

Some Performance Results for the
ButterflyTM Parallel Processor

Bob Thomas

BBN Laboratories

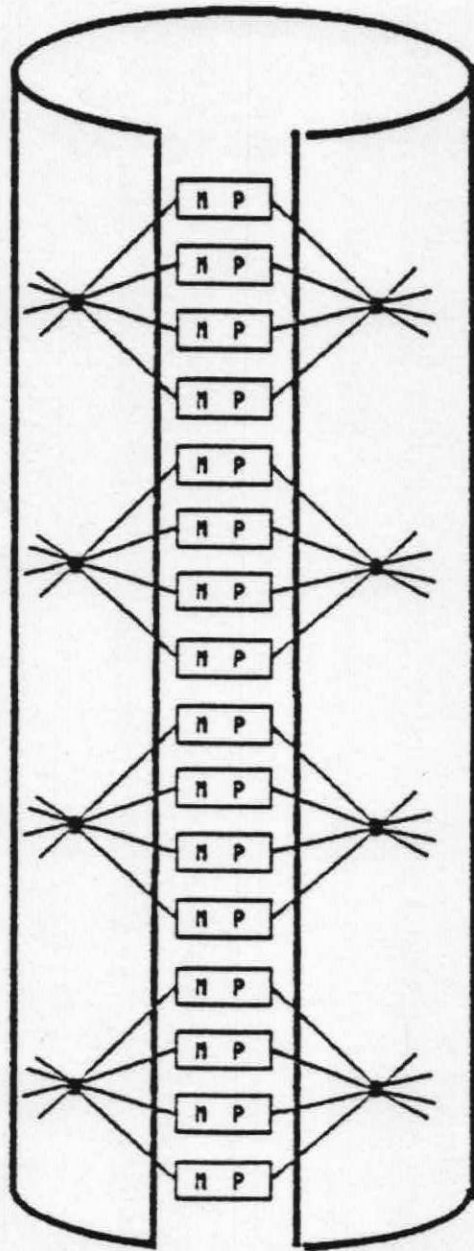
Butterfly is a trademark of Bolt Beranek and Newman Inc.

Outline

1. Butterfly Parallel Processor.
2. Benchmarking Technique.
3. Programming Methodology.
4. Some Results.

1. The Butterfly Parallel Processor is:

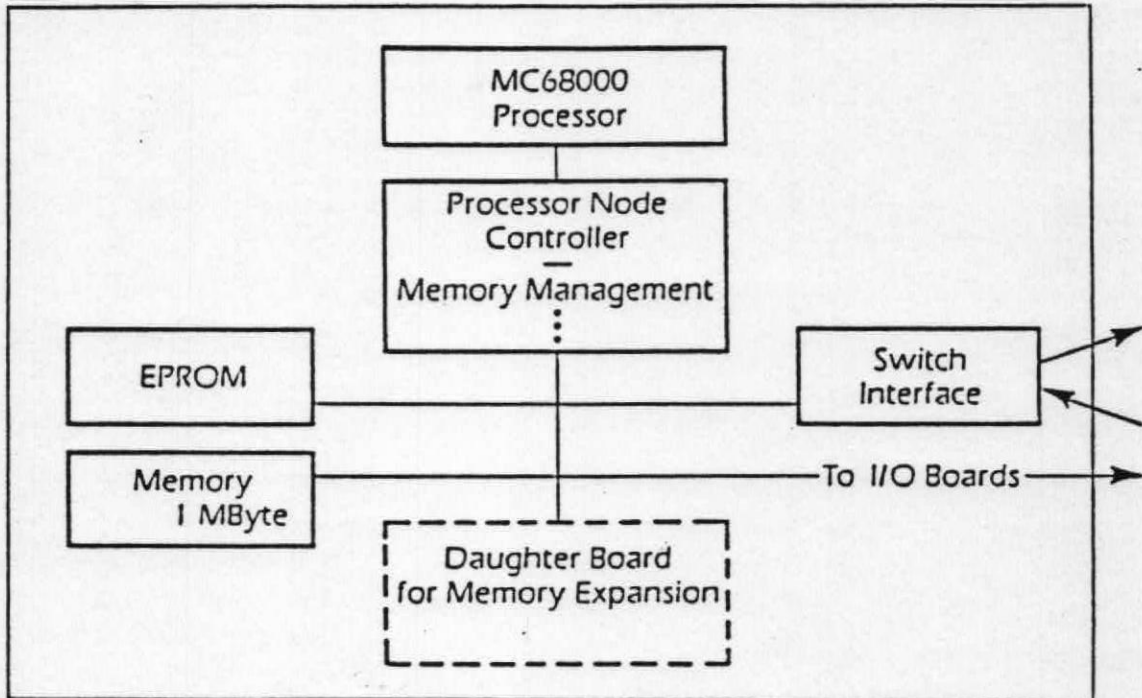
- o a MIMD machine.
- o a tightly coupled,
shared memory machine.
- o expandable over a wide
range of configurations.
- o a homogeneous multiprocessor.



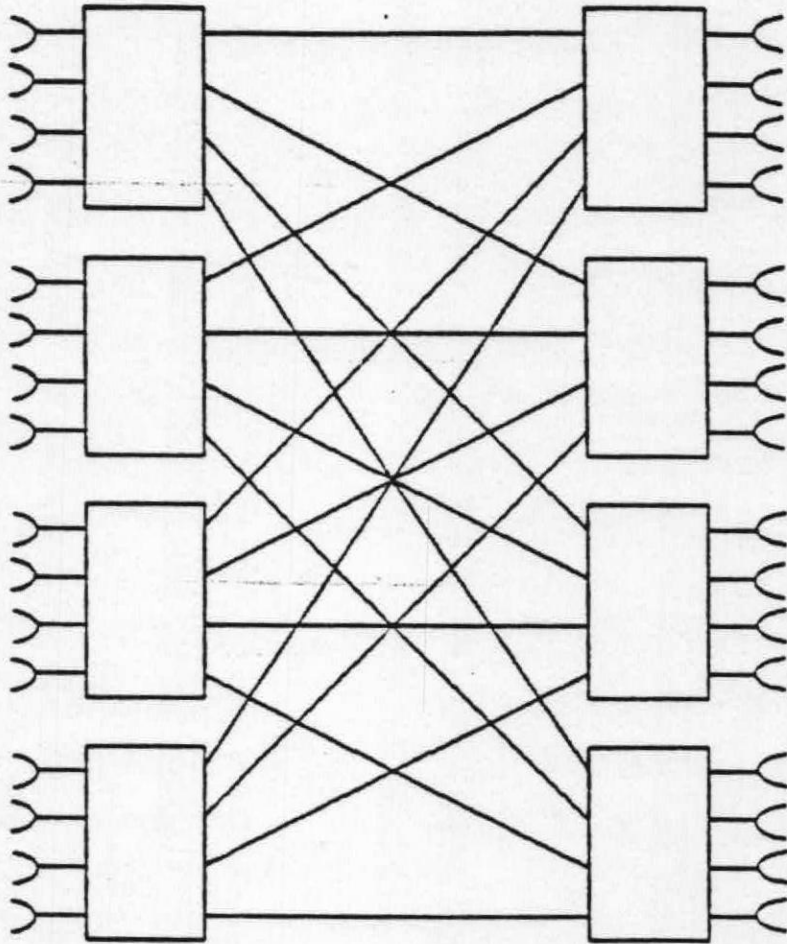
P = Processor
 M = Memory

Processor nodes and switch for a 16 processor configuration.

Processor Node



Butterfly Processor Node Block Diagram.



A 16 input-16 output Butterfly switch.

2. Benchmarking Technique.

Objective:

To measure:

1. Ability of various applications to effectively utilize multiple processors.
2. Multiprocessor overhead.

Technique:

1. Measure runtime of program on 1, 2, ..., N processors.
2. Compare runtime of optimized uniprocessor version of application with runtime of multiprocessor version on a single processor.

3. Programming Methodology.

Concerns:

o Storage Management:

Goal: Keep all memories equally busy to prevent slowdown that occurs when many processors access a single memory.

Approach: Uniformly distribute data across the memory of the machine.

o Processor Management:

Goal: Keep all processors busy to prevent slowdown that occurs when some processors sit idle.

Approach: Decompose problem into T tasks, where $T \gg P$.
Rely on statistics to minimize idle time.

Programming Methodology

Processor Management (Cont.)

Applications are structured to have two parts:

1. Set of routines that perform the tasks.
2. One or more "task generators" that identify "next" task for execution.

Task Generator Notion:

- o 2 Procedures:

Generator procedure (G) that takes as parameters a worker procedure (W) and description of data.

- o G calls W specifying subsets of the data until work is completed.
- o Processors are used as they are available to execute the calls of W.

4. Results

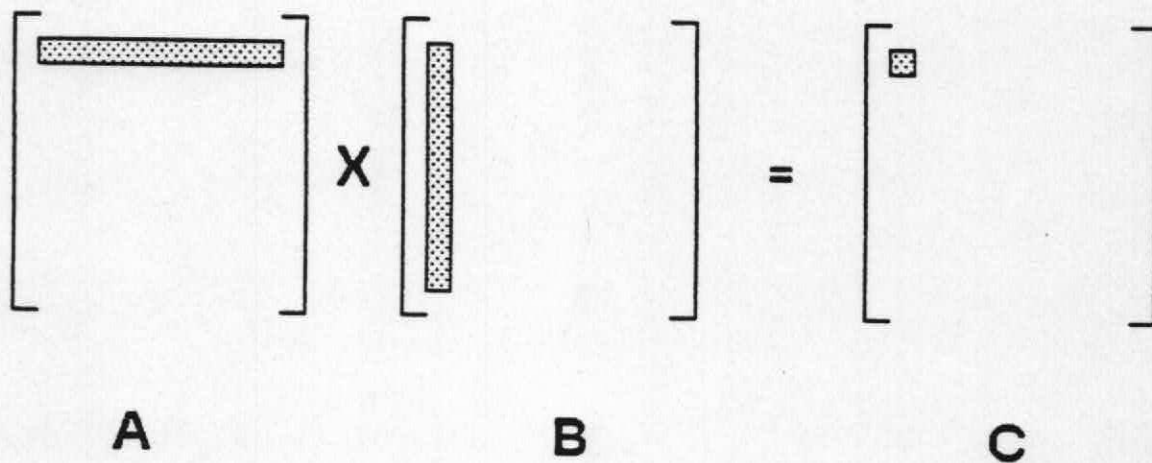
- o Matrix Multiplication.
- o Solution of Simultaneous Linear Equations.
- o Image Processing Utilities.

Convolution.

Region relabeling.

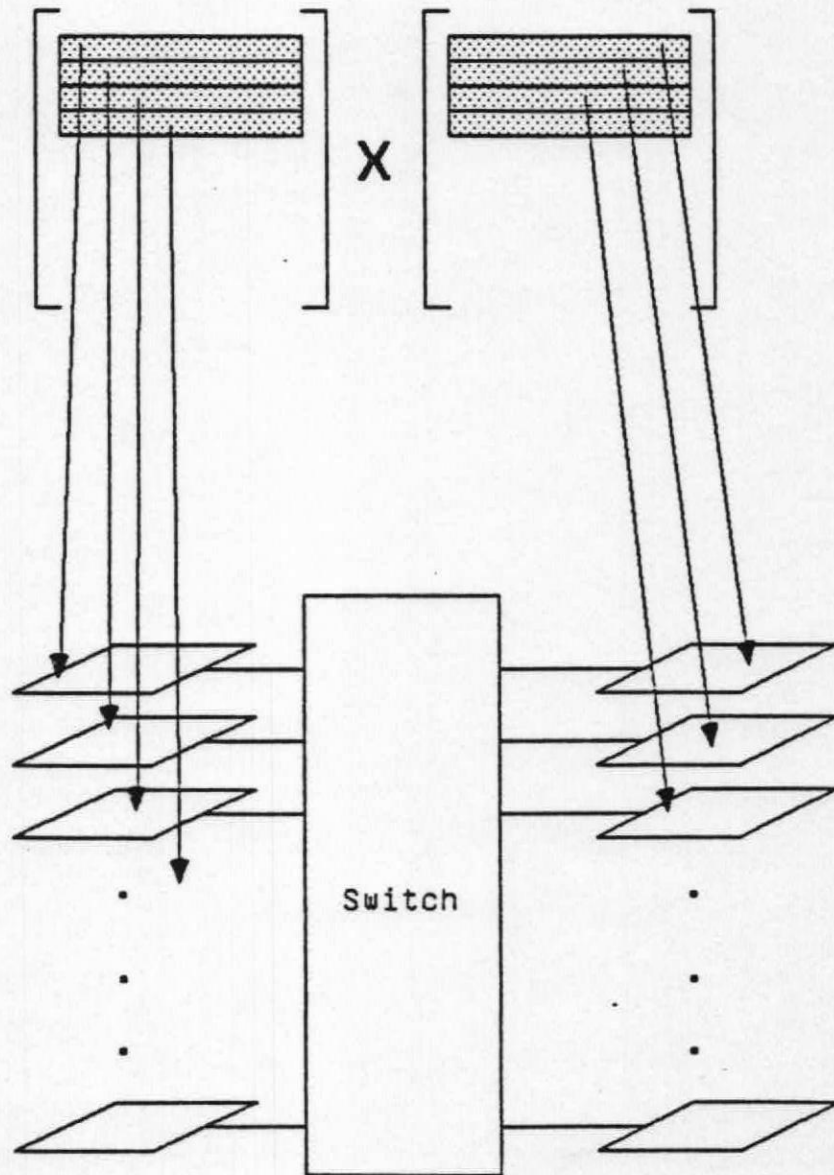
Region of interest histograms.

Matrix Multiplication



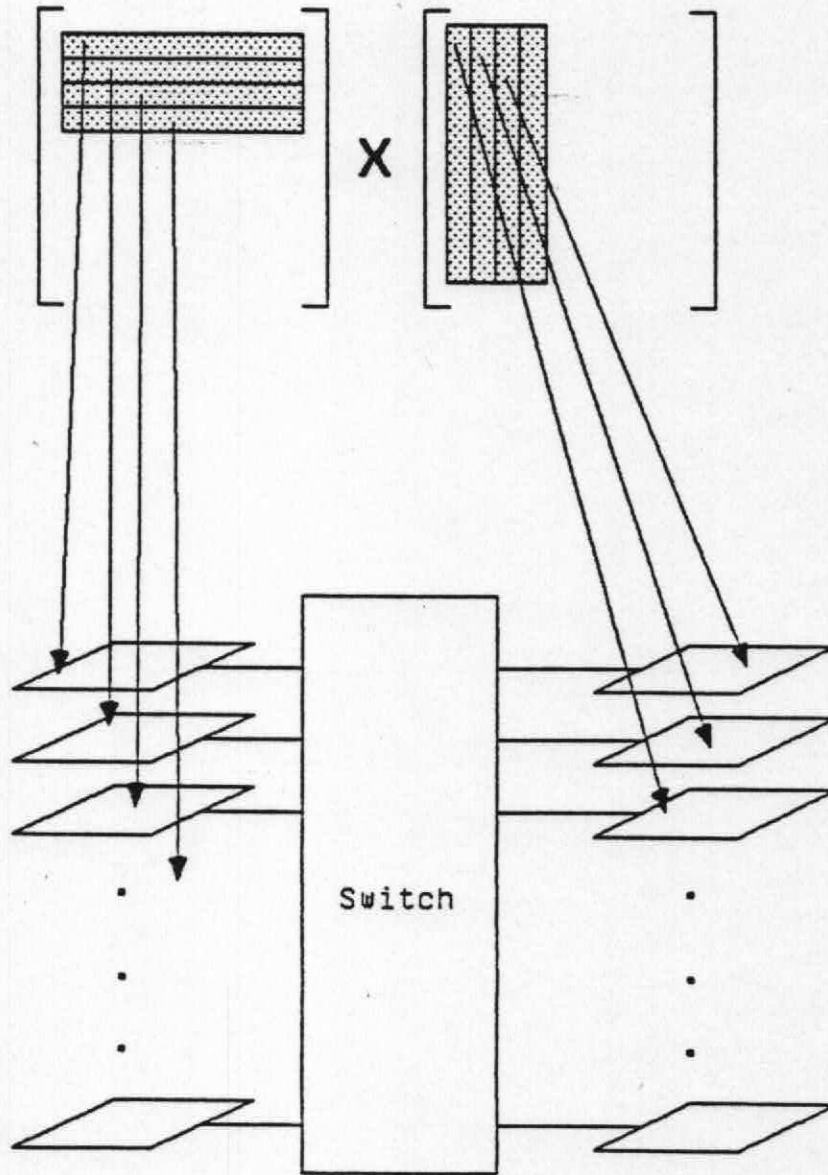
Matrix Multiplication

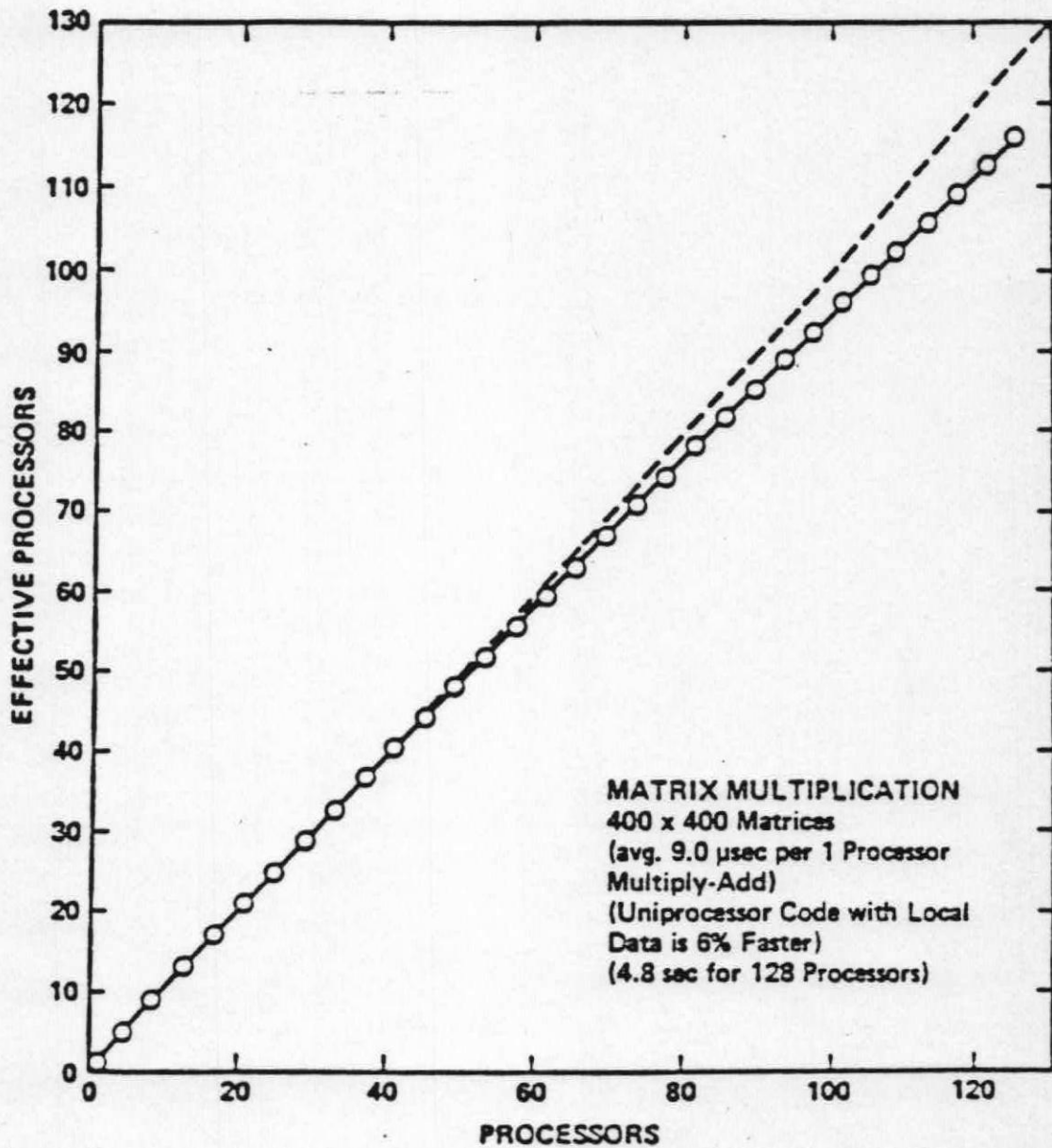
(storage allocation)



Matrix Multiplication

(storage allocation :
an optimization)





Gaussian Elimination

$$\begin{bmatrix} A_{11} & A_{12} & \dots & A_{1n} \\ A_{21} & A_{22} & \dots & A_{2n} \\ & & \ddots & \\ & & & A_{nn} \end{bmatrix} \mathbf{X} = \begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \cdot \\ \cdot \\ \cdot \\ b_n \end{bmatrix}$$

↓
Eliminate
↓

$$\begin{bmatrix} C_{11} & C_{12} & \dots & C_{1n} \\ & C_{22} & \dots & C_{2n} \\ & & \ddots & \\ & & & C_{nn} \end{bmatrix} \mathbf{X} = \begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_n \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \\ \cdot \\ \cdot \\ \cdot \\ d_n \end{bmatrix}$$

↓
Backsubstitute
↓

$$\begin{aligned} x_n &= d_n / C_{nn} \\ x_{n-1} &= (d_{n-1} - \dots) / C_{n-1, n-1} \\ &\vdots \\ &\vdots \end{aligned}$$

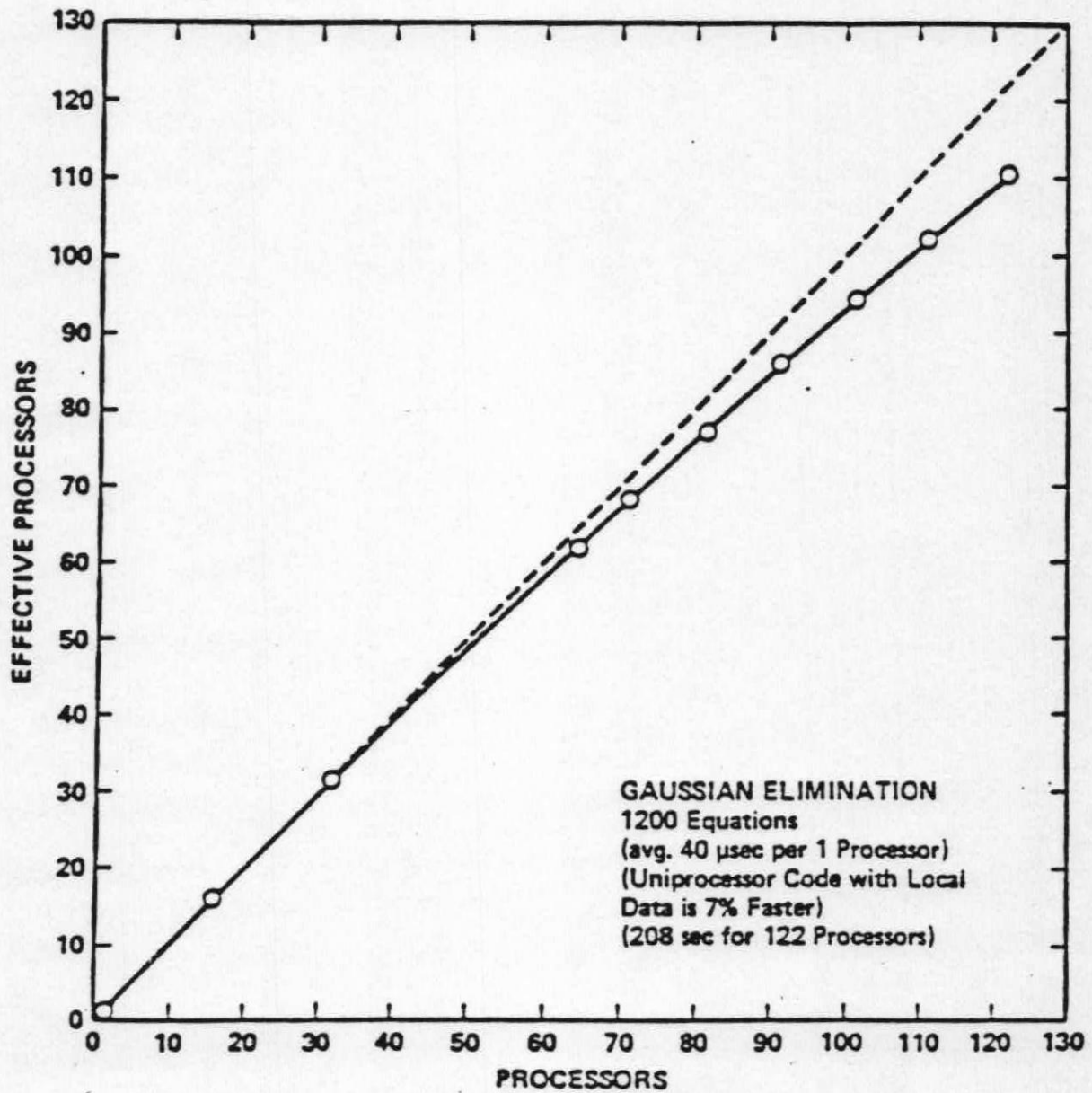
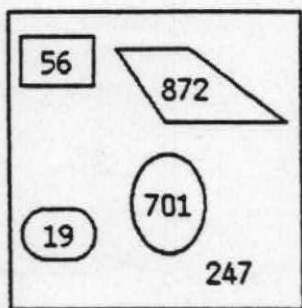
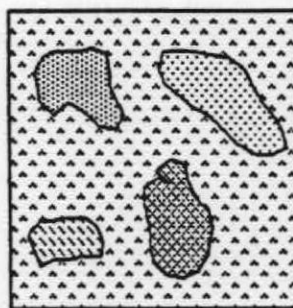


Image Utilities

Region Relabeling

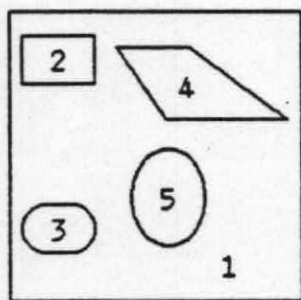


Region Map

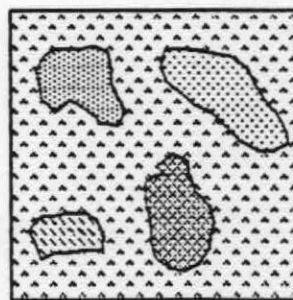


Image

↓
Relabel
↓



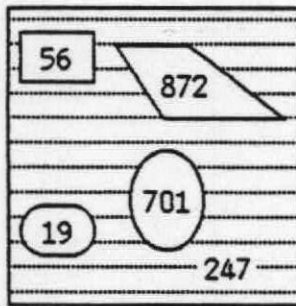
Region Map



Image

Image Utilities

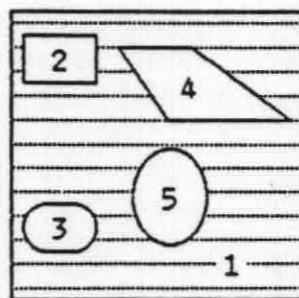
Region Relabeling



Region Map

19	3
56	2
247	1
701	5
872	4

Relabel Table



Region Map

Relabel 4

Effective Processors

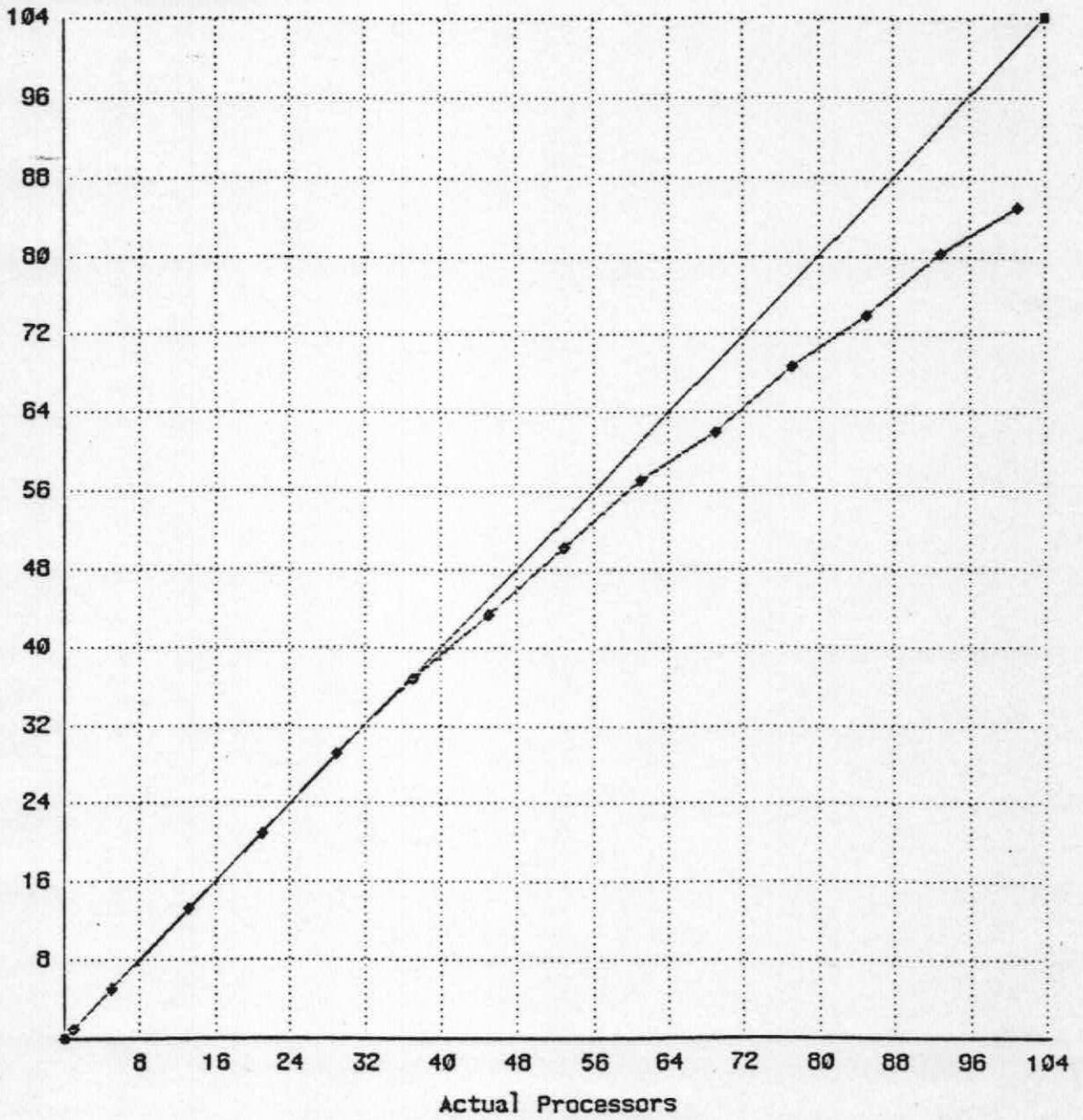
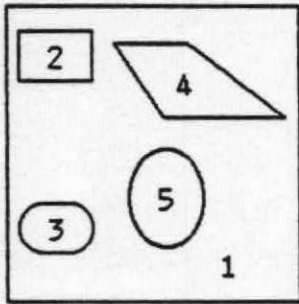
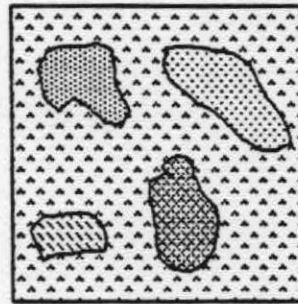


Image Utilities

Region Histograms



Region Map



Image

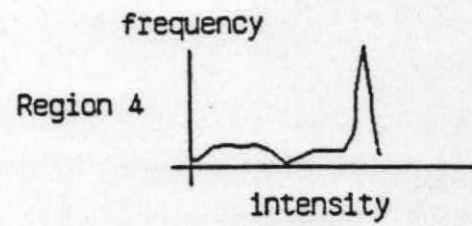
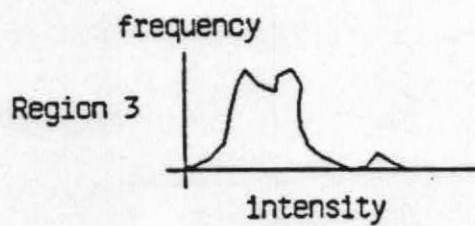
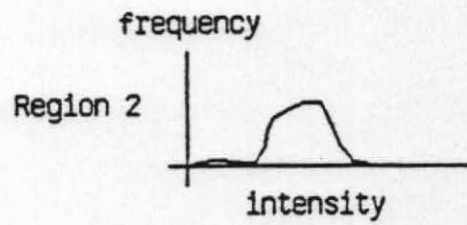
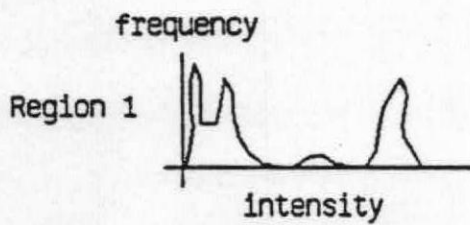
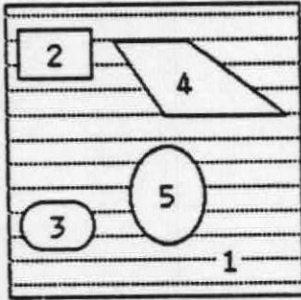
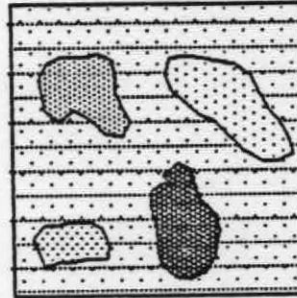


Image Utilities

Region Histograms



Region Map



Image

histogram 1
histogram 2
histogram 3
...
histogram n

Roihist 4

Effective Processors

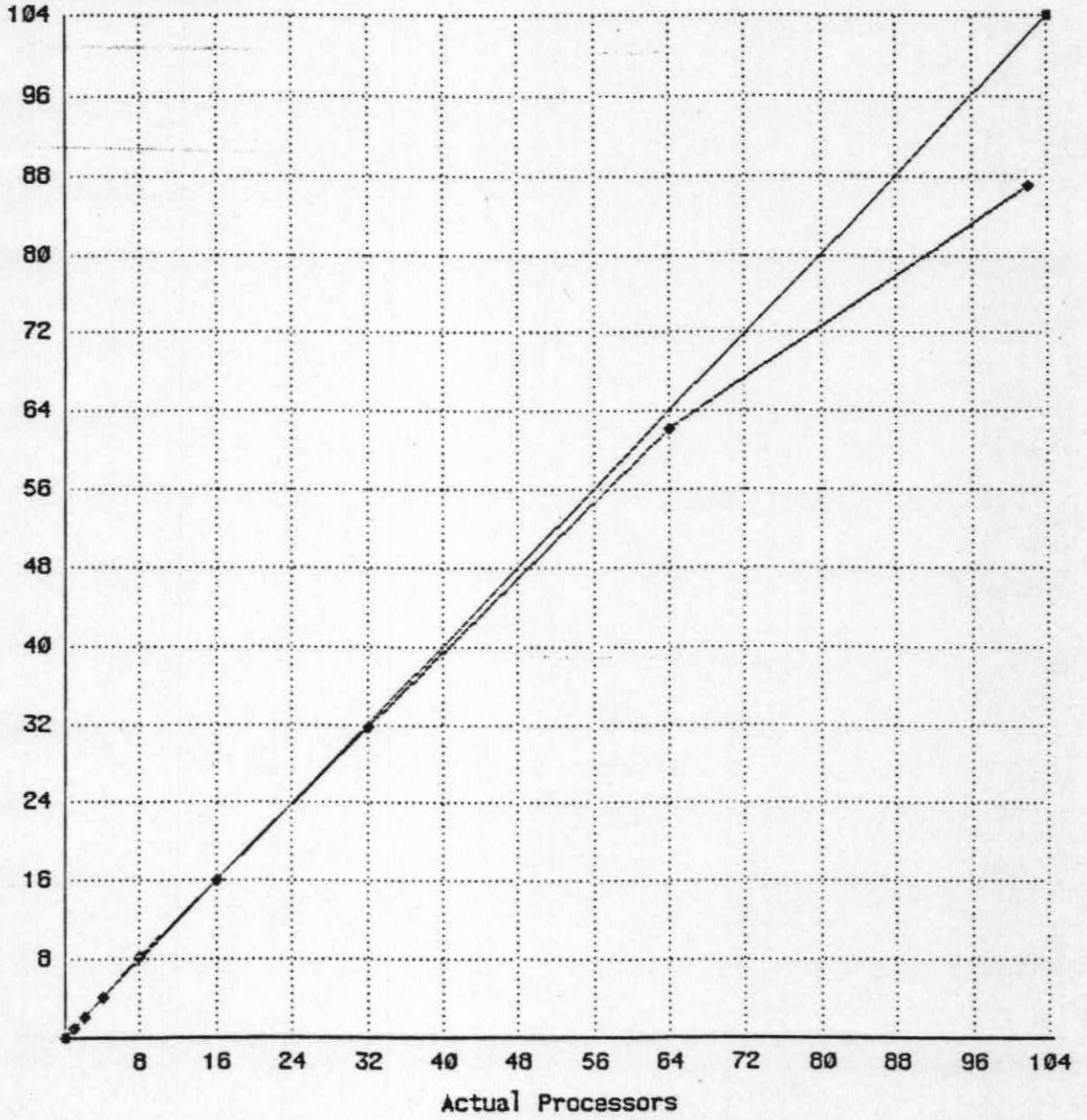
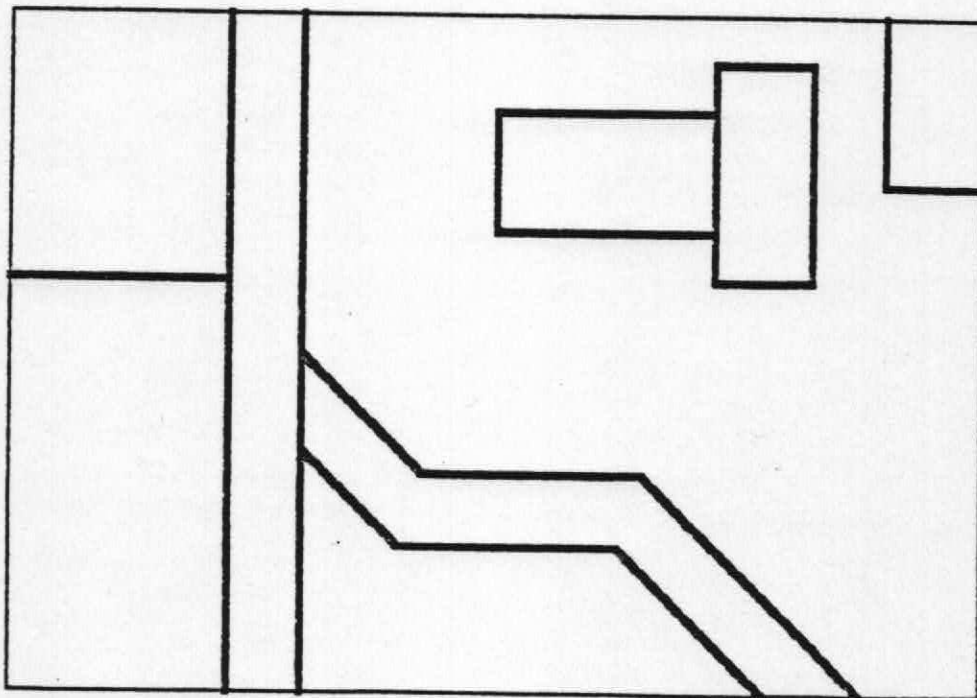
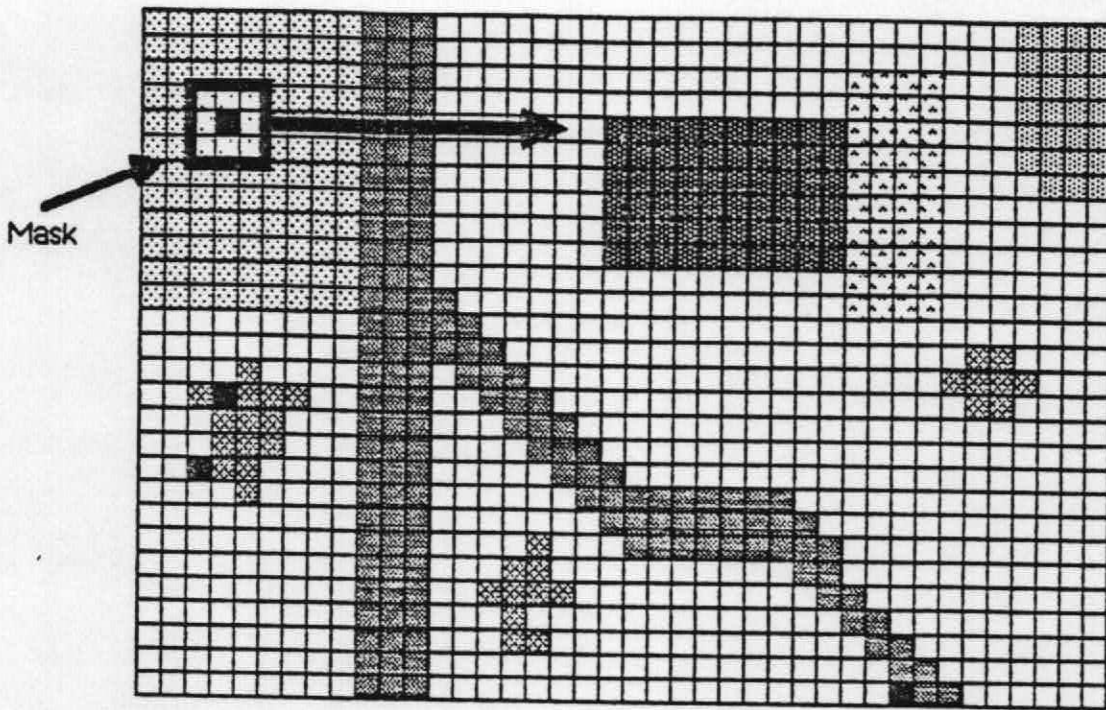


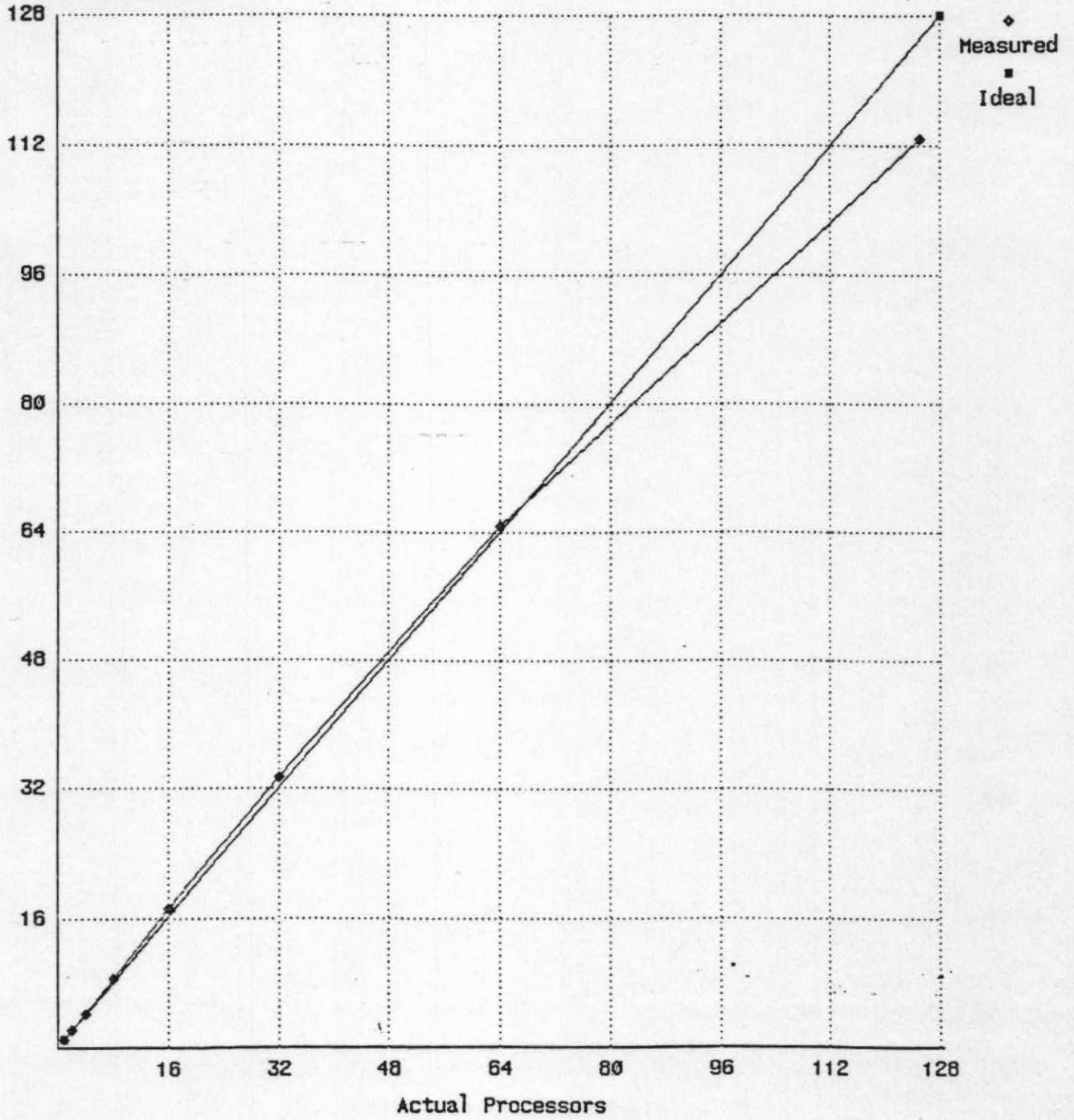
Image Utilities

Convolution



Convolution of 1024 x 1024 Image with 3 x 3 Mask

Effective Processors



Kevin McAuliffe
IBM Corporation
T. J. Watson Research Center
P.O. Box 218
Yorktown Heights, NY, 10598
914-789-7240
kpmac.yktvmh.ibm@csnet

John W. Middleton
Denelcor Inc.
17000 East Ohio Place
Aurora, CO 80017
303-337-7900

Alan Mink
NBS
Building 223 Room B364
Gaithersburg, MD 20899
301-921-3274

Harry Nelson
Lawrence Livermore Laboratory
LLNL L-73
P.O. Box 808
Livermore, CA 94550
415-422-4204

Rod R. Oldehoeft
Colorado State University
Computer Science Department
Fort Collins, CO 80523
303-491-7017
rro.ColoState@CSnet-relay

John Riganati
NBS
Building 225 Room A221
Gaithersburg, MD 20899
301-921-3427

John W. Roberts
NBS
Building 223 Room B364
Gaithersburg, MD 20899
301-921-3274
roberts@nbs-vms

Sandra B. Salazar
NBS
Building 223 Room B364
Gaithersburg, MD 20899
301-921-3274
salazar@nbs-vms

Paul Schneck
NSA
Supercomputer Research Center
6411 Ivy Lane
Greenbelt, MD 20770
982-0990

Herb Schwetman
MCC Corporation
9430 Research Boulevard
Echelon #1, Suite 200
Austin, TX 78759
512-834-3405

Zary Segall
Carnegie-Mellon University
Computer Science Department
Pittsburgh, PA 15208
412-578-3736
segall@cmu-cs-a

Carl Smith
NBS
Building 223 Room B364
Gaithersburg, MD 20899
301-921-3274
smith@nbs-vms

Steve Squires
DARPA
1400 Wilson Boulevard
Arlington, VA 22209
694-5917

Robert H. Thomas
BBN Laboratories
10 Moulton Street
Cambridge, MA 02138
617-497-3483
bthomas@bbn

Michael M. Tsao
IBM Corporation
T. J. Watson Research Center
P.O. Box 218
Yorktown Heights, NY 10598
914-789-7162
mtsao.yktumh@ibm

Leonard Uhr
University of Wisconsin
Computer Science Department
1210 W. Dayton Street
Madison, WI 53706
608-262-1204
uhr@uwisc

Walt Vandevender
Sandia National Laboratory
Division 2646
Albuquerque, NM 87185
505-844-4802

Robert G. Voigt
NASA/ICASE
MS 132-C
NASA Langley Research Center
Hampton, VA 23185
804-865-2513
RGV%ICASE@CSNET-RELAY