$$C_2 = 0.1/a + 0.9/b + 0.7/c + 1/d$$

- A fuzzy decision $D$ can be obtained as (using *min* for intersection)

$$D = G' \cap C_1 \cap C_2 = 0.1/a + 0.6/b + 0.5/c + 0.2/d$$

Master of Technology

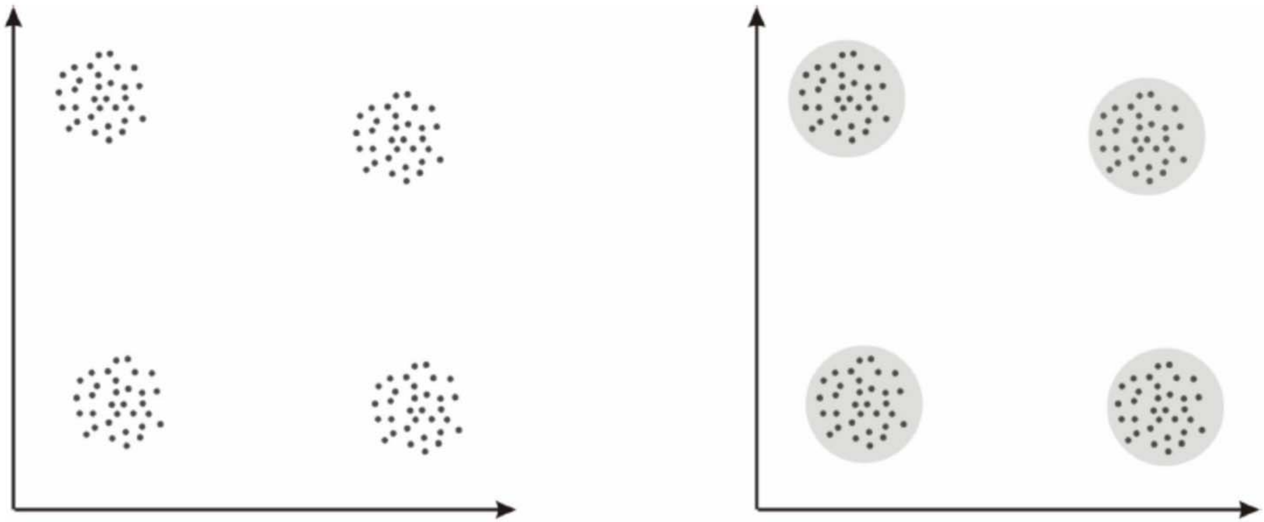
# Computational Intelligence II

# Fuzzy Clustering

Dr. TIAN Jing
Institute of Systems Science,
National University of Singapore
Email: tianjing@nus.edu.sg

# Recap: C-means clustering

- K-means clustering partitions a collection of $n$ vectors (data points) $x_k$, $k = 1, \ldots, n$, into predetermined $c$ groups $G_i$, $i = 1, \ldots, c$, and finds a cluster centre in each group such that a cost function of dissimilarity measure is minimized.



# C-means clustering

- Dissimilarity measure

$$J = \sum_{i=1}^{c} J_i = \sum_{i=1}^{c} \left( \sum_{k, x_k \in G_i} \|x_k - c_i\|^2 \right)$$

where $\|x_k - c_i\|^2$ is the square of Euclidean distance, $x_k$ is input vector, $c_i$ is cluster centre, $G_i$ is the *i*-th group (cluster), *k=1, …, n, i=1, …, c.*

# C-means clustering

- Partitioned groups are defined by $c \times n$ binary membership matrix

$$\mathbf{U} = \begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1n} \\ u_{21} & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots \\ u_{c1} & \cdots & \cdots & u_{cn} \end{bmatrix}$$

where

$u_{ik} = 1$ if the $k$-th data point $x_k$ belongs to group $i$

(i.e. $\|x_k - c_i\|^2 \leq \|x_k - c_j\|^2$, for $j \neq i$ )

$u_{ik} = 0$ otherwise

- A cluster centre $c_i$ is the mean of all vectors in the group $i$

$$\boldsymbol{c}_i = \frac{1}{|G_i|} \sum_{k, x_k \in G_i} x_k$$

# C-means clustering

Given: a data set $\{x_k\}$, $k = 1, \ldots, n$

- Step 1: Initialize the cluster $c_i$, $i = 1, \ldots, c$. This is typically achieved by *randomly* selecting *c* points from among all of the data points.
- Step 2: Determine the membership matrix U.
- Step 3: Compute the cost function (such as dissimilarity measure). Stop if either it is below a certain tolerance value or its improvement over previous iteration is below a certain threshold.
- Step 4: Update the cluster centres. Go to Step 2.

# Fuzzy C-means clustering

- Fuzzy C-means clustering
  - » An improvement over earlier *hard C-means* (HCM).
  - » Each data point belongs to a cluster to a degree specified by a membership grade.
- The major difference between FCM and HCM
  - » FCM employs fuzzy partitioning such that a given data point can belong to several groups with the degree of belongingness specified by membership grade.
  - » The membership matrix U is allowed to have elements with membership values between 0 and 1.

# Fuzzy C-means clustering

- The summation of degrees of belongingness for each data point always be equal to unity (also applies to c-means clustering)

$$\sum_{i=1}^{c} u_{ik} = 1, \qquad \forall k = 1,\dots,n$$

- The cost function

$$J = \sum_{i=1}^{c} J_i = \sum_{i=1}^{c} \left( \sum_{k, x_k \in G_i} \mu_{ik}^{m} \| x_k - c_i \|^2 \right)$$

where, $\| x_k - c_i \|^2$ is the square distance between *i*-th cluster centre and *k*-th data point, *m>1* is a weighting exponent governs the influence of membership grades.

# Fuzzy C-means clustering

- FCM determines the cluster centres $c_i$ and the membership matrix *U* using the following steps:
  - » Step 1: Initialise the membership matrix U with random values between 0 and 1 such that

$$\sum_{i=1}^{c} u_{ik} = 1, \quad \forall k = 1,\ldots,n$$ is satisfied

  - » Step 2: Calculate fuzzy cluster centres $c_i$ using

$$\boldsymbol{c}_i = \frac{\sum_{k=1}^{n} u_{ik}^{m} x_k}{\sum_{k=1}^{n} u_{ik}^{m}}$$

# Fuzzy C-means clustering

- Step 3: Compute the cost function. Stop if either it is below a certain tolerance value or its improvement over previous iteration is below a certain threshold.
- Step 4: Compute a new U. Go to Step 2. The element $u_{ik}$ in the membership matrix is computed as

$$\mu_{ik} = \frac{1}{\sum_{j=1}^{c} \left( \frac{\|x_k - c_i\|^2}{\|x_k - c_j\|^2} \right)^{2/(m-1)}}$$

# Example

- Consider a dataset of six points given in the following tables, each of which has two features $f_1$, and $f_2$. The parameters of FCM algorithm are set to be $c = 2, m = 2$, the initial cluster centres are $v_1 = (5,5)$, $v_2 = (10,10)$. Apply the FCM algorithm to find the new cluster centre **after ONE iteration**.

| | $f_1$ | $f_2$ |
|---|---|---|
| $x_1$ | 2 | 12 |
| $x_2$ | 4 | 9 |
| $x_3$ | 7 | 13 |
| $x_4$ | 11 | 5 |
| $x_5$ | 12 | 7 |
| $x_6$ | 14 | 4 |

# Example

The initial membership values of the data point, $x_1$, in the two clusters can be calculated as

$$\mu_{c1}(x_1) = \frac{1}{\sum_{j=1}^{2}\left(\frac{\|x_1 - v_1\|^2}{\|x_1 - v_1\|}\right)^2}$$

$$\|x_1 - v_1\|^2 = 3^2 + 7^2 = 9 + 49 = 58$$

$$\|x_1 - v_2\|^2 = 8^2 + 2^2 = 64 + 4 = 68$$

$$\mu_{c1}(x_1) = \frac{1}{\frac{58}{58} + \frac{58}{68}} = \frac{1}{1 + 0.853} = 0.5397$$

Similarly, we obtain the following:

$$\mu_{c2}(x_1) = \frac{1}{\frac{68}{58} + \frac{68}{68}} = 0.4603$$

$$\mu_{c1}(x_2) = \frac{1}{\frac{17}{17} + \frac{17}{37}} = 0.6852$$

$$\mu_{c2}(x_2) = \frac{1}{\frac{37}{17} + \frac{37}{37}} = 0.3148$$

$$\mu_{c1}(x_3) = \frac{1}{\frac{68}{68} + \frac{68}{18}} = 0.2093$$

$$\mu_{c2}(x_3) = \frac{1}{\frac{18}{68} + \frac{18}{18}} = 0.7907$$

$$\mu_{c1}(x_4) = \frac{1}{\frac{36}{36} + \frac{36}{26}} = 0.4194$$

$$\mu_{c2}(x_4) = \frac{1}{\frac{26}{36} + \frac{26}{26}} = 0.5806$$

$$\mu_{c1}(x_5) = \frac{1}{\frac{53}{53} + \frac{53}{13}} = 0.197$$

$$\mu_{c2}(x_5) = \frac{1}{\frac{13}{53} + \frac{13}{13}} = 0.803$$

$$\mu_{c1}(x_6) = \frac{1}{\frac{82}{82} + \frac{82}{52}} = 0.3881$$

$$\mu_{c2}(x_6) = \frac{1}{\frac{52}{82} + \frac{52}{52}} = 0.6119$$

# Example

Now by using equation $v_1 = \frac{\sum_{k=1}^{6} (\mu_{c1}(x_k))^2 \times x_k}{\sum_{k=1}^{6} (\mu_{c1}(x_k))^2}$ the new co-ordinate values for the centre $v_1$ can be calculated as:

$$\frac{0.5397^2 \times (2,12) + 0.6852^2 \times (4,9) + 0.2093^2 \times (7,13) + 0.4194^2 \times (11,5) + 0.197^2 \times (12,7) + 0.3881^2 \times (14.}{0.5397^2 + 0.6852^2 + 0.2093^2 + 0.4194^2 + 0.197^2 + 0.3881^2}$$

$$= \frac{732761}{1.0979}, \frac{10.044}{1.0979} = (6.6273,\ 9.1484)$$

Similarly, by using the equation $v_2 = \frac{\sum_{k=1}^{6} (\mu_{c2}(x_k))^2 \times x_k}{\sum_{k=1}^{6} (\mu_{c2}(x_k))^2}$ the new co-ordinate values for the center $v_2$ can be calculated as:

$$\frac{0.4603^2 \times (2,12) + 0.3148^2 \times (4,9) + 0.7909^2 \times (7,13) + 0.5806^2 \times (11,5) + 0.803^2 \times (12,7) + 0.6119^2 \times (14,4)}{0.4603^2 + 0.3148^2 + 0.7909^2 + 0.5806^2 + 0.803^2 + 0.6119^2}$$

$$= \frac{22.326}{2.2928}, \frac{19.4629}{2.2928} = (9.7374,\ 8.4887)$$

Master of Technology

## Computational Intelligence II

# Other Fuzzy Technologies

Dr. TIAN Jing
Institute of Systems Science,
National University of Singapore
Email: tianjing@nus.edu.sg

# Fuzzy logic and machine learning

- “*the connection between the fuzzy and the core machine learning community is not well established at all. On the contrary, the two communities seem to be sharply separated, with very little (if any) interaction in the form of joint meetings, research initiatives or mutual conference attendance. For example, contributions on fuzzy machine learning are almost exclusively published in fuzzy journals and conferences, whereas it is extremely difficult to find a fuzzy paper in a core machine learning conference or journal*.”

Does machine learning need fuzzy logic?

Eyke Hüllermeier

*Department of Computer Science, University of Paderborn, Germany*

Reference: E. Hullermeier, Does machine learning need fuzzy logic? ***Fuzzy Sets and Systems***, Vol. 281, 2015, pp. 292-299.

# Selected other fuzzy technologies

[Note: These are **NOT** covered in assessment]

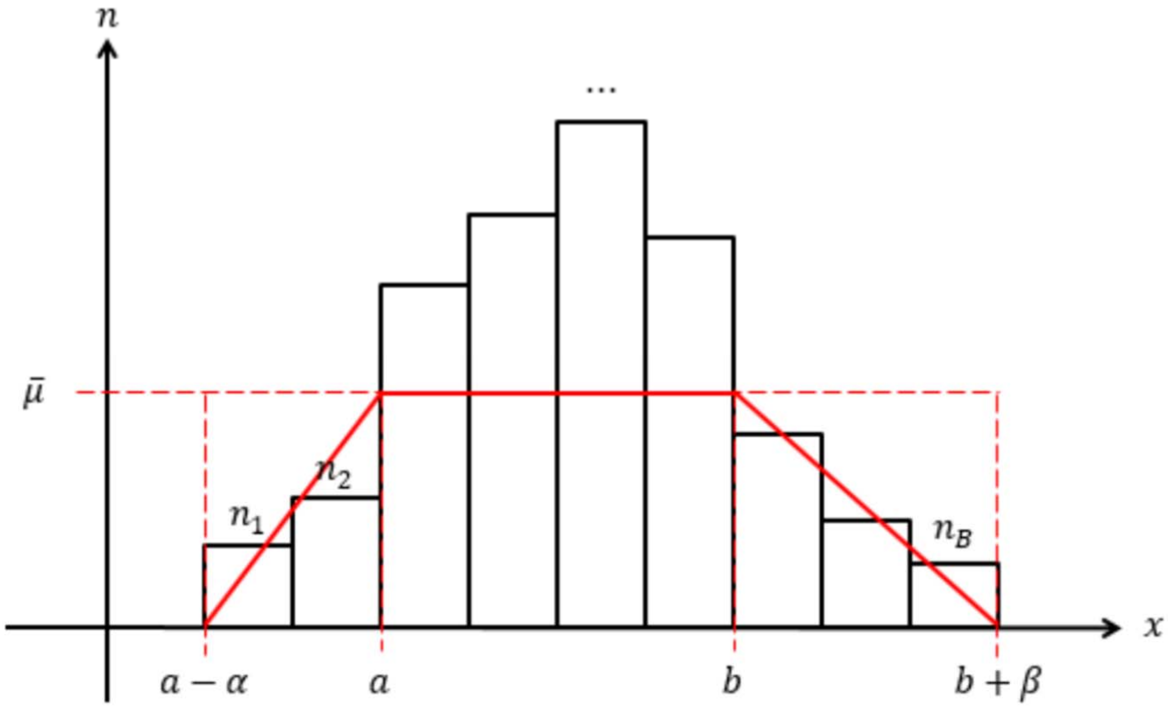
1. Data driven membership function
2. Fuzzy feature
3. Fuzzy histogram
4. Fuzzy logic control to parameter fine-tuning in neural network training
5. Fuzzy measure for classifier fusion

# 1. Data driven membership function

- Objective: How to avoid tedious job in manually determine the membership function.
- Case: Automatically build data-driven membership function for scene understanding.
- Reference
  - » C. H. Lim, A. Risnumawan, C. S. Chan, "Scene image is non-mutually exclusive - A fuzzy qualitative scene understanding," *IEEE Trans. On Fuzzy Systems*, Vol. 22, No. 6, Dec. 2014, pp. 1541-1556.
  - » Fuzzy computer vision toolbox, https://github.com/cs-chan/FuzzyComputerVision

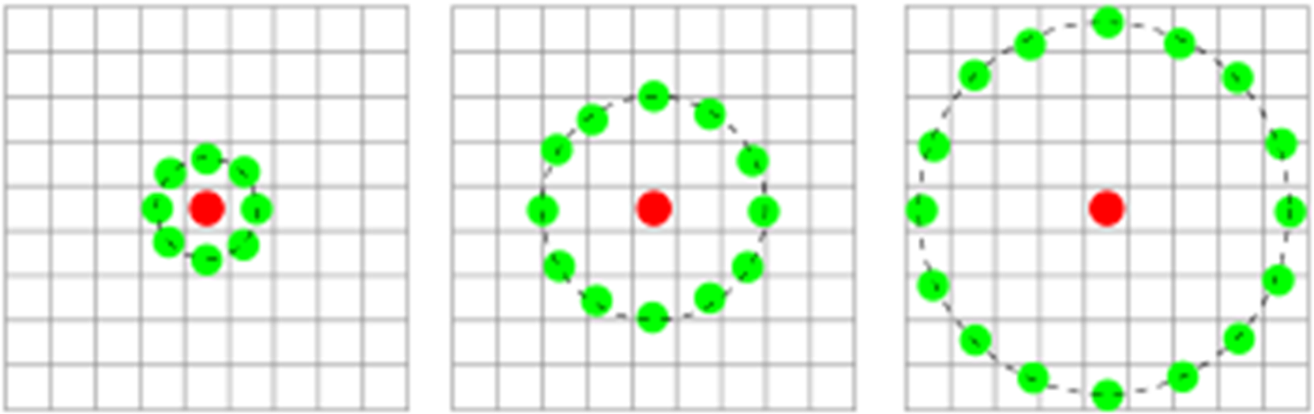
# 1. Data driven membership function

- The parametric representation of a histogram, $x$ is the feature value, $n$ denotes the occurrence of training data from its respective bin $n_1, n_2, \cdots, n_B$. $a$ and $b$ represent the lower and upper bound of $\mu = \sum_{i=1}^{B} n_i / b$, while $a - \alpha$, and $b + \beta$ represent the minimum and maximum of $x$ value. The dominant region is the area of $[a, b]$.

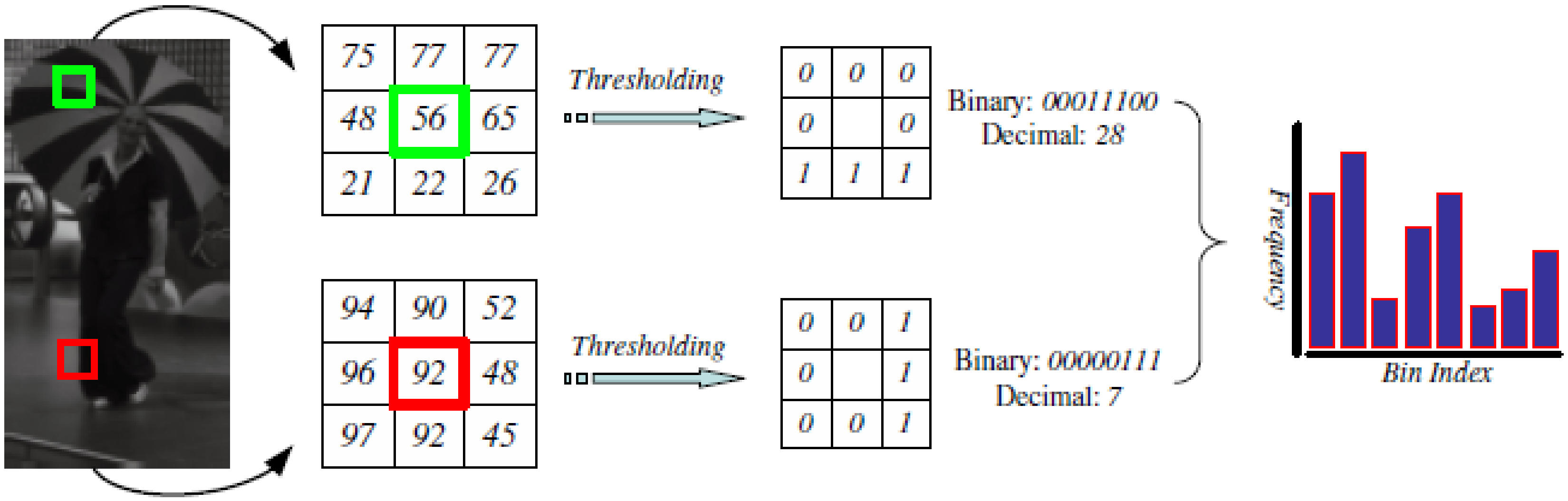


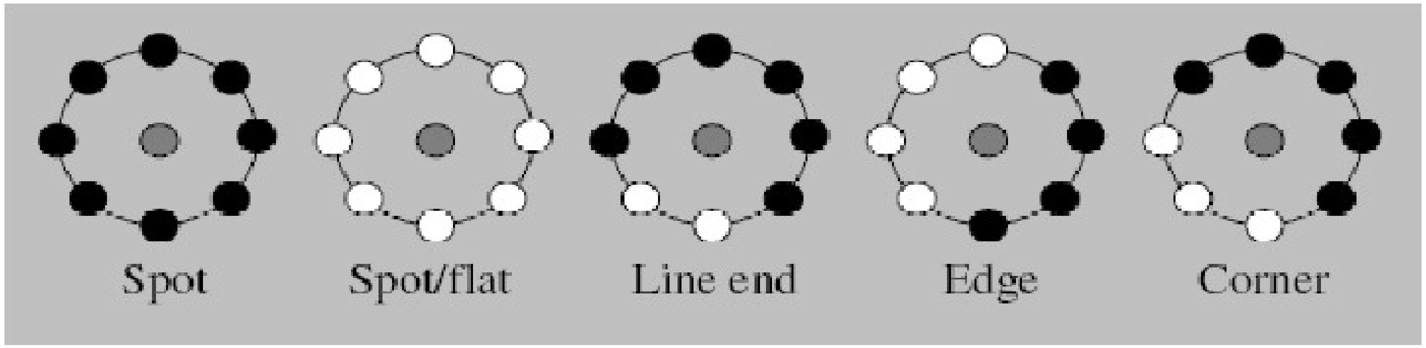
# 2. Fuzzy feature

- Local binary patterns: For each data point in a cell, compare the value to each of its (eight) neighbors (on its left-top, left-middle, left-bottom, right-top, etc.). Follow the data point along a circle, i.e. clockwise or counter-clockwise. Where the center data point's value is greater than the neighbor, write "1". Otherwise, write "0". This gives an 8-digit binary number.
- Compute the histogram, over the cell, of the frequency of each "number" occurring (i.e., each combination of which data point are smaller and which are greater than the center).



# 2. Fuzzy feature



# 2. Fuzzy feature

- In the LBP representation, a pattern is represented by a set of elements $P = \{p_{center}, p_0, p_1, \cdots p_7\}$, where $p_{center}$ represents the value of the central position, and $p_i (0 \le i \le 7)$ represent the values of the a $3 \times 3$ neighbourhood. They can be characterized by a set of binary values $d_i (0 \le i \le 7)$ where

$$d_i = \begin{cases} 1 & \text{if } p_i \ge p_{center} \\ 0 & \text{if } p_i < p_{center} \end{cases}$$

- Then, a LBP code is calculated as

$$LBP = \sum_{i=0}^{7} d_i \cdot 2^i$$

# 2. Fuzzy feature

- Rule $R_0$: The smaller $p_i$ is, with respect to $p_{center}$, the greater the certainty that $d_i$ is 0.
- Rule $R_1$: The bigger $p_i$ is, with respect to $p_{center}$, the greater the certainty that $d_i$ is 1.
- We define two membership functions with a threshold $T$ that controls the degree of fuzziness as

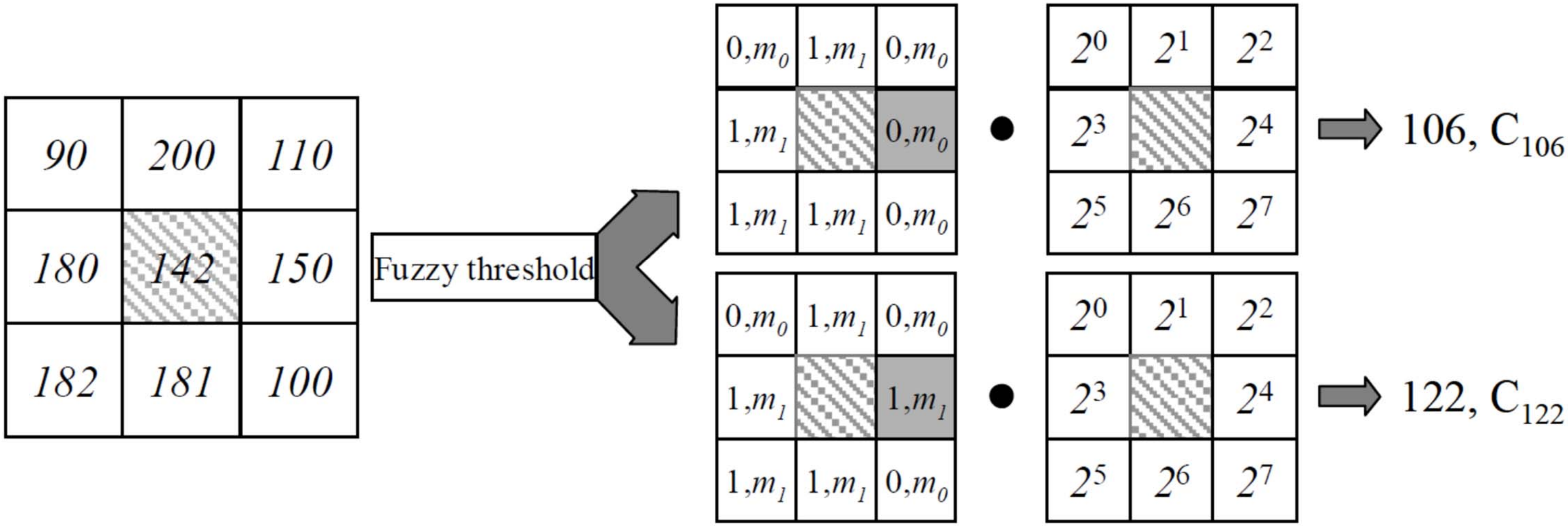
$$m_0(i) = \begin{cases} 0 & \text{if } p_i \geq p_{center} + T \\ \dfrac{T - p_i + p_{center}}{2T} & \text{if } p_{center} - T < p_i < p_{center} + T \\ 1 & \text{if } p_i \leq p_{center} - T \end{cases}$$

$$m_1(i) = 1 - m_0(i)$$

Reference: D. K. Iakovidis, E. G> Keramidas, and D. Maroulis, "Fuzzy local binary patterns for ultrasound texture characterization image analysis and fecognition," Campilho A., Kamel M. (eds) Image Analysis and Recognition. ICIAR 2008, pp. 750-759.

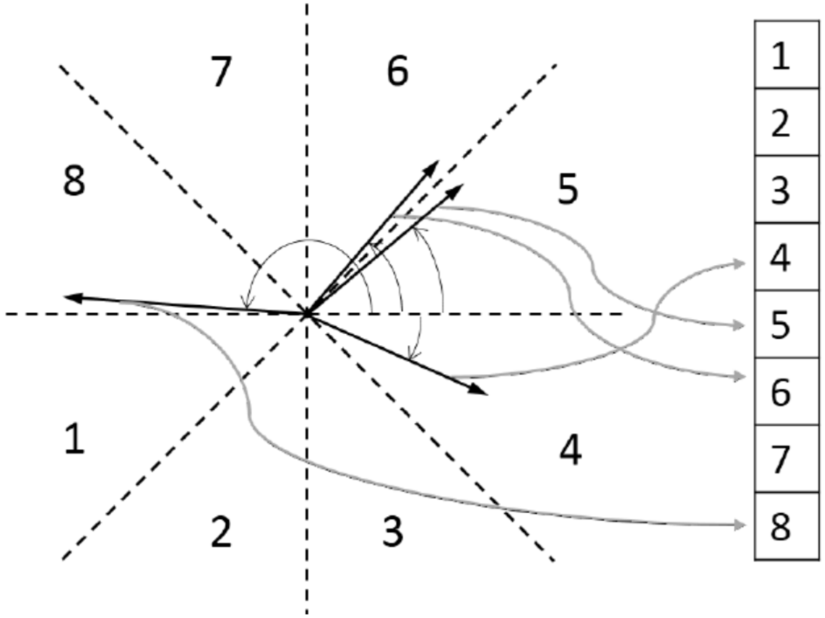
# 2. Fuzzy feature

- For a set of elements $P = \{p_{center}, p_0, p_1, \cdots p_7\}$, we can obtain both the original LBP code and its fuzzy value $C = \prod_{i=0}^{7} m_{d_i}(i)$, where $d_i \in \{0,1\}$.



# 3. Fuzzy histogram

- Histograms of motion: The optical flow vector $v = \left[v_x, v_y\right]^T$ is computed at every pixel location of the image. The magnitude $\rho = \sqrt{v_x^2 + v_y^2}$ and orientation $-\pi \le \theta < \pi$ of the motion is derived from the optical flow vectors. Thus, for each pixel, we will obtain a motion magnitude and orientation pair $(\rho_i, \theta_i)$. Then, all the flow vectors are binned according to their angle and weighted according to their magnitude.



# 3. Fuzzy histogram

- For an image $I$ of size $N \times M$, the Histogram of motion features are represented as $H(I) = [h_1, h_2, \cdots h_n]$, where $n$ is the number of histogram bins $h_i = \frac{h_i'}{\sum_{k=1}^{n} h_k'}$, and $h_i' = \sum_{j=1}^{M\times N} \rho_j \, \mu_{ij}$

$$\mu_{ij} = \begin{cases} 1 & if \; -\pi + \frac{2\pi}{n}(i-1) \le \theta_j < -\pi + \frac{2\pi}{n} i \\ 0 & \text{otherwise} \end{cases}$$

Hard assignment

$$\mu_{ij} = exp\left(\frac{-(\theta_j - c_i)^2}{2\sigma^2}\right)$$

Fuzzy assignment

Where $c_i$ is the $i$th bin center, and $\sigma$ is the variance of the Gaussian membership function.

Reference: S. L. Happy and A. Routray, "Fuzzy histogram of optical flow orientations for micro-expression recognition," *IEEE Trans. on Affective Computing*, accepted, https://ieeexplore.ieee.org/document/7971947/

# 4. Parameter fine-tuning

- Objective: How to apply fuzzy technology to make decision in parameter fine-tuning for neural network.
- Case: Fuzzy logic is employed to control the learning parameters where the objective is to reduce the possibility of overshooting during the learning process, increase the convergence speed and minimize the error.
- Reference
  - » C. Hatri and J. Boumhidi, "Fuzzy deep learning based urban traffic incident detection," *Cognitive Systems Research*, Vol. 50, 2018, pp. 206-213.

# 4. Parameter fine-tuning

- There are four parameters used to create the rules for the fuzzy logic control system; the *relative error* (RE), *change in relative error* (CRE), *sign change in error* (SC) and *cumulative sum of sign change* in error (CSC).

$$\begin{cases} RE(t) = E(t) - E(t-1) \\ CRE = RE(t) - RE(t-1) \\ SC(t) = 1 - \left\| \frac{1}{2} [sign(RE(t-1) + sign(RE(t)) \right\| \\ CSC = \sum_{m=t-4}^{t} SC(m) \end{cases}$$

# 4. Parameter fine-tuning

- The fuzzy logic system contains two inputs RE and CRE, and one output; the change in the learning parameter $\Delta\beta$. The range of RE and CRE is 0-1. The linguistic values of RE, CRE are NL (‘Negative Large), NS (Negative Small), ZE (Zero), PS (Positive Small) and PL (Positive Large).

Fuzzy rule table

| CRE | RE | | | | |
|---|---|---|---|---|---|
| | NL | NS | ZE | PS | PL |
| NL | NS | NS | NS | NS | NS |
| NS | NS | ZE | PS | ZE | NS |
| ZE | ZE | PS | ZE | NS | ZE |
| PS | NS | ZE | PS | ZE | NS |
| PL | NS | NS | NS | NS | NS |

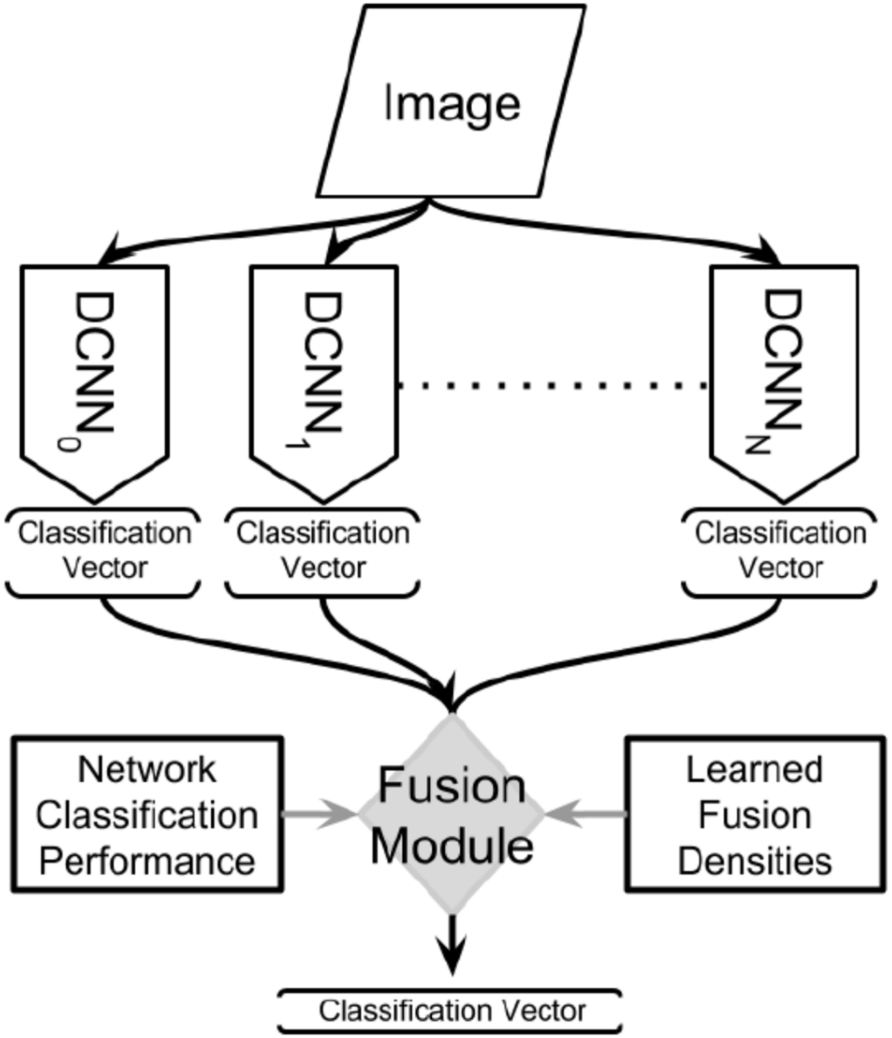
# 5. Classifier fusion

- Objective: How to apply fuzzy technology to fuse results of various classification models.
- Case: Land cover image classification
- Reference
  - » G. J. Scott, R. A. Marcum, C. H. Davis, and T. W. Nivin, “Fusion of deep convolutional neural networks for land cover classification of high-resolution imagery,” *IEEE Geoscience and Remote Sensing Letters*, Vol. 14, No. 9, Sept. 2017, pp. 1638-1642.
  - » G. J. Scott, K. C. Hagan, R. A. Marcum, J. A. Hurt, and D. T. Anderson, "Enhanced fusion of deep neural networks for classification of benchmark high-resolution image data sets,” *IEEE Geoscience and Remote Sensing Letters, accepted*, https://ieeexplore.ieee.org/document/8385173/

# 5. Classifier fusion

- DCNN: deep convolutional neural networks. CaffeNet, GoogleNet, ResNet50 are used in this paper.
- Classification vector: 21-class classification probability score vector.

| | |
|---|---|
| $M$ | Number of classes, classifier output size. |
| $N$ | Number of classifiers. |
| $C$ | Classification matrix, $M \times N$. |
| $\vec{c}_j$ | Classification output from network, $j \in \{1 \ldots N\}$. |
| $P$ | Cross-validation performance matrix, $M \times N$. |



# 5. Classifier fusion

- Simple Voting

This takes the highest confidence class for each $\vec{c}_j$, and then selects the class by the most occurring class $i$. In other words, we compute a vector $V$ of a selected class per classifier by

$$V(C) = (\underset{i=1,\cdots M}{\operatorname{argmax}}(\vec{c}_1), \cdots, \underset{i=1,\cdots M}{\operatorname{argmax}}(\vec{c}_N))$$

Then, the classification is simply $mode(V(C))$.

# 5. Classifier fusion

- Accuracy weighted sum

This takes the classification matrix $C$ and computes a weighted confidence across all classifiers for each class using the classification accuracy $\vec{p}_j$ as the weights in the sum. The vector $\vec{p}_j$ is computed as the per class, average cross-validation performance for each network, $j \in N$. Let $C_j$ denote the $j$th row of the classification matrix, then

$$V(C) = (\vec{p}_1 \cdot C_1^T, \cdots, \vec{p}_M \cdot C_M^T)$$

The predicted class is predicted as $\underset{i=1,\cdots M}{\operatorname{argmax}}(V(C))$.

# 5. Classifier fusion

- Key idea: Evaluate both the real possibilities relating to objective evidence (*classifier confidence score*) and the expectation which defines the level of importance of a subset of sources (*classifier capability*).

$$\mathrm{F(C,P)} = \underset{i=1,\cdots,M}{\operatorname{argmax}}\left(\mathrm{CHFI}_{i=1}^{M}(\vec{c}_j(i), \vec{p}_j(i))\right)$$

Specifically, $F(C, P)$ calculates $M$ independent aggregations across the $N$ DCNN, and the class with the largest aggregated value is selected. Herein, we use the Choquet fuzzy integral (ChFI). The ChFI has $2^N$ parameters called the *fuzzy measure* (FM), one for each subset of inputs (DCNN). We let the *densities*, $\mu^j = \mu(\{c_j\})$, or worth of the individual sources be the accuracies of the DCNN from the training data, e.g., Fig. 1, which is given by $\vec{p}$. Next, the Sugeno $\lambda$-FM is used to take these $N$ densities and derive the remaining parameters according to Sugeno's characteristic polynomial function

$$1 + \lambda = \prod_{j=1}^{N}(1 + \lambda\mu^j) \tag{5}$$

which has a single solution ($\lambda > -1$). From here, each parameter can be calculated as $\mu(A \cup B) = \mu(A) + \mu(B) + \lambda\mu(A)\mu(B)$ for sets $A, B \in 2^X$. The ChFI for class $i$ is

$$\mathrm{ChFI}(h, \mu) = \sum_{j=1}^{N} h_{(j)}[\mu(A_j) - \mu(A_{j-1})] \tag{6}$$

where $h_j = c_j(i)$, $(j)$ denotes a sorting on $h$, such that $h_{(1)} \geq h_{(2)} \geq \ldots, A_j = \{x_{(1)}, \ldots, x_{(j)}\}$, and $A_0 = 0$.

The ChFI is a generalized expectation operator, such that the output values lie between the minimum and maximum values of $c_1(i), \ldots, c_N(i)$. The fusion happens at a per-class level across the set of classifiers, fusing independently with respect to the output classes. Therefore, there are $M$ fusion operations, each with $N$ classification confidence inputs. The

# Conclusions

- Fuzzy decision making
- Fuzzy clustering
- Fuzzy technology for machine learning

# Thank you!

**Dr TIAN Jing**
**Email: tianjing@nus.edu.sg**