Efficient Memoization Strategies for Object Recognition with a Multi-Core Architecture

George Viamontes, Mohammed Amduka, Jon Russo, Matthew Craven, Thanhvu Nguyen
Lockheed Martin Advanced Technology Laboratories
3 Executive Campus, 6th Floor • Cherry Hill, NJ 08002
Phone (856) 792-9766 • Fax (856) 792-9925
{gviamont, mamduka, jrusso, mcraven, tnguyen}@atl.lmco.com

11th Annual Workshop on
High Performance Embedded Computing (HPEC)

MIT Lincoln Laboratory
18-20 September 2007

This work was sponsored by DARPA/IPTO in the Architectures for Cognitive Information Processing program; contract number FA8750-04-C-0266.

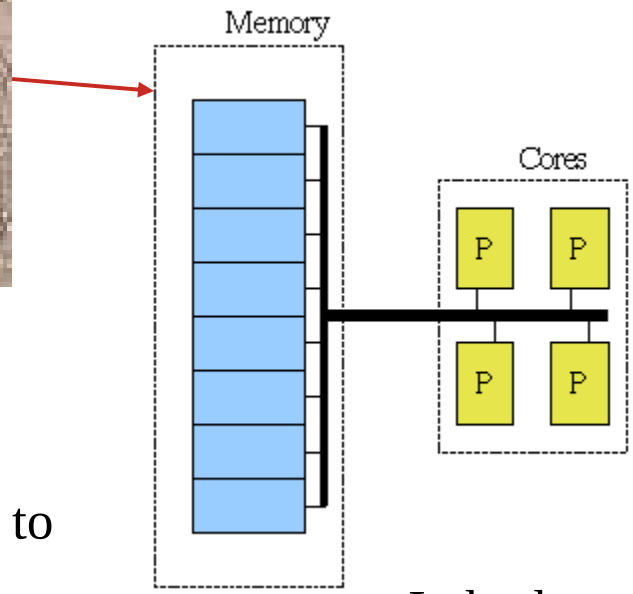
LOCKHEED MARTIN

Object Recognition

- Goal: efficiently identify objects in a 2D image
- Good techniques must account for translational, rotational, and size variance as well as partial obscurement when comparing a real object to a library
- Examples include:
  - The Chunky SLD algorithm from Sandia is representative of the best statistical methods and has been ported to FPGAs
  - Geometric hashing makes use of simple distance calculations in "hash space" to account for image variations
- Object recognition techniques are highly parallelizable and require only simple calculations, but they are extremely memory intensive

Memory Bottleneck in Multi-Core

Statistical methods require simple computations on individual pixels. The image library in memory must be compared to many times.

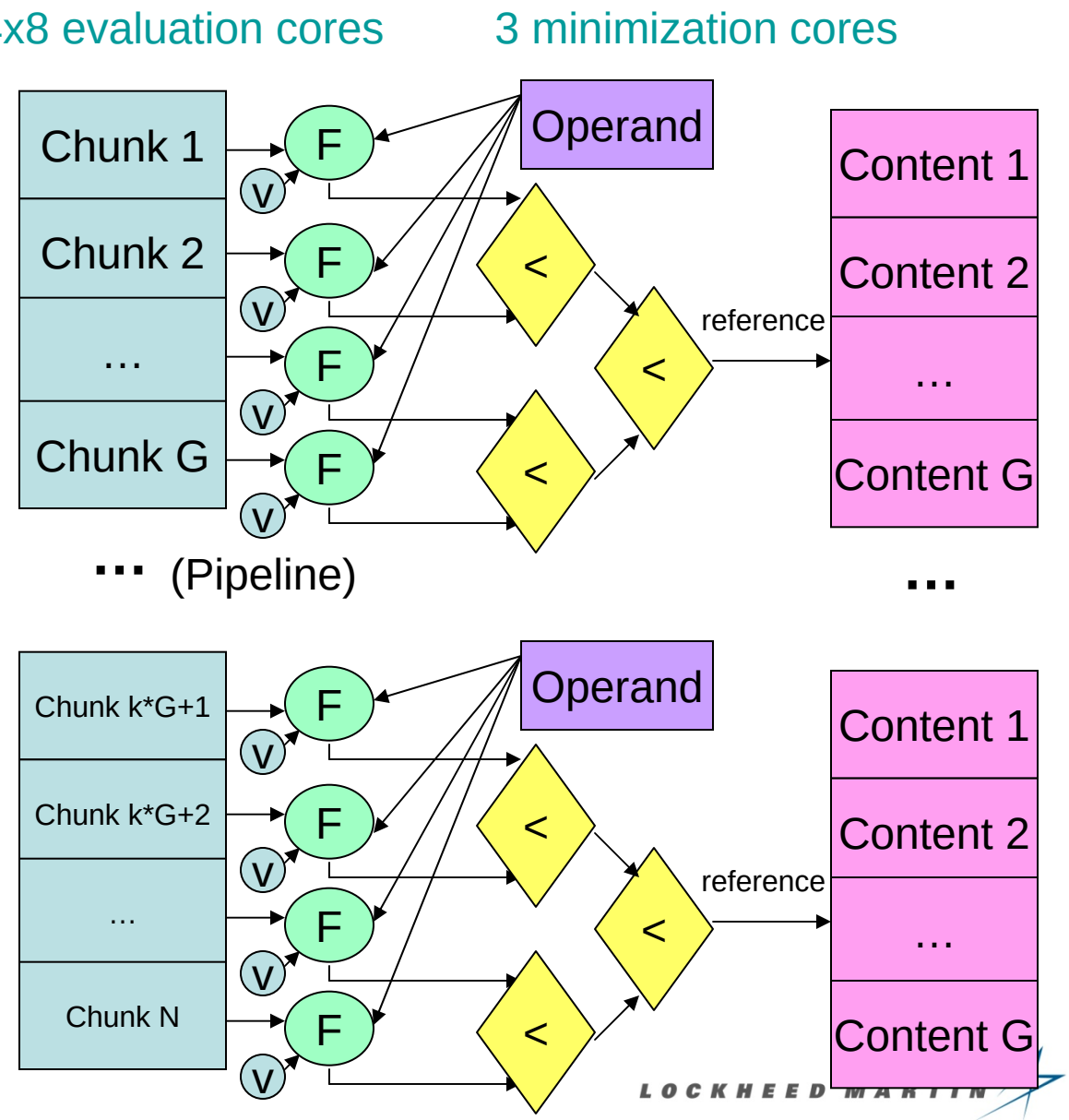


Geometric hashing applies a hash function on pixels or small groups of pixels (features) to map to a location in memory. Simple distance calculations in hash space account for image variations.

In both types of algorithms, memory access is the bottleneck, not processing power.

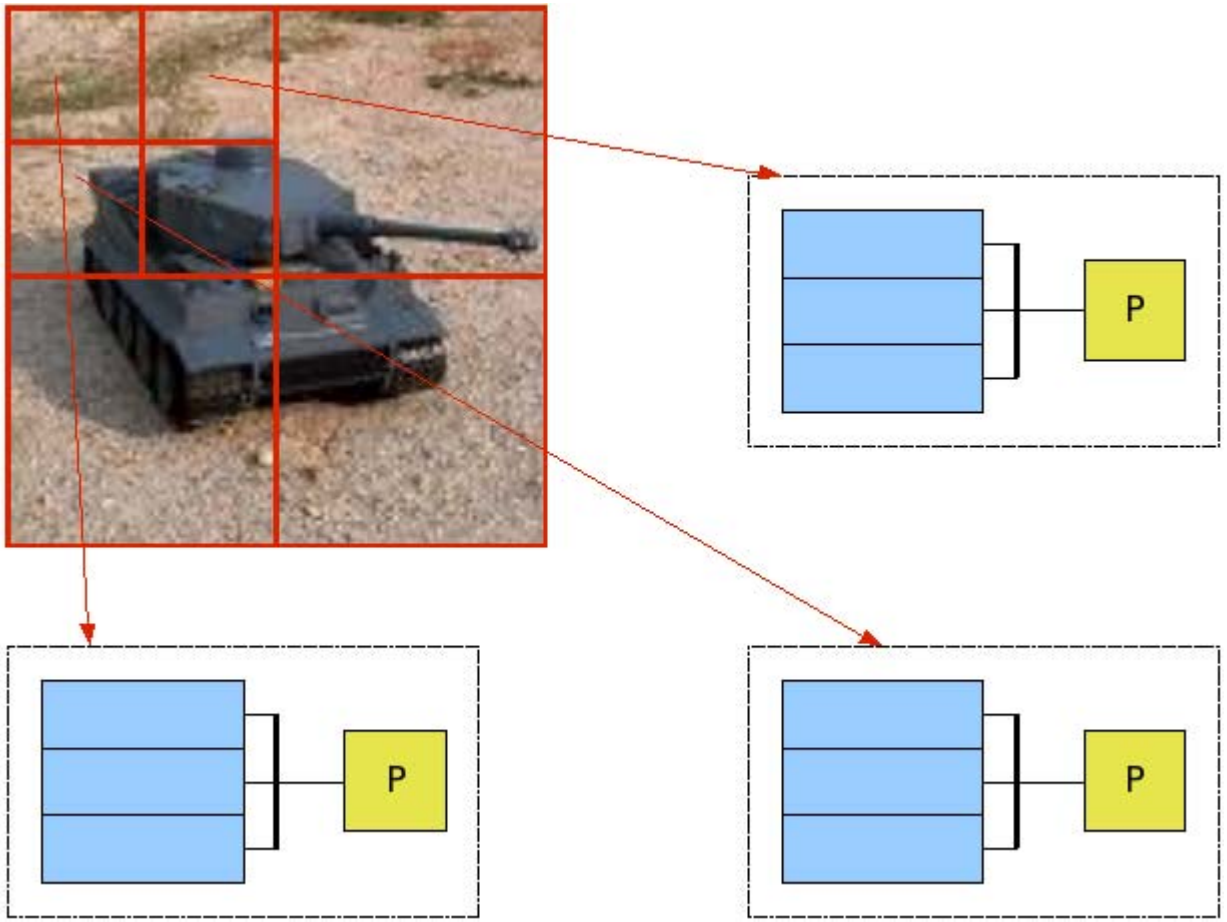
Programmable Objective Evaluation Memory (POEM) For Hardware-level Memoization

- **Programmable Objective**
  - **Inexact associative lookup**
  - **vector chunks and operands**
  - **Room for additional variable (random, decay)**
- **Example:**
  - **Conditional Sum of Absolute Differences**
  - **Computed SIMD parallel (vector and chunk)**
    - **8 x N dual port memory channels**
    - **Vector operation parallelized and pipelined**
  - **Trained to optimize spanning coverage of correct lookup**
- **Performance**
  - **FPGA: 50 MegaChunks per Second**
  - **Projected Chip: 200 MegaChunks per Second**



Proposed Architecture

Avoid memory bottleneck by distributing image across POEM chunks



Each memory/processor chunk is POEM-based to implement hash space calculations