Facial Recognition Using Neural Networks over GPGPU

V Latin American Symposium on High Performance Computing

Juan Pablo Balarini, Martín Rodríguez and Sergio Nesmachnow

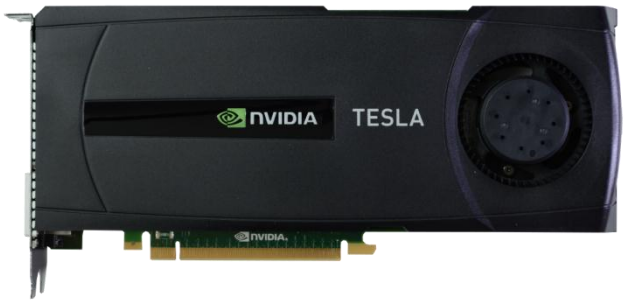
Centro de Cálculo, Facultad de Ingeniería
Universidad de la República, Uruguay

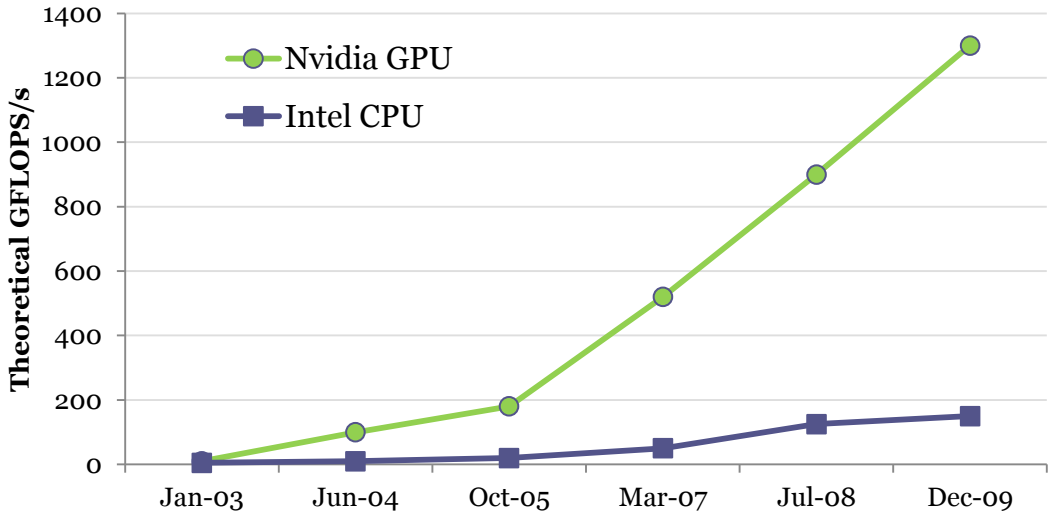
Introduction

- A parallel neural network approach implemented over Graphic Processing Units (GPU) is used to solve a facial recognition problem, which consists in deciding where the face of a person in a certain image is pointing.
- Experimental evaluation demonstrates that a significant reduction on computing times can be achieved.
- Speedup greater than 8 is achieved when contrasted with a sequential implementation and classification rate superior to 85 % is also obtained.

GPU Computing

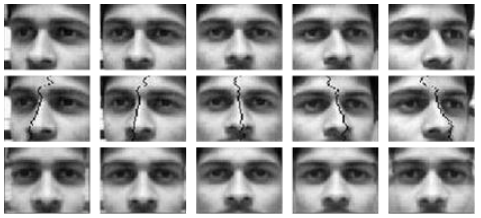
- GPUs were originally designed to exclusively perform the graphic processing in computers.
- In the last ten years, GPUs have been used as a powerful parallel hardware architecture to achieve efficiency in the execution of applications.
- Introduction of the CUDA architecture.





Face Pointing Direction

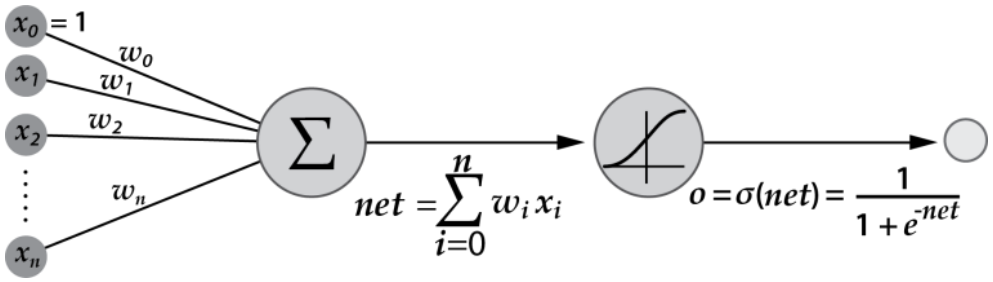
- The face pointing direction problem consists in recognizing where a human face is pointing (up, left, right and center) in a certain image.

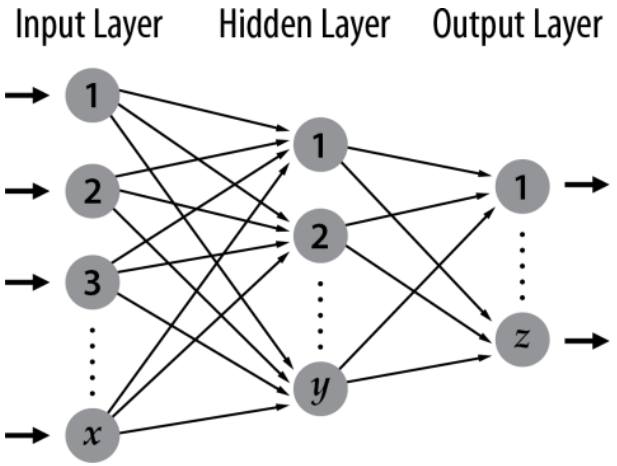


- Practical applications include:
  - Detecting if a driver feel asleep
  - Computer mouse for impaired people
  - Digital camera software that only takes a photograph if everyone is looking
- In general sequential implementations are used

Artificial Neural Networks

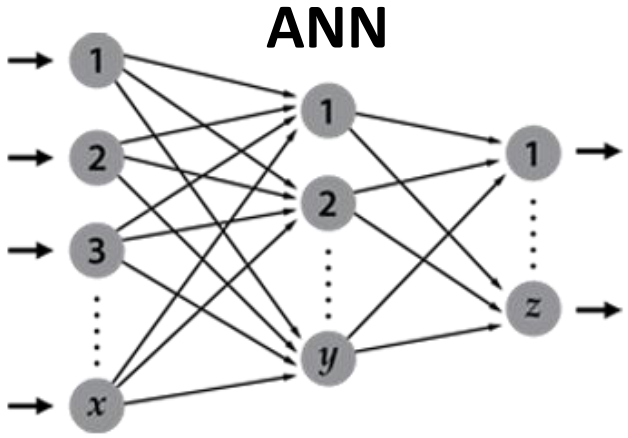
- Inspired by the human brain
- ANNs provide a general practical method for learning real-valued, discrete-valued, and vector-valued functions from examples
- Robust to errors in the training data
- Their main disadvantage is that the time needed for training can be quite high

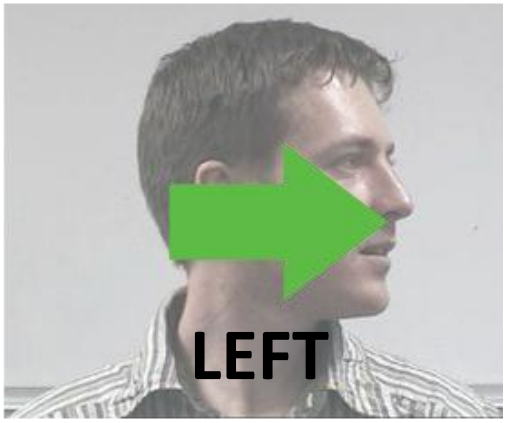




ANN for Face Detection in GPU

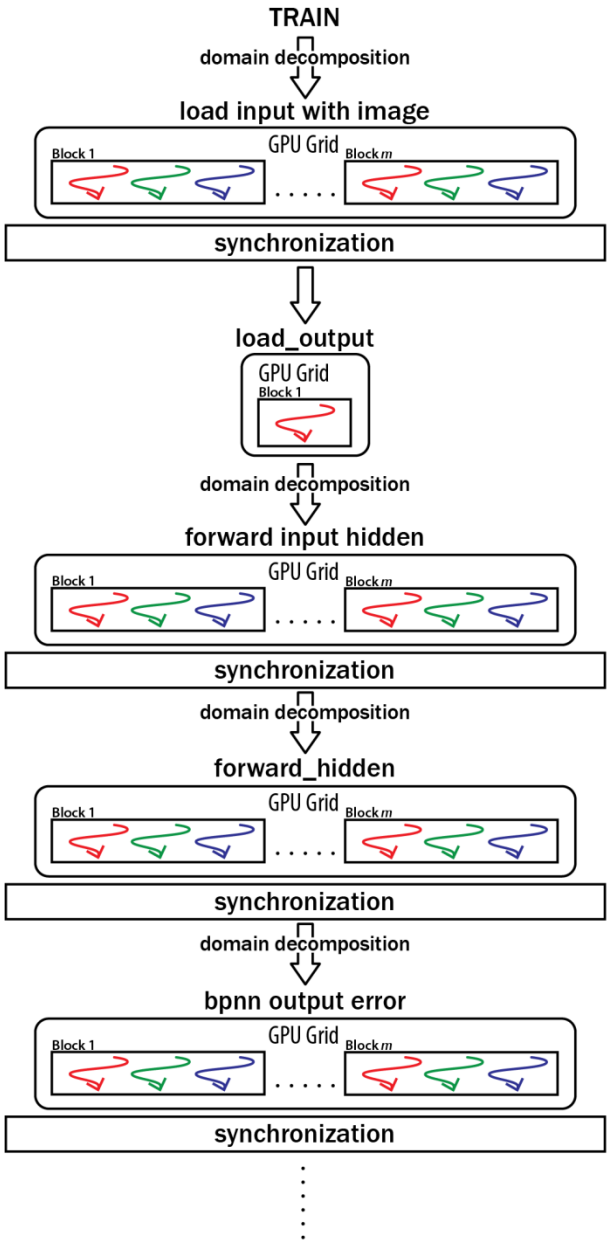
- Training and execution time can be improved by using the parallel architecture of the GPU.
- An Artificial Neural Network is used to solve the face recognition problem, trained using the backpropagation algorithm.
- The proposed parallel implementation applies the ideas in the sequential algorithm by Shufelt and Mitchell.





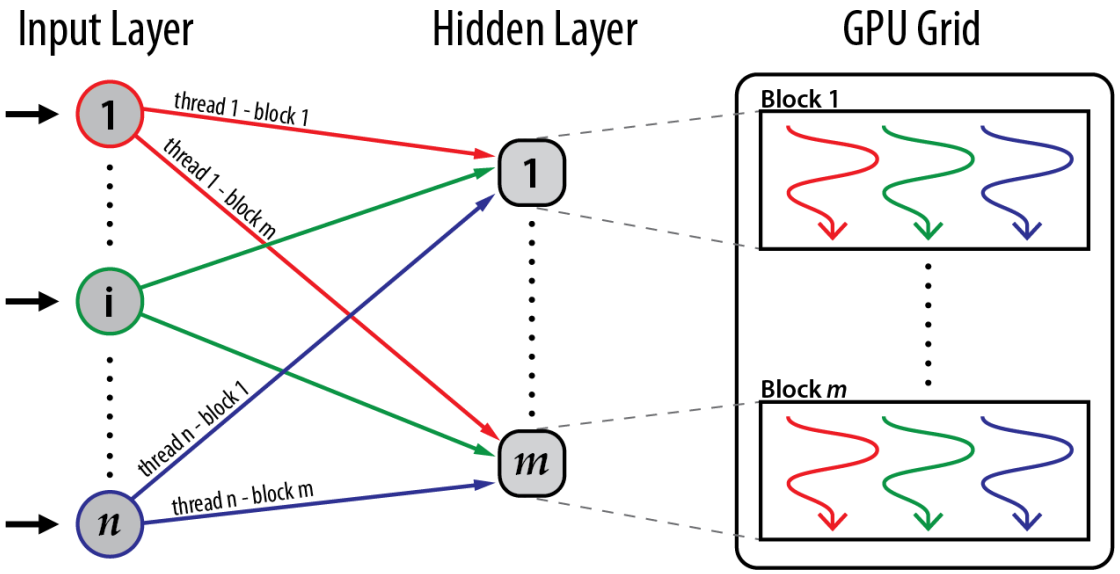
Implementation Details

- Training and execution consist of a concatenation of functions that run in parallel.
- Before calling each function a function-dependent domain decomposition is applied. In general, certain GPU threads are assigned to execute over certain neurons on the ANN.



Implementation Details

- Domain decompositions are always performed to maximize the parallel execution and to avoid serializations in the memory access.



Experimental Analysis

- **Development and Execution Platform**
  - Two platforms were considered:

Development Platform	Execution Platform
AMD Athlon II X3 445 processor @ 3.10 GHz	Core i7-2600 processor @ 3.40 GHz processor
6 GB DDR3 RAM memory	16 GB DDR3 RAM memory
GeForce GTS 450 GPU with 1 GB of RAM	GeForce GTX 480 GPU with 1536 MB of RAM

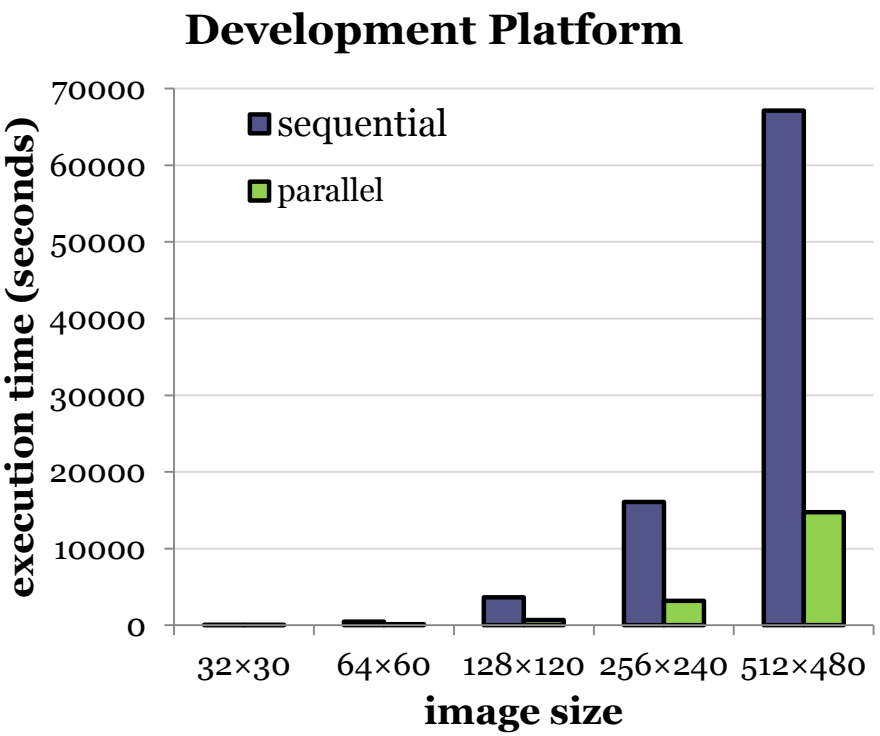
  - The limited computing power of the development platform did not allow taking full advantage of the parallel features proposed by the algorithm.

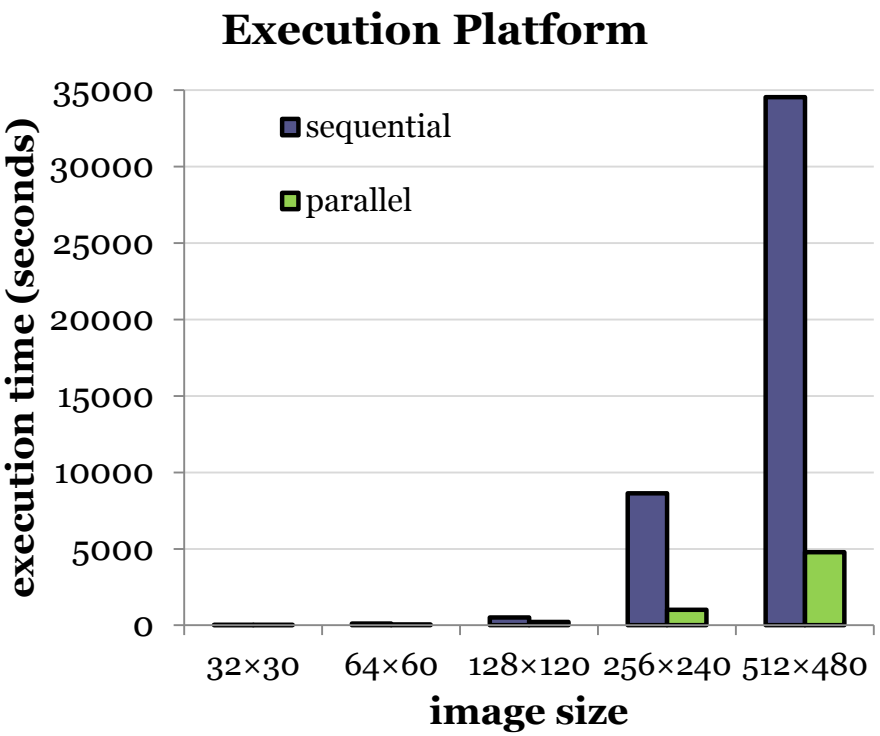
Experimental Analysis

- **Problem Instances**
  - Both sets used for training and classification were obtained from the work by Shufelt and Mitchell. These are images of different people in different poses.
  - Five image sizes:
    - 32×30 pixels, 64×60, 128×120 pixels, **256×240 and 512×480**
  - There are about 620 images, which are divided in three sets, one for the network training and the other two to measure its effectiveness.

Results and Discussion

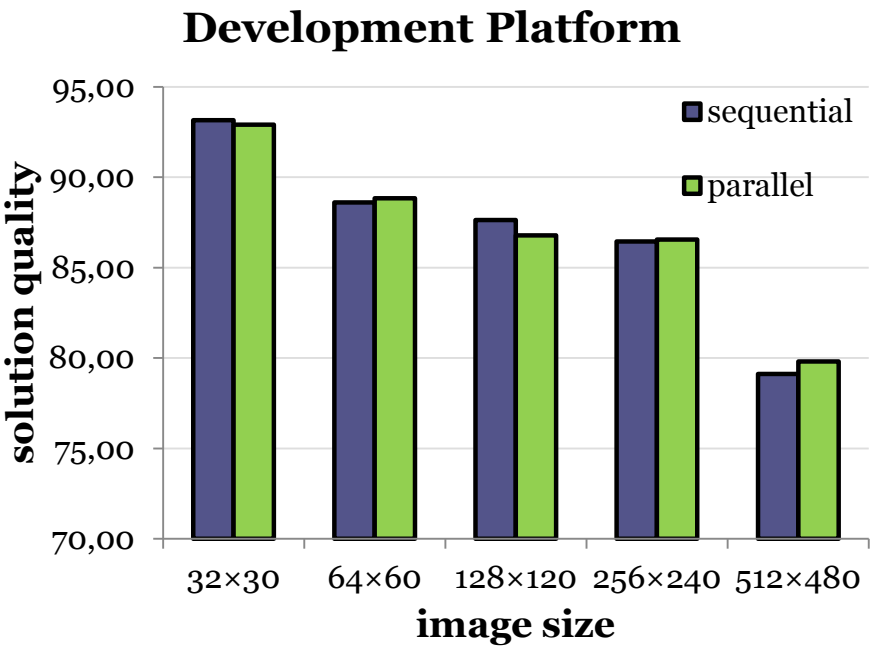
- **Execution Times**
  - All execution times are the averages and its correspondent standard deviation values, computed in 50 independent execution of the parallel algorithm for each scenario.

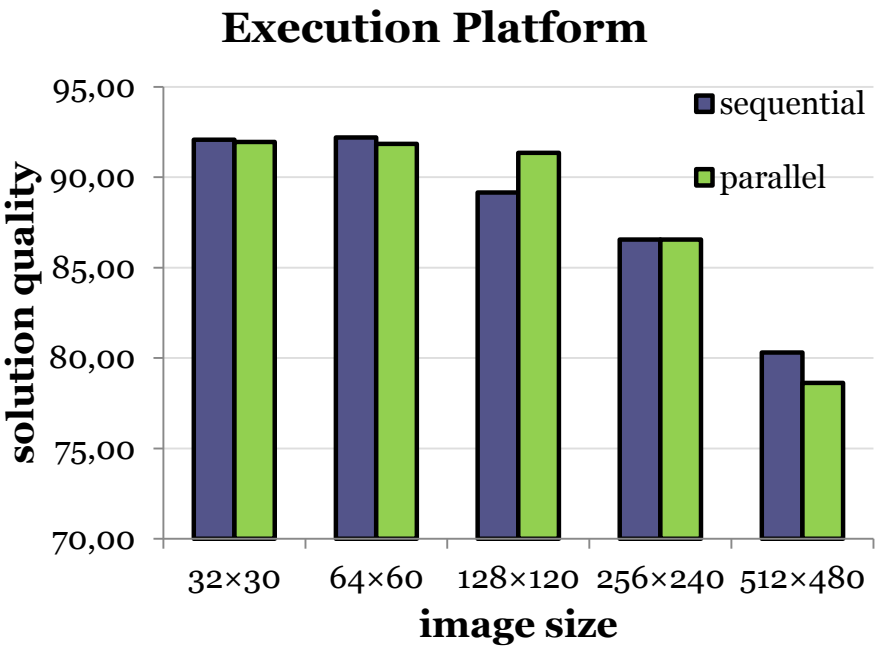




Results and Discussion

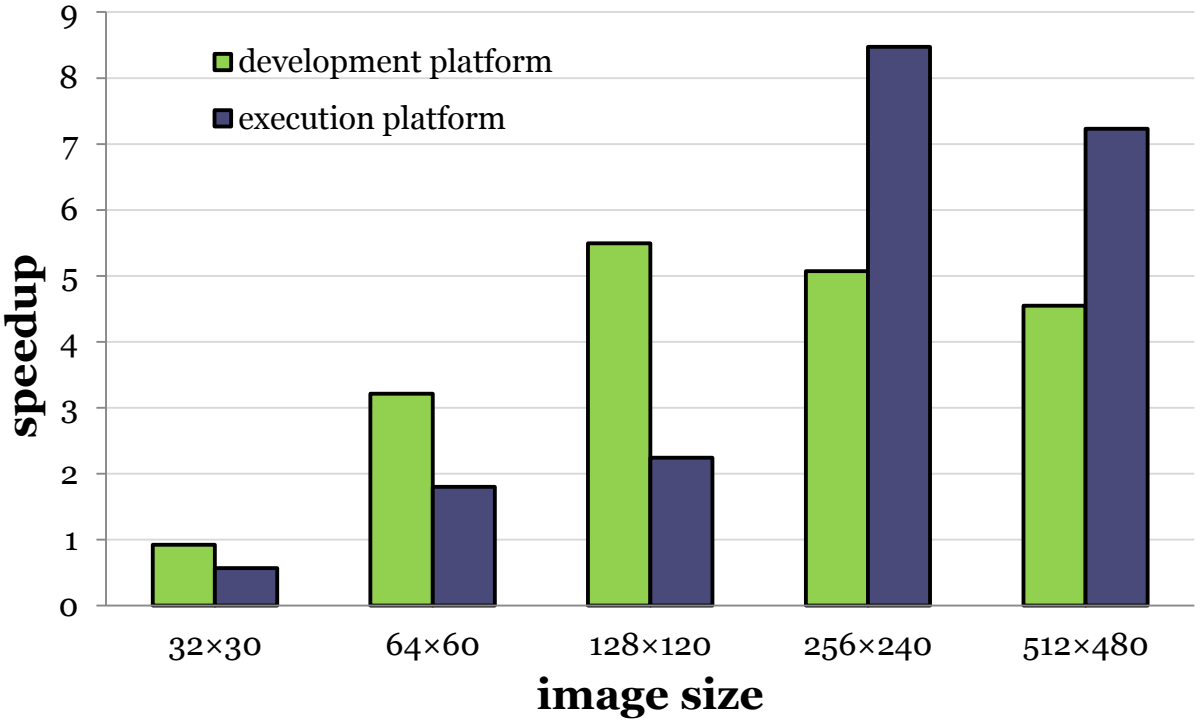
- **Solution Quality**
  - Classification rates superior to 85% are achieved for both development and execution platform.





Results and Discussion

- Speedup greater than 8 is achieved



Conclusions and Future Work

- A parallel algorithm that executes on GPU achieves a speedup gain greater than 8 when contrasted with a CPU implementation.
- Accurate classification rates (> 85%) in reasonable execution times are achieved.
- The algorithm can be easily modified to recognize other features of a human face, without significant changes in the expected execution times.
- Future Work:
  - Improve the computational efficiency and classification rates.
  - Tackle other classification/image processing problems using ANNs implemented on GPU.

Thank you for your attention

Further information available at
http://www.fing.edu.uy/inco/grupos/cecal/hpc