

INF???: Machine Learning Final Report

**Skin Segmentation**

Sasha Nicolas

Axelle Pochet

Professor: Ruy Milidiu

Pontifical Catholic University of Rio de Janeiro

Rua Marquês de São Vicente, 225, Gávea, Rio de Janeiro

State of Rio de Janeiro

22451-900

Table of contents

[1. The Dataset 4](#_Toc405978325)

[2. State of the Art 5](#_Toc405978326)

[3. Fuzzy decision tree 5](#_Toc405978327)

[3.1 General principle 5](#_Toc405978328)

[3.2 Fuzzy clustering 6](#_Toc405978329)

[3. 2.1 Fuzzy c-mean algorithm 6](#_Toc405978330)

[3.2.2 Membership functions 7](#_Toc405978331)

[3.3 Tree construction 8](#_Toc405978332)

[3.3.1 Usual decision trees 8](#_Toc405978333)

[3.2.2 Fuzzy decision trees 9](#_Toc405978334)

[3.4 Testing 11](#_Toc405978335)

[3.4.1 Rules extraction 11](#_Toc405978336)

[3.4.2 Example 12](#_Toc405978337)

[3.4.3 Testing results 12](#_Toc405978338)

[4. Application on new data 12](#_Toc405978339)

INTRODUCTION = motivations, tarefa

# The Dataset

Any visible color can be described by three components {R, G, B} (resp. Red, Green, Blue), each between 0 and 255. Visible colors are ranged between black {0, 0, 0} and white {255, 255, 255}. The present dataset is a collection of pixel colors, randomly sampled in photographs containing people from different ages, skin colors and gender (from the FERET database[[1]](#footnote-1) and the PAL database[[2]](#footnote-2)). Each pixel was classified as being skin or not skin.

The dataset then consists in 245 057 instances corresponding each to 1 pixel with its 3 attributes B, G, R ϵ [0,255] and its corresponding decision class Y ϵ [1;2] respectively Skin or Non-Skin (Figure 1).

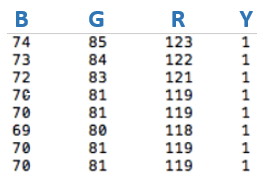
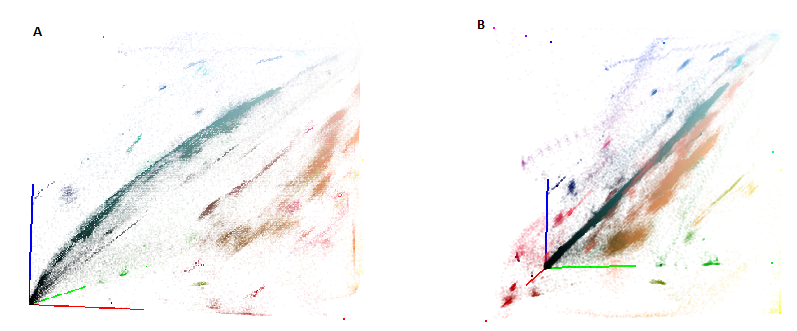


Figure : Dataset sample. https://archive.ics.uci.edu/ml/datasets/Skin+Segmentation.

The dataset is balanced as follows: 50859 Skin (Y = 1) against 194 198 Non-Skin (Y = 2).

Points {R, G, B} were displayed in 3D to allow visual appraisal of the data (Figure 2). Points were given a transparency: the more visible, the more point density.



B

B

G

G

R

R

Figure : Dataset in 3D, points coordinates are (R,G,B). Colors of the axis correspond to each attributes. Points were given a transparence, allowing to see better where they form clusters. (A) and (B) 2 different points of view.

# State of the Art

[Bhatt et al., 2009] used a Fuzzy Decision Tree to classify the data. The term “fuzzy” is a synonym of “blur”: here it represents the limits between different colors that are not clear, as shown in Figure 3. The Fuzzy Decision Tree takes into account the fuzziness of the data, as detailed in Part 3.1.



Figure : Colors vary smoothly, continuously. It is not possible to place exact limits between them.

[Bhatt et al., 2009] arrive at a result of 94.1% of accuracy. Their confusion matrix is given below (Figure 4).

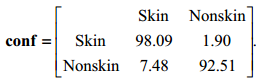


Figure : Confusion matrix from [Bhatt et al., 2009]. The main errors come from pixels asserted Skin whereas they were Non-Skin (7.48%).

# Fuzzy decision tree

## 3.1 General principle

As introduced in Part 2, Fuzzy Decision Trees take into account the blur that may occur at the limit between classes.

For example, say we want to determine which sport to play (decision class *Running* or *Volleyball*), knowing the temperature of the day (attribute *temperature*). We know that if it is hot, we will practice volleyball and if it’s cold we will rather run. The temperature for day *d* is 20◦C. Is it hot or cold? This appraisal depends on every person. Consequently, fuzzy logic allows one attribute value to belong to several sub-classes, with a degree of certainty. The sentence “20◦C is rather cold” could be quantified as 20◦C = [*hot*: 30%; *cold*: 70%]. Sub-classes *hot* and *cold* are called the attribute’s fuzzy subsets. Note that fuzzy subsets are different from the decision class: they are specifically linked to one attribute and just allow to “cut” its values into different meaningful sub-classes (see Table 1).

|  |  |  |
| --- | --- | --- |
| **Temperature** | | **Decision class** |
| *Cold* [0◦C, 25◦C] | *Hot* [20◦C, 40◦C] | [Volleyball;Running] |

Table : Difference between attribute (temperature), fuzzy subsets (cold, hot) and decision class (volleyball, running). Note that fuzzy subsets may overlap.

A Fuzzy Decision Tree includes fuzzy subsets in its ramification: nodes are attributes, branches are fuzzy subsets and leaves contain the probability to belong to each decision class.

In this project the following process was then followed:

1. Cut attributes (R,G,B) into fuzzy subsets using fuzzy clustering,
2. Build the tree with a training set of 80% of the data,
3. Extract rules to read the tree and test it with the remaining 20% of the data.

## 3.2 Fuzzy clustering

### 3. 2.1 Fuzzy c-mean algorithm

The fuzzy c-mean algorithm was used to cluster the data: it allows one instance to belong to several clusters at the same time, with an associated degree of membership. The method was first developed by [Dunn et al., 1973] and consists in minimizing the following objective function:

|  |  |  |
| --- | --- | --- |
|  |  | ( 1 ) |

Where *µij* is the degree of membership of point *xi* to cluster *j* with center *cj*. *N* is the number of data points and *C* is the number of clusters to extract. *m* is the so-called fuzzifier that will give information on the degree of fuzzification (mixing) of the data. It is generally set to 2 (). The norm here represents the Euclidean distance.

This minimization is carried out through an iterative process that updates *µij* and *cj* until for every *µij*, with *k* the iteration step.

The algorithm proceeds as follows:

1. Initialize matrix with random values between 0 and 1;
2. Calculate for current step (k) the centroids vectors as follows:

|  |  |
| --- | --- |
|  |  |

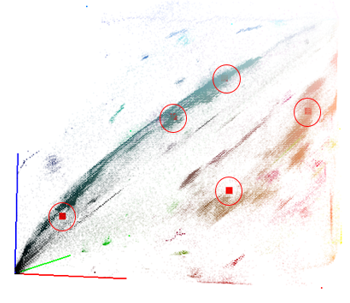
1. Update *U(k)* to *U(k+1)*:

|  |  |
| --- | --- |
|  |  |

1. If , Stop. Else go to step (2).

For clarity this algorithm was described for *µij* a 1D value; here it is a vector (R,G,B), an so is each centroid .

Following [Bhatt et al., 2009] the clustering was first done with 5 clusters (*C* = 5). Figure 5 shows the cluster centers (centroids, (c1, c2, c3, c4, c5)) in the 3D representation.



**C5**

**C4**

**C3**

**C2**

**C1**

B

G

R

Figure :

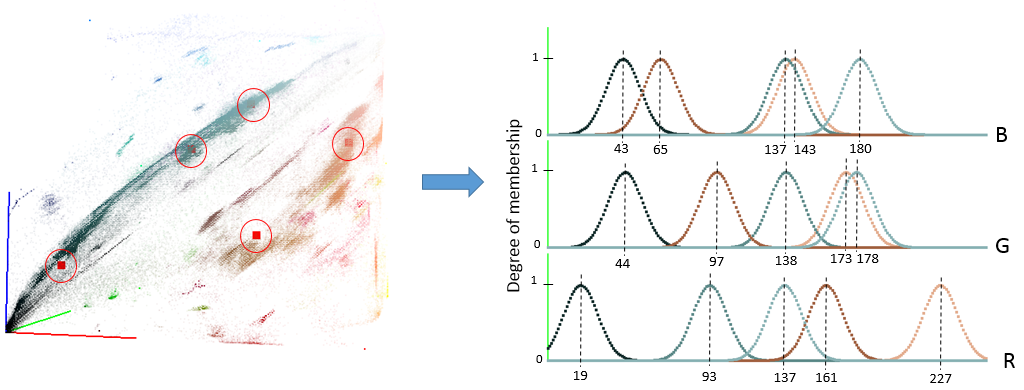
Given centroids, it is now possible to extract membership functions. Membership functions allow to give the degree of membership of any point (R,G,B) to each cluster, and will be used to build the Fuzzy Decision Tree.

### 3.2.2 Membership functions

Membership functions are associated to each cluster. They are tools that will assign a degree of membership to each cluster for any point (R,G,B). For example, one could use the Euclidean distance from a point to each centroid to assign the degree of membership: the closer is a point to a centroid, the higher is its probability to belong to the associated cluster; if a point is too far from a centroid, it sure does not belong to the associate cluster (degree of membership = 0). Following this idea, we applied Gaussian functions on each centroid:

|  |  |  |
| --- | --- | --- |
|  |  | ( 2 ) |

With the degreeof membership of point to the current cluster, *µ* the center of the cluster and *σ* a parameter that assigns a width to the Gaussian. Here we can see the Gaussian membership function as a 3D Gaussian function around each centroid, however we separated it to three 2D functions for each centroid, as shown in Figure 6.



**C5**

**C5**

**C5**

**C4**

**C4**

**C4**

**C3**

**C3**

**C3**

**C2**

**C2**

**C2**

**C1**

**C1**

**C1**

**C5**

**C4**

**C3**

**C2**

**C1**

B

G

R

Figure : ε = 0.0005. σ = 10.

Functions overlap: one point (R,G,B) can belong to many clusters. An example is given in Figure 7, where the point (19, 97, 50) is expressed using fuzzy subsets.

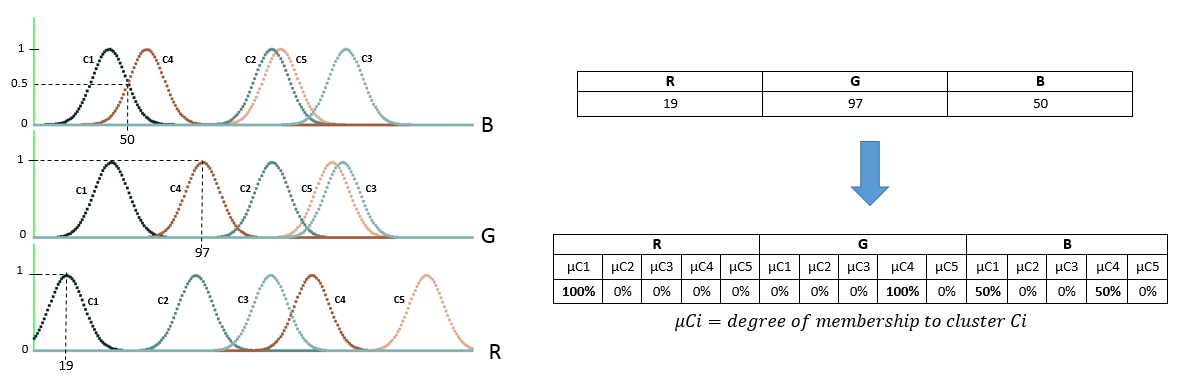
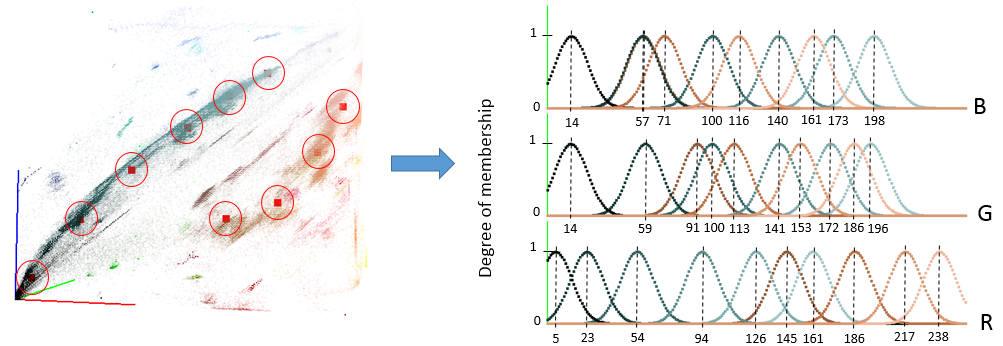


Figure : For attribute B, the point belong to two clusters c1 and c5 with the degree of membership 50% for each.

The choice of parameter σ has consequences on the fuzziness of the data: larger Gaussian functions overlap more.

Visually on Figure 6, two clusters may correspond to skin color (c4 and c5). However it seems like there is a wider range of colors for human skin and looking at the initial data it was assumed that more clusters could be found for skin color. The second clustering was then increased to 10 clusters. Figure 8 shows the result for *C* = 10, with the same parameters as before.



**C10**

**C10**

**C10**

**C9**

**C9**

**C9**

**C8**

**C8**

**C8**

**C7**

**C7**

**C7**

**C6**

**C6**

**C6**

**C5**

**C5**

**C5**

**C4**

**C4**

**C4**

**C3**

**C3**

**C3**

**C2**

**C2**

**C2**

**C1**

**C1**

**C1**

**C10**

**C9**

**C8**

**C7**

**C6**

**C5**

**C4**

**C3**

**C2**

**C1**

Figure : ε = 0.0005. σ = 10.

The second clustering highlighted new shades for skin color (c7, c8, c9 and c10). However the process was longer, as resumed in Table 2. In the following, the two parameterizations are used to build the decision tree and show that better results are obtained with 10 clusters.

## 3.3 Tree construction

### 3.3.1 Usual decision trees

An example of simple decision tree is given in Figure 9, where the goal is to determine which sport to play according to the current climate. Here attributes are categorical (the entry is not a temperature degree for example, but directly the appreciation hot or cold). Nodes of the tree correspond to attribute, branches are the different values for that attribute and leaves contain the final decision with a degree of certainty (basically the frequency of the decision class in the training table for the given path).

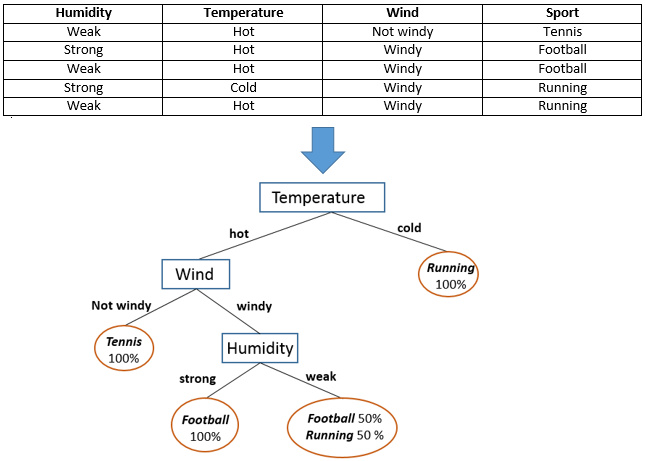


Figure : Attributes are Humidity, Temperature and Wind. Classification concerns the sport to play: Tennis, Football or Running. The reading of the training table leads easily to the given decision tree.

Decision trees can quickly become hard to follow and to conceptualize with other kinds of problems (high number of attributes or decision classes, ...). The ID3 (Iterative Dichotomiser 3) algorithm aims at automatizing the tree construction. It was developed by [Quinlan, 1986] and is widely used because of its simplicity and efficiency.

The idea is to preferentially create nodes with attributes that are the most “relevant”, that is to say the attribute that best separates the data, that has the best *Information Gain.* The Information Gain is calculated from the global entropy of the data. Equations ( 3 ) and ( 4 ) give the formulas of Entropy and Information Gain for a set S={x1, x2,...,xk} with *N* decision classes *c*.

|  |  |  |
| --- | --- | --- |
| **Global Entropy** | , with | ( 3 ) |
| **Attribute Information Gain** | ,  *with the size of subset Sv ϵ S of training examples xj with attribute v.* | ( 4 ) |

Fuzzy decision trees are slightly different because each attribute is divided into continuous fuzzy subsets, consequently the Entropy has to be calculated for each attribute. In addition, one instance of the training table can correspond to several paths in the tree.

### 3.2.2 Fuzzy decision trees

In the Fuzzy Decision Tree, nodes are the attributes, branches correspond to fuzzy subsets and leaves contain the final decision with its probability to be accurate (“truthfulness”).

In our context, the training table is given by Table 2 in the case of 5 clusters (ie. 5 fuzzy subsets). Each attribute now does not store one unique value but 5 degrees of membership to each of its fuzzy subset.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **R** | | | | | **G** | | | | | **B** | | | | | **Y** |
| µC1 | µC2 | µC3 | µC4 | µC5 | µC1 | µC2 | µC3 | µC4 | µC5 | µC1 | µC2 | µC3 | µC4 | µC5 | **Skin**/**Non-Skin** |

Table : Attributes are R,G,B. Decision class Y corresponds to Skin or Non-Skin. µCi is the degree of membership of the current example to the fuzzy subset i.

The tree is once again built on the principle of maximizing the Information Gain at the creation of each node. Equations ( 5) and ( 6 ) give the formulas of Entropy and Information Gain in the case of fuzzy classification, for a set S={x1, x2,...,xk} with *N* decision classes *c*. The global entropy is not modified (see Equation ( 3 )).

|  |  |  |
| --- | --- | --- |
| **Attribute Entropy** | ,  with membership degree of *j*th pattern at class *i*. | ( 5 ) |
| **Attribute Information Gain** | , | ( 6 ) |

Another important issue in the tree construction is the condition to create a leaf:

1. The β *cut* is the minimal frequency of the given decision class to stop tree expansion. It is usually given between [0, 0.4] and was here chosen as 0.2.
2. The number of remaining training examples at each node has to be above a given *n*. Here it was set to 1.
3. The Information Gain has to be above an ε chosen here as 0.2.

Using those definitions, the FID3 (Fuzzy ID3) algorithm proceeds as follows:

1. Calculate Information Gain for each remaining attribute;
2. If one of this condition is met :
   1. the frequency of one class is inferior to the β cut,
   2. the number of instances is inferior *n*,
   3. all information gains are inferior to ε;

then create a leaf and give the frequency of each decision class.

1. Else select the Attribute with the higher information gain and create a node.
2. Create one branch for each fuzzy subset. If no instances are left for one subset, do not create a branch for it.
3. For each branch, if some attributes remain, go to step (1).

In the current work, the tree was built with the free software FID3.4[[3]](#footnote-3). To do so, three entry files were generated (an example of each file can be found in annex):

* data.file gives the information about the name, number and type (categorical or numerical) of each attribute, the number of fuzzy subsets with their name, type and range, and the name and type of the decision class. The membership functions are simplified as shown in Figure 10 and the 4 inflexion points are passed to the file.

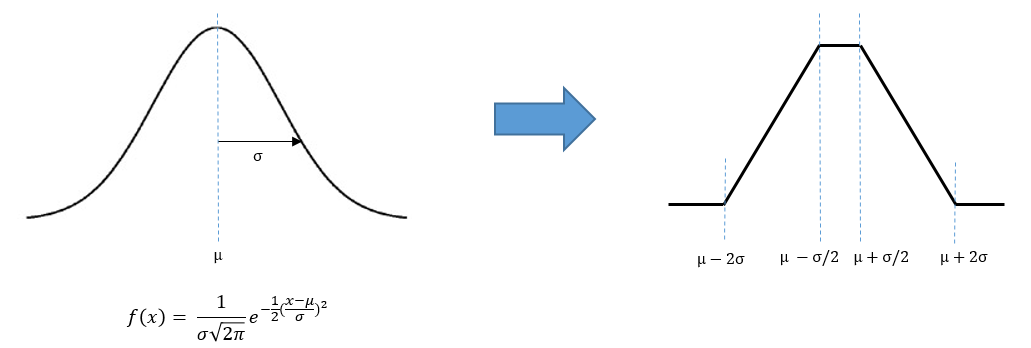


Figure : Simplification of the Gaussian membership functions for entry file in FID3.4 software. To describe the function, only 4 points are given, scaled between 0 and 1: {(µ-2σ),( µ-σ/2),( µ+σ/2),( µ+2σ)}.

* par.template contains all the parameters and options to construct the tree;
* train.file gives the training set with values for each attribute and given decision class. Here we used 200 000 instances for training (~80% of the dataset).

The output files are the following:

* tree.file reminds some information on the chosen parameters and details the tree in a compact form;
* test.file is an optional file that gives results of a testing set if a test file was given.

A visual simplification of the resulting tree for 10 clusters is presented on Figure 11. The tree has 75 leaves.

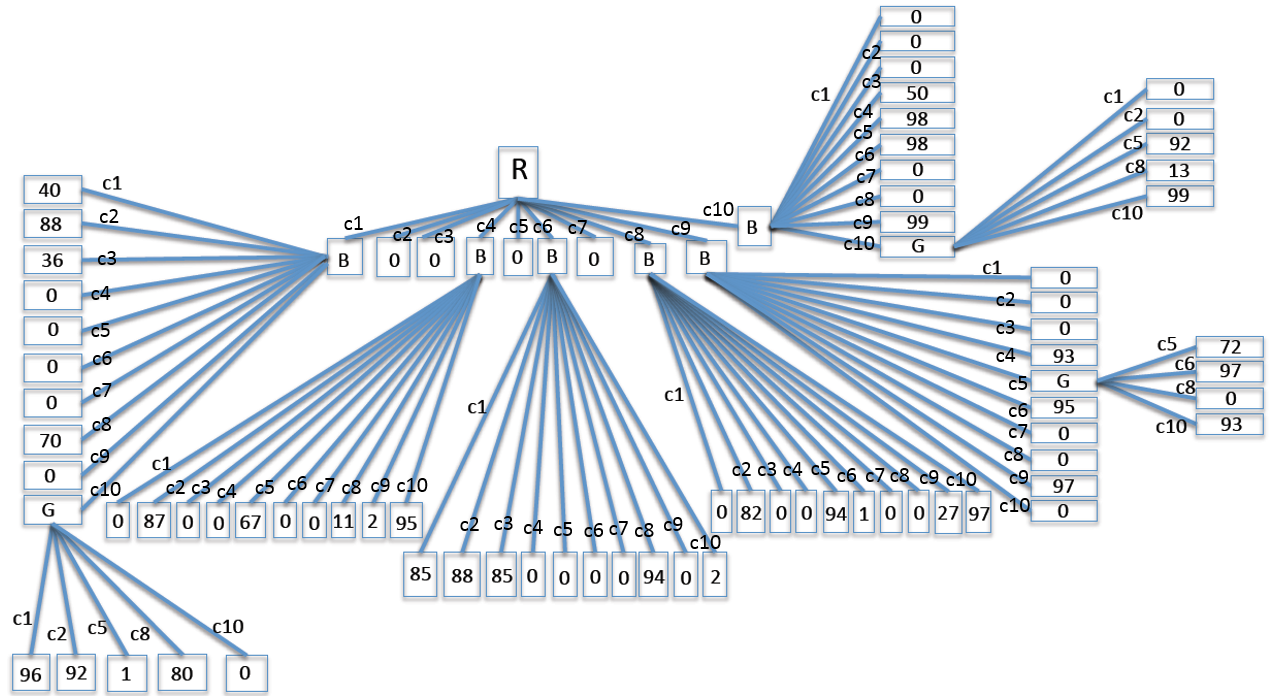


Figure : Resulting fuzzy decision tree with 10 clusters (resp. c1 to c10). The attribute with the global maximum information gain is R. Values on the leaves correspond to the frequency of decision class Skin. β-cut =0.2; n = 1; ε = 0.2.

## Testing

### 3.4.1 Rules extraction

Given a new instance, we now have to read the tree in order to determine the class [Skin / Non-Skin] of this instance. To do so, rules are extracted from the tree. Each path of the tree can be described with basic IF, AND, THEN rules. Taking the simple example of Figure 9, paths are described as follows:

* Path 1: IF hot AND not-windy THEN tennis 100%;
* Path 2: IF hot AND windy AND strong THEN football 100%
* Etc...

In the Fuzzy Decision Tree, a new instance *e* can follow different paths and so a weight *W* is given to each path *r*:

|  |  |  |
| --- | --- | --- |
|  | ,  *with the degree of membership of e associated to branch a.* | ( 7 ) |

From here the degree of membership of *e* as regards to decision class *ck* is calculated as follows:

|  |  |  |
| --- | --- | --- |
|  | *,*  *with the truthfulness of decision class ck at the end of path r, that is its final frequency.* | ( 8 ) |

### 3.4.2 Example

The instance *e*(10, 186, 180) has to be classified as Skin or Non-Skin using the Decision Tree generated with 10 clusters given in Figure 11. Figure 12 shows the corresponding color.



Figure : Color (10,186,180) is between green and blue. It can be surely classified as Non-Skin at least for Earth human beings.

Reading graphs on Figure 8, we express *e* by its degree of membership to each cluster (Table 3, Table 4 and Table 5).

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **R** | | | | | | | | | |
| µC1 | µC2 | µC3 | µC4 | µC5 | µC6 | µC7 | µC8 | µC9 | µC10 |
| 0.5 | 0.5 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Table : Degree of membership of e to each cluster for Attribute R.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **G** | | | | | | | | | |
| µC1 | µC2 | µC3 | µC4 | µC5 | µC6 | µC7 | µC8 | µC9 | µC10 |
| 0 | 0 | 0 | 0 | 0.5 | 0.5 | 0 | 0 | 0 | 1 |

Table : Degree of membership of e to each cluster for Attribute G. Here e partially belongs to 3 clusters, showing that the sum of the degrees of membership is not necessarily equal to 1.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **B** | | | | | | | | | |
| µC1 | µC2 | µC3 | µC4 | µC5 | µC6 | µC7 | µC8 | µC9 | µC10 |
| 0 | 0 | 0 | 0 | 0.5 | 0.5 | 0 | 0 | 0 | 0 |

Table : Degree of membership of e to each cluster for Attribute B.

We then calculate the weight for each path:

* Path 1: = 0.5 \* 0 \* 0 = 0;
* ...
* Path 13: = 0.5 \* 1 \* 0.5 = 0.25;
* ... and so on for the 75 paths.

Finally only 2 paths are not equal to 0 (Figure 13): and . The final probability for *e* to be Skin is:

*0.0025*

*e* has a 0.25% chance to be Skin and consequently a 99.75% chance to be Non-Skin. It is classified as Non-Skin.

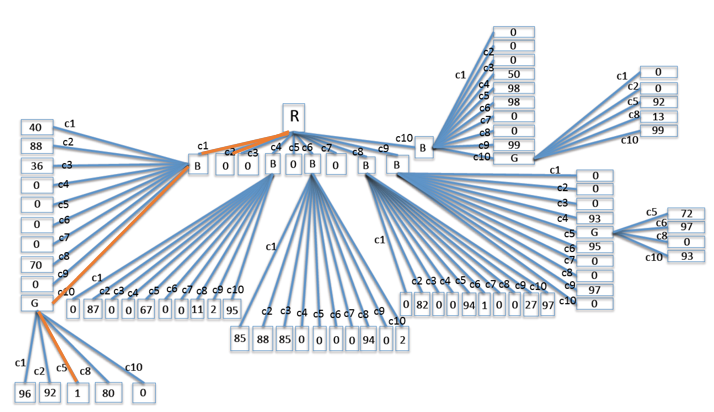


Figure : Given instance e, only 2 paths have a weight different from 0. They are underlined in orange.

### 3.4.3 Testing results

A test.file was generated with the remaining 45 057 instances (~20% of the dataset) and passed to FID3.4. The given results are presented in Table 2. With 5 clusters, although using the same number of clusters, the results are inferior to [Bhatt et al., 2009]. This may be due to the method to build the tree that is slightly different (they use a Low Complexity Tree, meaning that they reduce its size to the maximum). For 10 clusters however the results obtained here are better (0.96 against 0.94).

|  |  |  |  |
| --- | --- | --- | --- |
| **Number of clusters** | **Clustering time** | **Building tree** | **% success** |
| 5 | 5 min | < 1 min | 0.8 |
| 10 | 15 min | < 1 min | 0.96 |

Table : Results from cross-validation, testing tree with FID3.4 for different entries.

The confusion matrix is given in Table 7:

|  |  |  |
| --- | --- | --- |
| ***Decision***  ***Truth*** | **Skin** | **Non-Skin** |
| **Skin** | 95.30 | 4.60 |
| **Non-Skin** | 3.80 | 96.10 |

Table : Confusion matrix from cross validation, testing tree with FID3.4 with 10 clusters. The main error comes from pixels asserted Non-Skin whereas they were Skin (4.60%). This error is smaller than the worst error in [Bhatt et al., 2009] confusion matrix (7.48%).

# Application on new data

The algorithm was finally run on new images where the detection of the skin can be estimated visually but where we don’t know each pixel class in advance. Values of each image pixel was converted into a test file containing one (R,G,B) value per line. Given the output, all pixels asserted as Skin were set white, all pixels asserted Non-Skin were set black. Figure 14 and Figure 15 present results when using the 10 clusters, while Figure 16 allow to compare the results between 5 an d10 clusters.



Figure : Most of the faces are well detected, however some skin remain undetected and in the back a shirt was asserted as skin.



Figure : In this picture The skin is really well extracted, however some non-skin elements were extracted as well: writing on the shirts, shirts, a piece of the ceiling.



Figure : Comparison for results using a Fuzzy Decision Tree built from 5 clusters and the Fuzzy Decision Tree built from 10 clusters. For the second, more Skin was extracted but also more Non-Skin (background).

Conclusion

**ANNEX 1**

Examples or samples of files involved in FID3.4. For details, see the user manual: http://www.cs.umsl.edu/~janikow/fid/fid34/manuals/userMan34.pdf

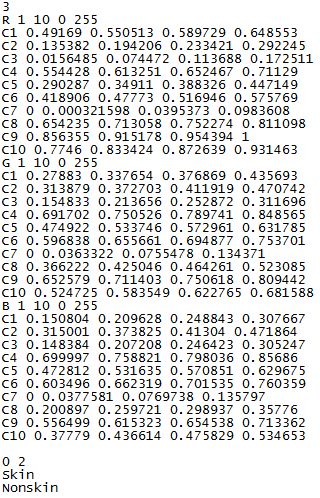


Figure : (INPUT FILE) data.file in the case of 10 clusters. 3 is the number of attributes. R is a numerical attribute (label 1) with 10 fuzzy subsets and values between 0 and 255. The 4 values in front of each cluster are the 4 inflexion points of the simplified membership function, scaled between 0 and 1.In the end the decision class is described, it is categorical (label 1) with only 2 possibilities: Skin or NonSkin.

TRAIN.FILE

PAR.TEAMPLATE

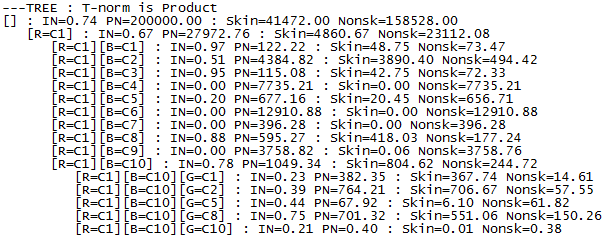


Figure 18: (OUTPUT FILE) Sample of the tree.file. This is the compact form to draw the tree. IN = information gain; PN = number of instances. [R=C1] [B=C1] corresponds to branch between R and B with label cluster C1, then branch C1 from B to a leaf with information given on the corresponding line.

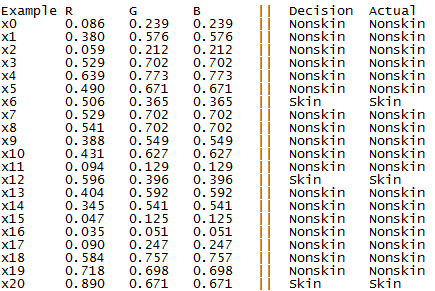


Figure : (OUTPUT FILE) Sample of the test\_results.file. Here every example was asserted well. Values for R,G,B are scaled between 0 and 1.

References

Rajen Bhatt, Abhinav Dhall, 'Skin Segmentation Dataset', UCI Machine Learning Repository.

BHATT, R., DHALL, A., SHARMA, G. and CHAUDHURY, S. Efficient skin region segmentation using low complexity fuzzy decision tree model. *Proceedings of 2009 Annual IEEE India Conference, pages 1–4, 2009.*

DUNN, J.C. Dunn. A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters. *Journal of Cybernetics* 3: 32-57, 1973

QUINLAN, J.R. Induction of decision trees. *Machine learning* 1.1: 81-106, 1986

DONNER LES REFS SUR LES FID3 QUI EXPLIQUENT LE GAIN, L’EXTRACTION DE REGLES ETC

1. http://face.nist.gov/colorferet/request.html [↑](#footnote-ref-1)
2. http://agingmind.utdallas.edu/facedb/ [↑](#footnote-ref-2)
3. http://www.cs.umsl.edu/~janikow/fid/index.html [↑](#footnote-ref-3)