Robust Facial Recognition with Reconfigurable Platforms

S. Gibson, M.C. Smith, C. Gupta, A. Kruzner, J. Merwin, D. Rollend, A. Nallamuthu
Clemson University, College of Engineering and Science

Motivation
• Biometric applications have grown rapidly in the last decade with U.S. Homeland Security and Bank of America among the early adopters
• Biometric techniques are not only desirable to potentially identify threats, but also useful for keyless access
  – In “Smart Environments,” facial recognition is used to recognize people and remember their preferences
  – A smart environment contains many types of smart devices which are continuously working to make users’ lives more comfortable
• Facial recognition techniques have advantages over other biometric techniques in terms of collectability & acceptability
  – Collectability – ease of acquisition for measurement
  – Acceptability – indication of ease and willingness of public to use
• Facial recognition with improved robustness will find a number of applications where other biometric features are difficult to use
  – Surveillance tasks at airports, train stations and other public places
  – Identity verification at ATMs and controlled or secure areas
• The robustness of facial recognition suffers due to variations in
  – Illumination conditions
  – Facial expression
• Existing facial recognition algorithms are not robust for all types of variations in input data → no silver bullet!
• Pixel by pixel based matching is computationally intense
• Improving robustness increases computational requirements
• Real-time face recognition is even more challenging
  – Accurate recognition requires extensive computation
  – Large databases of possible matches must be searched
  – Less control over variations in images
• Smaller images are often used to reduce the number of required calculations
  – Decreases match accuracy
  – Makes circumvention more feasible

Analysis
• GP-GPUs
  – Extremely fast processing of graphical or spatial information [e.g. video display] where app fits a streaming floating point profile
  – High power consumption → Heat
  – Challenging SDK and no MATLAB path
• CellBE
  – Broader range of applications
  – Programming SDK has long learning curve and no MATLAB path
• Reconfigurable Computing [FPGAs]
  – Demonstrated performance advantages in several app domains
  – Excel at comparison algorithms and integer operations
  – Best ratio of FLOPS/WATT
  – Challenging SDK but proven MATLAB path!
• Comparison of PCA, LDA and ICA
  – Neither of these algorithms is robust enough to deal with all variation possibilities
  – ICA2 + cosine distance metric combination is best choice for temporal changes
  – PCA + L1 distance outperformed all others for illumination changes
  – LDA + cosine distance provided better results than others for different facial expressions

A consensus made by comparing the results of multiple algorithms may improve accuracy/reliability for “real-world” variations

Implementation Goal
• Parallel implementation of PCA, LDA, and ICA algorithms in hardware

Requirements
• Pre-Processed Training Images
• Each algorithm uses the Common Image Database

Objectives
• Improve speed of algorithms through accelerator implementation
• Concurrently project training images into multiple feature spaces such as PCA, ICA, LDA
• Compare the test image projection to training dataset in the respective feature space forms using distance metrics such as L1 norm, L2 norm, cosine, etc.
• Establish identity using a consensus algorithm

Implementation
• Most calculations in selected face recognition algorithms involve large-scale matrix arithmetic and transformations
  FPGA vs GPUs vs CellBE
• Once the PCA projection of the images is found, the ICA1, ICA2 and LDA algorithms can be performed in parallel
  Acceleration through parallelism
• A consensus algorithm (TBD) will process the distance calculation results from each projection
  – Exploring methods for possible adaptive consensus algorithm
• Initial algorithms developed and tested in MATLAB that meet the following criteria:
  – FPGA amenable
  – Utilize the same pre-processed database
• Parallel implementation of algorithms in MATLAB
  – Results indicate that image variations affect which algorithm has better recognition accuracy
  – As expected, “parallel” processing of the algorithms is slow in software

Future Work
• Fully implement algorithms on FPGAs
• Collect performance data on reconfigurable computing platforms
• Analyze results for execution and recognition performance
• Implementation studies on other platforms [e.g. GP-GPU and CellBE]
• Development of consensus algorithm

References

For further information, contact: smithmc@clemson.edu