A Heterogeneous Vision Architecture

A.W.G. Duller, R.H. Storer, A.R. Thomson, M.R. Pout, E.L. Dagless
Dept. of Electrical and Electronic Engineering, University of Bristol
Bristol BS8 1TR, UK

Real-time computer vision needs huge computing power and diverse programming approaches. As the processing progresses away from the pixel level, we expect to deal with fewer, more complex objects. No one architecture or language seems able to meet these demanding requirements, but rather an integration of different architectures and software techniques - a heterogeneous vision architecture.

GLiTCH

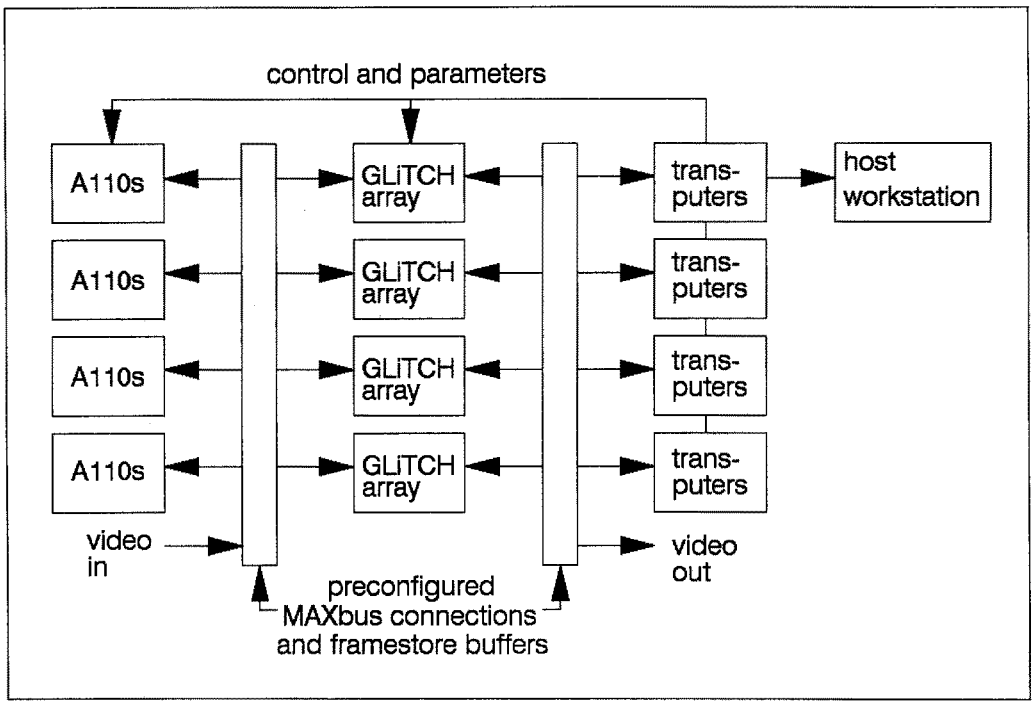
GLiTCH (**G**oes **Li**ke **T**he **C**lappers, **H**opefully) [1] [2] is a full custom VLSI associative processor array [3] designed for computer vision by the authors in collaboration with the SERC IC Design and Test Centre at UMIST.

The GLiTCH chip has 64 processing elements (PEs), each with 68 bits of content addressable memory (CAM). Any number of GLiTCH chips can be linked to form a 1-D array of PEs with communication hardware which can simulate a 4, 6 or 8 connected mesh. The array has a common microcode sequencer, program and data memories all under the control of a single transputer. A video shift register runs down the edge of the array, having one stage in each PE. By shifting concurrently with PE operation, it enables an image region to be loaded into the array while a region loaded previously is being processed.

Research already completed [4] shows GLiTCH to be suitable for a number of typical vision tasks including Laplacian of Gaussians filtering, histogram equalisation, connected component labelling, Hough and fast Fourier transforms and image resampling.

A Proposed Heterogeneous Vision Architecture

We envisage an architecture where feedback from high level processes directs the nature of the lower level processes and the regions of the image on which they work.



A Heterogeneous Vision Architecture

This type of flexibility becomes increasingly important as improved technology provides us with higher resolution images; the high resolution is of benefit in areas of the image containing information we require, but a hindrance in areas which are receiving, unneccesarily, the same amount of processing.

In order to provide an environment where processing power can be applied to selected regions of interest, an architecture is proposed (see figure) containing compatible modules of three different types. The three types of module can be connected in a variety of configurations, usually with intermediate framestores as buffers, allowing modules to work on different sized regions or load data at different rates. Each module contains at least one controlling transputer which communicates parameters with other modules.

It is important that the modules are combined in a way which does not limit their functionality by imposing too rigid or inappropriate a structure; a point-to-point video bus allows us complete flexibility of interconnection between modules.

- A110 modules: These comprise a number of INMOS (TM) A110 chips fed by MAXbus (TM) video interfaces. They perform fast, low-level processing on a stream of video data.
- GLiTCH modules: These contain an array of GLiTCH chips, described above, with MAXbus interfaces. The controlling transputer coordinates the transfer of video data over MAXbus and communicates non-video results with other

modules over transputer links.

- Transputer modules: These assimilate and analyse data from the modules above, processing higher-level information such as edge tokens, region descriptions etc. The results of this processing provide the final output of the subsystem and feedback to the GLiTCH and A110 modules.

Typically, the top level transputers will be responsible for controlling (via transputer links) which regions of an image are processed in the lower level modules and the parameters to be used in those processes. In this way all the available parallelism can be concentrated where it is most effective. Several of these subsystems will operate in parallel, each performing different operations on the entire image, each communicating with its fellows through the links of their transputer modules.

The aim of building the system described here is to discover the optimum configurations for a wide range of computer vision problems and to develop the powerful software tools needed to program and control a heterogeneous architecture.

Acknowledgements

We wish to acknowledge the financial support of the Science and Engineering Research Council, BP International Limited, the MOD at RSRE and the EEC Esprit programme for this work.

INMOS is a trademark of the INMOS group of companies.

MAXbus is a trademark of Datacube.

References

[1] Duller A.W.G., Storer R., Thomson A.R & Dagless E.L.
An Associative Processor Array for Image Processing
Image and Vision Computing Vol. 7 No. 2 (Butterworth Press) May 1989.

[2] Duller A.W.G., Storer R., Thomson A.R., Dagless E.L., Pout M.R., Marriott A.P., & Goldfinch J.
Design of an Associative Processor Array
Proceedings IEE Vol. 136 Part E, No. 5 pp 374-382, September 1989.

[3] Foster C.C.
Content Addressable Parallel Processors
Van Nostrand Reinhold, New York 1976.

[4] Duller A.W.G., Storer R., Thomson A.R., Pout M.R., & Dagless E.L.
Image Processing Applications Using An Associative Processor Array
Proceedings of the 5th Alvey Vision Conference, pp289-292 September 1989.