2019 IEEE Distinguished Lecture at Santa Clara Valley Section

Algorithm/Architecture Co-design for Smart Signals and Systems in Cognitive Cloud/Edge

Chris Gwo Giun Lee, PhD
Director, Bioinfotronics Research Center,
Professor, Department of Electrical Engineering,
National Cheng Kung University
Tainan, Taiwan

Outline

- Introduction
- Analytics Architecture: Abstraction at the System Level
- Algorithm Architecture Co-Design Space Exploration via Machine Learning
    - Algorithmic Intrinsic Complexity Metrics and Assessment
    - Intelligent Parallel/Reconfigurable Computing
- Case studies
    - Multimedia: MPEG
    - Mobile Health: Reconfigurable CNN

Introduction

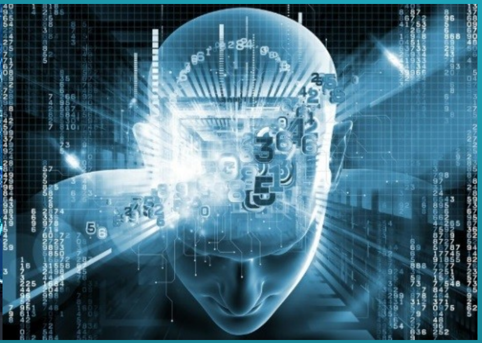
Vibrant & Fast Changing World

Industry 1.0: Energy

Industry 2.0: Electricity

Industry 3.0: Information

Industry 4.0: AI



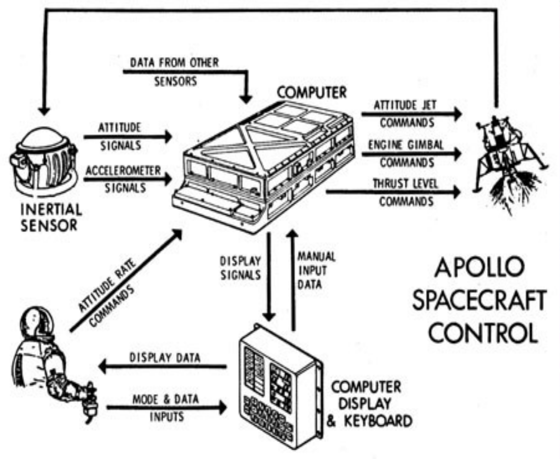
Before Industrial Revolution:

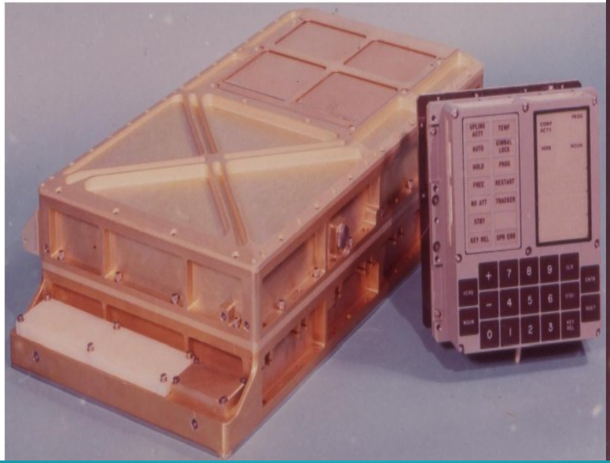
- Innovations in **ENERGY** and **ELECTRICITY** brought forth automation
- Revolutionary changes to traditional Artisan craftsmanship from the social, political, and economical perspectives.

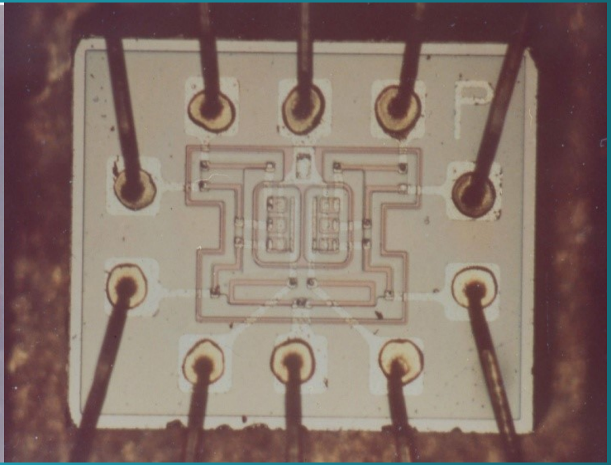
Today:

- **INFORMATION** explosion resulting in **BIG DAT**
- McKinsey forecasted on changes by **AI** to be 10 times faster and 300 times larger in scale as compared to former Industrial Revolution!

Apollo Navigation Computer Half a Century Ago

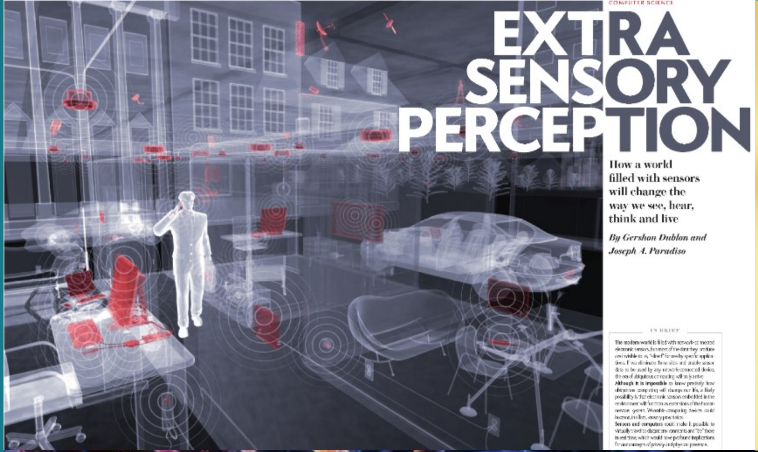






“That's one small step for an engineer; one giant leap for engineering.”

Reaching Out Even Further via IoT & Going in Ever Deeper.. Ubiquitous Computers



COMPUTER SCIENCE

EXTRA SENSORY PERCEPTION

How a world filled with sensors will change the way we see, hear, think and live

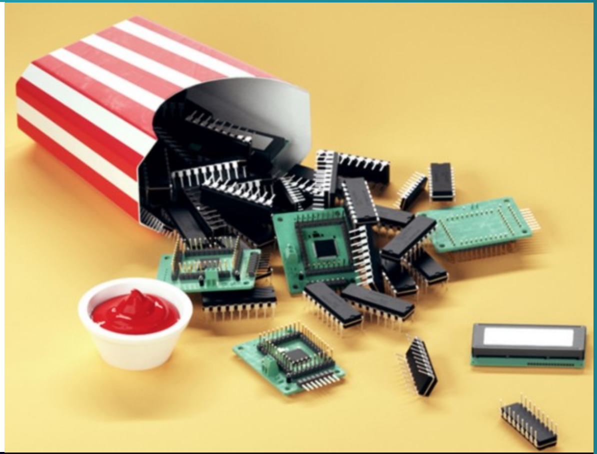
By Gershon Dublon and Joseph A. Paradiso

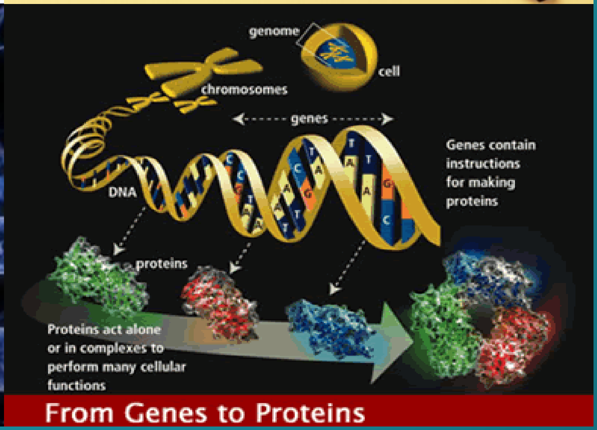
IN BRIEF

The modern world is filled with network-connected electronic sensors, but most of the data they produce are invisible to us, "siloed" for use by specific applications. If we eliminate those silos and enable sensor data to be used by any network-connected device, the era of ubiquitous computing will truly arrive.

Although it is impossible to know precisely how ubiquitous computing will change our lives, a likely possibility is that electronic sensors embedded in the environment will function as extensions of the human nervous system. Wearable computing devices could become, in effect, our sensory prosthetics.

Sensors and computers could make it possible to virtually travel to distant environments and "be" there in real time, which would have profound implications for our concepts of privacy and physical presence.





genome
cell
chromosomes
genes
DNA
Genes contain instructions for making proteins
proteins
Proteins act alone or in complexes to perform many cellular functions

From Genes to Proteins

Ever More Complex Analytics Algorithms Should Run on Analytics Architecture

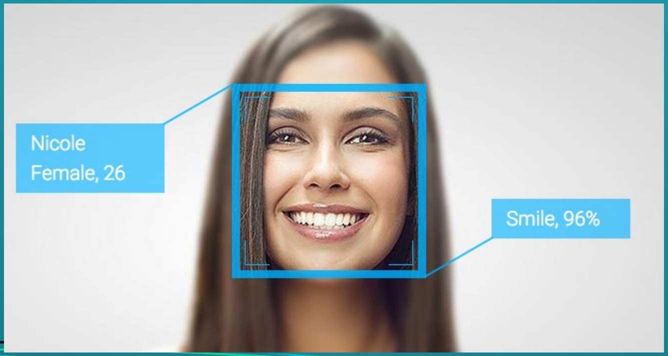
Analytics Algorithm:
Analyzes speech & images

Analytics Architecture:
Analyzes dataflow

TECHNOLOGY
ROBOTS WITH HEART
Before we can share our lives with machines, we must teach them to understand and mimic human emotion
By Pascale Fung

Speech recognition with feelings

Facial emotion detection



Algorithm/Architecture Co-Design: Analytics Architecture for SMART SoC

New Design Paradigm: Moving from programming to design and beyond…

Wirth from ETHZ (1975):

Programming = Algorithm + Data Structure

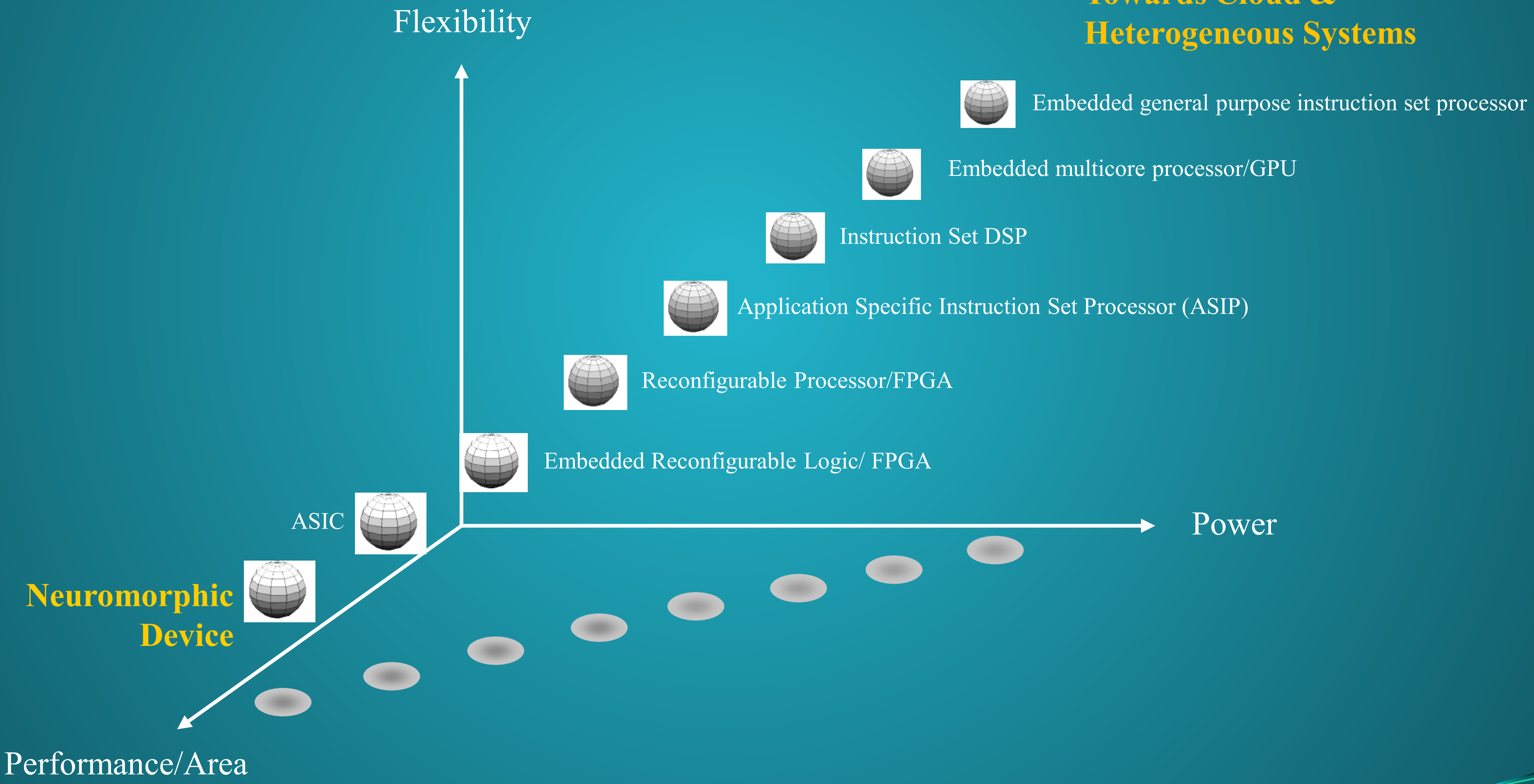
⬇

Lee from NCKU (2007):

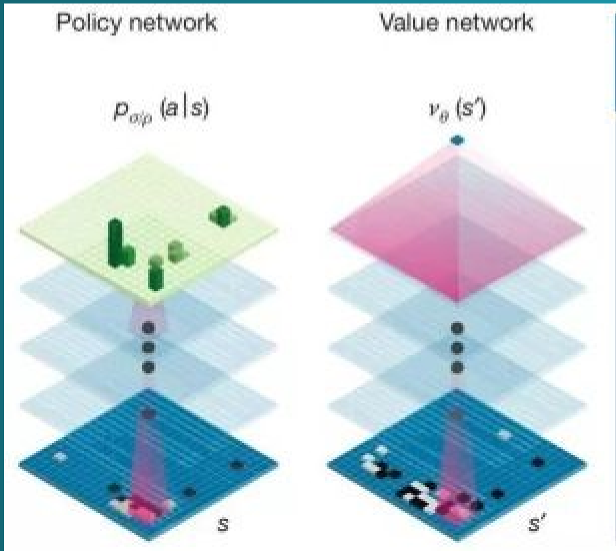
Design = Algorithm + Architecture

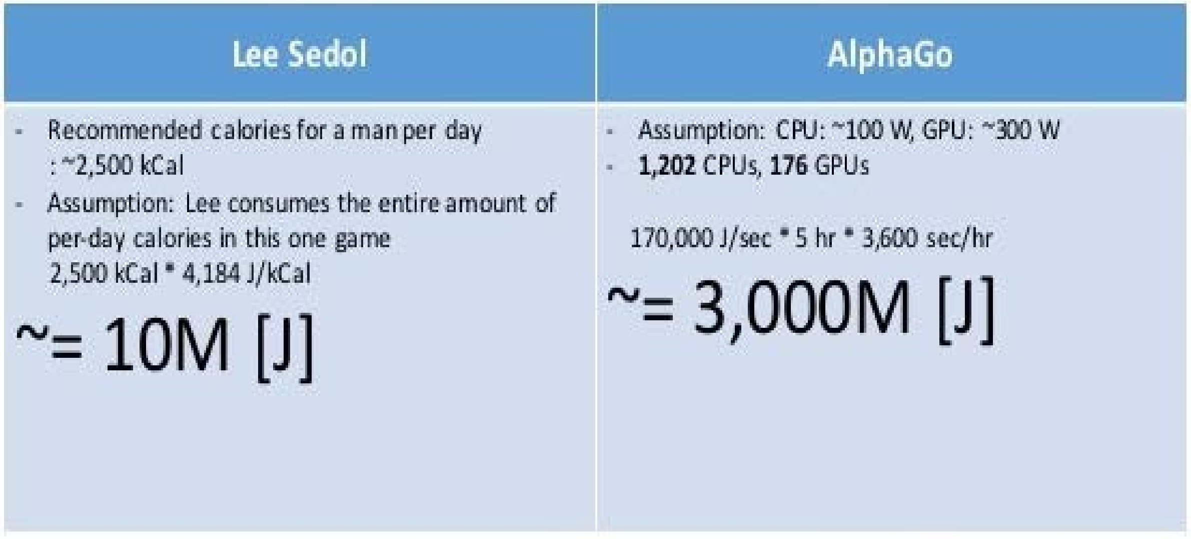
Architectural Platforms Beyond Cloud…Post Moore's law

Towards Cloud & Heterogeneous Systems


Flexibility
Embedded general purpose instruction set processor
Embedded multicore processor/GPU
Instruction Set DSP
Application Specific Instruction Set Processor (ASIP)
Reconfigurable Processor/FPGA
Embedded Reconfigurable Logic/ FPGA
ASIC
Power
Neuromorphic Device
Performance/Area

Human Brain: THE Most Power Efficient Intelligence





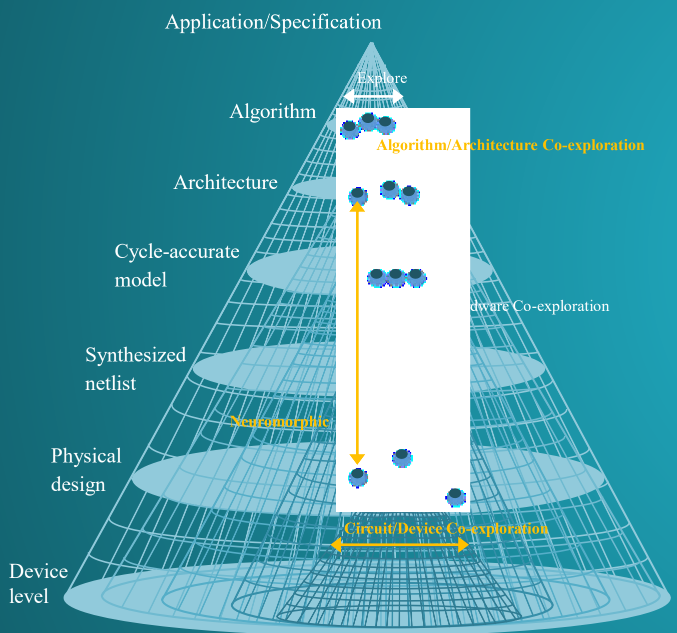
Lee Sedol	AlphaGo
- Recommended calories for a man per day : ~2,500 kCal - Assumption: Lee consumes the entire amount of per-day calories in this one game 2,500 kCal * 4,184 J/kCal ~= 10M [J]	- Assumption: CPU: ~100 W, GPU: ~300 W - **1,202** CPUs, **176** GPUs 170,000 J/sec * 5 hr * 3,600 sec/hr ~= 3,000M [J]

AlphaGo 13 Layer CNN

Human is 300x more power efficient

https://www.slideshare.net/ShaneSeungwhanMoon/h ow-alphago-works

Design Space w/
Different Levels of Abstraction



Different instances or realizations

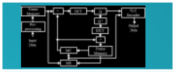
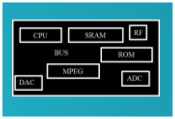
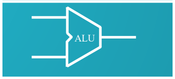
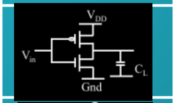
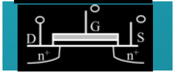
Application

System Architecture

Microarchitecture

Circuit

Device

Levels	Symbols	Features	Time units	Modeling Tool
Algorithm		System functionality	Seconds (sec)	R, Matlab, Python, C/C++
Architecture		System architecture: No. of operations, data transfer rate, data storage, degree of parallelism	No. of cycles	SystemC, CAL, DIF, LIDE
IP(Macro)	Motion estimator	IP functionality	No. of cycles	Verilog, VHDL
Module		Arithmetic/Logic operation	Cycle	Verilog, VHDL
Gate		Logic operation, **Timing Delay**	Nanosecond (ns)	Verilog, VHDL
Circuit		Voltage, current, resistance, capacitance, inductance	Picosecond (ps)	SPICE
Device		Electron	Picosecond (ps)	PyNN, PyNCS, Corelet

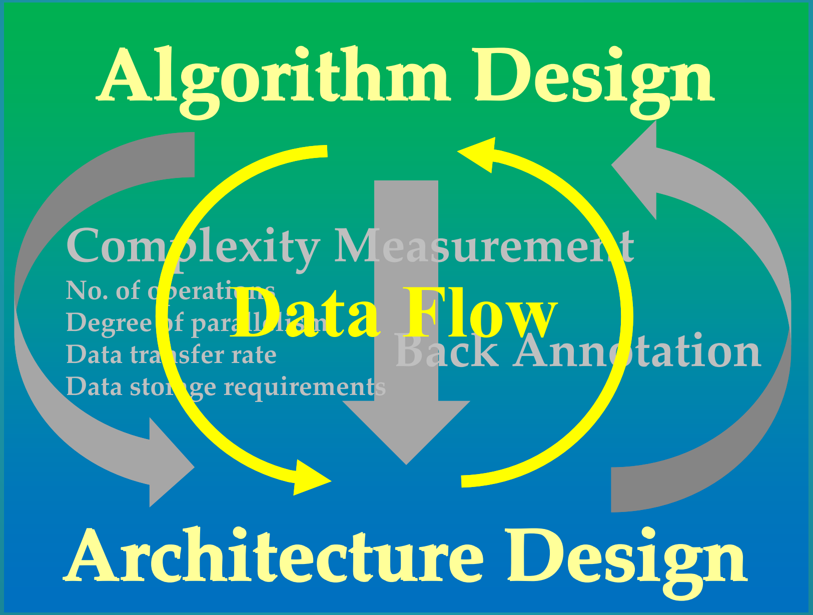
Traditionally Software/Hardware Co-design

Current Algorithm/Architecture Co-exploration for yet larger systems but how?

Algorithm/Architecture Co-Design:
Cross Level Abstraction at the System Level

Algorithm/Architecture Co-Exploration

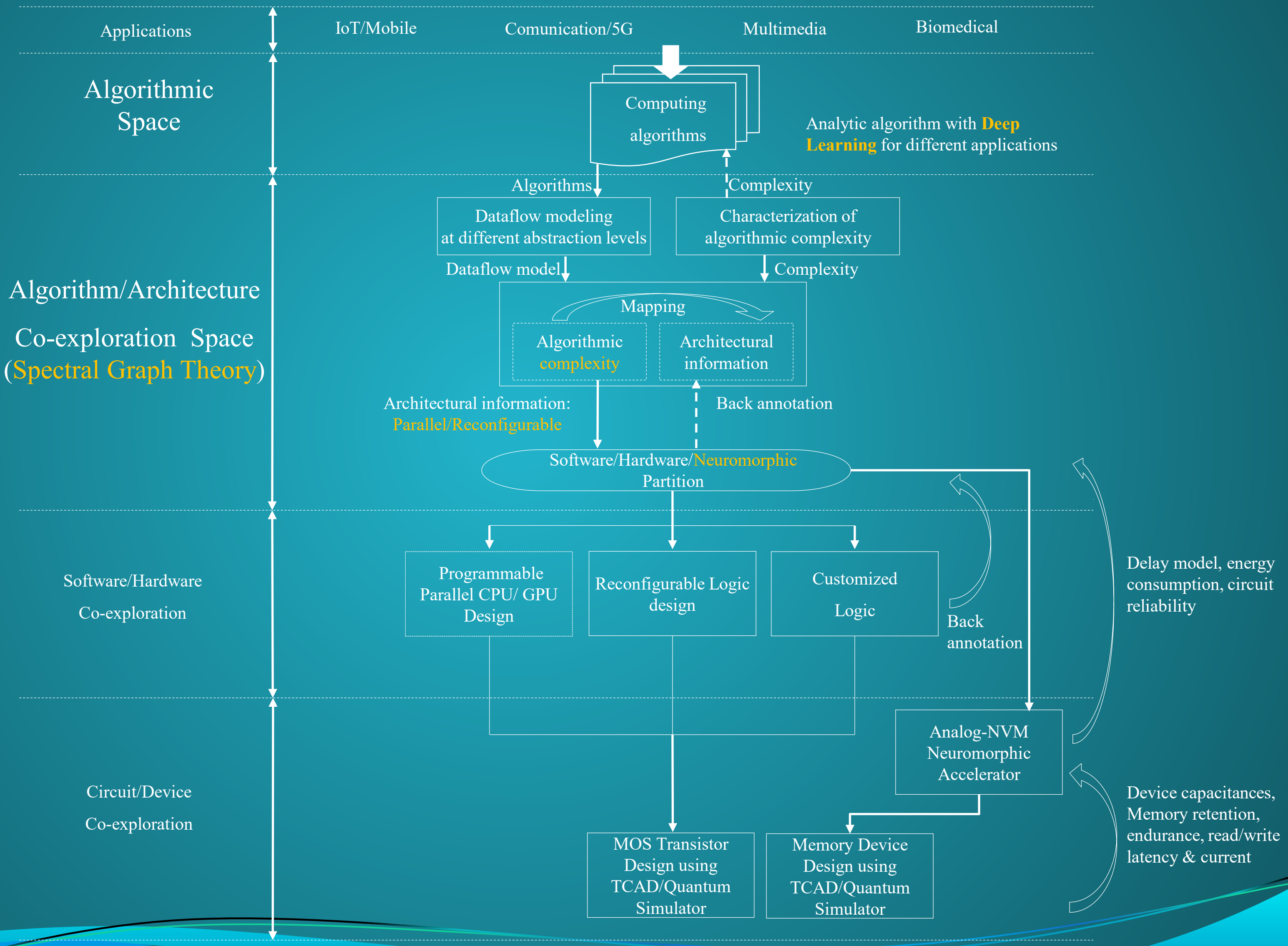
Traditional design flow



- Know software/hardware ingredients in early design phase hence from top level
- Extract complexity features from dataflow graph models

G. G. (Chris) Lee, Y.-K. Chen, M. Mattavelli, and E. S. Jang, "Algorithm/Architecture Co-Exploration of Visual Computing: Overview and Future Perspectives," *IEEE Trans. on Circuits and Systems for Video Technology*. Vol. 19, Iss. 11, pp. 1576-1587, Nov. 2009.

Exploring Algorithm/Architecture Co-Design Space Via Machine Learning/Spectral Graph Theory



Applications

IoT/Mobile

Comunication/5G

Multimedia

Biomedical

Algorithmic Space

Computing algorithms

Analytic algorithm with **Deep Learning** for different applications

Algorithms

Complexity

Algorithm/Architecture Co-exploration Space (Spectral Graph Theory)

Dataflow modeling at different abstraction levels

Characterization of algorithmic complexity

Dataflow model

Complexity

Mapping

Algorithmic complexity

Architectural information

Architectural information: Parallel/Reconfigurable

Back annotation

Software/Hardware/Neuromorphic Partition

Software/Hardware Co-exploration

Programmable Parallel CPU/ GPU Design

Reconfigurable Logic design

Customized Logic

Back annotation

Delay model, energy consumption, circuit reliability

Circuit/Device Co-exploration

Analog-NVM Neuromorphic Accelerator

MOS Transistor Design using TCAD/Quantum Simulator

Memory Device Design using TCAD/Quantum Simulator

Device capacitances, Memory retention, endurance, read/write latency & current

MODELING the COMPUTATIONAL PLATFORM

via

Dataflow Graphs (DFG)

Dataflow…

Dataflow Graph Modelling Computational Platform @ Various Data Granularity

- They should contain:
  - Algorithmic information or behavior
  - Architectural information (Software/Hardware) for implementation
- Some important dataflow models are:
  - Directed acyclic graph (DAG)
  - Synchronous dataflow (SDF) graph
  - Control data flow graph (CDFG)
  - Kahn process networks (KPN)
  - Y-chart application programming interface (YAPI)

Algorithm/Architecture Co-Design Space Exploration

Via

Machine Learning

How Big is Big?

Algorithmic Intrinsic Complexity Metrices/Features

PLATFORM INDEPENDENCE

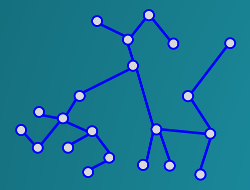
Number of Operations

- Estimates the number of each type of operations
  - Addition/Subtraction
  - Multiplication
  - Division
  - Shift
  - Logic operations
- Operations with constant input and variable input should be differentiated to provide high accuracy
  - X + Y vs. X + 5 (X and Y are variables)
  - X×Y vs. X×5 (X and Y are variables)
- In addition, the precision of each operand should be taken into account, since it can significantly influence complexity

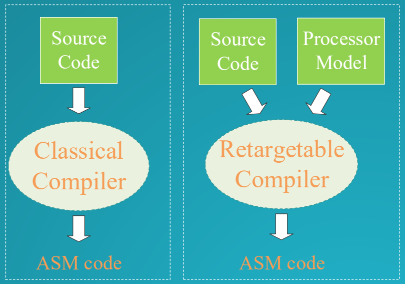
SMART TRANSFORM PAIR
via
Spectral Graph Theory (SGT)
for

Intelligent Parallel/Reconfigurable Computing

Parallel Computing (Forward Transform): Efficient & Flexible Cognitive Cloud



(a)



Source Code

Classical Compiler

ASM code

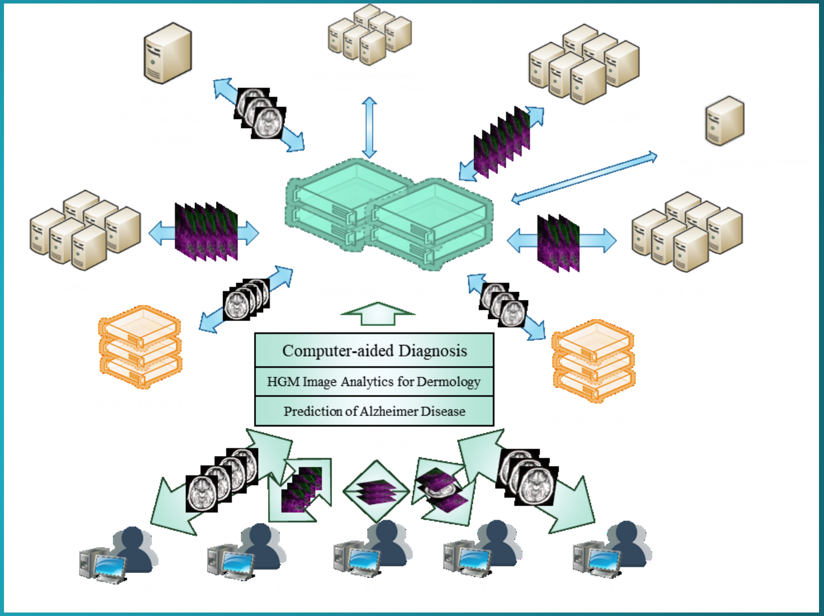
Source Code

Processor Model

Retargetable Compiler

ASM code

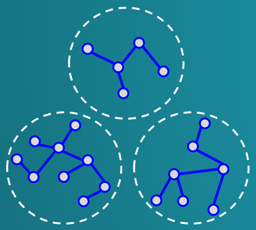
(c) Retargetable Compiler



Computer-aided Diagnosis

HGM Image Analytics for Dermology

Prediction of Alzheimer Disease



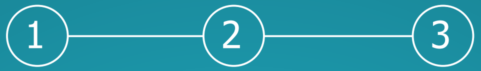
(b)

(d) Stream Processor within Nvidia GPU

- Using SGT as machine learning in exploring the AAC space:
  - Connected component are eigen-decomposed where spectrum of unconnected graph components serves as information or features extracted
  - decision making performed via the bi-partite or k-partitioning based on principle axis theorem optimized for data independency

Spectral Graph Theory

Graph



Adjacency matrix **A**

$$\mathbf{A}(\mathrm{i,j}) = \begin{cases} 1 & \text{if vertex}_{i} \text{ and vertix}_{j} \text{ are adjacent to each other} \\ 0 & \text{otherwise} \end{cases} \Rightarrow \quad \mathbf{A} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

Laplacian matrix **L=D-A**, where **D** is **a** diagonal matrix where the diagonal elements represents the number of edges connected to that node.

$$\mathbf{L}(\mathrm{i,j}) = \begin{cases} \mathbf{D}(\mathrm{i,j}) & \text{if } i = j \\ \text{-1} & \text{if } i \neq j \text{ and vertex}_{i} \text{ is adjacent to vertix}_{j} \\ 0 & \text{otherwise} \end{cases} \Rightarrow \quad \mathbf{L} = \begin{bmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{bmatrix}$$

- Gwo Giun Lee, He-Yuan Lin, Chun-Fu Chen, Tsung-Yuan Huang, "Quantifying Intrinsic Parallelism Using Linear Algebra for Algorithm/Architecture Co-Exploration," IEEE Transactions on Parallel and Distributed Systems, vol. 23, iss. 5, pp. 944-957, May 2012
- Gwo-Giun Lee, He-Yuan Lin, "Method of analyzing intrinsic parallelism of algorithm," USA, Patent No. US8522224 B2, Aug. 27, 2013.
- Gwo-Giun Lee, Ming-Jiun Wang, He-Yuan Lin, "Method and Algorithm Analyzer for Determining a Design Framework," USA, Patent No. US8621414 B2, Dec. 31, 2013.
- (Boston, MA, June 1, 2015, GLOBE NEWSWIRE)

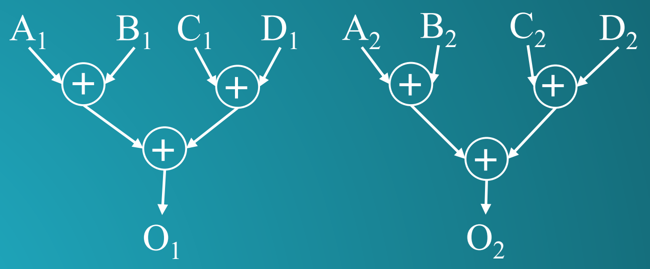
Degree of Parallelism: Eigen-Analysis of DFGs using SGT

Algorithm

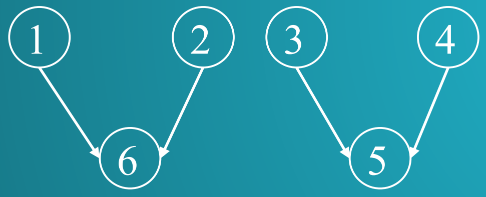
$O_1=A_1+B_1+C_1+D_1$

$O_2=A_2+B_2+C_2+D_2$

Dataflow diagram



Causation graph



Laplacian matrix

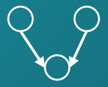
$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & -1 \\ 0 & 1 & 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & -1 & -1 & 2 & 0 \\ -1 & -1 & 0 & 0 & 0 & 2 \end{bmatrix}$$

Spectrum

Eigenvalue: 0, 0, 1, 1, 3, 3

Eigenvector:
$$\begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ -1 \\ 1 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} 1 \\ -1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \\ -2 \\ 0 \end{bmatrix} \begin{bmatrix} -1 \\ -1 \\ 0 \\ 0 \\ 0 \\ 2 \end{bmatrix}$$

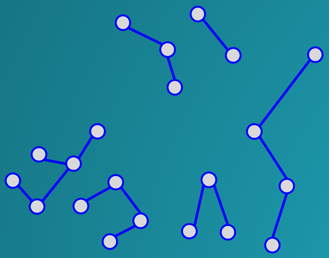
Parallelism

2 × 

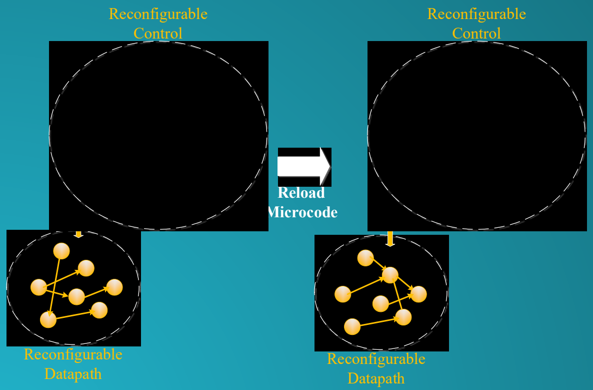
(Homogeneous)

Quantification of parallelization,
Instruction Set Architecture (ISA) design

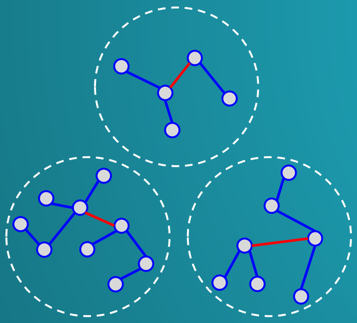
Reconfigurable Computing (Inverse Transform): Efficient & Flexible Mobile Edge



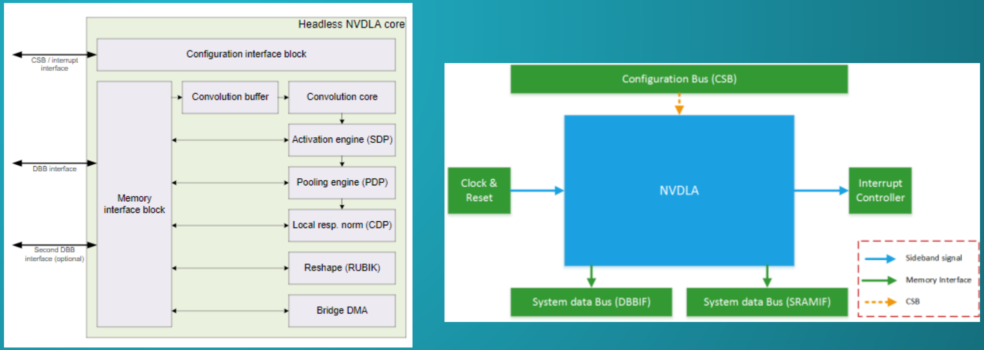
(a)



(c) Reconfigurable architecture



(b)



(d) Reconfigurable Architecture of NVDLA

- Commonalities are analyzed on DFGs for reuse when synthesizing or reconfiguring the CNN computational platform.
  - Introduce efficient flexible architecture with algorithmic convolution for CNN are eigen-transformed to matrix operations with higher symmetry

Reconfigurable Architecture: Commonality Extraction from DFGs

- Observe the common parts between *each dataflow graph*.

MPEG-4 filter coefficients

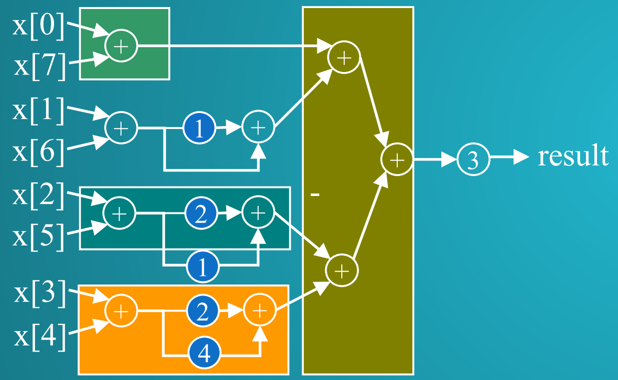
[-8,24,-48,160,160,-48,24,-8]

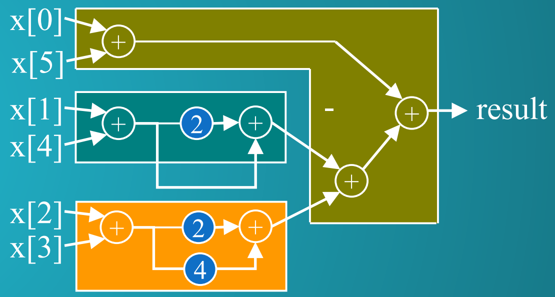
Divide by 8

[-1,3,-6,20,20,-6,3,-1]

AVC/H.264 filter coefficients

[1,-5,20,20,-5,1]

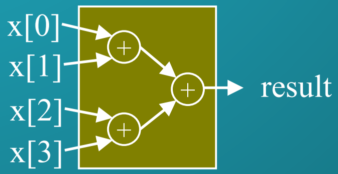


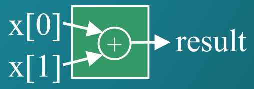


MPEG-4 Chroma interpolation, MPEG2, and AVC/H.264 chroma prediction

[1 1]

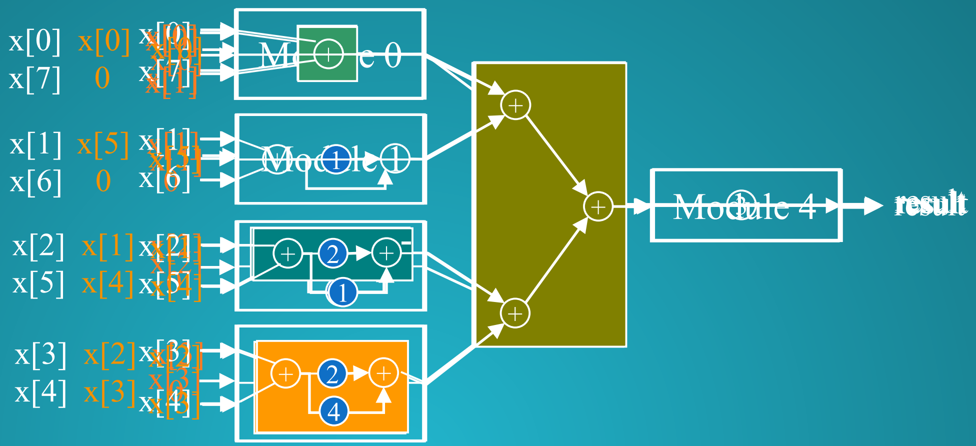
[1 1 1 1]

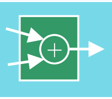
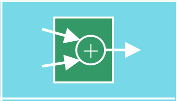
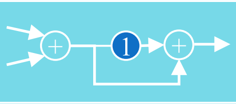
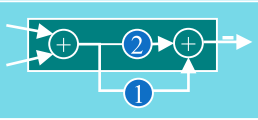
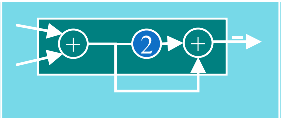
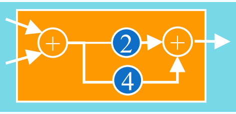
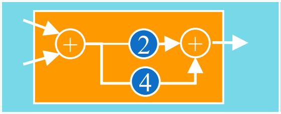
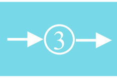




ⓘ : left shift by i ⓘ : right shift by i ⊕ : addition

Reconfigurable fractional interpolation



Coefficients	[-8,24,-48,160,160,-48,24,-8]	[1,-5,20,20,-5,1]	[1 1 1 1]	[1 1]
Module 0		→	→	
Module 1		→	→	→
Module 2			→	→
Module 3			→	→
Module 4		→	→	→

Four Symmetrical Patterns After PCA

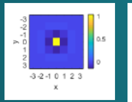
- Performed PCA to extract the commonality in Gabor filter.
- The transformed Gabor filter bank has four symmetry patterns. Their coefficient are illustrated in the following:

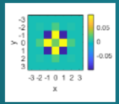
Pattern 1:

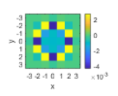
c_0	c_1	c_2	c_3	c_2	c_1	c_0
c_1	c_4	c_5	c_6	c_5	c_4	c_1
c_2	c_5	c_7	c_8	c_7	c_5	c_2
c_3	c_6	c_8	c_9	c_8	c_6	c_3
c_2	c_5	c_7	c_8	c_7	c_5	c_2
c_1	c_4	c_5	c_6	c_5	c_4	c_1
c_0	c_1	c_2	c_3	c_2	c_1	c_0

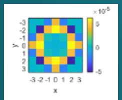
Coefficient:

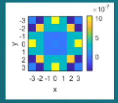
$[c_0, c_1, c_2, c_3, c_4, c_5, c_6, c_7, c_8, c_9]$


w_0


w_2


w_5


w_9

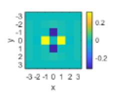

w_{13}

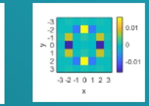
Pattern 2:

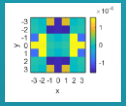
0	c_1	c_2	c_3	c_2	c_1	0
$-c_1$	0	c_5	c_6	c_5	0	$-c_1$
$-c_2$	$-c_5$	0	c_8	0	$-c_5$	$-c_2$
$-c_3$	$-c_6$	$-c_8$	0	$-c_8$	$-c_6$	$-c_3$
$-c_2$	$-c_5$	0	c_8	0	$-c_5$	$-c_2$
$-c_1$	0	c_5	c_6	c_5	0	$-c_1$
0	c_1	c_2	c_3	c_2	c_1	0

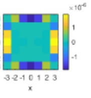
Coefficient:

$[c_1, c_2, c_3, c_5, c_6, c_8]$


w_1


w_4


w_8

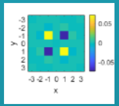

w_{12}

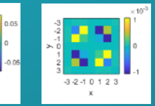
Pattern 3:

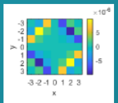
c_0	c_1	c_2	0	$-c_2$	$-c_1$	$-c_0$
c_1	c_4	c_5	0	$-c_5$	$-c_4$	$-c_1$
c_2	c_5	c_7	0	$-c_7$	$-c_5$	$-c_2$
0	0	0	0	0	0	0
$-c_2$	$-c_5$	$-c_7$	0	c_7	c_5	c_2
$-c_1$	$-c_4$	$-c_5$	0	c_5	c_4	c_1
$-c_0$	$-c_1$	$-c_2$	0	c_2	c_1	c_0

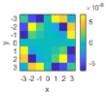
Coefficient:

$[c_0, c_1, c_2, c_4, c_5, c_7]$


w_3


w_7


w_{11}

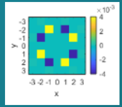

w_{15}

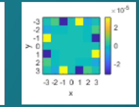
Pattern 4:

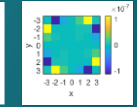
0	c_1	c_2	0	$-c_2$	$-c_1$	0
$-c_1$	0	c_5	0	$-c_5$	0	c_1
$-c_2$	$-c_5$	0	0	0	c_5	c_2
0	0	0	0	0	0	0
c_2	c_5	0	0	0	$-c_5$	$-c_2$
c_1	0	$-c_5$	0	c_5	0	$-c_1$
0	$-c_1$	$-c_2$	0	c_2	c_1	0

Coefficient:

$[c_1, c_2, c_5]$


w_6


w_{10}

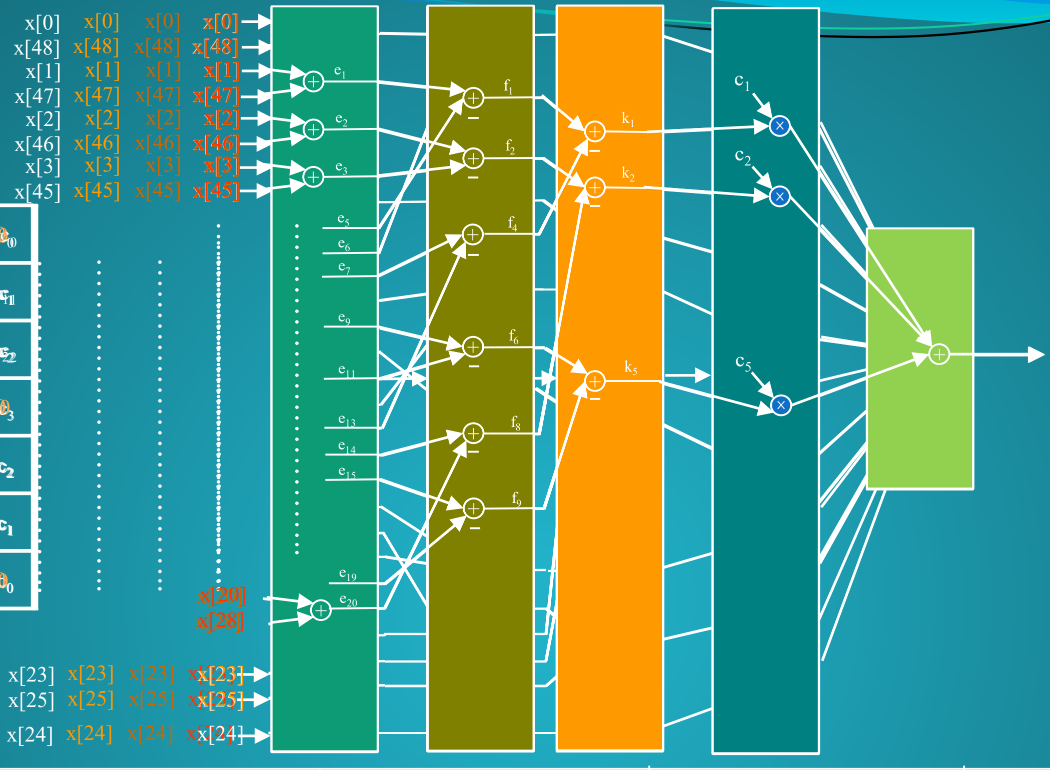

w_{14}

🟦 : Coefficient 0 : Zero Coefficient

□ : Repeated Coefficient

Reconfigurable Transformed Gabor Filter Bank

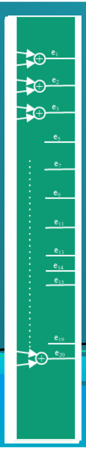
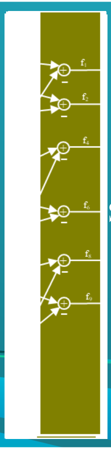
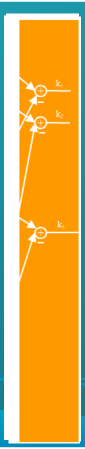
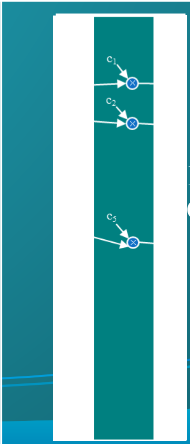
c_0	c_1	c_2	c_3	$-c_2$	$-c_1$	$-c_0$
$-c_1$	c_4	c_5	c_6	$-c_5$	$-c_4$	c_1
$-c_2$	$-c_5$	c_7	c_8	$-c_7$	c_5	c_2
$-c_3$	$-c_6$	$-c_8$	c_9	$-c_8$	$-c_6$	$-c_3$
c_2	c_5	$-c_7$	c_8	c_7	$-c_5$	$-c_2$
c_1	$-c_4$	$-c_5$	c_6	c_5	c_4	$-c_1$
$-c_0$	$-c_1$	$-c_2$	c_3	c_2	c_1	c_0



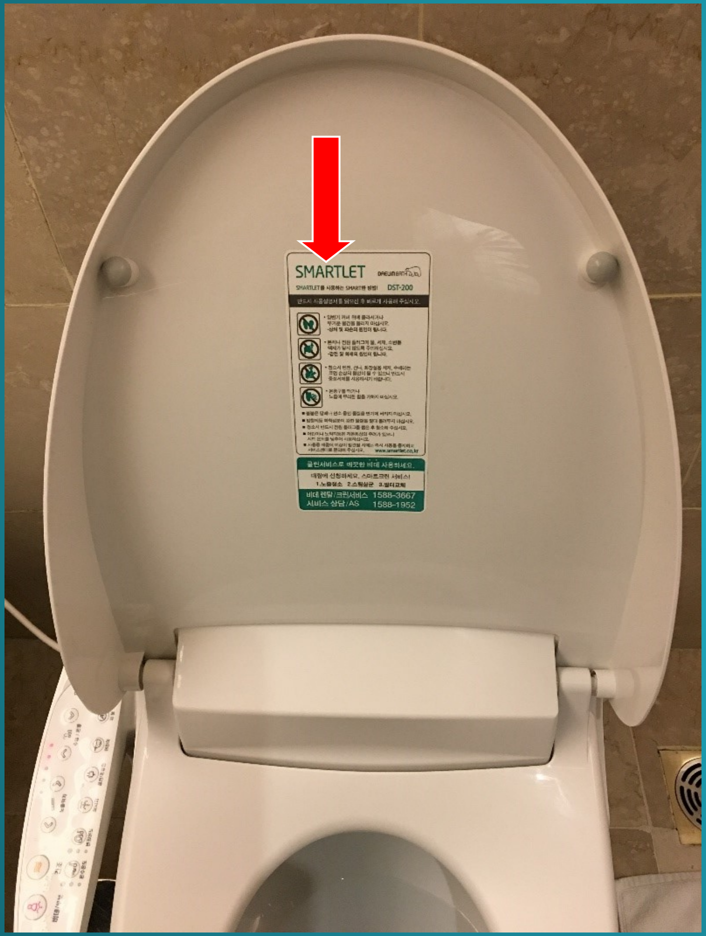
⊗ : multiplication

⊕ : addition

— : multiplied by -1

Coefficients	Module 0	Module 1	Module 2	Module 3
Pattern 1 [c_0, c_1, c_2, c_3, c_4, c_5, c_6, c_7, c_8, c_9]	 Even Point Symmetry	 Odd Vertical Symmetry	 Odd Diagonal Symmetry	 Multiplying Coefficients

A Very Useful SMART Sensor System



SMARTLET: SMART toiLET

SMART is a BUZZ word that SELLS

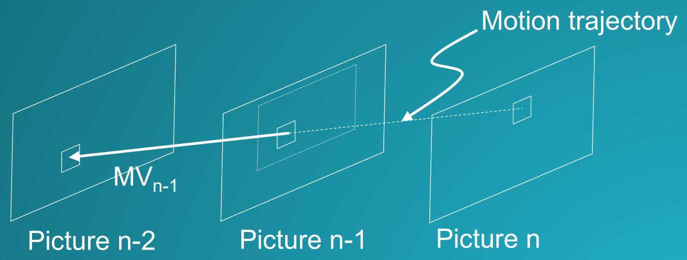
AAC Study Cases:

Observe and Learn from Nature in Engineering Innovations.

Multimedia

Algorithm/architecture co-design of spatial-temporal recursive motion estimator

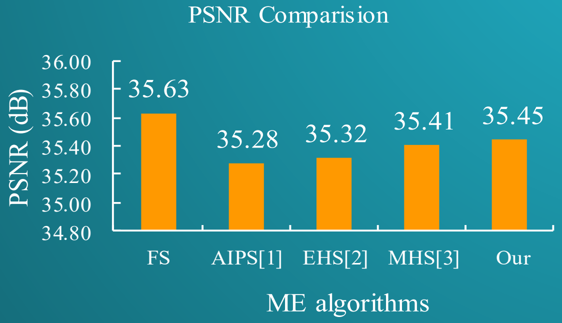
- Spatial-temporal recursive ME



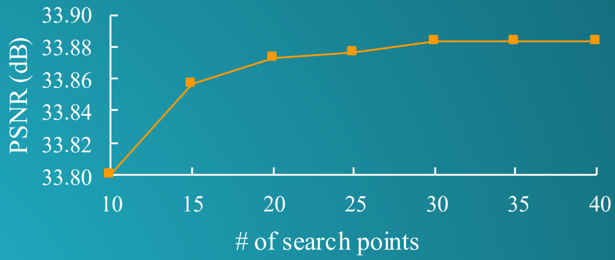
  - Initial candidates from spatial and temporal references followed by local search

    ➔ blocks cannot be processed in parallel

- Performance comparison



- Complexity vs. performance



# of search points	10	15	20	25	30	35	40
Clock rate (MHz)	54	81	108	135	162	189	216

Assumption: 16 processing elements performing accumulation of absolute difference with utilization 75%

- Architecture comparison

Terms	MHS[3]	Our
PSNR	35.41 dB	35.45 dB
Application	1920x1080 @ 30 FPS	1920x1080 @ 30 FPS
Search range	H: ± 128, V: ± 64	H: ± 128, V: ± 64
Technology	0.18 mm	0.18 mm
Clock rate	108 MHz	81 MHz
Total cell area	562.5 K gates	51.2 K gates

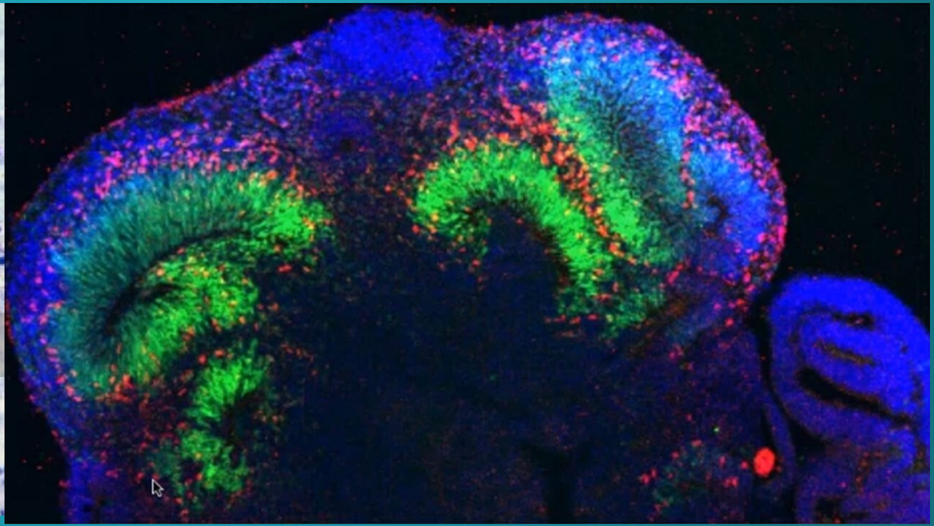
H: horizontal; V: vertical

[1] Y. Nie and K. Ma, “Adaptive irregular pattern search with matching prejudgment for fast block-matching motion Estimation,” IEEE Trans. Circuits Syst. Video Technol., vol. 15, no. 6, pp. 789–794, Jun. 2005.

[2] C. Zhu, X. Lin, L. Chau, and L. Po, “Enhanced hexagonal search for fast block motion estimation,” IEEE. Trans. Circuits Syst. Video Technol., vol. 14, no. 10, pp. 1210–1214, Oct. 2004.

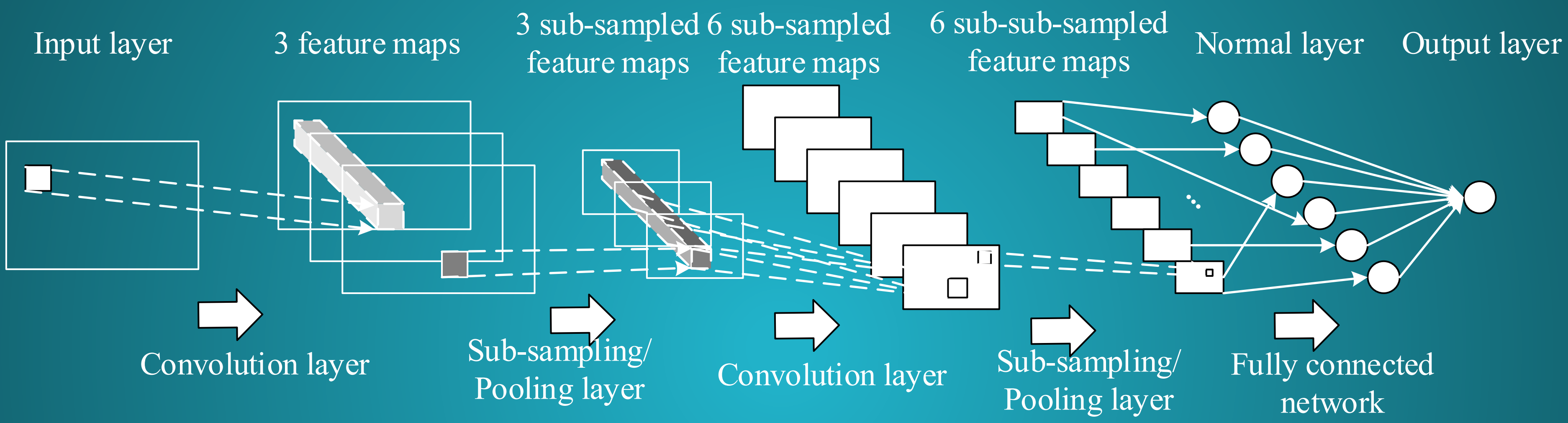
[3] Y. Murachi, K. Hamano, T. Matsuno, J. Miyakoshi, M. Miyama, and M. Yoshimoto, “A 95 mW MPEG2 MP@HL motion estimation processor core for portable high-resolution video application,” IEICE Trans. Fund. Electron. Commun. Comput. Sci., vol. E88-A, pp. 3492–3499, Dec. 2005.

Mobile Health:
Deep Learning



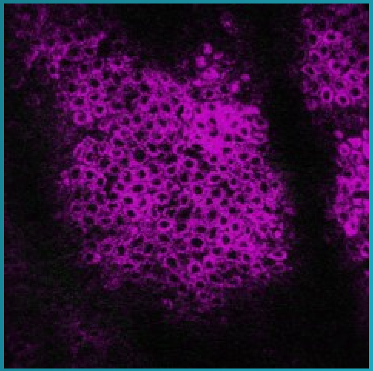
Self Organizing Cerebral Organoids by Madeline Lancaster

Convolutional Neural Network (CNN)

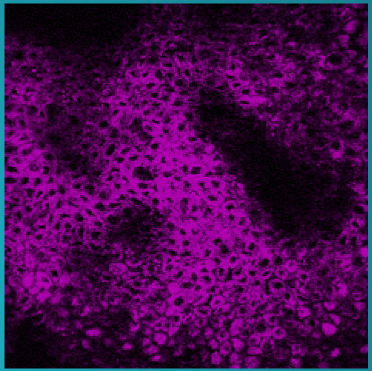

Input layer
3 feature maps
3 sub-sampled feature maps
6 sub-sampled feature maps
6 sub-sub-sampled feature maps
Normal layer
Output layer
Convolution layer
Sub-sampling/ Pooling layer
Convolution layer
Sub-sampling/ Pooling layer
Fully connected network

- Convolution layers for feature extraction & fully connected network as classifiers
- Feature layers updated or information mined from large amount of data via supervised learning
- Bayesian learning and Linsker’s self-organizing Kohonen feature map
- Convolution/feature layers constitute multiresolution pyramid like Azriel Rosenfeld?

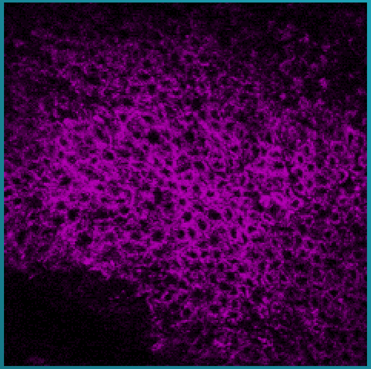
Third Harmonically Generated Melasma Images with different Dendricity Levels as described by Medical Experts



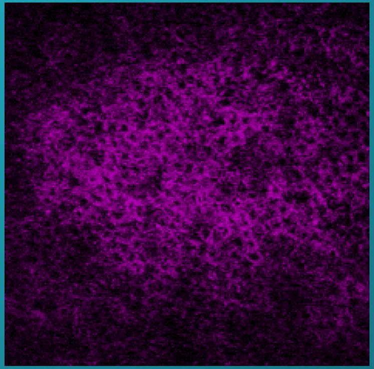
Normal Image



Dendritic Image

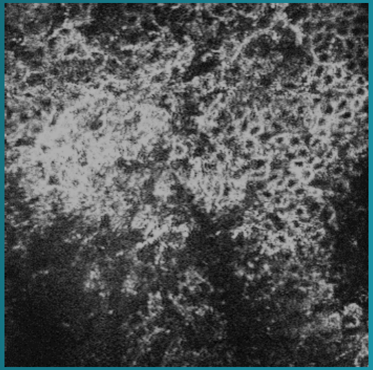


More Dendritic Images

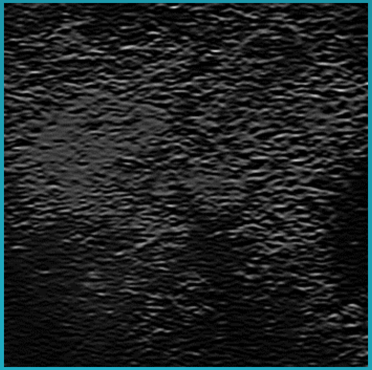


Most Dendritic Image

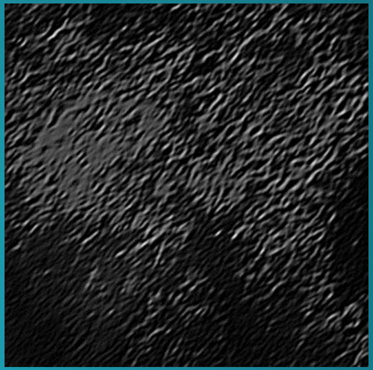
Gabor Features to Characterize Dendricity Directions



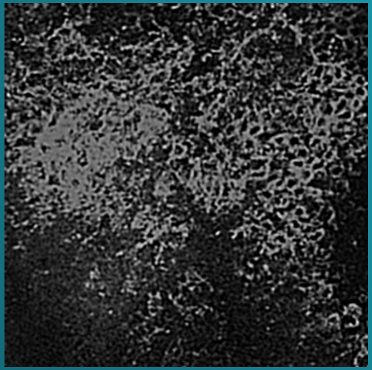
Original Image



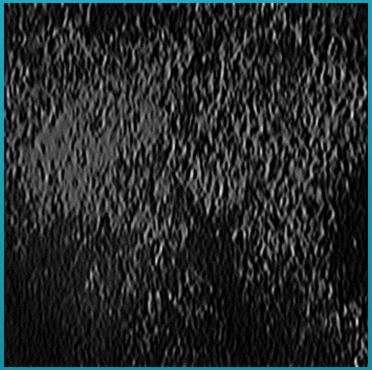
Direction = 0°



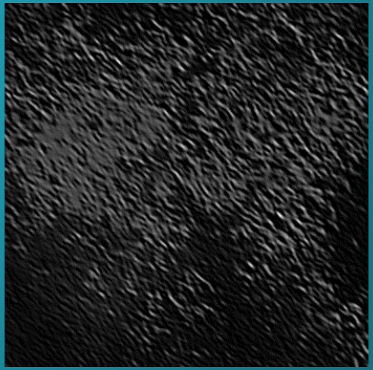
Direction = 45°



Combination Results

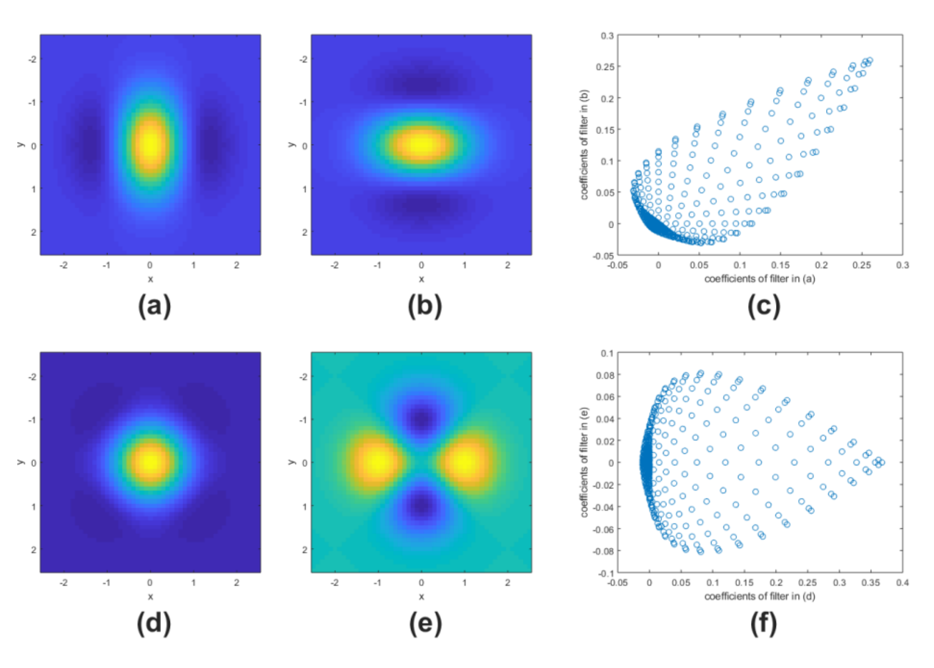


Direction = 90°



Direction = 135°

Gabor Features and PCA Transformed Filters with Higher Symmetry



(a) (b) (c)

(d) (e) (f)

(a) Gabor filter with parameters: $\sigma_x = \sigma_y = 2.46 / \pi$, $\omega = \pi / 2$, and $\theta = 0$.
(b) Gabor filter with parameters: $\sigma_x = \sigma_y = 2.46 / \pi$, $\omega = \pi / 2$, and $\theta = \pi / 2$.
(c) Coefficients distribution for two Gabor filters.
(d) Transformed filter for first Gabor filter.
(e) Transformed filter for second Gabor filter.
(f) Coefficients distribution for two transformed filters.

Comparison for Gabor Filter Bank and Transformed Gabor Filter Bank

- We perform the Gabor filter bank consisted of 16 Gabor filters over the 512×512 image with zero-padding.

Dataflow Model		Gabor Filter Bank	Transformed Gabor Filter Bank	Remark
Number of operations	**Addition**	201,326,592	72,351,744	
	Multiplication	205,520,896	60,817,408	
Storage requirement (bits)		139,776	119,808	
Data transfer (bits)		142,606,336	142,606,336	Estimated for total or average but peak data transfer should drop.
Degree of parallelism		16	1	Implementation of Gabor filter bank is performing convolution with 16 Gabor filters sequentially, since we want the same starting point to compare.
Execution time (sec)	**Intel Core i7-5820k**	1.85	0.19	
	Intel Core i7-3770	2.51	0.27	
	Intel Core i5-3550	2.01	0.21	
	Intel Core i7-930	2.78	0.33	

Conclusion

- Algorithm and Architecture needs to be looked at together
- Provides flexible, high accuracy, high efficiency, low power, and LOW COST designs
- Methodology was adopted by industry in deploying 50+ million units of LCD Panels worldwide
- Cross level of abstraction framework which systematically models computational (5^{+}G) platforms to solve cross disciplinary problems in SMART manners for another half a century (?)

Computer, Communication, Control

&

Care