# Characterization and normalization of the image database

Szilárd Vajda

## ▶ To cite this version:

Szilárd Vajda. Characterization and normalization of the image database. [Contract] A02-R-480 ‖ vajda02a, 2002. inria-00107622

HAL Id: inria-00107622

https://inria.hal.science/inria-00107622

Submitted on 19 Oct 2006

# EDF PROJECT

## Multi-oriented and multi-size alphanumerical character recognition

## PART I.

## Characterization and normalization of the image database

This report corresponds to the task n.1 of the contract entitled:
**"Caractérisation et normalisation de données textuelles"**

July 2002 – August 2002

Szilard VAJDA

**READ Group**
**LORIA**

# 1. Plan

- Database
- Data normalization
- Remarks concerning the database and the normalization

# 2. Database

We have a database containing 39889 images. In the database their exist the numbers from 0..9, respectively the letters from A..Z and a..z. and @ the junk class. By junk class I mean objects which couldn't be identified as letters or numbers.
The labelling process of these images were made by our Java tool, developed by Francois Campana, Read Group. In the Tableau 1: Representation of the characters in the database it is mentioned exactly how many samples are in each character class**.**

**Tableau 1: Representation of the characters in the database**

| Character class | Samples Number | Character Class | Samples Number | Character Class | Samples Number |
|---|---|---|---|---|---|
| 0 | 4032 | 1 | 2321 | 2 | 1731 |
| 3 | 1361 | 4 | 861 | 5 | 1484 |
| 6 | 629 | 7 | 622 | 8 | 662 |
| 9 | 699 | A | 1170 | B | 502 |
| C | 811 | D | 578 | E | 1170 |
| F | 705 | G | 248 | H | 194 |
| I | 544 | J | 111 | K | 23 |
| L | 475 | M | 504 | N | 704 |
| O | 485 | P | 915 | Q | 21 |
| R | 1110 | S | 787 | T | 989 |
| U | 324 | V | 407 | W | 7 |
| X | 85 | Y | 35 | Z | 25 |
| a | 876 | b | 231 | c | 340 |
| d | 544 | e | 1967 | f | 103 |
| g | 97 | h | 117 | i | 428 |

| | | | | | |
|---|---|---|---|---|---|
| **j** | 26 | **k** | 156 | **l** | **397** |
| **m** | 330 | **n** | 708 | **o** | **757** |
| **p** | 193 | **q** | 10 | **r** | **678** |
| **s** | 644 | **t** | 655 | **u** | **674** |
| **v** | 123 | **w** | 8 | **x** | **916** |
| **y** | 27 | **z** | 27 | **@** | **897** |

**Tableau 2: Some character samples**

| Class 0 |  |  |  |  |  |
|---|---|---|---|---|---|
| Class A | | | | | |
| Class D | | | | | |
| Class F | | | | | |
| Class 9 | | | | | |

The distribution of the classes it's not uniform as you can see. Because these characters are coming from the EDF's maps, we will find characters which are not so well represented, like q, w, z, y, Q, W, X, Y, Z as some others like 0, 1, 2, 3, A, B, C, n, o, s, u, etc.

This distribution reflect the frequency of the characters in the French language.

I will point out to some important values concerning the database.

We have:

- **57** classes where the number of samples is more than 25
- **52** classes where the number of samples is more than 50
- **50** classes where the number of samples is more than 100
- **43** classes where the number of samples is more than 200
- **41** classes where the number of samples is more than 300

- **33** classes where the number of samples is more than 500
- **17** classes where the number of samples is more than 750
- **9** classes where the number of samples is more than 1000

## 3. Data normalization

Because the images from EDF are multi-size and multi-oriented we are using the transformation of Goshtasby which is size and rotation invariant transformation and it is using shape matrices.

The main idea of this transformation is to bound our object in circle with minimal radius and after to draw some other circles with radius less than the first one. The centre of these circles is the gravity centre of the object.

After the drawing of these concentrical circles, we are drawing diameters in order to have intersections with the circles mentioned above. With these intersection points, we can create a shape matrix which is invariant to the rotation and scale. In order to keep many as possible local information from the object, we modified this transformation and the resulting image is not binary, but has multiple levels. The levels of each image is falling into 0..9.

***The Goshtasby's algorithm***:

For i:=$\overline{0, n-1}$ do

    For j:=$\overline{0, m-1}$ do

        If the P($\dfrac{iL}{n-1}, j\dfrac{360}{m}$) point given by the polar coordinates is in the object then

        M(i,j):=1

        Else

        M(i,j):=0;

    End For

End For

Here we presenting an example of the transformation, the image and the shape matrix obtained with the Goshtasby's transformation.

We will present also the modified Goshtasby algorithm, used by us, in order to stock as many as possible local information.
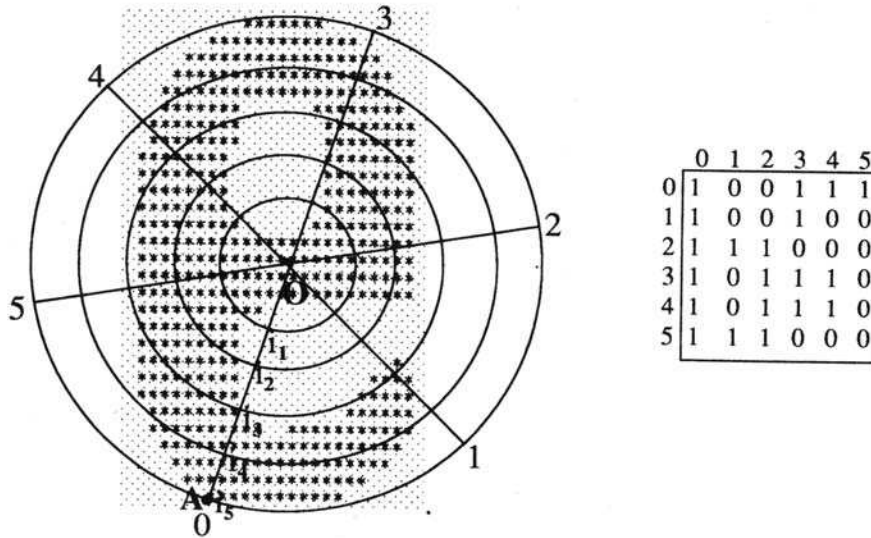
***The modified Goshtasby's algorithm***:

For i:= $\overline{0, n-1}$ do

      For j:= $\overline{0, m-1}$ do

          If the P($\dfrac{iL}{n-1}, j\dfrac{360}{m}$) point given by the polar coordinates is in the object

then

          M(i,j):=NeighboursNumber(i, j);
          Else
          M(i,j):=0;
      End For
End For

By *NeighboursNumber(i, j)* is denoted a method (function) which is returning in the original map the number of neighbours including also the pixel itself, of the pixel with the coordinates (i, j).

Goshtasby is proposing a method to compare the similarity between two shape matrices. The algorithm proposed is:

***Goshtasby's similarity algorithm***:

S=0

For I=0 to m-1

      For j=0 to n-1

          S=S+M1(I,j) xor M2(I,j)

Similarity=1- $\dfrac{S}{(m-2)n}$

Where M1 and M2 are shapes which we want to compare.

In the literature we can find  proposals for more efficient and strong formules, in order to calculate the similarity (distance) between shape matrices.

As we can see from the nature of the algorithm, in certain cases , for different characters (objects) the planar shapes will be the same.

Empirically we can prove the geometrical invariance features (size and rotation) of this transformation by using the notion of similarity.

The similarity of the character classes confirms the algorithm, presented above, and unfortunately the "distance" between some characters classes is very small. These classes are: (0, o, O, C, D, Q), (v, V, A), (I, i, l, 1), (z, Z), (s, S), (8, B), (6, 9, p, b). These confusions are the side-effects of this transformation.

By having this precognition, in the first period, we should treat this character classes in the same manner by that we mean to label as one this classes. In the after treatment part, we should use another neural network(s) to separate these classes, which were treated in the same manner. Such kind of combined scheme is presented in [6] where is proposed a general purpose classifier and after that for each object class is a class specific verifier. This combined method was used successfully for numbers recognition.

We also tried the so-called Ring Projection transformation which is similar to the transformation presented above but the results are not good, so we gave out this solution.

The main idea of this transformation used for Hangul characters is to make a features vector from the planar shape, by using a cumulative summation of the different columns of the shape matrix of the given object.

## 4. Remarks concerning the database and the data normalization

**Character Representation**

As mentioned in **Tableau 1: Representation of the characters in the database** , the character representation is not uniform. Some characters are less represented like "W", "z", "y", etc. whereas others are highly represented. These two kind of aspects are negative for pattern recognition, because in the first case, we don't have enough different aspects of the characters. In the second case, we cannot know which samples are the more representative samples

**Character Shapes**

- First, the characters are multi-oriented and multi-sized leading to some confusing between characters such as : (6, 9), (V, A), (u, n), (b, d) for the rotation invariant property and (v, V), (s, S, 8), (u, U), (v, V), (x, X), (p, P), (0, o, O, D) for the size invariant property.
- Second, the quality of the different samples is very low. Some parts are absent and the strokes are cut. These drawbacks may lead to more confusions and character recognition failures.