



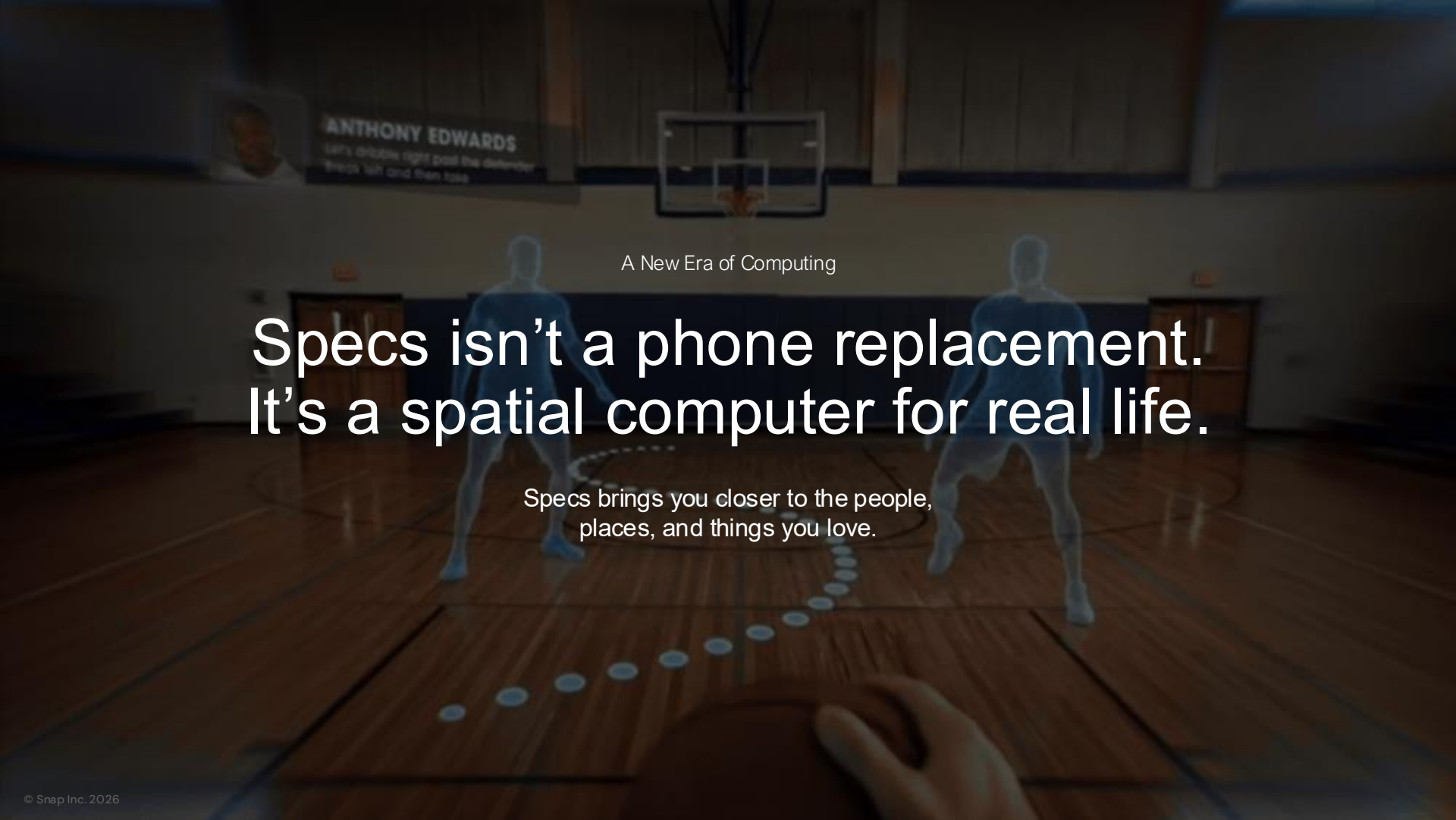
S P E C S

Computing Without Looking Away

Orlando Moreira

Chief Core Architecture, SNAP SPECS

Part-time Professor, Eindhoven University of Technology

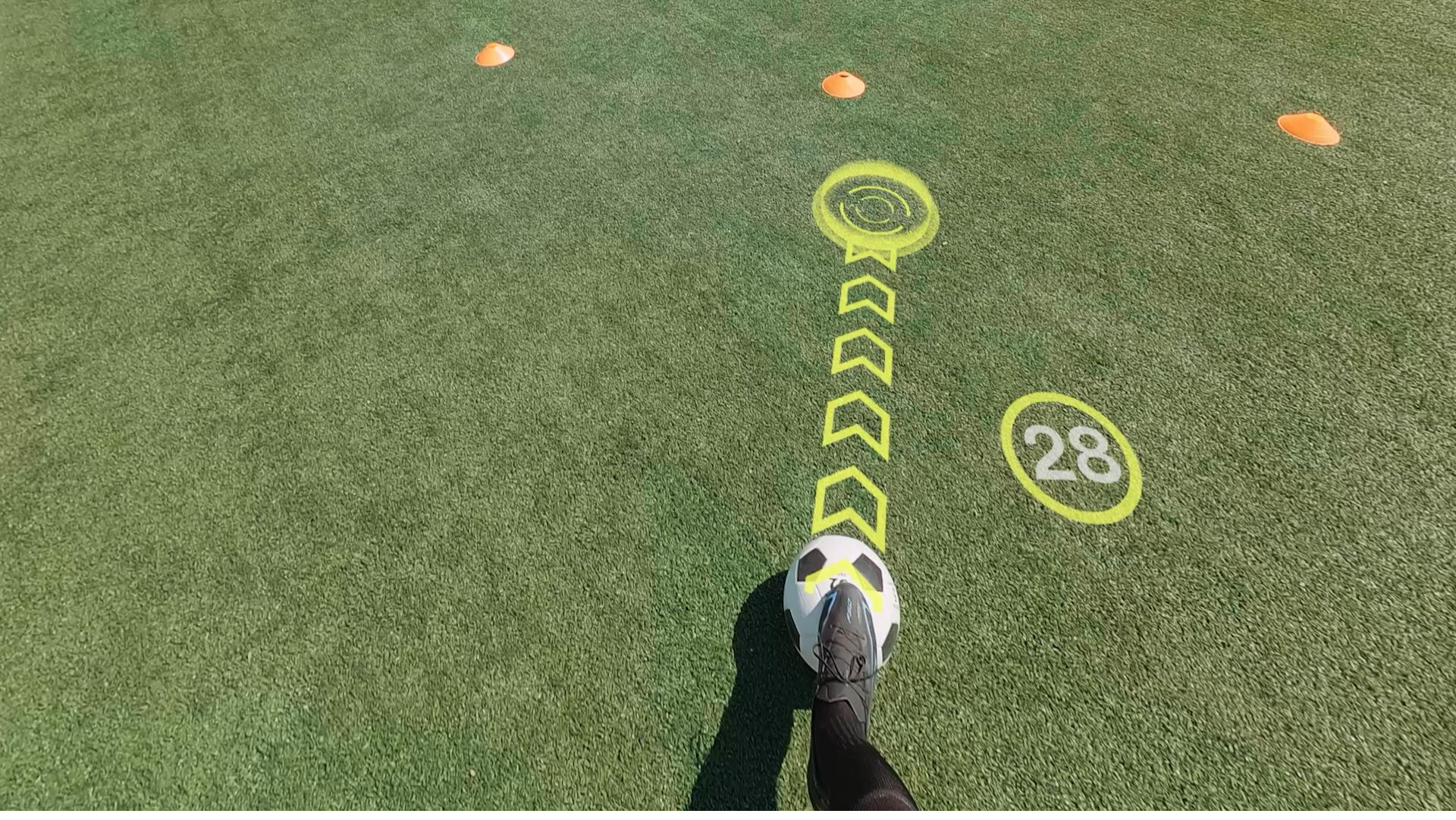


A New Era of Computing

Specs isn't a phone replacement.
It's a spatial computer for real life.

Specs brings you closer to the people,
places, and things you love.





What Makes Specs Different

New Era of Spatial Computing

Snap has spent more than a decade building for AR, giving Specs a real-world advantage from day one.

A New Device Category

Specs aren't smart glasses. They're a spatial computer you wear, designed for life, not screens.

Truly Spatial

Not floating screens or single-eye displays. Specs use a full-display AR canvas to anchor digital experiences naturally in the world around you.

Human-First Experiences

Built to support presence, not compete with it. Technology that enhances connection, creativity, and capability without pulling you out of the moment.

Real-World Awareness

Specs understand space, context, and motion. Digital information shows up where it's useful, blending seamlessly with the physical world

S P E C S



What It Takes: Perception That Never Sleeps

Every experience you just saw depends on perception that runs all the time, on your face

ALWAYS ON

- **6DoF head tracking**
where am I, where am I looking
- **Hand tracking**
the primary input, at 240 FPS
- **Scene understanding**
surfaces, objects, context
- **Contextual AI**
what do I need right now

CONTENTS

- 01 Always-On AI and Its Constraints
- 02 Why Sparsity
- 03 Neural Accelerator Architecture
- 04 Neural Architecture Search
- 05 Temporal Sparsity
- 06 What about Transformers?
- 07 Conclusions
- 08 Technology and Humanity

The constraint: all of it in a pair of glasses, at less than 1 W per SoC and cool enough to wear

01

Always-On AI and Its Constraints

- 01 Always-On AI and Its Constraints
- 02 Why Sparsity
- 03 Neural Accelerator Architecture
- 04 Neural Architecture Search
- 05 Temporal Sparsity
- 06 What about Transformers?
- 07 Conclusions
- 08 Technology and Humanity

Contextual Always-On AI

Your glasses infer what you need from context and offer help before you ask.

1 Sense

Cameras, IMU and microphones run all the time

2 Understand

Where am I? What am I looking at?

3 Anticipate

You're at a train station: you'll want the departure times

4 Act

Proposes an action: show the next departures

Always on → it only works if the AI runs at **very low power**

Always-On Perception Requires **Restraint**

01

Respect the user

Ask first. Let people control what is connected, stored, shared or deleted.

02

Respect the people around them

Understand without recording. Make recording visible.

03

Minimize by architecture

Process locally. Expose understanding, not raw sensor data.

04

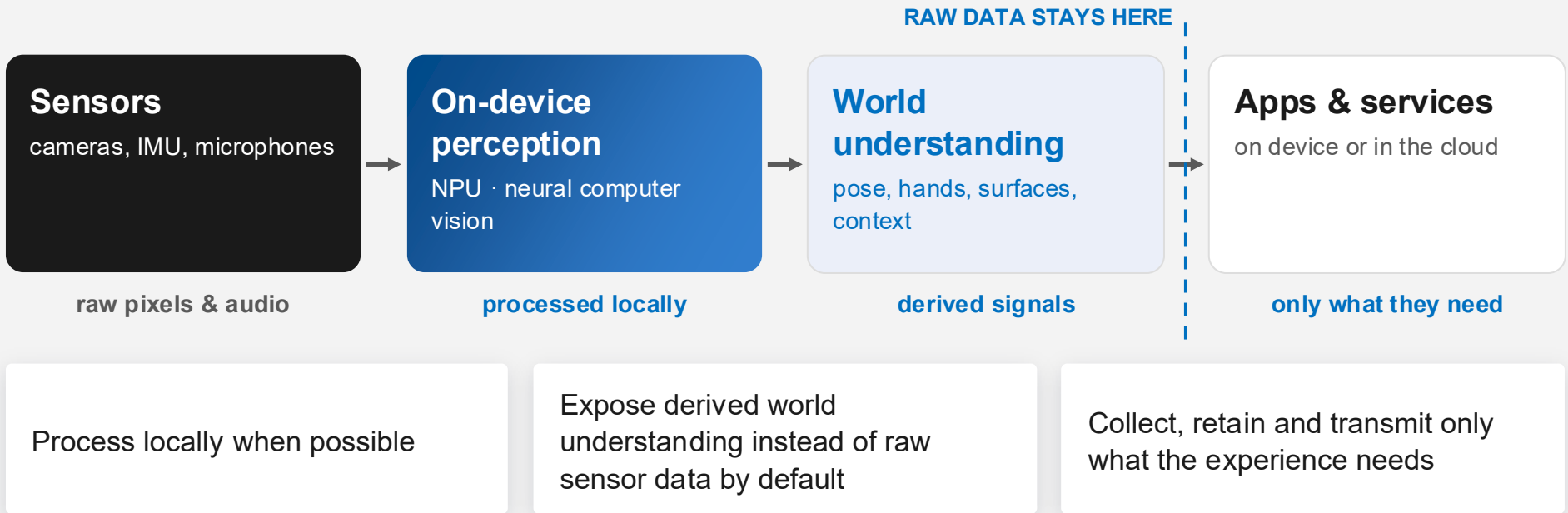
Know when not to intervene

Help when useful, otherwise stay in the background.

Trust is a system property: what we sense, infer, expose, retain, transmit, and when we choose to do nothing

Minimize by Architecture

Privacy is designed into the data path, not added on top of it



Efficient on-device AI is a privacy feature: the less leaves the glasses, the less can leak

Challenges: Thermal Envelope, Energy Consumption

01

~40 °C

Close to the skin

Frame temperatures must remain comfortable

02

mm-thin

Limited cooling space

Compact frames need effective heat spreading, such as vapor chambers

03

+ sun

Outdoor operation

Warm weather and direct sunlight reduce thermal headroom

04

< 1 W

Power budget

Sustained performance requires power-efficient computation and workload allocation

Heat is the real budget: every milliwatt saved is thermal headroom for always-on AI

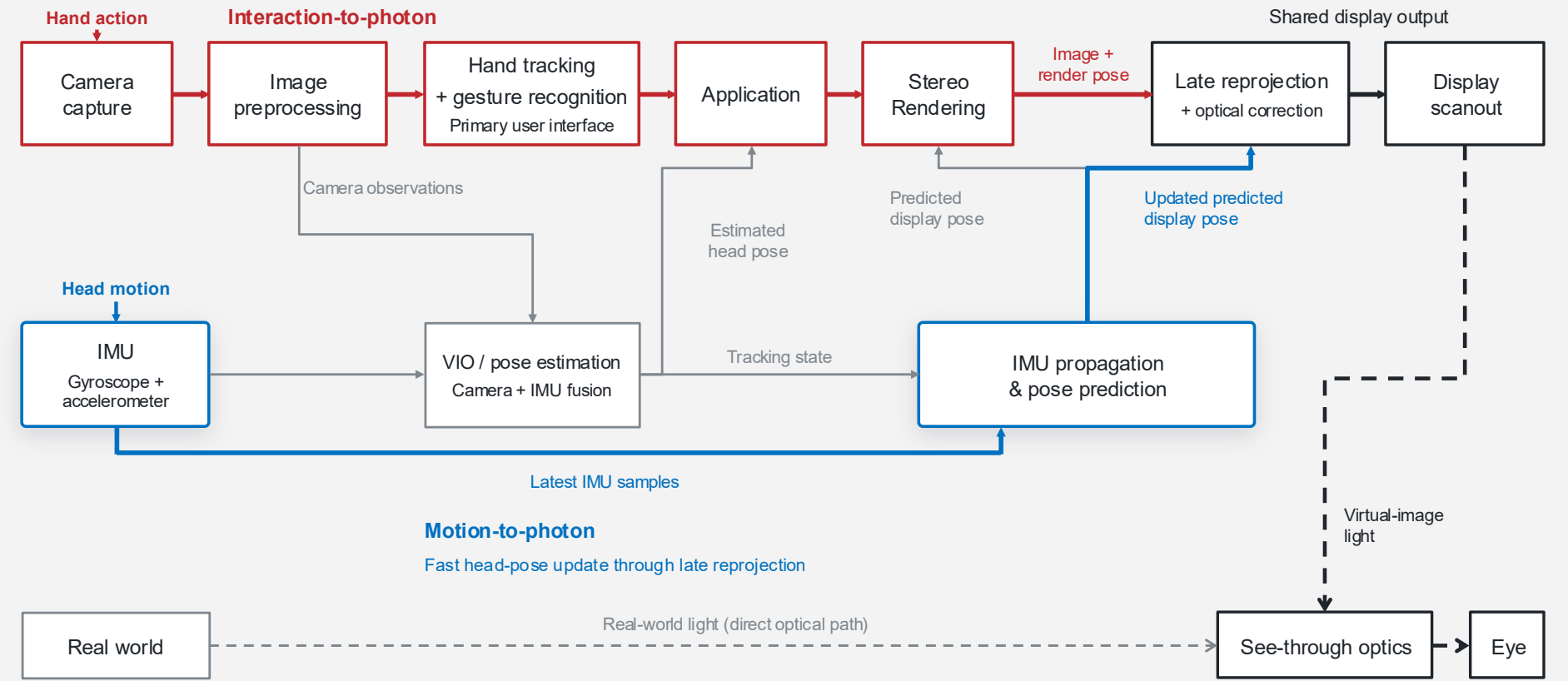
Optical see-through AR: Latency paths

Interaction-to-photon

Motion-to-photon

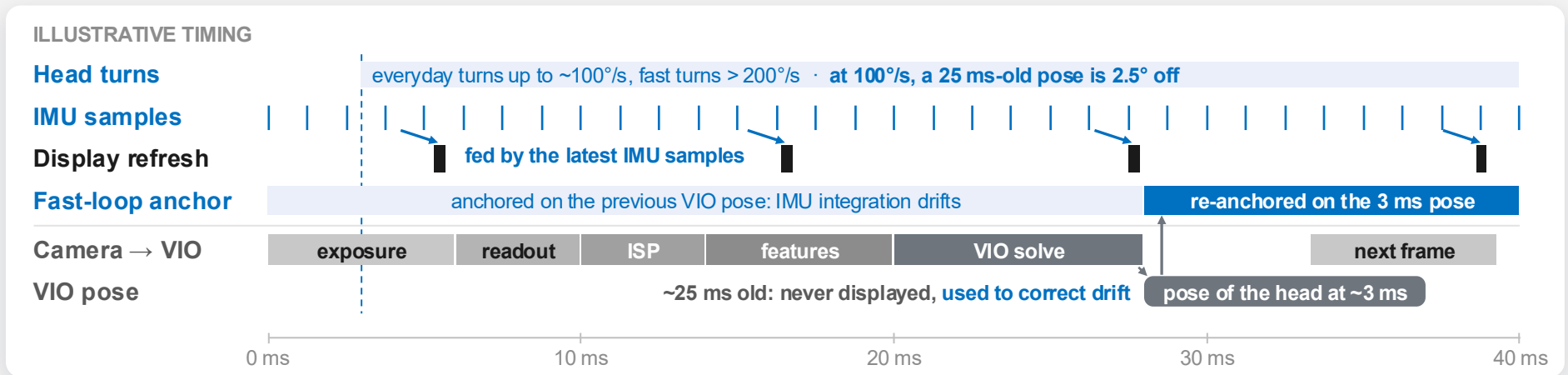
< 10 ms

< 5 ms



Disclaimer: diagram and numbers are indicative, do not necessarily match SPECS product.

Two Loops: Fast but Drifting, Accurate but Late



Disclaimer: diagram and numbers are indicative, do not necessarily match SPECS product.

Fast loop · IMU → propagation → reprojection

- Hundreds of Hz, almost instant; predicts to display time
- Drifts on its own within a fraction of a second

Sets motion-to-photon latency

Slow loop · camera → IPU → VIO

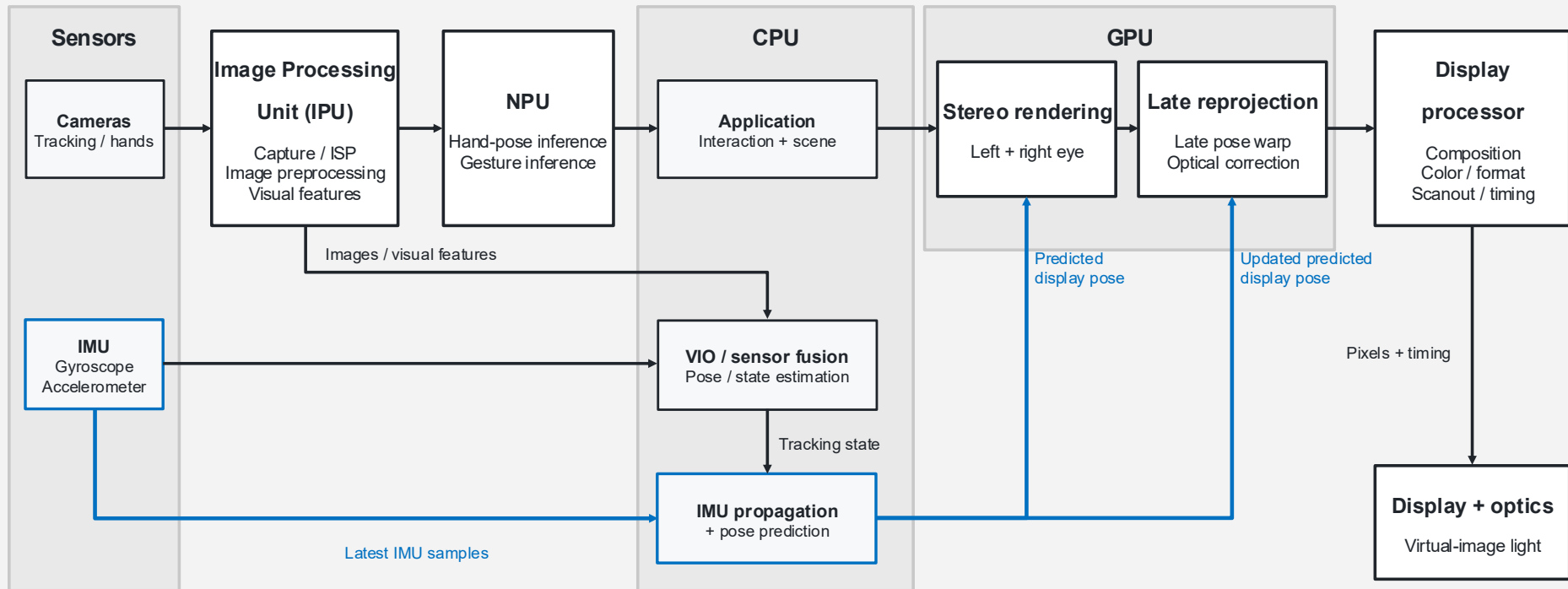
- Tens of Hz; every stage adds delay before the pose exists
- Drift-free and world-anchored, but tens of ms late

Sets stability: re-anchors the fast loop

Responsiveness from the IMU, stability from the cameras.

SoC: hardware deployment of optical see-through AR

Representative allocation of the processing functions: does not necessarily correspond to current deployment in SPECS HW.



Illustrative mapping. Work may move between CPUs, DSPs / MCUs and accelerators depending on the SoC.
Arrows show selected logical data flows. Shared memory, interconnect and display-timing feedback are abstracted.

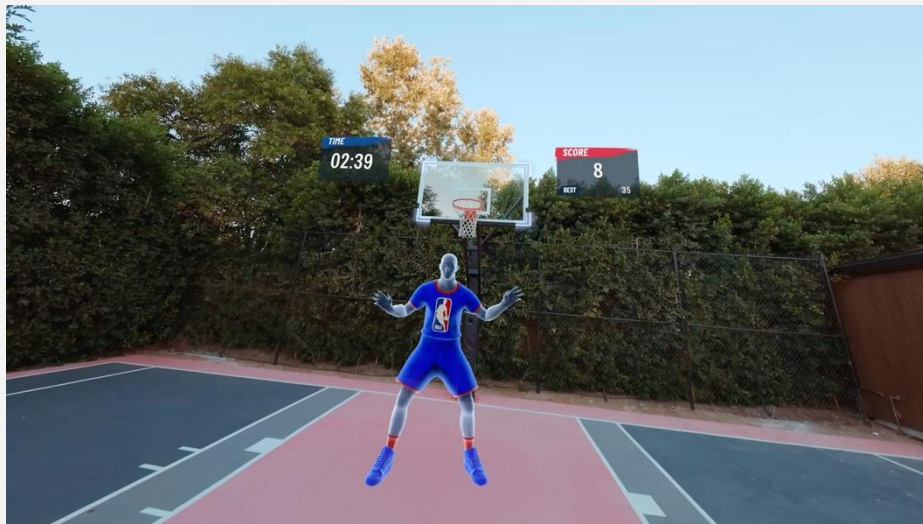
02

Why Sparsity

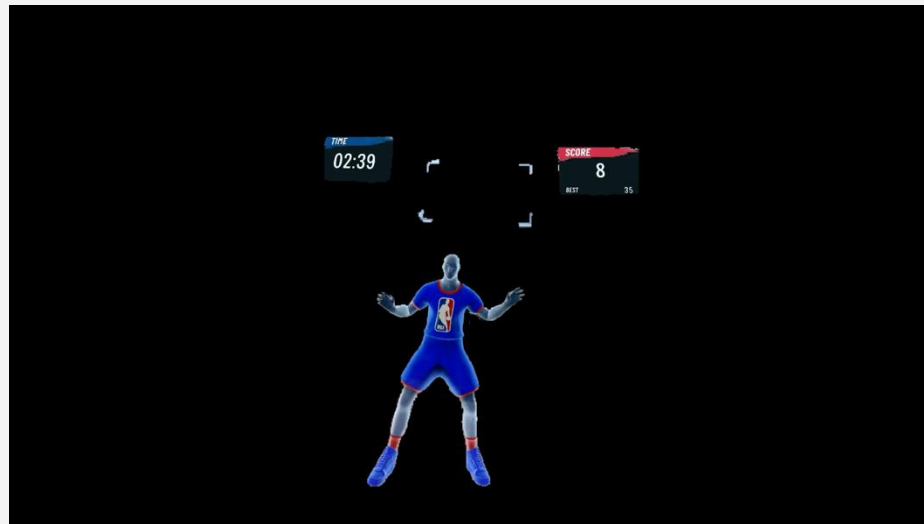
- 01 Always-On AI and Its Constraints
- 02 **Why Sparsity**
- 03 Neural Accelerator Architecture
- 04 Neural Architecture Search
- 05 Temporal Sparsity
- 06 What about Transformers?
- 07 Conclusions
- 08 Technology and Humanity

Spatial Sparsity in the Output

What you see



What is rendered and displayed



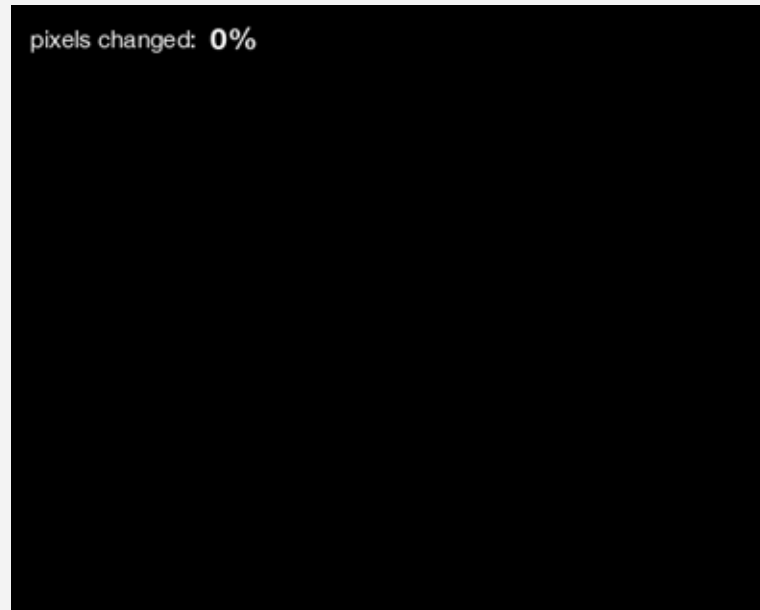
AR output is naturally sparse → most display pixels stay dark

Temporal Sparsity at the Sensor

Camera frames



Difference between consecutive frames



Little changes from frame to frame → only process what changed

Why Sparsity?

Do less work: save time and energy

Exploiting sparsity reduces latency and energy together.
Many traditional optimizations trade one for the other, e.g. clock frequency

Sparsity follows the dynamics

Work scales with what changes in the scene, not with the frame size

SPARSITY APPLIES ACROSS THE WHOLE PIPELINE

Sensor capture

Sensor processing

Neural computer vision

Rendering

Display

▲ focus of this talk

Next: sparsity in neural computer vision, and the neural processing unit that exploits it

03

Neural Accelerator Architecture

- 01 Always-On AI and Its Constraints
- 02 Why Sparsity
- 03 Neural Accelerator Architecture**
- 04 Neural Architecture Search
- 05 Temporal Sparsity
- 06 What about Transformers?
- 07 Conclusions
- 08 Technology and Humanity

Example Workload: Hand Tracking

Hand tracking is the primary input of Specs, so it runs almost all the time



2 hands

tracked simultaneously

10 mm

accuracy, ~2 mm jitter

< 10 ms

latency per frame

240 FPS

4 cameras × 60 FPS

< 250 mW

power budget

Always on

indoors & outdoors

Neural Networks: Sparsity

Activation sparsity

~50%

of activations are zero with ReLU: irrelevant data yields 0-valued activations. Can be increased with training for activation suppression.

Temporal sparsity



Little changes from instant to instant, so why re-process the whole image?

Full frame



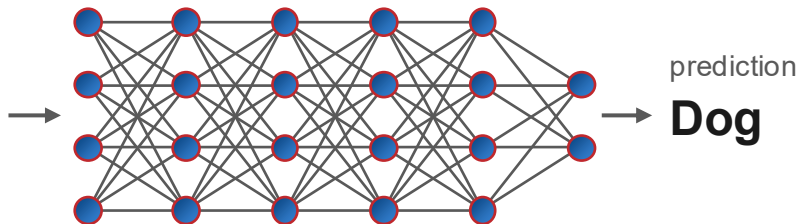
Difference consecutive frames



Sparse Neural Network Inference

Dense: process everything

GPUs, CPUs

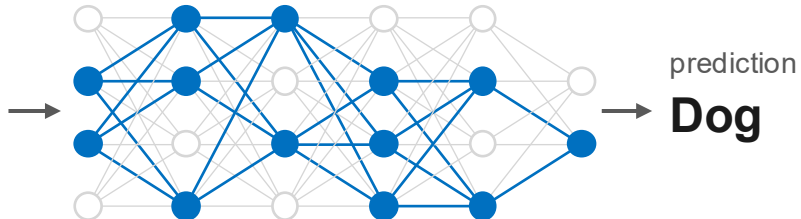


✗ Redundant

✗ Inefficient

Sparse: process only non-zeros

Neuronflow




Low power


Low latency

Fewer activations → less compute load

Consequences of Activation Sparsity for Computer Architecture

01

Irregular compute demand

Breaks sequential memory access; sporadic, non-deterministic activity

02

Skip activations, don't compute them

Needs input-stationary execution (next slide)

03

Resilient neuron state

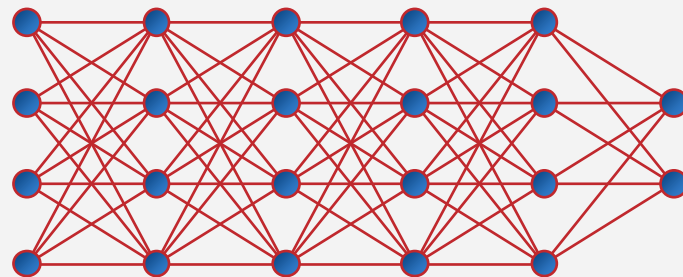
Temporal sparsity stores state from frame to frame

04

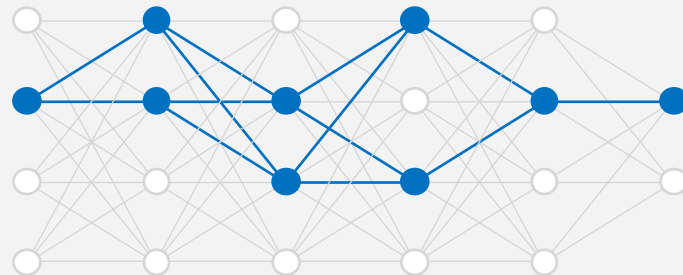
Network and Training matter

The price to pay for an unconventional architecture

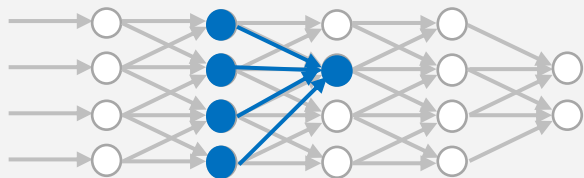
Dense every neuron computes



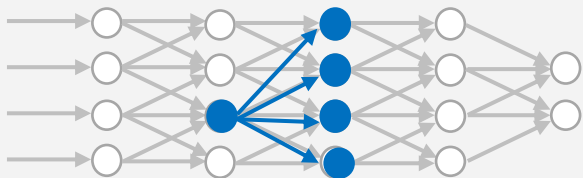
Sparse only some neurons fire, data-dependent



Convolution Execution Order

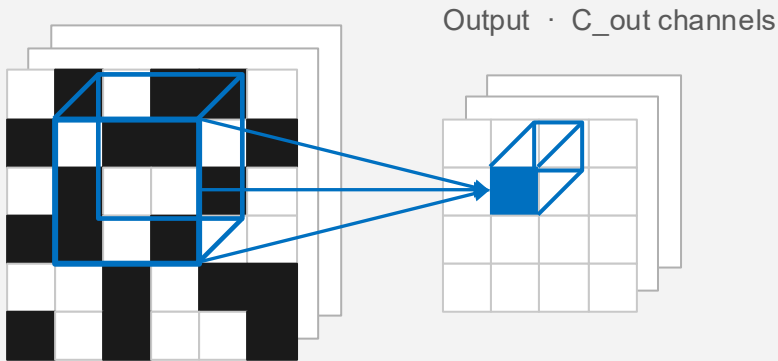


Input · C_{in} channels



Input · C_{in} channels

Output · C_{out} channels

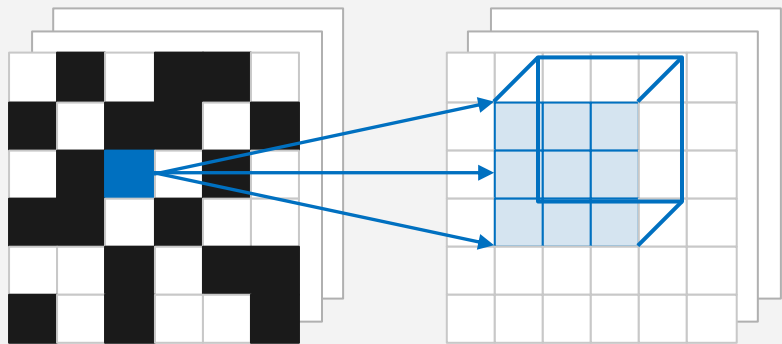


■ Zero activation (50%) □ Non-zero activation

Standard convolution

output-centric

Each output reads a $k \times k$ window over all input channels, zeros included



■ Event being processed

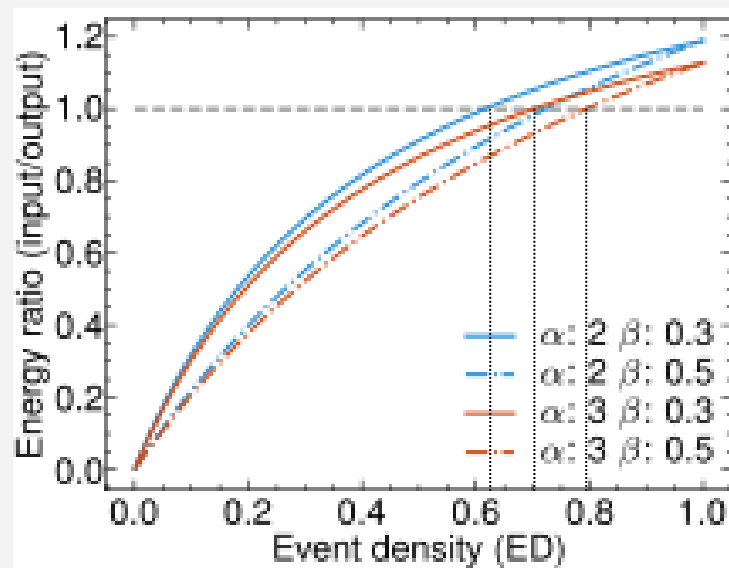
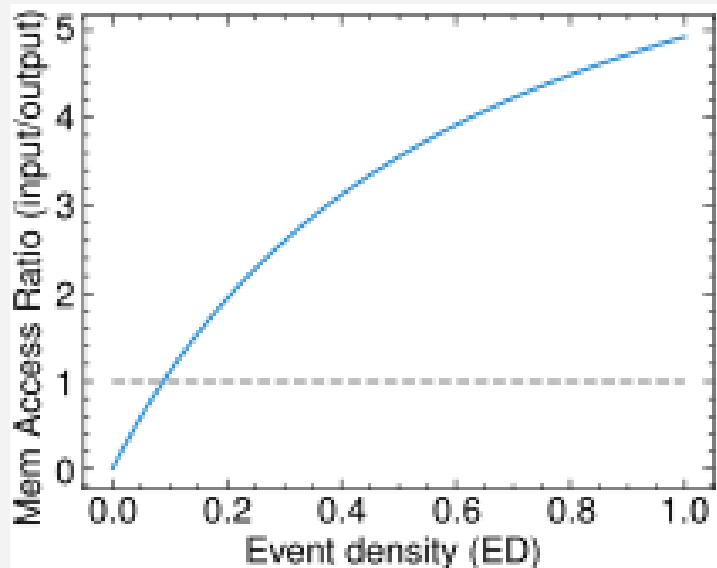
Event-driven convolution

input-centric

Only non-zeros become events; each updates a $k \times k$ region in all output channels:

Easy 0 skip, but more read-writes.

Event-Driven vs Standard Convolution with Sparsity

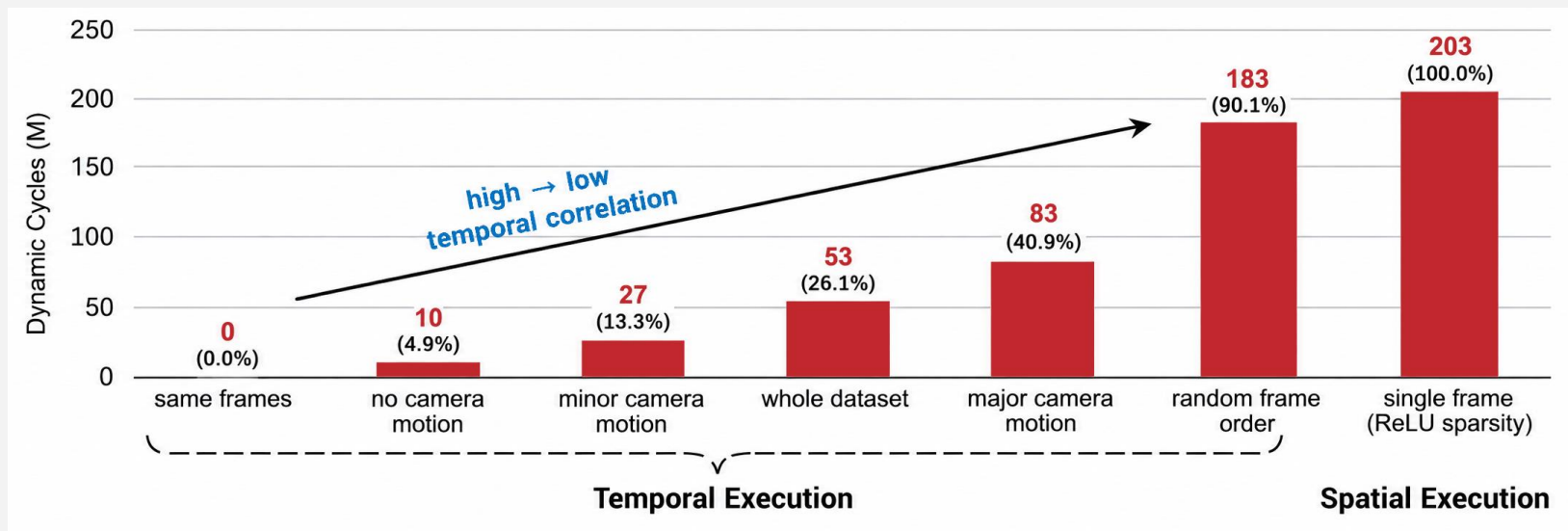


Alpha: logic energy/memory energy

Beta: nop logic energy/avg logic energy

Event-driven: more memory accesses, but much lower energy with sparsity!

Temporal Sparsity: Hand-Tracking Dataset on NAS-Designed Network



In parentheses, we show dynamic cycle density wrt the **sparse** baseline of 203 Mcycles.

Neuronflow Array

12 × 12

cores

2 × 4

SIMD · V2

4 × 4

SIMD · V3

Local memory per core

Weights and neuron state stored next to the compute

Packet-switched NoC

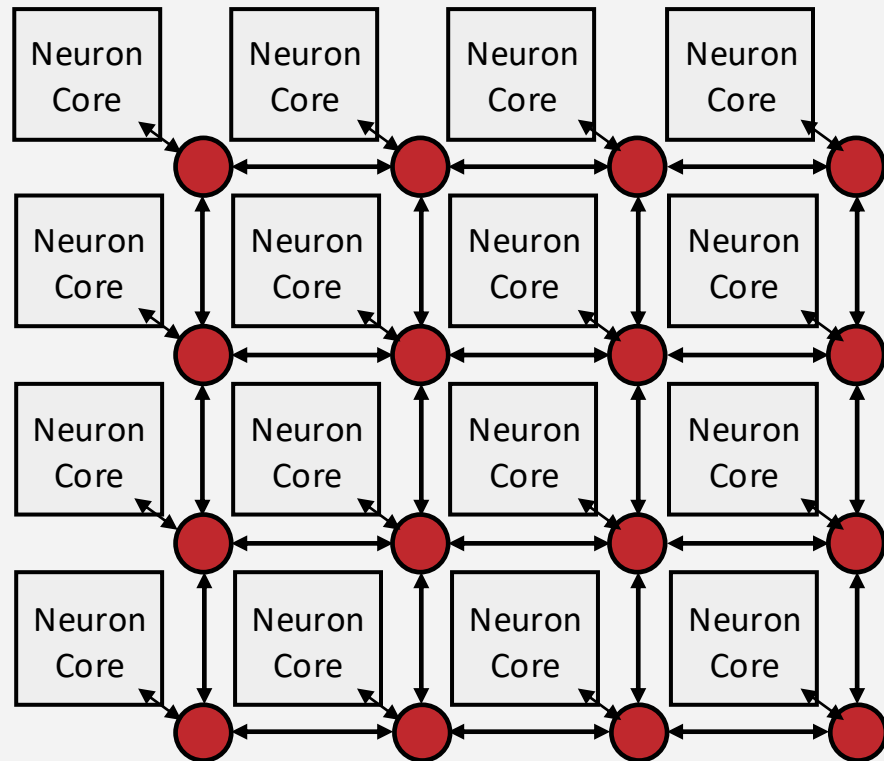
Cores exchange events, not bulk data

Event-driven, sparse schedule

In-place state updates; no bulk data movement

Dataflow

The arrival of input data triggers execution



Neuronflow Core: Four-Stage Event Pipeline

1

Event Queue

Buffers incoming events and picks what runs next

2

Event Decoder

Decodes 2 events in parallel, fetches weights and states

3

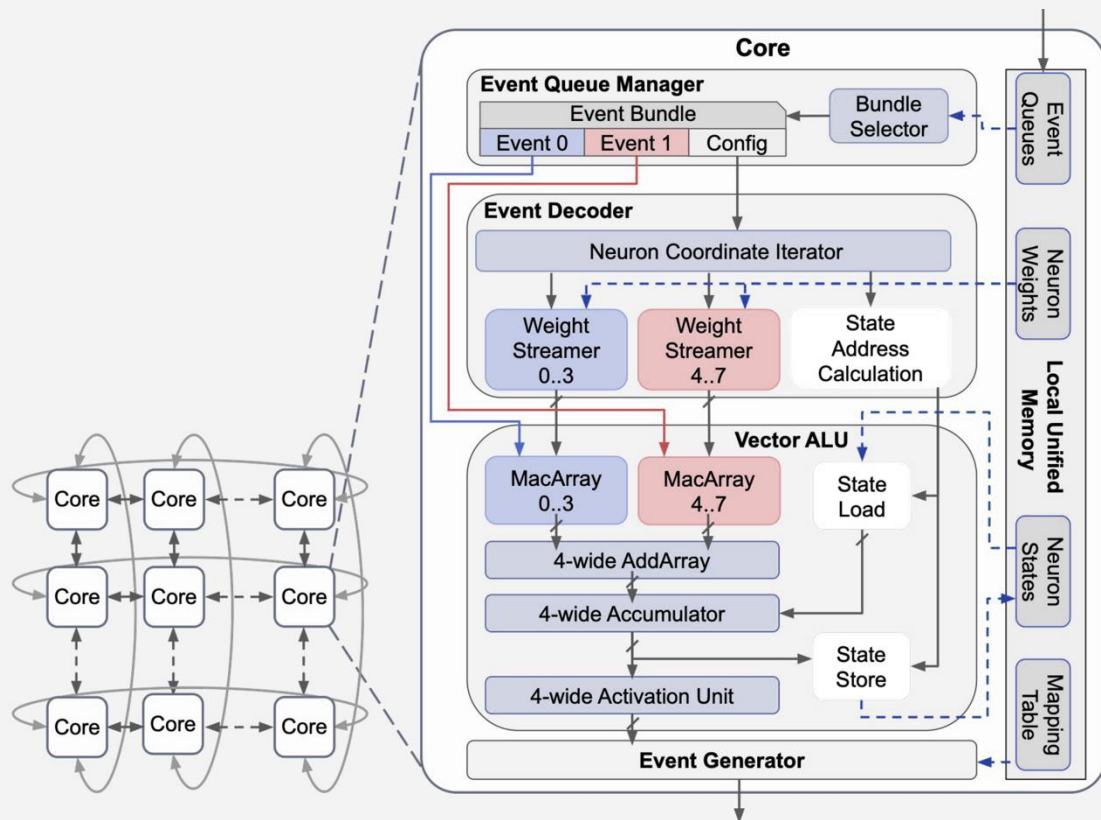
Vector ALU

8 MACs per cycle, then activation

4

Event Generator

Sends non-zero outputs to the next cores



[Read the paper](#)

Neuronflow V2: Sparse and Energy Efficient Event-driven AI Processing

Moreira et al., IEEE Micro 2026 (Fig. 2)
doi.org/10.1109/MM.2026.3722146

Impact of Training on Performance

Weight quantization & pruning

2–16 bit

weights: fewer computations and less memory. Fits larger networks, and less power spent reading weights

Activation suppression

**up to
2×**

fewer events out of the box with ReLU; a further 1.4–1.7× when trained for sparsity. A suppressed event skips all its computation

Temporal sparsity

> 2×

energy and performance on top of activation sparsity, depending on input dynamics. Costs ~5× state memory on mobile CNNs, so use sparingly

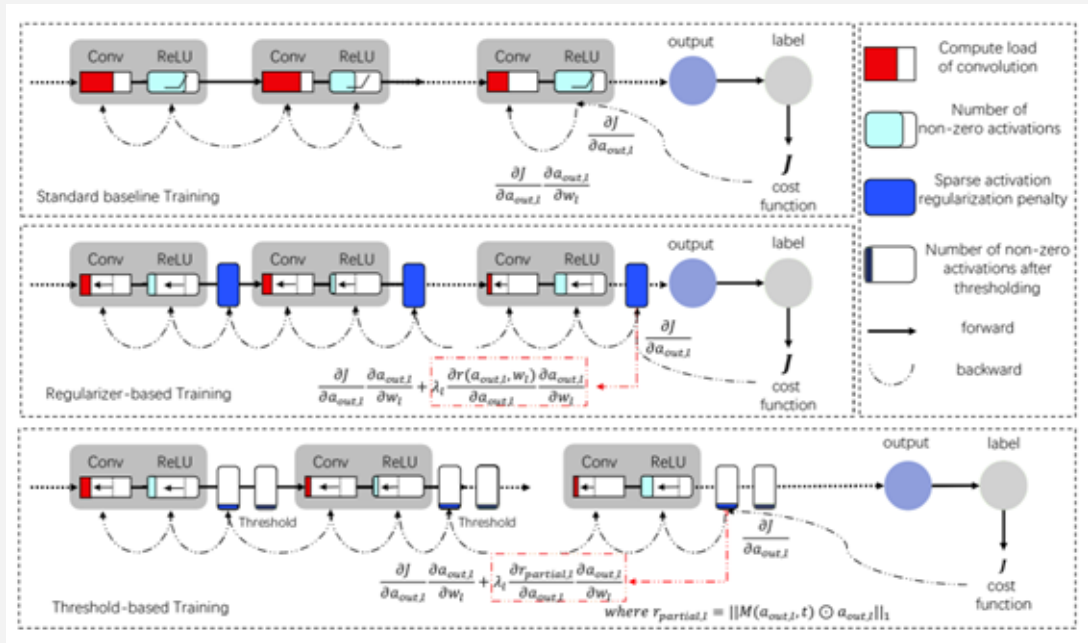
Neural architecture search

~2×

energy and latency gains by redesigning the network for the Neuronflow architecture

Best combined: temporal sparsity + NAS

STAR: Training to Induce Activation Sparsity



1

Baseline

Train normally to establish the reference accuracy

2

Suppress

Force all activations lower: sparse, but accuracy drops

3

Recover

Let above-threshold activations (**only those**) grow back to recover accuracy.



[Read the paper](#)

STAR: Sparse Thresholded Activation under Partial-Regularization

Zhu et al., CVPR Workshops 2023 · doi.org/10.1109/CVPRW59228.2023.00479

Activation Suppression Results (Neuronflow V2, 5 nm)

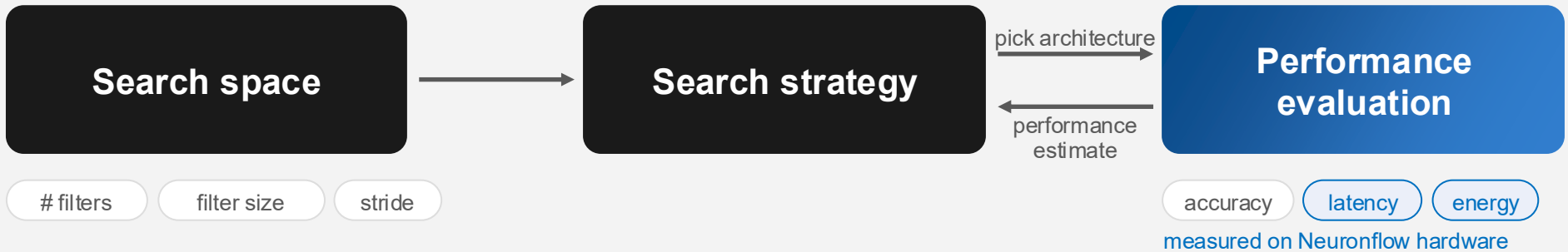
Sparsity	Top 1 Acc	Compute Density	Latency Reduction	Act.Power Reduction
RESNET50 on ImageNet				
Natural	75.8%	39.3%	1.0x	1.0x
Induced	75.2%	22.7%	1.7x	1.7x
MobileNetV1 on ImageNet				
Natural	71.0%	57.9%	1.0x	1.0x
Induced	71.0%	34.9%	1.6x	1.7x
EfficientNetLiteB0 on ImageNet				
Natural	74.1%	69.0%	1.0x	1.0x
Induced	73.9%	49.2%	1.4x	1.4x

04

Neural Architecture Search

- 01 Always-On AI and Its Constraints
- 02 Why Sparsity
- 03 Neural Accelerator Architecture
- 04 Neural Architecture Search**
- 05 Temporal Sparsity
- 06 What about Transformers?
- 07 Conclusions
- 08 Technology and Humanity

Neural Architecture Search



Let the search find the network

Fit your dataset and hardware: move from network-centric to data-centric specification.

NAS for event-based processors

Add processor and mapping KPIs to the evaluation; redesign networks to fit our hardware.

Evaluation is the hard part

Compiler + hardware in the loop is slow.

Need to develop models/performance proxies.

Why Do We Need NAS?

Hardware has preferences

Different platforms favour different operations

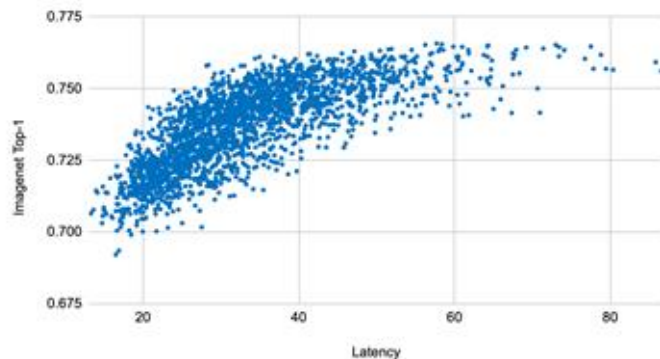
Neuronflow is non-standard

Mobile network designs may not play to its strengths

Hardware-aware NAS

Searches architectures under the constraints of the target platform

Accuracy vs Latency (GrAI-VIP)



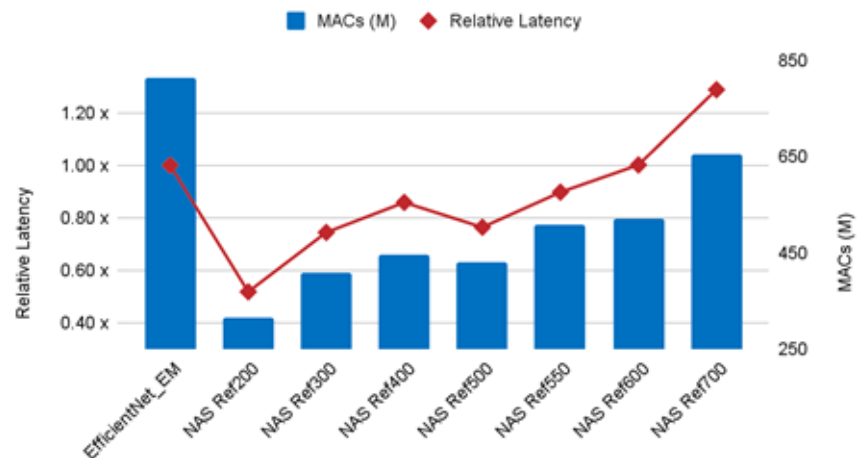
Model	Top-1 (%)	GPU latency	CPU latency	Mobile latency
Proxyless (GPU)	75.1	5.1 ms	204.9 ms	124 ms
Proxyless (CPU)	75.3	7.4 ms	138.7 ms	116 ms
Proxyless (mobile)	74.6	7.2 ms	164.1 ms	78 ms

Each network is fastest on the hardware it was searched for

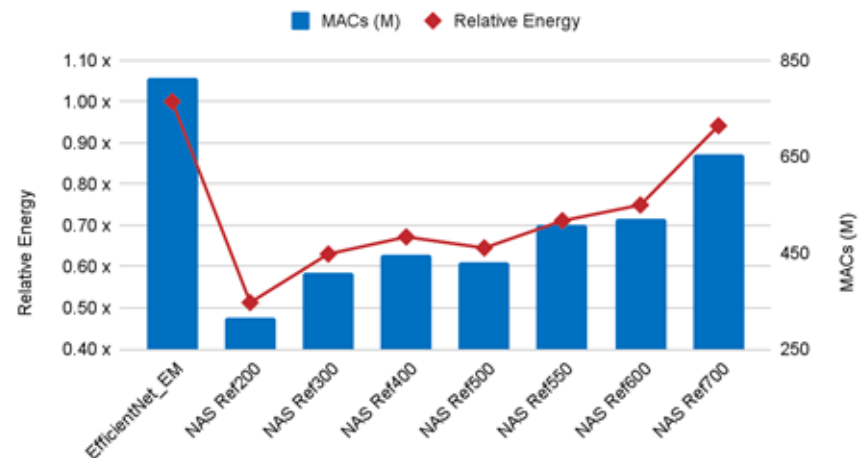
ProxylessNAS, Cai et al., ICLR 2019

NAS Results

Relative Latency



Relative Energy



Measured on the Neuronflow V2 chip, relative to EfficientNet_EM on FreiHand

NAS for Neuronflow: close to 2× energy and latency gains

05

Temporal Sparsity

- 01 Always-On AI and Its Constraints
- 02 Why Sparsity
- 03 Neural Accelerator Architecture
- 04 Neural Architecture Search
- 05 Temporal Sparsity**
- 06 What about Transformers?
- 07 Conclusions
- 08 Technology and Humanity

MEET: Towards MEemory-Efficient Temporal Neural Networks



Problem

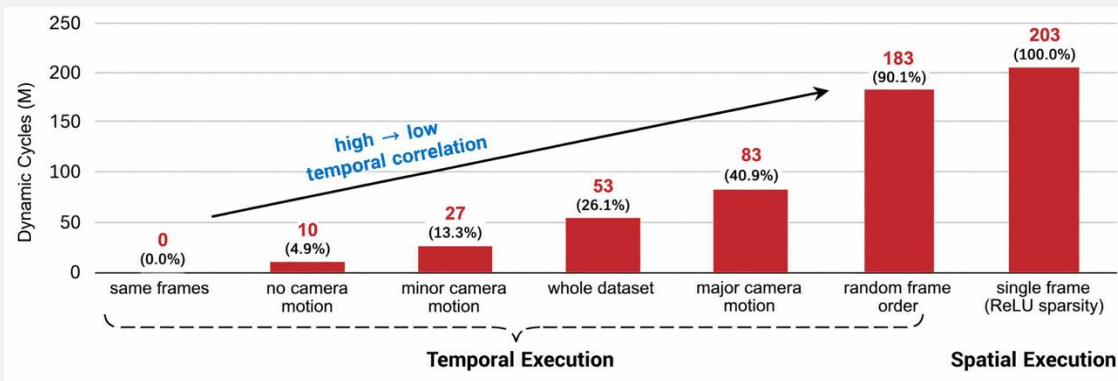
Temporal NNs (TNNs) exploit temporal redundancy: **2.5× to >10× less computation**

But they need **~5× memory** to store full feature maps, so they don't fit the on-chip memory



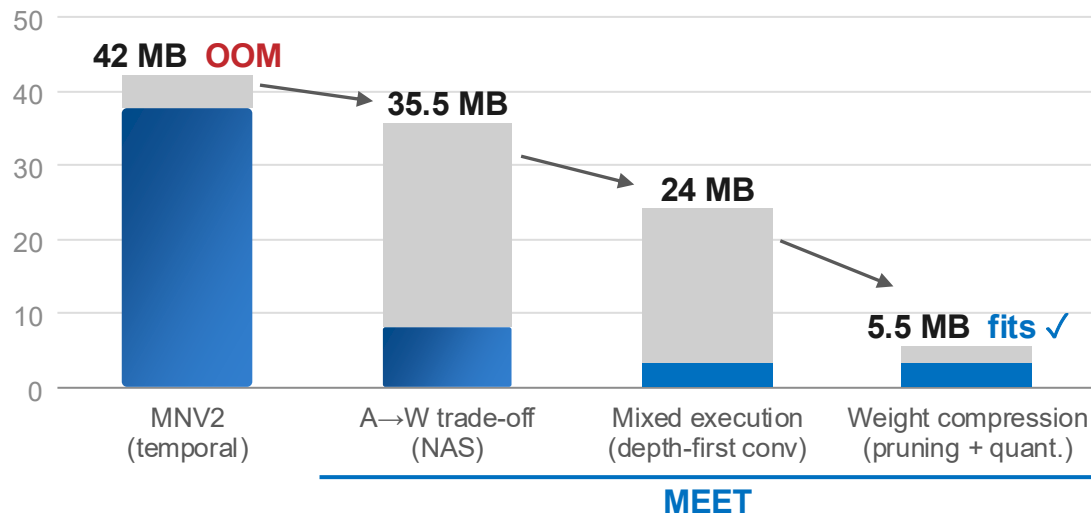
Idea

Reduce **state memory** costs by increasing **weight memory** costs



MEET: Towards MEemory-Efficient Temporal Neural Networks

Memory usage (MB) ■ state (hard to compress) ■ weight (easy to compress)



42 → 5.5 MB

temporal MobileNetV2: memory needed, before and after MEET

The problem

External memory cost negates the energy savings from sparse compute

MEET

NAS redesigns neural networks **trading off weights for activations**. Significantly reduces **memory cost**, while keeping accuracy and efficiency.



[Read the paper](#)

MEET: Towards Memory-Efficient Temporal Sparse Deep Neural Networks

Zhu et al., CVPR 2025 · doi.org/10.1109/CVPR52734.2025.02729

MEET Results: Hand Detection on Neuronflow V2

Method	Compute Density	Event triggered MACs (Millions)	State Memory (MB)	Latency Reduction	Active Power Reduction
MobileNetV1+SSD on EgoHands (Neuronflow V2)					
Natural	46%	138	0	1.0x	1.0x
Induced	26.5%	78 (1.8x)	0	1.5x	1.7x
Temporal	11.3%	34 (3.8x)	11.0	2.1x	2.9x
MEET	11.3%	48 (2.9x)	3.2	1.9x	2.4x
MobileNetV2 +SSD on EgoHands (Neuronflow V2)					
Natural	72%	125	0	1.0x	
Induced	40%	69 (1.8x)	0	1.6x	1.7x
Temporal	25%	43 (2.9x)	16.9	OOM	OOM
MEET	18%	43 (2.9x)	4.8	1.8x	2.7x

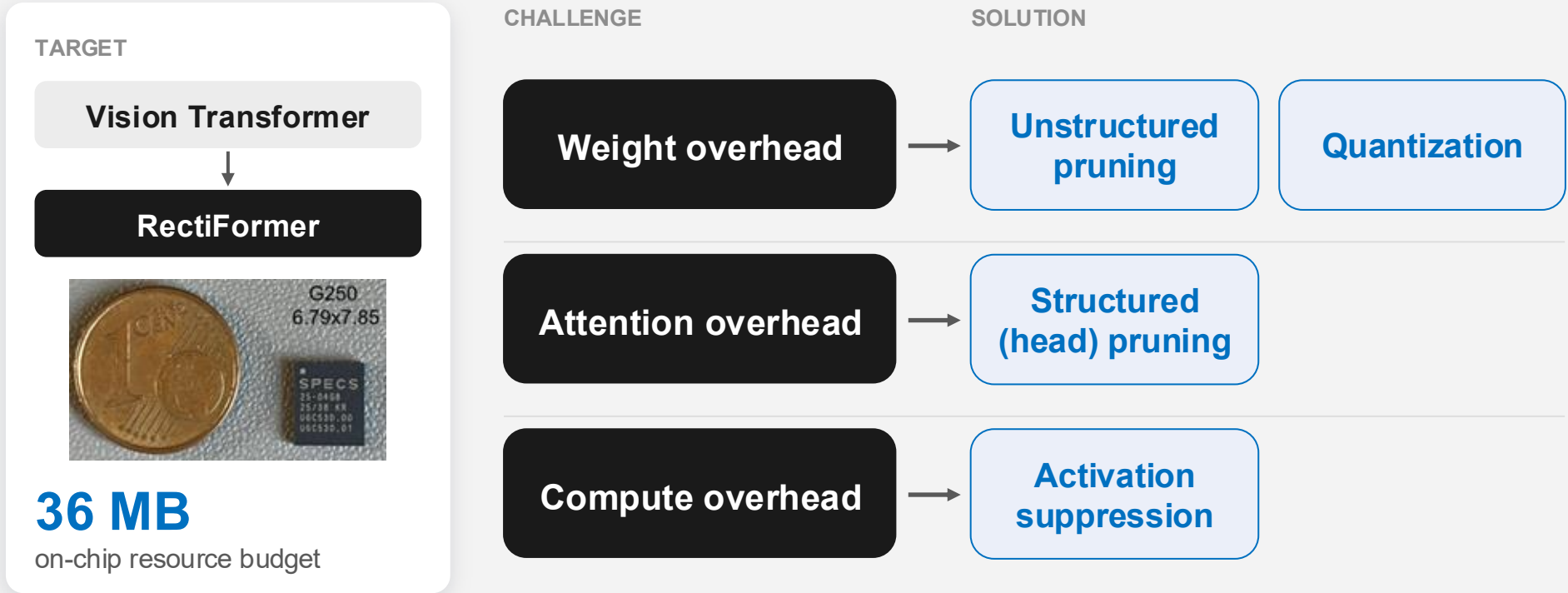
06

What about Transformers?

- 01 Always-On AI and Its Constraints
- 02 Why Sparsity
- 03 Neural Accelerator Architecture
- 04 Neural Architecture Search
- 05 Temporal Sparsity
- 06 What about Transformers?**
- 07 Conclusions
- 08 Technology and Humanity

Transformer Optimization

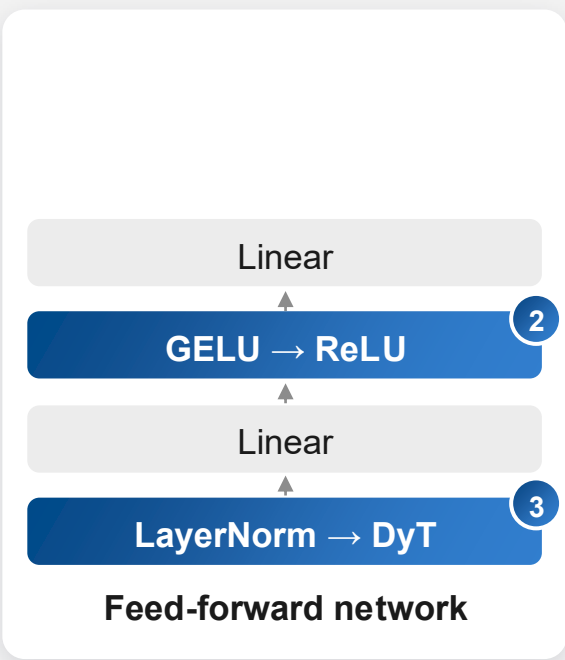
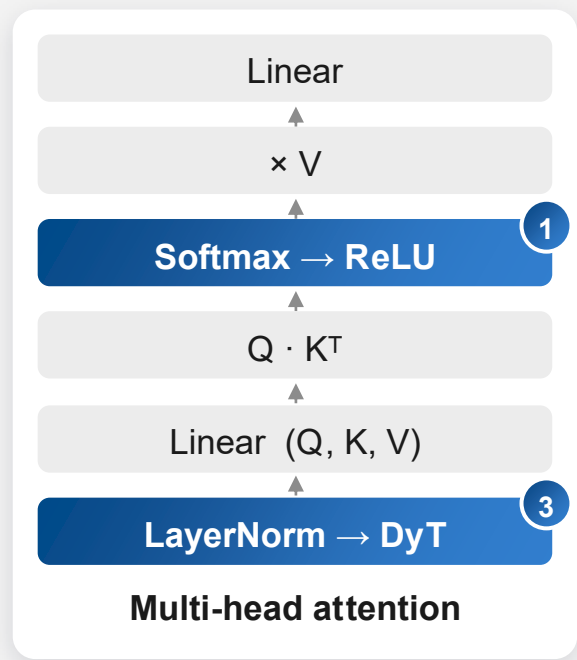
Goal Enable vision transformers on Neuronflow V2, and improve their performance



RectiFormer

Original Vision Transformer → Neuronflow V2 compatible

Neuronflow V2 was designed for CNNs only: the exercise was to prove transformers could be supported as part of a product



- ① Softmax → ReLU
- ② GELU → ReLU
- ③ LayerNorm → Dynamic Tanh

Model	Latency Reduction	Power Reduction
Sparse Baseline	1.00x	1.00x
Induced Sparsity	2.22x	1.56x

- ✓ Accuracy preserved
- ✓ Sparsity gains similar to CNNs
- ✓ Sparse base ~30% sparser than original ViT-tiny

07

Conclusions

- 01 Always-On AI and Its Constraints
- 02 Why Sparsity
- 03 Neural Accelerator Architecture
- 04 Neural Architecture Search
- 05 Temporal Sparsity
- 06 What about Transformers?
- 07 Conclusions**
- 08 Technology and Humanity

Conclusions: Sparse Neural Acceleration

01

Event-driven execution pays off

- Very low latency and energy by exploiting activation and temporal sparsity

02

Training and network design multiply the gains

- Induced sparsity through training
- NAS for low latency and low power
- NAS to shrink state, unlocking cheap temporal sparsity

03

Vision Transformers work too

- Sparsity is still there to be exploited
- Increased by accuracy-neutral adaptations

Conclusions: The Bigger Picture



The smartphone put our lives in our pockets. Specs put computing into the world.

Evan Spiegel, Snap CEO

01

A new kind of computer

Computing in the world, not on a screen: look up, stay present

02

Instinctual and contextual

AI that understands what you see and offers help at the right moment

03

Shared and human

Experiences built for people together, making us more capable, not less present

04

Trust built in

Privacy designed in from the very beginning, across hardware and software

Engineering makes it wearable: power, heat and latency set what Specs can do. Sparse, event-driven AI is one of the levers

08

Technology and Humanity

- 01 Always-On AI and Its Constraints
- 02 Why Sparsity
- 03 Neural Accelerator Architecture
- 04 Neural Architecture Search
- 05 Temporal Sparsity
- 06 What about Transformers?
- 07 Conclusions
- 08 Technology and Humanity

When AI Enters the Social Situation

THE PROMISE

Understand what is happening around me

Help me without forcing me to look away

Reduce friction, compensate for limitations, support connection

THE RISK

Invisible mediation

am I talking to you, or to you plus an unseen coach?

Asymmetric perception

does one person know more about the other than the other knows?

Persistent context

does a passing moment become stored data?

Optimised interaction

does assistance become persuasion or manipulation?

The design question is no longer only “Can we do it?”

It is also what should the system perceive, infer, remember, reveal — and when should it do nothing?

Identify the failure states, and design around them.

Technology Amplifies: We Can Nudge It Toward Good

Builders don't decide how technology is used, but we can nudge it in the right direction.

IT CAN...

Isolate us

Leave people behind

Keep knowledge for a few

Turn us into passive consumers

...OR IT CAN

Bring us together

Help with our insufficiencies

Democratize knowledge, even reasoning

Empower and enable our creativity

Technology is neither good nor bad by itself. But we must never forget how much good it can do.

Solipsism or Relation: Technology Enables Both

TOWARD SOLIPSISM

Simulations built to centre me

feeds and virtual companions that only ever reflect me back

Others that can never refuse me

relation without risk, and so without weight

And AI?

if I spend more time talking to an AI than to my family...

TOWARD RELATION: WHAT IT DID FOR ME

My crisis, two years ago

I shared my despair openly, and so many answered

Games with my son in Portugal

distance stops being absence

Playing roleplaying games online

Games that keep people imagining together...

And AI can help.

“Only the real Other can refuse you.”

from Signals Across the Abyss



[Read the essay](#)

Signals Across the Abyss

Essentia Foundation · 2026



[Read the essay](#)

Meaning Is Not in Yourself

Essentia Foundation · 2025

We can choose relation, and as creators of technology we have the right and the duty to build for it.

Humanity Matters to Humanity

AI can do a lot.

But human connection is still essential.

When I love a work of art, what it makes me feel connects my experience to its creator, and to everyone else who loves it.

We know that the others feel because we feel it.

That is how we bridge the distance between us.

Human authorship matters.

Michelangelo, Night (1526–31), Medici Chapel, Florence · Photo: George M. Groutas, CC BY 2.0

Acknowledgements

This work is a team effort. Thank you to my co-authors and co-workers:

Batuhan Akkaya

Hamid Tabani

Kees van Berkel

Luc Waeijen

Arash Pourtaherian

Zeqi Zhu

Steven Roos

Peter Kievits

Savvas Sioutas



S P E C S

Thank you

Questions?

omoreira@snapchat.com

READ THE PAPERS



Neuronflow V2
IEEE Micro 2026



STAR
CVPRW 2023



MEET
CVPR 2025