Parallel Processing

Term Project

Team Members:

- Abdelaziz Sayed Abdelaziz 20120235
- Abdelrahman Mostafa Elattar 20120234
- Ahmed Samir Abdelzaher 20120029

Project Name: Testing/Prediction of an image using MPCA algorithm using MPI.NET and C#

-This project is a part of a face recognition system

- Generally, a face recognition algorithm contains 2 parts:
  - The training part (not in this project.)
  - Thee testing/prediction part (our objective in this project.)

- In this project we use the prediction of an algorithm called MPCA

- You can have more info about this algorithm from the following paper: http://www.utdallas.edu/~sxb027100/dock/25-429.pdf
- This project works on the part starting in page 4 equation #11.

- In this project we have :

- A training data of 5 persons, 8 images for each. (Will be loaded and not generated in this project.) you can find the training data in the train.txt file.
- And we have 5 images to test the algorithm (Included in the test folder in the .rar file.)

- In order to run this project you will need the following:

- MPI.net SDK from the following link. http://www.crest.iu.edu/research/mpi.net/software/
- N.B: it may ask you to download Microsoft cluster pack so you must download it then.
- Change the paths of the training and test files.
- build the application using an IDE (e.g. visual studio) to generate the exe file and follow the instruction in this link： http://www.crest.iu.edu/research/mpi.net/documentation/tutorial/installation.php

- The parallel scenario :

- The master loads the training data along with the test image that we need to know to which label it does belong.
- The training data is divided to (N = 16) parts for better feature extraction. As we have training data for 5 persons so we have 5*16 parts.
- In our parallel scenario, in the master, we specify a number of persons for each process that it takes (range * 16) image parts, where 'range' is the number of persons per process.

- Then each process uses the training data to compute the distance (which is the variable Dp in the paper mentioned above) between the persons that it's responsible for and the test image and send the min distance and the related label to the master process.
- The master process then receives all the min distances and labels of all the slaves and gets the min. distance and label of them all and finally print the result.
- This project handles the case where the number of persons is not divisible by the number of slaves. And it does also handle the remainder part.