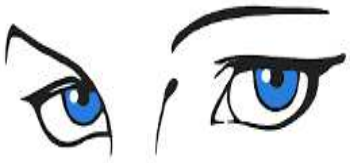
PROFI



Project number:	FP6-511572
Project acronym:	PROFI
Title:	Perceptually-relevant Retrieval Of Figurative Images

Deliverable No: D6.1: Simulation tool software

Short description:
The choice of the number and of the actual vantage objects is closely related to the occurrences of clusters in feature space of the shape elements. In order to experimentally investigate the effect, we need a simulation tool to generate distance values. We will start with randomly generated distance values, since these are easier to describe, analyse, and test than real data. However, real image data differs in a number of ways from such randomly generated test data. In particular, the distribution of image and model features is not uniform, correlations are present, and additional information is available about feature properties and groupings. Before we have available all the real data, which are the result of the other workpackages about segmentation and distance computation (matching) we need to be able to experiment with different possible distributions, and see the effect on the index structure.

Due month:	M9
Delivery month:	M9
Lead partner:	Utrecht University
Partners contributed:	Utrecht University
Classification:	RE

Information Society Technologies

Project funded by the European Community under the "Information Society Technologies" Programme

1 Results obtained

1.1 Objectives

PROFI work package 6 embodies Shape Indexing. Its three objectives are in order of delivery date: the development of simulation tool software (6.1), the development of strategies for selecting vantage objects (6.2) and investigating how partial matching can be best supported by the vantage indexing method (6.3). This document reports the progress made on 6.1: the simulation tool software.

One of the indexing techniques which are investigated within the PROFI project is called Vantage Indexing, due to Vleugels and Veltkamp [1]. This technique reduces image similarity queries on a database containing n images to a nearest neighbor or range query in a k-dimensional space, where $k \ll n$. This result is achieved by applying the following strategy: pick a small number k representative objects from the database; these are called the vantage objects. Calculate the distances (dissimilarities) between the query object and the vantage objects. Interpret every object in the database (and the new object as well) as a k-dimensional vector in Euclidean space: each vector represents the dissimilarity of the object to the set of vantage objects. Select only these objects from the database, whose vector of dissimilarities to the vantage objects have a Euclidean distance to the vector of the new image within a range of ϵ. A query now only needs k dissimilarity calculations. One of the most important research questions within work package 6 is how to chose a proper set of vantage objects.

In order to be able to experiment with different techniques for selection of vantage objects, as well as to compare vantage indexing to other indexing techniques, data sets are necessary. Experimentation will take place both on real world data sets and within a controlled environment, i.e. on synthetic data. To obtain these synthetic data sets, a simulation tool will be developed to generate these data sets. The development of this simulation tool is the objective of deliverable 6.1.

1.2 Results

A simulation tool for generating synthetic data sets has been developed. In this section, the functionality of this tool will be described.

1.2.1 Nature of the data

During the PROFI project, a distance measure is developed to compute the distance between two images. This distance measure is defined on the images themselves, i.e. in *object space*. Because data sets are large and the similarity measure is complex, it is computationally infeasible in practice to calculate the distance between every pair of objects present in the data set. However, theoretically a $n \times n$ dissimilarity matrix could be computed with on position (i, j) the distance between images i and j. In real applications the developed distance measure and the database of images together form the input for the indexing stage.

In order to simulate this input, the simulation tool generates a dissimilarity matrix, as if the n^2 distance calculations were actually made.

There are two possible approaches for generating the required distance data, with both their drawbacks and advantages. The first approach fits the course of the PROFI project most naturally: the direct simulation of a dissimilarity space. The second approach reflects

the ideas of more traditional pattern recognition: the simulation of feature vectors. Both approaches are integrated in the simulation tool.

1.2.2 Simulating a dissimilarity space

During the PROFI project distance measures will be developed to compute the distance between two images. Together with the objects, this distance measure forms the input for the indexing stage. So all that is known, are the distances (calculating on demand) between the objects. No spatial information is present. To simulate this type of input, the simulation tool generates a dissimilarity matrix without the notion of an underlying vector space.

Many questions arise when generating the values for this matrix. Does the set of the distances obey the triangle inequality, or maybe a relaxed instance of the triangle inequality? And does it obey the other metric properties: positivity, self-identity and symmetry? What is the distribution of the distances, and within what range are they chosen? All these questions translate to options the user of the simulation tool can adjust to the needs of the experiment the data will be used for.

The advantage of generating a dissimilarity matrix directly is that its intrinsic dimensionality is close to its representational dimensionality. It is unlikely that a $n \times n$ dissimilarity matrix, which can be seen as n points in n-dimensional space [1], can be embedded into a k-dimensional space where $k << n$ such that no relative distance information is lost. This kind of very high dimensional data is exactly the type of input we are dealing with in this project.

A drawback of this method however is that it is very difficult to be in control of a clustering of objects, since it is not even clear what a clustering exactly looks like in pure distance data where no spatial information is present. But in real data sets clusters do probably occur, therefore another simulation approach has been integrated in the tool: the simulation of a feature space.

1.2.3 Simulating a feature space

Whereas the simulation of a dissimilarity space resembles the PROFI project most naturally (the development of a distance measure for the images in object space), simulating a feature space represents the more traditional pattern recognition approach, namely the extraction of features prior to the distance computations which take place in a feature space.

For each object, a d-dimensional feature vector is generated, where d is user-defined. To obtain the actual distances, the Euclidean distance between these feature vectors is computed. When generating the feature vectors the user can again adjust a number of options to the needs of the experiment for which the data will be used. It is possible to choose from a number of statistical distributions for which the user can set the necessary parameters. In addition it is also possible to force the existence of a clustering of objects. This is done by placing a number of cluster centers in the feature space around which clusters are formed. The user is in control of both the way these cluster centers are placed and the type of distribution of the clusters around the centers.

While the possibility to simulate clustered spaces is an advantage of this method over simulating a dissimilarity space, there is also a drawback of this method. The dimensionality

[1] However, please note that since the n objects present in the database are selected from the infinite universe $\mathbb{U}$, in theory the dimensionality is infinite.

should be high enough, otherwise we can select $d + 1$ vantage objects and the embedding will always be without information loss. But when both d is high and the range of the d components is large, due to the curse of dimensionality, we have to generate a large amount of points to prevent the space from being sparsely filled. This is important to keep in mind while generating a distance matrix this way.

1.2.4 Simulation to support analysis

The algorithm for selecting vantage objects (work package 6.2) will be a randomized incremental construction algorithm. During the execution of this algorithm, database objects will be added to the index one by one in random order. Each time a new object has been added, the index will be checked on a number of important properties. When the index does not have the satisfactory properties anymore, repair steps are necessary.

To find the expected running time of a randomized incremental construction algorithm, one must know the probability that in a certain iteration these properties are not satisfactory anymore. This depends of course on the data set and the distance measure. However, for the purpose of run time analysis, we will assume various distributions (e.g. Gaussian, exponential), and find these probabilities. For some distributions of data, the derived probability on unsatisfactory index properties cannot be determined analytically. In order to be able to do the run time analysis, we will determine the probabilities experimentally, using the simulation tool.

1.2.5 The simulation tool

The simulation tool itself is called *MISE en scène* (Mappings In Synthetic Environments) and consists of two parts: the actual core components which are written in C++, and a graphical user interface on top of these components which is written in Java. The use of this graphical user interface is optional, since the simulation core components can be executed from a command line as well, for which convenient help messages have been generated.

New features can be added easily in the future, when experiments should demand further functionality of the simulation tool, because an effort has been made to keep the software maintainable and extendable. In Figure 1 some screen shots of the graphical user interface are displayed to give an impression of the system.

2 Deviations from plan

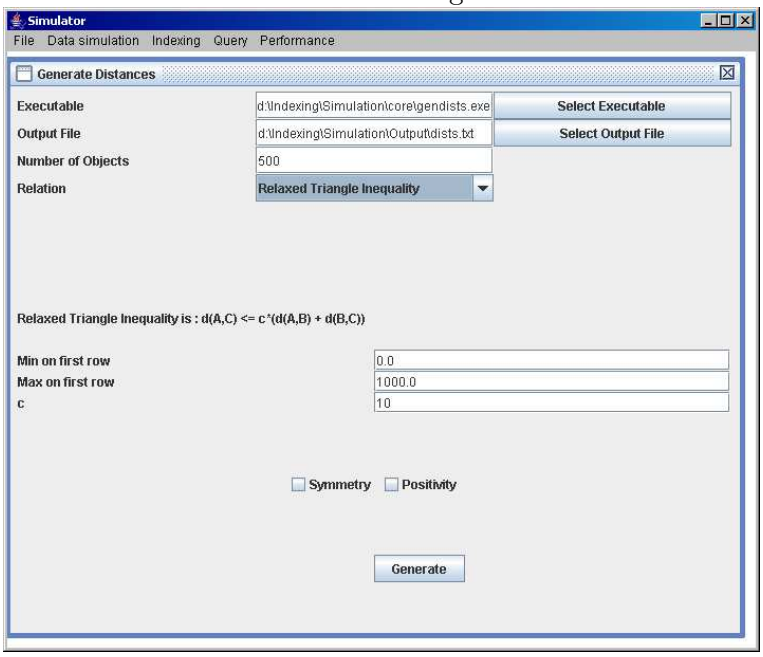
There have been no deviations from plan.

3 Appendix: *MISE en scène* User's Manual.

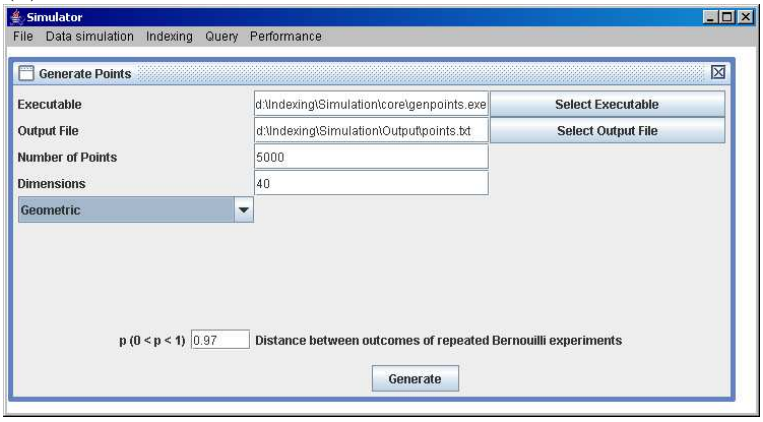
3.1 Simulating a dissimilarity space

A dissimilarity space can be simulated by the module gendists. This module generates a distance matrix directly, without an underlying vector space or coordinate system. Below are a number of customizable properties for this dissimilarity matrix.

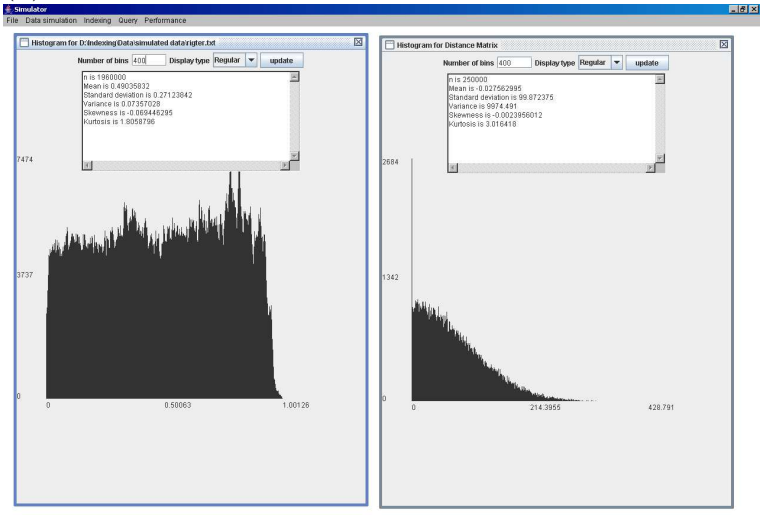
Figure 1: GUI Screenshots



(a) generating a dissimilarity space



(b) generating a feature space



(c) displaying properties of distance matrices

- Distances obey strict triangle inequality. $d(A,C) <= d(A,B) + d(B,C)$ for each triplet of objects present in the simulated dissimilarity space. In this case, the user must specify the minimum and maximum of the values in the first row.
- Distances obey relaxed triagle inequality. $d(A,C) <= c \times (d(A,B + d(B,C))$ for each triplet of objects present in the simulated dissimilarity space. In this case, the user must specify the minimum and maximum of the values in the frist row, and a value for c.
- Distances do not obey any form of the triangle inequality, they are chosen completely at random. Possible distributions are:
  1. Normal distribution. User must specify mu and sigma
  2. Uniform distribution. User must specify min and max
  3. Geometric distribution. User must specify p.
  4. Exponential distribution. User must specify lambda.

Dump of help file:

```
Tool for generating n x n distance matrices.
Created by Reinier van Leuken, last revision august 11th 2005
----------usage:----------
All required options:

Required general options:
  --help                : produce help message
  --out arg             : define outputfile
  --npoints arg         : define number of points
  --rel arg             : define relation
                              s for strict triangle inequality,
                              l for relaxed triangle inequality
                              d for random
  --sym                 : Symmetry?
  --pos                 : Positivity?

Required options for strict triangle inequality:
  --min arg             : define min on first row
  --max arg             : define max on first row

Required options for relaxed triangle inequality:
  --min arg             : define min on first row
  --max arg             : define max on first row
  --c arg               : define multiplicative relaxation

Required options for random distances:
  --distr arg           : define distribution
                               n for normal
                               u for uniform
```

```
                                g for geometric
                                e for exponential

Required specific options for normal distribution:
  --mu arg                : define mu
  --sigma arg             : define sigma

Required specific options for uniform distribution:
  --min arg               : define min
  --max arg               : define max

Required specific options for geometric distribution:
  --p arg                 : define p

Required specific options for exponential distribution:
  --l arg                 : define lambda
```

3.2 Simulating a vector space

To reflect the ideas of more traditional pattern recognition, the possibility of simulating a vector space has been provided. With the module genpoints, a number of points d-dimensional points can be generated as if they were feature vectors. With the module calcdists, the distance between these feature vectors can be computed. For the simulation of the vector space, the user can choose from a number of distributions and must specify some parameters accordingly. Below are dumps of the help files.

Module genpoints:

```
Tool for simulating a feature vector space.
Created by Reinier van Leuken, last revision august 11th 2005
----------usage:----------
All required options:

Required general options:
  --help                  : produce help message
  --out arg               : define outputfile
  --npoints arg           : define number of points
  --dim arg               : define dimensionality
  --distr arg             : define distribution
                                  n for normal
                                  u for uniform
                                  g for geometric
                                  e for exponential
                                  c for clustering

Required specific options for normal distribution:
  --mu arg                : define mu
  --sigma arg             : define sigma
```

```
Required specific options for uniform distribution:
  --min arg              : define min
  --max arg              : define max

Required specific options for clustering:
  --nc arg               : number of clusters (let nc be divider of npoints)
  --cmin arg             : define min for centers
  --cmax arg             : define max for centers
  --csigma arg           : define sigma for clusters
```

Module calcdists:

```
Tool for calculating the distances between feature vectors.
Created by Reinier van Leuken, last revision august 11th 2005
----------usage:----------
Allowed options:
  --help                 : produce help message
  --out arg              : define outputfile
  --in arg               : define inputfile
  --measure arg          : define distance measure
  --p arg                : define p in case of LP-metric
```

References

[1] J. Vleugels and R. C. Veltkamp, "Efficient image retrieval through vantage objects," in *Visual Information and Information Systems*, 1999, pp. 575–584.