FUZZY IMAGE SEGMENTATION CONSIDERING SURFACE CHARACTERISTICS AND FEATURE SET SELECTION STRATEGY

M. Ameer Ali
Gippsland School of Computing and IT, Monash University, Churchill, VIC-3842, Australia
Tel: +61-3-9902-6568, Fax: +61-3-9902-6879, Email: Ameer.Ali@infotech.monash.edu.au

Gour C Karmakar, *Member*, IEEE
Gippsland School of Computing and IT, Monash University, Churchill, VIC-3842, Australia
Tel: +61-3-9902-6252, Fax: +61-3-9902-6879, Email: Gour.Karmakar@infotech.monash.edu.au

Laurence S Dooley, *Senior Member*, IEEE
Gippsland School of Computing and IT, Monash University, Churchill, VIC-3842, Australia
Tel: +61-3-9902-6628, Fax: +61-3-9902-6842, Email: Laurence.Dooley@infotech.monash.edu.au

Abstract

The image segmentation performance of any clustering algorithm is sensitive to the features used and the types of object in an image, both of which compromise the overall generality of the algorithm. This paper proposes a novel *fuzzy image segmentation considering surface characteristics and feature set selection strategy* (FISFS) algorithm which addresses these issues. Features that are exploited when the initially segmented results from a clustering algorithm are subsequently merged include connectedness, object surface characteristics and the arbitrariness of the fuzzy c-means (FCM) algorithm for pixel location. A perceptual threshold is also integrated within the region merging strategy. Qualitative and quantitative results are presented, together with a full time-complexity analysis, to confirm the superior performance of FISFS compared with FCM, possibilistic c-means (PCM), and suppressed FCM (SFCM) clustering algorithms, for a wide range of disparate images.

Keywords – Fuzzy image segmentation, Surface characteristics, Connectedness, Intensity, Location.

1. Introduction

Object-based image segmentation is an intractable task because of the myriad number of objects in an image, and the enormous variations between them, which means segmenting every object using one particular feature set is not feasible. Most images possess some ambiguous information and hence any segmentation produces fuzzy regions. In such circumstances, fuzzy image segmentation performs better than traditional "crisp" techniques. Fuzzy image segmentation algorithms are broadly classified into six classes: (i) fuzzy geometry-based, (ii) fuzzy thresholding-based, (iii) fuzzy integral-based, (iv) fuzzy rule-based, (v) soft computing-based, and (vi) fuzzy clustering, which is the most popular and widely used [1]. In general, object-based segmentation [2]-[14] uses different feature types, such as brightness (gray scale pixel intensity) and geometric information (pixel location) to measure the similarity. The effectiveness of these algorithms however is highly dependent on the types of features used and *a priori* information concerning the objects in an image. This domain specific information about which feature type produces the best results for which image, has a considerable impact on an algorithm's generalisation capability [3]. For instance, clustering algorithms such as FCM [3], PCM [4] and SFCM [15] cannot separate image regions that have similar pixel intensities by considering only their pixel intensity. They may however, be able to segment by exploiting information on either the location of pixels or using a combination of pixel intensity and location (CIL)[1]. Likewise, clustering cannot always segment adjacent and enclosed regions with different pixel intensities by only considering PL, but may well do so by using PI. It has been found [16] that in many cases, clustering algorithms that combined both image features (PI and PL) do not provide the expected results for objects having similar PI and similar surface variations (SSV). This provided the foundation for the new segmentation strategy proposed in this paper, of merging the individual initially segmented results generated by any clustering algorithm and judiciously selecting the most appropriate feature sets.

[1] Throughout this paper *PL*, *PI* and *CIL* refer respectively to *pixel location*, *pixel intensity* and a *combination* of pixel intensity and normalized pixel location.

The new *fuzzy image segmentation considering surface characteristics and feature set selection strategy* (FISFS) algorithm incorporates a number of features; (i) the topological property of connectedness; (ii) identification of objects having SSV and dissimilar surface variations (DSV); (iii) human visual perception; (iv) arbitrariness of FCM for PL; and (v) automatic selection of feature sets for initial segmentation. Embedded within FISFS are two constituent algorithms, which handle the *merging of individually segmented results* (MISR) and the *separation of objects having similar and dissimilar surface variations* (SOSDS). A full time-complexity analysis for the FISFS algorithm is presented and results analytically compared with FCM, PCM and SFCM for all three cases, PI, PL and CIL. The numerical performance is assessed using the objective segmentation evaluation method, *discrepancy based on the number of misclassified pixels* [1].
The paper is structured as follows: In Section 2, a proof of the arbitrariness of FCM for PL is presented, while the identification of objects having SSV is detailed in Section 3. Section 4 discusses the region merging strategy used in the MISR algorithm, with the new FISFS and SOSDS algorithms described in Section 5. A comprehensive time-complexity analysis of the FISFS algorithm is presented in Section 6, while its segmentation performance is fully analysed and compared in Section 7. Section 8 provides some concluding remarks.

2. Arbitrariness of FCM For Pixel Location

FCM using PL arbitrarily divides objects irrespective of their size and surface variations, as proven by the following lemma.

Lemma 1: The *decision boundary* between two segmented clusters generated by FCM using PL is orthogonal to a line connecting the two respective cluster centres and passes through its midpoint.

Proof: In
Figure 1, if C_1 and C_2 are two cluster centres produced by FCM using PL and C is the midpoint, then it needs to be proved that the line $AE \perp C_1C_2$ passing through C is a decision boundary.
Hypothesis: AE is the decision boundary between C_1 and C_2 for n data points where n_1 and n_2 data points are already classified into clusters C_1 and C_2 respectively.
Basis Step: Assume an arbitrary data point G, so A is the intersection point of extended C_1G and AE. Since $C_1C=CC_2$ and $AC \perp C_1C_2$, $\Delta AC_1C \cong \Delta AC_2C$, then $AC_1=AC_2$ and $\angle AC_1C= \angle AC_2C$. From
Figure 1, $\angle AC_2C > \angle C_1C_2G$ i.e. $\angle AC_1C_2= \angle GC_1C_2 > \angle C_1C_2G$, which implies that $C_2G>C_1G$, so G is closer to C_1 than C_2. As FCM minimizes intra-cluster distance and maximizes inter-cluster distance, G will be classified into C_1.

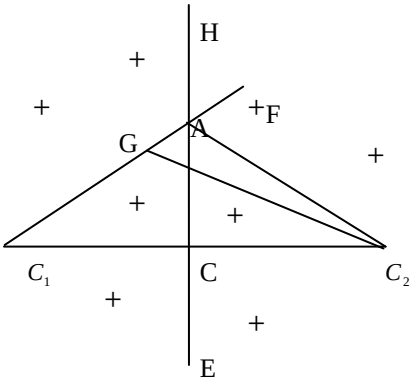


Figure 1: Decision boundary for two clusters.

If G is on the opposite side of AE, a similar argument confirms it will be classified into C_2, so in conclusion, in order to both minimize intra- and maximize inter-cluster distances, any data point on the left of AE belongs to cluster C_1 and vice versa, i.e., AE creates a data dichotomy and is a decision boundary.

Induction Step: Assume the hypothesis that AE is the decision boundary between C_1 and C_2 for n data points is true. Applying the same reasoning as in the *Basis Step*, AE will classify new data into the appropriate clusters, thus implying it is a decision boundary for $(n+1)$ data points and by induction also for an arbitrary number of data points. ■
While the above proof is only for $\Re$ =2 clusters, the theory is generic and extendible to an arbitrary number of clusters. There will then be decision boundaries between each pair of clusters as shown in Figure 2 for the case of $\Re$ =3, with C_1, C_2, and C_3 being the cluster centres, and L_1, L_2 and L_3 the corresponding decision boundaries. For classification, a datum and its corresponding cluster centre must be the same side of each decision boundary (see Figure 2).

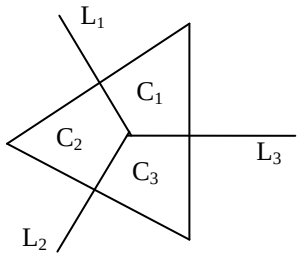


Figure 2 Decision boundaries for multiple clusters.

The maximum possible number of boundaries for $\Re$ clusters is ${}^{\Re}C_2 = \Re(\Re-1)/2 = O(\Re^2)$, so the decision boundary increases by $O(\Re^2)$ which means the arbitrariness of FCM using PL commensurately increases.

3. Identification of Similar Surfaces

The flowchart in Figure 3 outlines the basic steps of the FISFS algorithm. The first and most challenging step is to determine between objects in an image having similar and dissimilar surfaces. Before examining how this can be achieved, it is important to note that there are two possible scenarios whereby surfaces are considered to be similar with respect to brightness, namely: (i) when they have similar PI (see example in Figure 4(a)) and (ii) possess SSV (Figure 4(b)). For clarity, the background of both images in Figure 4 has been removed leaving only the foreground objects. The two objects in Figure 4(a) have also been artificially modified so they exhibit very similar PI. It has been proven previously [17] that in most cases where FCM using CIL was unsuccessful in segmenting objects, FCM using PI also failed. The reason for this is that FCM

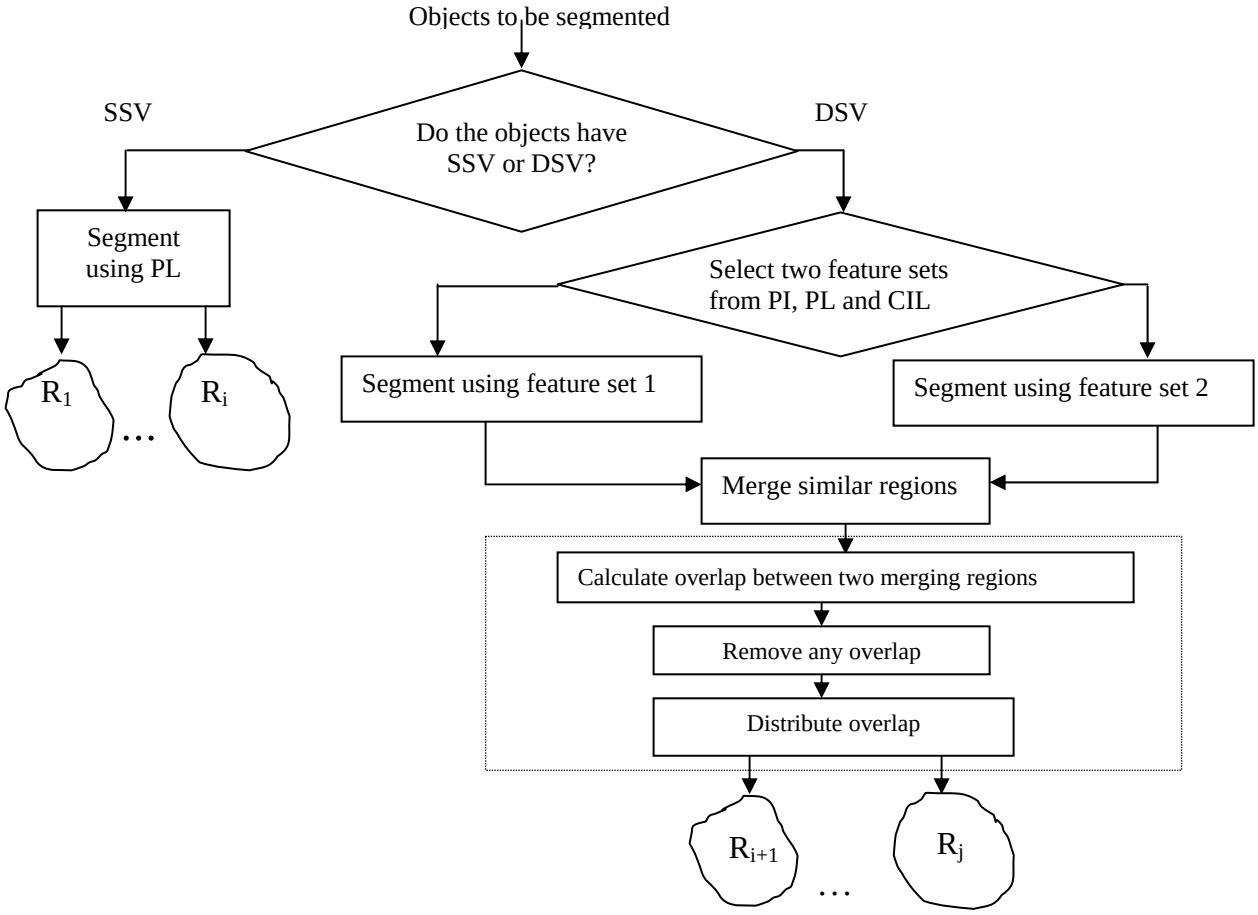
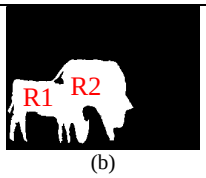


Figure 3 : Flowchart of the FISFS algorithm.

using CIL makes a decision by considering only PI, with the implication that only PL should be used to provide improved segmentation results. The potential of FCM using CIL as a meaningful feature is now explored by separately considering the above two scenarios. In (i), both objects in Figure 4(a) have similar PI, so FCM using CIL generates cluster centre coordinates for intensity which are very close. This means PL will dominate PI and the segmented results produced by CIL are analogous to those of PL (Lemma 2). In case (ii), the objects in Figure 4(b) have SSV so FCM using CIL generates cluster centres for location that are very close, because of the distribution of bright and dark pixels over most of the surface of



(b)

(c)

Figure 4: (a) Cow image; (c) Tiger image.

both objects. PI will therefore dominate PL and yield segmentation results for CIL and PI (Lemma 3) which are very similar.

Lemma 2: If the cluster centre coordinates for PI produced by FCM using CIL are very close to each other, the segmented results will be very similar to that produced by FCM using only PL.

Proof: The basis of a fuzzy clustering algorithm such as FCM is to minimize the intra and maximize the inter-cluster distances by iteratively minimizing an objective function.
Let C_1 and C_2 be two cluster centres, X_j a datum and d_1 and d_2 the distances of X_j from C_1 and C_2 respectively, which for FCM using CIL are defined as:

$$d_1^2 = \left(C_1^{L_X} - X_j^{L_X}\right)^2 + \left(C_1^{L_Y} - X_j^{L_Y}\right)^2 + \left(C_1^{P} - X_j^{P}\right)^2 \quad (1)$$

$$d_2^2 = \left(C_2^{L_X} - X_j^{L_X}\right)^2 + \left(C_2^{L_Y} - X_j^{L_Y}\right)^2 + \left(C_2^{P} - X_j^{P}\right)^2 \quad (2)$$

where L_X, L_Y and P represent the X-dimension, Y-dimension and PI of a feature respectively. The difference $\left(d_1 - d_2\right)$ is used as a measure of the degree of belonging of X_j to a particular cluster. For example, if $\left(d_1 - d_2\right)$>0, X_j is classified into cluster C_2, otherwise it is classified into C_1. Since the cluster centres for PI are very close to each other i.e. $C_1^P \to C_2^P$, from (1) and (2) $\left(d_1^2 - d_2^2\right)$ becomes:

$$\left(d_1^2 - d_2^2\right) = \underset{C_1^P \to C_2^P}{Lt}\left(d_1^2 - d_2^2\right) =$$

$$\underset{C_1^P \to C_2^P}{Lt}\left\{\left(C_1^{L_X} - X_j^{L_X}\right)^2 + \left(C_1^{L_Y} - X_j^{L_Y}\right)^2 + \left(C_1^{P} - X_j^{P}\right)^2\right\} -$$

$$\underset{C_1^P \to C_2^P}{Lt}\left\{\left(C_2^{L_X} - X_j^{L_X}\right)^2 + \left(C_2^{L_Y} - X_j^{L_Y}\right)^2 + \left(C_2^{P} - X_j^{P}\right)^2\right\}$$

$$= \left(C_1^{L_X} - X_j^{L_X}\right)^2 + \left(C_1^{L_Y} - X_j^{L_Y}\right)^2 + \left(C_2^{P} - X_j^{P}\right)^2$$

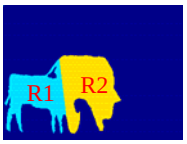
$$- \left(C_2^{L_X} - X_j^{L_X}\right)^2 - \left(C_2^{L_Y} - X_j^{L_Y}\right)^2 - \left(C_2^{P} - X_j^{P}\right)^2$$

$$= \left\{\left(C_1^{L_X} - X_j^{L_X}\right)^2 + \left(C_1^{L_Y} - X_j^{L_Y}\right)^2\right\}$$

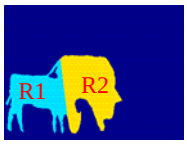
$$- \left\{\left(C_2^{L_X} - X_j^{L_X}\right)^2 + \left(C_2^{L_Y} - X_j^{L_Y}\right)^2\right\} \quad (3)$$

As (3) is independent of PI, X_j is classified using only PL and so objects that have similar PI will be segmented using PL despite CIL being used. ■

In the example in Figure 4(a), both objects have similar PI, and the segmentation results produced by using CIL (Figure 5(a)) are the same as those for FCM using PL (Figure 5(b)).



(a)



(b)

Figure 5: (a) FCM using CIL; (b) FCM using PL.

Lemma 3: If the cluster centre coordinates for PL produced by FCM using CIL are very close to each other, the segmented results are very similar to those produced by FCM using only PI.

Proof: As detailed in Lemma 2, if the cluster centres are very close to each other for PL, i.e., $C_1^L \to C_2^L$, from (1) and (2), then $\left(d_1^2 - d_2^2\right)$ can be expressed as:-

$$\left(d_1^2 - d_2^2\right) \underset{C_1^L \to C_2^L}{Lt} \left(d_1^2 - d_2^2\right) =$$

$$\underset{C_1^L \to C_2^L}{Lt} \left\{\left(C_1^{L_X} - X_j^{L_X}\right)^2 + \left(C_1^{L_Y} - X_j^{L_Y}\right)^2 + \left(C_1^P - X_j^P\right)^2\right\}$$

$$- \underset{C_1^L \to C_2^L}{Lt} \left\{\left(C_2^{L_X} - X_j^{L_X}\right)^2 + \left(C_2^{L_Y} - X_j^{L_Y}\right)^2 + \left(C_2^P - X_j^P\right)^2\right\}$$

$$= \underset{C_1^{L_X} \to C_2^{L_X}}{Lt} \left\{\left(C_1^{L_X} - X_j^{L_X}\right)^2 - \left(C_2^{L_X} - X_j^{L_X}\right)^2 + \left(C_1^P - X_j^P\right)^2\right\}$$

$$- \underset{C_1^{L_Y} \to C_2^{L_Y}}{Lt} \left\{\left(C_1^{L_Y} - X_j^{L_Y}\right)^2 - \left(C_2^{L_Y} - X_j^{L_Y}\right)^2 + \left(C_2^P - X_j^P\right)^2\right\}$$

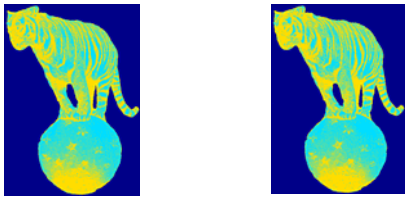
$$= \left\{\left(C_2^{L_X} - X_j^{L_X}\right)^2 - \left(C_2^{L_X} - X_j^{L_X}\right)^2 + \left(C_1^P - X_j^P\right)^2\right\}$$

$$- \left\{\left(C_2^{L_Y} - X_j^{L_Y}\right)^2 - \left(C_2^{L_Y} - X_j^{L_Y}\right)^2 + \left(C_2^P - X_j^P\right)^2\right\}$$

$$= \left(C_1^P - X_j^P\right)^2 - \left(C_2^P - X_j^P\right)^2 \quad (4)$$

As $\left(d_1^2 - d_2^2\right)$ is independent of PL, i.e. both L_X and L_Y, X_j will be classified by considering only PI. Thus for objects with SSV, FCM using CIL produces the same results as using PI. ■

For the Figure 4(b) example, FCM using CIL produces the segmented regions shown in Figure 6(b), which are exactly the same as those for FCM using PI (Figure 6(a)). In this case however, the objects could not be separated at all because of their SSV characteristics.



(a) (b)

Figure 6: (a) FCM using PI (b) FCM using CIL.
Note in both cases the objects cannot be separated.

Lemma 3 confirms that when objects have SSV with a repetitious pattern of bright and dark pixels, FCM using CIL is unable to segment them. This provides the basis to apply FCM using CIL to determine the type of surface variations of an object. Whenever it fails to separate a group of objects, they all have SSV and the only way to separate them is by using FCM with PL.

4. The MISR Algorithm

As alluded to in Section 1, since many objects contain ambiguous information, no single feature or even group of features is always appropriate for segmenting each object in an image. This was the rationale behind independently merging the segmented results produced by FCM using PL, PI and CIL in [16, 17]. The various steps involved in the *merging of individually segmented results* (MISR) algorithm are detailed in Algorithm 1. R^I, R^L and R^C represent the individual segmented regions produced by FCM using PI, PL, and CIL respectively. To merge similar regions, their similarity is determined by summing the absolute differences of pixel intensity on a bitwise basis (Step 1)—so the smaller the difference, the greater the similarity between regions. Region R_j^I is considered similar to R_k^L if:-

$$Similar\left(R_j^I\right) = \min_{1 \le k \le \Re} \left\{\sum \left|P_j^I(x,y) - P_k^L(x,y)\right|\right\} \quad (5)$$

where $P(x,y)$ is a pixel at location (x,y) and $P_j^I(x,y) \in R_j^I \wedge P_k^L(x,y) \in R_k^L$

Similar regions are merged (Step 2) by computing the union of the relevant regions. The merging of two similar regions R_j^I to R_k^L is defined as:-

$$R_j = \left\{P(x,y) \middle| P(x,y) \in R_j^I \vee P(x,y) \in R_k^L\right\} \quad (6)$$

Since the merged region is formed by combining two similar regions produced by FCM with different features, the result may contain some overlapping pixels which are treated as misclassified. The overlapping pixels between two merged regions R_i and R_j are expressed as:-

$$R_{ij}^{O} = \left\{ P(x,y) \middle| P(x,y) \in R_i \wedge P(x,y) \in R_j \right\} \quad (7)$$

where $i \neq j \; AND \; 1 \leq i, j \leq \Re$.

To derive the final segmented result, the overlapping pixels need to be redistributed between the merged regions. This requires all misclassified pixels to be removed (Step 3) from the corresponding merged regions using the following equations:-

$$R_i^{'} = \left\{ P(x,y) \middle| P(x,y) \in R_i \wedge P(x,y) \notin R_{ij}^{O} \right\} \quad (8)$$

$$R_j^{'} = \left\{ P(x,y) \middle| P(x,y) \in R_j \wedge P(x,y) \notin R_{ij}^{O} \right\} \quad (9)$$

There are certain scenarios where it is propitious to apply the connectivity topological feature within the MISR algorithm and these are examined in Section 5.3. If this feature is used, all misclassified pixels are distributed to the corresponding merged pair using 8-connectivity (Step 4), to ensure that all weak object connections are considered. If there are still any remaining non-connected pixels, these are redistributed by FCM using CIL (Step 5) in order to consider both PI and PL. To choose a suitable pair of initially segmented regions R^I, R^L and R^C for MISR requires the selection of an appropriate feature set. Both the selection strategy and the criterion for applying the connectivity feature are discussed fully in the next section.

Algorithm 1: Merging of individually segmented results (MISR).

Precondition: A selected pair of initially segmented regions R^I, R^L and R^C; $\Re$, Connectivity.
Post-condition: The segmented regions R.

1. Determine similar regions using (5).
2. Merge these similar regions using (6).
3. Calculate the overlap between the two merging regions using (7) and remove overlapping pixels using (8) and (9).
4. IF (Connectivity=TRUE) THEN distribute 8-connected objects of the overlap to merging regions R_i and R_j using 8-connectivity.
5. Redistribute any remaining overlapping pixels by a clustering algorithm using CIL.

5. The FISFS Algorithm

As the FISFS flowchart in Figure 3 confirms, a key issue is being able to identify and separate objects in an image having SSV and DSV. Once this is achieved, SSV objects are segmented by FCM using PL, while DSV object segmentation requires selecting the most appropriate pair of feature sets from PI, PL or CIL. Similar regions of the initially segmented regions are then merged and the overlap for each merging region pair is calculated and removed. The overlapping pixels are the distributed to between this pair of merging regions. Before detailing the FISFS algorithm, the following sections examine firstly the issue of separating objects having SSV and DSV and secondly, the best feature set selection.

5.1 The SOSDS Algorithm

Any strategy that is able to identify and distinguish objects having SSV and DSV undoubtedly affords the potential of improved image segmentation. A description of one such strategy is given in Algorithm 2 called the *separation of objects having similar and dissimilar surface variations* (SOSDS) algorithm. As discussed in Section 3, R^C is used as the initial segmented regions to determine whether an object has either SSV or DSV. To locate SSV regions, the area $A^{R_i^C}$ of the segmented region R_i^C is calculated using a convex hull:-

$$A^{R_i^C} = Area\left(Convexhull\left(R_i^C\right)\right) \quad (10)$$

where $Area(\bullet)$ and $Convexhull(\bullet)$ are respectively the area and vertices of the convex hull of a segmented region. To identify objects having SSV, similar regions are merged using (6) in (Step 2) to form R_k^M and then calculate the area $A^{R_k^M}$, $A^{R_i^C}$, $A^{R_j^C}$ of region R_k^M, R_i^C and R_j^C respectively using (10) (Step 3), where M is the number of merged regions in R_k^M, $2 \leq M \leq \Re$, $1 \leq k \leq \left\lfloor \Re/2 \right\rfloor$ and $1 \leq i, j \leq \Re$. The difference between the area of the largest merged region of the k^{th} merging region and $A^{R_k^M}$ (the area of the k^{th} merging region) is a measure of shape distortion (Step 5), because the merging region R_k^M always

contains the largest merged region. If this distortion measure is less than 0.5*dB*, the human eye will not perceive a change in shape and the segmentation algorithm is unable to separate the objects, i.e. the region has SSV. A perceptual threshold $T_{\max}$ which incorporates both shape distortion and human perception is now introduced into the SSV identification process. The maximum value of $T_{\max}$ is calculated as follows:-

$$0.5 = 20\log\frac{\text{Area of Merging Region}}{\text{Area of the Largest Merged Region}}$$

$$\Rightarrow \text{Area of Merging Region} - \text{Area of the Largest Merged Region} = 0.059 \times \text{Area of the Largest Merged Region}$$

$$\Rightarrow \frac{\text{Area of Merging Region} - \text{Area of the Largest Merged Region}}{\text{Area of the Largest Merged Region}} = 0.059$$

so the threshold is bounded $T_{\max} = 0.059$ (11)

Finally, objects having SSV are separated from those with DSV which are represented by region R^D, where D, $0 \le D \le \Re$ is the number of DSV objects. Those clusters not merged are treated as clusters containing objects having DSV.

Algorithm 2: Separation of objects having similar and dissimilar surface variations (SOSDS)

Precondition: Initially segmented regions R^C and $\Re$.
Post condition: Objects having SSV (R_i^M) and DSV (R^D).

1. Set $M = 1$ and $k = 1$.
2. Form region R_k^M by combining R_i^C and R_j^C by (6).
3. Calculate areas $A^{R_k^M}$, $A^{R_i^C}$ and $A^{R_j^C}$ using (10).
4. Find the maximum area $A_i^{'C}$ of $(M+1)$ regions in R_k^M.
5. IF $\frac{\left|A^{R_k^M} - A_i^{'C}\right|}{A_i^{'C}} \le T_{\max}$ THEN two regions R_i^C and R_j^C have SSV and M is incremented.
   Repeat Steps 2-5 forming R_k^M by merging R_k^M and another region (which has not already been merged) from R^C
6. IF $(M \ge 2)$ THEN increment k and GOTO Step 1
7. Separate region R^D which has D objects with DSV from R^C.

5.2 Selection Strategy for Feature Sets

To segment objects having DSV, an appropriate feature set needs to be determined as any clustering algorithm separately using PL, PI and CIL will be unable to segment these objects [16]. Section 2 showed that for $\Re$ regions, the degree of arbitrariness of FCM using PL increases by $O(\Re^2)$, so in proposing a feature set selection strategy, two scenarios are considered: i) $\Re > 2$ ii) $\Re = 2$. In the former, FCM using CIL provides comparatively better results than using only PL because of the arbitrariness of FCM using PL. Thus selecting CIL and PI as the feature set is fully justified for the initial segmentation in the MISR algorithm. For $\Re = 2$ however, it is necessary to choose a pair of feature sets from PI, PL and CIL. To select the best set, the amount of overlap between pairs of merged regions representing the misclassified pixels of these regions is exploited. The less the degree of overlap, the lower the distribution time complexity and the more visually distinctive are the objects. Conversely, ambiguity during the distribution of overlapping pixels increases proportionally, so the misclassification risk is proportional to any overlap. To minimize this risk therefore, emphasis is given to the pair of feature sets that produces the minimal overlap. The approach used in this paper to calculate the overlap between a pair of merged regions is formalised in Lemma 4.

Lemma 4: The amount of overlap between a pair of merging regions is proportional to the value of the acute angle between the decision boundaries of the initial segmented regions, separately produced using the two selected feature sets.

Proof: Let L_1 and L_2 be two non-parallel decision boundaries for FCM using PL and CIL respectively and θ_1 be the acute angle between them (Figure 7(c)). The two segmented regions yielded by these two decision boundaries are shown in Figure 7(a) (R_1^L and R_2^L) and Figure 7(b) (R_1^C and R_2^C) respectively. The merging of R_i^L with its similar region R_j^C produces a merged region R_K^M. The overlapping area between a pair of merging regions R_K^M and R_{K+1}^M for initially produced region R_i^L

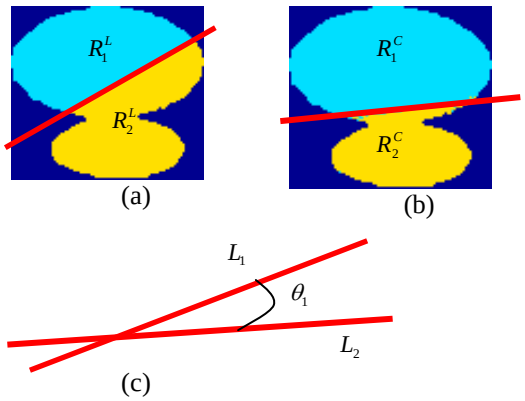


Figure 7: Examples of the initial segmented results produced by FCM using a two-region synthetic image, (a) Only PL, (b) Only CIL, (c) Angle between the two decision boundaries.

considering PL is defined as:-

$$A^{R^M_{k,k+1}} = \sum_{i=1}^{2} \rho_i \theta_1 * A^{R_i^L} \Big/ \pi \qquad (11)$$
$$= \theta_1 \Big/ \pi \times \left(\rho_1 * A^{R_1^L} + \rho_2 * A^{R_2^L} \right)$$

where ρ_i is a matching factor used in calculating the exact area of R_i^L subtended by θ_1. Since $A^{R_1^L}$ and $A^{R_2^L}$ are constant for an image and specific feature set assuming ρ_1 and ρ_2 are constant, then:-

$$A^{R^M_{k,k+1}} \propto \theta_1 \qquad (12)$$

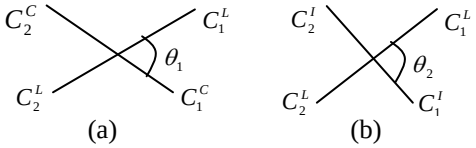


Figure 8: Acute angle between two decision boundaries produced by FCM separately using (a) PL and CIL, (b) PL and PI.

In the best case, L_1 and L_2 are the same so $\theta_1 = 0$ and the overlapping region $A^{R^M_{k,k+1}} = 0$. For the average case $\theta_1 = \pi/4$, while the worst case $\theta_1 = \pi/2$ which corresponds to maximum overlap $A^{R^M_{k,k+1}} = 1/2\, f$, where f is the foreground of objects. Since the maximum overlap is effectively half the foreground, it represents the average cluster size. ■

As previously mentioned, misclassification increases with overlap, so to limit to some extent, the arbitrariness effect of FCM using PL in selecting the feature sets, a constraint is applied to θ_1 by comparing it with its average value i.e. $\pi/4$. To select the best set, the following two cases are now considered, $\theta_1 > \pi/4$ and $\theta_1 \le \pi/4$.

Lemma 5: For any acute angle $\theta_1 > \pi/4$ between the decision boundaries of FCM separately using CIL and PL, the feature sets CIL and PI are used in the MISR algorithm.

Proof: For $\theta_1 > \pi/4$, CIL strongly dominates PL in the segmentation process and has a high misclassification risk (Lemma 4) when merging. It is because of the existence of two objects with quite different brightness values that PI outweighs PL. The feature set CIL and PI will thus produce less overlapping regions. ■

For the case $\theta_1 \le \pi/4$, to minimise misclassification the feature sets are selected based on the minimum value of the angle between the corresponding decision boundaries as follows:-

$$feature\ sets = \begin{cases} CIL, PL & if\ \theta_1\ is\ minmum \\ PL, PI & if\ \theta_2\ is\ minmum \\ CIL, PI & if\ \theta_3\ is\ minmum \end{cases} \quad (13)$$

where θ_1 = Angle between the decision boundaries for FCM using only CIL and PL; θ_2 = Angle for FCM using only PL and PI; θ_3 = Angle for FCM using only CIL and PI.

5.3 The FISFS Algorithm

Having described both the MISR (Section 4) and SOSDS (Section 5.2) algorithms, these are embedded in the *fuzzy image segmentation considering surface characteristics and feature set selection strategy* (FISFS) algorithm detailed in Algorithm 3. Steps 1 and 2 respectively segment the foreground f and separate those objects in the image having SSV and DSV, using the SOSDS algorithm. FCM using CIL is applied since as alluded in Section 3 R^C is used to determine whether objects have either SSV or DSV. For those objects having SSV, FCM using PL is used for segmentation (Step 3), while those objects with DSV, since they are visually distinct in terms of pixel intensity, are segmented by the MISR algorithm. Step 4 considers whether the connectivity feature is to be employed within the MISR algorithm. For two regions, if $\theta_1 > \pi/4$ then CIL and PI (Lemma 5) are used in MISR, however it can be intuitively argued that connectivity should not be applied because while each region has a distinct PI, one or more may possess a similar PI to another region that is actually connected to it. In such circumstances, to eliminate the possibility of misclassification, connectivity is not applied. For all other scenarios, the feature set selection strategy (Section 5.2) with connectivity is exploited in the MISR algorithm due to the potential impact of PL over PI in CIL.

Algorithm 3: Fuzzy image segmentation considering surface characteristics and feature set selection (FISFS)

Precondition: The foreground region f to be segmented, $\Re$, θ_1 and θ_2.
Post condition: The final segmented regions R.
1. Segment f by FCM using CIL into $\Re$ regions represented by R^C.
2. Find R_k^M and R^D using **SOSDS** for R^C.
3. IF $(k \geq 1)$ THEN FOR $i = 1,\ldots,k$
   Segment R_i^M into M regions by FCM using PL.
4. IF $(D \geq 2)$ THEN
   Connectivity=TRUE
   IF $D = 2$ THEN
   IF $(\theta_1 > \pi/4)$ THEN
   Connectivity=FALSE
   Segment R^D into D regions using **MISR** for R^I and R^C (Lemma 5).
   ELSE
   Select feature sets using (13).
   Segment R^D into D regions by **MISR**.
   ELSE
   Segment R^D by MISR using R^I and R^C.

6. Complexity Analysis of FISFS Algorithm

Any visually meaningful object is considered as a region of interest for object-based image segmentation and so the number of regions to be segmented is always constrained.

Assumption 1: Let $\Re$ and n be the number of regions and total number of pixels respectively, where $\Re << n$ so $\Re$ is considered a constant i.e., $O(1)$ with respect to n. ■

The complete time complexity of the FISFS algorithm is now formally presented.

Lemma 6: $\Re$ similar regions are computed in $O(n)$ time.

Proof: The average cluster size is $n/\Re$ pixels. To find one similar region requires $O\left(\Re \times n/\Re\right) = O(n)$ time, so to calculate $\Re$ similar regions needs $O(\Re \times n) = O(n)$, using Assumption 1. ■

Lemma 7: Forming $\Re$ merging regions can be performed in $O(n)$ time.

Proof: A merging region formed from a pair of similar regions needs in the worst case $O\left(2n/\Re\right)$ time, so the total time required for merging $\Re$ similar pairs is $O\left(\Re \times 2n/\Re\right) = O(n)$. ■

Lemma 8: The total time complexity for calculating the overlap between ${}^{\Re}c_2$ pairs of merging regions is $O(n)$.

Proof: In the worst case, the maximum size of any merging region is n, so the time required to calculate the overlap is $O(n)$. Since, the number of pairs of merging regions is ${}^{\Re}C_2$, the total calculation time of the overlap for ${}^{\Re}C_2$ pairs is:-

$O\left({}^{\Re}C_2 \times n\right) = O\left(\Re(\Re-1)/2 \times n\right) = O\left(n\Re(\Re-1)/2\right) = O(n)$. ■

Lemma 9: Removing the overlap from $\Re$ merged regions requires $O(n)$ time.

Proof: On average, the maximum overlap between a pair of $\Re$ merging regions is $\frac{1}{2}\left(n/\Re + n/\Re\right) = n/\Re$ so to remove the overlap from a corresponding pair of merging regions takes $O\left(n/\Re\right)$ time (Lemma 8). For ${}^{\Re}c_2$ merging pairs, the total required time is $O\left({}^{\Re}C_2 \times n/\Re\right) = O(n)$. ■

Lemma 10: The connected components of a set of n pixels can be found in $O(n)$ time [20]. ■

Lemma 11: The distribution of the connected components of n pixels using connectivity requires $O(n\log n)$ time [1]. ■

Lemma 12: The time complexity for the distribution of n pixels using FCM is $O(n)$ [1]. ■

Lemma 13: The time required for the MISR algorithm is $O(n\log n)$.

Proof: From Lemma 6 and Lemma 7, the time required for Steps 1 and 2 are $O(n)$ and $O(n)$ respectively. Step 3 is completed in $O(n)+O(n)=O(n)$ time (Lemma 8 & Lemma 9), while Steps 4 and 5 are in $O(n\log n)$ and $O(n)$ respectively (Lemma 11 & Lemma 12) so the time complexity for MISR is:- $O(n)+O(n)+O(n)+O(n\log n)+O(n)=O(n\log n)$ ■

Lemma 14: The convex hull of n pixels and its area can be computed in $O(n\log n)$ time [21] [18], [19]. ■

Lemma 15: The time complexity for SOSDS is $O(n\log n)$.

Proof: For one iteration, Steps 1, 2, 3, 4, 5, and 6 require $O(1)$, $O(n)$ (Lemma 7), $O(n\log n)$ (Lemma 14), $O(M+1)=O(\Re)$, $O(1)$ and $O(1)$ respectively, since the maximum value of M in Step 4 is $(\Re-1)$. In the worst case, the maximum number of iterations for Steps 2 to 5 is ${}^{\Re}C_2$ considering both internal and external loops for Steps 5 and 6 respectively. The total computational time is thus $O(1)+O\left({}^{\Re}C_2 \times n\right)+O\left({}^{\Re}C_2 \times n\log n\right)+O\left({}^{\Re}C_2 \times \Re\right)+O\left({}^{\Re}C_2 \times 1\right)+O\left({}^{\Re}C_2 \times 1\right)=O(n\log n)$. ■

Lemma 16: The computational time complexity for the FISFS algorithm is $O(n\log n)$.

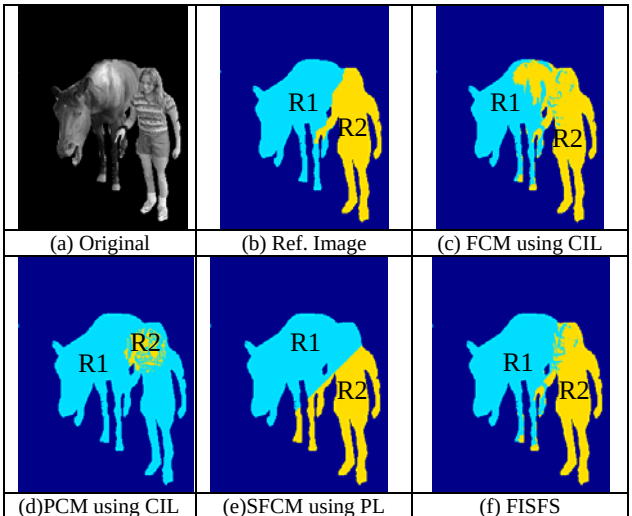
Proof: Using Lemma 12, Steps 1 and 3 are executed in $O(n)$ and $O(n)$ time respectively since in the worst case, the maximum value of k in Step 3 is $\Re/2$, while from Lemma 15, Step 2 requires $O(n\log n)$ time. In Step 4, if connectivity is not applied (Lemma 13), the required time is $O(n)$, otherwise it is $O(n\log n)$ time. Thus, the total time needed for the FISFS algorithm is $O(n)+O(n\log n)+O(n)+O(n\log n) or\ O(n)=O(n\log n)$. ■

7. Experimental Results

The FCM [3], PCM [4], SFCM and new FISFS algorithms were all implemented using Matlab 6.1 (The Mathworks Inc.). For FCM, PCM and SFCM only PI, PL and CIL were used. A total of 146 different natural and synthetic gray-scale images were randomly selected for the experimental analysis, comprising up to 5 different regions (objects) having various

degrees of surface variation, (obtained from IMSI[2], own collection, and the Internet). To segment only the foreground objects in an image, the background was manually removed by setting it to zero. Any zero-valued foreground object pixels were replaced by 1, which had no effect upon visual perception and avoided the possibility of foreground pixels merging with the background. PL in the form of the *x, y* coordinates were normalised within the range [0, 255] in order to constrain them to the same range of PI for 8-bit gray-scale images. The perceptual threshold was set $T_{\max} = 0.05$ as discussed in Section 5.1.

To quantitatively appraise the performance of all the various fuzzy clustering algorithms, the efficient objective segmentation evaluation method, *discrepancy based on the number of misclassified pixels* [1] was used. Two types of error, namely Type I, $errorI_i$ and Type II, $errorII_i$ are computed, the former being the percentage error of all i^{th} region pixels misclassified into other regions, while the latter is the error percentage of all region pixels misclassified into i^{th} region. Representative samples of the manually segmented reference regions together with their original images are shown in Figures 9(a)-9(b), 10(a)-10(b) and 11(a)-11(b). To provide a better visual interpretation of the segmented results, both the reference and segmented regions are displayed using different colours rather than their original gray-scale intensities. Note that only the best segmentation results of FCM, PCM and SFCM with related feature are presented as a comparison with FISFS in Figures 9, 10 and 11.



(a) Original | (b) Ref. Image | (c) FCM using CIL

(d)PCM using CIL | (e)SFCM using PL | (f) FISFS

Figure 9: (a) Original horse image, (b) Manually segmented reference of (a). (c)-(f) Segmented results of (a).

The *horse* image in Figure 9(a) has two different regions: the horse (R_1) and the woman (R_2). The best segmented results for FCM, PCM, SFCM and FISFS are shown in Figure 9(c)-(f). If the segmented results in Figure 9(c)-(e) are compared with the manually segmented reference regions in Figure 9(b), it is visually apparent a large number of pixels of region (R_1) have been misclassified into (R_2) and vice versa. This is because both regions contain certain grey level pixel intensity variations (Figure 9 (c)-(d)) and two objects having different shape and inappropriate orientation for PL are connected to each other (Figure 9 (e)). In contrast, many of these pixels have been correctly classified by the FISFS algorithm in Figure 9(f) because of the strategy employed to select the most appropriate feature sets. The corresponding average Type I and Type II errors for FCM, PCM and SFCM for all feature sets (PI, PL and CIL) and FISFS are given in Table 1, which confirms the improvement of FISFS with an average error of 5.3%, while the next best performance was achieved by FCM using CIL with an average error of 13%.

A second sample image (*scene*) is shown in Figure 10(a), which contains three different regions: sky (R_1), hill (R_2) and water (R_3), each having different pixel intensities. The best segmented results for FCM, PCM, SFCM, and FISFS are shown in Figure 10(c)-(f) respectively. For the results produced by FCM, PCM and SFCM shown in Figure 10(c)-(e), some pixels in R_2 are misclassified into R_3 and a considerable number of pixels of R_3 are misclassified into R_2 and R_1 due to region PI variation. The number of misclassified pixels is significantly reduced with the FISFS algorithm (Figure 10(f)) as a consequence of it merging the individual results produced separately by FCM using PI and CIL. This improvement is again confirmed in Table 1, which shows the average and the next best average error percentages of FISFS and SFCM using CIL were 7.45% and 11.1% respectively.

[2] IMSI's Master Photo Collection, 1895 Francisco Blvd. East, San Rafael, CA 94901-5506, USA.

Table 1: Percentage errors for the segmented regions in Figure 9, 10 and 11.

Error	Region	Image	FCM			PCM			SFCM			FISFS
			PL	PI	CIL	PL	PI	CIL	PL	PI	CIL	
Type I	R1	Horse	10.5	43.4	21	97.2	7.4	3.8	10	43.4	43.4	3
		Scene	51	25.4	40.2	99.7	3.6	96.4	49.4	24.5	24.5	21.2
		Animal	47.8	30.1	0	0	12.8	3.6	0	37.1	37.1	0
	R2	Scene	31.6	0.7	0.34	0	1.14	0.3	33.3	0.8	0.8	0.3
		Animal	0	71.9	38.8	93.1	100	24.3	19.9	69.1	69.1	8.4
	R3	Scene	38.4	15.8	36.6	100	83	0.4	39.3	16.2	16.2	6
		Animal	0.3	92.1	22.1	0	59.4	29	51	86.9	87.7	0.01
	R4	Animal	0	69.8	31.2	99.4	47.9	93.3	0	71.5	75.4	24.4
	R5	Animal	100	67.6	21.9	0	93.2	2.2	100	64.7	64.7	0
Type II	R1	Horse	18.3	27.4	5	0	89.9	89.1	23.7	27.4	27.4	7.6
		Scene	35.8	5.5	22.7	0	49.5	0.2	37.1	6	6	2.45
		Animal	0.02	26.4	0	0	23.9	0	0	25.7	25.7	0
	R2	Scene	18.4	3.5	1.28	99.9	2.3	0.5	18	3.4	3.4	1.7
		Animal	0.03	5.4	4.4	0	0	6.2	0	6.7	6.7	0
	R3	Scene	10.7	16.5	25	0.01	2.2	62.6	10.2	16	16	13.1
		Animal	0	14.2	9.7	23.2	8.4	6.8	0	17.5	23.9	2.1
	R4	Animal	16	18.9	3.5	0	41	0	16	19.3	14.5	0
	R5	Animal	12	16.4	10.5	33.3	3.9	31.3	13.7	12.9	12.9	8.2
Average Error		Horse	14.4	35.4	13	48.6	48.6	46.5	16.9	35.4	35.4	5.3
		Scene	31	11.2	21	49.9	23.6	26.7	31.2	11.1	11.1	7.45
		Animal	17.6	41.3	14.2	24.9	39.1	19.7	20.1	41.1	41.7	4.3

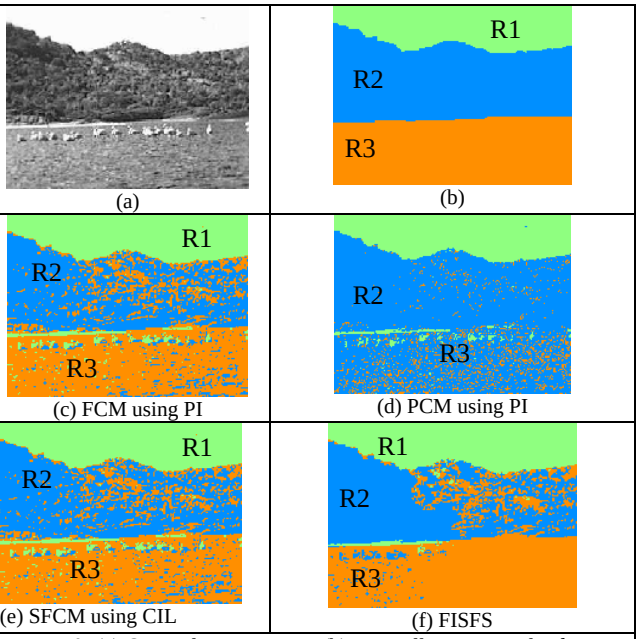
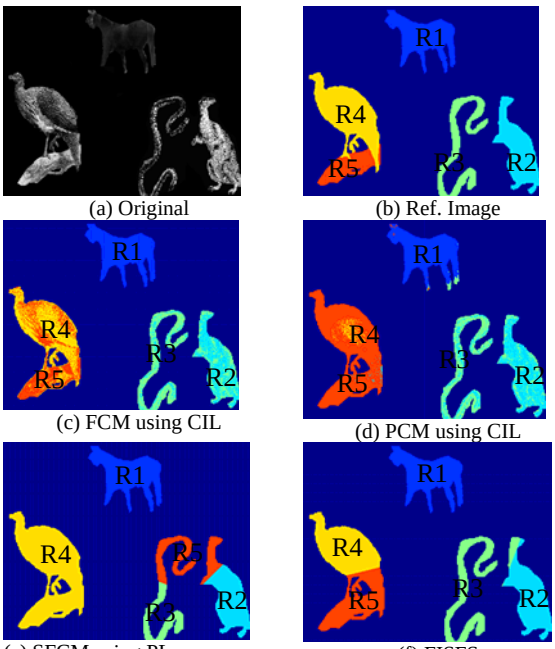


Figure 10: (a) Original scene image, (b) Manually segmented reference of (a). (c)-(f) Segmented results of (a).

The final sample image (*animal*) in Figure 11(a) has five separate regions: horse (R_1), kangaroo (R_2), snake (R_3), peacock (R_4) and the tree branch (R_5). Note, particularly that (R_2) and (R_3) as well as (R_4) and (R_5) both have SSV, while (R_1) has only DSV. The best segmented results of each algorithm are shown in Figure 11(c)-(f) respectively. For objects having SSV, FCM and PCM using CIL and SFCM using PL could neither separate (R_4) from (R_5) nor also (R_2) from (R_3) (Figure 11(c)-(e)) because they do not consider SSV for each object. FISFS in contrast clearly separated (R_3) from (R_2) and (R_5) from (R_4), with exception of a very small number of misclassified pixels of (R_2) and (R_4) due to the arbitrariness of FCM using PL. This again emphasises the consistent superior performance of FISFS compared to FCM, PCM and SFCM, which is due to embracing the concept of SSV and DSV for objects and incorporating a strategy to select the most appropriate feature sets. FISFS also generated a lower average error of 4.3% compared with the best average error values of 14.2%, 19.7% and 20.1% for FCM, PCM and SFCM respectively (see Table 1) for this image.



(a) Original (b) Ref. Image

(c) FCM using CIL (d) PCM using CIL

(e) SFCM using PL (f) FISFS

Figure 11: (a) Original animal image, (b) Manually segmented reference of (a). (c)-(f) Segmented results of (a).

To assimilate the overall segmentation performance of FISFS, it needs to be highlighted that only the best results for each clustering algorithm (FCM, PCM and SFCM) and the three feature sets (PL, PI and CIL) were considered. This means 9 different combinations[3] (see Figures 9-11) were compared with the FISFS algorithm. Of the 146 test images, FISFS produced superior results for 52 and of the remaining 94 images; FCM, SFCM and PCM provided better results for only 27, 23 and 14 images respectively (Figure 12). Figure 13 shows that the average percentage error of the new FISFS algorithm, for all 146 images is 16.77% compared with the best average percentage errors for FCM, SFCM and PCM of 20.5%, 24.7% and 33.7% respectively, again endorsing the improved performance of the FISFS algorithm compared with the other three clustering algorithms for all three feature sets. An analysis of the distribution of images where superior results were obtained revealed a high dependency upon the actual number of clustering algorithms used for comparative purposes, the different features selected and number of objects used in the experiments. Since the test image set was specifically constructed so that all possible data sets were considered, embracing different types of objects and features using different clustering algorithms, the overall superiority of the FISFS algorithm is considerably significant.

Finally, the sensitivity of the perceptual threshold $T_{\max}$ in the SOSDS algorithm was analysed in respect to its effect upon the average percentage error of all images. Figure 14 shows $T_{\max}$ plotted over a range of different threshold settings and the results confirm the negligible impact upon overall performance, thus vindicating the choice of $T_{\max} = 0.05$ in Section 5.1.

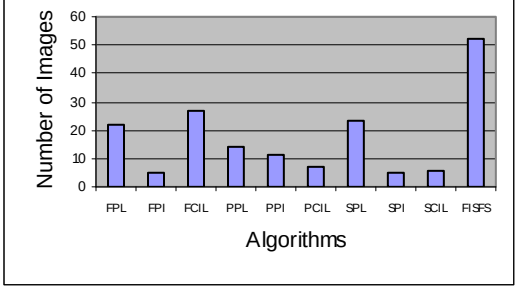


Figure 12: The best results for different algorithms[3]

[3] To clarify the nomenclature in Figures 12 and 13, the initial letters F, P, S represent the FCM, PCM and SFCM algorithms respectively, while the following letters indicate which feature was used (PI, PL or CIL).

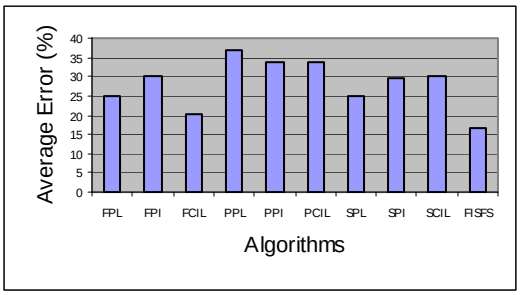


Figure 13: The average error percentages for different algorithms[3]

8. Conclusions

This paper has presented a new *fuzzy image segmentation considering surface characteristics and feature set selection strategy* (FISFS) algorithm which exploits features such as connectivity, the arbitrariness of FCM and object surface similarity, to provide superior segmentation performance compared with the FCM, PCM and SFCM clustering algorithms for three separate features and a wide variety of image types. Two special algorithms specifically manage the important merging of initially segmented regions and the separation of objects having similar and dissimilar surface variations in an image, with a perceptual threshold incorporated to assist in surface variation identification. A concise time-complexity and threshold sensitivity analysis of the FISFS algorithm has also been presented.

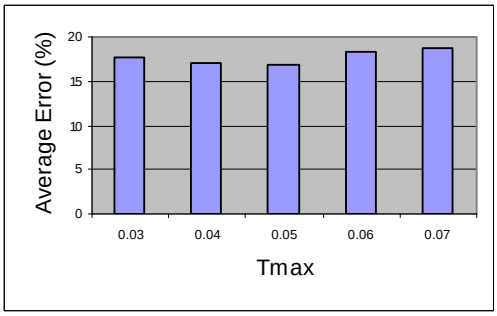


Figure 14: Sensitivity analysis of the threshold T_{max}.

References

[1] Gour C Karmakar, 2002. An integrated fuzzy rule-based image segmentation framework, PhD, Thesis Dissertation.

[2] I. Gath, andA. B. Geva, 1989. Unsupervised optimal fuzzy clustering, Pattern Analysis and Machine Intelligence, 2(7), 773-781.

[3] J.C. Bezdek, 1981. Pattern recognition with fuzzy objective function algorithm, Plenum Press, New York.

[4] R. Krishnapuram and J. M. Keller, 1993. A possibilistic approach to clustering, Fuzzy Systems, 2(2), 98-110.

[5] Y. A. Tolias and M. Panas, 1998. On applying spatial constraints in fuzzy image clustering using a fuzzy rule-based system, IEEE Intern. Con. on Signal Processing Letters, 5(10), 245-247.

[6] A. W. C. Liew, S. H. Leung and W. H. Lau, 2000. Fuzzy image clustering incorporating spatial continuity, IEE Proc.-Vis on Image Signal Process, 147(2), 185-192.

[7] Z. Chi, H. Yan, and T. Pham, 1996. Fuzzy algorithms: with applications to image processing and pattern recognition, World Scientific Publishing Co. Pte. Ltd., Singapore.

[8] Jian Yu, Houkuan Huang and Shengfeng Tian, 2002. An efficient optimality test for the fuzzy c-means algorithm, IEEE Intern. Con. on Fuzzy Systems, 98-103.

[9] Carl G. Looney, 2002. Interactive clustering and merging with a new fuzzy expected value, Pattern Recognition Letters, 35(11), 2413-2423.

[10] A. W. C. Liew, S. H. Leung and W. H. Lau, 2000. Fuzzy image clustering incorporating spatial continuity, IEE Proc.-Vis on Image Signal Process, 147(2), 185-192.

[11] David Altman, 1999. Efficient fuzzy clustering of multi-spectral images, IEEE Intern. Con. on Geoscience and Remote Sensing Symposium, 3, 1594-1596.

[12] Gerardo Beni and Xiaomin Liu, 1994. A least biased fuzzy clustering method, Pattern Analysis and Machine Intelligence, 16(9), 954-960.

[13] Frank Hoppner and Frank Klawonn, 2001. A new approach to fuzzy partitioning, NAFIPS Intern. Conference, 3, 1419-1424.

[14] Hao-Jun Sun, Sheng-Rui Wang and Zhen Mei, 2002. A fuzzy clustering based algorithm for feature selection, Intern. Con. on Machine Learning and Cybernetics, 1993-1998.

[15] Jiu-Lun Fan, Wen-Zhi Zhen and Wei-Xin Xie, 2003. Suppressed fuzzy c-means clustering algorithm, Pattern Recognition Letters, 24, 1607-1612.

[16] M. Ameer Ali, G.C. Karmakar and L. S. Dooley, 2004. Image segmentation using fuzzy clustering incorporating spatial information, IASTED Intern. Con. on Artificial Inteligence and Applications, 2, 878-881.

[17] M. Ameer Ali, L. S. Dooley, and G.C. Karmakar, 2003. Fuzzy image segmentation using location and intensity information, IASTED Intern. Con. on Visualization, Imaging and Image processing, 399-404.

[18] Herve B, John Iacono, J Katajainen, P Morin, J Morrison and G Toussaint, in-place planar convex hull algorithms, http://photon.poly.edu/~hbr/ publi/inplace-latin02.pdf

[19] G Aloupis and B Kaluzny. The three-coins algorithm for convex hulls of polygons, http://cgm.cs.mcgill.ca/~beezer/cs507/main.html

[20] M. B. Dillencourt, H. Samet and M. Tamminen, 1992. A general approach to connected component labelling for arbitrary image representations, Journal of the ACM, 39, 253-280.

[21] P.F. Preparata and I.M Shamos, 1985. Computational geometry, Springer Verlag New York Inc., New York.