Assignment: "Images et géométrie discrète" 2015

Introduction

The objective of this project is to design a binary shape indexing/retrieval system.

We expect from you:

- A short report with a complete description of the your implementations choices (see below)
- A C++ project (`CMakeLists.txt` or `Makefile` plus couple of **commented** cpp program files or scripts).

Description

Let us consider that we have solved the segmentation problem and that we have a database of binary shapes. The challenge is to design shape indexing and shape retrieval processes for such database. More precisely, we consider a database of 70 classes with approx. 20 shapes in each class. For instance, there are 20 different hammers in the `hammer` class.

The problem consists in the following process:

- Shape indexing: for each shape, we create a signature vector $\vec{v} \in \mathbb{R}^d$ of d scalar quantities describing the shape. The signature should characterize the shape geometry and could distinguish classes: signatures of shapes in the same class should be *close* and signatures of two shapes in different classes should be *different*.
- Shape retrieval: given a input shape, we compute its feature vector $\vec{w}$ and we output for instance the k *nearest* shapes for a given similarity metric, or *distance*, on signature vectors. We could also output the class to which the input shape should belong to (classification problem). We could also return a probability measure for a shape to belong in a given class.

As you may have seen, requirements and specifications are imprecise. This reflects the fact that many solutions exist and we would like you to explore your own. Hence, the shape descriptor could be based either on contour information (length, curvature,...), or on volumetric information (geometrical moments,...). Similarly, the *distance* between shape signature could be a simple l_2 norm of the vectors or even better statistical similarity.

Whatever the choices you consider to design your indexing/retrieval system, we would like you to have a complete theoretical and experimental evaluation of your descriptor/system. For example, you must clearly discuss about the following points:

- Can I demonstrate that my descriptor is invariant under translation, scaling, rotation, shear, partial occlusion... ?
- Can I demonstrate (or experimentally check) the robustness of the descriptor in presence of noise ?
- How can I evaluate the retrieval system (accuracy of the classification, precision, recall...) ?
- ...

Evaluation

As you may have noticed, many solutions exist. To evaluate your work, the following points will be considered:

- The relevance of the discussion on the proposed descriptor;
- The quality of the experimental evaluation;
- The quality of the pros/cons discussion.

Technically, we will also evaluate:

- The quality of the code;
- The fact that we can reproduce the results (e.g. with shell scripts);
- Finally, the quality of the indexing/retrieval system.

Details

The image database is available in the course github project (Part of the MPEG-7 CE-Shape-1 Dataset, `assignment/ShapeIndexing/database`).

In `DGtalSkel`, you will find an example (`PGMReader.cpp`) to load a PGM file with DGtal tools and to convert pixels with intensity greater than 0 into a digital set. `imgAddNoise` tool can be used to generate a noisy version of a binary image using a Kanungo noise model (pixel value swapping procedure from a power law on the distance to the boundary, see DGtal documentation).

The final projects archive file (named after the *binome* authors) must contain

- A report (PDF, 5-10 pages)
- Source files with either a `makefile` or a `CMakeLists.txt`
- Scripts/executable to evaluate the system (for example, taking an input image file and returning its class or the closest images).