Force Histogram Calculation: Case of 2D Vector Data
Version 3.0, June 2010

Pascal Matsakis (pmatsaki@uoguelph.ca), Daniel Recoskie (drecoski@uoguelph.ca)
University of Guelph, Ontario, Canada

Let us know if you run into any bugs. Please do not pass this software package to others outside your research team—ask them to contact us and we will be happy to send them the latest version of the package. Please cite [1] and [2] in any work that uses this package. For a recent survey on force histograms and related concepts, see [3].

[1] P. Matsakis, *Relations spatiales structurelles et interprétation d'images*, PhD. Thesis, Institut de Recherche en Informatique de Toulouse, France, 1998.

[2] P. Matsakis, L. Wendling, "A new way to represent the relative position of areal objects", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 21(7):634-643, 1999.

[3] P. Matsakis, L. Wendling, J. Ni, "A General Approach to the Fuzzy Modeling of Spatial Relationships", in: R. Jeansoulin, O. Papini, H. Prade, S. Schockaert (Eds.), *Methods for Handling Imperfect Spatial Information*, Springer-Verlag Publications, in press.

Purpose

This software package allows you to calculate force histogram values $\varphi_r^{AB}(\theta)$, for any real number r, any argument object A, any referent object B, and any direction θ. Here, however, an *object* is a fuzzy subset of the 2D Euclidean space; it is assumed that the number of membership degrees is finite and that any α-cut with $0 < \alpha \leq 1$ can be expressed—using the union and difference set operations—in terms of a finite number of distinct simple polygons P_1^α, P_2^α, …; these polygons P_1^α, P_2^α, … should be such that an edge of a polygon P_i^α does not intersect an edge of another polygon P_k^α. Each object is described in a text file (see Appendix).

Included Files

Besides this README.pdf and the file FHistogramVector.c (which includes its own "readme"), fourteen other files are included in the package. They provide examples of input objects and output force histograms. Six objects are considered: the object A is crisp, connected, with no holes; B is crisp and has two connected components with no holes; C is crisp, connected, with a hole; D has three connected components, and one of these components is contained in the hole of another component; E is a fuzzy (noncrisp) object with no holes; F is a fuzzy (noncrisp) object with a hole. The text file A-B.txt, for instance, lists the force histogram values $\varphi_1^{AB}(0)$, $\varphi_1^{AB}(\pi/2)$, $\varphi_1^{AB}(\pi)$, $\varphi_1^{AB}(3\pi/2)$ and $\varphi_1^{AB}(2\pi)$ associated with the objects A and B described in the text files A.vdata and B.vdata. These objects and histogram are shown in examples.pdf.

Compilation and Runtime Examples

Compilation under Linux is as follows:

```
>gcc FHistogramVector.c -lm
```

Here is a way to generate A-B.txt:

```
>./a.out *.vdata

Enter arguments.
You are supposed to know the types and domains.

Histogram will be stored in file ('quit' to exit): A-B.txt
Enter 1 for single sum scheme, 2 for double sum scheme: 1
Number of directions to be considered is: 4
Type of force is: 1.0
Argument object is defined by the text file: A.vdata
Referent object is defined by the text file: B.vdata

Histogram will be stored in file ('quit' to exit): quit
>
```

Here is a way to generate A-B.txt, A-C.txt, C-C.txt, B-D.txt, C-D.txt and E-F.txt:

```
>./a.out <input *.vdata
```

Note that for crisp objects, the single and double sum schemes produce the same results.

Appendix

Each object is described in a text file, which contains different integer and floating point values separated by space or line feed characters (' ' or '\n'). The object is described as a set of α-cuts (sorted by increasing α), each α-cut is described as a set of polygons P_1^α, P_2^α, … (in any order), each polygon as a set of vertices (listed either clockwise or counterclockwise), and each vertex as a pair of coordinates x (from left to right) and y (from bottom to top).

$m\ \ \mu_1\ \ \mu_2\ \ \ldots\ \ \mu_m$
n_1
$n_{1,1}$
$x_{1,1,1}\ \ y_{1,1,1}\ \ x_{1,1,2}\ \ y_{1,1,2}\ \ \ldots\ \ x_{1,1,n_{1,1}}\ \ y_{1,1n_{1,1}}$
$n_{1,2}$
$x_{1,2,1}\ \ y_{1,2,1}\ \ x_{1,2,2}\ \ y_{1,2,2}\ \ \ldots\ \ x_{1,2,n_{1,2}}\ \ y_{1,2,n_{1,2}}$
…
n_2
$n_{2,1}$
$x_{2,1,1}\ \ y_{2,1,1}\ \ x_{2,1,2}\ \ y_{2,1,2}\ \ \ldots\ \ x_{2,1,n_{2,1}}\ \ y_{2,1,n_{2,1}}$
$n_{2,2}$
$x_{2,2,1}\ \ y_{2,2,1}\ \ x_{2,2,2}\ \ y_{2,2,2}\ \ \ldots\ \ x_{2,2,n_{2,2}}\ \ y_{2,2,n_{2,2}}$
…
…

- ☐ description of the μ_1-cut of the object:
  - ◆ description of the 1st polygon that defines the μ_1-cut of the object
  - ◆ description of the 2nd polygon that defines the μ_1-cut of the object
  - ◆ …
- ☐ description of the μ_2-cut of the object:
  - ◆ description of the 1st polygon that defines the μ_2-cut of the object
  - ◆ description of the 2nd polygon that defines the μ_2-cut of the object
  - ◆ …
- ☐ …

int m = number of distinct nonzero membership degrees
double $\mu_1, \mu_2, \ldots \mu_m$ = nonzero membership degrees in increasing order
int n_i = total number of the vertices that define the μ_i-cut of the object ($n_i = \Sigma_j\, n_{i,j}$)
int $n_{i,j}$ = number of vertices of the j^{th} polygon that defines the μ_i-cut of the object
double $x_{i,j,k}$, $y_{i,j,k}$ = coordinates of the k^{th} vertex of the j^{th} polygon that defines the object's μ_i-cut