20th January 2022

IFIP WG 10.4

Can we Rely on Self-Driving Cars?

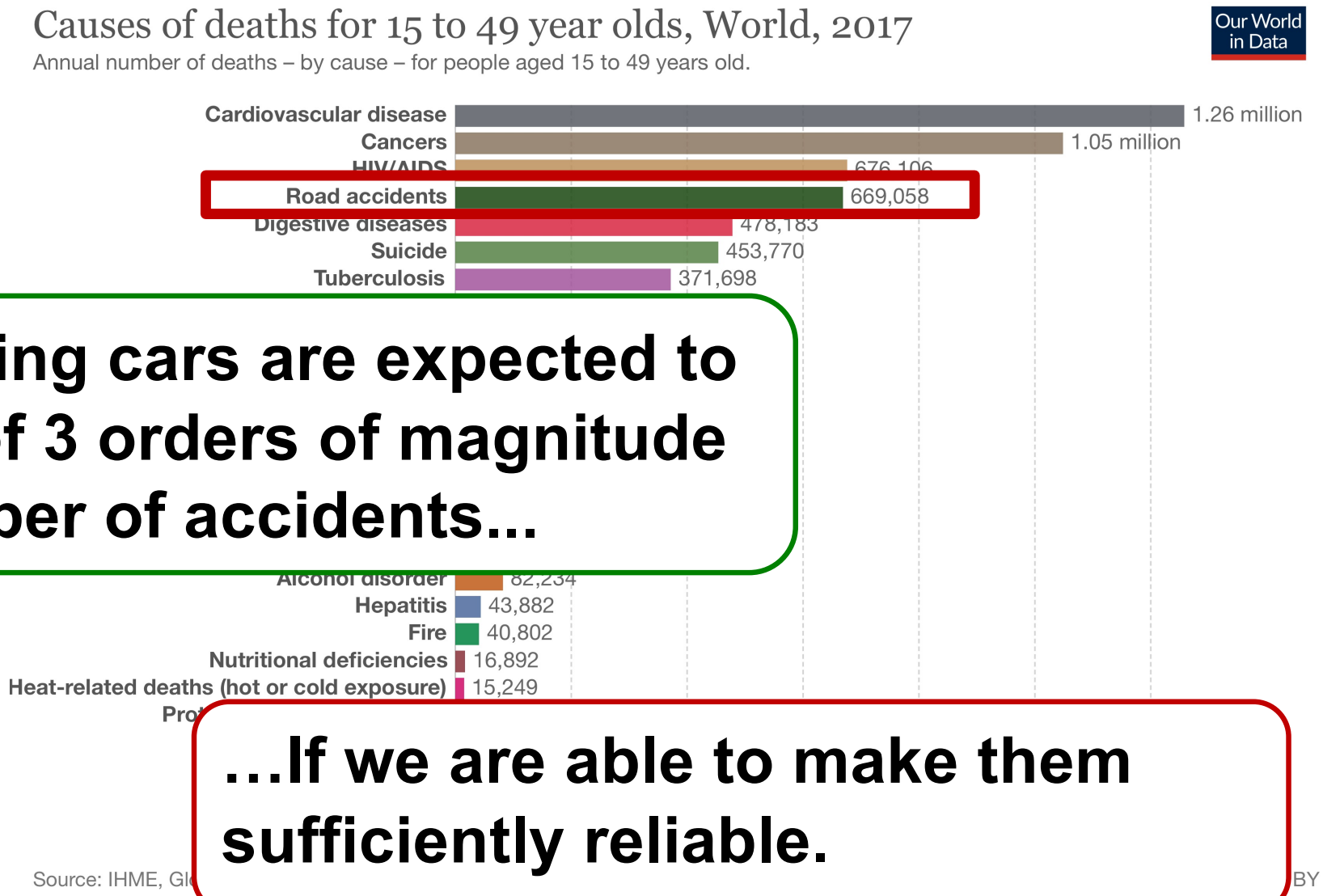
Evaluation and Mitigation of Neutron-Induced Errors in Convolutional Neural Networks for Autonomous Vehicles

Paolo Rech

UNIVERSITÀ DI TRENTO

UFRGS
UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Self-Driving Cars importance



Self-driving cars are expected to reduce of 3 orders of magnitude the number of accidents...

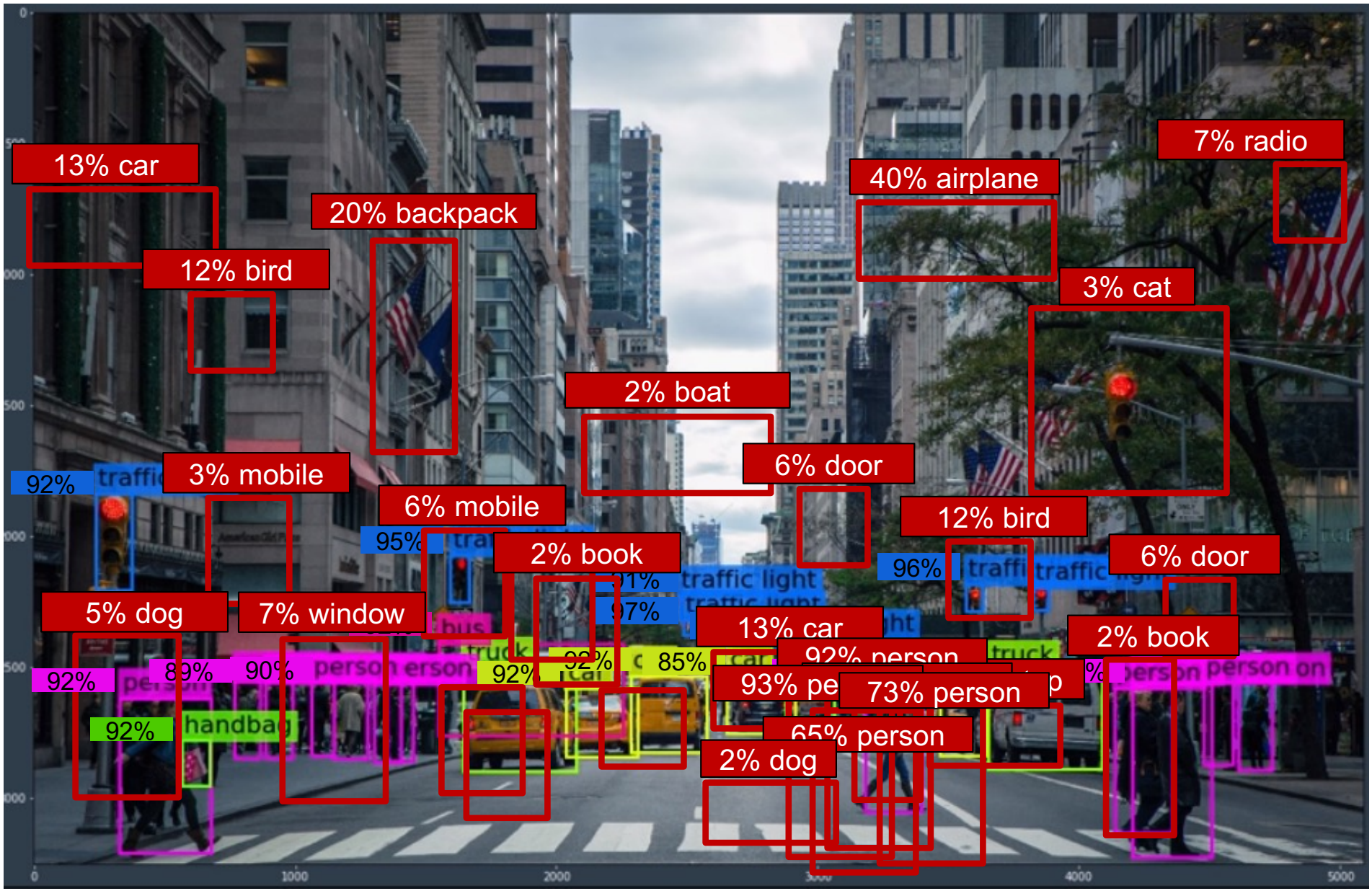
...If we are able to make them sufficiently reliable.

CNNs Reliability

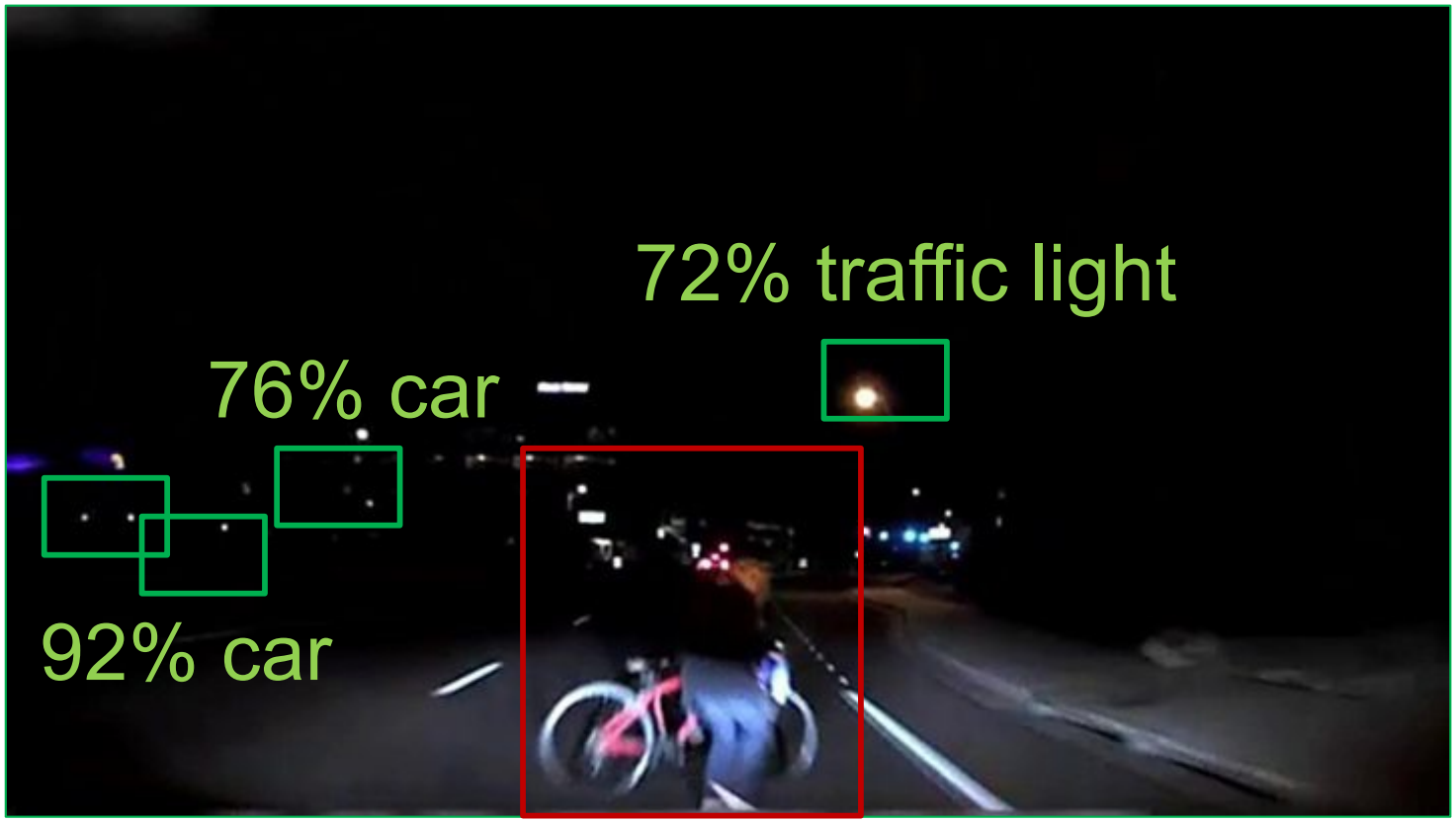
CNNs Reliability

CNNs Reliability

CNNs Reliability



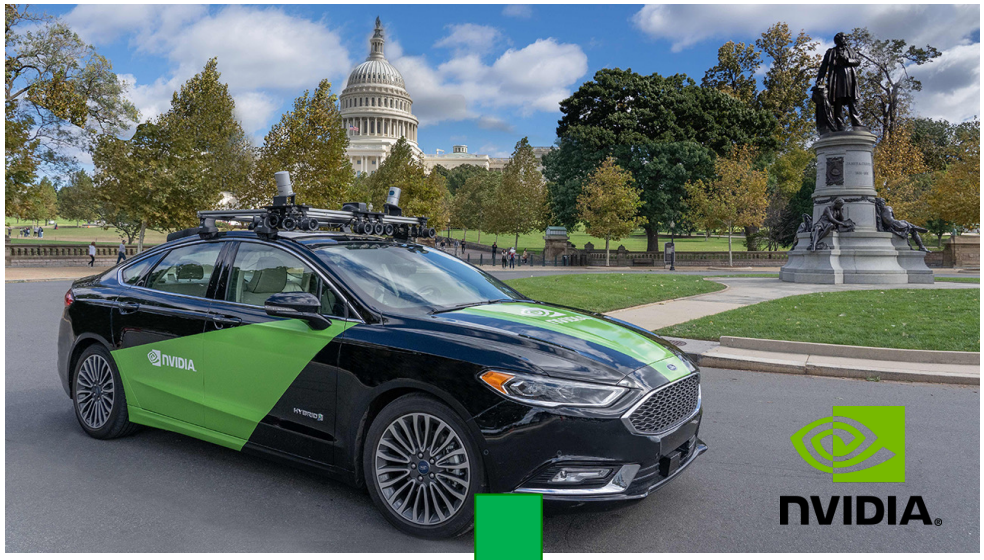
SW Problems

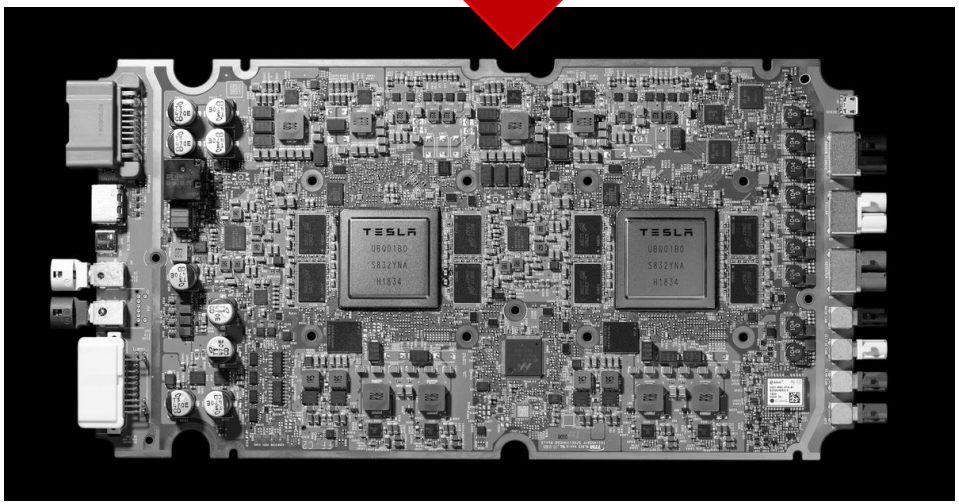


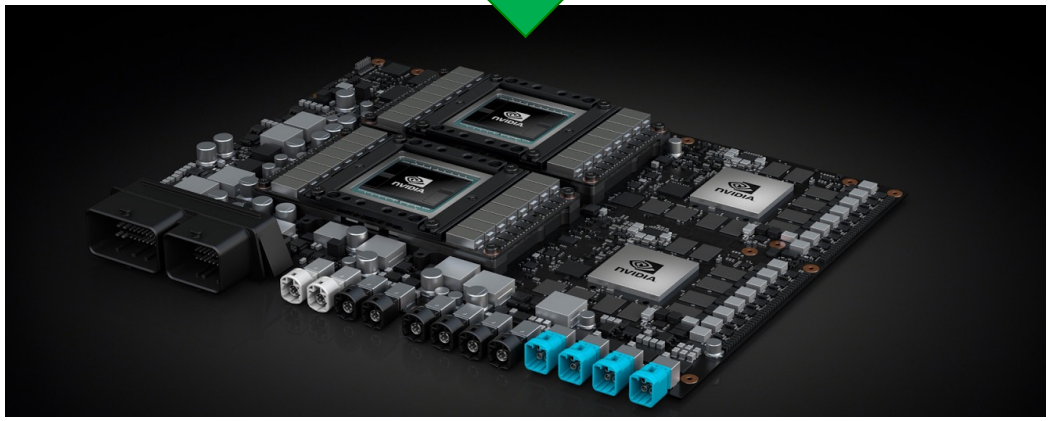
woman with a bike
probability < threashold

What about the HW?

Today's self-driven cars

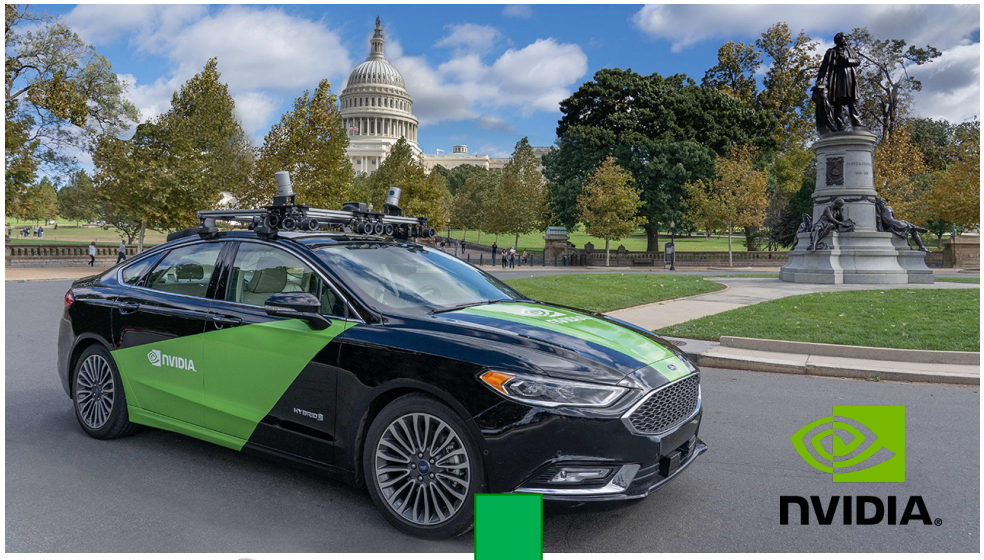


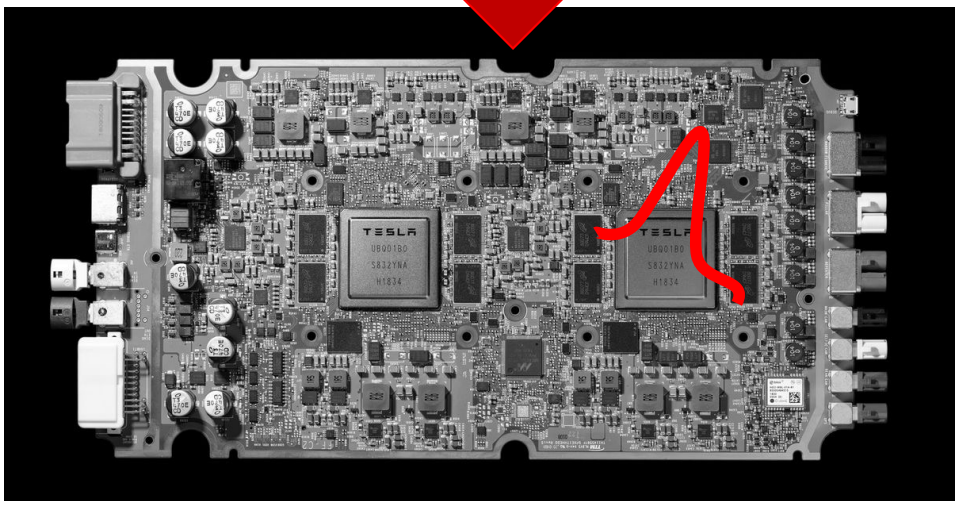


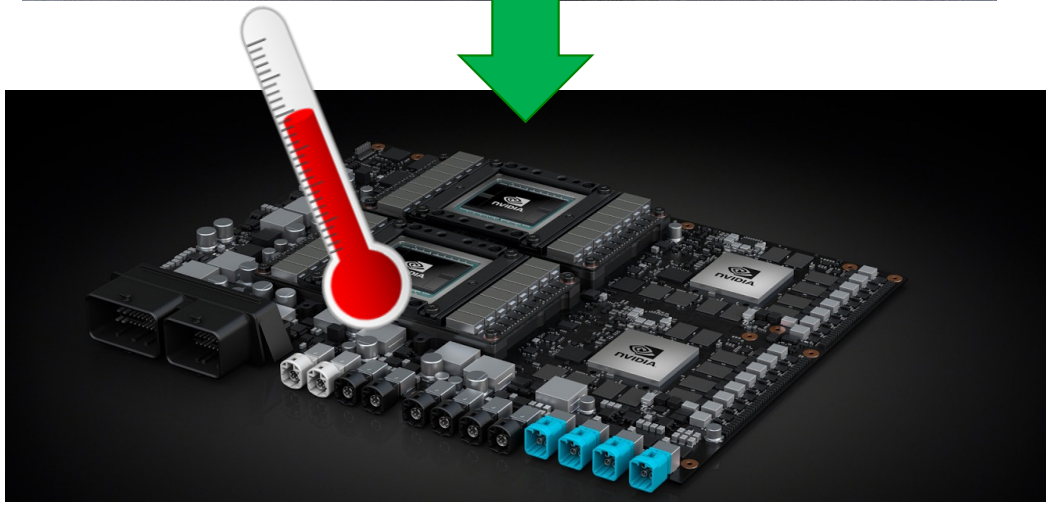


What about the HW?

Today's self-driven cars

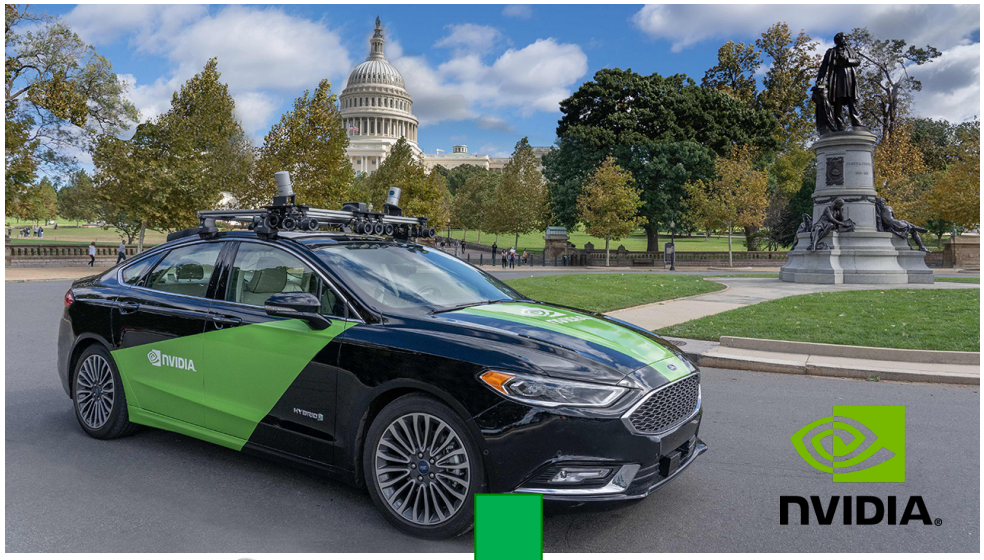


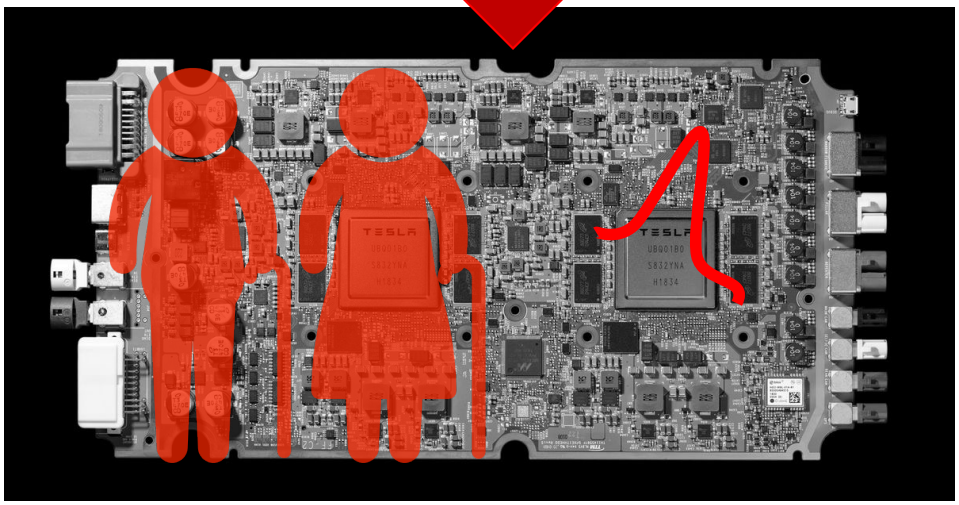


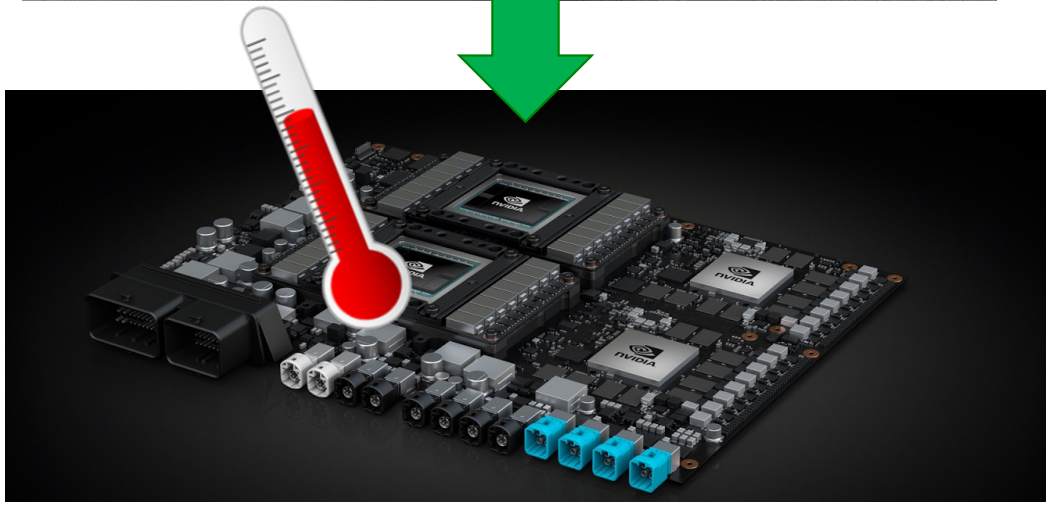


What about the HW?

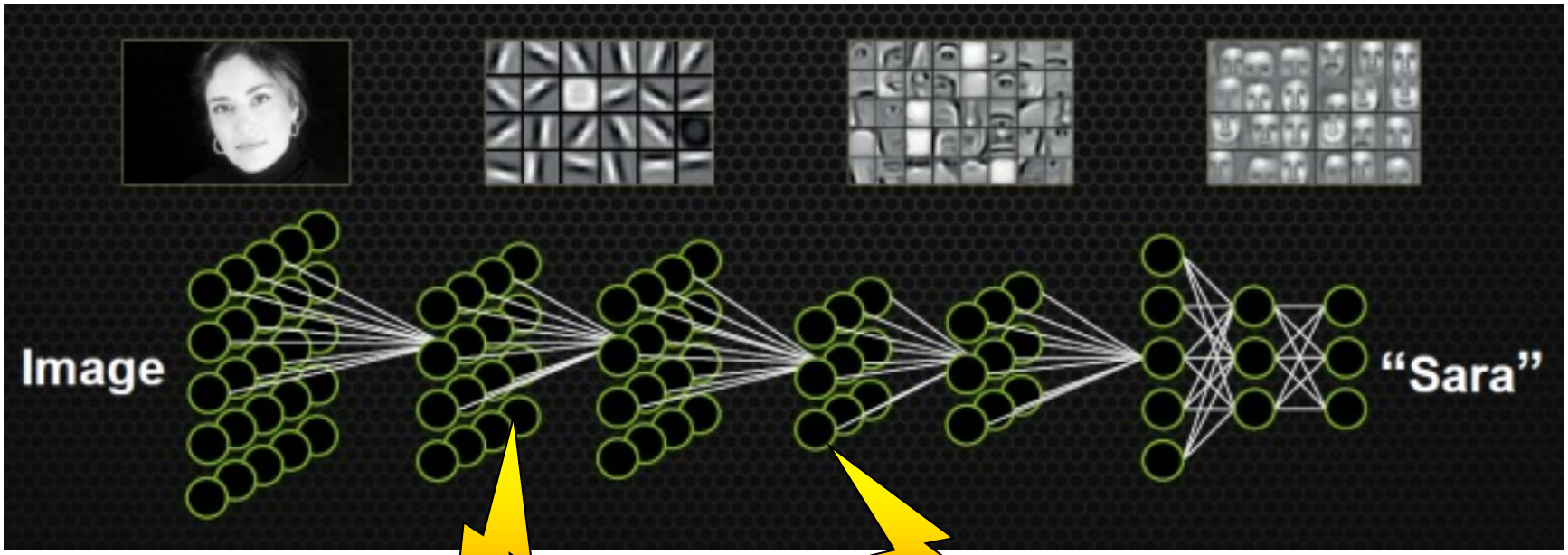
Today's self-driven cars

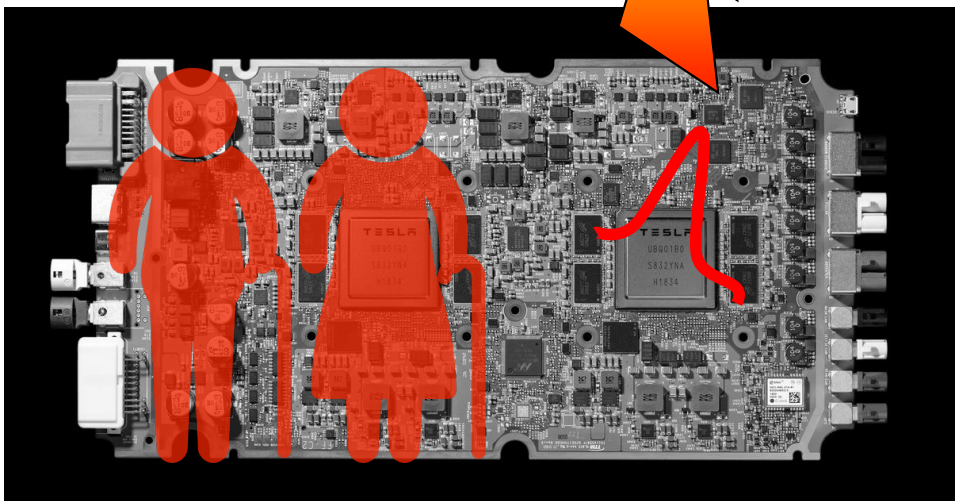


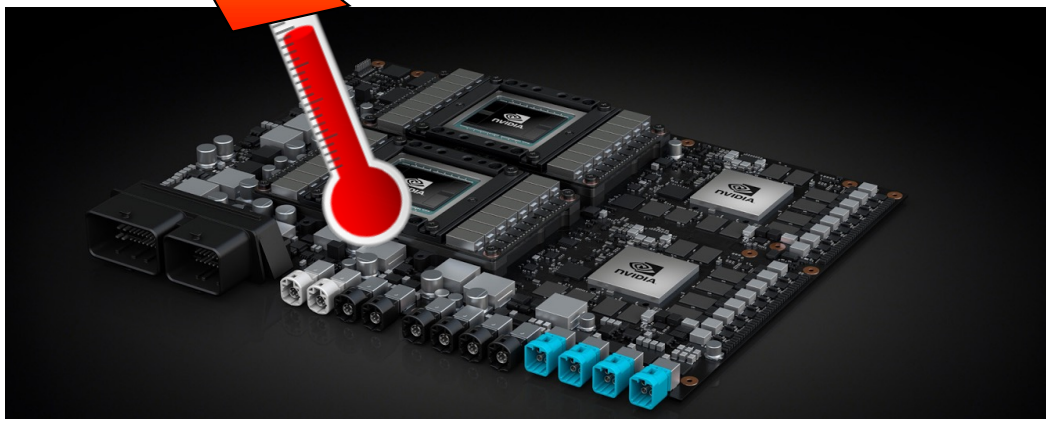




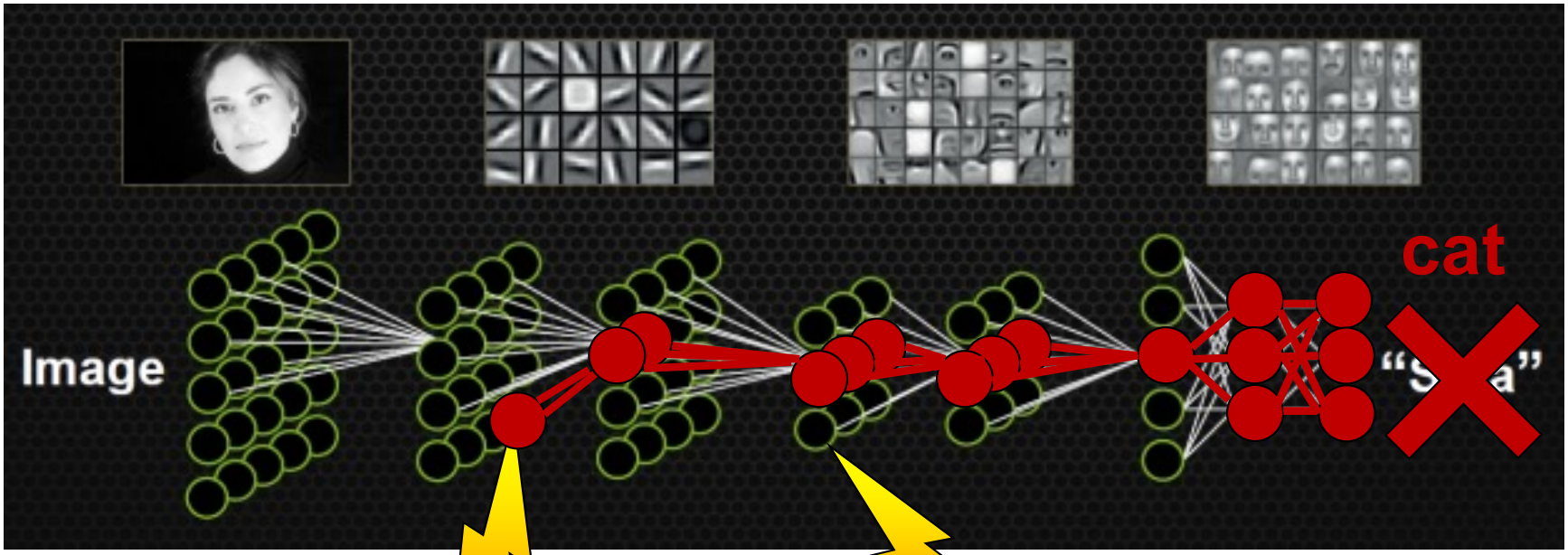
What about the HW?

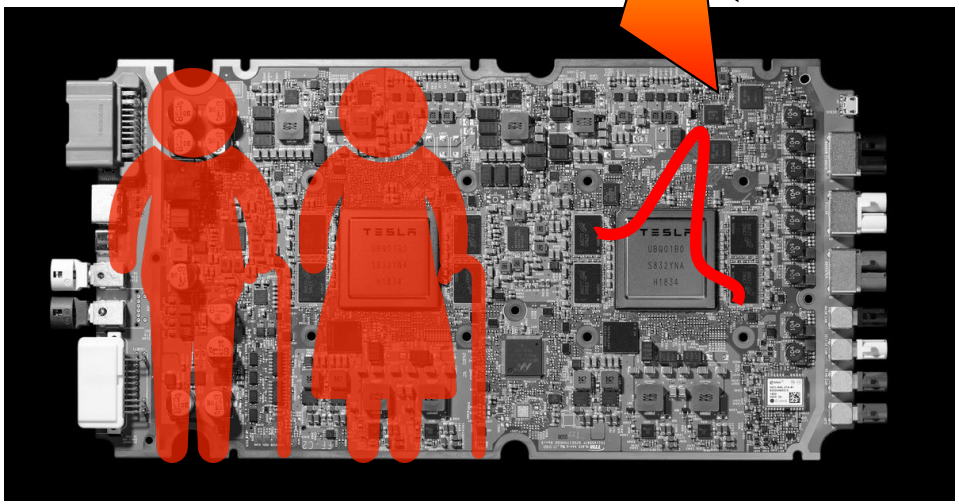


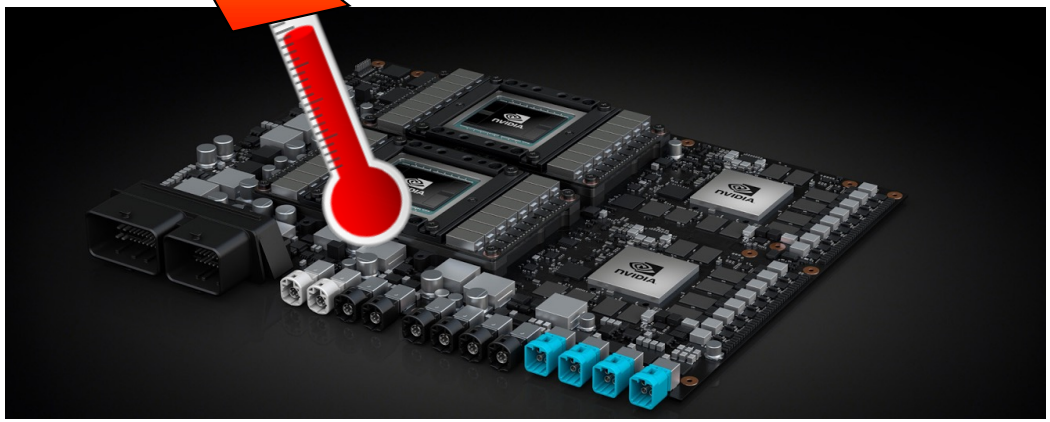




What about the HW?







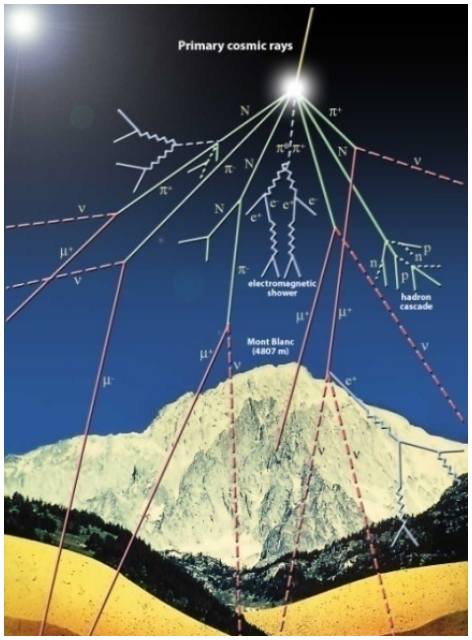
Outline

- Neutrons-induced effects in computing devices
- Evaluating neutron-induced errors probabilities
- Cross layer faults propagation in CNNs
- Some (interesting) efficient solutions
- Conclusions and Future Work

Outline

- **Neutrons-induced effects in computing devices**
- Evaluating neutron-induced errors probabilities
- Cross layer faults propagation in CNNs
- Some (interesting) efficient solutions
- Conclusions and Future Work

Terrestrial Radiation Environment



Galactic cosmic rays interact with atmosphere

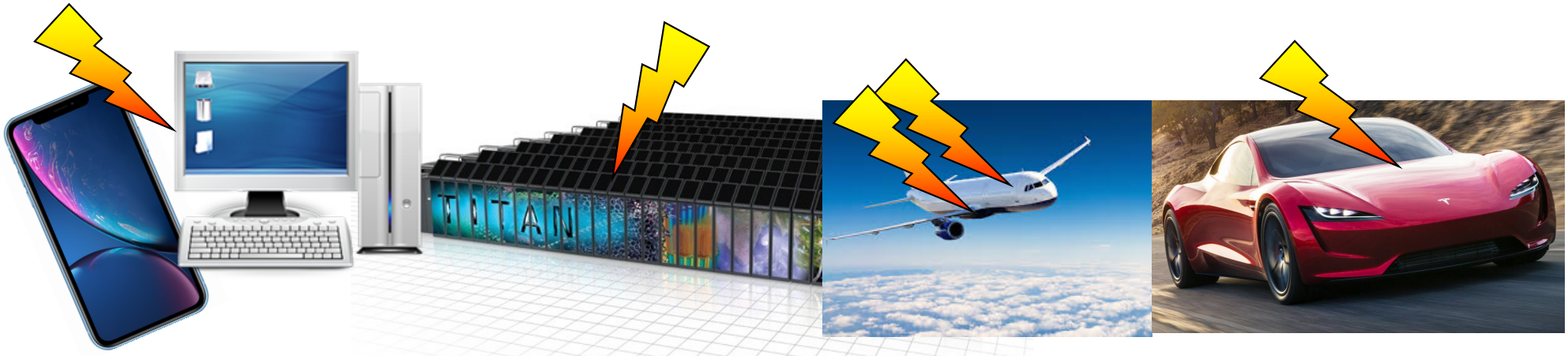
shower of energetic particles:

Muons, Pions, Protons, Gamma rays, Neutrons

13 n/(cm^2·h) @sea level*

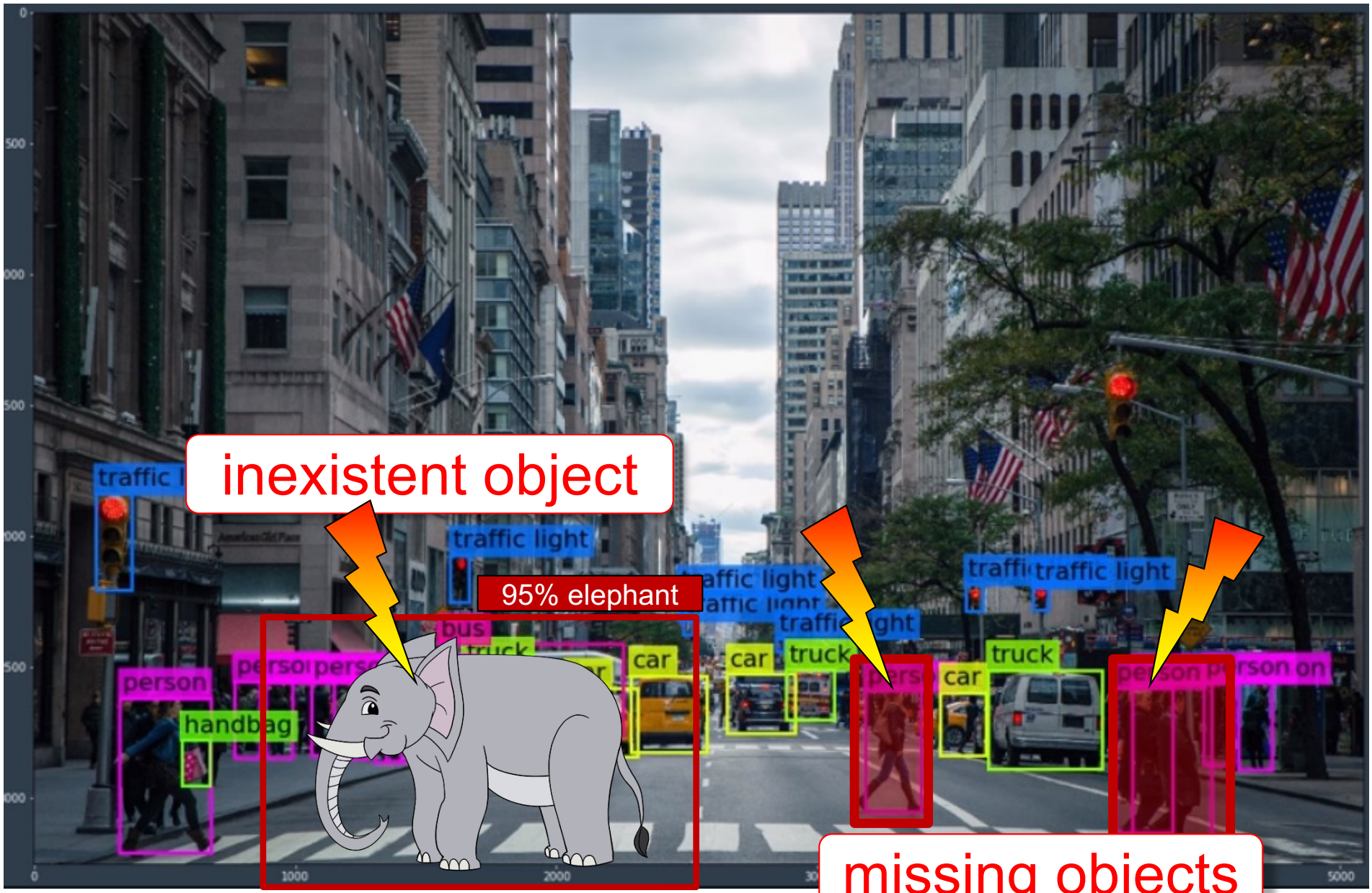
*JEDEC JESD89A Standard

Neutrons induce faults in modern computing systems



CNNs Reliability

CNNs Reliability



Radiation Effects - Soft Errors

Soft Errors: the device is not permanently damaged, but the particle may generate:

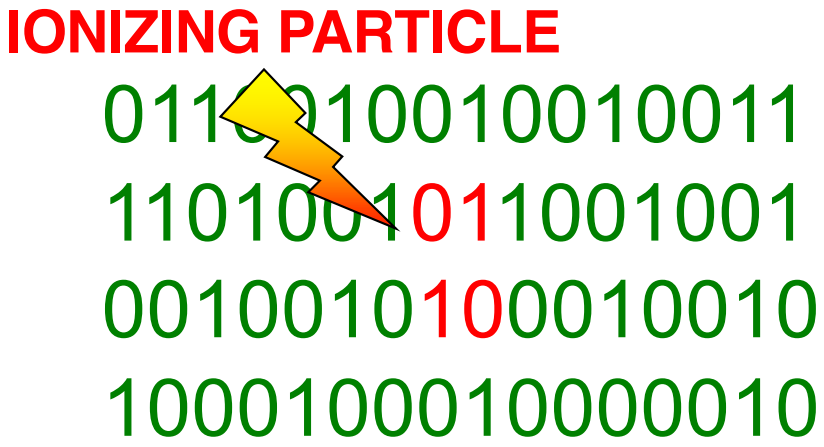
- One or more bit-flips
  - Single Event Upset (SEU)
  - Multiple Bit Upset (MBU)

0110010010010011
1101001101001001
0010010010010010
1000100010000010

Radiation Effects - Soft Errors

Soft Errors: the device is not permanently damaged, but the particle may generate:

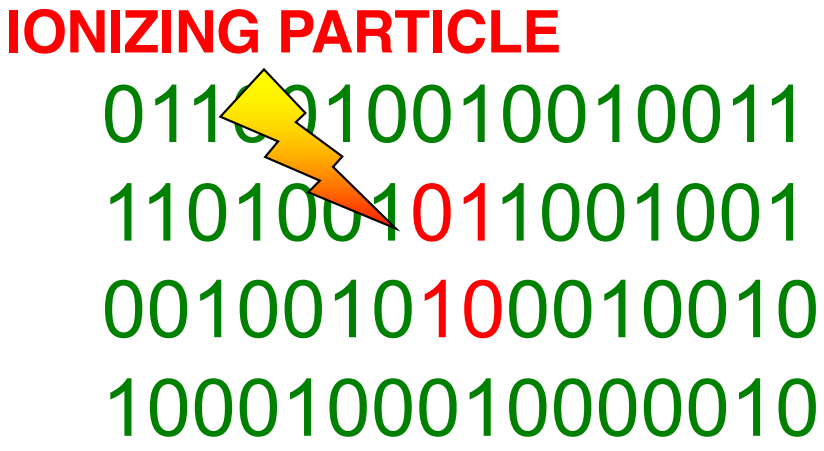
- One or more bit-flips
  - Single Event Upset (SEU)
  - Multiple Bit Upset (MBU)



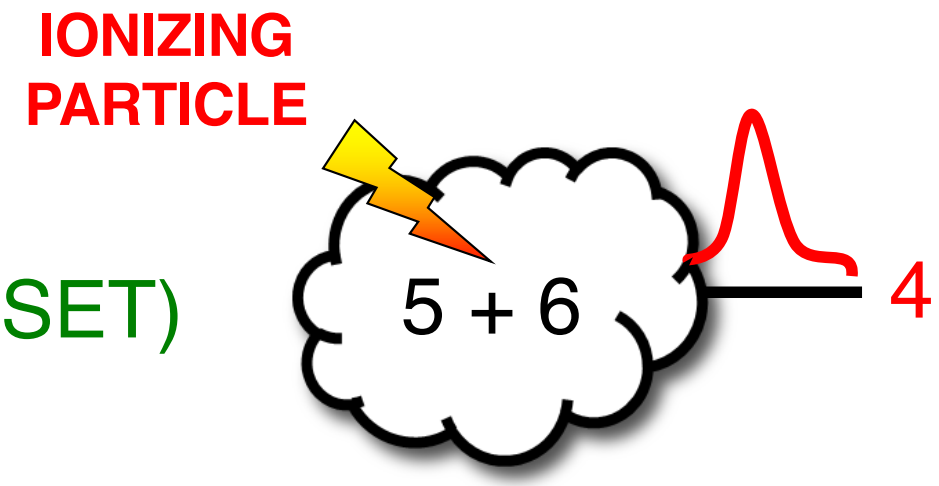
Radiation Effects - Soft Errors

Soft Errors: the device is not permanently damaged, but the particle may generate:

- One or more bit-flips
  - Single Event Upset (SEU)
  - Multiple Bit Upset (MBU)



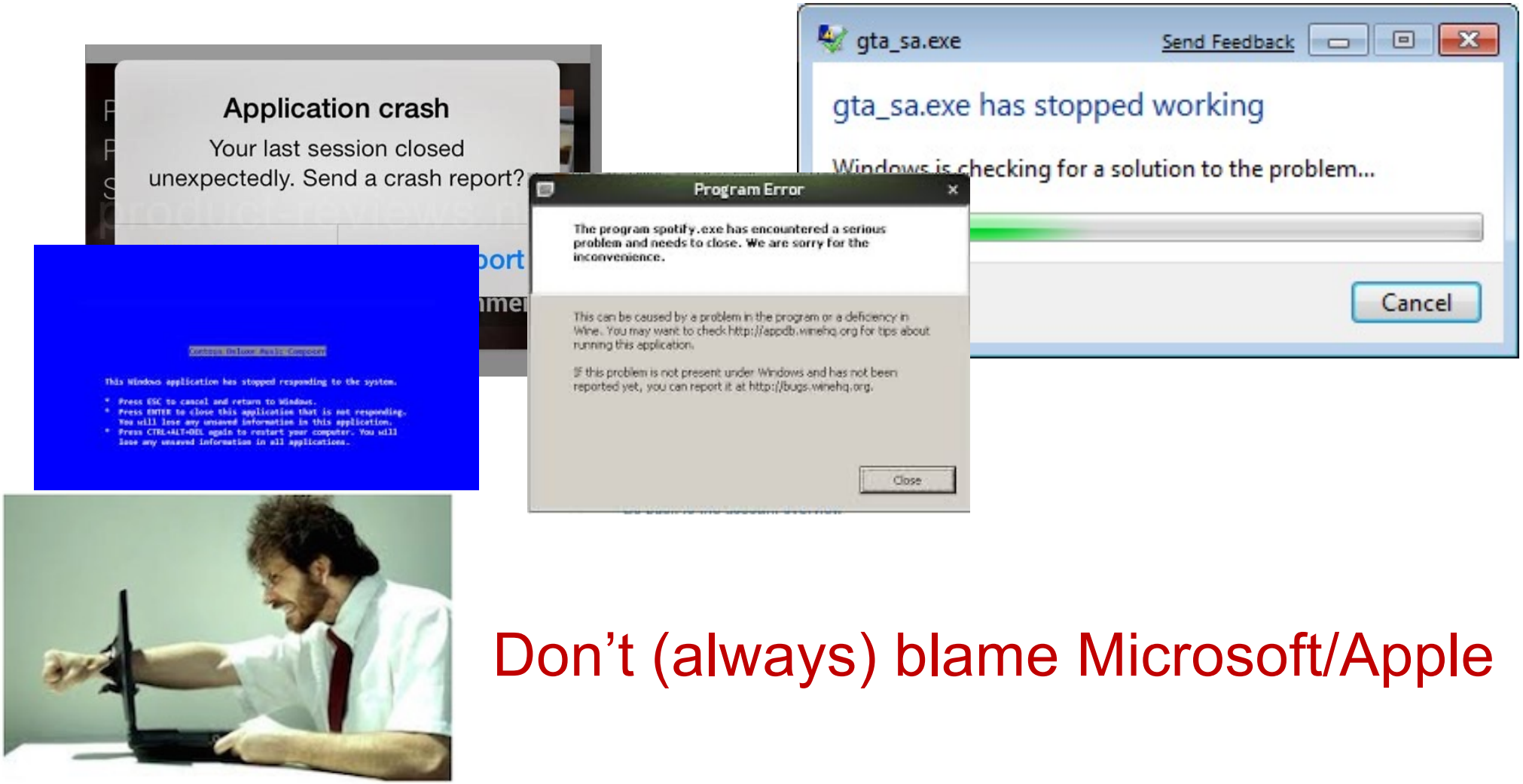
- Transient voltage pulse
  - Single Event Transient (SET)



Silent Data Corruption vs Crash

Neutron-induced faults can also induce

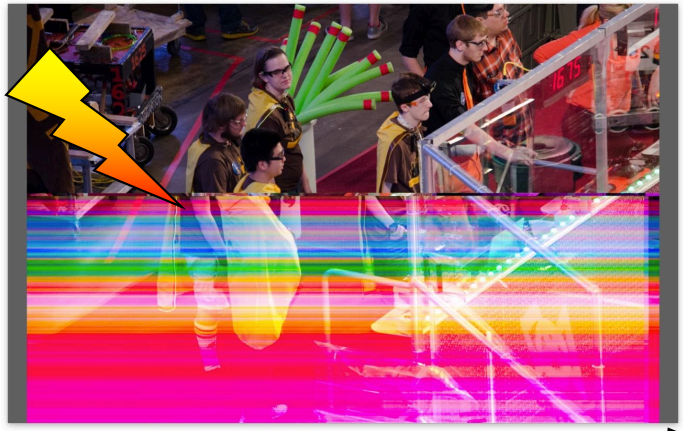
Application Crash or Device Reboot

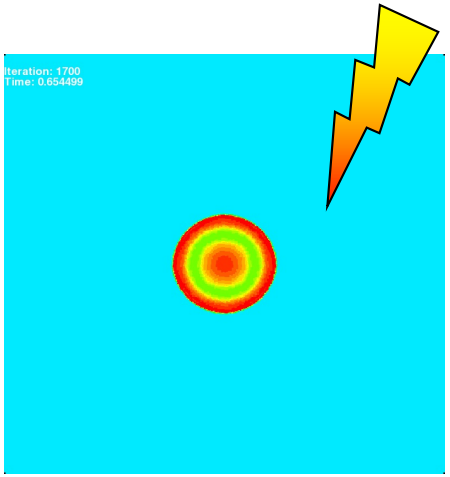


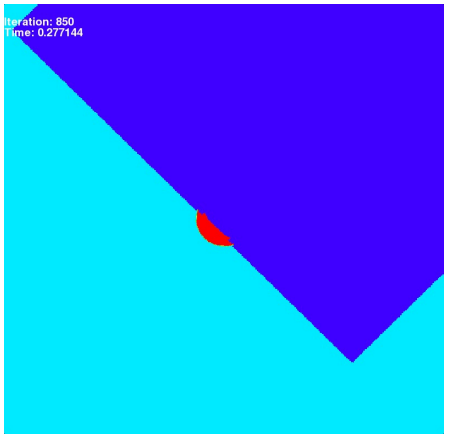
Don’t (always) blame Microsoft/Apple

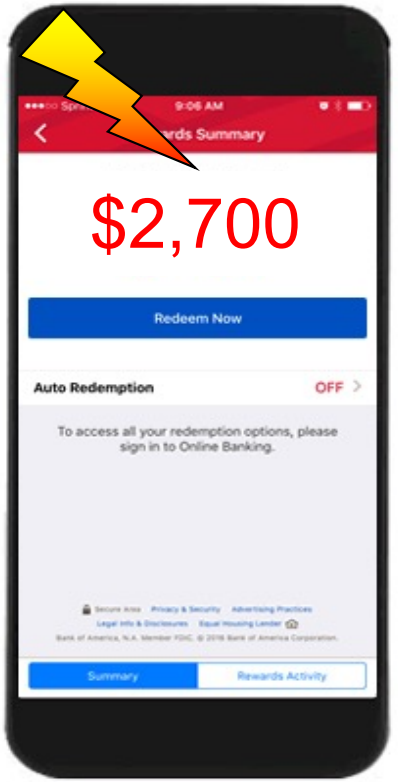
Silent Data Corruption vs Crash

Silent Data Corruption: the application provides wrong answers. **Silent** = no flag/no indication of error.

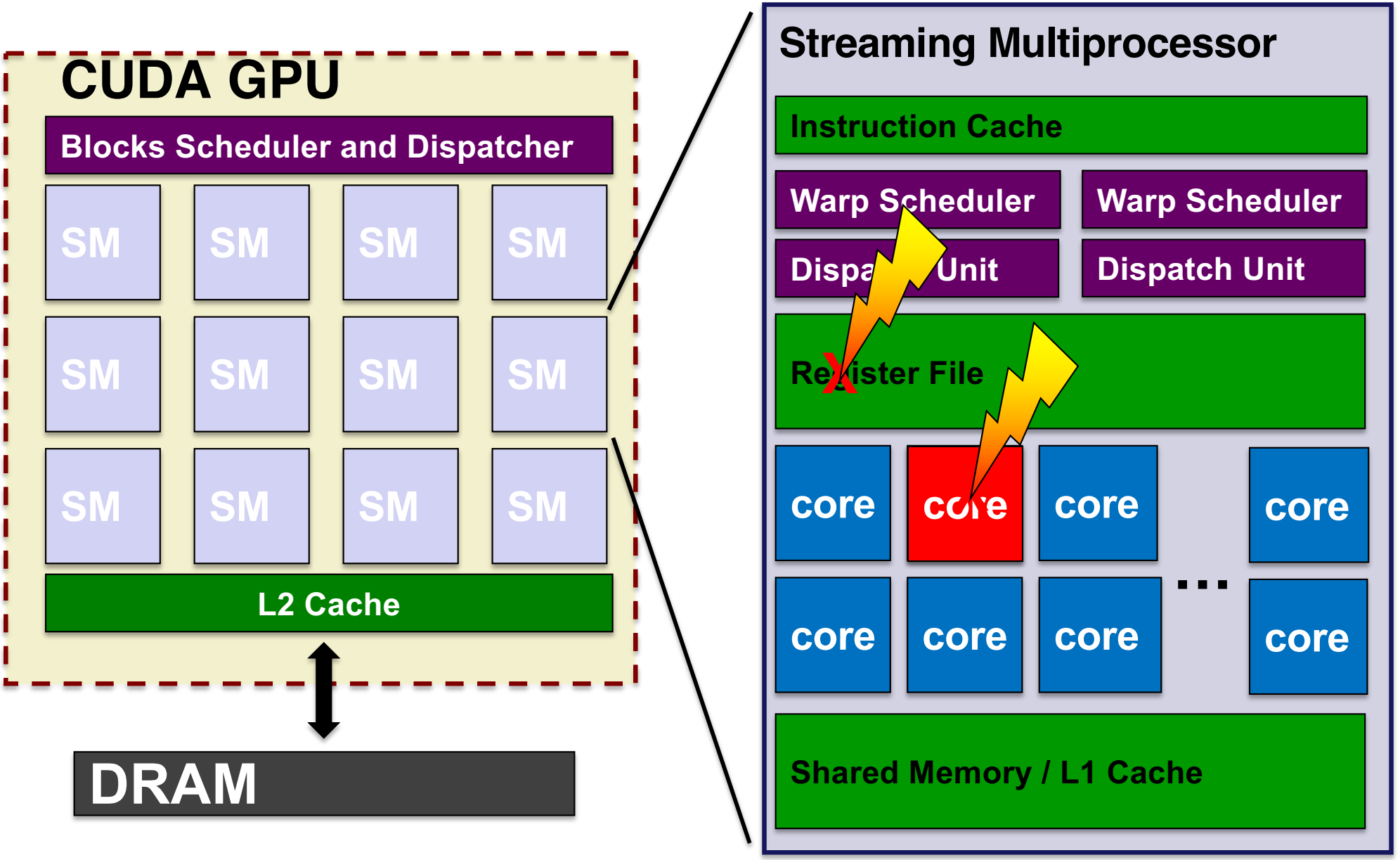




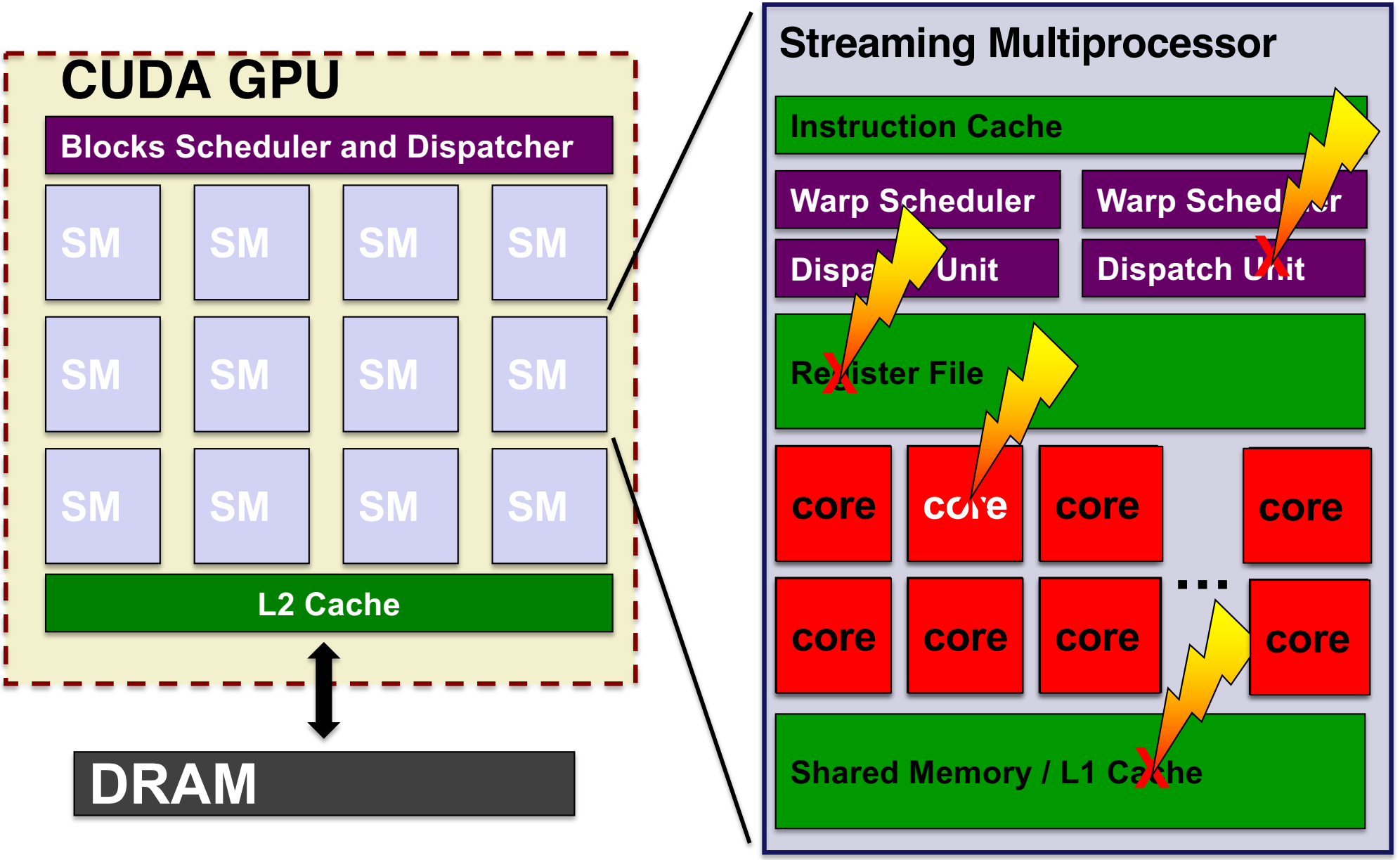




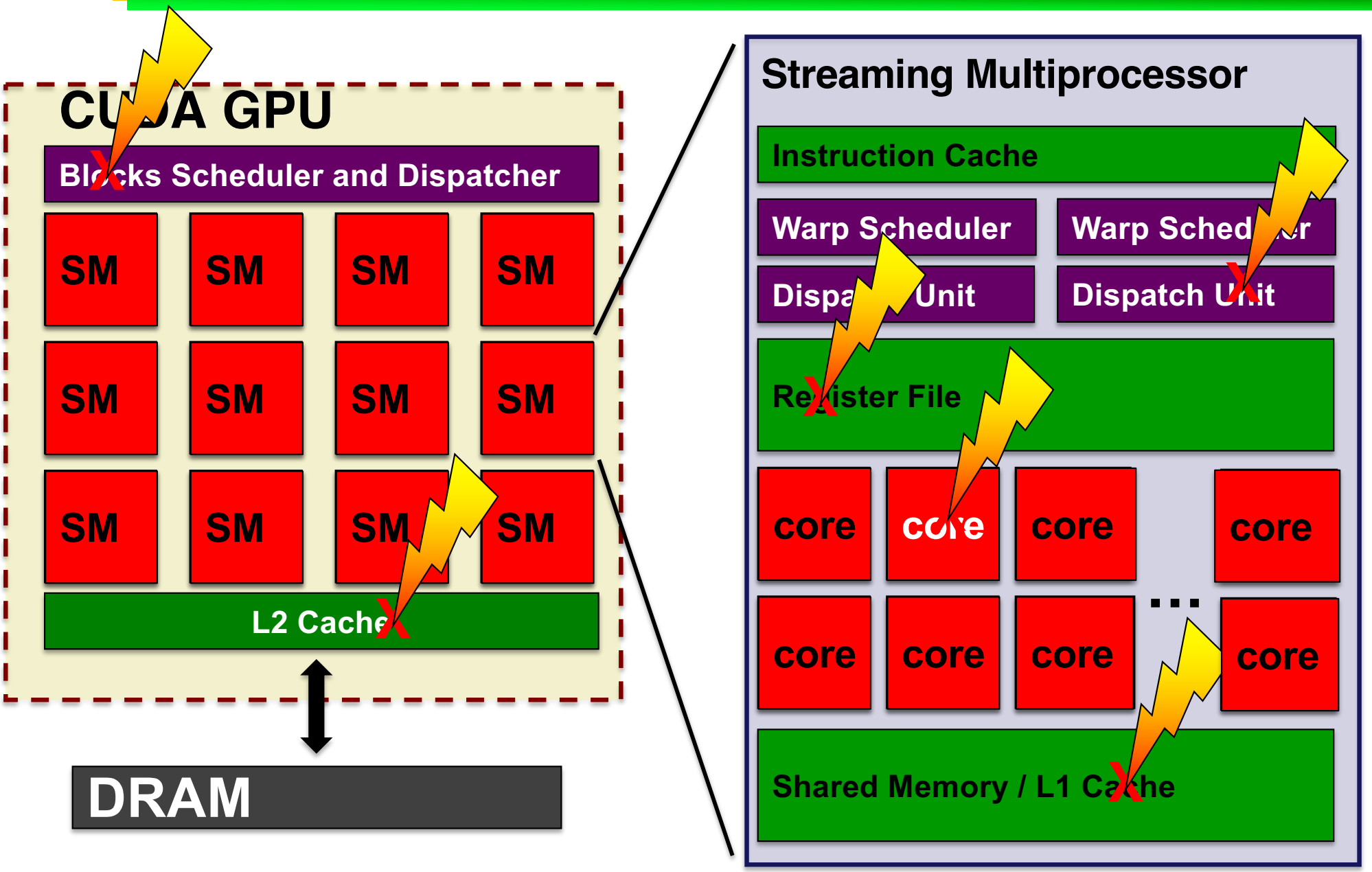
Radiation Effects on Parallel Accelerators



Radiation Effects on Parallel Accelerators

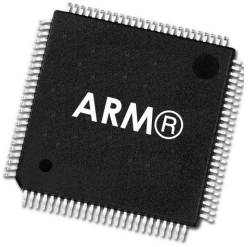


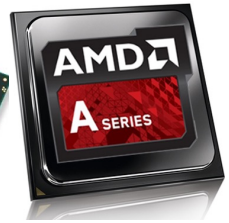
Radiation Effects on Parallel Accelerators

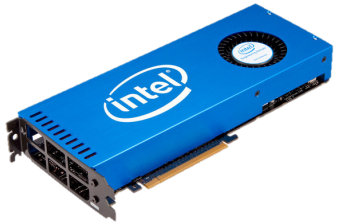


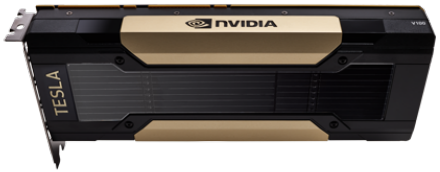
One device, different reliability requirements

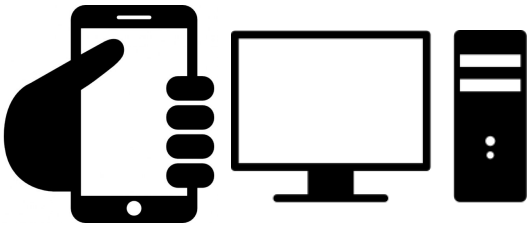
ARM CPU co-processor GPU FPGA/SoC







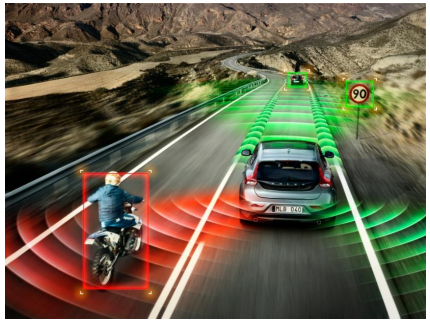




Consumer

Data Center

HPC



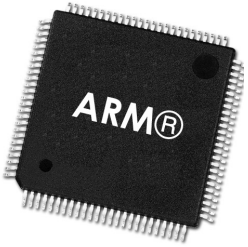
automotive

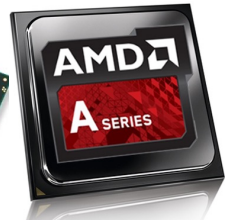
Good **High** **Very High**

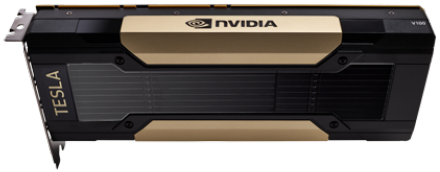
Reliability Requirements (*Arijit Biswas, SELSE 2018)

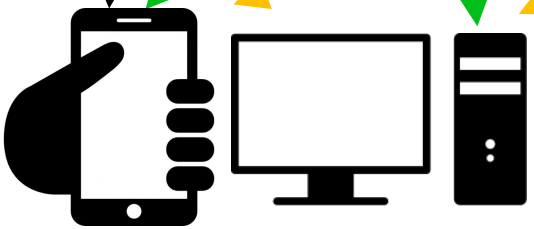
One device, different reliability requirements

ARM CPU co-processor GPU FPGA/SoC

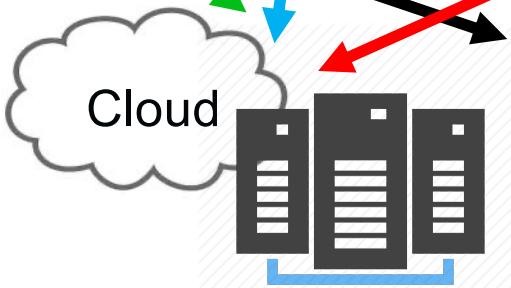








Consumer



Data Center

HPC

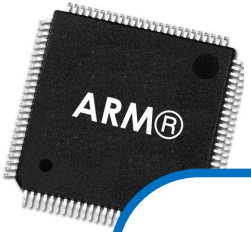
automotive

Good **High** **Very High**

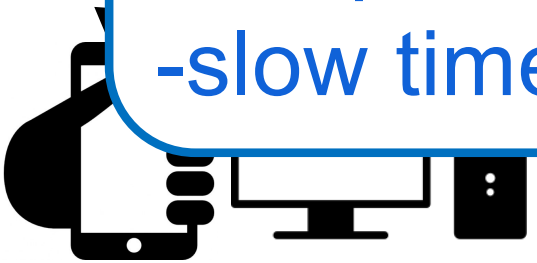
Reliability Requirements (*Arijit Biswas, SELSE 2018)

One device, different reliability requirements

ARM CPU co-processor GPU FPGA/SoC



Dedicated and highly reliable HW has:
-high cost
-low performances
-slow time-to-market


Consumer


Data Center


HPC


automotive

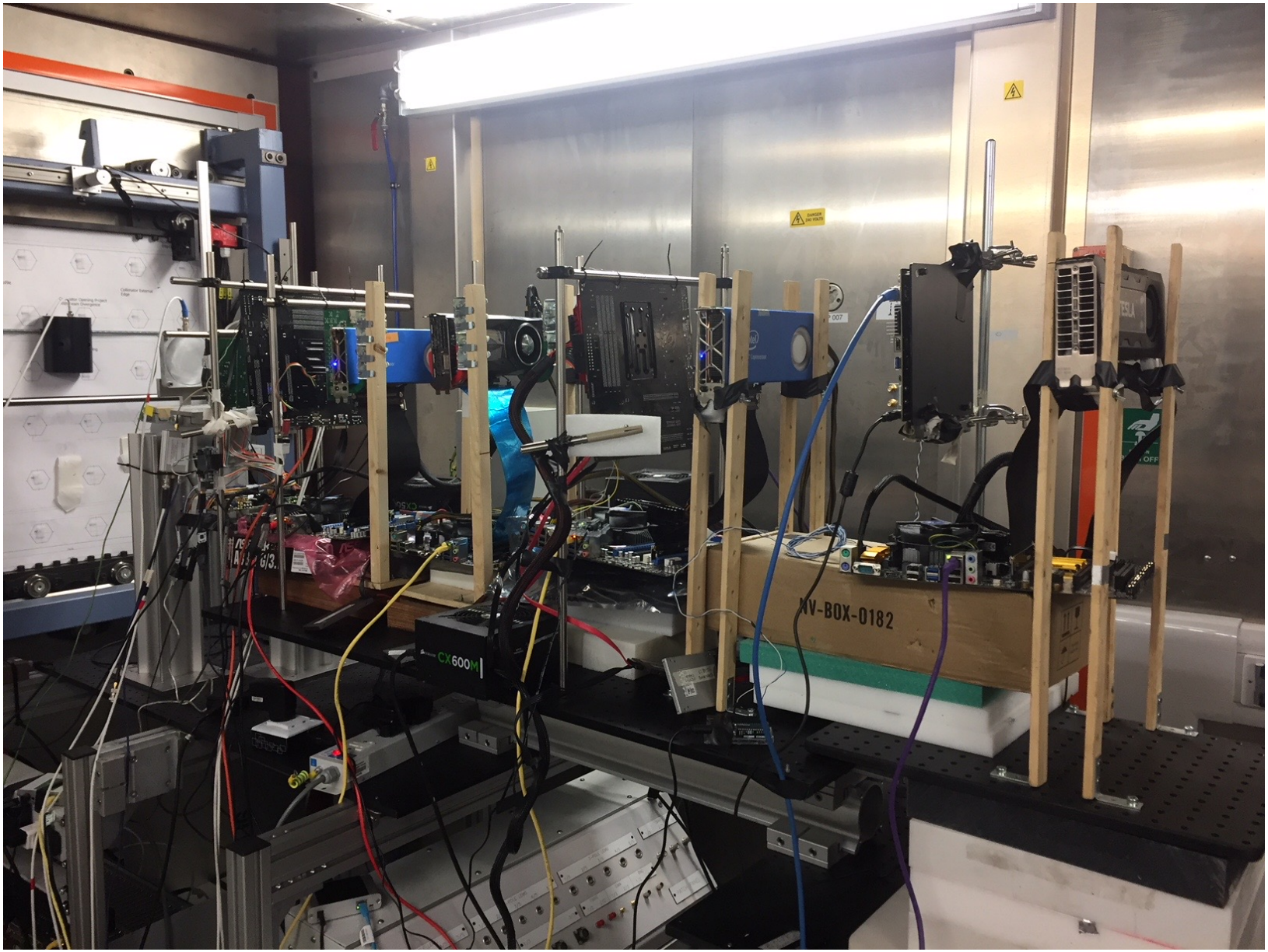
Good **High** **Very High**

Reliability Requirements (*Arijit Biswas, SELSE 2018)

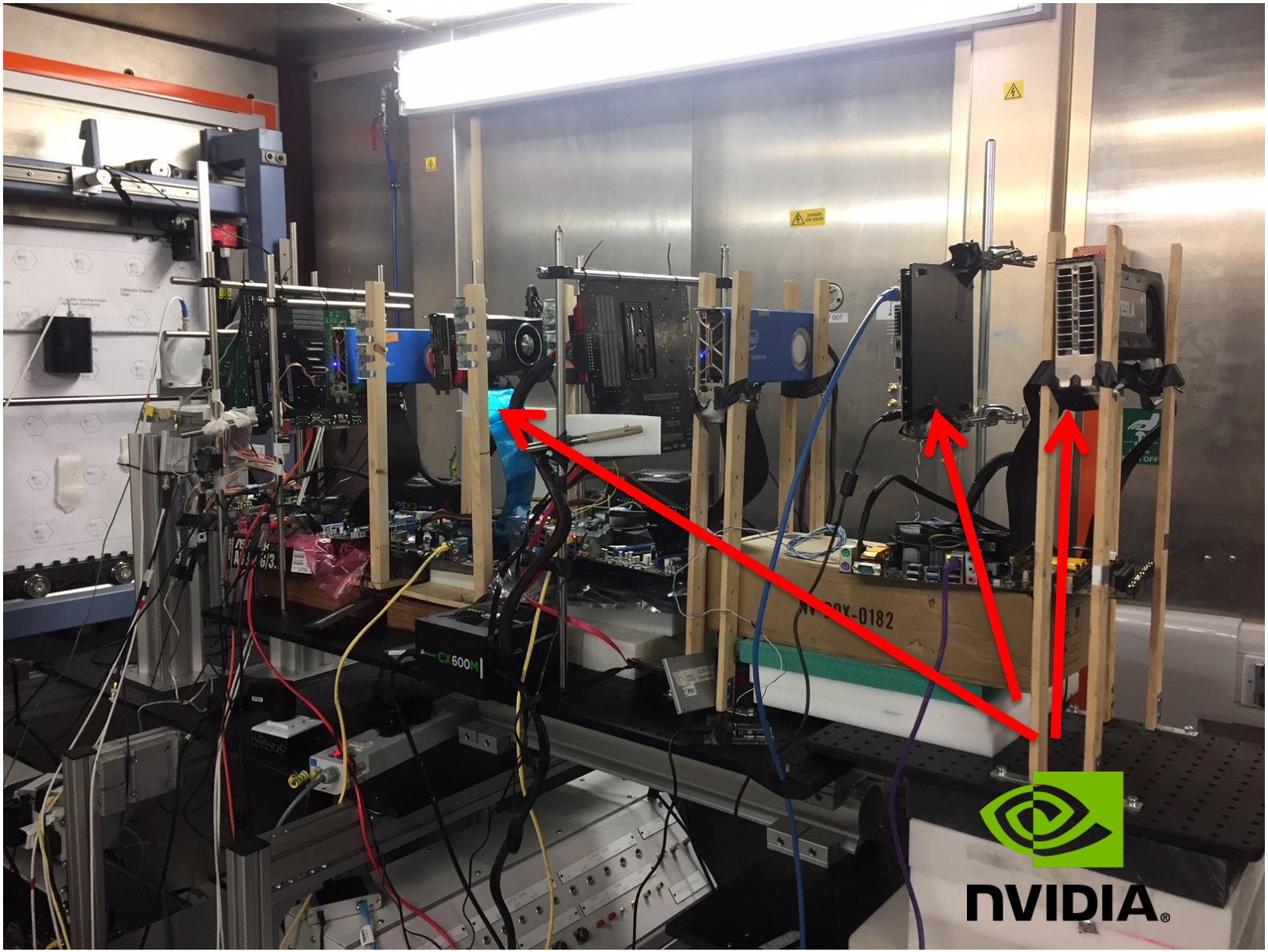
Outline

- Neutrons-induced effects in computing devices
- **Evaluating neutron-induced errors probabilities**
- Cross layer faults propagation in CNNs
- Some (interesting) efficient solutions
- Conclusions and Future Work

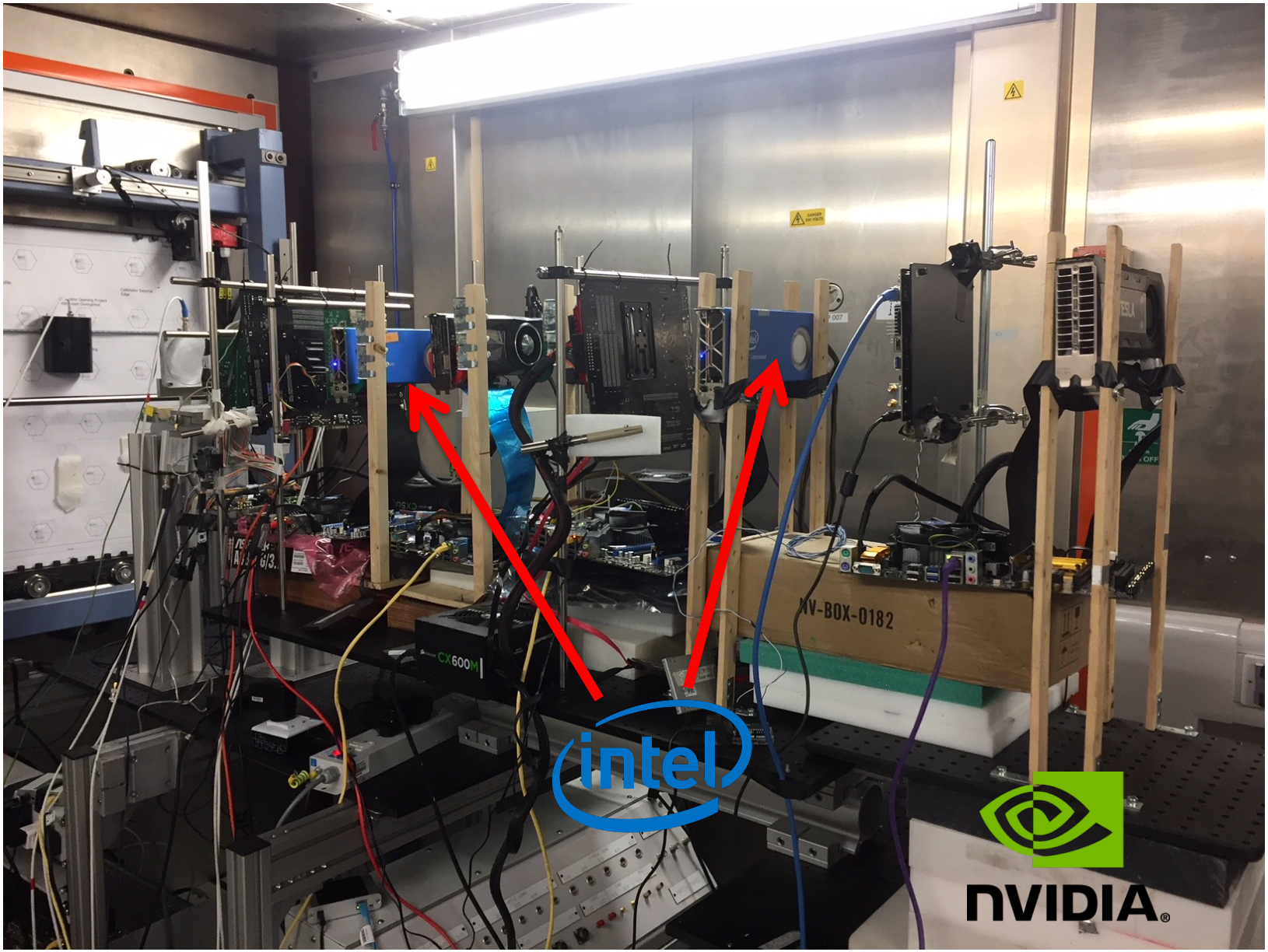
Experiment @ChipIR



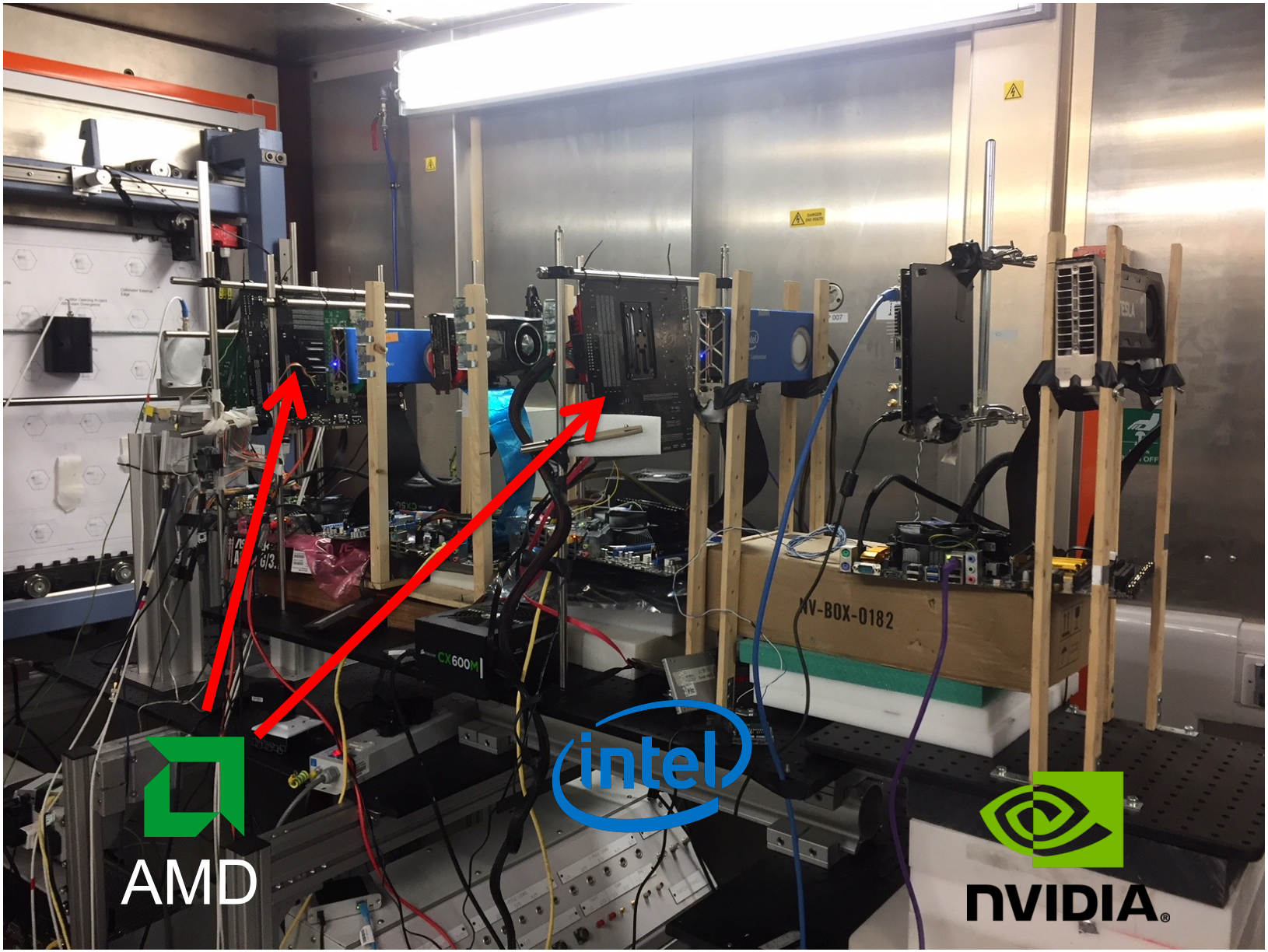
Experiment @ChipIR



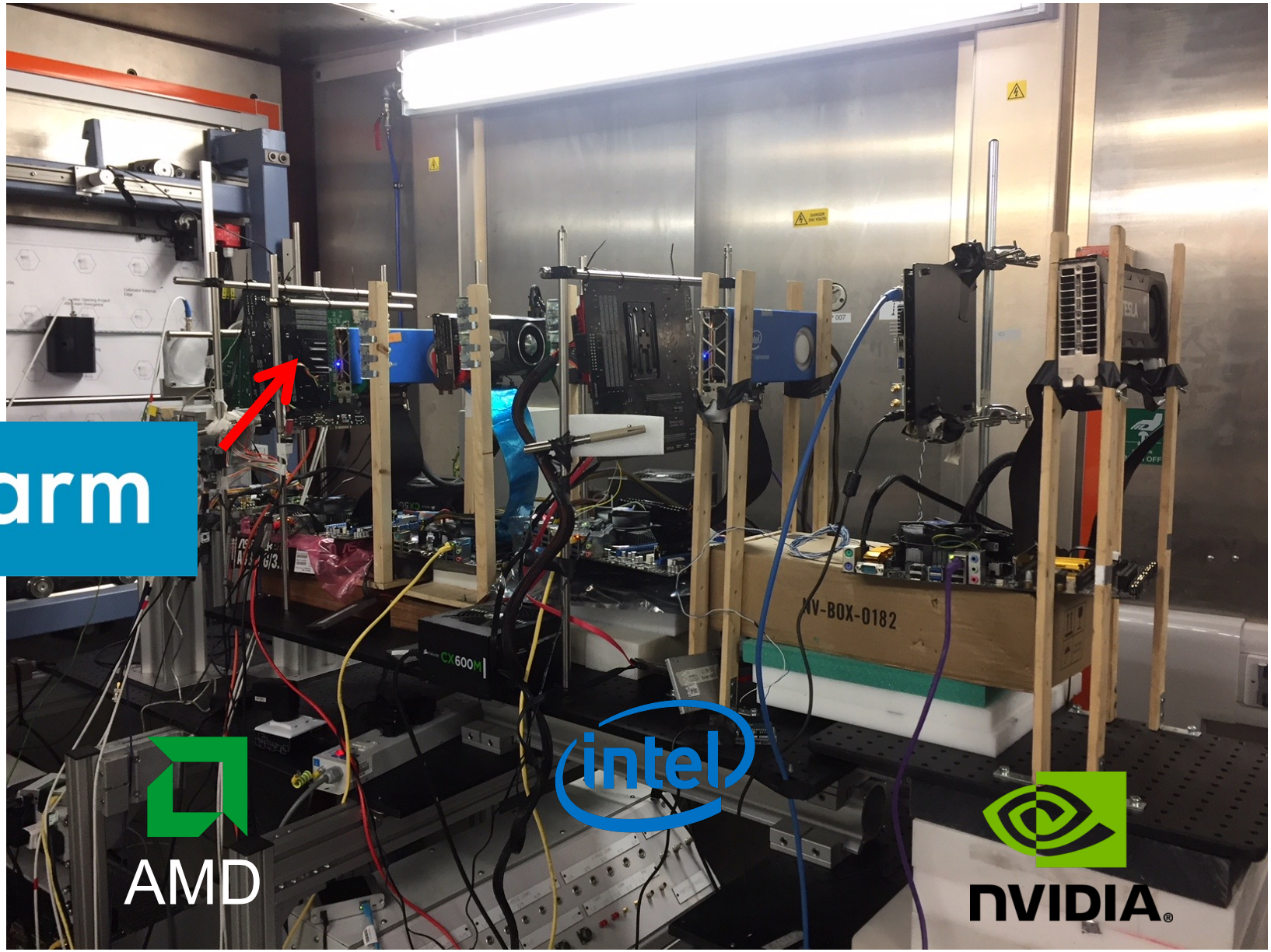
Experiment @ChipIR



Experiment @ChipIR

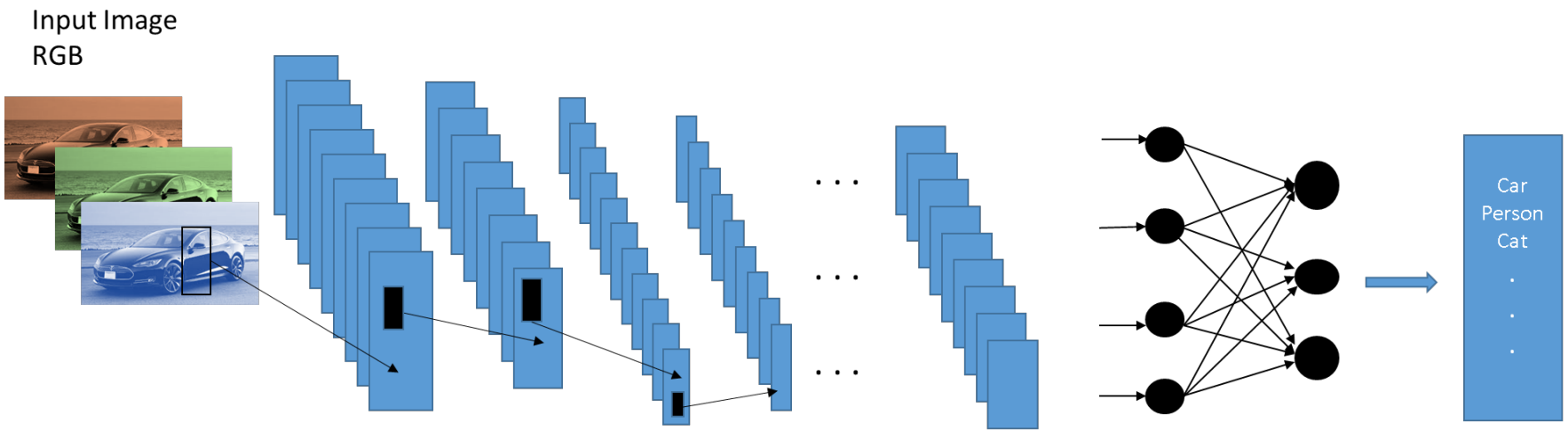


Experiment @ChipIR



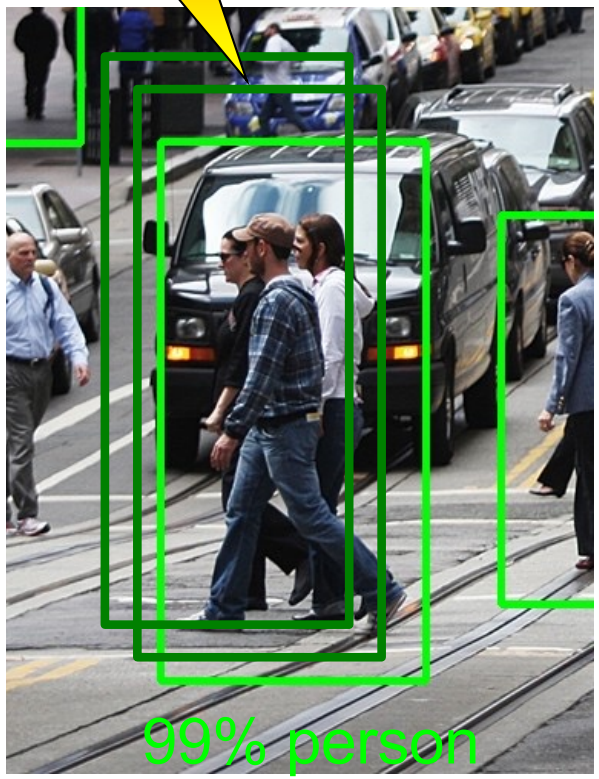
Self Driving Car

The new trend for automotive market is Self Driving Car!



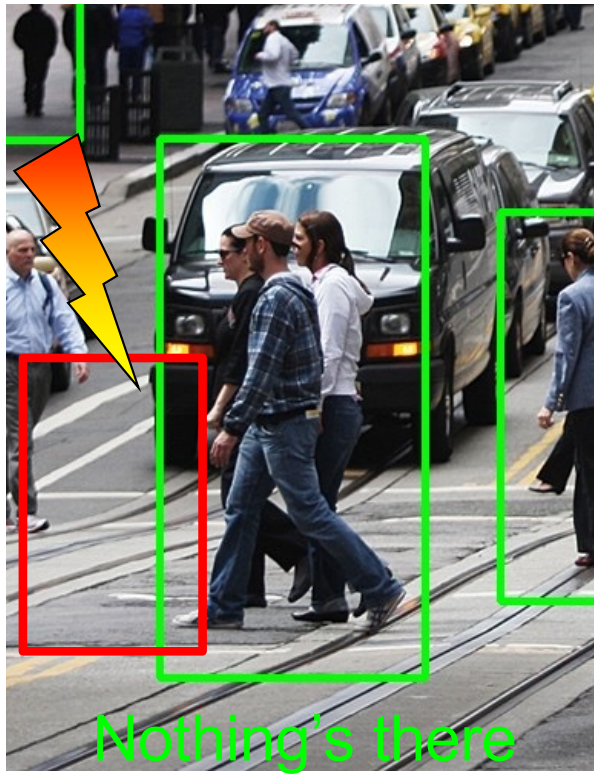
Examples of observed errors

Expected



Tolerable

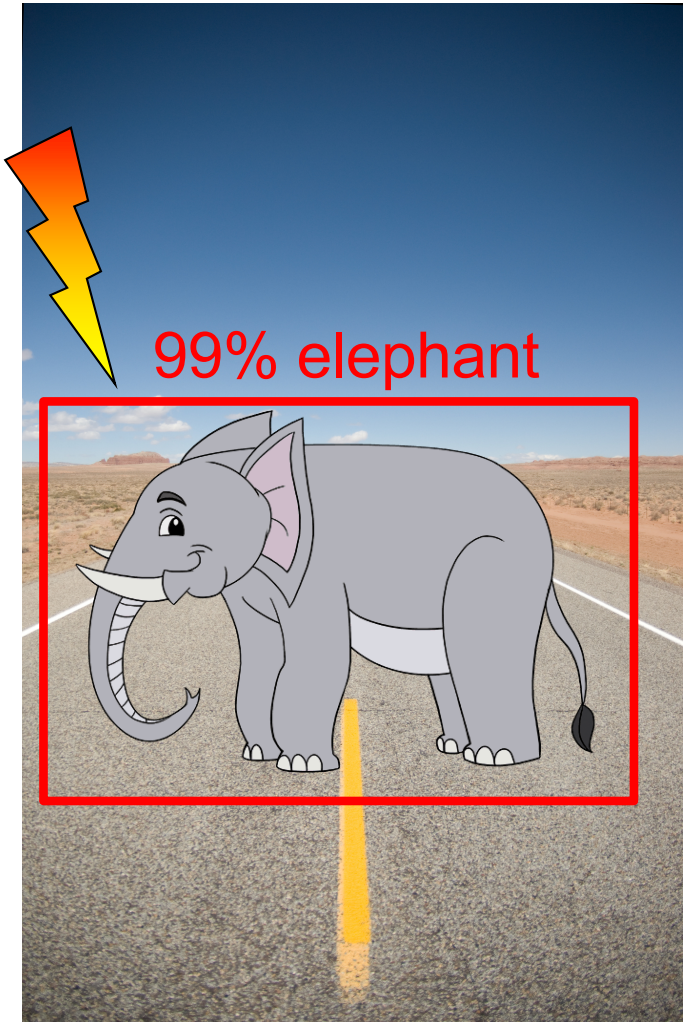
Slight modification of detection



Critical

Missing an object

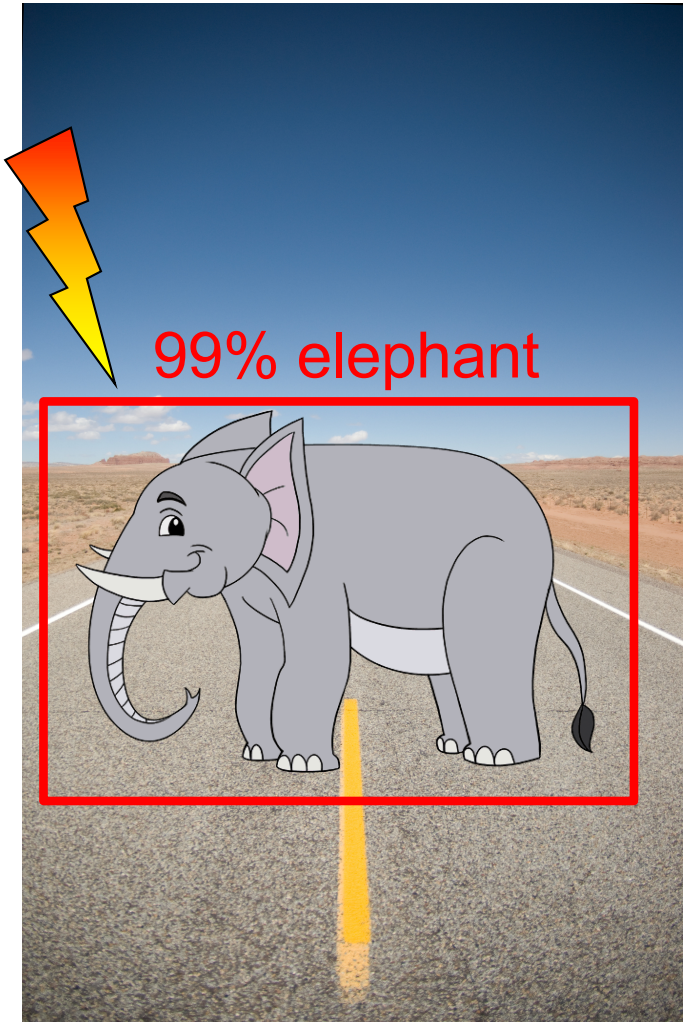
Examples of observed errors



False positive

Unnecessary stops

Examples of observed errors



False positive

Unnecessary stops

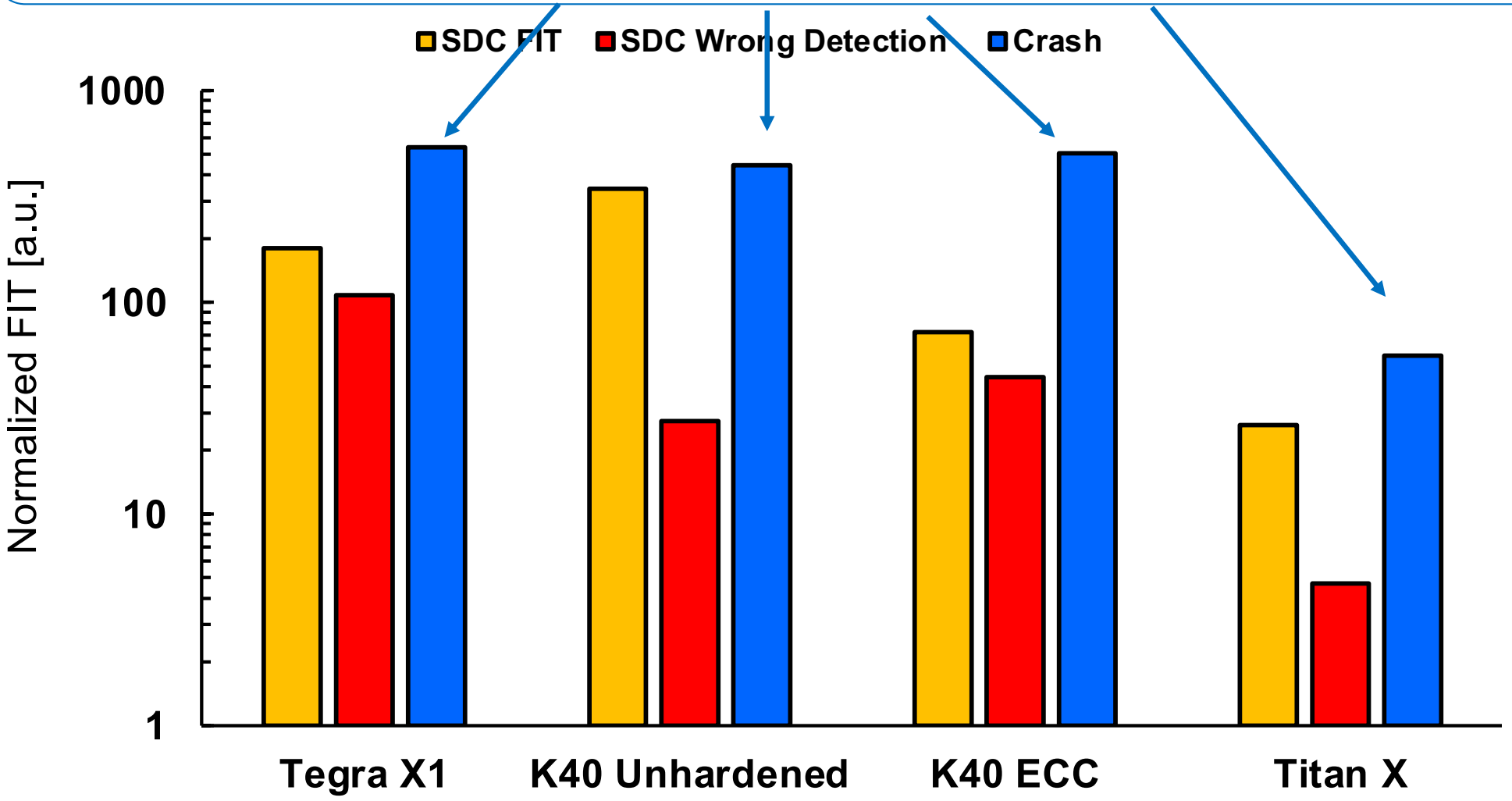
*G. Li, et at SC17

Classification Error

wrong object detects

Results – FIT*

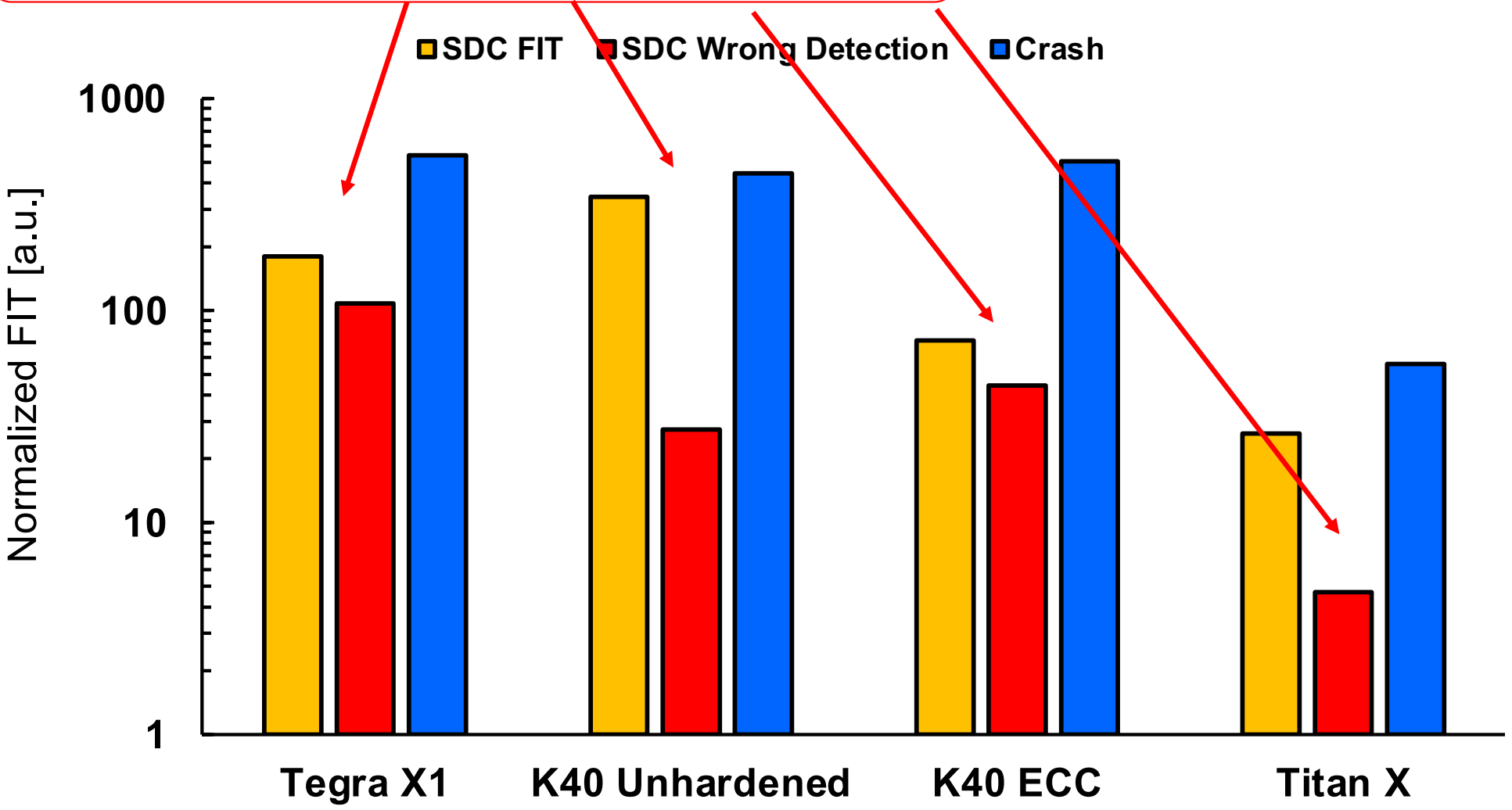
Crashes are always more probable than SDC.
(we know something happened => we can deal with it)



*F. F. dos Santos, et al. Trans. on Reliability 2019

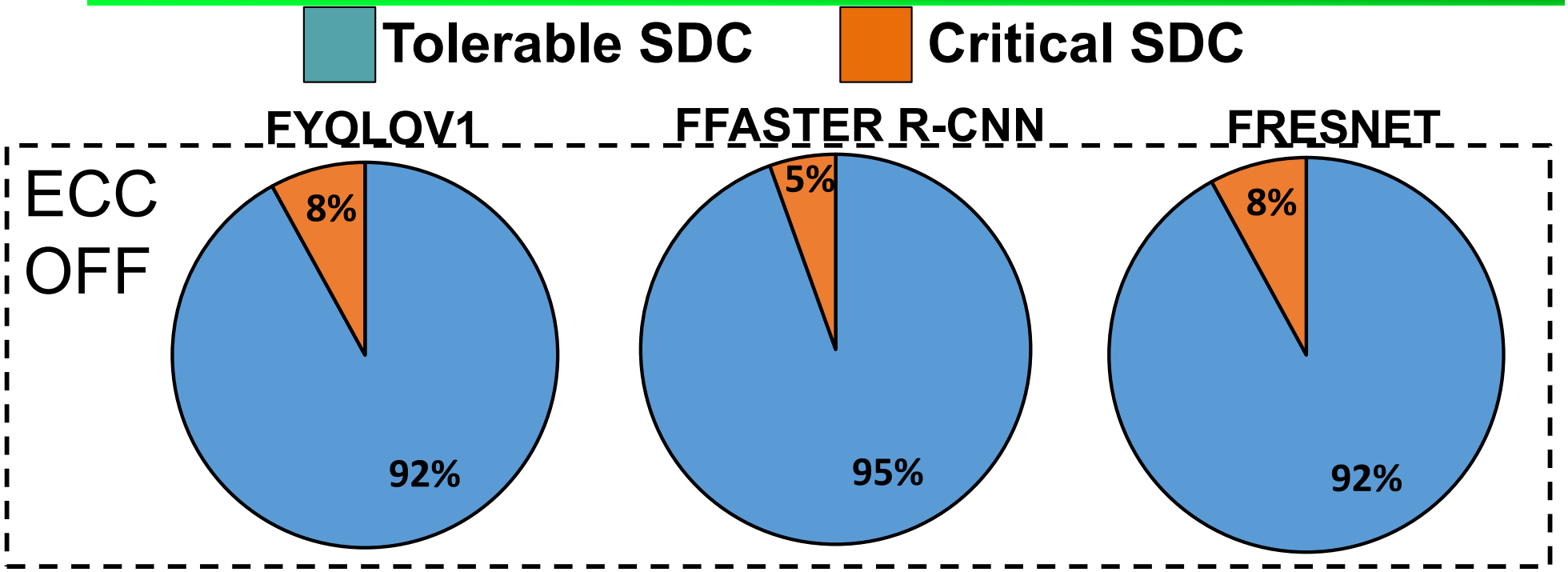
Results – FIT*

Not all SDCs affect detection!

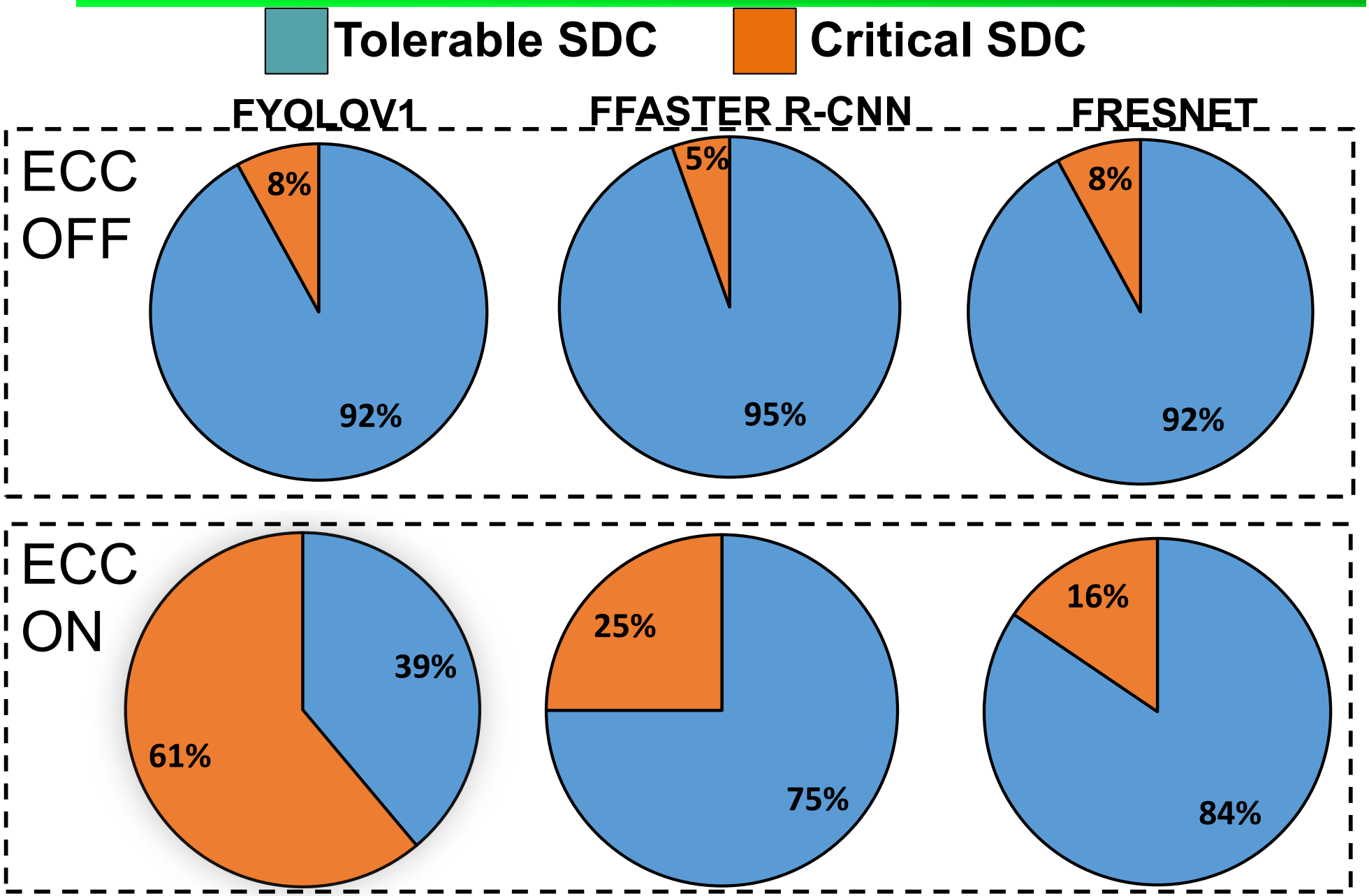


*F. F. dos Santos, et al. Trans. on Reliability 2019

Tolerable or Critical?



Tolerable or Critical?



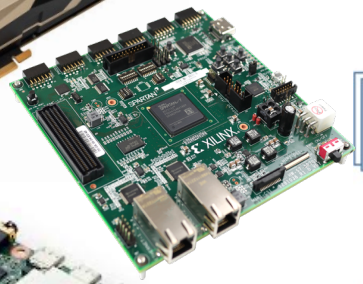
Outline

- Neutrons-induced effects in computing devices
- Evaluating neutron-induced errors probabilities
- **Cross layer faults propagation in CNNs**
- Some (interesting) efficient solutions
- Conclusions and Future Work

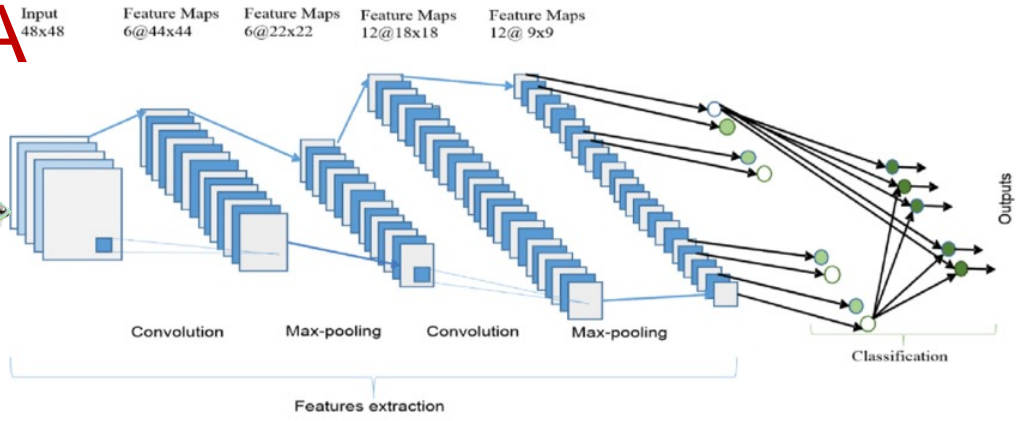
Faults Propagation

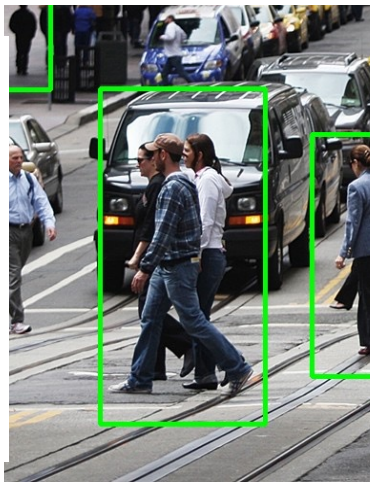
GPU

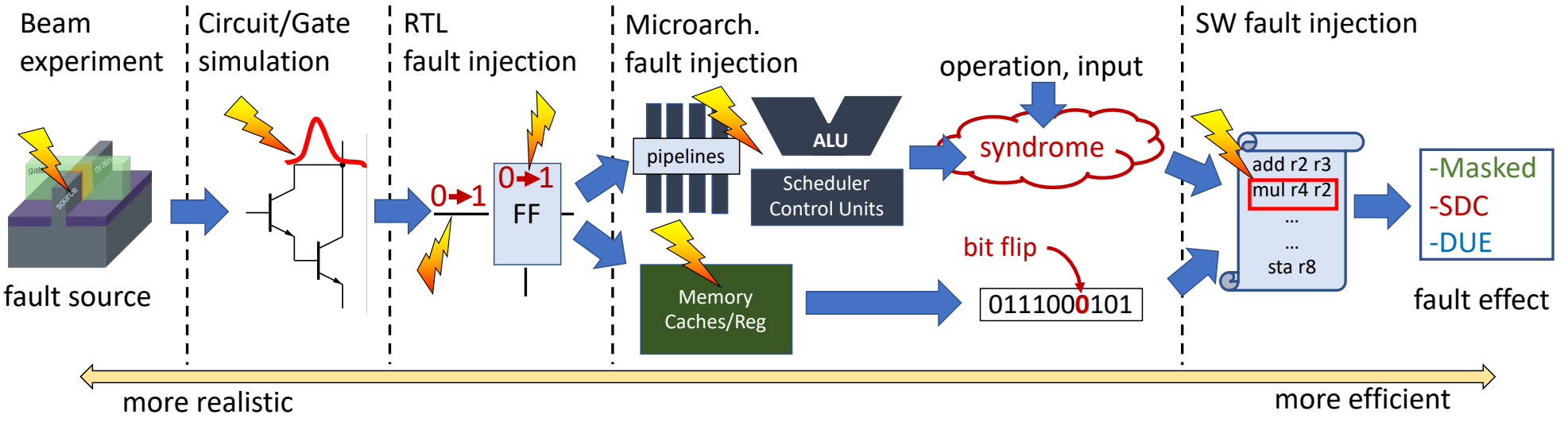
FPGA



TPU





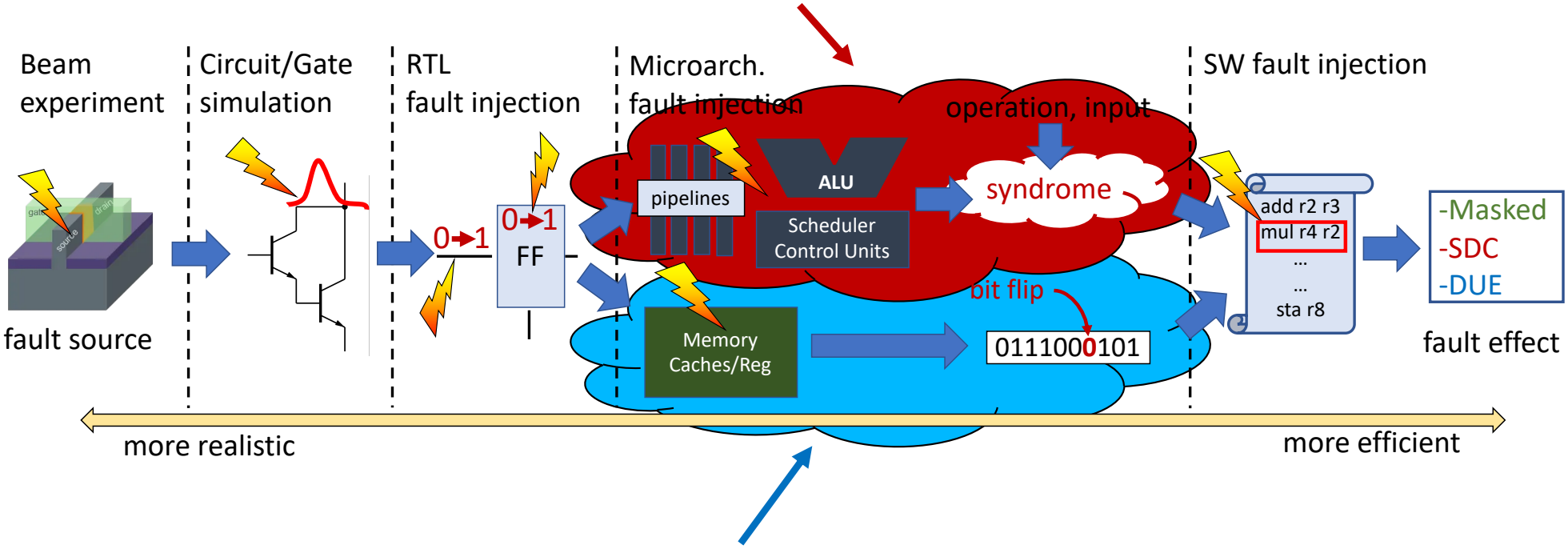


Faults Propagation

-Faults in logic have not-trivial syndrome on the output

-Largely unknown for complex devices

-No efficient protection available



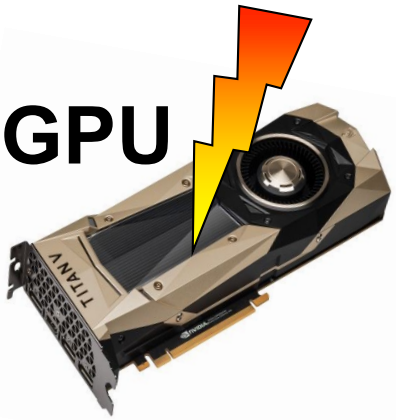
-Memory has a naïve fault model: single bit flips

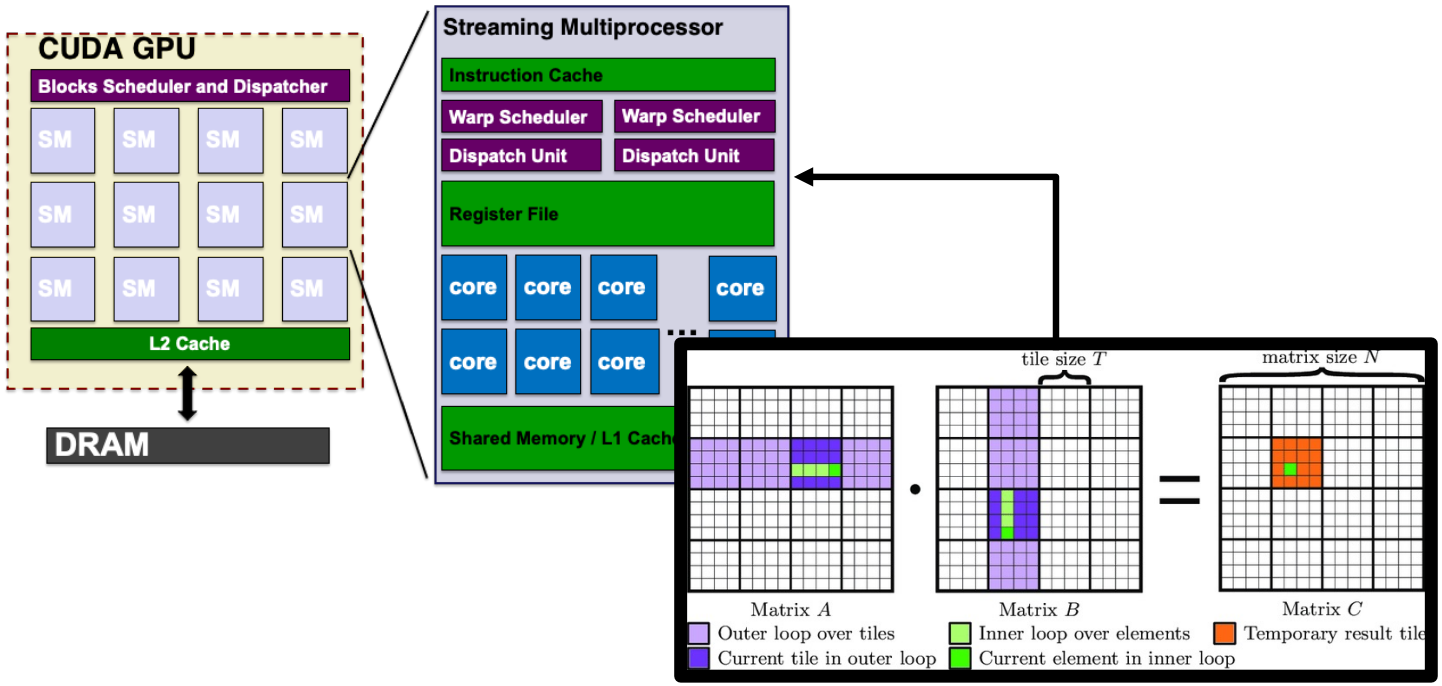
-Well studied for SRAM and DDR (since the 80s)

-Memory is easily protectable (ECC)

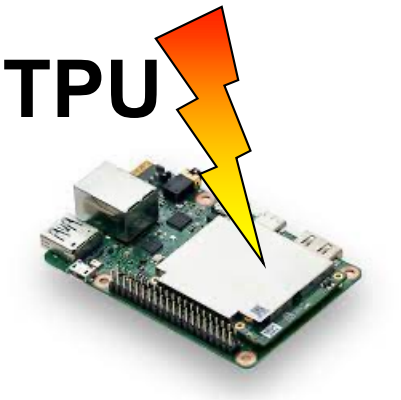
Convs errors on GPUs vs TPUs

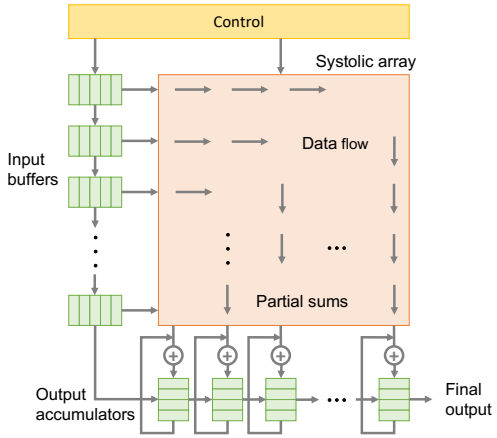
GPU





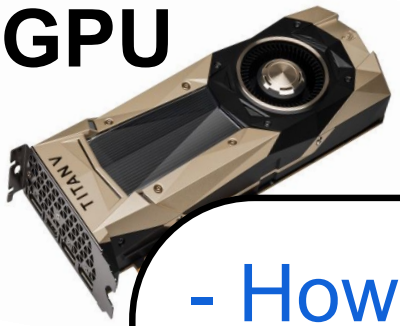
TPU

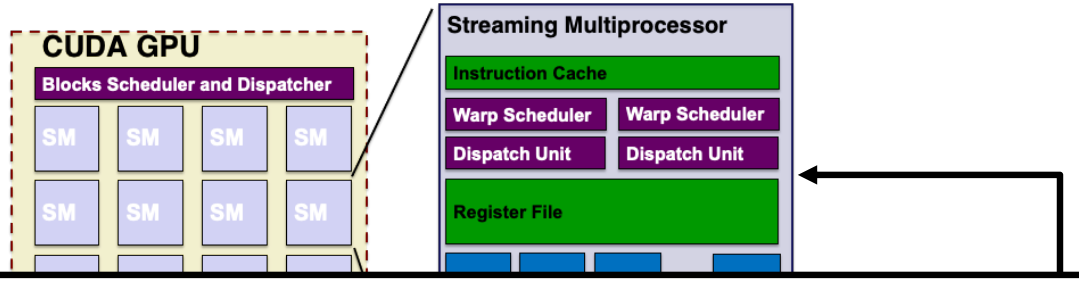




Convs errors on GPUs vs TPUs

GPU

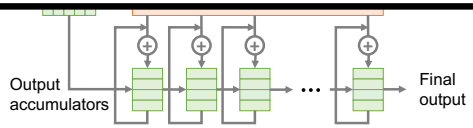




- How many elements in the convolution output matrix are corrupted?

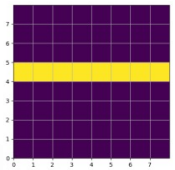
- How are they distributed?

*F. F. dos Santos, et al., Trans. on Reliability 2019
*R. L. Rech, et al., DATE 2022



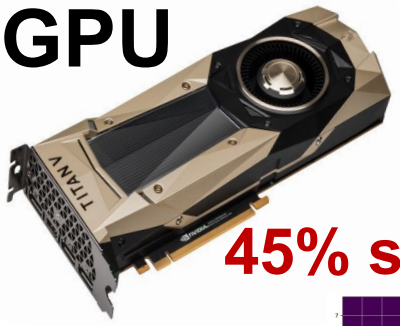
Convs errors on GPUs vs TPUs

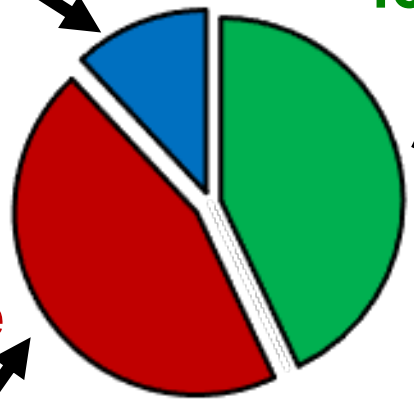
12% line



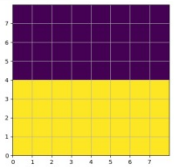
43% single

GPU





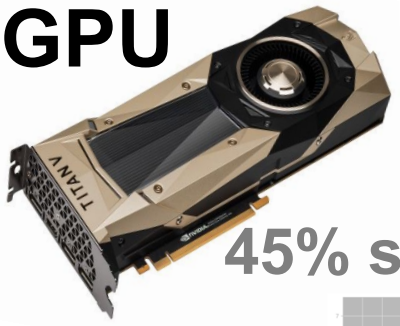
45% square



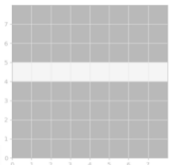
TPU

Convs errors on GPUs vs TPUs

GPU

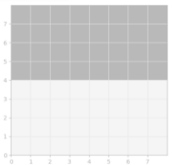


12% line



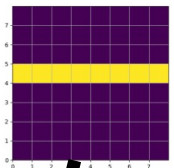
43% single

45% square

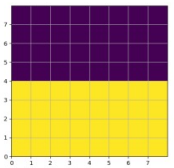


TPU

6% line

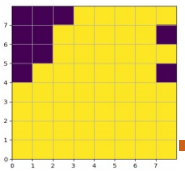


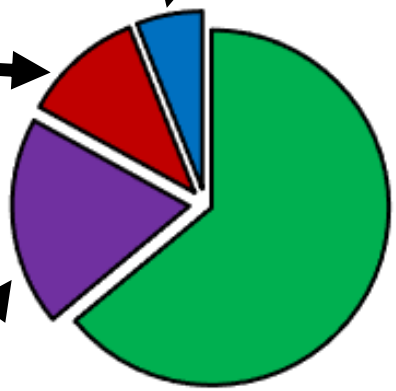
11% square



64% single

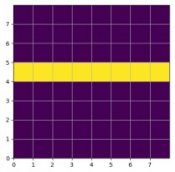
19% random





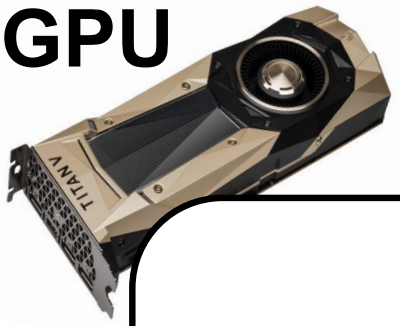
Convs errors on GPUs vs TPUs

12% line



43% single

GPU

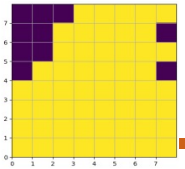


How different is the corrupted value from the expected one?

*F. F. dos Santos, et al., Trans. on Reliability 2019
*R. L. Rech, et al., DATE 2022

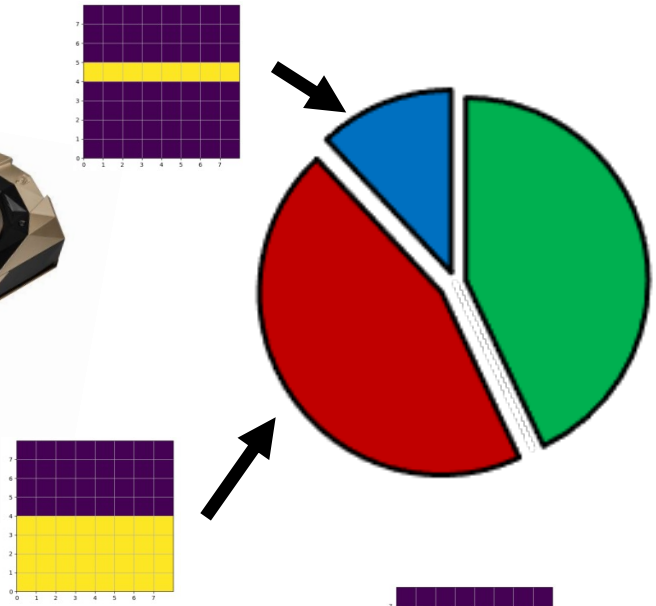
TPU

19% random

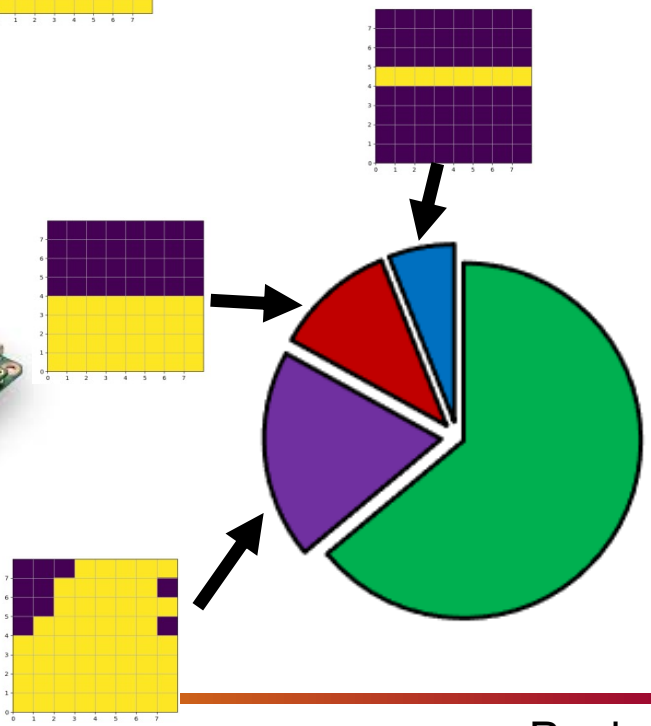


Convs errors on GPUs vs TPUs

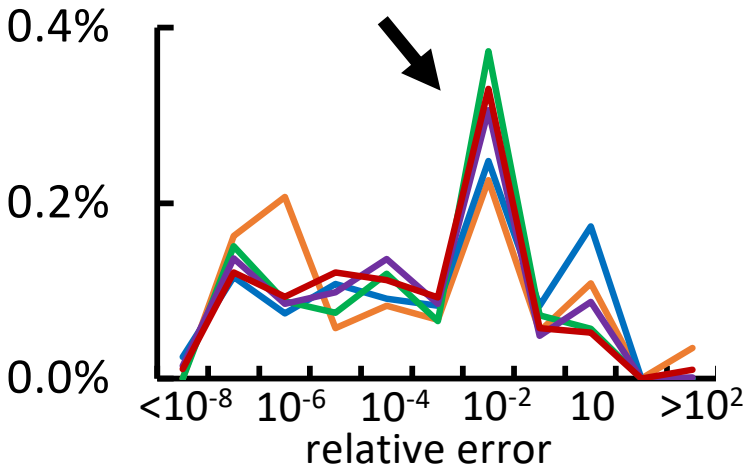
GPU



TPU

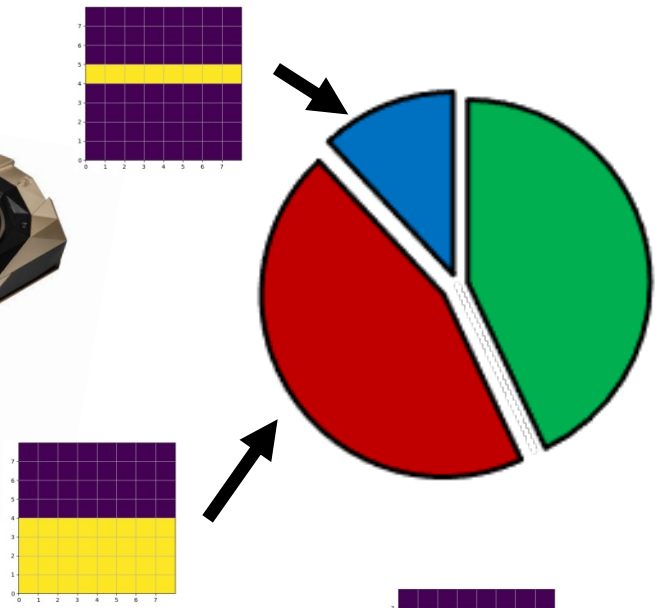


wide distribution of corrupted values

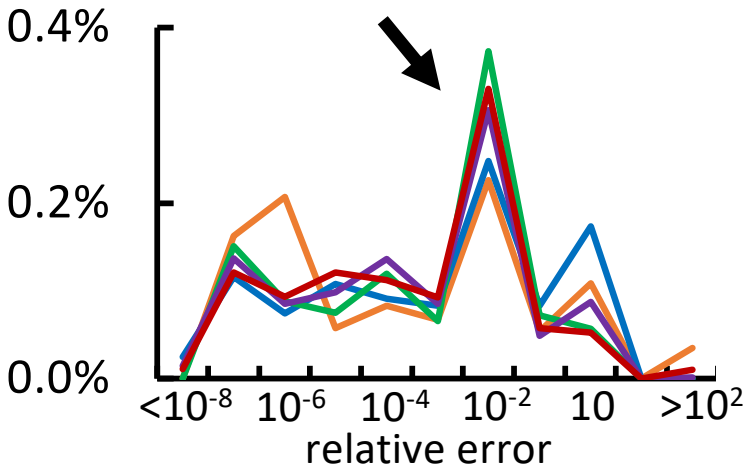


Convs errors on GPUs vs TPUs

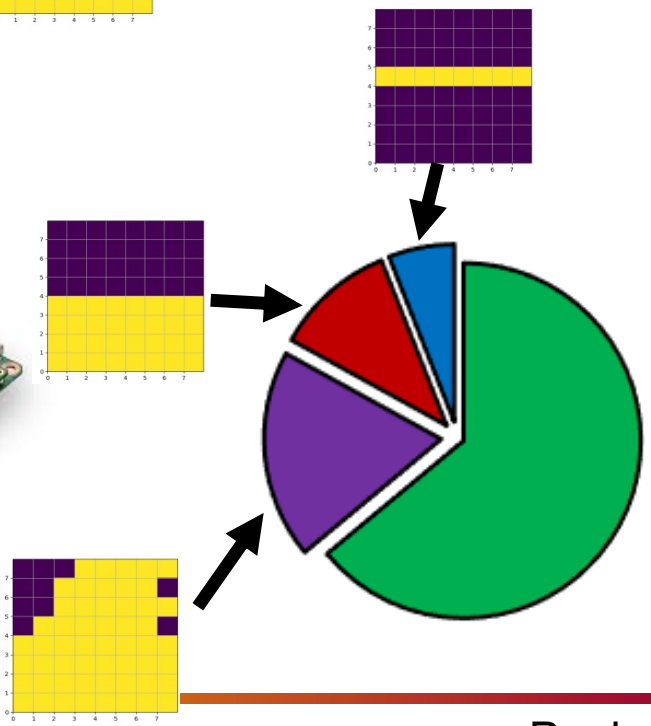
GPU



wide distribution of corrupted values

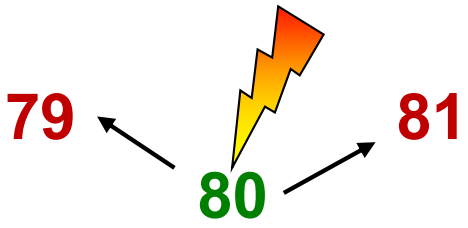


TPU

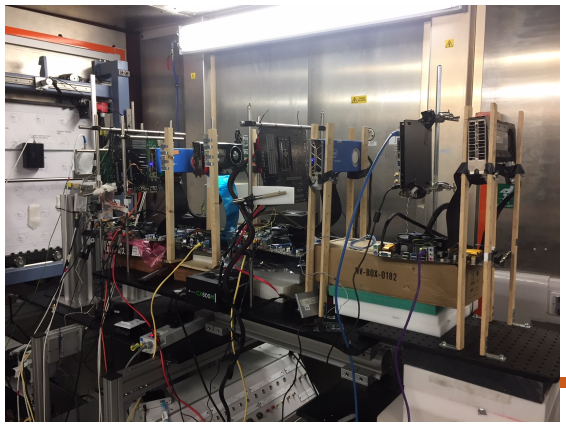


INT8

91% of errors are +/- 1

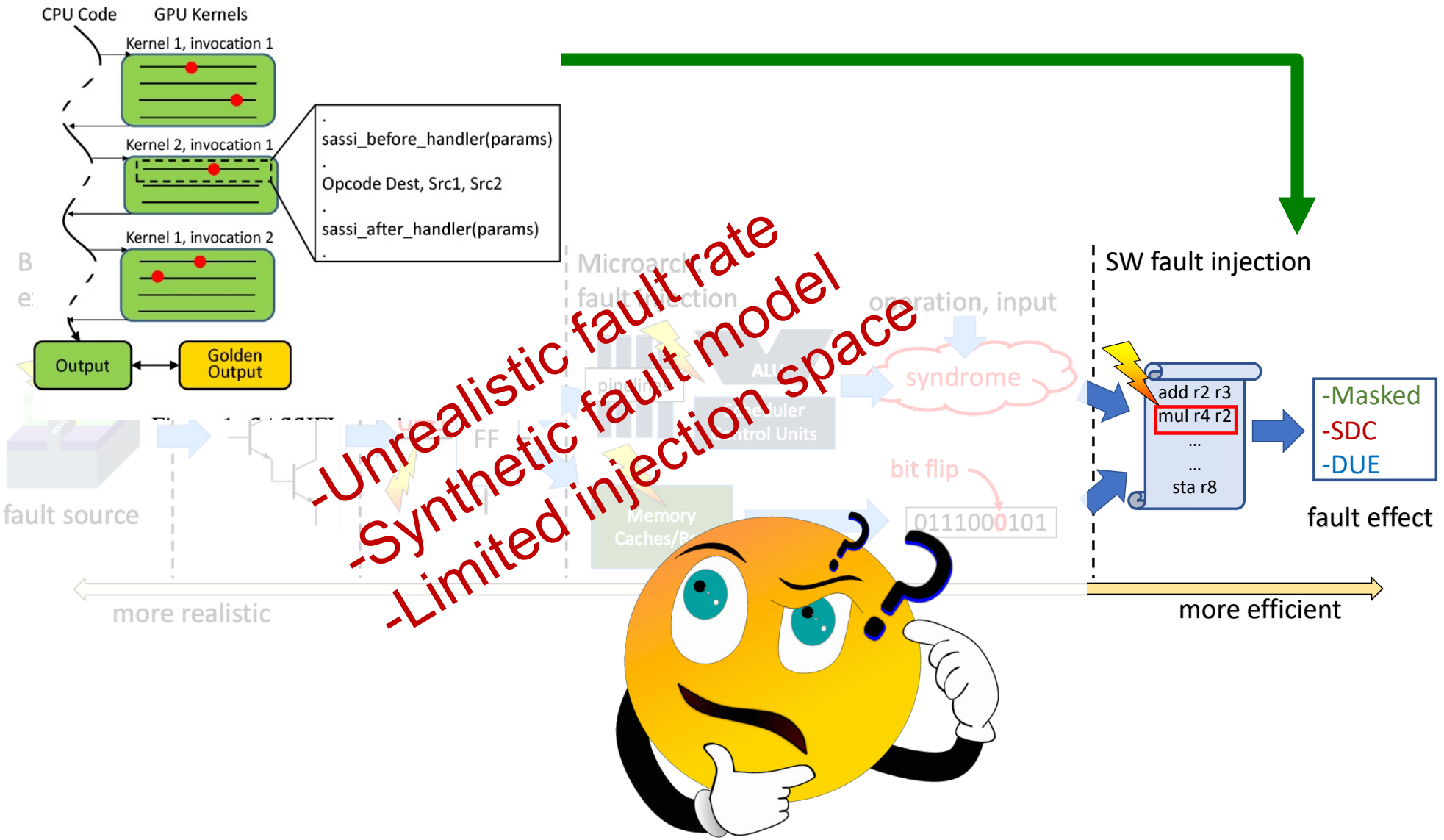


Faults Propagation



-Realistic error rate
-Realistic fault model
-All HW is exposed to neutrons

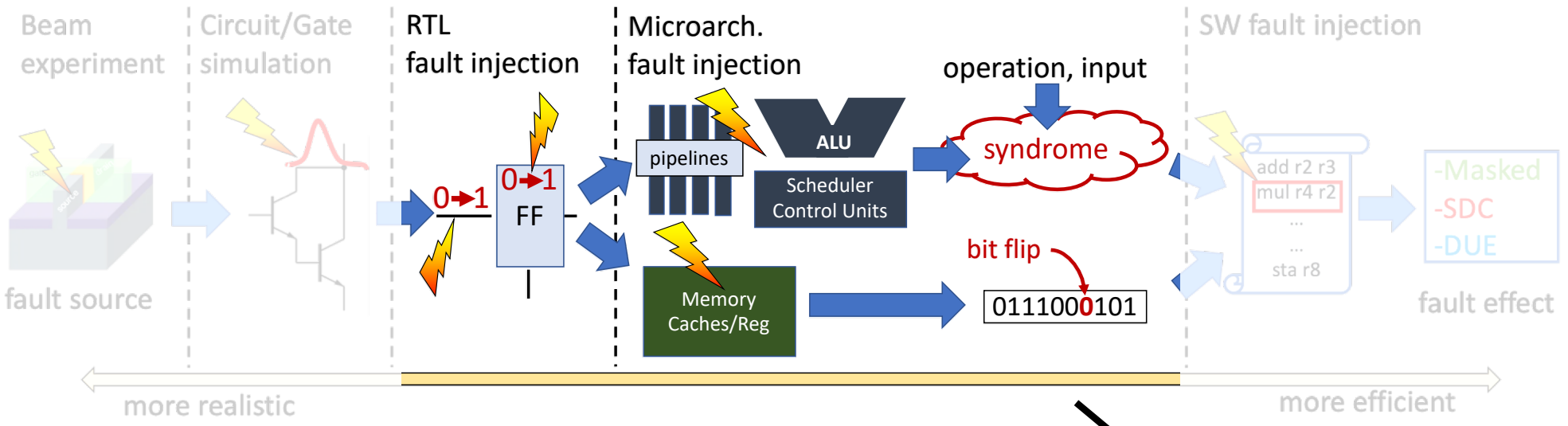
Faults Propagation



Faults Propagation

FlexGrip+ GPU model (F. F. dos Santos, DSN 2021)

GeFIN ARM model (P. Bodmann, Trans. Comp. 2021)



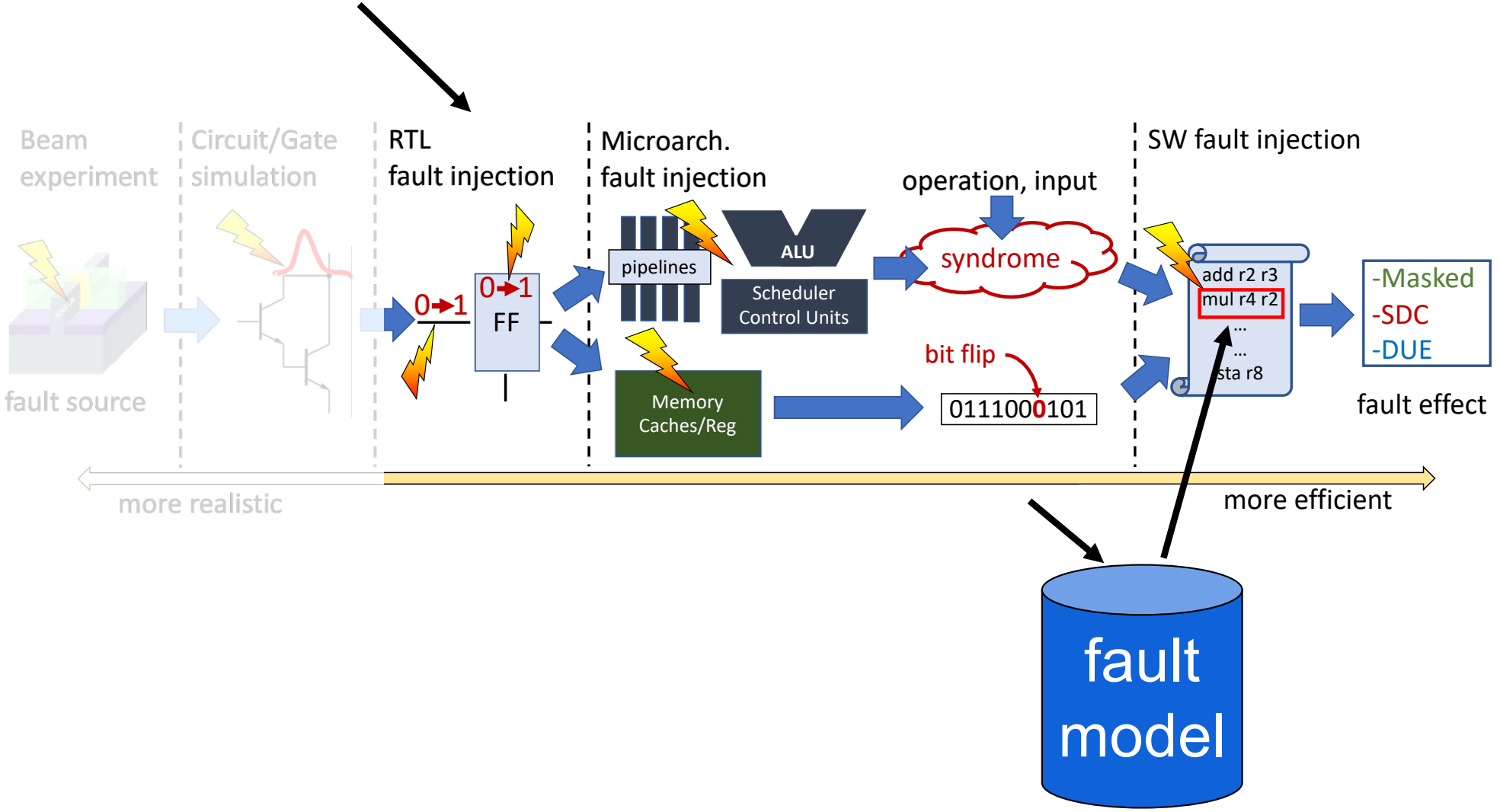
fault model

Characterization of the effects on micro-instructions

Faults Propagation

FlexGrip+ GPU model (F. F. dos Santos, DSN 2021)

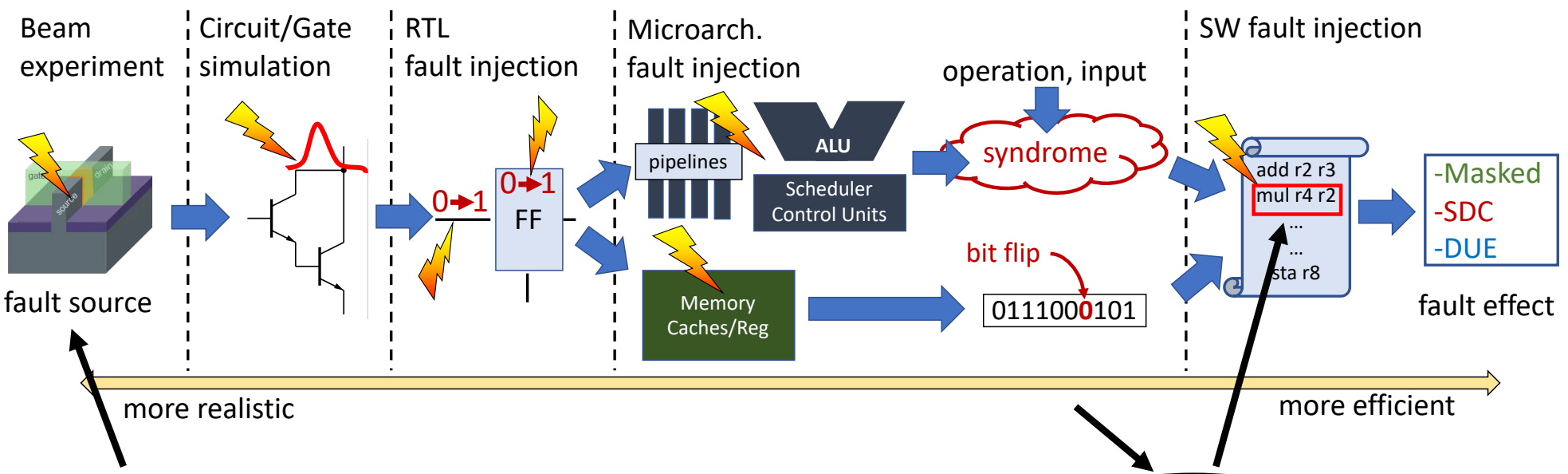
GeFIN ARM model (P. Bodmann, Trans. Comp. 2021)



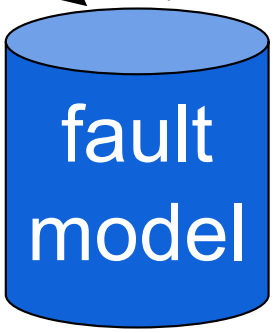
Faults Propagation

FlexGrip+ GPU model (F. F. dos Santos, DSN 2021)

GeFIN ARM model (P. Bodmann, Trans. Comp. 2021)

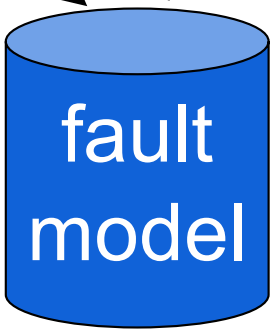


Beam experiments on micro-instructions (F. F. dos Santos, IPDPS 2021)



Faults Propagation

Beam experiments
on micro-instructions
(F. F. dos Santos, IPDPS 2021)



Outline

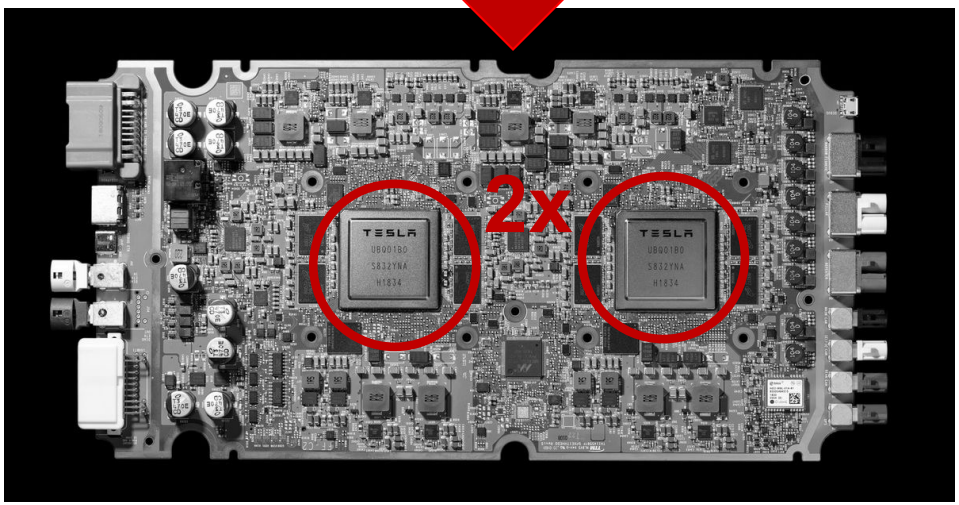
- Neutrons-induced effects in computing devices
- Evaluating neutron-induced errors probabilities
- Cross layer faults propagation in CNNs
- **Some (interesting) efficient solutions**
- Conclusions and Future Work

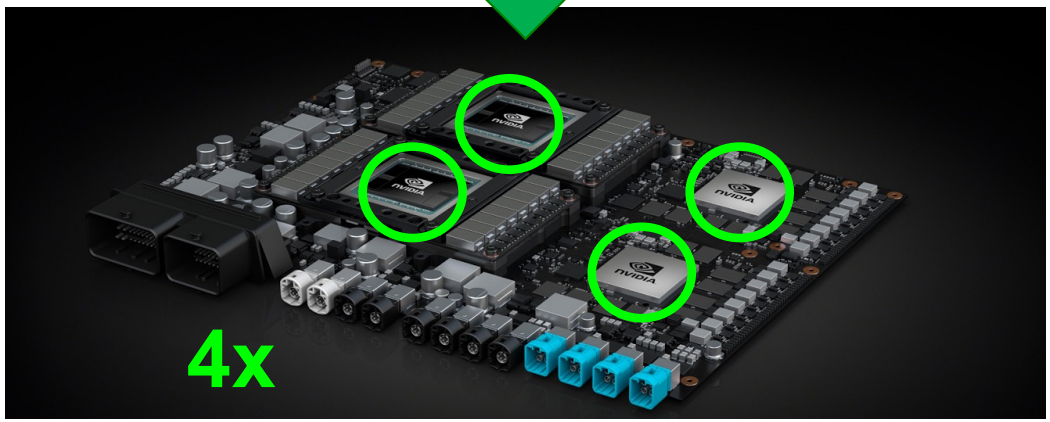
Self-Driven Cars

Naïve (expensive) solutions in today's self-driven cars

Replication is **very costly**!
And it might **not work always!**

We need to find **smarter ways** to detect neutron-induced errors.





Algorithm-Based Fault Tolerance

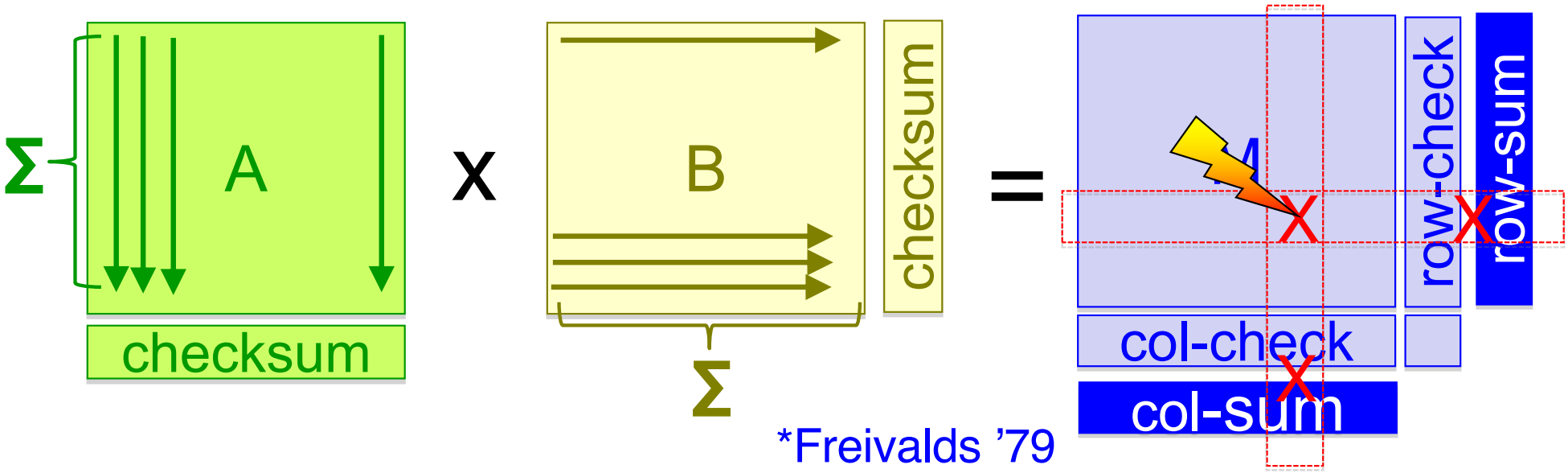
70% of CNN operations are GEMM-related

10% are the other kernels

20% CPUxGPU operations.

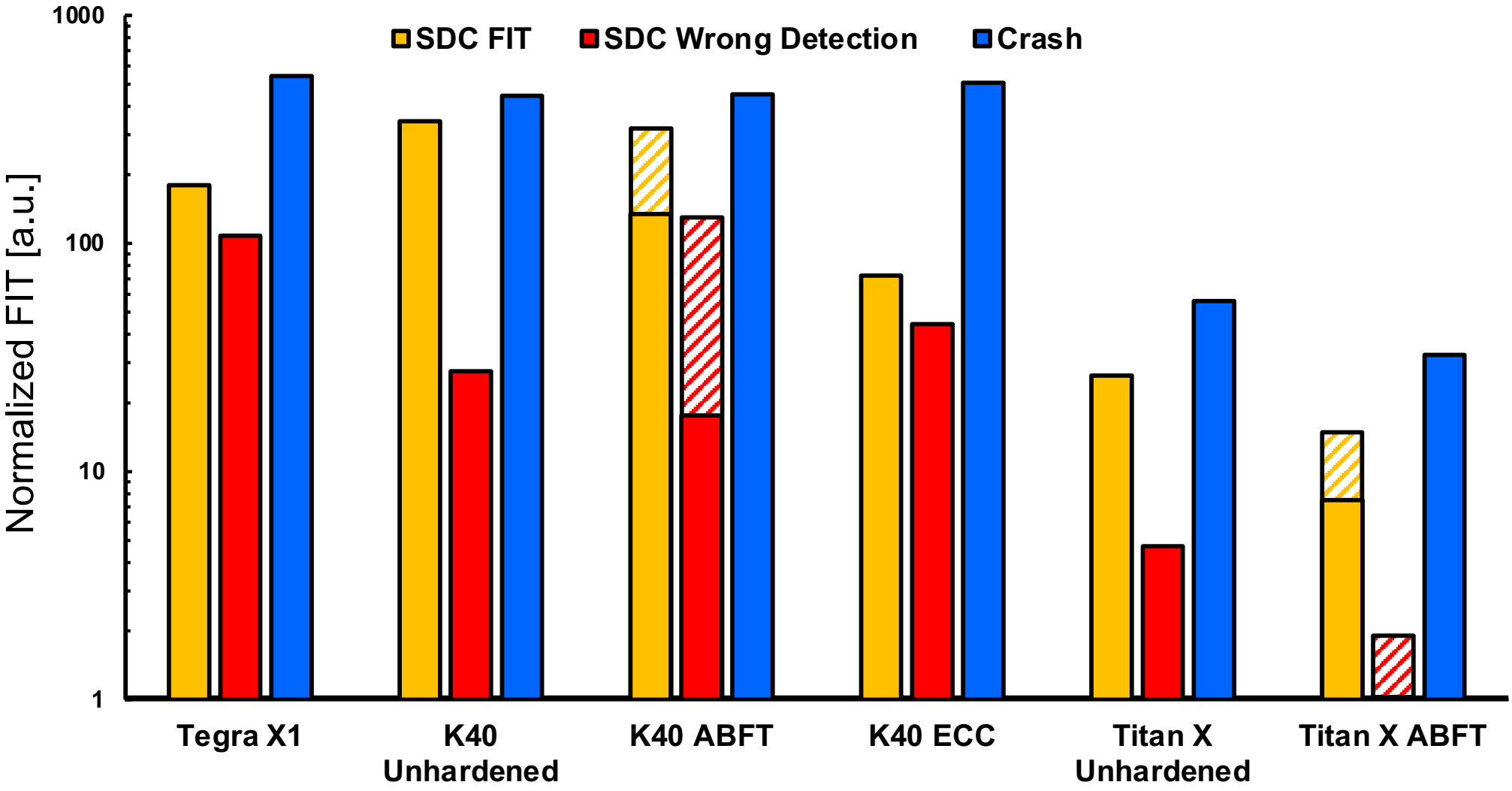
Proposed hardening: ABFT for Matrix multiplication*

* Huang and Abraham 1984
Rech et al., TNS 2013
S. Hari et al., TDSC 2021



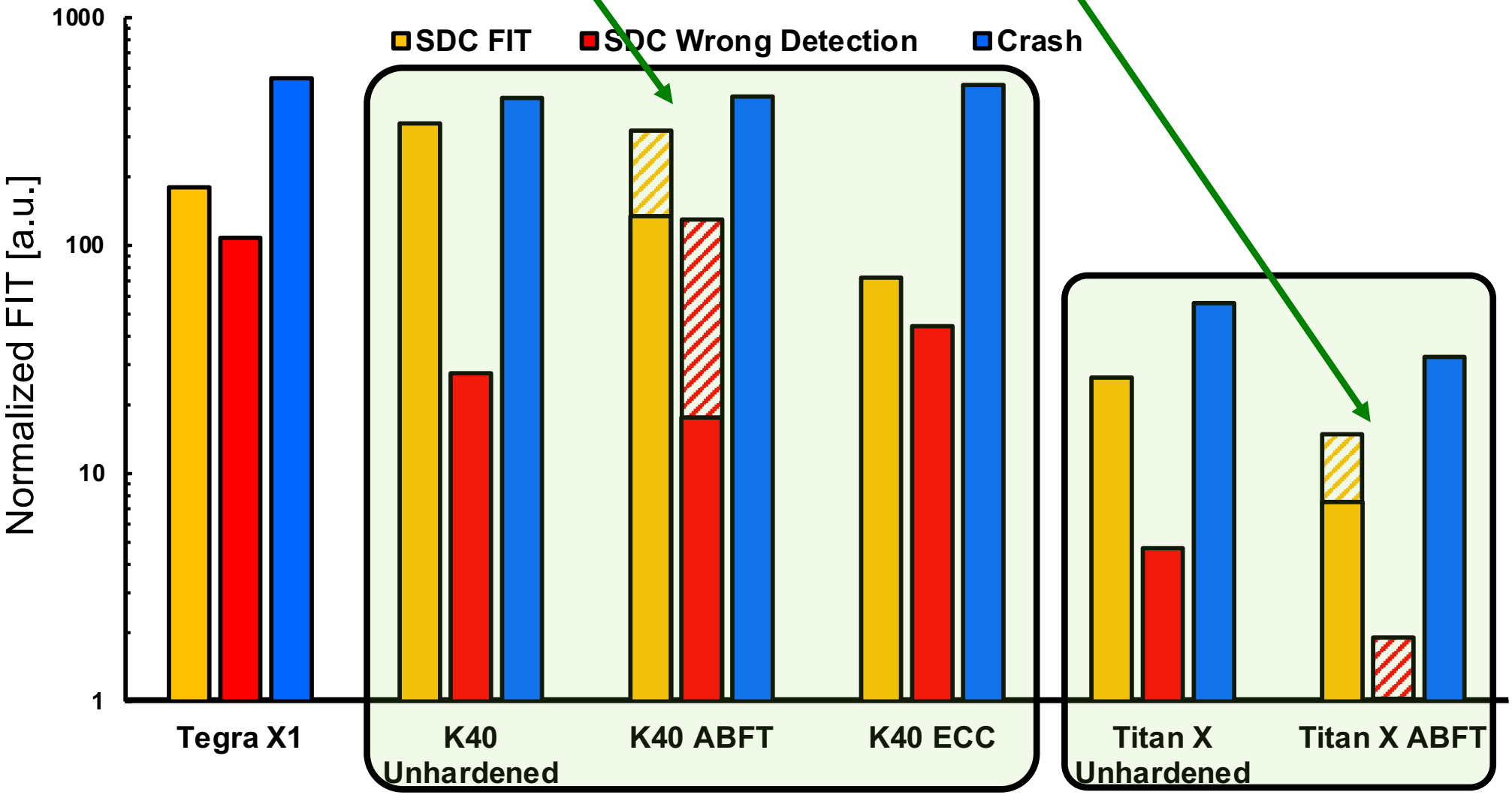
*Freivalds '79

ABFT works!

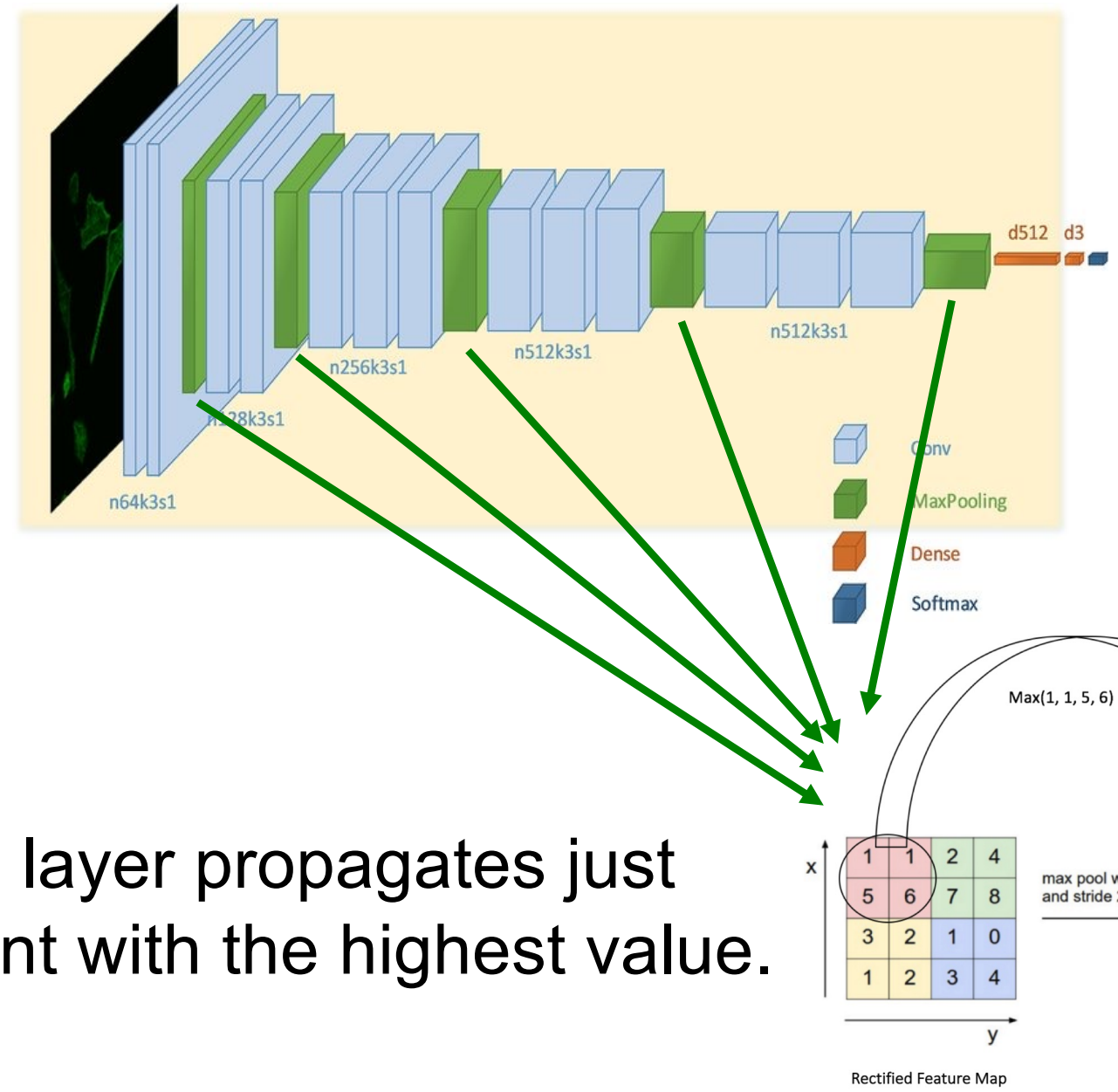

SDC FIT
SDC Wrong Detection
Crash
Normalized FIT [a.u.]
1000
100
10
1
Tegra X1
K40 Unhardened
K40 ABFT
K40 ECC
Titan X Unhardened
Titan X ABFT

ABFT works!

Our ABFT corrects 87% of Critical SDC



Max-Pool

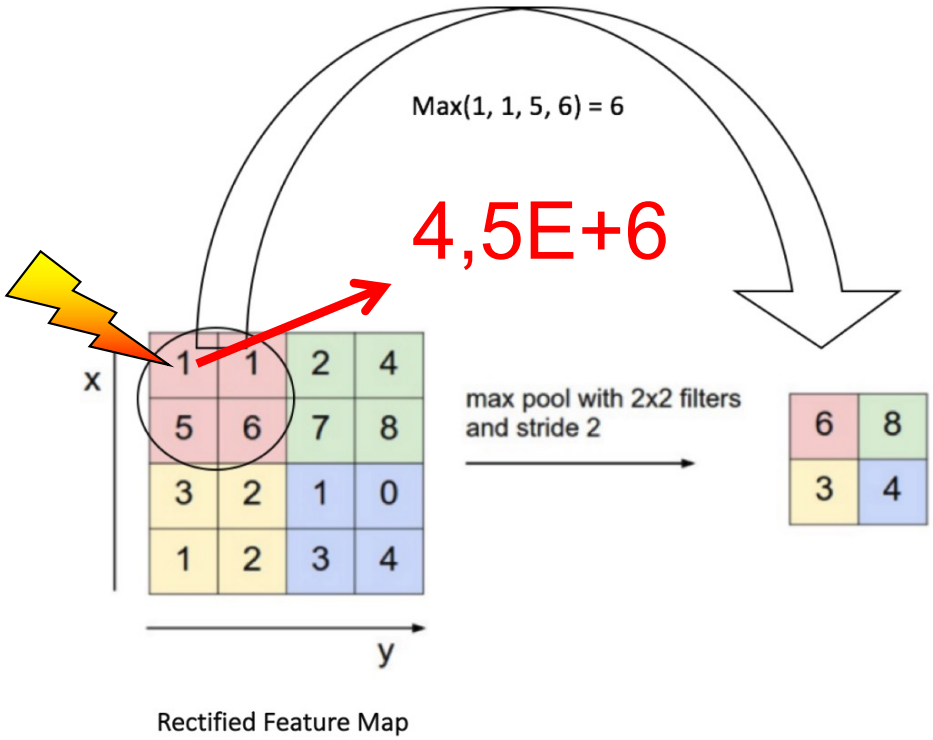


Max-Pool layer propagates just the element with the highest value.

Smart Max-Pool

If the value of the element to propagate is ***not reasonable*** (10x max value of a fault-free execution) we detect the error and discard the frame.

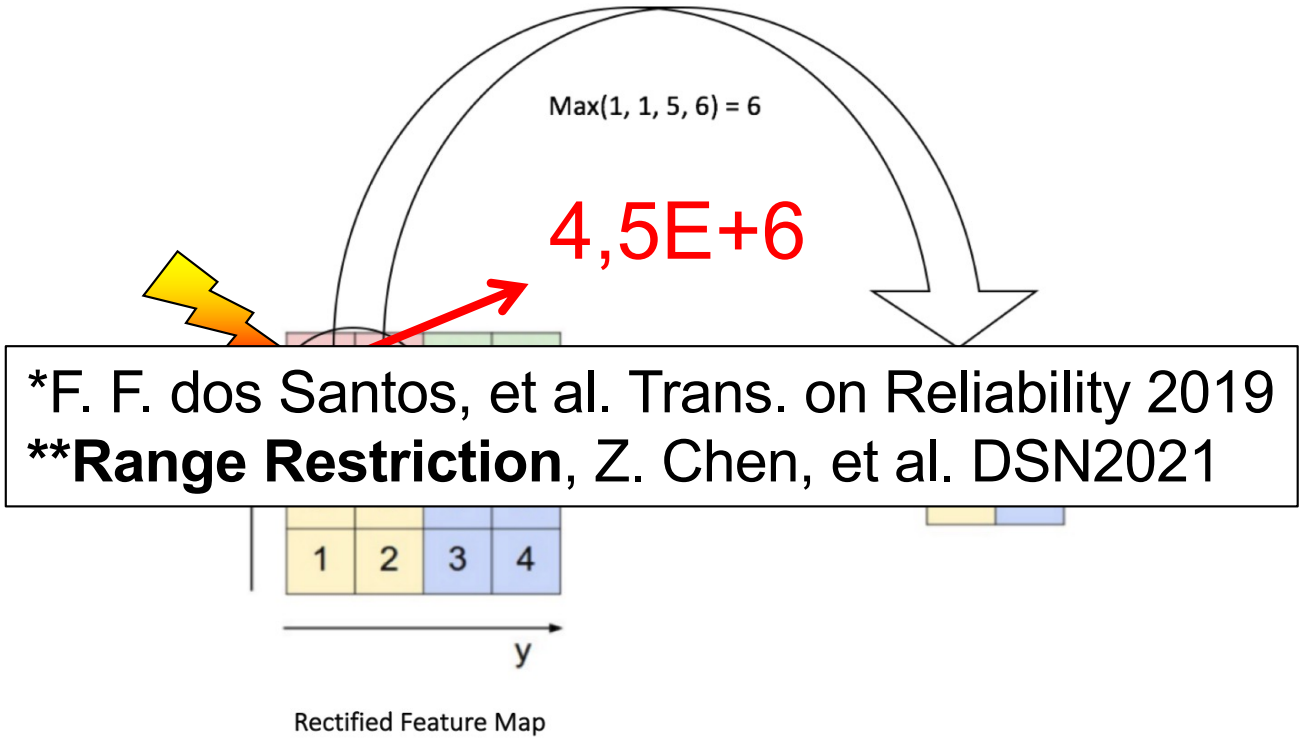
4 additional variables, detection in O(1)



Smart Max-Pool

If the value of the element to propagate is ***not reasonable*** (10x max value of a fault-free execution) we detect the error and discard the frame.

4 additional variables, detection in O(1)



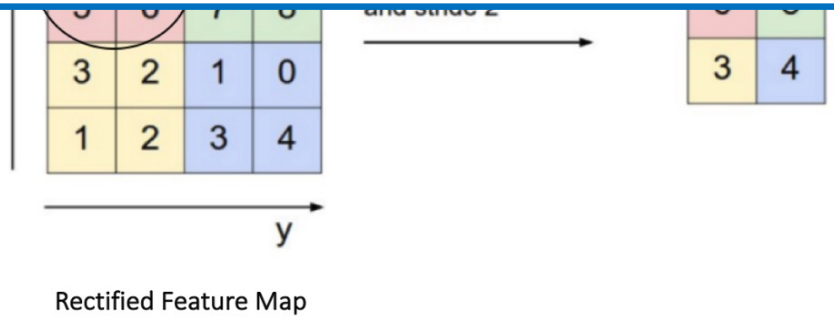
*F. F. dos Santos, et al. Trans. on Reliability 2019
****Range Restriction**, Z. Chen, et al. DSN2021

Smart Max-Pool

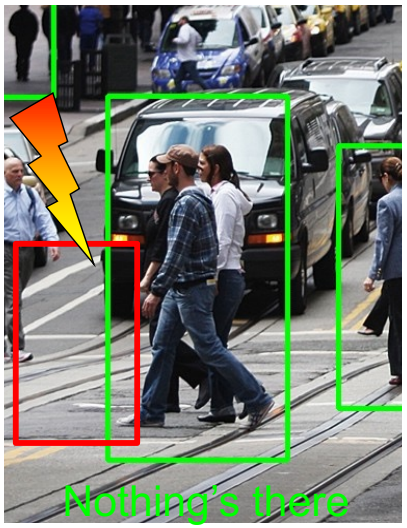
If the value of the element to propagate is ***not reasonable*** (10x max value of a fault-free execution) we detect the error and discard the frame.

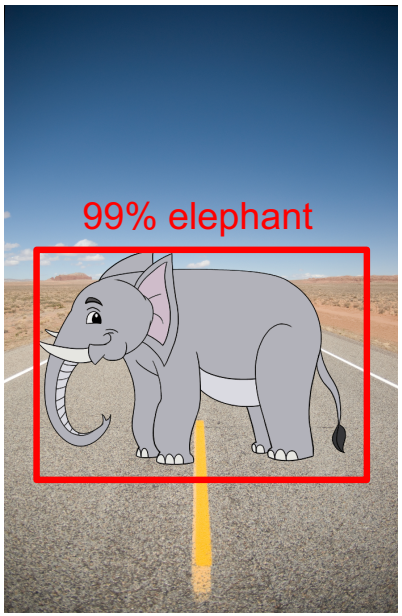
4 additional variables, detection in O(1)

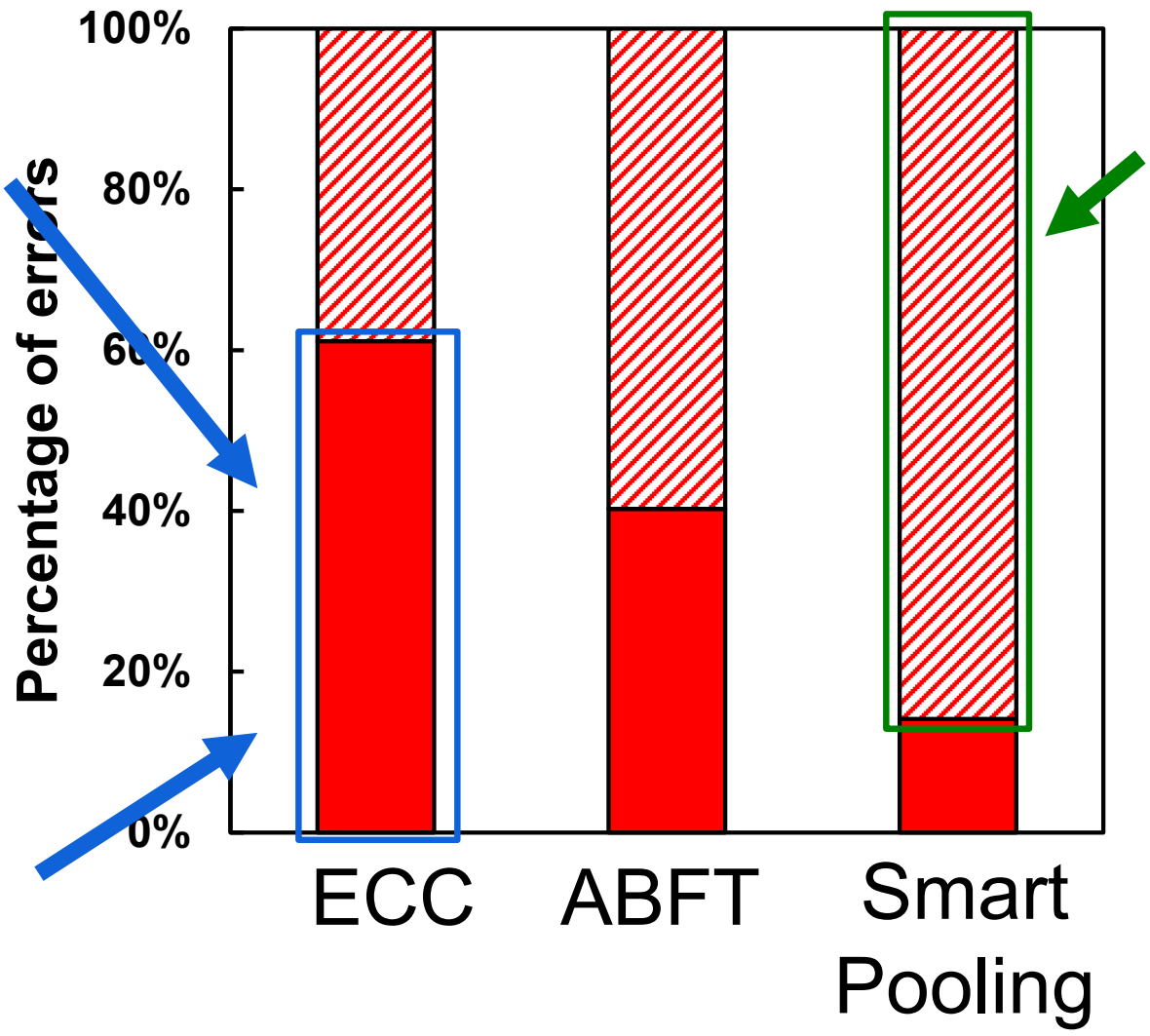
Smart-pool detects more than 90% of critical SDCs

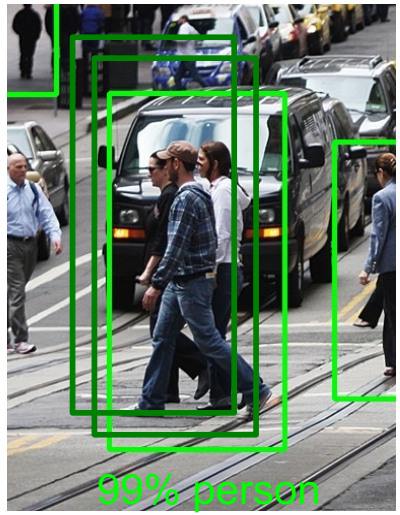


ECC vs ABFT vs Smart Pooling*









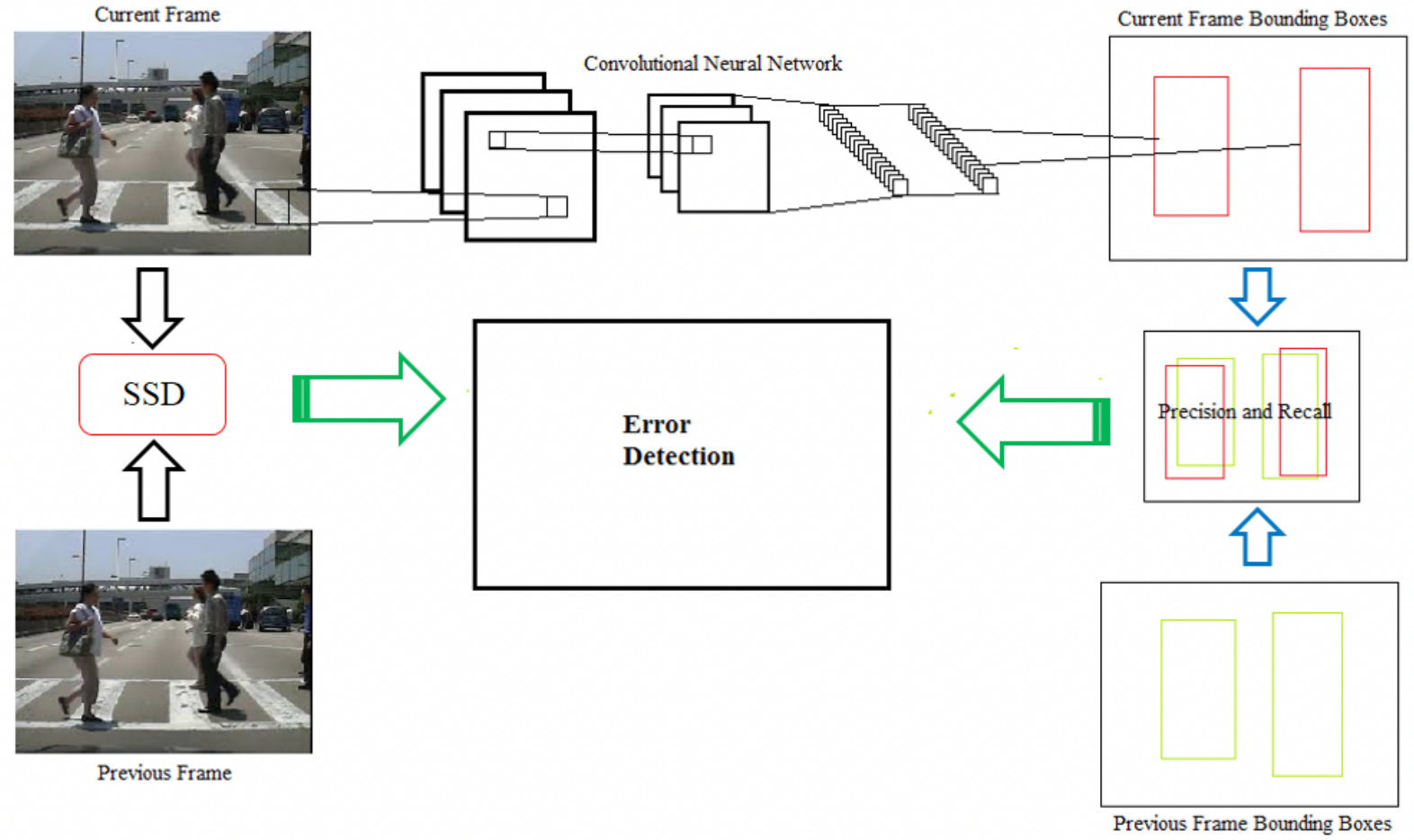
*F. F. dos Santos, et al. Trans. on Reliability 2019

Space-Time Correlation

CNN processes each frame independently from others.

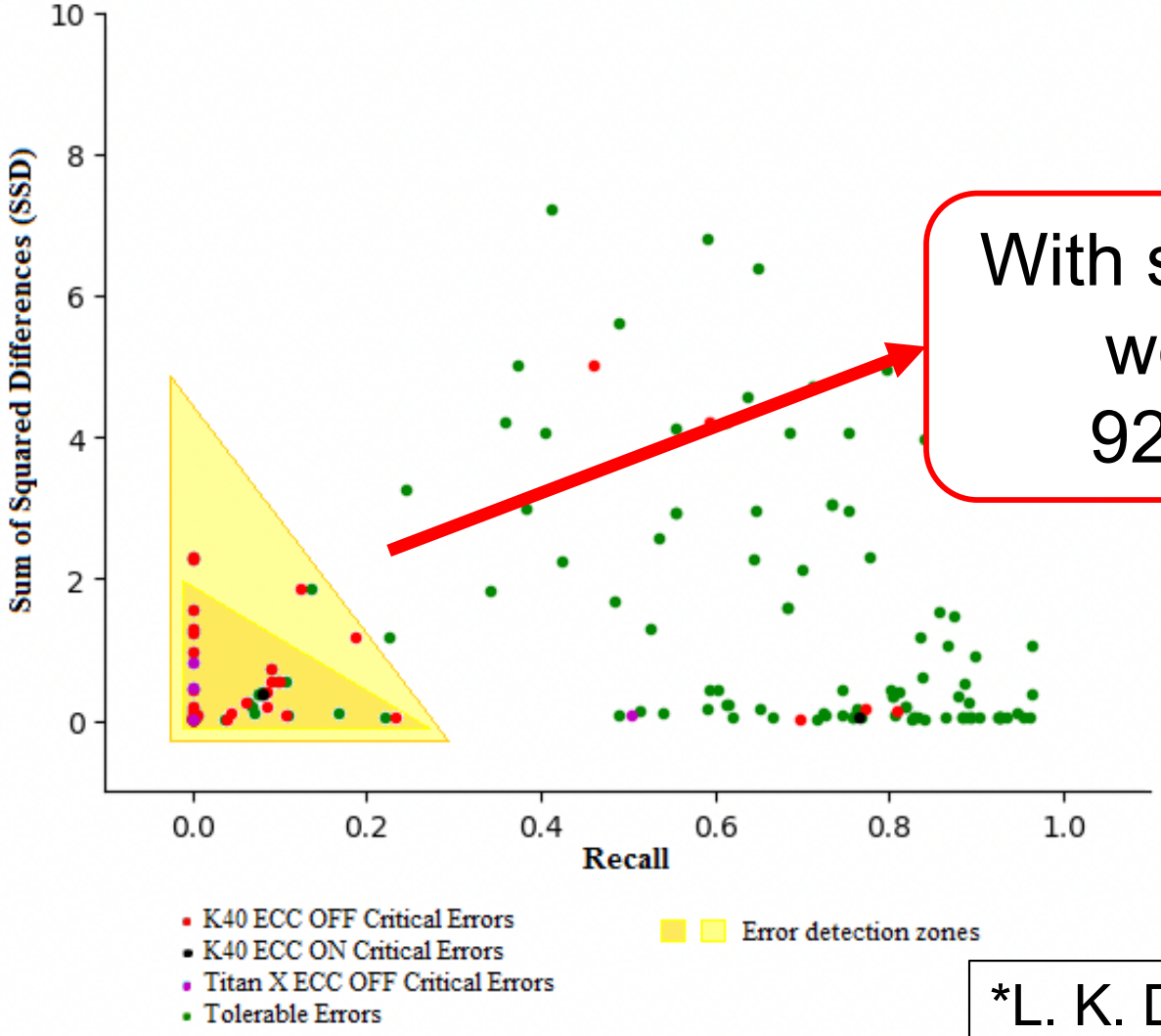
We process frames correlating subsequent frames.

Frames are highly correlated. So should detection.



Space-Time Correlation*

If similar frames produce uncorrelated detection probably an error happened

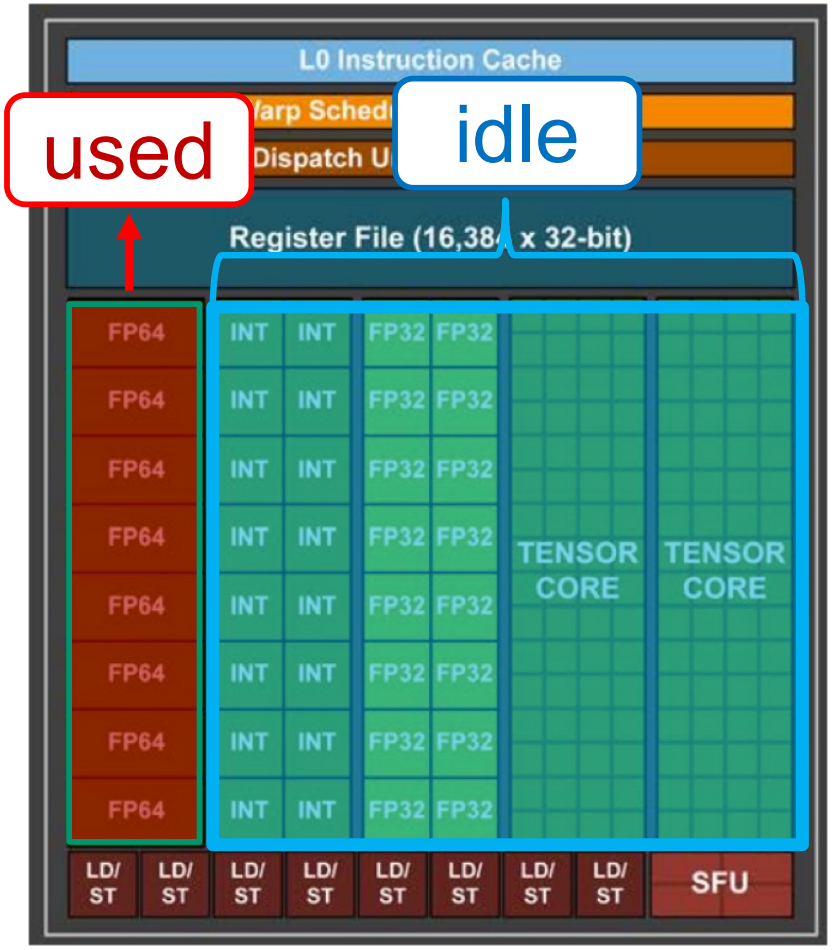


With space-time correlation we can detect up to 92% of critical errors

*L. K. Draghetti, et al., IOLTS 2019

Mixed-Precision Hardening

GPUs have dedicated functional units to execute **FP64**, **FP32**, **FP16** operations and **Tensor Core**

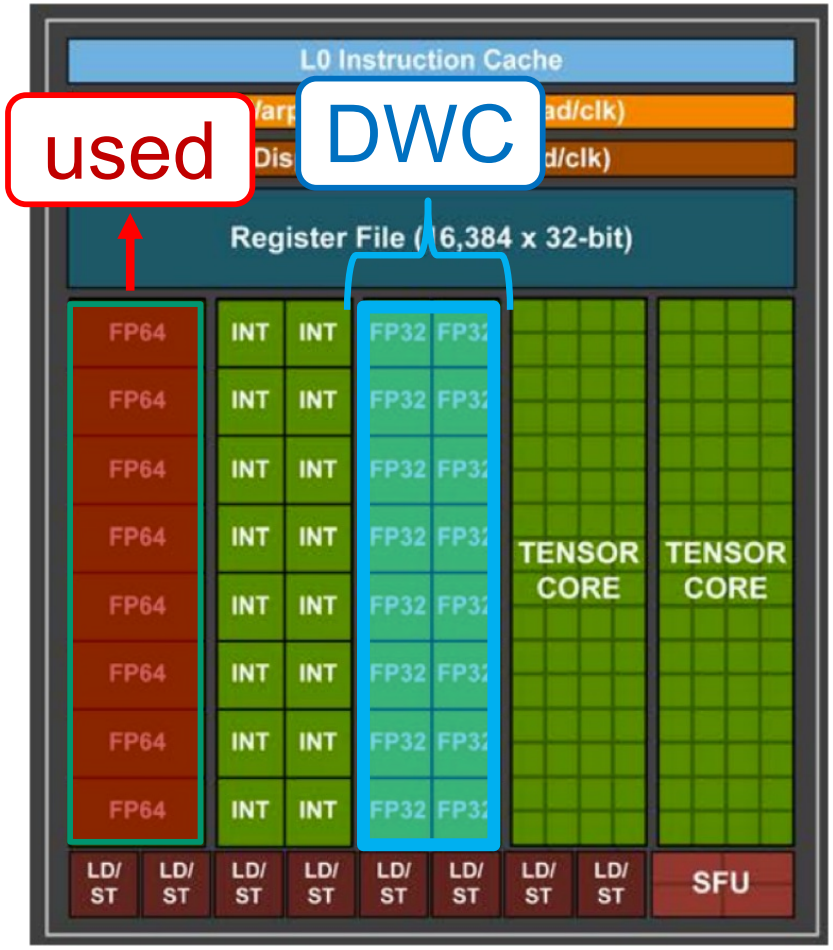


Source: NVIDIA

When a FP64 application is executed, the other units are idle.

Mixed-Precision Hardening

GPUs have dedicated functional units to execute **FP64**, **FP32**, **FP16** operations and **Tensor Core**



Source: NVIDIA

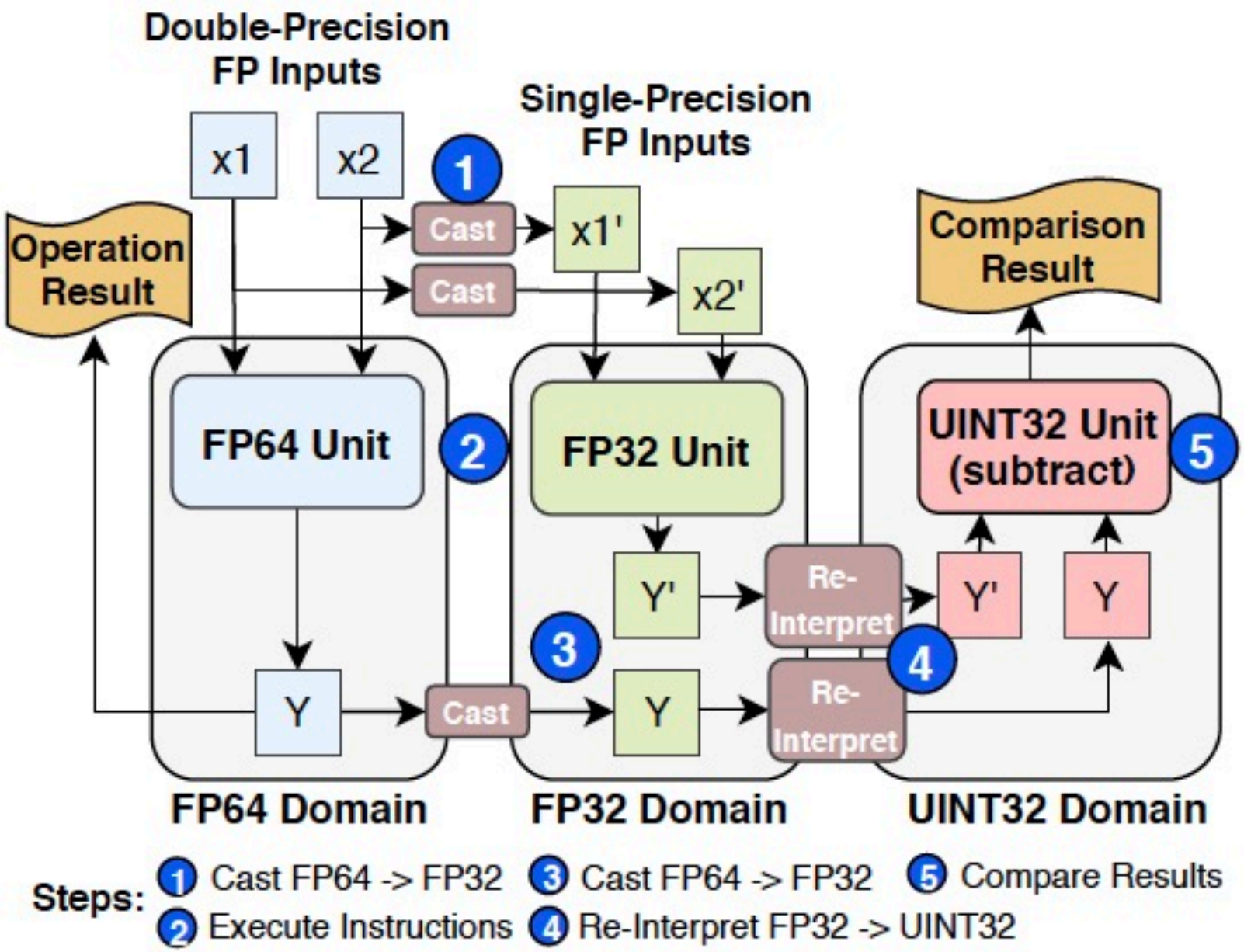
When a FP64 application is executed, the other units are idle.

Our idea is to **run the same code**, in parallel, **in the available FP32 cores**.

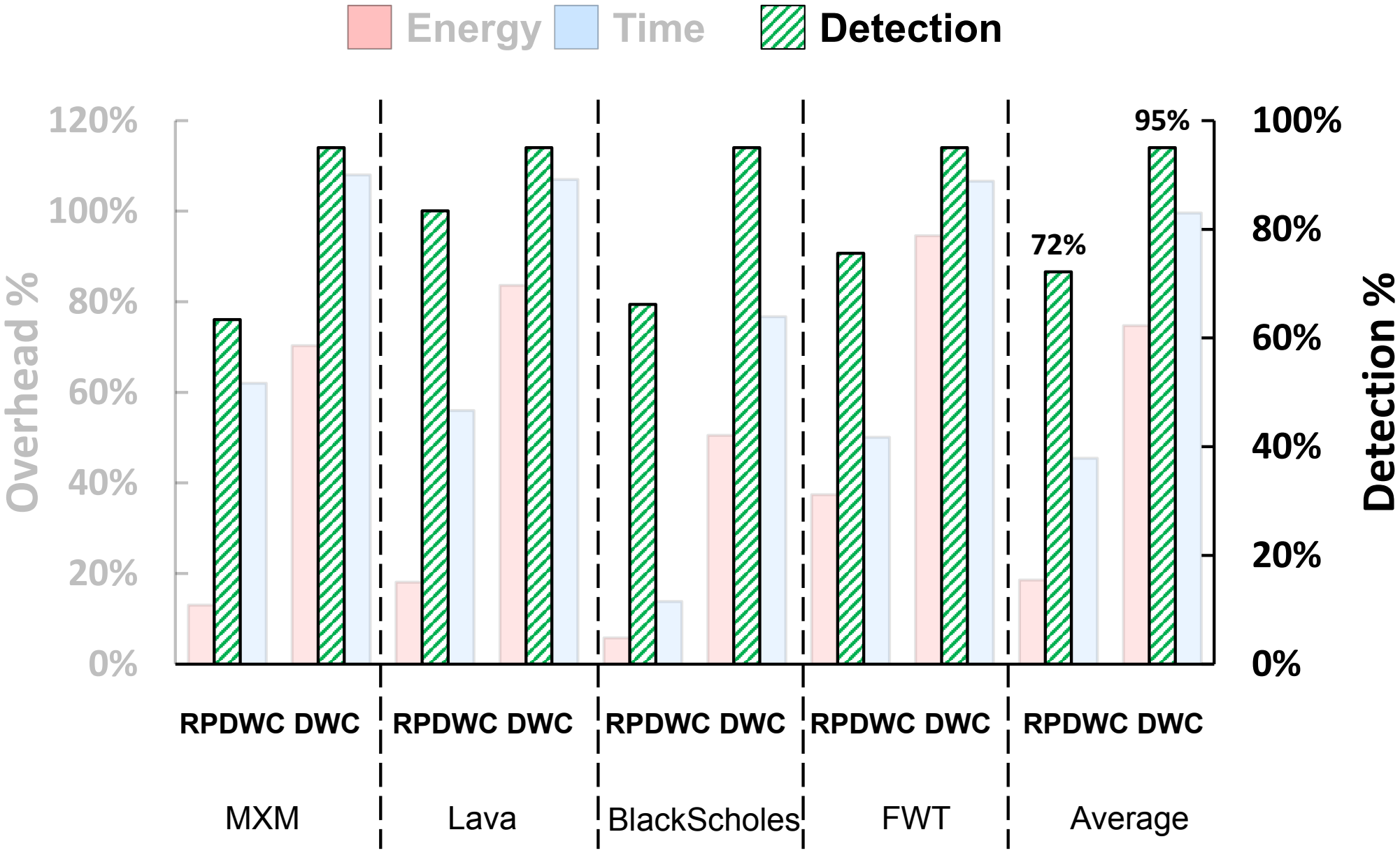
Reduced-Precision RP-DWC*

*F. F. dos Santos, et al. Trans. Comp. 2021

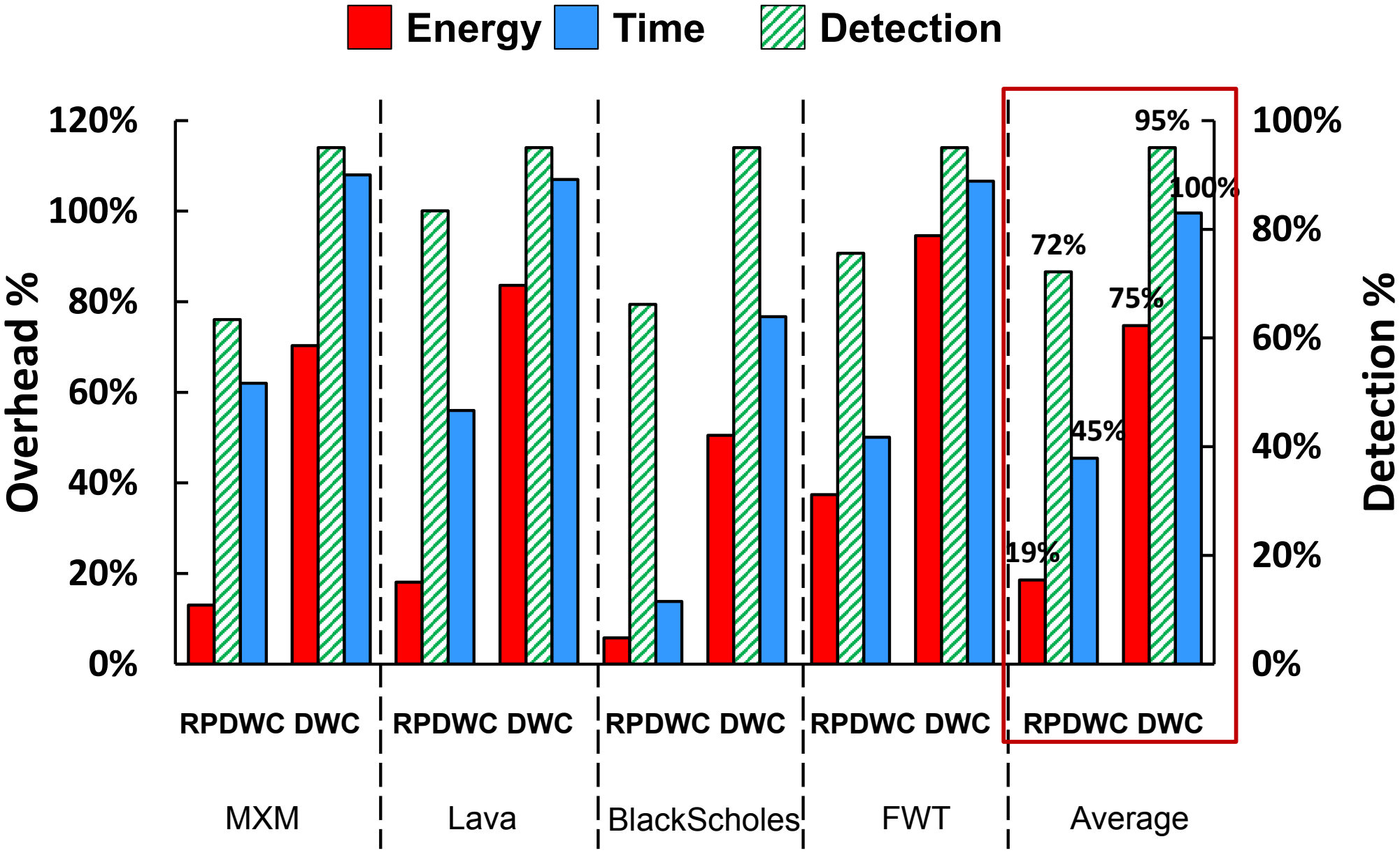
Mixed-Precision Hardening



Mixed-Precision Hardening

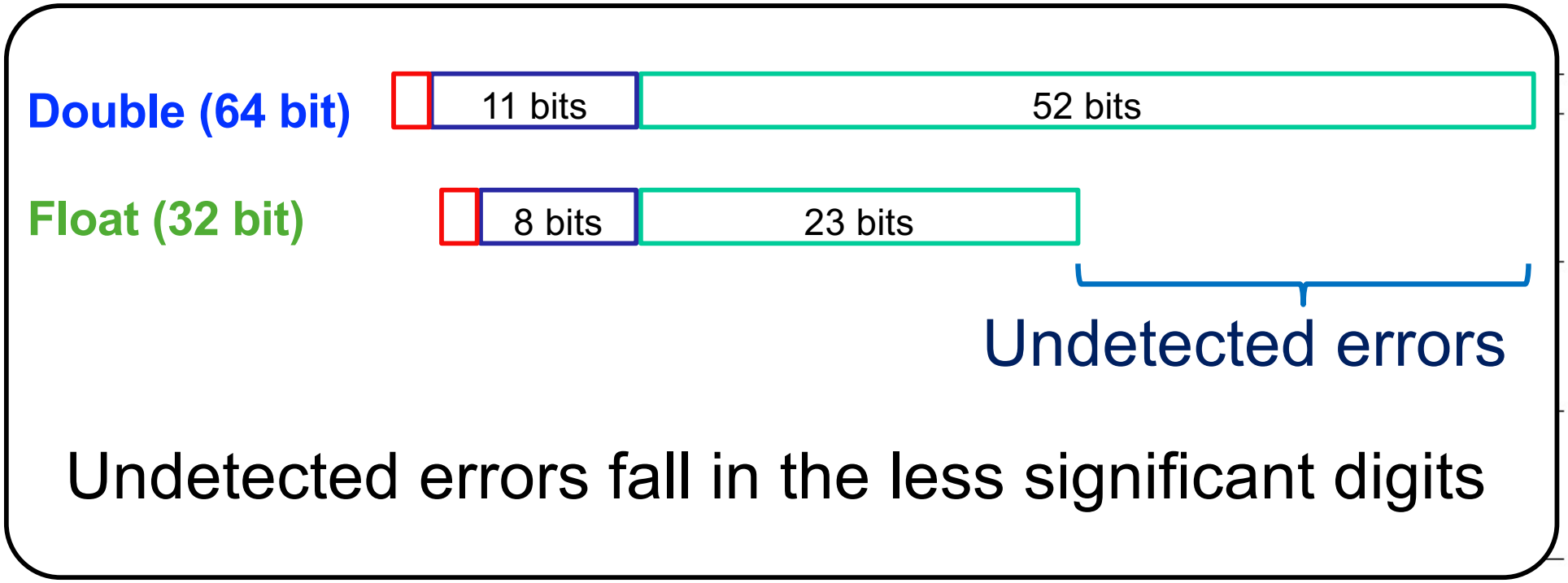


Mixed-Precision Hardening



Mixed-Precision Hardening

Detection goes from 57% to 76%. As expected, lower than traditional DWC (~80-90%)



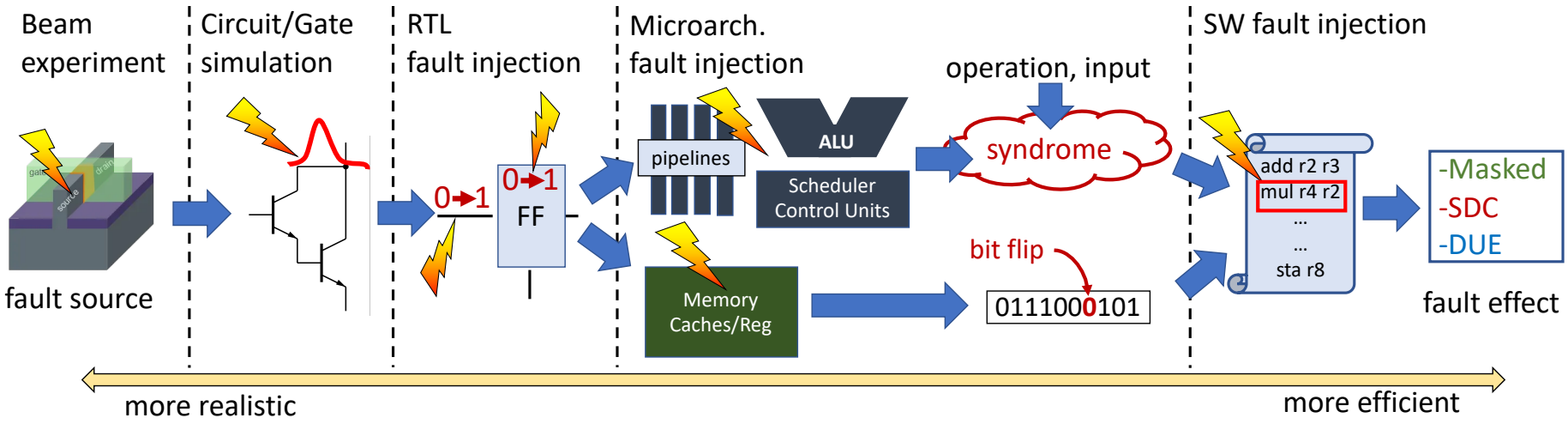
Undetected errors fall in the less significant digits

Outline

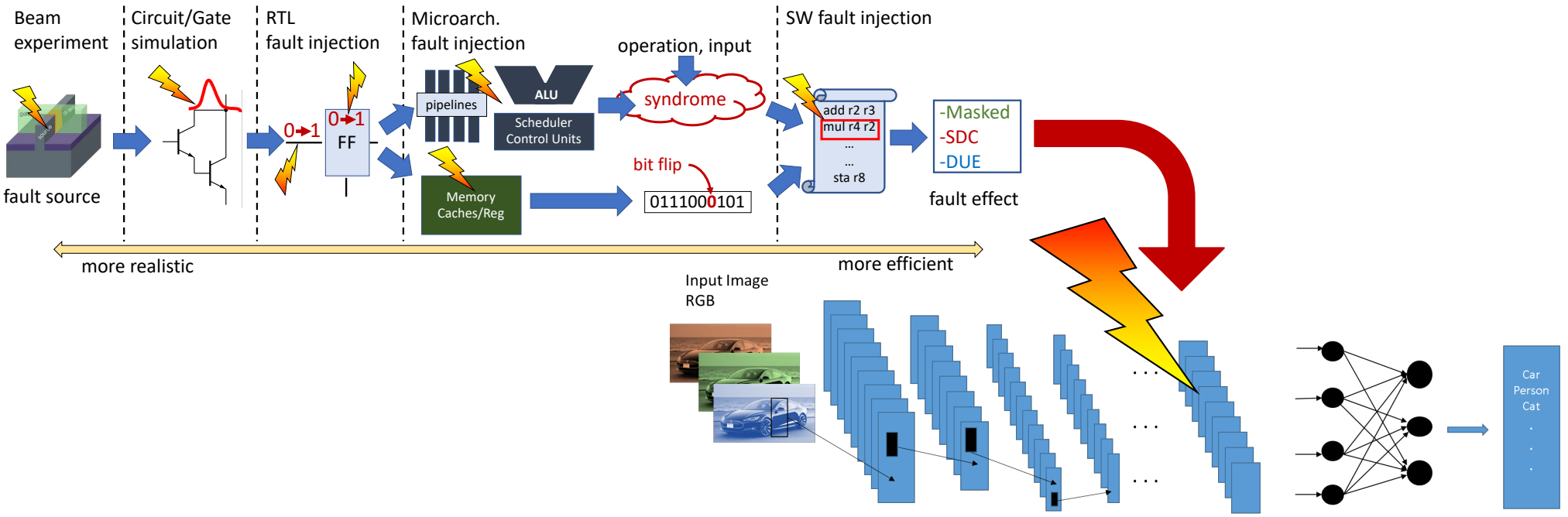
- Neutrons-induced effects in computing devices
- Evaluating neutron-induced errors probabilities
- Cross layer faults propagation in CNNs
- Some (interesting) efficient solutions
- **Conclusions and Future Work**

Can We Rely on Self-Driving Cars?

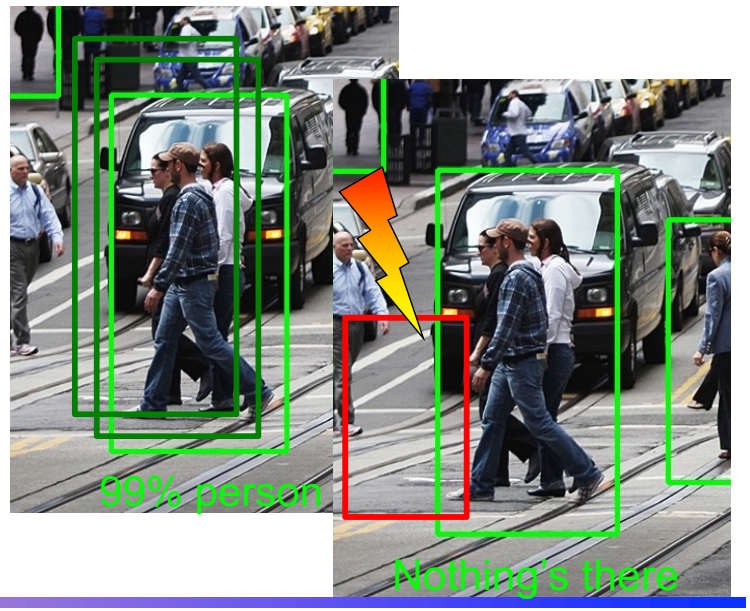
- not all faults reach the software level
- the fault model is not naïve in modern architectures
- the corrupted value(s) depend(s) on several variables



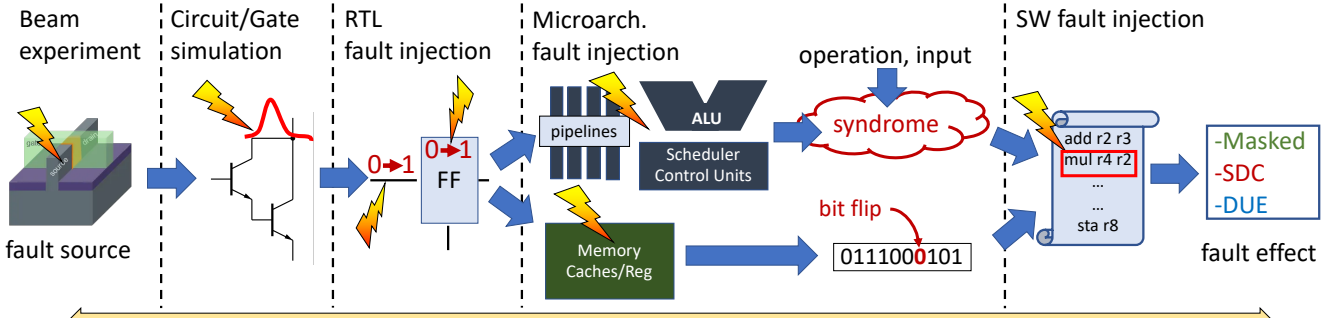
Can We Rely on Self-Driving Cars?

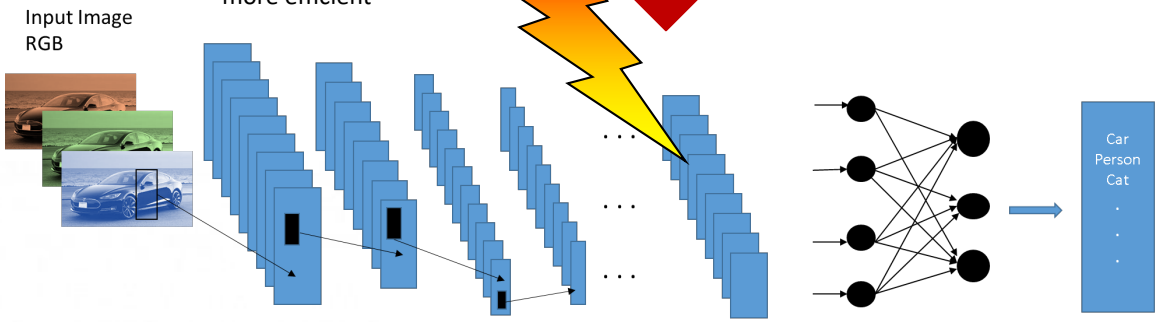


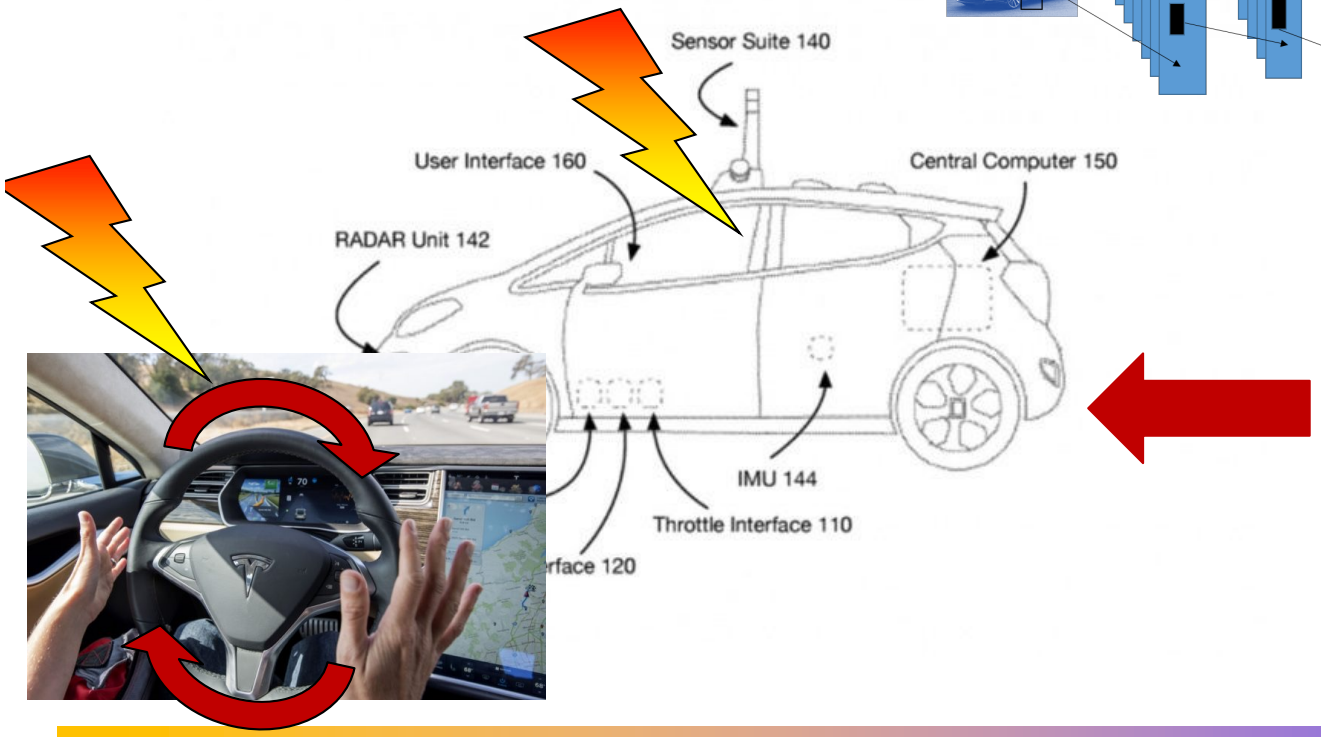
- not all errors are critical for CNNs
- SW/HW solutions can be efficient
- realistic fault model is necessary to design effective hardening

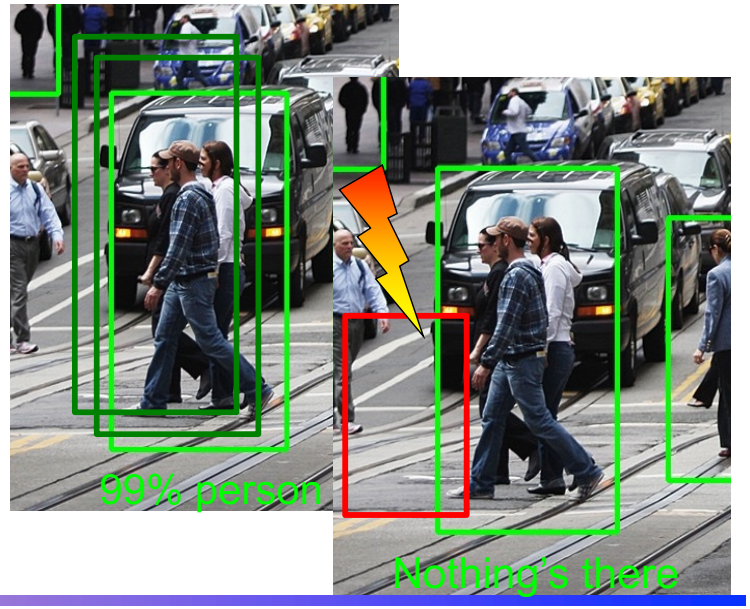


Can We Rely on Self-Driving Cars?

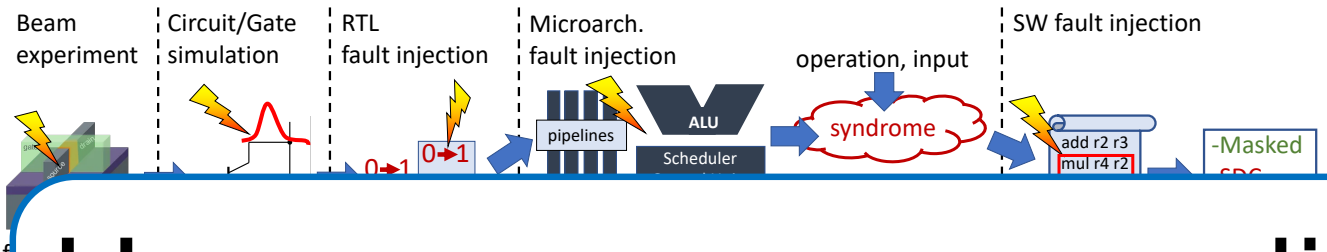




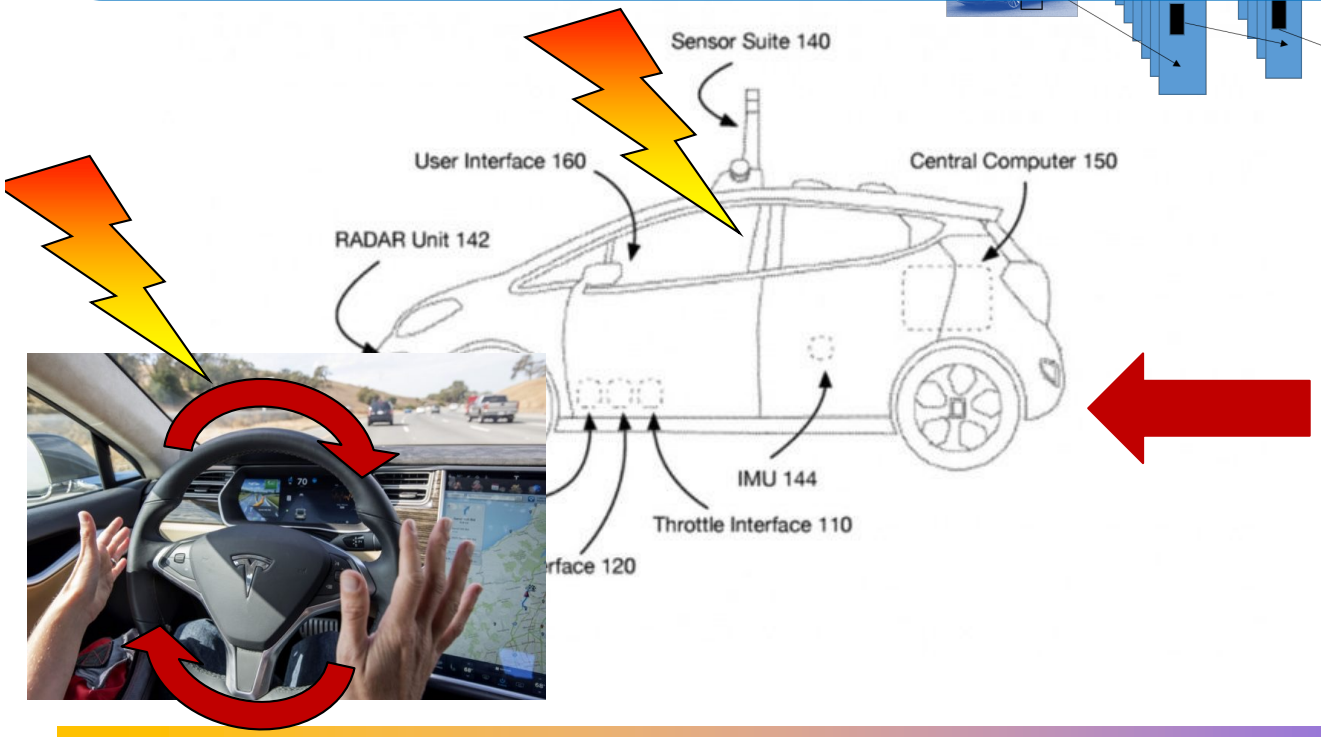


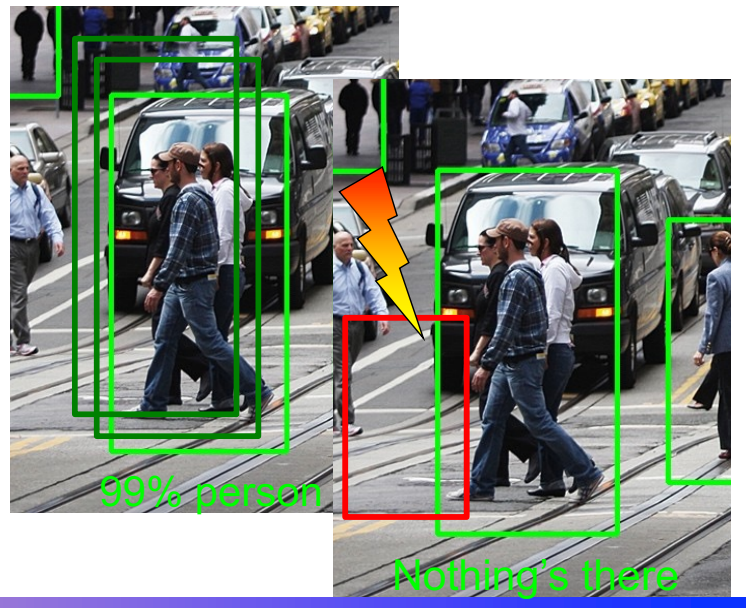


Can We Rely on Self-Driving Cars?



How many errors modify the vehicle behavior?





Conclusions and Future Work

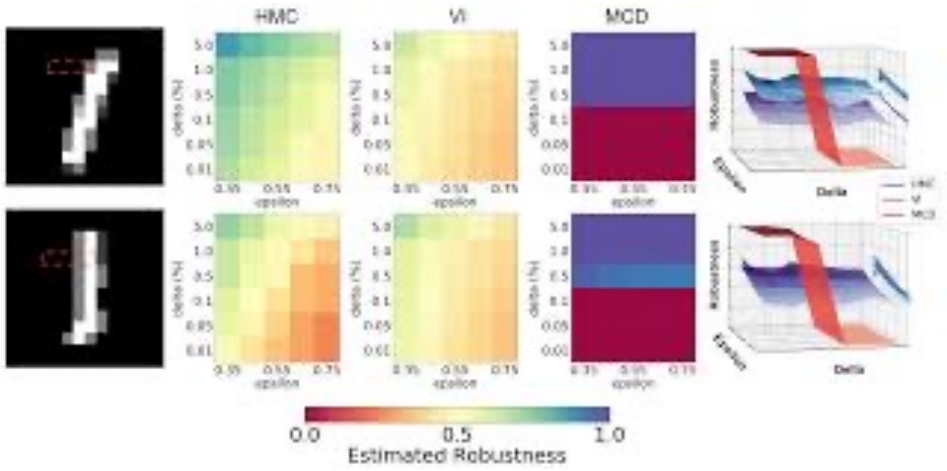
-Reliability is a serious issue for safety-critical applications such as autonomous vehicles

-Self-driving cars will be adopted in large-scale only when sufficiently reliable

-We need to focus on critical errors, critical variables, critical resources to have efficient hardening

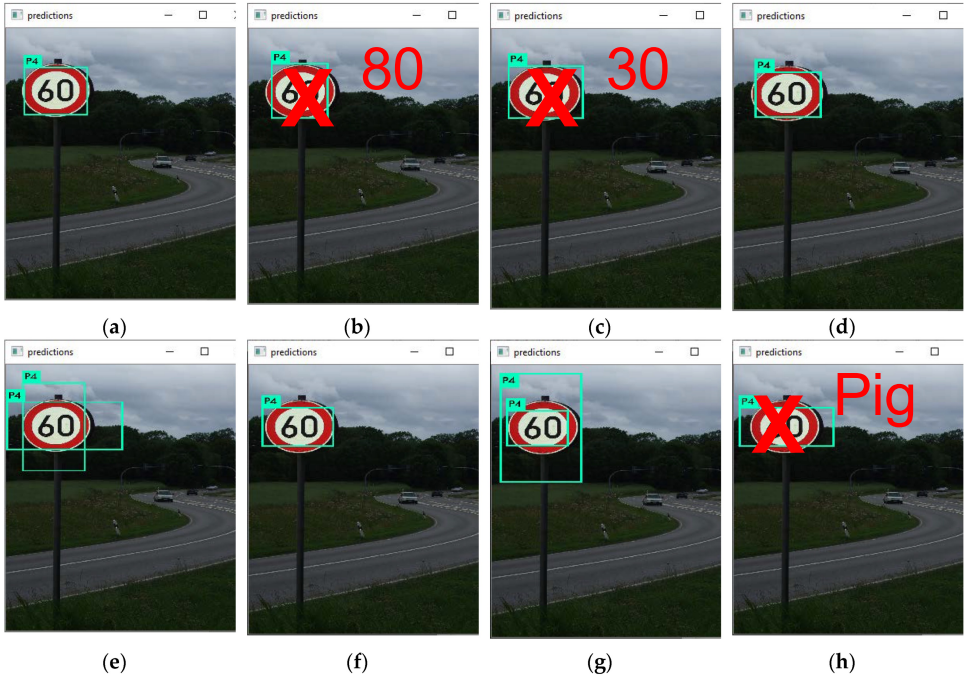
-Future work: reliability-aware training

CNN Robustness and Reliability

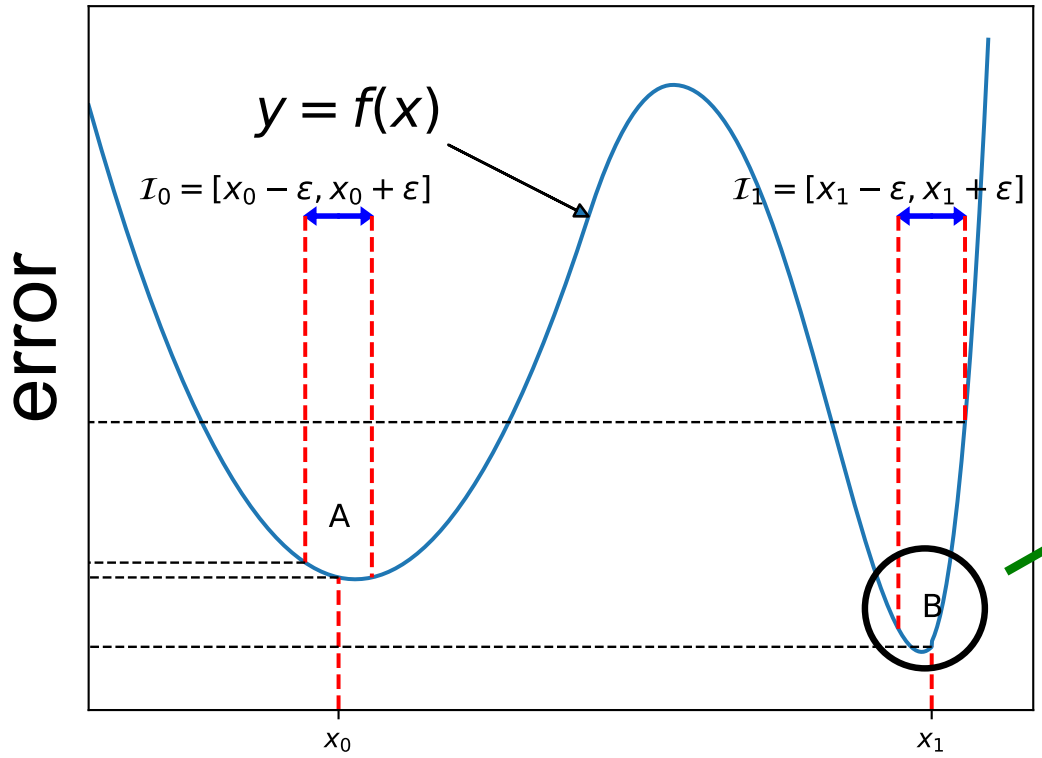


Maintain high accuracy even if the input is “noisy”

Avoid adversarial attacks to “fool” the CNN



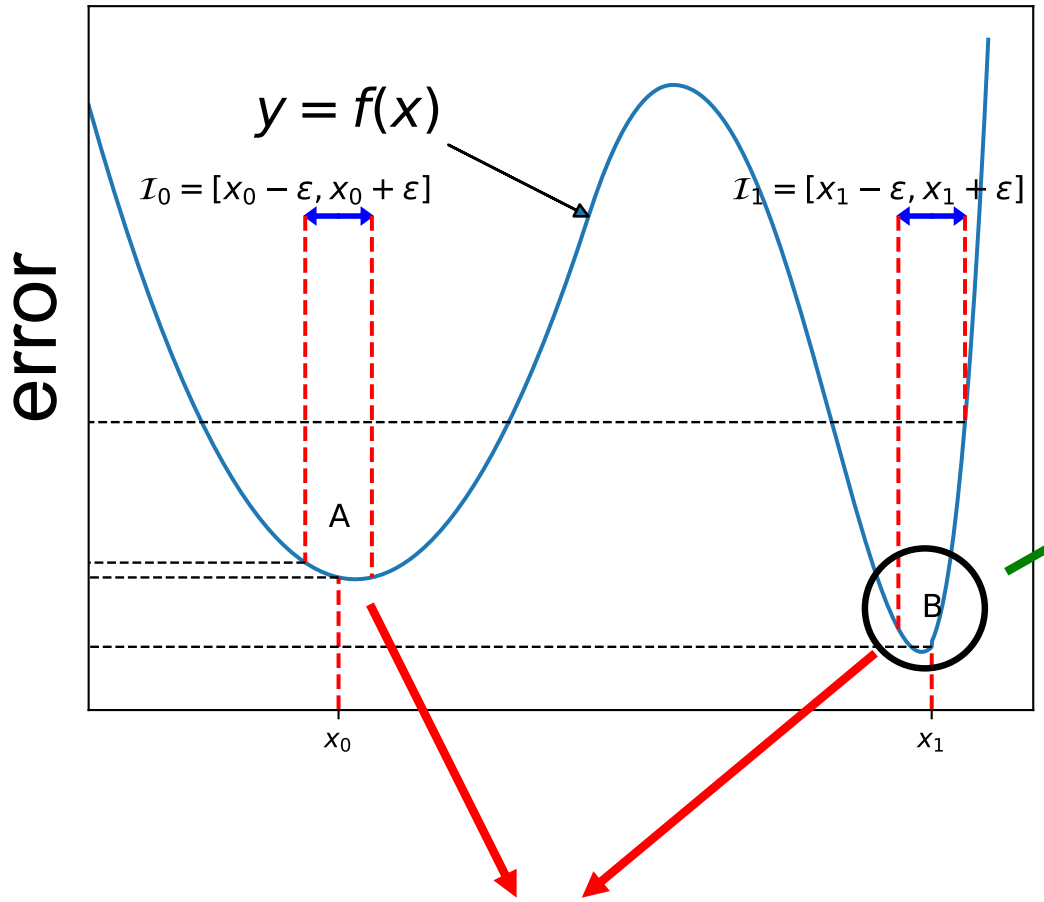
Sharpness-Aware CNN*



This is more accurate
(lower error)

*Sun, et al. 2021
*Foret, et al. 2021

Sharpness-Aware CNN*



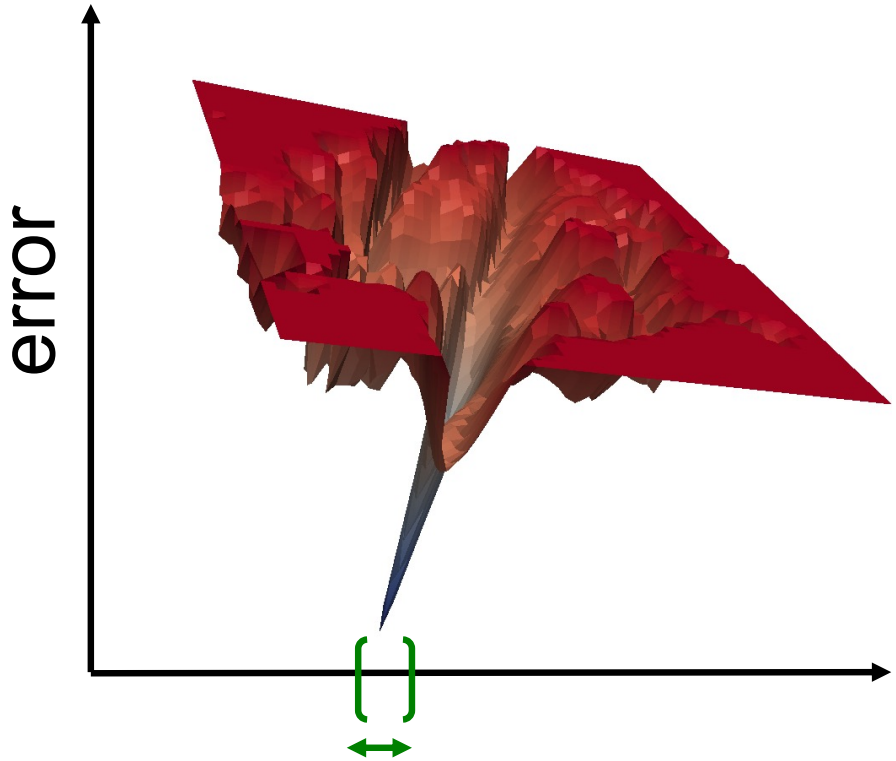
This is more accurate
(lower error)

which is more reliable?

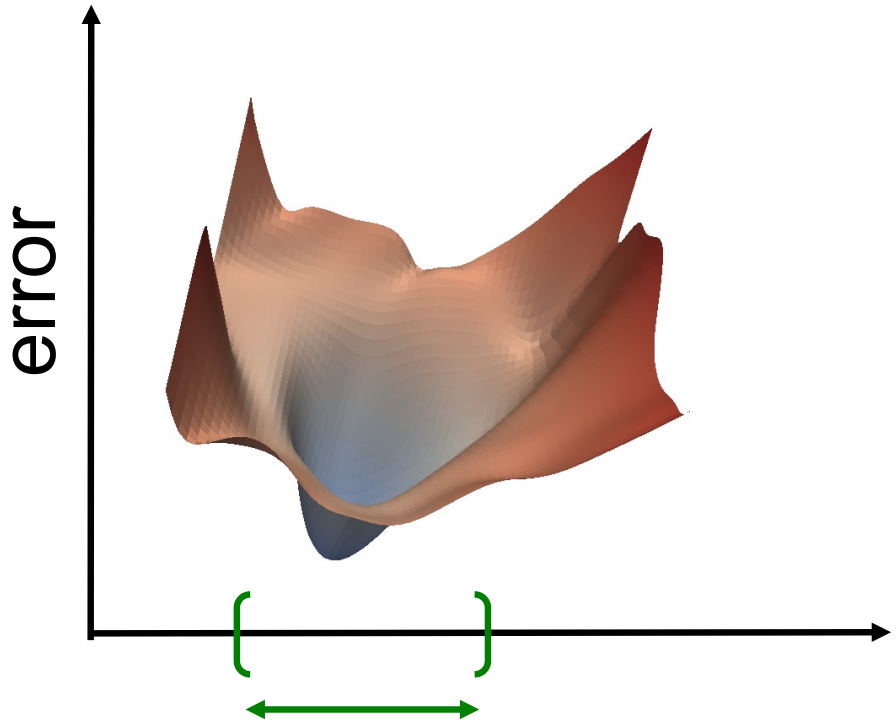
*Sun, et al. 2021
*Foret, et al. 2021

Sharpness-Aware CNN

traditional training



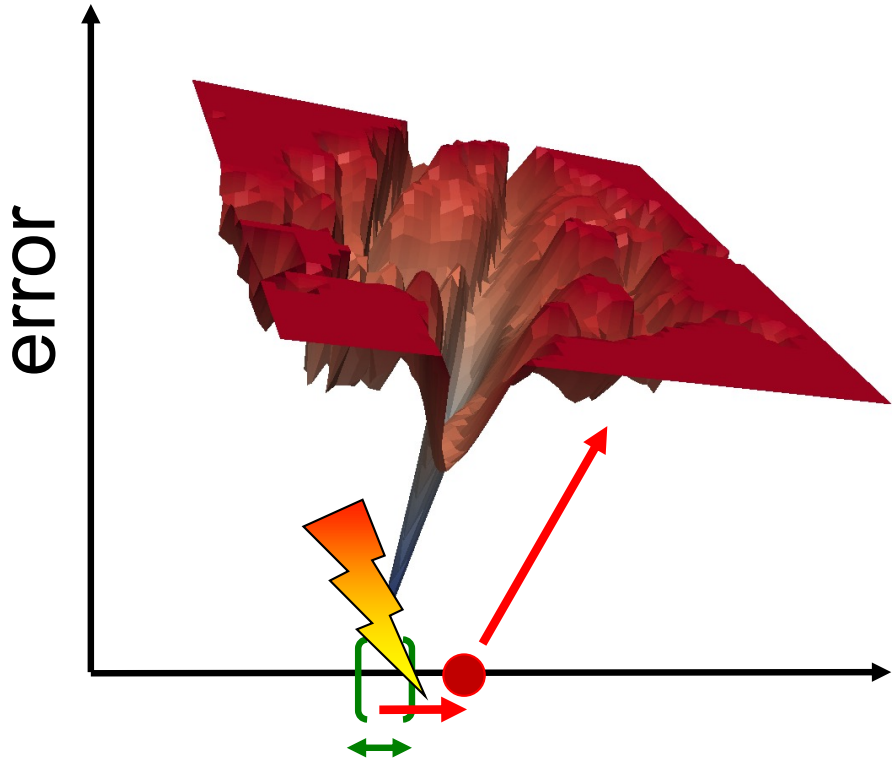
sharpness-aware training



*Sun, et al. 2021
*Foret, et al. 2021

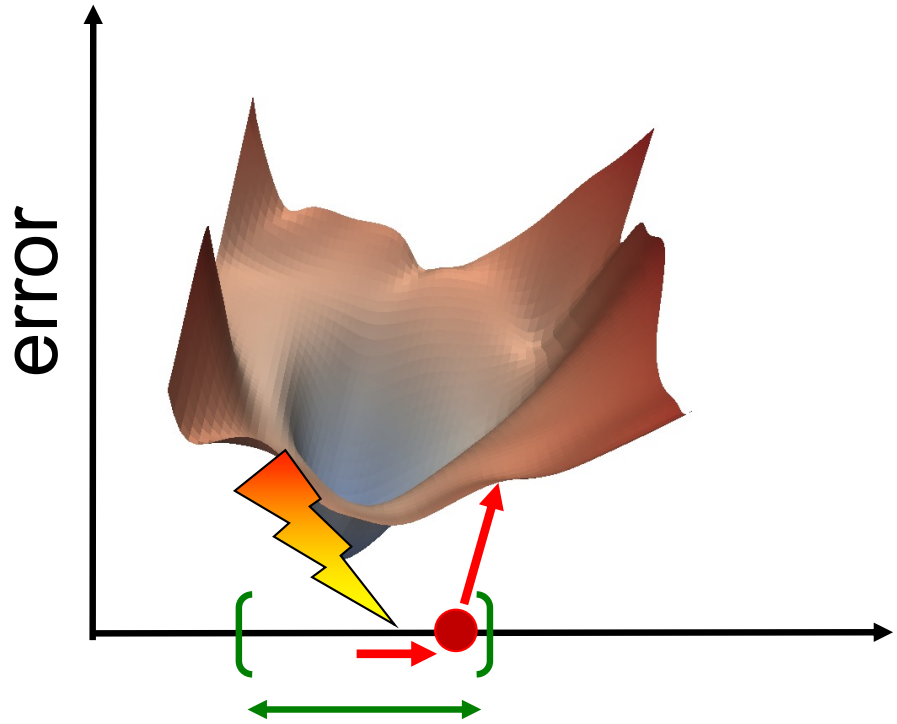
Sharpness-Aware CNN

traditional training



small variations
lead to high error
(mis-detection)

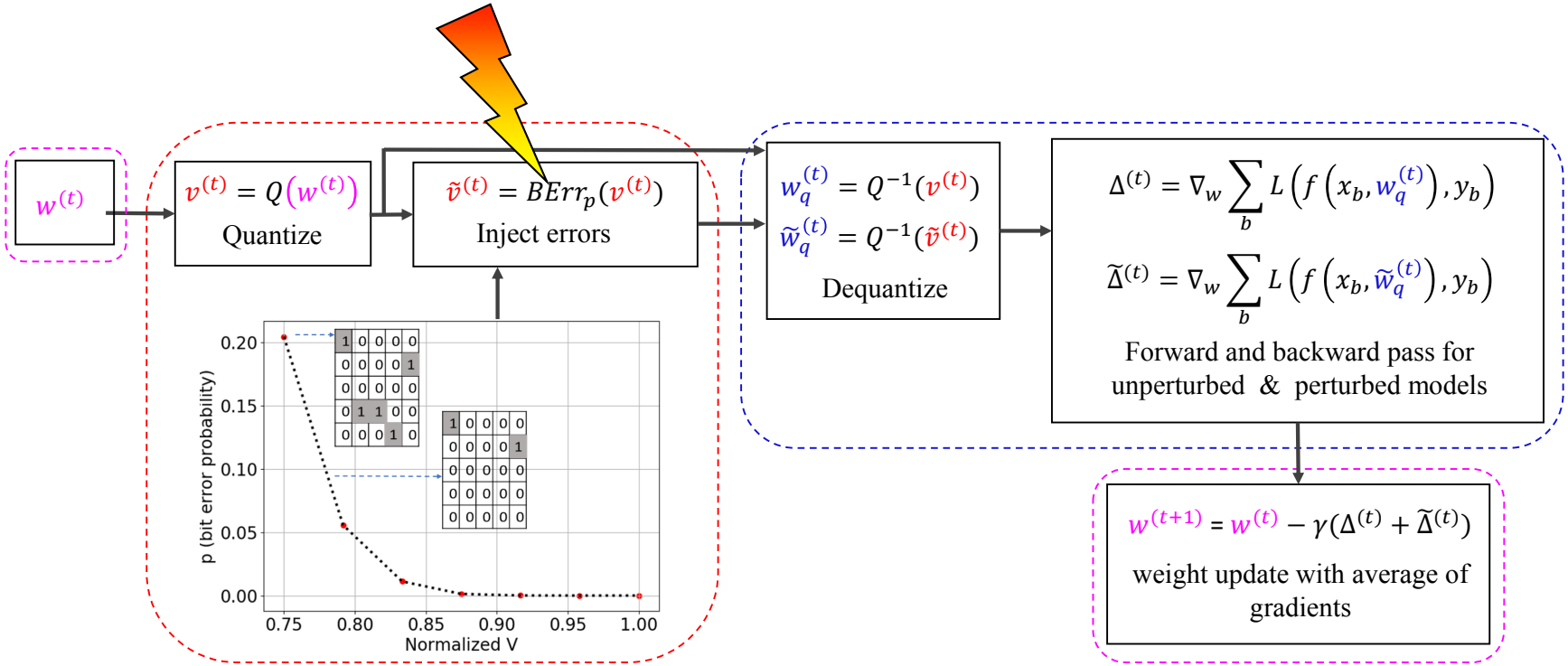
sharpness-aware training



small variations
can still have an
acceptable error

Fault-Injection during Training*

inject errors during training…



…forcing the CNN to still detect objects correctly

*Stutz, et al. 2021

Acknowledgments

Caio Lunardi
Daniel Oliveira
Fernando Santos
Lucas Klein
Pedro Pimenta
Philippe Navaux
Luigi Carro

Heather Quinn
Elizabeth Auden
Thomas Fairbanks
Nathan DeBardeleben
Sean Blanchard
Steve Wender
Gus Sinnis

Chris Frost
Carlo Cazzaniga
Philip King
ISIS User Office

Timothy Tsai
Siva Hari
Michael Sullivan
Steve Keckler

Pete Harrold
Balaji Venu
Reiley Jeyapaul

Matteo Sonza Reorda
Luca Sterpone
DAUIN