# Técnicas de Reconhecimento de Padrões
# Pattern Recognition Techniques

2009/2010

*Project Assignment*

## Audio Steganalysis in Computer Forensics
## *SAFE-Audio*: Stego Analysis For Evaluation of Audio


**(a) Cover image carrier and (b) Stego with hidden information (in [1])**

"Steganography is the art of covered or hidden writing. The purpose of steganography is covert communication to hide a message from a third party. This differs from cryptography, the art of secret writing, which is intended to make a message unreadable by a third party but does not hide the existence of the secret communication. Many common digital steganography techniques employ graphical images or audio files as the carrier medium. The art and science of steganalysis is intended to detect or estimate hidden information." in http://www.fbi.gov/hq/lab/fsc/backissu/july2004/research/2004_03_research01.htm

## Introduction

Steganography is the art of hiding information in digital content in the internet. Stego messages can be hidden inside all sorts of cover information: text, images, audio, video and more. Most currently used steganographic techniques hide information inside images, as this is relatively easy to implement. The original images are called *cover* and the result of joining the cover and the message to hide is called *stego*. One of the most important properties of a cover source is the amount of data that can be stored inside it without being perceptible. When an image is distorted, the cover source will be suspicious and may be checked more meticulously. An undetected image with a secret message inside can easily be spread over the world wide web or in newsgroups.

When the carrier is audio, one of the most popular digital audio formats, MP3 encoding, provides a high compression ratio with a faithful quality, which enables MP3 audio to become an excellent carrier for steganography. Few works been reported on successful steganalysis of the information-hiding behavior in MP3 audios.

In [2] feature extraction of MP3 audio samples are performed using different methods (i) measuring distortion by extracting the distribution parameters of generalized Gaussian density from individual frames (ii) design moment statistical features on the second derivatives, as well as (iii) Markov transition features and neighboring joint density of the MDCT coefficients based on each specific frequency band on MPEG-1 Audio Layer III.

The other common carrier is WAV digital data files which can also embed hidden messages. In [3] feature extraction is performed and audio steganograms created by several ways (indicated below).

## Stego Recognition: Problem Statement

Your task is to develop a **stego detector** for a computer forensics system "SAFE-Audio" (nonexistent), which allows to detect stego secret messages in innocent digital media, such as audios in two formats MP3 and WAV digital audio files. To achieve this goal, the original medium (cover)has been imperceptibly modified to embed encrypted messages by using a shared key, and the receiver can extract and decrypt messages from the modified carriers (steganogram). In audio steganography, data can be hidden in several ways. The classes of the dataset are:

1. cover
2. stego

Your task is to implement a Matlab function `C = classify(testdata)`, which classifies a single unknown example given as a matrix testdata. The output class $C$ should correspond to the classes shown above.

## Practical Assignment

### Downloading STEGO Audio Data Base

The Audio data base is available in two formats MP3 files and WAV signal audio files.

1. The MP3 steganalysis dataset.
   The dataset contains 1000 mono MP3 cover audio with the bit rate of 128 kbps and the sample rate of 44 KHz. Each audio has the duration of 18 seconds. 1000 stego-audio were sampled by hiding random messages with embedded data (modification 20%). The classes, denoted by 1 and 2 as label, are shown in the last column of CSV file.

   The feature extraction methods are explained in the paper "Feature Mining and Intelligent Computing for MP3 Steganalysis" [2].

   (a) Features Description:

   #1-81: inter-frame Makov features

   #2-162: inter-frame joint density features

   #163-738: 2nd derivative spectrum analysis feature for 576 sub-bands

   #739-742: moment statistical features of shape parameter (beta)

   (b) Download from http://eden.dei.uc.pt/~bribeiro/stego_audio_MP3.rar

2. The WAV files dataset.
   The data set contains 6000 mono 44.1-kHz 16-bit quantization in uncompressed, PCM coded WAV audio signal files, covering different types such as digital speech, on-line broadcast in different languages, for instance, English, Chinese, Japanese, Korean, and Spanish, and music (jazz, rock, blues). Each audio has the duration of 19 s. The same amount of the stego-audio signals by hiding different message in these audio signals has been produced.

   The feature extraction methods are explained in the paper "Temporal Derivative-Based Spectrum and Mel-Cepstrum Audio Steganalysis" [3].

   (a) Features Description:
   The second derivative based mel-cepstrum (D-MD) features extracted from WAV audio covers and several types of WAV steganography audio files are included in the dataset.

   The hiding tools/algorithms include (i) Hide4PGP V4.0, (ii) Invisible Secrets, (iii) LSB matching, and (iv) Steghide. The hidden data include voice, video, image, text, and executable codes, which were encrypted before embedding by using different keys. Audio steganograms were also produced by hiding random bits. The covert data in any two audio streams are different. [1]

   (b) Download from http://eden.dei.uc.pt/~bribeiro/stego_audio_WAV.rar

## Building Train and Test Data Sets

In your Project you should be able to construct the `Train data set` and to build the learning control system for the "*SAFE - Audio* " as well as to construct the `Test data set` in order to test your model.

### Experimental Setup

You should be able to design your experiences in order to run the pattern recognition algorithms in the given data. Define the performance metrics to evaluate your method. Run the experiments and present average results and standard deviation.

## Pattern Recognition Methods

You can write your own code or use basic algorithms available in the Statistical Pattern Recognition Tool STPRTool used in classes(since you are already familiarized with it). The methods used in your work should be described as well as discussion of used parameters. You should justify your decisions in the Project too. For example, the number of features used and why (see documentation below).

## Results and Discussion

Present and discuss final results obtained in your Project assignment.

## Requirements

Practical assignment is meant to be done in groups of two persons. If someone wants to work alone, this is also possible. Larger groups than group of two persons are not allowed.

[1] All the covers and steganograms are available at http://www.cs.nmt.edu/~IA/steganalysis.html

### Matlab function

The Matlab function that performs classification must have name *classify*. Also, if you are using classification technique which needs some extra parameter(s) (eg $K$ in `K-nearest-neighbors` classifier), this parameter(s) must be fixed or determined inside recognize function.
Remember to comment your codes. Write also a help section to your codes that tells the purpose of the function, usage, and explanation of parameters. In Matlab, comments following the first line of a function will show when help command is used with the name of the function.

## Documentation

Write paper documentation (in Portuguese or in English) about your project. The documentation should include a cover page where course name, project title, date, names and student numbers of the authors are mentioned.

Describe the methods used for classification in such detail that reader would be able to implement the same kind of functions for feature extraction and classification just based on your documentation and some basic background in pattern recognition. Justify your choices, i.e., justify reasons to end up to classifier (and feature extraction technique) you were using. Include classification results with given data to you documentation. Use data training and testing sets, and perform classification. At the end of your documentation you should have a list of all references used.

### Project Deadlines

Submission: The deadline of this task is December 18th, 2009.
Presentation and Discussion: 12th January 2010.

## Grading

The classifiers will be graded according to their classification performance, i.e., the smaller classification error the better classifier. Classification result (*classification error*) will be calculated in a data set specially designed for testing. Operation of a classifier must be reasonable, e.g., a classifier that returns randomly selected class labels is not acceptable and will lead to rejection of practical assignment. Also, your classifier should perform better than purely random classifier (which is maybe based on a priori probabilities).

## Acknowledgments

Authors from references [3] (Dr Qingzhong Liu) and [2] (MSc Mengyu Qiao) as well as Prof A H Sung from New Mexico Tech, USA, are gratefully acknowledged for providing the Datasets.

## References

[1] Ferreira, R., Ribeiro, B., Silva, C., Liu, Q.,Sung, A.H.,: Building Resilient Classifiers for LSB Matching Steganography, Proc. of Int Joint Conf on Neural Networks, (2008)

[2] Qiao, M., Sung, A.H., Liu, Q.: Feature Mining and Intelligent Computing for MP3 Steganalysis, Proc. of International Joint Conference on Bioinformatics, Systems Biology and Intelligent Computing (2009)

[3] Liu, Q., Sung, A.H.,Qiao, M. : Temporal Derivative-Based Spectrum and Mel-Cepstrum Audio Steganalysis, IEEE Transactions on Information Security **4**(3) (2009) 359–368

# Técnicas de Reconhecimento de Padrões
# Pattern Recognition Techniques

2008/2009

*Project Assignment*

## Gesture Recognition: "*Gladius* Master"

"A child being sensed by a simple gesture recognition algorithm detecting hand location and movement."
http://en.wikipedia.org/wiki/Gesture_recognition

### Introduction

"Gesture Recognition can be seen as a way for computers to begin to understand human body language, thus building a richer bridge between machines and humans than primitive text user interfaces or even GUIs (Graphical User Interfaces), which still limit the majority of input to keyboard and mouse.

Gesture Recognition enables humans to interface with the machine (HMI) and interact naturally without any mechanical devices. Using the concept of Gesture Recognition, it is possible to point a finger at the computer screen so that the cursor will move accordingly. This could potentially make conventional input devices such as mouse, keyboards and even touch-screens redundant."
Source: http://en.wikipedia.org/wiki/Gesture_recognition

"A primary goal of gesture recognition research is to create a system which can identify specific human gestures and use them to convey information or for device control. How is information encoded in gestures? How humans use gestures to communicate with and command other people."
Source: http://homepages.inf.ed.ac.uk/rbf/CVonline/...
...LOCAL_COPIES/COHEN/gesture_overview.html

### Gesture Recognition: Problem Statement

Your task is to develop a **learning control system** for a computer game "Gladius Master" (nonexistent), which allows the player to control swordfights by swinging around a sensor equipped with accelerometers. Seven people have demonstrated their performance on seven different moves, and the developed system should be able to

classify new demonstrations as one of the demonstrated moves. The information from the demonstrations is available as `wiicontroltrn.tar.gz`, which will contain several files named `pXmYdZ.mat`, where `X` is the number of the demonstrator, Y is the class (type of move), and `Z` indexes different demonstrations. Thus, each file will contain a single demonstration, with three values on each line, the values corresponding to accelerometer measurements at the corresponding time instant. Each demonstration thus contains measurements over the whole motion. The moves (`Y`) are as follows:

1. slash left
2. slash right
3. cross strike
4. parry & strike
5. parry up
6. overhead strike
7. stab

Your task is to implement a Matlab function `C = classify(testdata)`, which classifies a single unknown example given as a `N * 3` matrix testdata. The output class $C$ should correspond to the moves as shown above.

## Practical Assignment

### Downloading wiicontrol Data Base

Download from http://eden.dei.uc.pt/~bribeiro/Gladi/wiicontroltrn.tar.gz.tar

### Building Train and Test Data Sets

In your Project you should be able to construct the `Train data set` and to build the learning control system for the "*Gladius* Master" as well as to construct the `Test data set` in order to test your model.

### Experimental Setup

You should be able to design your experiences in order to run the pattern recognition algorithms in the given data.

Define the performance metrics to evaluate your method or methods. Run the experiments and present average results and standard deviation. Compare with the Baseline method which can be considered as the one with all features and a simple PR algorithm such as `K- Nearest Neighbor`.

### Pattern Recognition Methods

You can write your own code or use basic algorithms available in the Statistical Pattern Recognition Tool STPRTool used in classes, since you are already familiarized with it. The methods used in your work should be described as well as discussion of used parameters. You should justify your decisions in the Project too. For example, the number of features used and why (see documentation below).

### Results and Discussion

Present and discuss final results obtained in your Project assignment.

## Requirements

Practical assignment is meant to be done in groups of two persons. If someone wants to work alone, this is also possible. Larger groups than group of two persons are not allowed.

## Matlab function

The Matlab function that performs classification must have name *classify*. The function(s) for feature extraction must be called from inside recognize function. Also, if you are using classification technique which needs some extra parameter(s) (eg $K$ in `K-nearest-neighbors` classifier), this parameter(s) must be fixed or determined inside recognize function.
Remember to comment your codes. Write also a help section to your codes that tells the purpose of the function, usage, and explanation of parameters. In Matlab, comments following the first line of a function will show when help command is used with the name of the function.

## Documentation

Write paper documentation (in Portuguese or in English) about your project. The documentation should include a cover page where course name, project title, date, names and student numbers of the authors are mentioned.

Describe the methods used for feature extraction and classification in such detail that reader would be able to implement same kind of functions for feature extraction and classification just based on your documentation and some basic background in pattern recognition. Giving an algorithm and explaining it in words is a good way to describe methods. Justify your choices, i.e., justify reasons to end up to classifier and feature extraction technique you were using. Include classification results with given data to you documentation. Use data training and testing sets, and perform classification. At the end of your documentation you should have a list of all references used.

## Deadline and submission

The deadline of this task is December 18th, 2008.

## Grading

The classifiers will be graded according to their classification performance, i.e., the smaller classification error the better classifier. Classification result (*classification error*) will be calculated in a data set specially designed for testing.

Operation of a classifier must be reasonable, e.g., a classifier that returns randomly selected class labels is not acceptable and will lead to rejection of practical assignment. Also, your classifier should perform better than purely random classifier (which is maybe based on a priori probabilities).

## Notes and tips

The data has been gathered using `WiiMote` of the `Nintendo Wii`. For more information, see e.g. `http://www.wiili.org/index.php/Wiimote`. Feature extraction will have a significant effect in the system. Try to either develop a good method yourself or look for more information in any source.

# Técnicas de Reconhecimento de Padrões
# Pattern Recognition Techniques

2007/2008

*Project Assignment*

**Faces Recognition**

"The face recognition problem is made difficult by the great variability in head rotation and tilt, lighting intensity and angle, facial expression, aging, etc. In particular, the correlation is very low between two pictures of the same person with two different head rotations."

Woody Bledsoe, 1966

## Introduction

A facial recognition system is a computer application for automatically identifying or verifying a person from a digital image or a video frame from a video source. One of the ways to do this is by comparing selected facial features from the image and a facial database.

Among the different biometric techniques facial recognition may not be the most reliable and efficient but its great advantage is that it does not require aid from the test subject. Properly designed systems installed in airports, multiplexes, and other public places can detect presence of criminals among the crowd. It is typically used in security systems and can be compared to other biometrics such as fingerprint or eye iris recognition systems.

## The Yale Data Base

The Yale database contains 165 gray scale images in GIF format of 15 individuals. There are 11 images per subject, one per different facial expression or configuration: center-light, w/glasses, happy, left-light, w/no glasses, normal, right-light, sad, sleepy, surprised, and wink. The images of 11 subjects that includes variation in both facial expression and lighting. Figure 1 shows

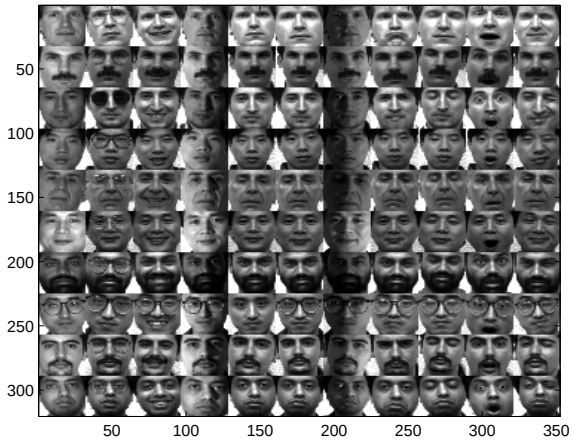



Figura 1: Sample Images from Yale Database

55 closely cropped images which include internal facial structures such as the eyebrow, eyes, nose, mouth and chin, but do not contain the facial contours.

## Practical Assignment

### Downloading Yale DataBase

Download from http://www.cs.uiuc.edu/homes/dengcai2/Data/FaceData.html the Yale DataBase

32x32 Data File: contains variables `fea` and `gnd`. Each row of `fea` is a face; `gnd` is the label.

### Building Data Training and Test

A random subset with p(=2,3,4,5,6,7,8) images per individual was taken with labels to form the training set, and the rest of the database was considered to be the testing set. For each given p, there are 50 randomly splits:

```
2 Train | 3 Train | 4 Train | 5 Train | 6 Train | 7 Train | 8 Train |
```

Each split file contains variables 'trainIdx' and 'testIdx'. The following matlab codes can be used to generate the training and test set:

```
%=========================================
fea_Train = fea(trainIdx,:);
fea_Test = fea(testIdx,:);
gnd_Train = gnd(trainIdx);
gnd_Test = gnd(testIdx);
%=========================================
```

### Experimental Setup

You should be able to design your experiences in order to run the pattern recognition algorithms for the 50 randomly split in each data set given. Define the performance metrics to evaluate your method or methods. Run the experiments and present average results and standard deviation. Compare with the Baseline method which can be considered as the one with all features in the images and a simple PR algorithm such as K- Nearest Neighbor.

See Table of results in the Yale Data Base web site.

### Pattern Recognition Methods

You can write your own code or use basic algorithms available in the Statistical Pattern Recognition Tool STPRTool used in classes, since you are already familiarized with it. The methods used in your work should be described as well as discussion of used parameters. You should justify your decisions in the Project too. For example, the number of features used and why (see documentation below).

### Results and Discussion

Present and discuss final results obtained in your Project assignment.

## Requirements

Practical assignment is meant to be done in groups of two persons. If someone wants to work alone, this is also possible. Larger groups than group of two persons are not allowed.

## Matlab function

The Matlab function that performs classification must have name *classify*. The function(s) for feature extraction must be called from inside recognize function. Also, if you are using classification technique which needs some extra parameter(s) (eg $k$ in $k - nearest - neighbors$ classifier), this parameter(s) must be fixed or determined inside recognize function.
Remember to comment your codes. Write also a help section to your codes that tells the purpose of the function, usage, and explanation of parameters. In Matlab, comments following the first line of a function will show when help command is used with the name of the function.

## Documentation

Write paper documentation (in Portuguese or in English) about your project. The documentation should include a cover page where course name, project title, date, names and student numbers of the authors are mentioned.

Describe the methods used for feature extraction and classification in such detail that reader would be able to implement same kind of functions for feature extraction and classification just based on your documentation and some basic background in pattern recognition. Giving an algorithm and explaining

it in words is a good way to describe methods. Justify your choices, i.e., justify reasons to end up to classifier and feature extraction technique you were using. Include classification results with given data to you documentation. Use data training and testing sets, and perform classification. At the end of your documentation you should have a list of all references used.

## Deadline and submission

The deadline of this task is January 14th, 2008.

## Grading

The classifiers will be graded according to their classification performance, i.e., the smaller classification error the better classifier. Classification result (*classification error*) will be calculated as an average of classification results for all the random splits given.

Operation of a classifier must be reasonable, e.g., a classifier that returns randomly selected class labels is not acceptable and will lead to rejection of practical assignment. Also, your classifier should perform better than purely random classifier (which is maybe based on a priori probabilities).

# Técnicas de Reconhecimento de Padrões
# Pattern Recognition Techniques

2006/2007

Project: *Assignment 1*

**Spam and Virus Filtering**

SPAM by Numbers

| | |
|---|---|
| 28 % | The amount of spam in the average users inbox |
| 30 billion | The number of spam messages delivered per day in 2006, up from 10 billion in 2002 |
| 12 billion | Lost productivity, downtime and unrecoverable data in the next 18 months |
| 80 hours | Lost productivity per employee over a year if they take 20 minutes per day to filter spam from legitimate business email |

Spam is nowadays an important issue in most email severs and information systems in general.

## Introduction

This practical assignment is to implement a Matlab function that classifies email messages in two classes: **Spam** and **Legitimate** messages. The function to implement has the form:
**S = Recognize (*trainfile,trainclass,testfile*)**
Input parameter *trainfile* is the name of an ASCII text file, which contains set of email messages. A row vector *trainclass* contains classes of training samples so that *ith* element of trainclass is the class of ith email message in *trainfile*. The class of spam messages is presented with a numeric value 1 and the class of ordinary messages is presented with a numeric value 0. ASCII text file *testfile* contains set of email messages to be classified. The return value $S$ is a row vector, which contains the classification result, i.e., numeric value 0 or 1 for each message in testfile. A text file for training *(trainfile.txt)* and a vector containing corresponding classes for messages *(trainclass.mat)* are given to help in design of the classifier. The text file contains 70 spam messages and 70 legitimate messages. A Matlab feature extractor function, *features = fe(filename,bins)*,

is provided. This function finds individual email messages in file filename and calculates a feature vector for each message. The feature vector is a hash value histogram of words in message. A hash value is calculated by summing ASCII values of the letters in a word and taking modulus from this sum. It is advisable that students develop alternative feature extractors using information from any source.

## Requirements

Practical assignment is meant to be done in groups of two persons. If someone wants to work alone, this is also possible. Larger groups than group of two persons are not allowed.

## Matlab function

The Matlab function that performs classification must have name *classify* and it must take and return parameters in the way described above. This means that function(s) for feature extraction must be called from inside recognize function. Also, if you are using classification technique which needs some extra parameter(s) (eg $k$ in $k - nearest - neighbors$ classifier), this parameter(s) must be fixed or determined inside recognize function.
Remember to comment your codes. Write also a help section to your codes that tells the purpose of the function, usage, and explanation of parameters. In Matlab, comments following the first line of a function will show when help command is used with the name of the function.

## Documentation

Write paper documentation (in Portuguese or in English) about your project. The documentation should include a cover page where course name, project title, date, names and student numbers of the authors are mentioned.

Describe the methods used for feature extraction and classification in such detail that reader would be able to implement same kind of functions for feature extraction and classification just based on your documentation and some basic background in pattern recognition. Giving an algorithm and explaining it in words is a good way to describe methods. Justify your choices, i.e., justify reasons to end up to classifier and feature extraction technique you were using. Include classification results with given data to you documentation. Divide data to equal training and testing sets, and perform classification. At the end of your documentation you should have a list of all references used.

## Deadline and submission

The deadline of this task is January 15th, 2007.

## Grading

The classifiers will be graded according to their classification performance, i.e., the smaller classification error the better classifier. Classification result (*classification error*) will be calculated as an average of classification result when

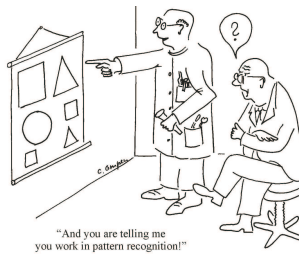
there are training and testing sets of same size. Data used in grading contains *spam* and *ordinary* messages in the same ratio as in given data.

Operation of a classifier must be reasonable, e.g., a classifier that returns randomly selected class labels is not acceptable and will lead to rejection of practical assignment. Also, your classifier should perform better than purely random classifier (which is maybe based on a priori probabilities).

# Técnicas de Reconhecimento de Padrões
# Pattern Recognition Techniques

2006/2007

Project: *Assignment 2*




**Shape OCR**

"A pattern is the opposite of a chaos; it is an entity vaguely defined, that could be given a name."

## Introduction

This practical assignment is to implement a Matlab function that classifies shapes: Circles, Squares and Triangles as shown in Figure 1.

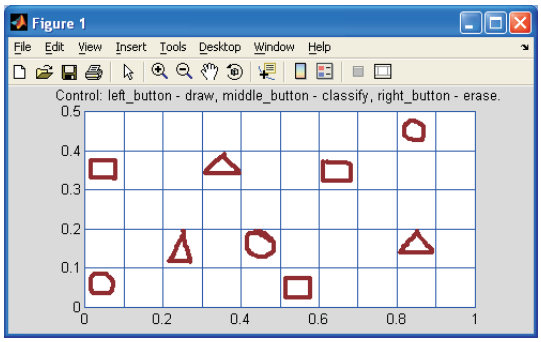



Figura 1: Shape Recognition OCR

The objective is find a classifier which performs shape recognition. Note that you can categorize Circle as a 'C', a Square as a 'S' and a Triangle as a 'T' according to what is shown in Figure 2.

You can make use of functions in the *demoOCR* given in classes and write the modifications for input/output according to your own project.

*collectMychars* - Collecting training examples for OCR.

*mpaper.m* - Allows to enter handwritten characters by mouse.

*saveMychars* - Saves images to file.

Build your own data training set and use as much data entries as you can to help you to have enough training examples. Write a Matlab function *MyTrainShapeOCR* which is able to perform training on your data set as input, being the output your *ShapeOCRModel* which will be used in the classification.

It is advisable that students develop alternative feature extractors using information from any source.

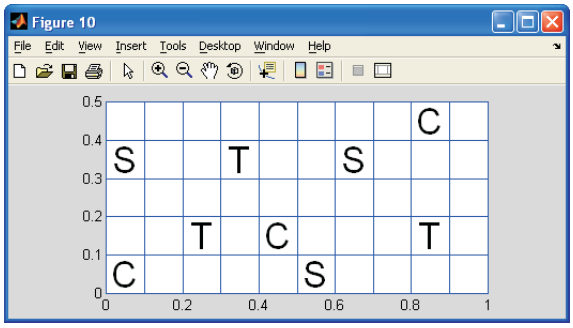



Figura 2: Shape Recognition OCR Classifier Results

## Requirements

Practical assignment is meant to be done in groups of two persons. If someone wants to work alone, this is also possible. Larger groups than group of two persons are not allowed.

## Matlab function

The Matlab function that performs classification must have name *MyShapeOCRfun* and it must take and return parameters in the way described above.

Remember to comment your codes. Write also a help section to your codes that tells the purpose of the function, usage, and explanation of parameters. In Matlab, comments following the first line of a function will show when help command is used with the name of the function.

## Documentation

Write paper documentation (in Portuguese or in English) about your project. The documentation should include a cover page where course name, project title, date, names and student numbers of the authors are mentioned.

Describe the methods used for feature extraction and classification in such detail that reader would be able to implement same kind of functions for feature extraction and classification just based on your documentation and some basic background in pattern recognition. Giving an algorithm and explaining it in words is a good way to describe methods. Justify your choices, i.e., justify reasons to end up to classifier and feature extraction technique you were using. Include classification results with given data to you documentation. At the end of your documentation you should have a list of all references used.

## Deadline and submission

The deadline of this task is January 15th, 2007.

## Grading

The classifiers will be graded according to their classification performance, i.e., the smaller classification error the better classifier.