Analysis of Computational System Performance in Automatic Target Recognition

Joseph A. O'Sullivan
Michael D. DeVore
Electronic Systems and Signals Research Laboratory

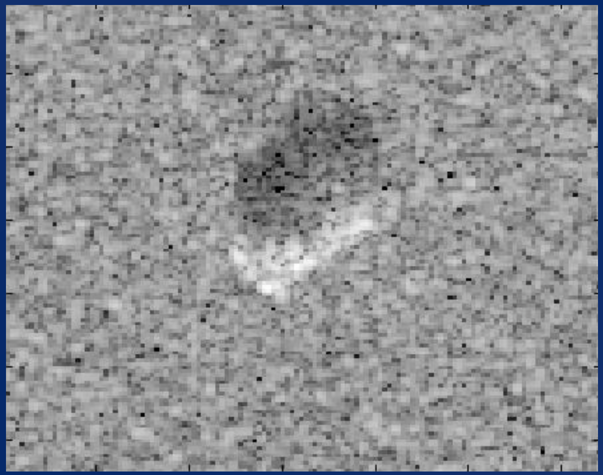
Mark A. Franklin
Roger D. Chamberlain
Computer and Communications Research Center

Washington University in St. Louis
School of Engineering and Applied Science

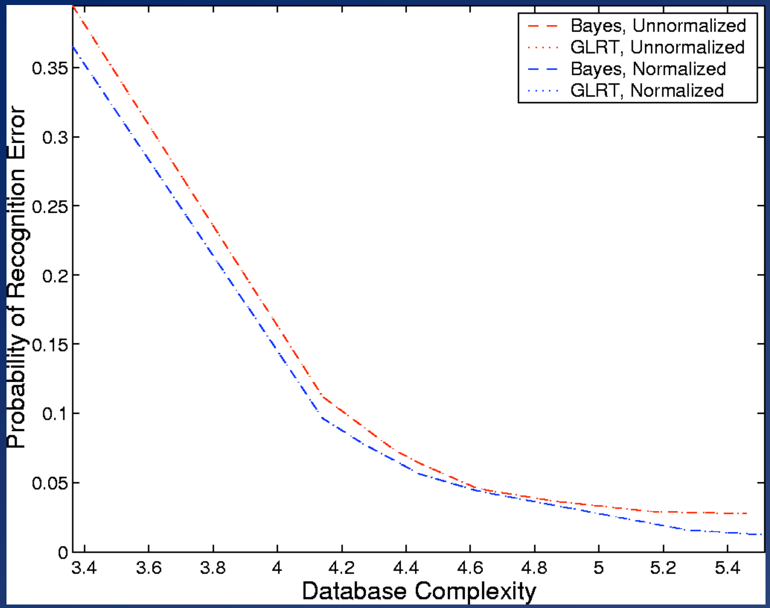
Supported by: DARPA grant DAAL01-98-C-0074
Boeing Foundation
ONR grant N00014-98-1-06-06

Overview

- Factors of Interest
  - Result Quality
  - Throughput
  - System Resources
- Illustration from Automatic Target Recognition (ATR)
- Relating Factors of Interest
- Computational Model
- Example
- Conclusions
- Future Work



Sample SAR Image (BMP2)



Result Quality vs. Complexity

Introduction

Goal:

A framework for explicit connections between application results and system performance

Approach:

Model the application and system to relate three factors

1. Quality of Results
2. Required Throughput (not latency)
3. System Resources

Results:

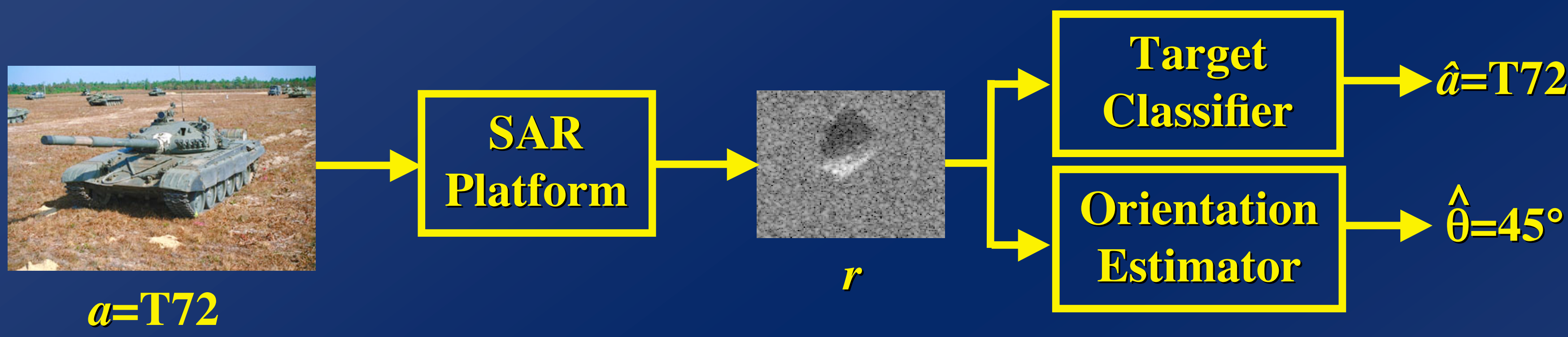
Apply the approach to automatic target recognition (ATR) from synthetic aperture radar (SAR) images

Factors of Interest

Dependencies between result quality, throughput, and computing resources help determine:

- type of platform (commercial off the shelf or custom)
- number and speed of processors
- interconnection network bandwidth
- memory bandwidth

ATR Illustration



For classification/estimation components we relate:

- Quality - Probability of erroneous classification
- Throughput - Target images processed per second
- Resources - Processors, memory and I/O bandwidth, etc.

Factor Inter-relationships

- ATR systems are explicitly or implicitly based on models of targets with some complexity C
- More complex target models require more computation but can yield better results; $\Pr(\text{error})=f(C,\alpha_{SAR})$
- Target model complexity and computational power determine overall system throughput; $T_{CHIP}=h(C,\alpha_{COMP})$
- Given an architecture, both result quality, Pr(error), and throughput, $R=1/T_{CHIP}$, are parameterized by target model complexity

ATR as an Optimization Problem

- ATR can be viewed as maximizing a measure of goodness over all classes, a, and orientations, θ.
- Likelihood based approaches maximize the probability density function of an observed image, $\boldsymbol{r}$.

- Example: Model pixel i as independent, zero mean, complex conditionally Gaussian, with variance $\sigma_i^2(\theta,a)$

$$p_{\boldsymbol{R}|\Theta,A}(\boldsymbol{r}|\theta,a)=\prod_i \frac{1}{\pi\sigma_i^2(\theta,a)}e^{-\frac{|r_i|^2}{\sigma_i^2(\theta,a)}}$$

- Variances, estimated from training data, must be stored

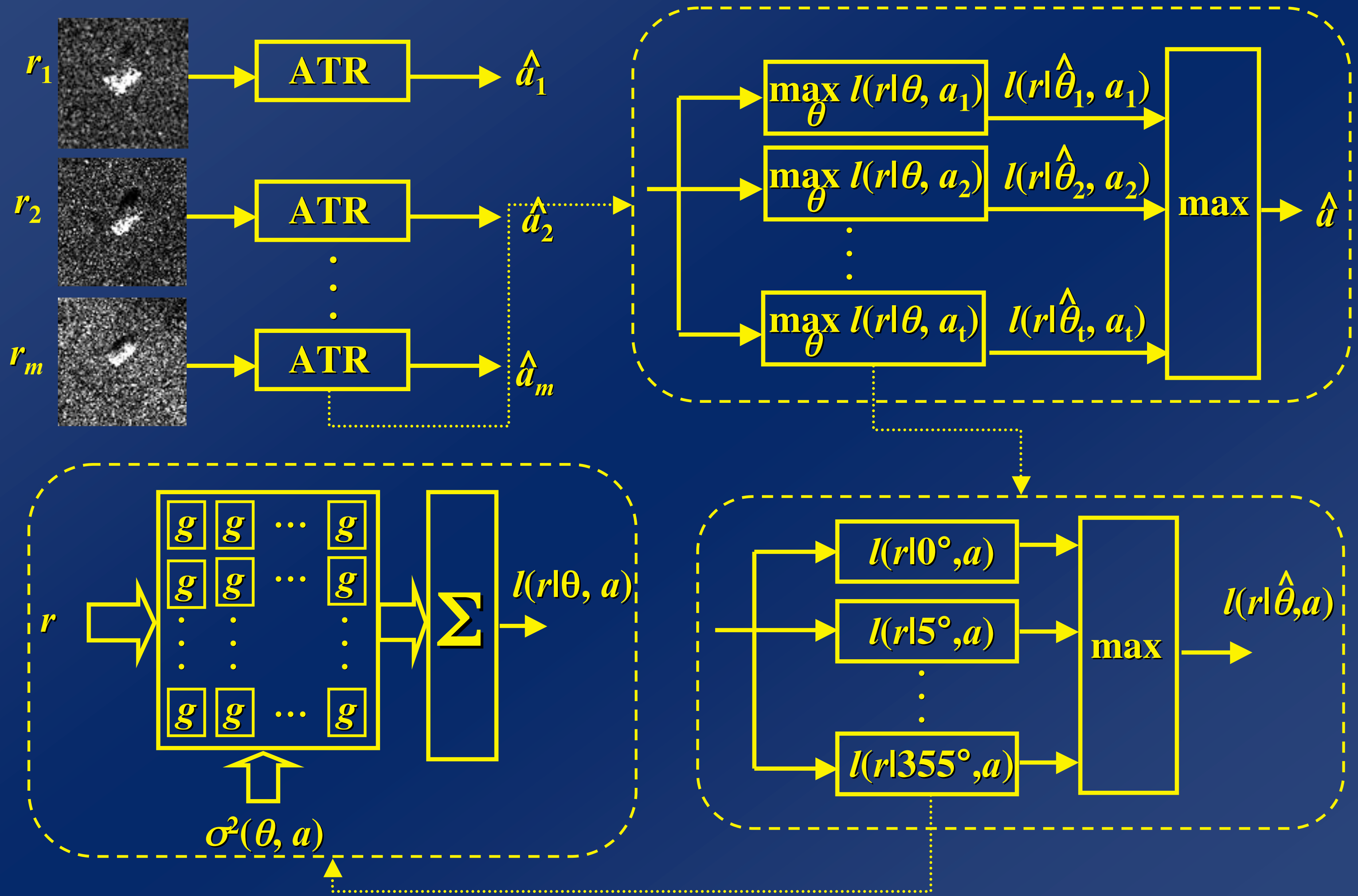
ATR as a Parallelizable Operation

- Maximizing $p_{R|\theta,A}$ is equivalent to maximizing the log-likelihood, $l(r|\theta,a) = k + \ln p_{R|\theta,A}$

$$l(\boldsymbol{r}|\theta,a) = -\sum_i \left[\ln \sigma_i^2(\theta,a) + \frac{|r_i|^2}{\sigma_i^2(\theta,a)} \right]$$

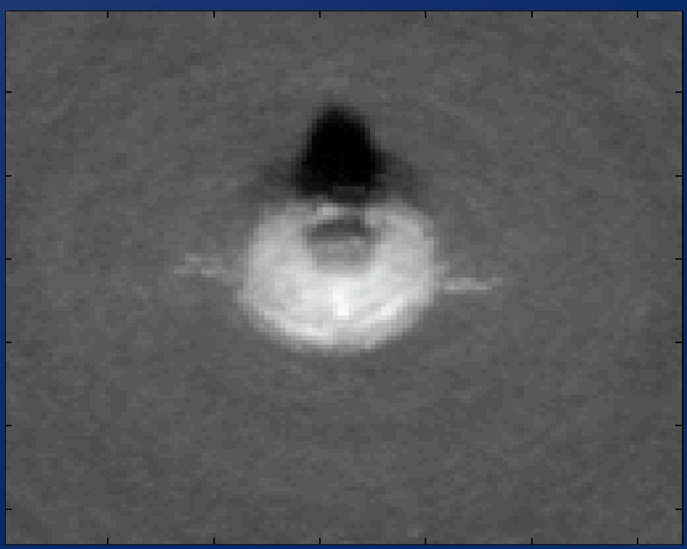
- Each measured value, r_i, undergoes operations of the same form for all pixels, orientations, and target classes

ATR as a Parallelizable Operation

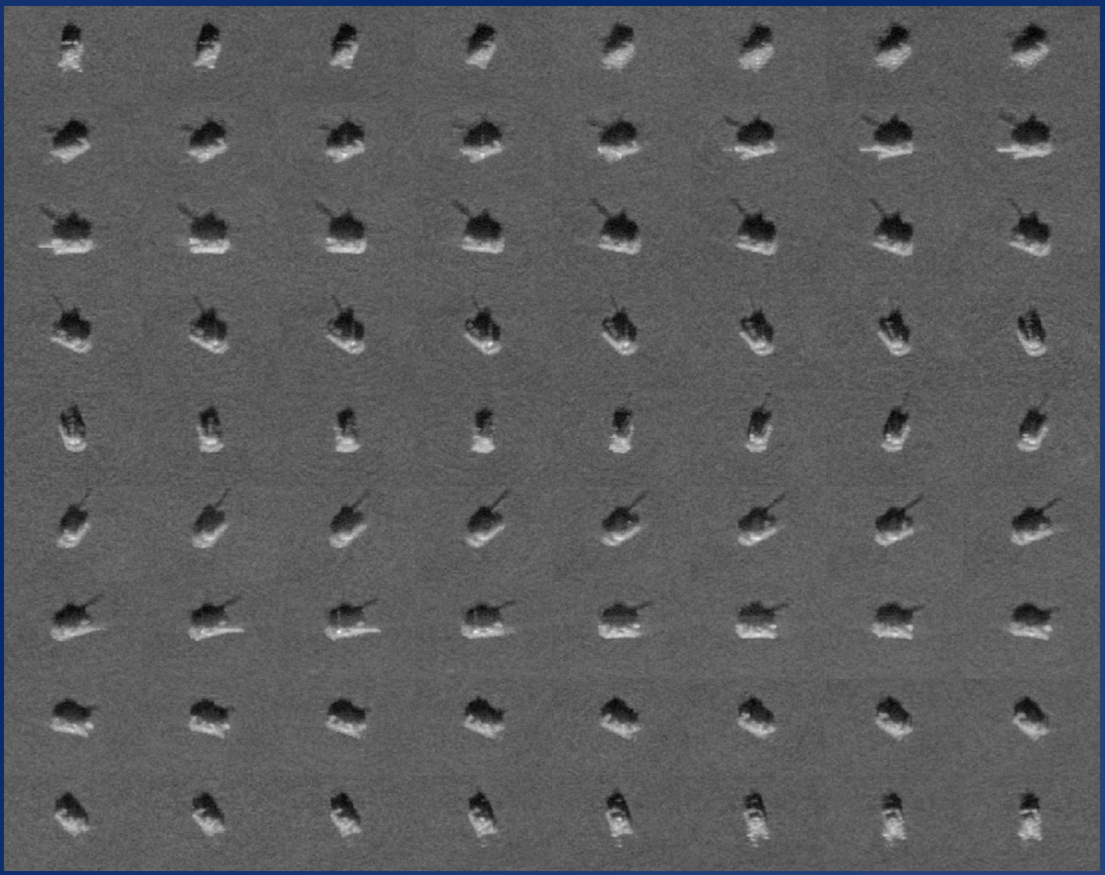


Quality of Results and Complexity

In this context, target model complexity relates to resolution in the approximation of $\sigma^2(\theta,a)$



Coarse model of aT62 tank, 1 template with 16K floats



Fine model of a T72 tank (1/5 relative scale), 72 templates totaling 1.1M floats

Result Quality and Throughput

- ATR hinges on likelihood function evaluation
- Each implementation decision sets a maximum number of function evaluations per unit time
- Maximum number of function evaluations determines what level of model can be used
- Level of model determines ATR performance
- Approach is to determine, for any combination of system parameters, the best achievable performance as a function of required chip rate

Computational Models

$$T_{CHIP} = 3\frac{LMN}{P}T_1 + \frac{LMN}{P}T_2 + T_3$$

T_{CHIP}	sec/SAR Image	L	templates/target
T_1	sec/clock cycle	M	targets
T_2	sec/template memory read	N	pixels/template
T_3	sec/SAR Image load	P	processors

Chip processing rate R=1/T_{CHIP}

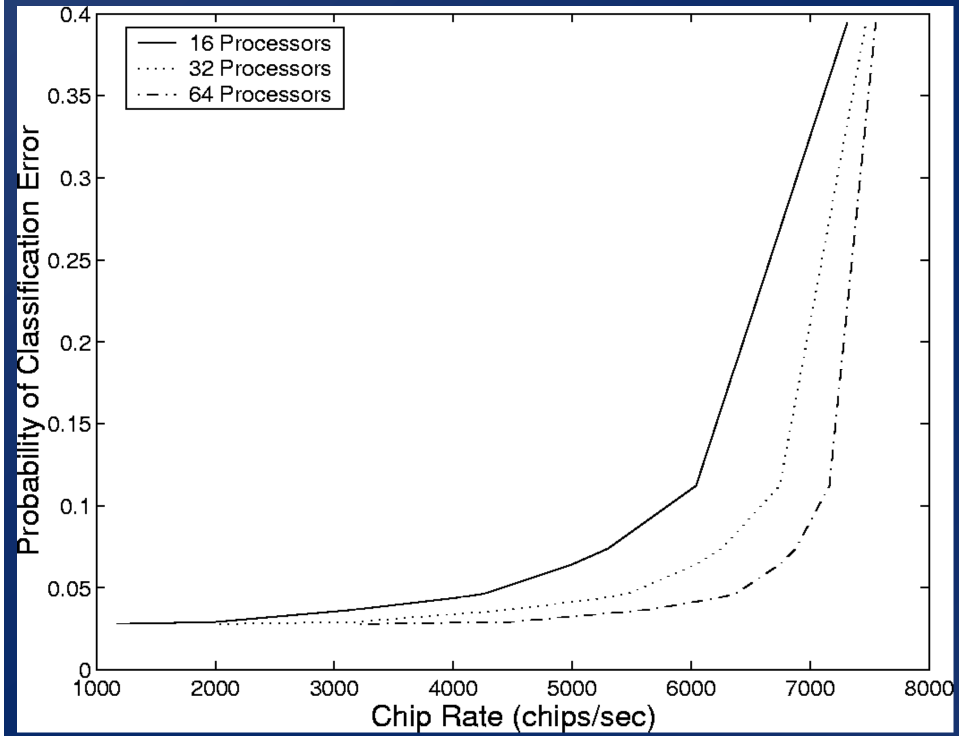
Assumptions:

- Each CPU optimizes over a region of the search space
- Multi-issue CPU with 2 instructions/clock cycle
- 6 instructions per pixel

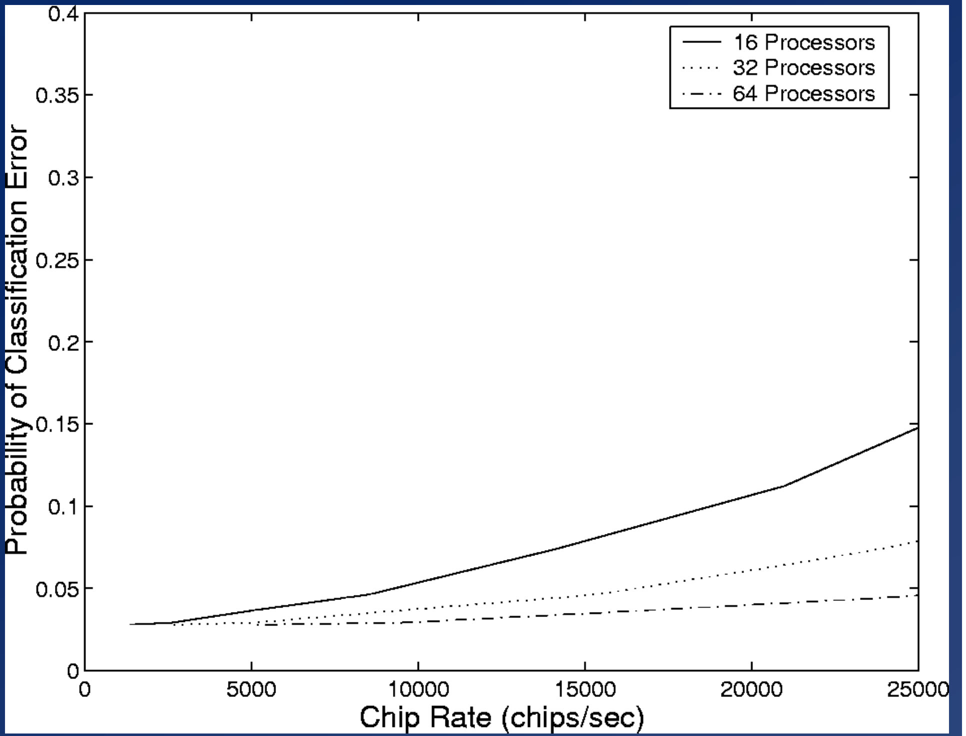
Example

$T_2 = T_1$ with prefetch
1 GHz clock
Varying target model complexity
(*L* templates/target and *N* pixels/template)

16 KB/SAR Image (4B floats)
M=10 targets



1 Gb/s Interconnection Network



10 Gb/s Interconnection Network

Example

- Figures show increase of chip rate provided by more processors for fixed probability of error
- Alternatively, they show decreased probability of error with more processors for fixed chip rate
- Curve convergence at low chip rates indicates small recognition improvement at high target model complexities
- For 1Gb/s bus, convergence at high chip rates indicates time to load SAR image dominates total chip processing time

Conclusions

- Throughput demands may vary with conditions of use
- Quality of results as a function of required throughput is determined in part by system implementation
- Models of application performance and system performance can be combined to find acceptable combinations of result quality, throughput, and system design.
- Framework for combining ATR performance and system performance

Future Work

- Development of ATR algorithms is ongoing
  - how to get the best quality results from the lowest complexity
  - accommodate target articulation and other pose parameters
  - configuration variations within target types
- Development of more advanced computation models
- Extensions to model to pixel-level parallelism