Hardware-software co-design for Computer Vision

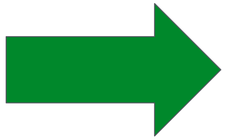
MEGVII 旷视

zsc@megvii.com

周舒畅

旷视研究院高级研究总监、AI计算组组长；2000 年入学清华电子系，博士毕业于中科院计算所，曾获得 NeuIPS2021 ML4CO DualTrack 第一名，NeuIPS2017 Learning to Run Challenge 第二名，美国国家标准技术研究院 TRAIT 2016 OCR 冠军。

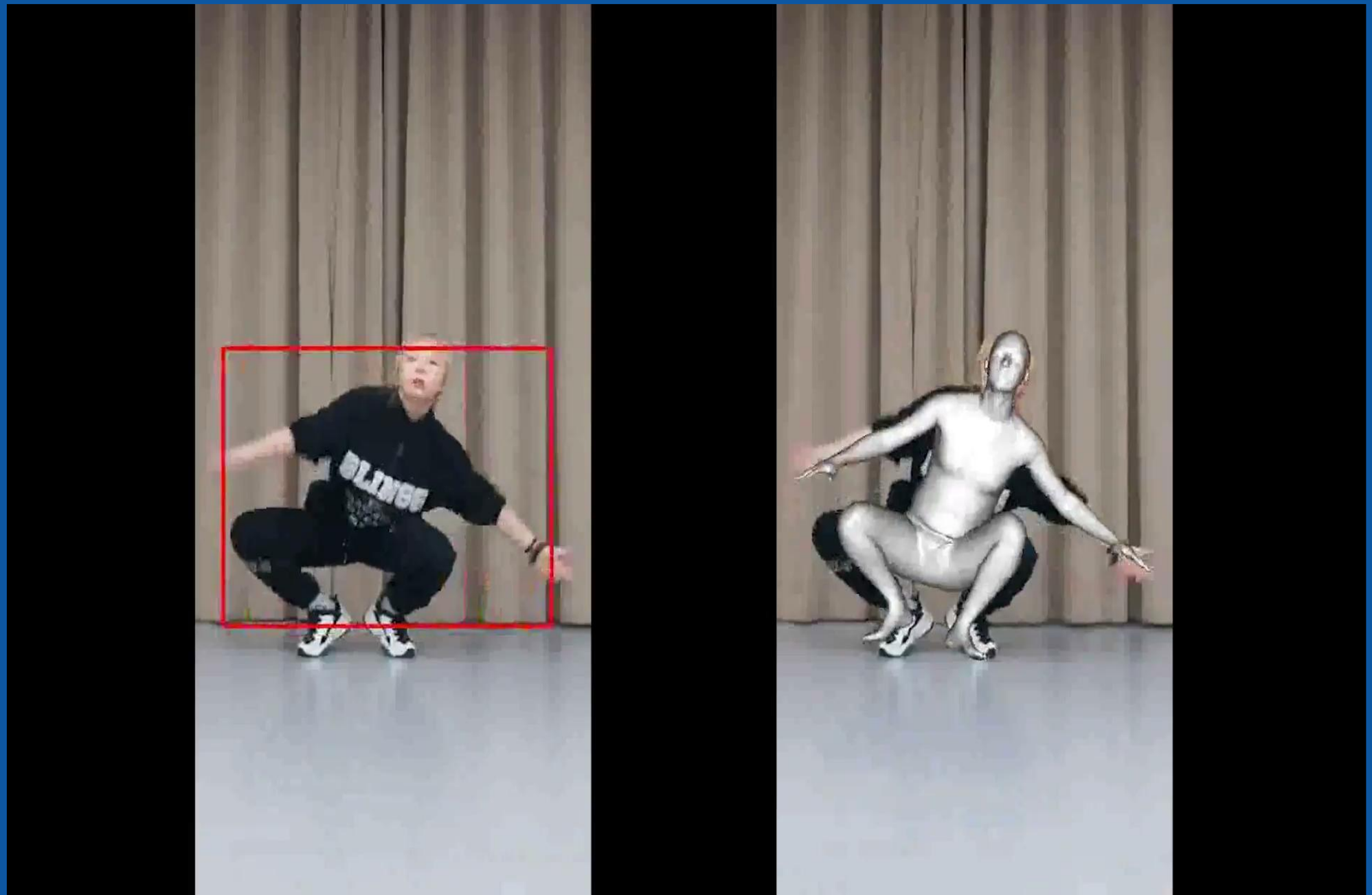
MEGVII 旷视



1. AI as of 2022
2. Accelerator as of 2022
3. Co-design of AI & Accelerator
4. Advices for future Architects
5. Q & A

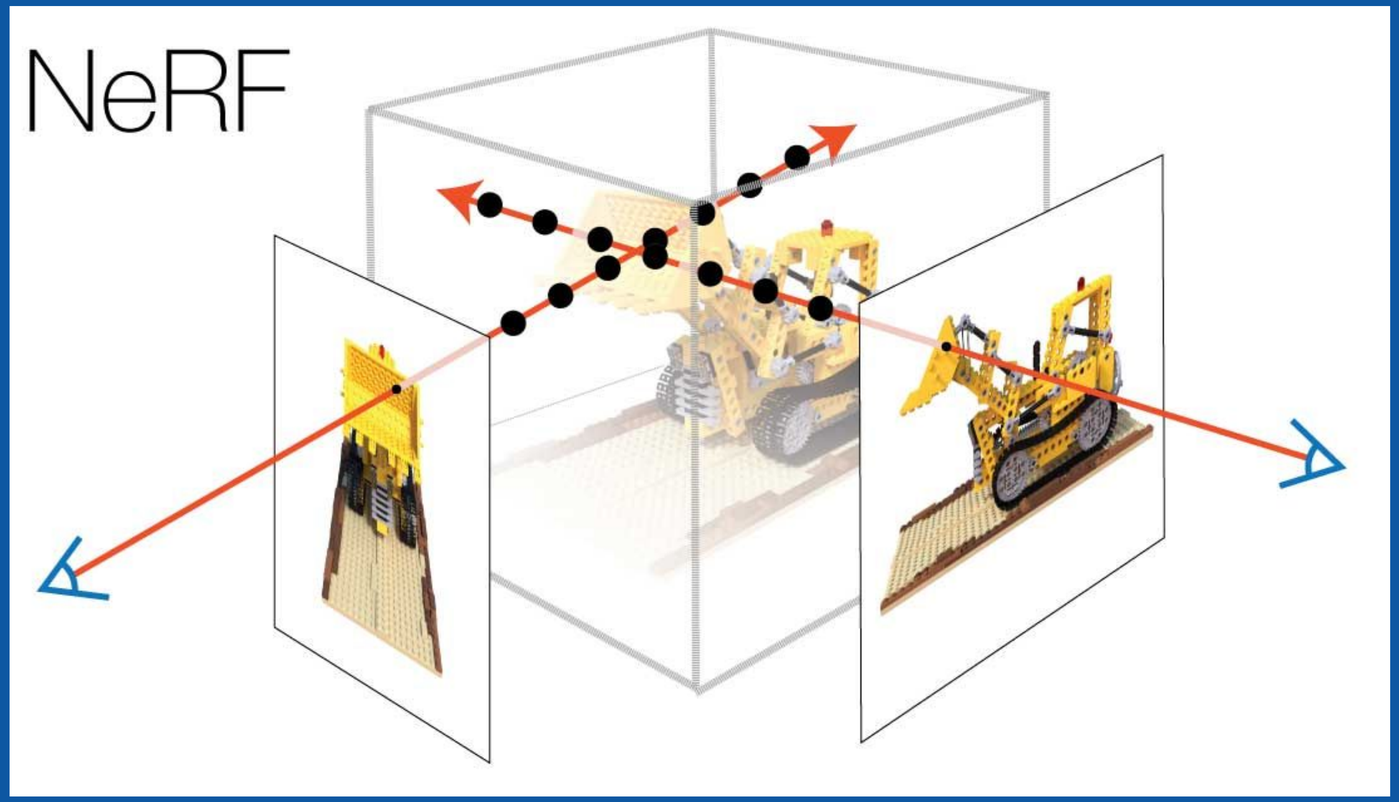
Computer Vision in 2020s

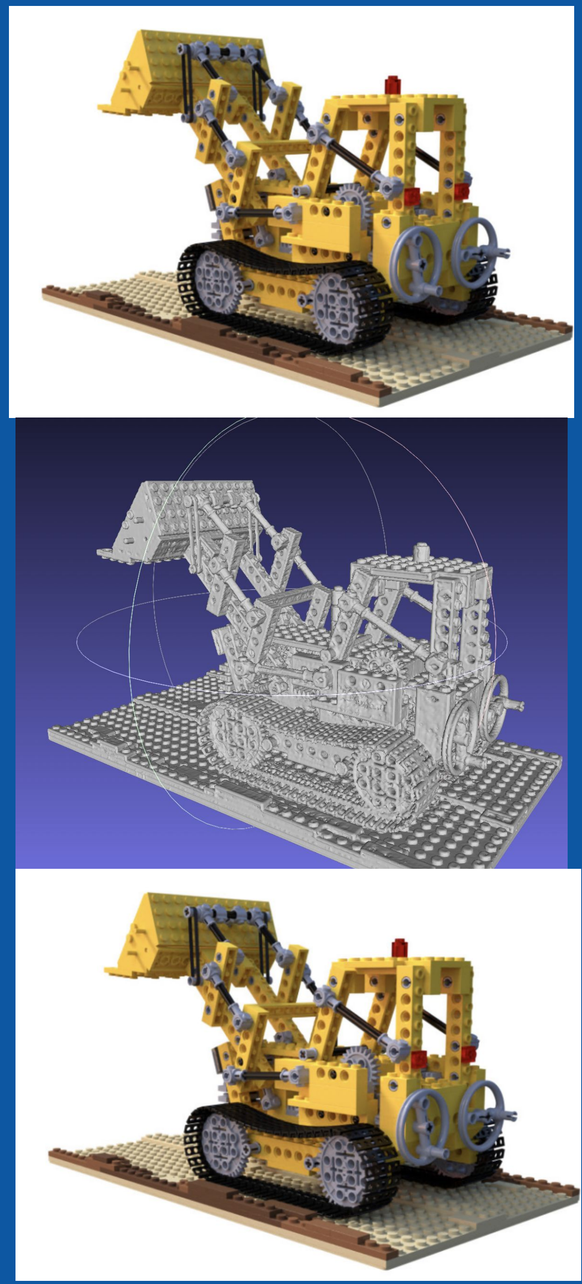
MEGVII 旷视



Computer Vision meets Computer Graphics

MEGVII 旷视





Image

Feature

Differentiable Rendering

Image

Loss

3D Reconstruction for Streetview

MEGVII 旷视

San Francisco by Block-NeRF
https://waymo.com/intl/zh-cn/research/block-nerf/

CV meets NLP: Text driven image generation

MEGVII 旷视

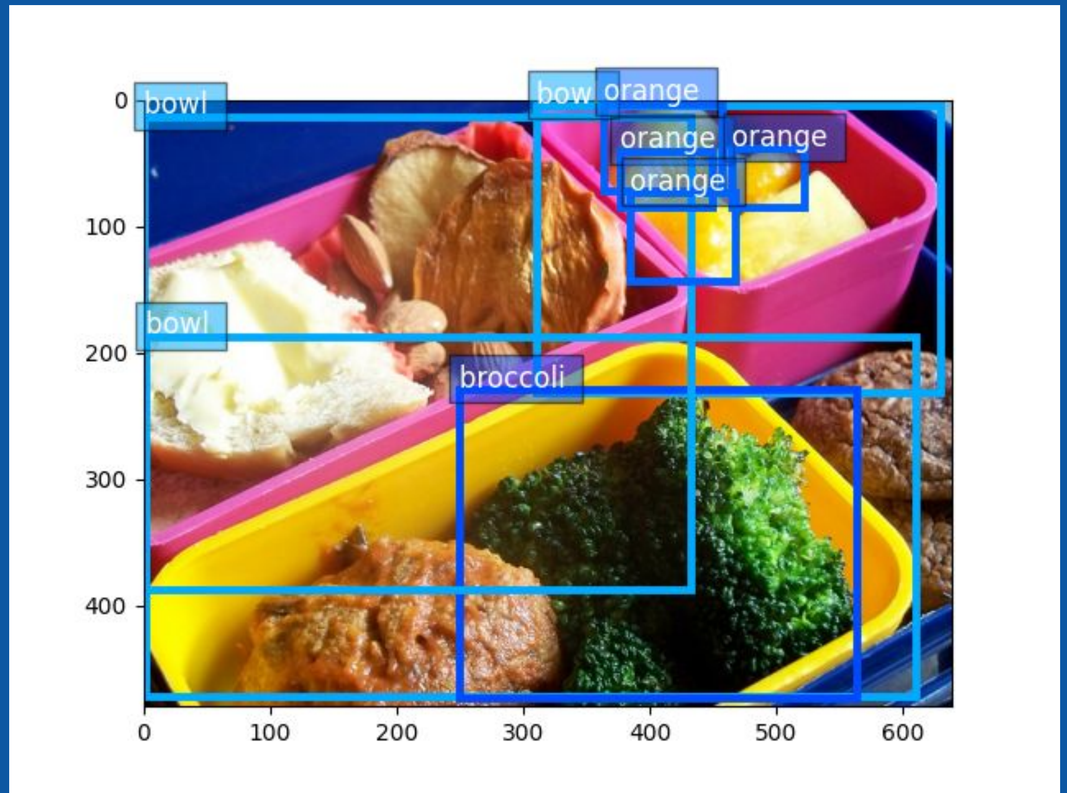
ultra cute cat, I was too scared at that moment

cat, sanxingdui, Chinese relics, bronze

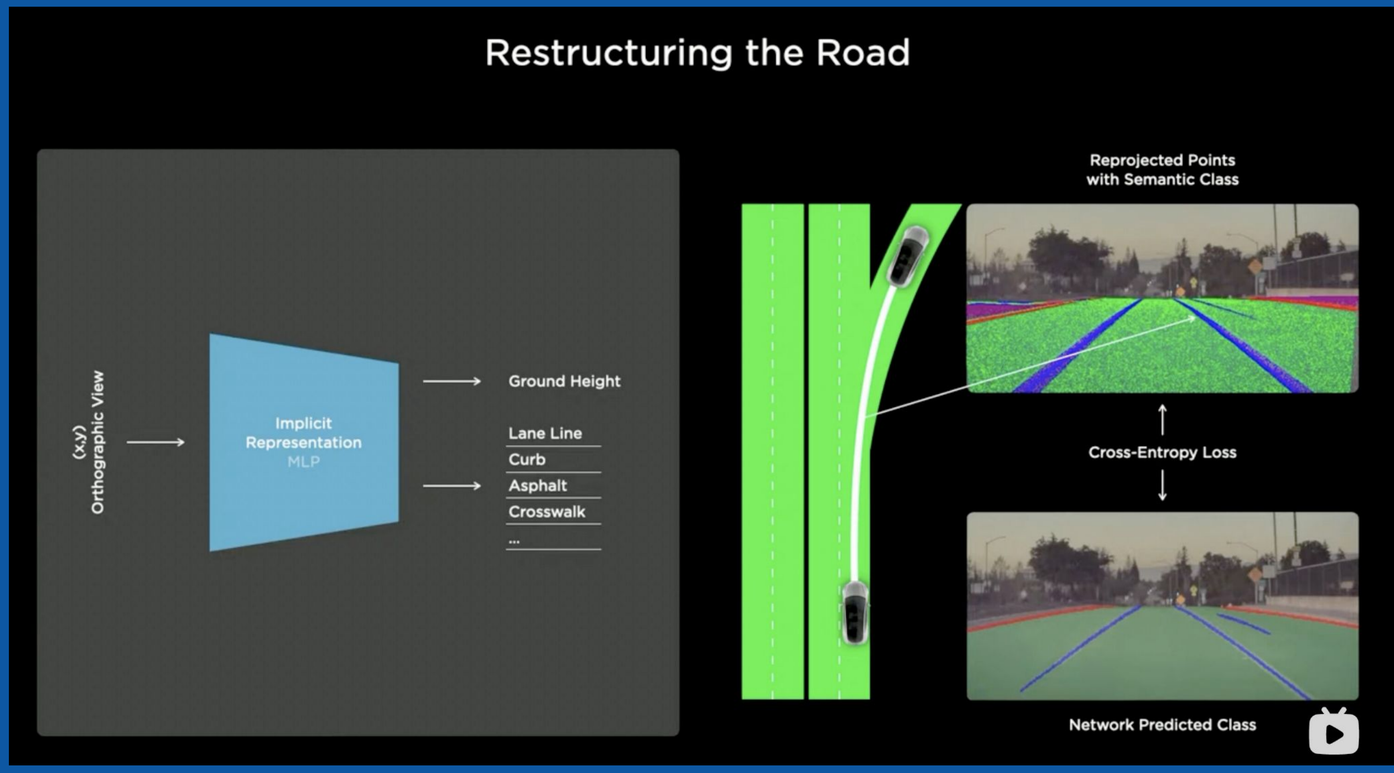
Computer Vision as of 2022

MEGVII 旷视

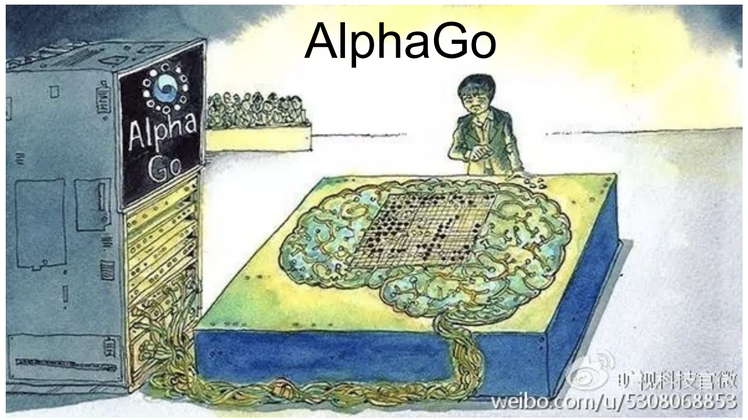
- Old School
  - 2D Vision + Autoencoder
  - Image To Labels
  - Image level annotation
  - CPU + GPU



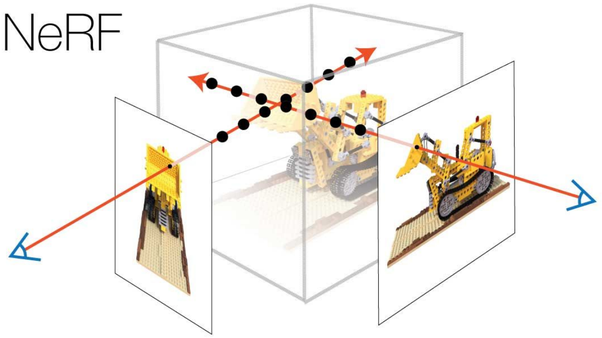
- New School
  - CV + CG: 3D Vision + 3D Generation
  - CV + NLP: Text to Image etc.
  - Point cloud annotation
  - CPU + Domain Specific Accelerators



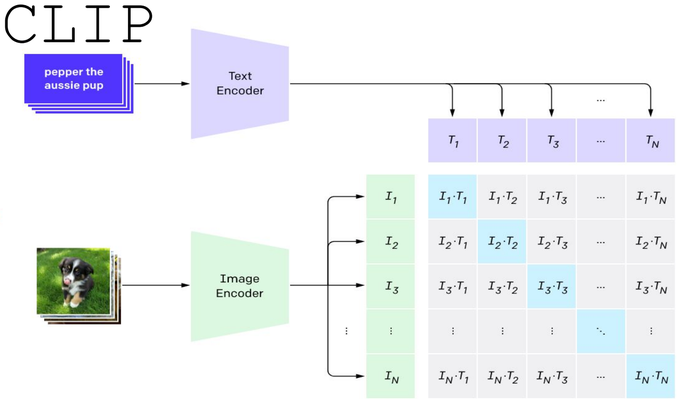
AlphaGo



NeRF

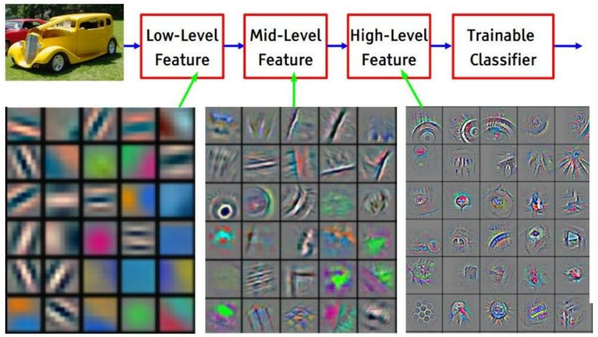


CLIP

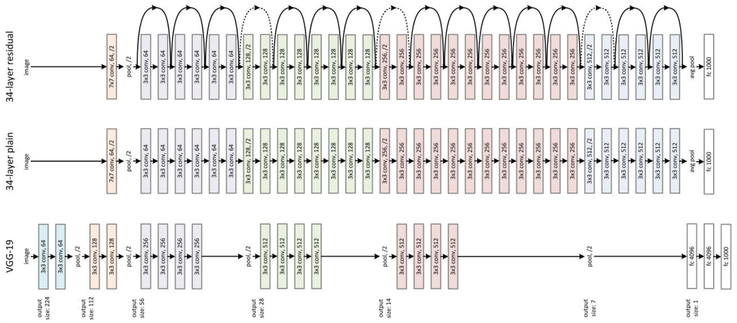


ImageNet

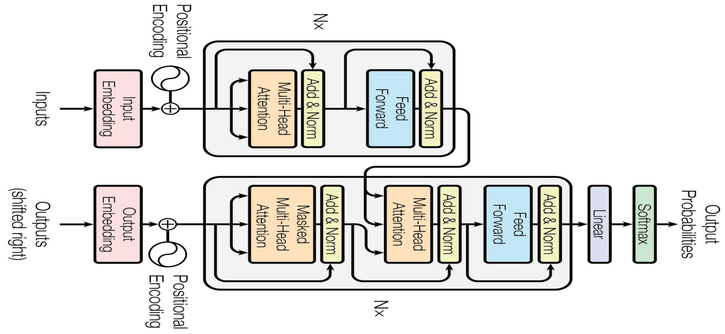
Convolutional NN



ResNet



Transformer

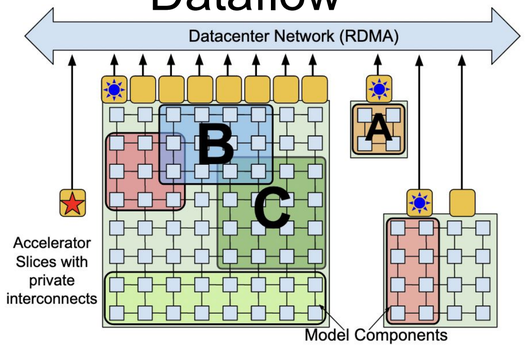


GPU

Accelerators

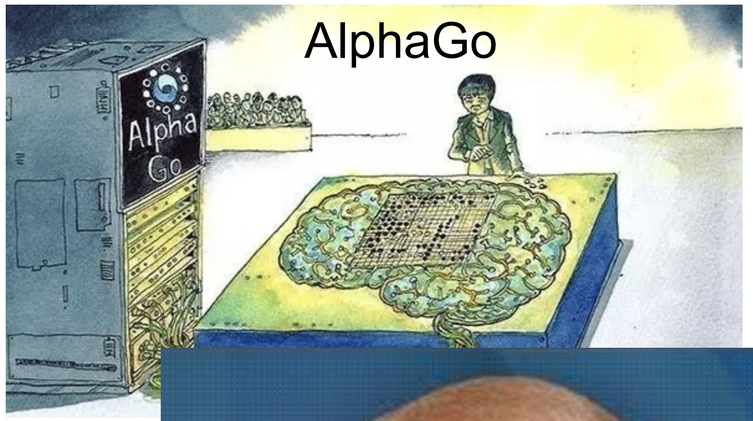
Quantization

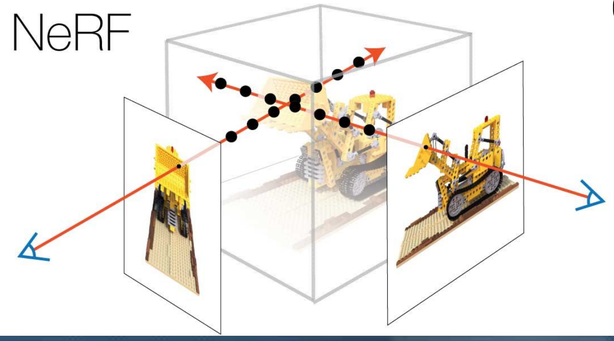
Dataflow

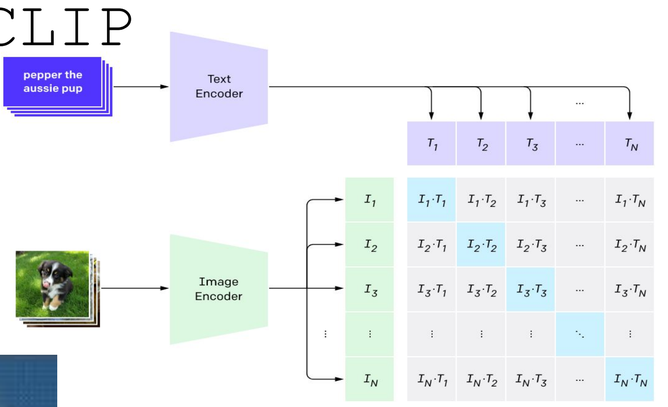


2011 *"BigBang"*

"Roaring 20s"

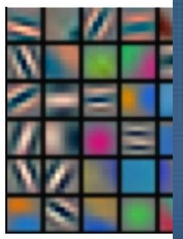






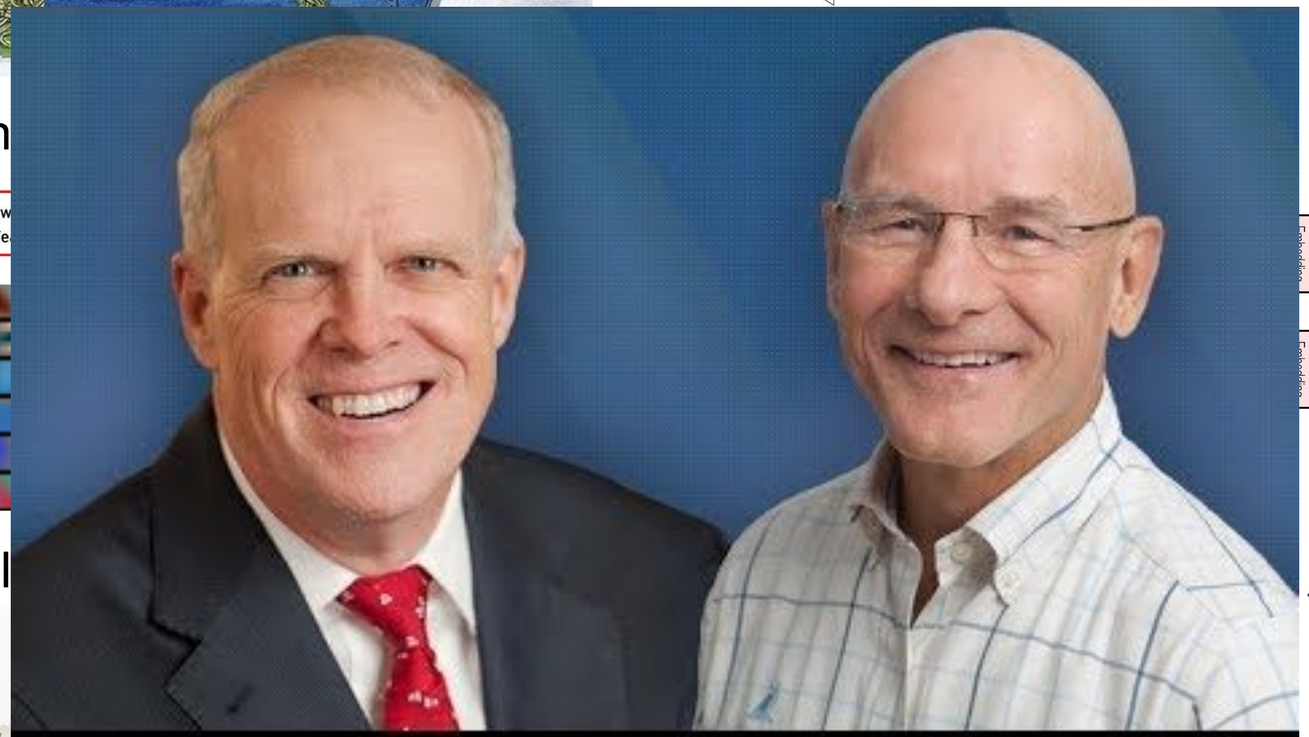
ImageNet

Con



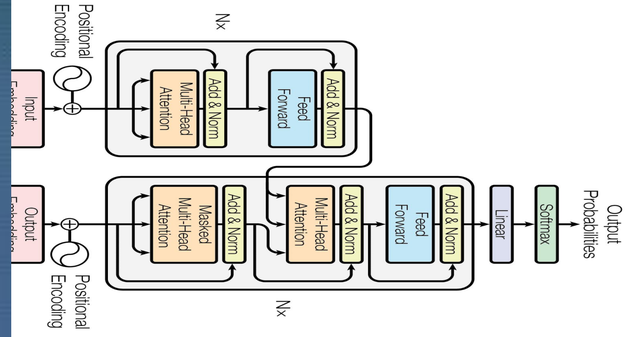
GPU

Accel

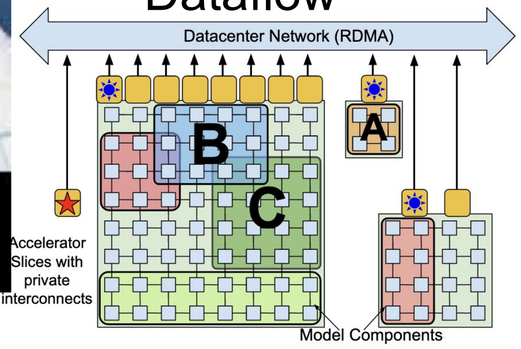


New *Computer Architecture Golden Age*

Transformer



Dataflow



W

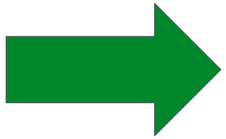
αW^B

2011 *"BigBang"*

2018 Turing Award

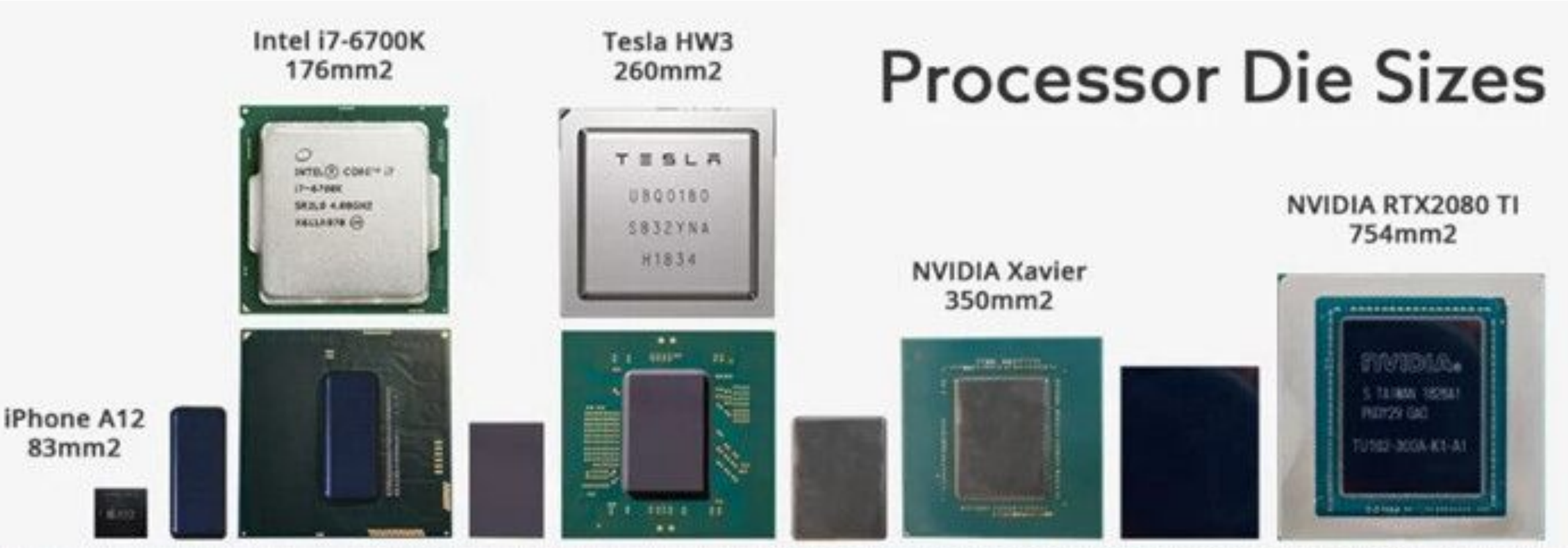
"Roaring 20s"

MEGVII 旷视



1. AI as of 2022
2. Accelerator as of 2022
3. Co-design of AI & Accelerator
4. Advices for future Architects
5. Q & A

Accelerator as of 2022: Gap between Cloud and Edge



9X die size

Accelerator as of 2022: Gap between Cloud and Edge

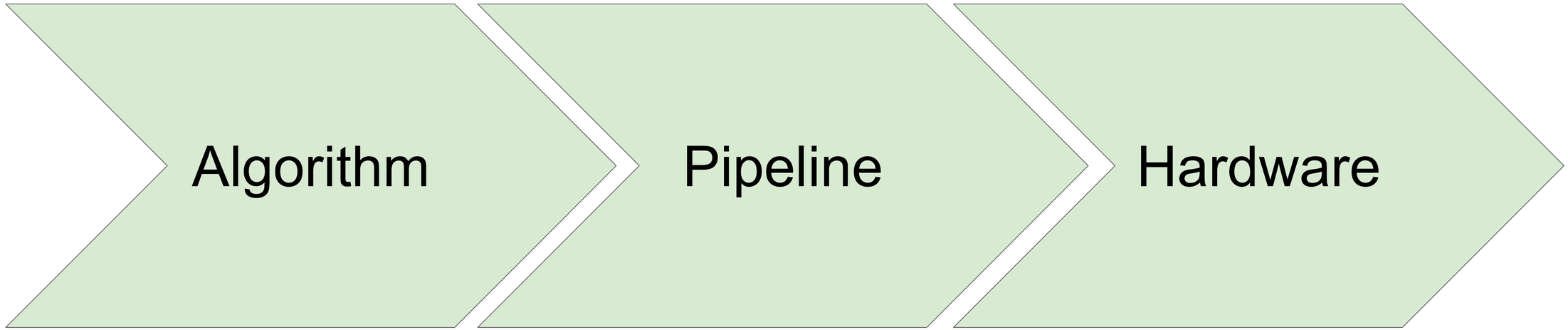
Form Factor	H100 SXM	A typical Edge Chip	GAP
FP64	30 teraFLOPS	No support	
FP64 Tensor Core	60 teraFLOPS		
FP32	60 teraFLOPS	100 GOPS	300X
TF32 Tensor Core	1,000 teraFLOPS* | 500 teraFLOPS		
BFLOAT16 Tensor Core	2,000 teraFLOPS* | 1,000 teraFLOPS		
FP16 Tensor Core	2,000 teraFLOPS* | 1,000 teraFLOPS		
FP8 Tensor Core	4,000 teraFLOPS* | 2,000 teraFLOPS	4 TOPS	500X
INT8 Tensor Core	4,000 TOPS* | 2,000 TOPS		
GPU memory	80GB	4GB	20X
Max thermal design power (TDP)	700W	5W	140X

Case study: Build a Real-time Vision Pipeline with a low end FPGA

- Zynq 7020 FPGA
- costs 18 USD, ~20Gops float16 ZynqNet: An FPGA-Accelerated Embedded Convolutional Neural Network (2020)
- FHD (1080p) is ~40 times the 224x224 resolution
- Need a 1000 FPS backbone for Real-time (25FPS) processing!

	Z-7020
Logic Cells (K)	85
Block RAM (Mb)	4.9
DSP Slices	220
Maximum I/O Pins	200
Maximum Transceiver Count	-

Algorithm-Pipeline-Hardware: the Codesign Triathlons



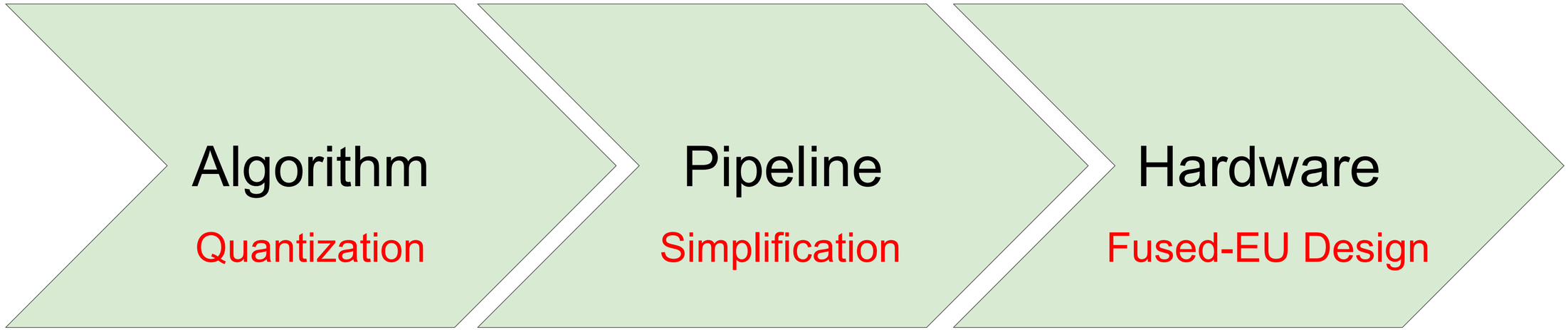
- Sets the quality upper bound
- Need triagable metrics
- Weekly update

- Memory/storage bandwidth & CPU
- Determinism
- Quarterly update

- Stable ISA for decoupled evolution
- Physical measurements
- Yearly update

Algorithm-Pipeline-Hardware: the Codesign Triathlons

For Codesign of Real-time 1080p Processing on a Low-end FPGA



- Sets the quality upper bound
- Need triagable metrics
- Weekly update

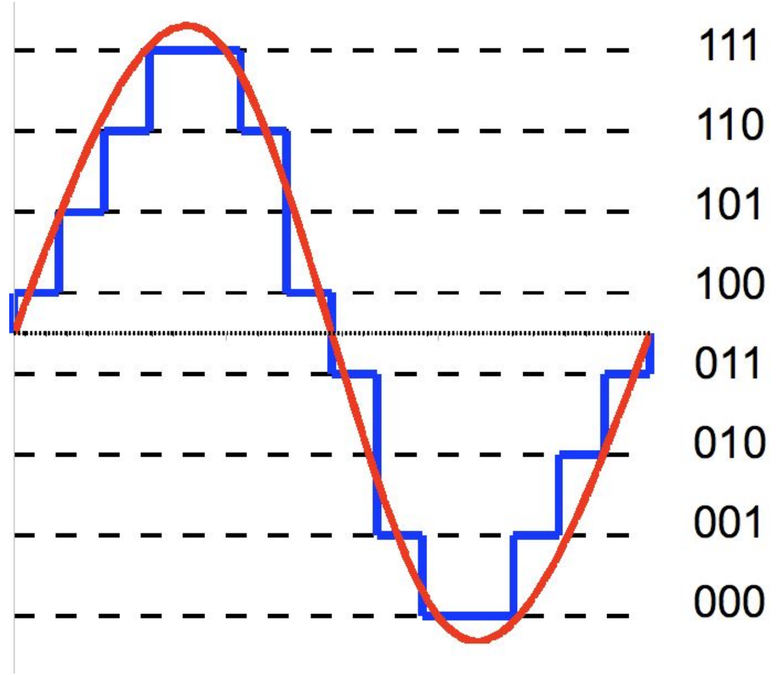
- Memory/storage bandwidth & CPU
- Determinism
- Quarterly update

- Stable ISA for decoupled evolution
- Physical measurements
- Yearly update

Fixed Points: Quantization

- Linear Quantization
  - Essentially rounding
  - Con: dynamic range of floating point numbers are problematic
- Logarithmic Quantization
  - Quantize log(x) instead of x
  - Achieve 12 bit quality with 8 bits
  - Con: addition not efficient on existing platforms

$$Q(x) = \Delta \cdot \left\lfloor \frac{x}{\Delta} + \frac{1}{2} \right\rfloor$$



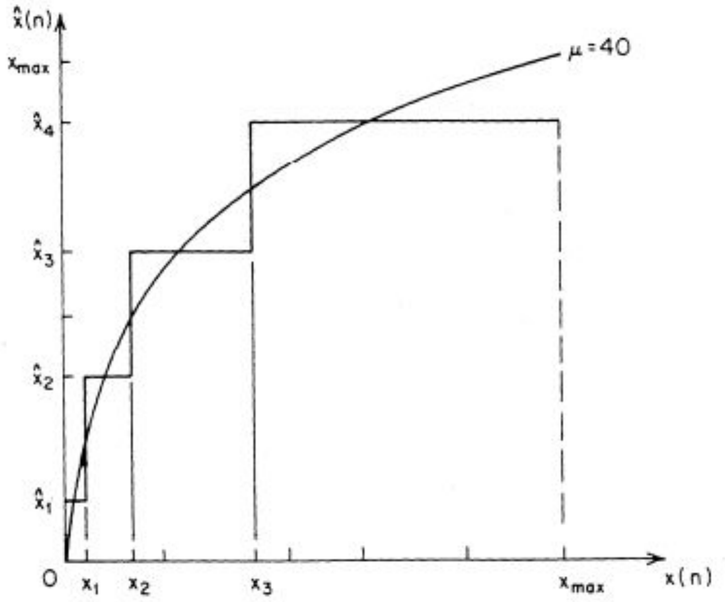
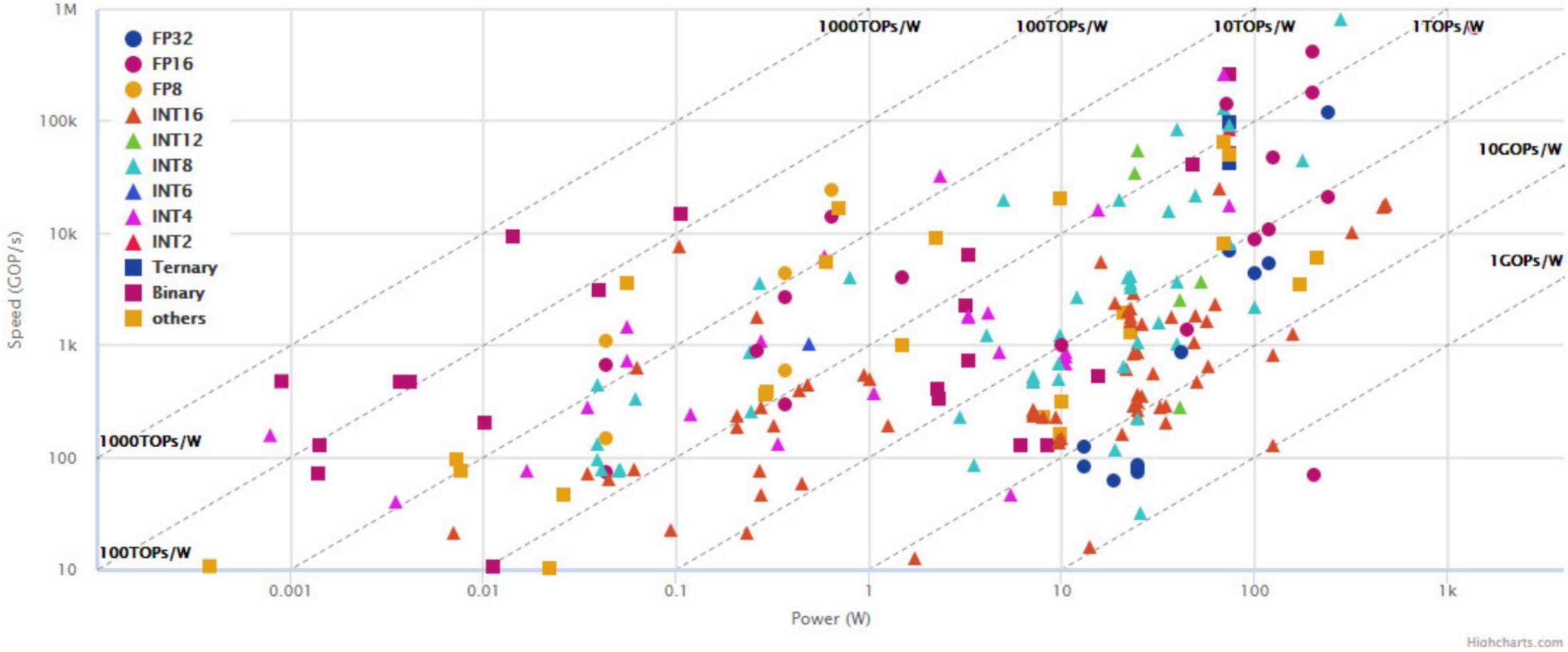


Fig. 5.16 Distribution of quantization levels for a μ-law 3-bit quantizer with $\mu = 40$.

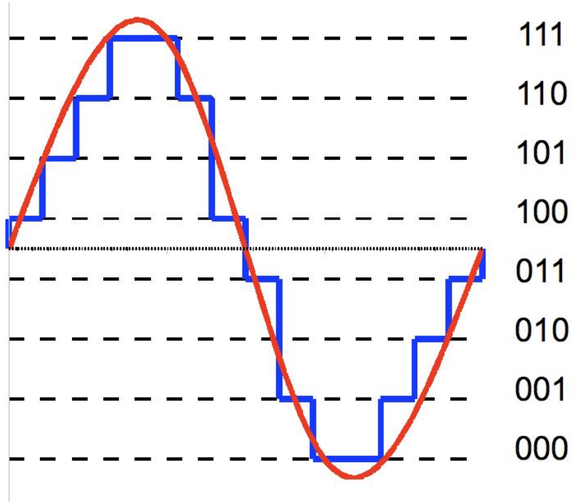
High TOPS/W generally due to low-bit QNN



Source: https://nicsefc.ee.tsinghua.edu.cn/projects/neural-network-accelerator/

Differentiable Quantization

- Bengio ’13: Estimating or Propagating Gradients Through Stochastic Neurons for Conditional Computation
  - REINFORCE algorithm
  - Decompose binary stochastic neuron into stochastic and differentiable part
  - Injection of additive/multiplicative noise
  - Straight-through estimator



Gradient vanishes after quantization.

Quantization also at Train time

- Neural Network can adapt to the constraints imposed by quantization
- Exploits “Straight-through estimator” (Hinton, Coursera lecture, 2012)

$$x \approx \hat{x}$$

$$\Rightarrow$$

$$\frac{\partial}{\partial x} \approx \frac{\partial}{\partial \hat{x}}$$

- Example

$$\textbf{Forward: } q \sim Bernoulli(p) \qquad q \approx \mathrm{E}[q] = p$$

$$\textbf{Backward: } \frac{\partial c}{\partial p} = \frac{\partial c}{\partial q}.$$

Alternative Simple Implementation: zero_grad(q - p) + p

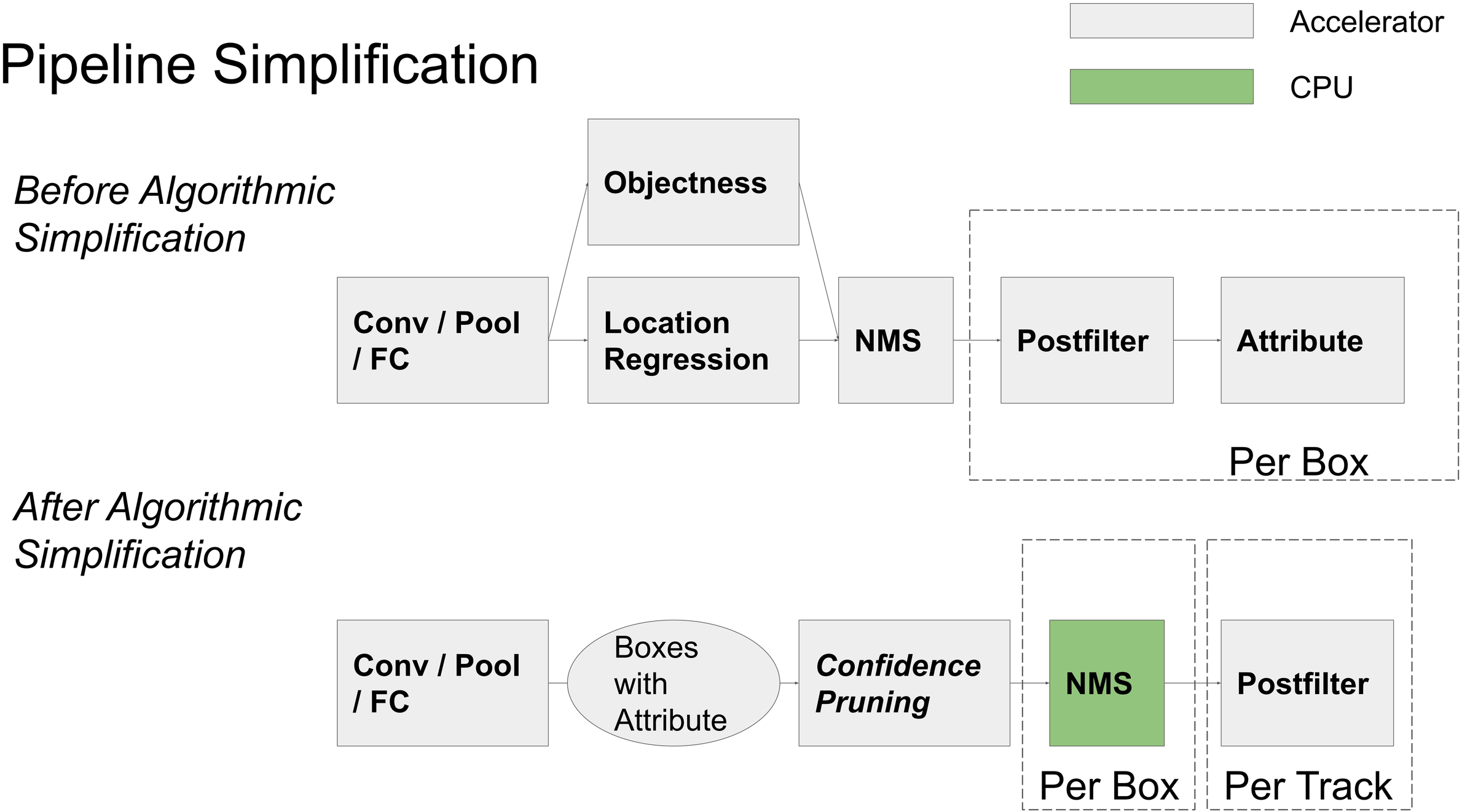
Benefits of Quantized Neural Networks

- Reduce weight size
- Reduce feature size
  - larger feature library size
- Speedup computation (usually require hardware support)
  - *O(Bit-width)* if SIMD, *O(Bit-width^2)* if special hardware
  - Saves bandwidth (intra-chip and off-chip), eases P & R
  - Dense computation

$$\boldsymbol{x} \cdot \boldsymbol{y} = \sum_{m=0}^{M-1} \sum_{k=0}^{K-1} 2^{m+k}\, bitcount[and(c_m(\boldsymbol{x}), c_k(\boldsymbol{y}))]$$

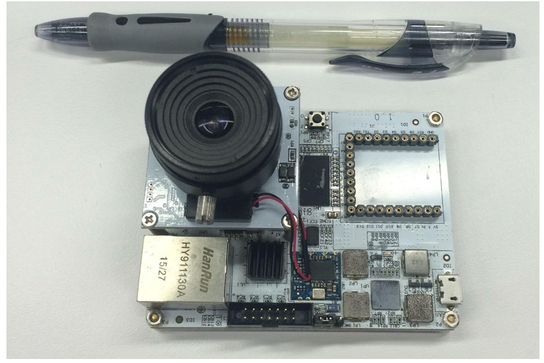
	INT8	INT4	Binary
Tesla T4	130 T	260 T	
Megvii Zynq 7020	0.05 T	0.2 T	3.2 T

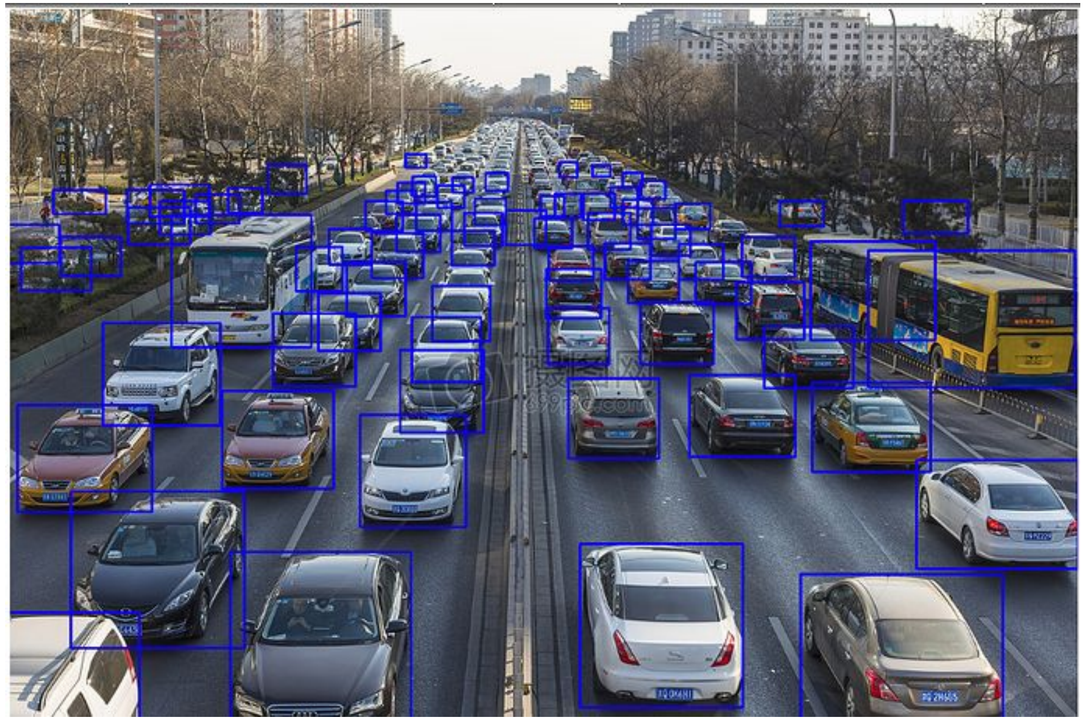
Pipeline Simplification



Real-time Object Detection on a low-end FPGA

Zynq 7020

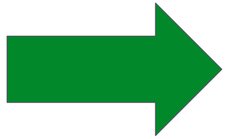




MEGVII 旷视

1 AI as of 2022

2 Accelerator as of 2022



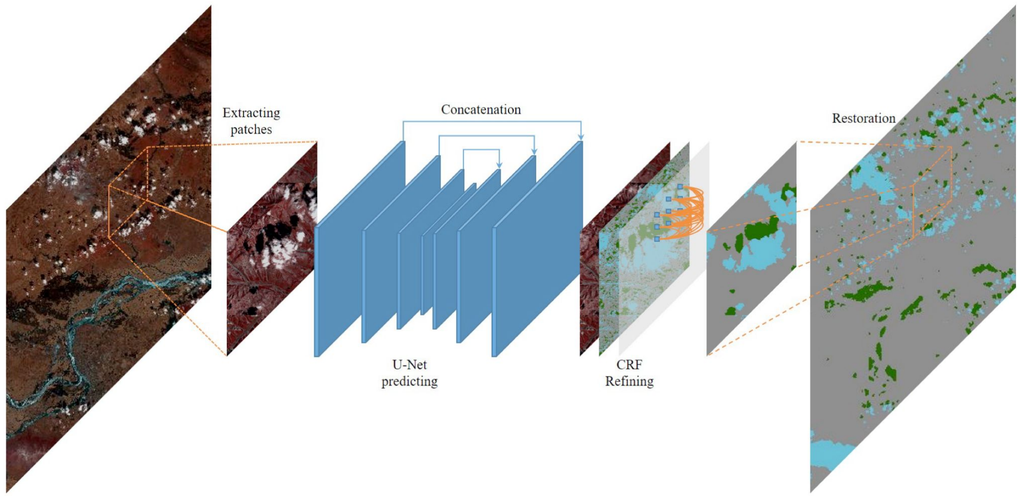
3 Co-design of AI & Accelerator

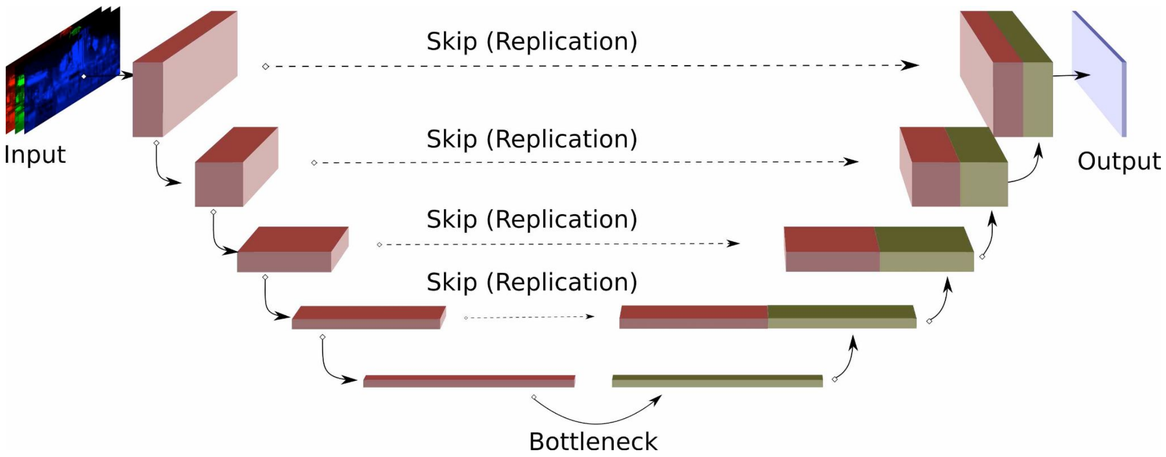
4 Advices for future Architects

5 Q & A

Image-to-Image Neural Models are Computation Intensive

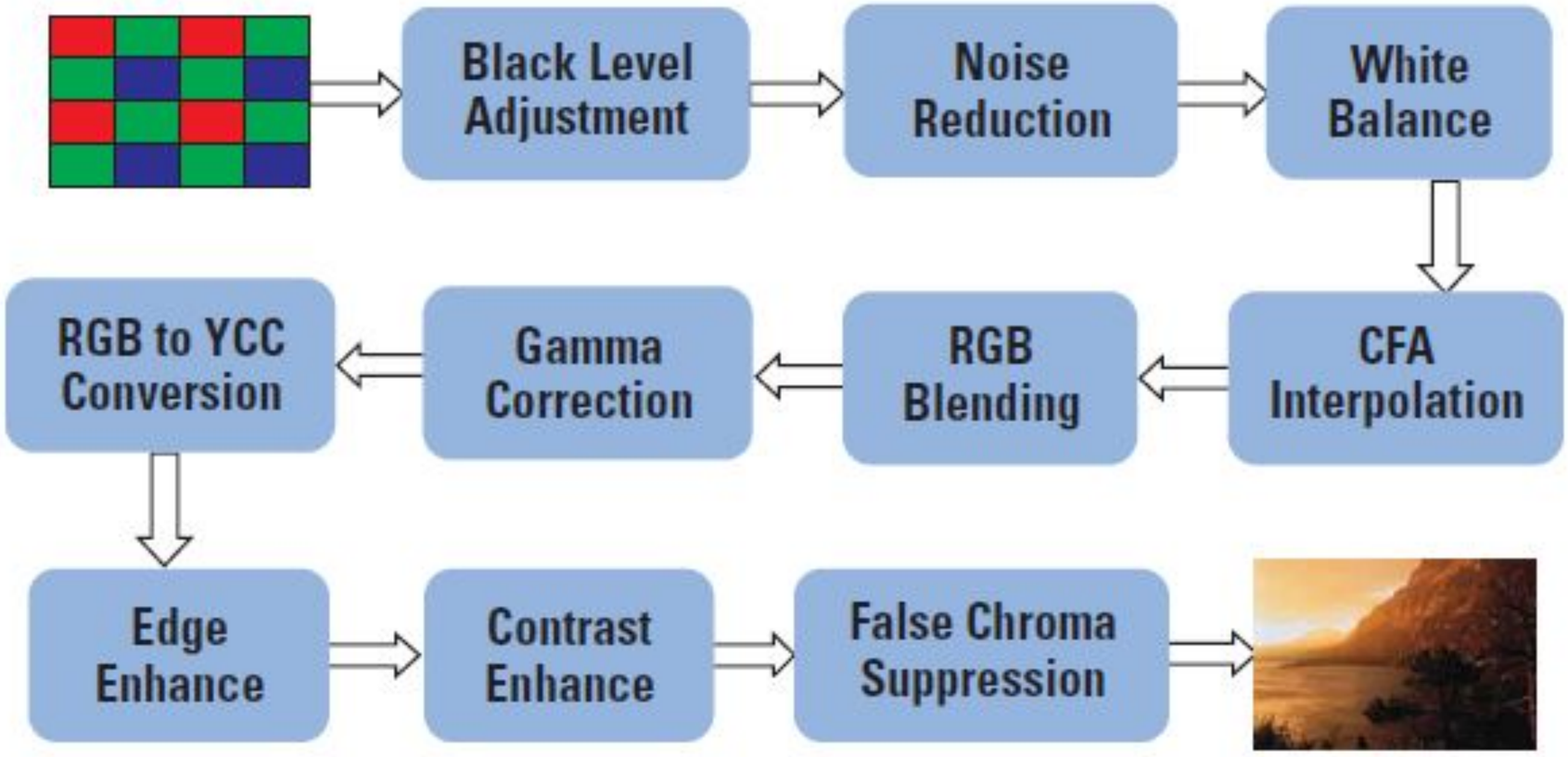
Semantic Segmentation | Optical Flow | Stereo Depth | Surface Normal



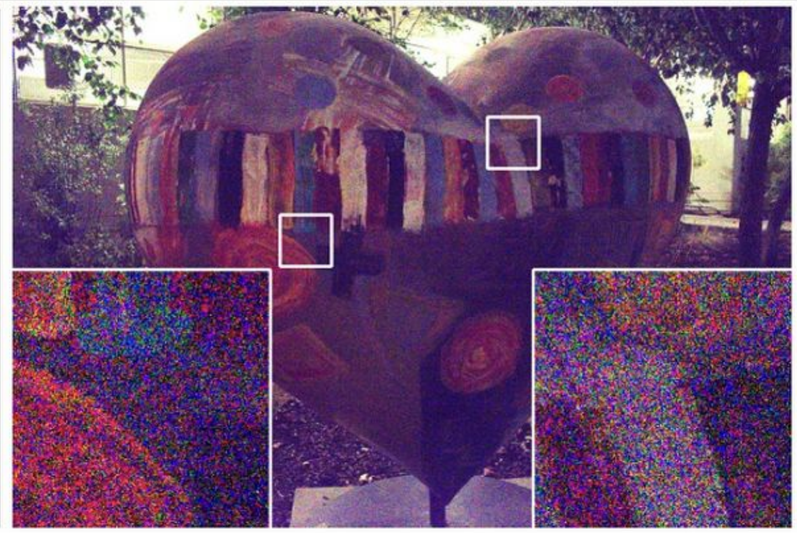


Each task requires several TOPS for real-time 1080p processing.

Computer Vision Pipelines: ISP



(a) JPEG image produced by camera

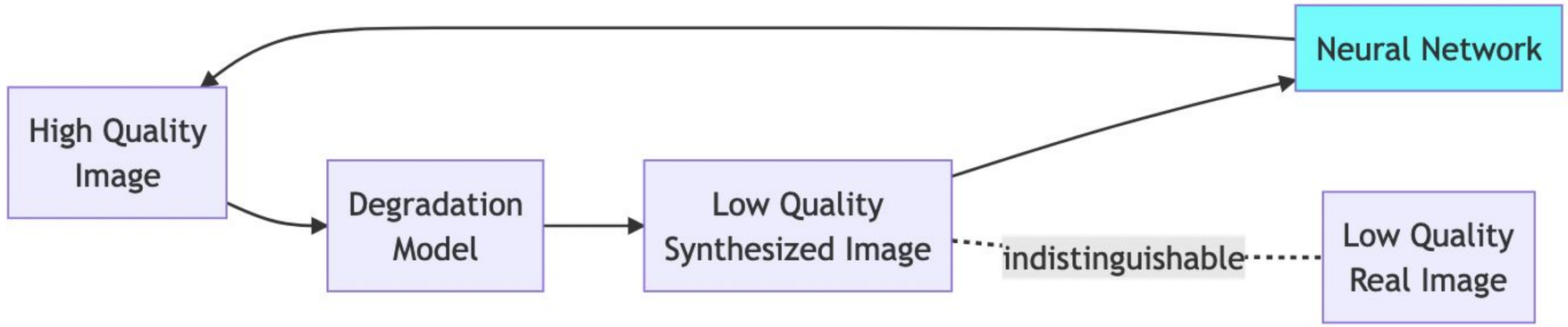


(b) Raw data via traditional pipeline

Co-design for AI-ISP: Solving Inverse Problems

- NN solution to Inverse Problem

Explicit model

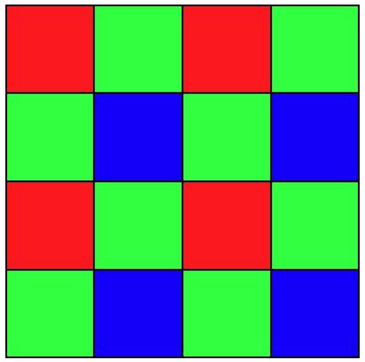


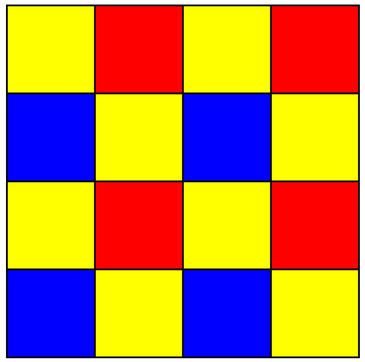
Implicit model:
Deep Image Prior (2017)

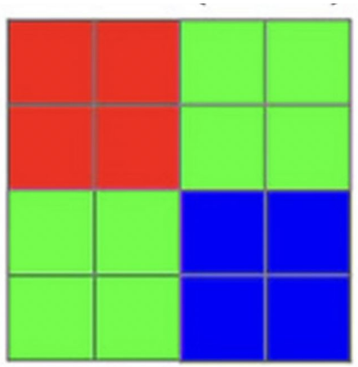
(a) Input (white=masked) (b) Encoder-decoder, depth=6

Benefits of AI-ISP

- Neural Network Capabilities
  - Denoising for Extreme Low Light
  - Higher Resolution for Demosaicing
- Flexible Pipeline
  - Change stage order, or Fuse stages
    - Joint Demosaicing and Denoising with Self Guidance, CVPR 2020
    - A Review of an Old Dilemma: Demosaicking First, or Denoising First? CVPRW 2020
  - Adapts to different input types



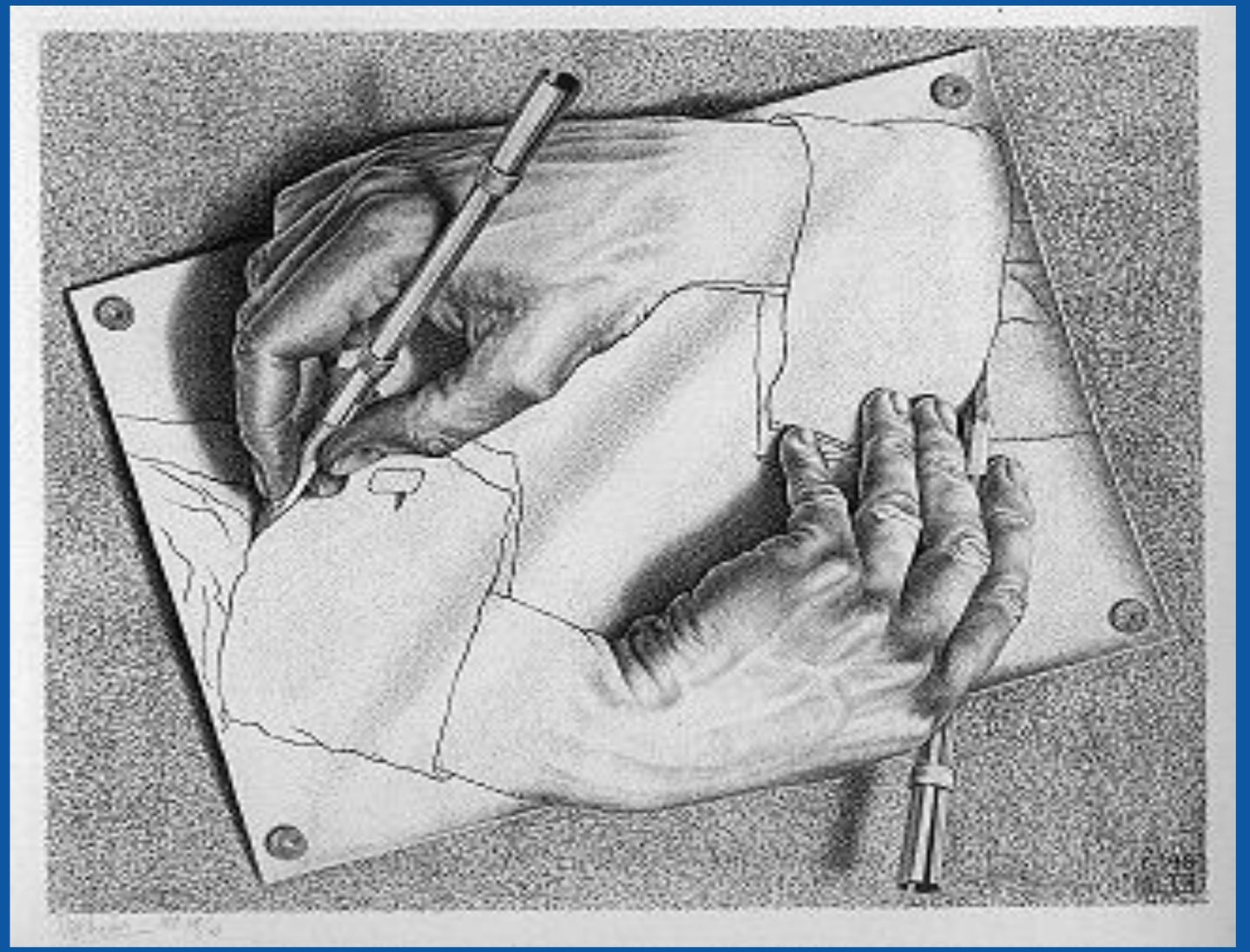




Self-Improving AI: A Higher Level of Co-design

MEGVII 旷视

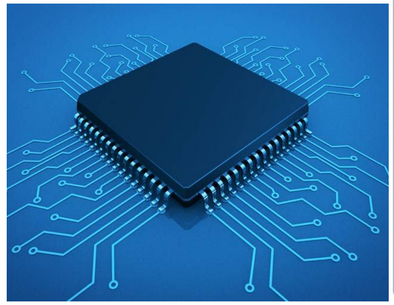
Success of AI spurs better AI Infrastructure

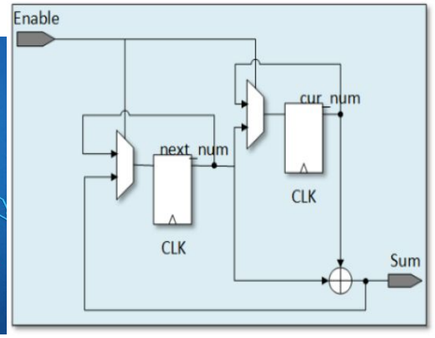


Better AI Infrastructure nutures better AI

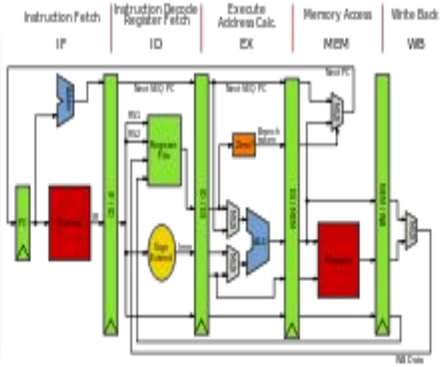
Software Hardware co-design for AI, itself

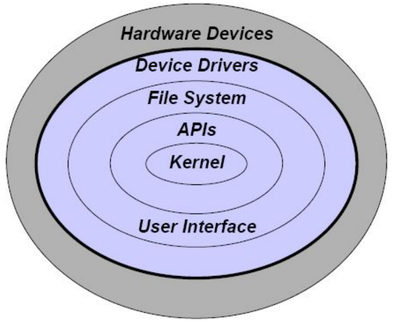
Self-Improving AI: the Computation Stack

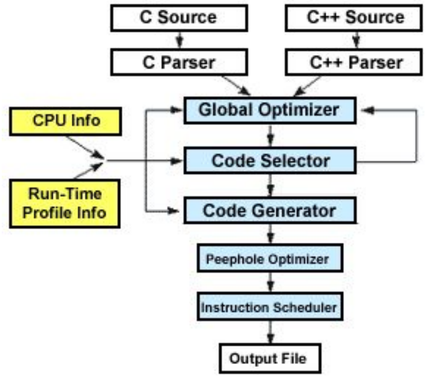


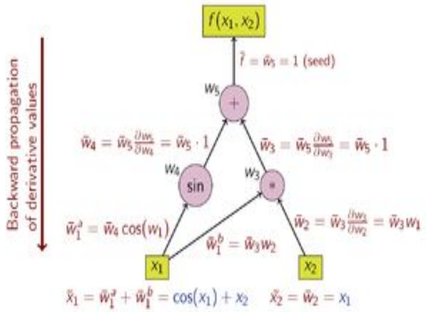


Circuit to Generate Fibonacci Series *(Fig. 13)*









Silicon
- Partitioning & Planning
- Place & Route
- Timing Closure

Verilog
- Karnaugh map
- Finite State Machine

Architecture
- ISA
- Micro-code
- Resource allocation

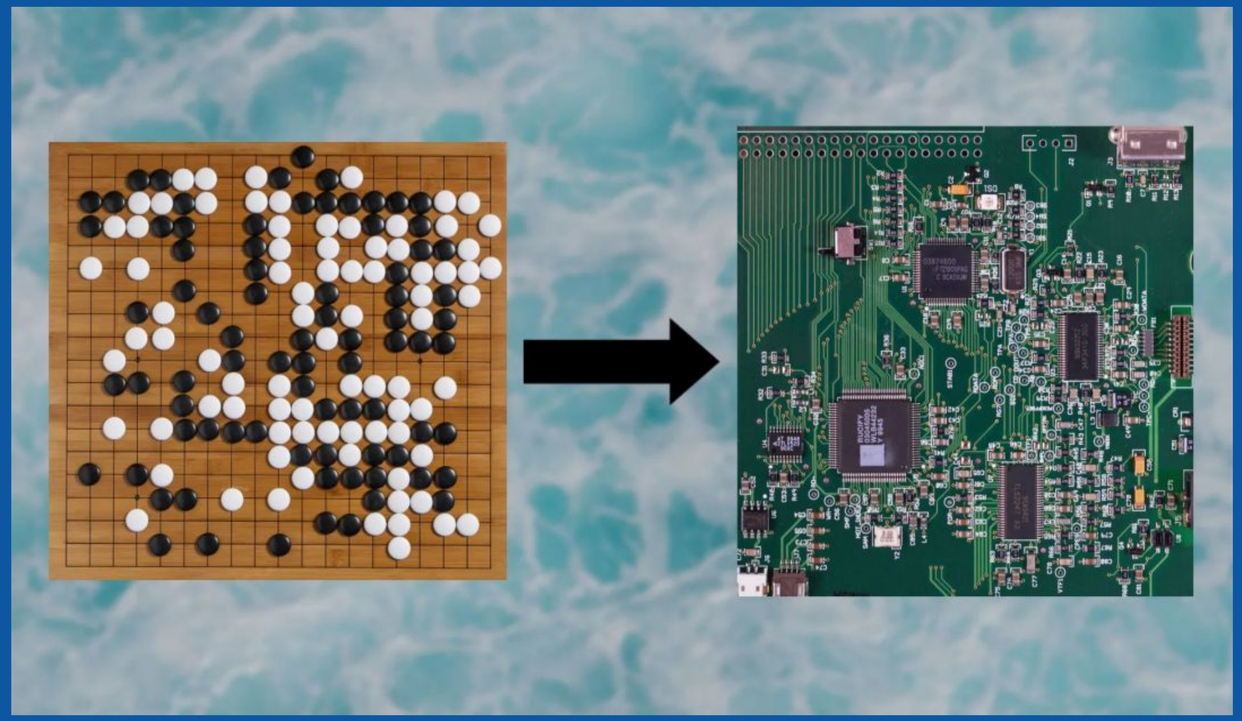
Operating System
- Page table
- File system
- Interrupts

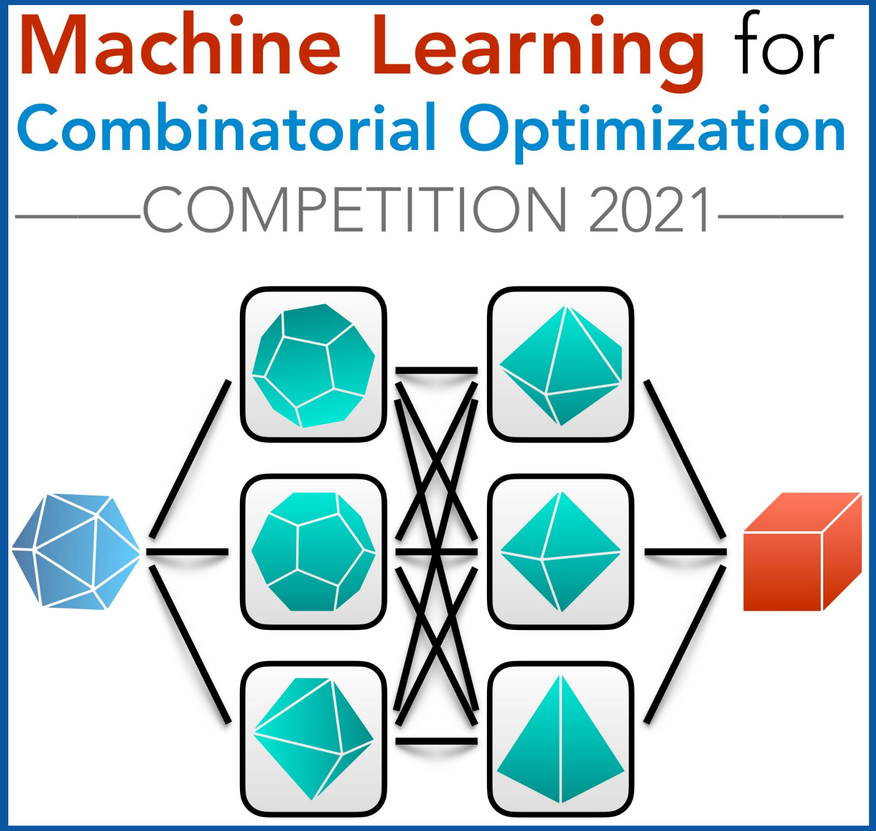
Compiler
- Parallelism mining
- Memory latency hiding

Computation Graph Engine
- Kernels
- Execution Plan

AI for Combinatorial Optimizations

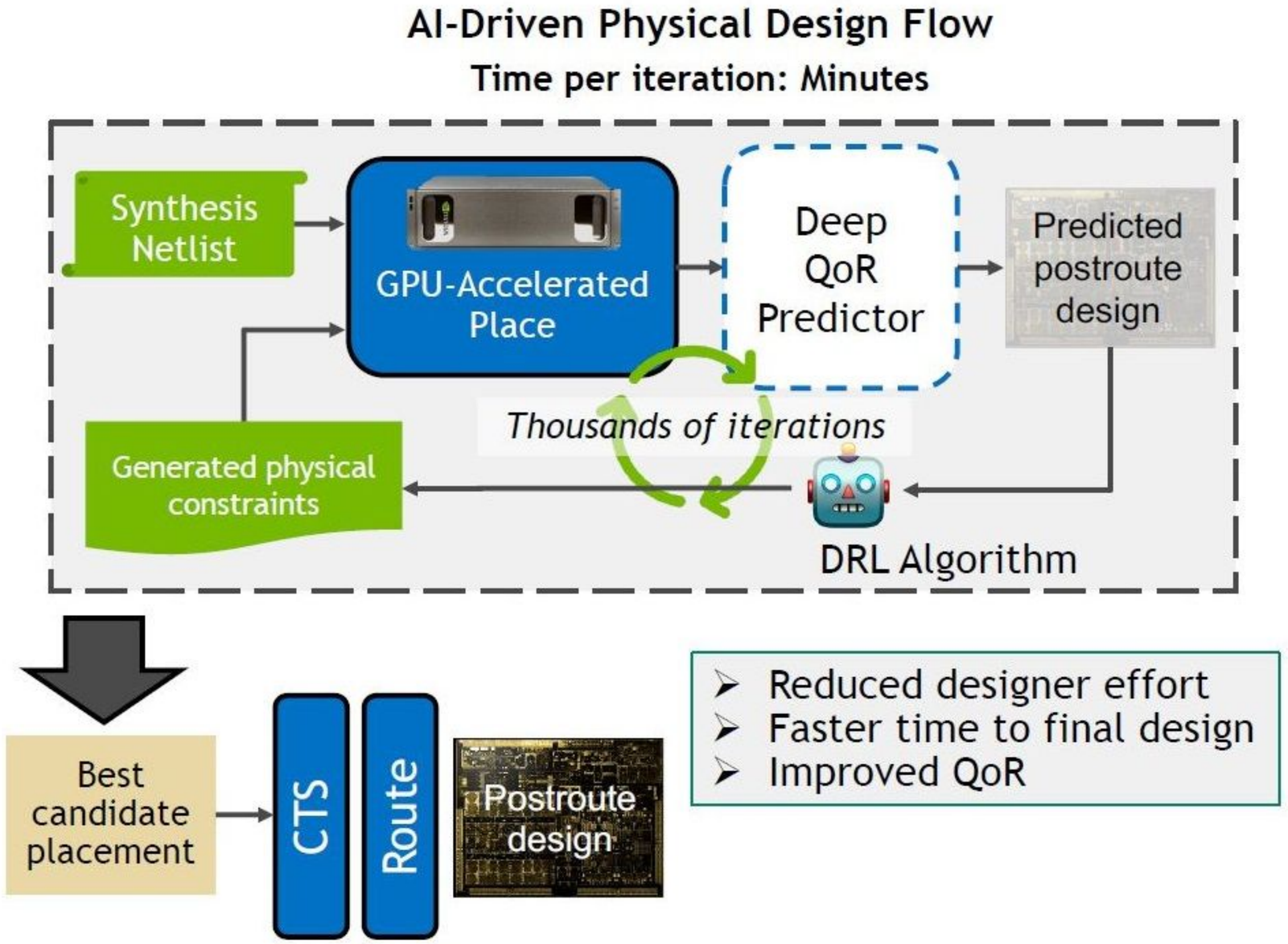
MEGVII 旷视





A trained AI → Proposals → Strict Combinatorial Search → Valid designs

AI for Hardware Design



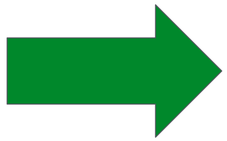
RL for Netlists, https://research.nvidia.com/publication/2020-09_Accelerating-Chip-Design

MEGVII 旷视

1 AI as of 2022

2 Accelerator as of 2022

3 Co-design of AI & Accelerator



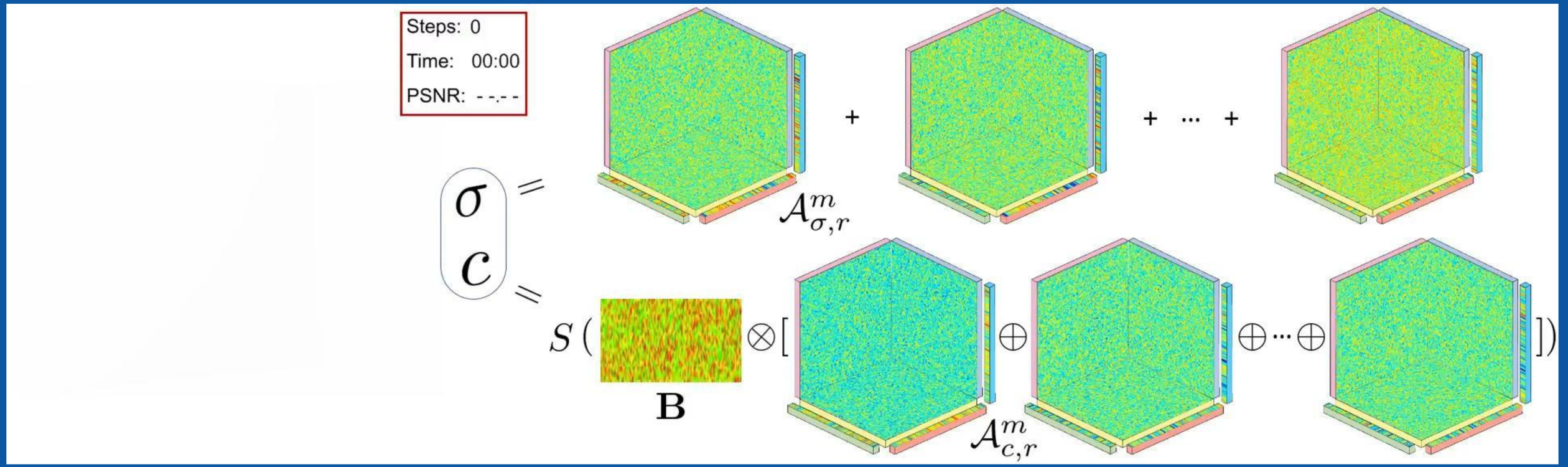
4 Advices for future Architects

5 Q & A

Survive the "Math and Science Death March"

MEGVII 旷视

TensoRF: Tensorial Radiance Fields, https://github.com/apchenstu/TensoRF



*An example of **Linear Algebra(Tensor Decomposition) in Neural Networks***

Survive the "Math and Science Death March"

MEGVII 旷视

- Linear Algebra
  - Neural Network Compressions: Low Rank, Sparse
  - Tensor Data Analysis
- Multivariate Calculus
  - Gradient Descent
  - Bitwidth Sensitivity Analysis
- Numerical Methods
  - Neural PDE, Physics Informed Neural Network
  - Stereo Vision (Bundle Adjustment)

Imagined by DDIM for ".latentdiffusion a hedgehog using a calculator"

Learn to be a "Priest"

MEGVII 旷视

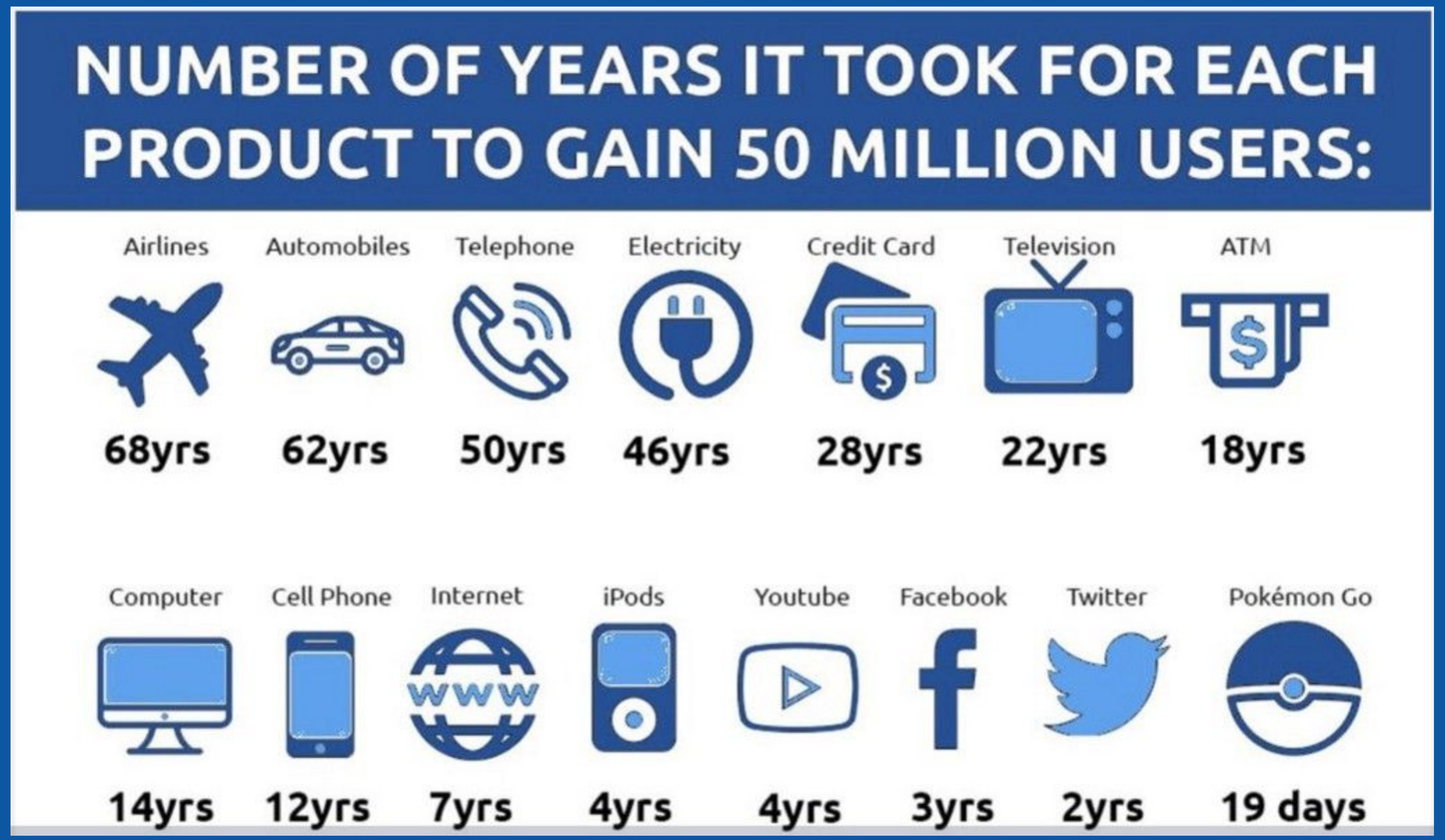
- An Architect needs to invent a world when *designing*
  - ***Complexity***: How to curb the entropy of the design? what can become more complex, and what cannot?
  - ***Differentiating***: What makes our world different?
  - ***Compatibility***: How to communicate with existing world? What are the IO's?
  - ***Growth***: How to grow user base? What are the target popul
  - ***Consensus Mechanism***: How to resolve conflicts?
  - ***Roadmap:*** How others can align with the evolution of your

Imagined by DDIM for ".latentdiffusion an immortal sacred robot priest mandating future of artificial intelligence"

Learn to be a "Community Player"

MEGVII 旷视

- Old School
  - Compiler cannot change code
  - Cascade development flow
  - Conference & Journal
- New School
  - Compiler can offer suggestions
  - User Community
    - User code contributions
    - Peer-to-peer helping
  - Micro-iterations
  - Arxiv / GitHub / Twitter



The age of instant response

Takeaways

MEGVII 旷视

- AI grows faster: Computer Vision, Computer Graphics and NLP meets each other.
- Accelerator and AI boosts each other by co-design.
- Advices
  - Survive the "Math and Science Death March"
  - Learn to be a "Priest"
  - Learn to be a "Community Player"
  - https://paperswithcode.com

https://zsc.github.io/

MEGVII 旷视

以非凡科技，为客户和社会持续创造最大价值

谢谢

Backup after this slide