arnes
povezujemo znanje

EURO
SLING

MREŽA ZNANJA
Ljubljana, 3.–5. december 2024

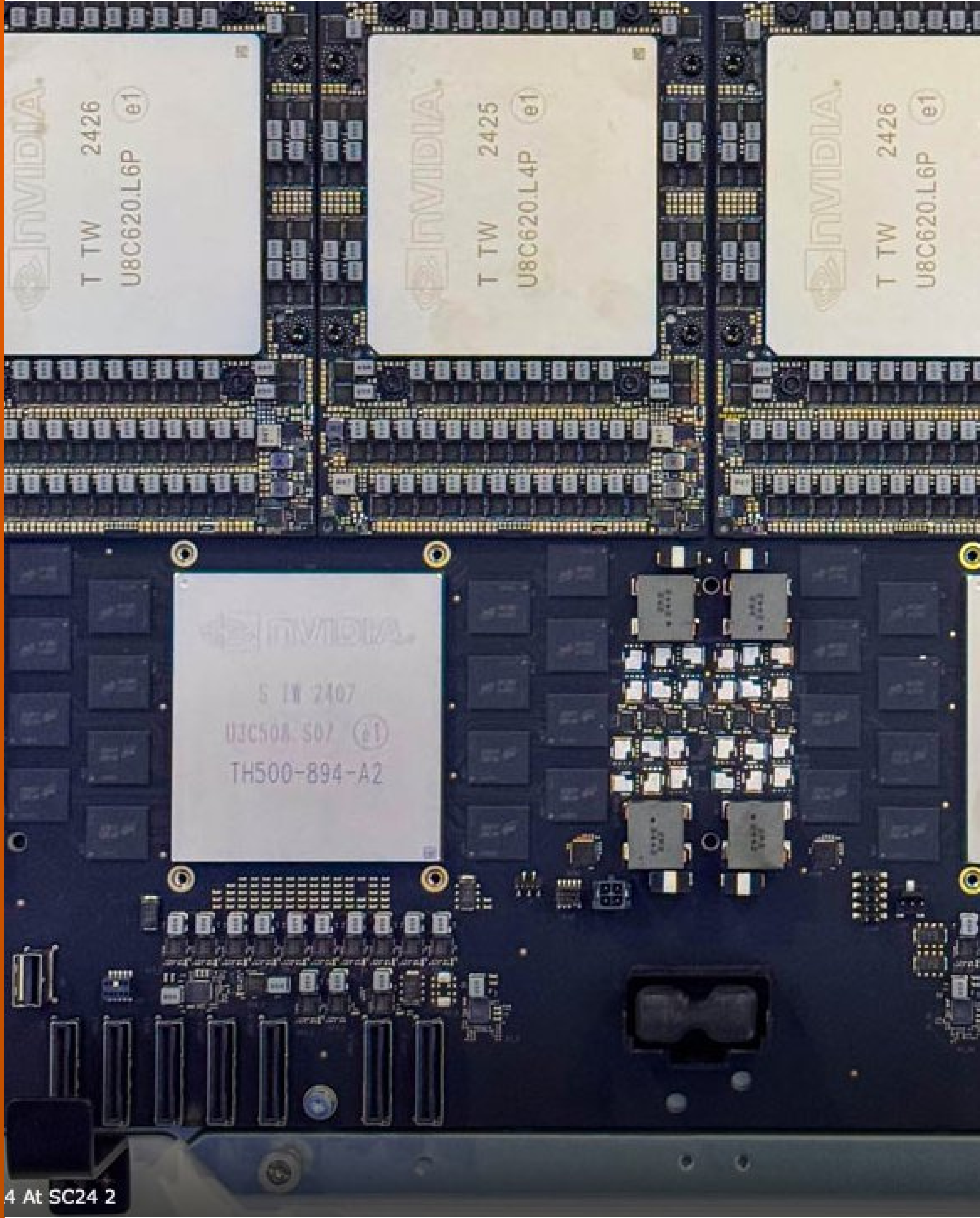
Supercomputers adore deep learning

prof. Uroš Lotrič
University of Ljubljana,
Faculty of Computer nad Information Science

Content

Graphics accelerators
Neural networks
Matrix operations
Conclusion

Graphics accelerators



Technology development

Shared memory systems

Distributed memory systems

Graphics accelerators

https://top500.org

Graphics accelerators: development

Special circuits for 2D and 3D acceleration

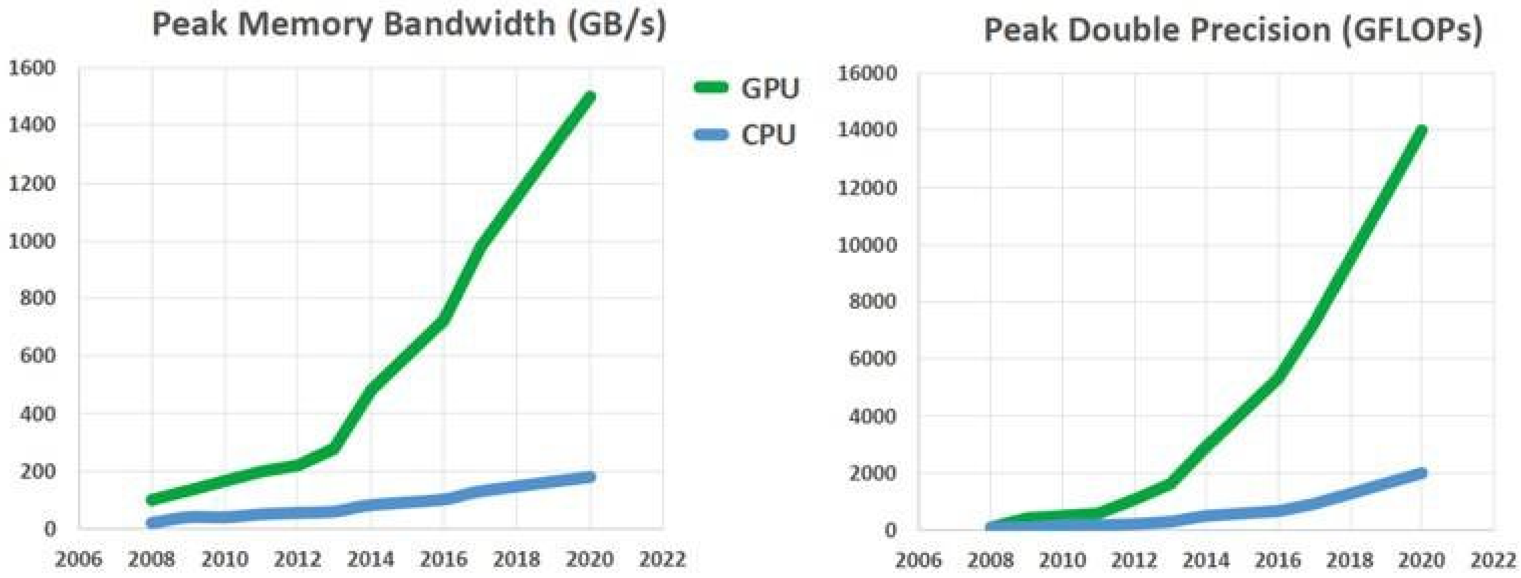
- pixel operations
- high level of parallelization

CUDA, Nvidia, 2006

- shaders that can accelerate 2D or 3D
- general-purpose programming

Integration to supercomputers

Enormous increase in computing power



Natoli: https://www.nextplatform.com/2019/07/10/a-decade-of-accelerated-computing-augurs-well-for-gpus/

Graphics accelerators: CPU architecture

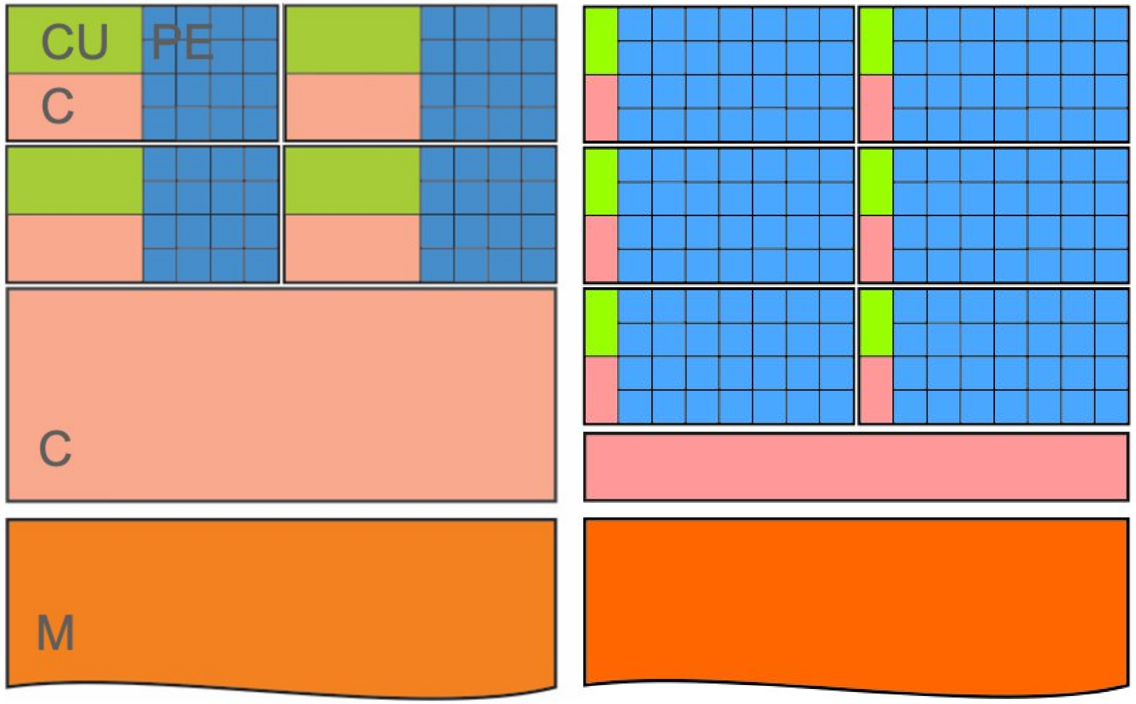
General purpose system

Serial code & parallel hardware

Complex control unit

Large cache

OS schedules threads



Graphics accelerators:
Graphics accelerators architecture

Focus on parallel execution

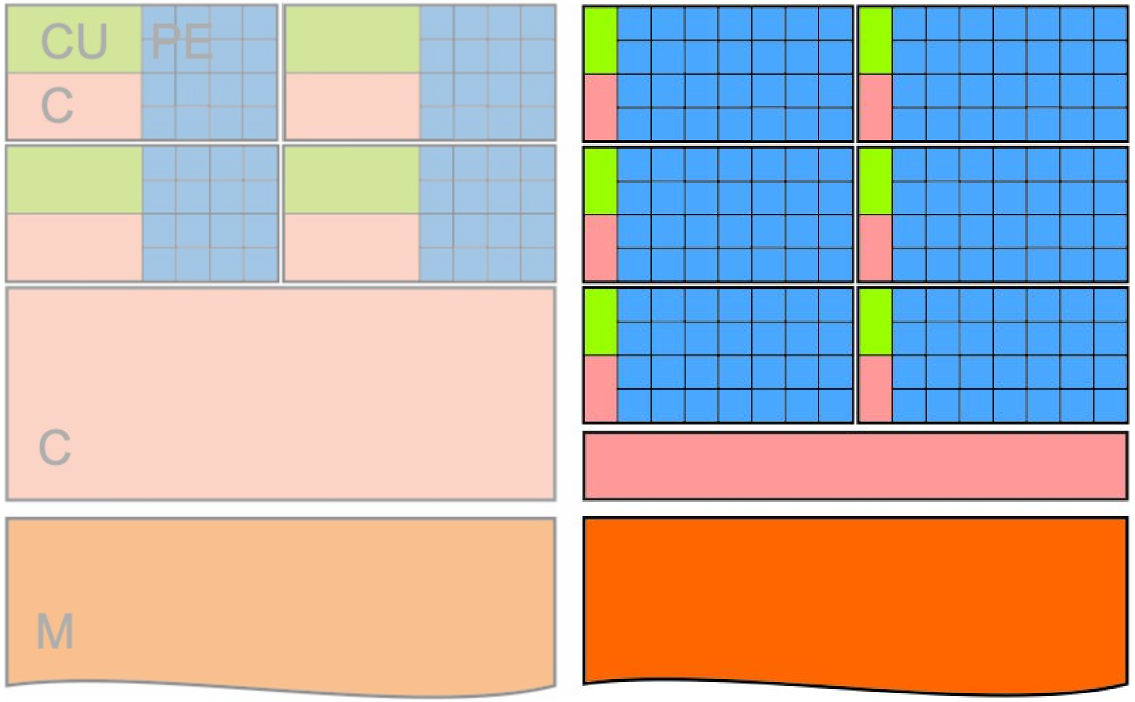
More silicon for computational units

- slim control units, less cache
- divergences in code slow down the execution
- (single instruction multiple data)

A massive number of parallel threads

- hiding memory access latency

Hardware dynamically schedules threads



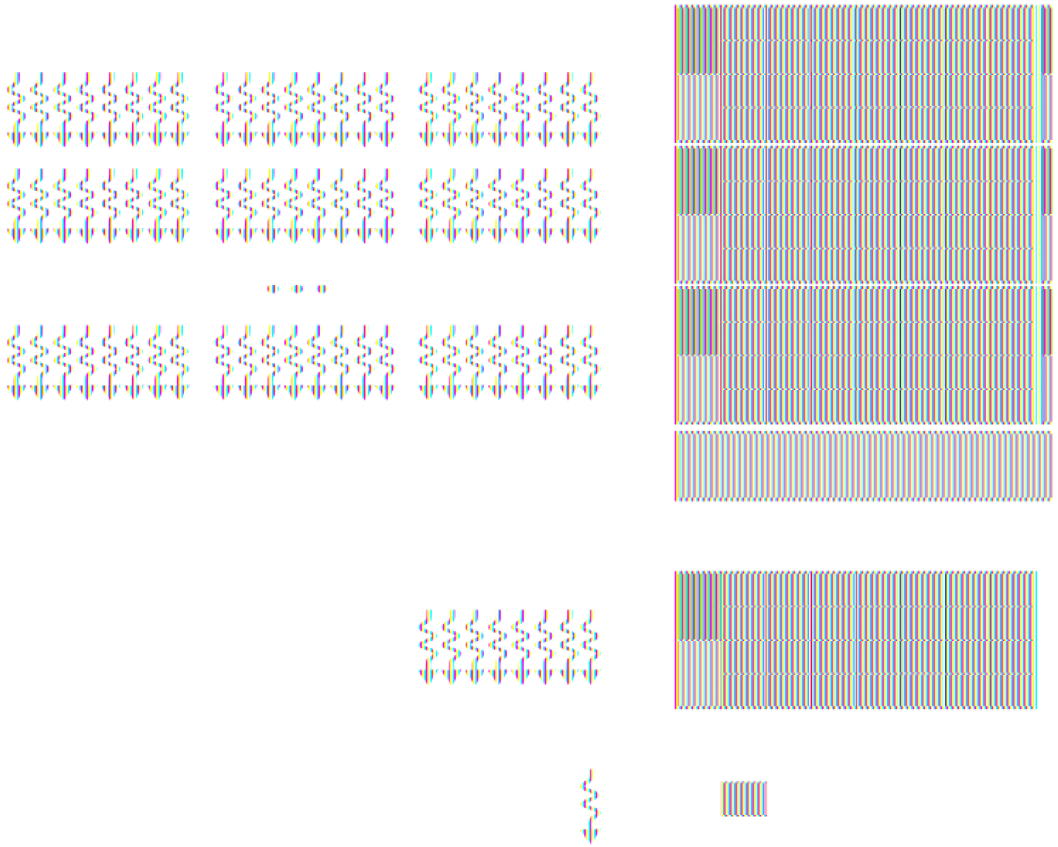
Graphics accelerators: Hieararchy

processors: compute units and processing elements

memory: global, shared, registers

threads: grid, block, warp, thread

It is not simple to adapt existing CPU code for graphics accelerators



Graphics accelerators: Perfect tasks

Execution of the same code on different data

Data is divided among a vast number of threads

Straightforward code without divergences

A lot of computation with little data transfers

Neural networks

Neural networks: Intro

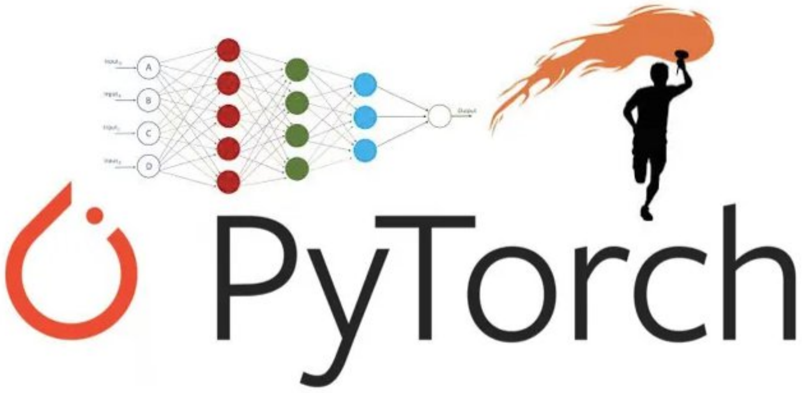
General-purpose mathematical models

A lot of free parameters

Different learning algorithms

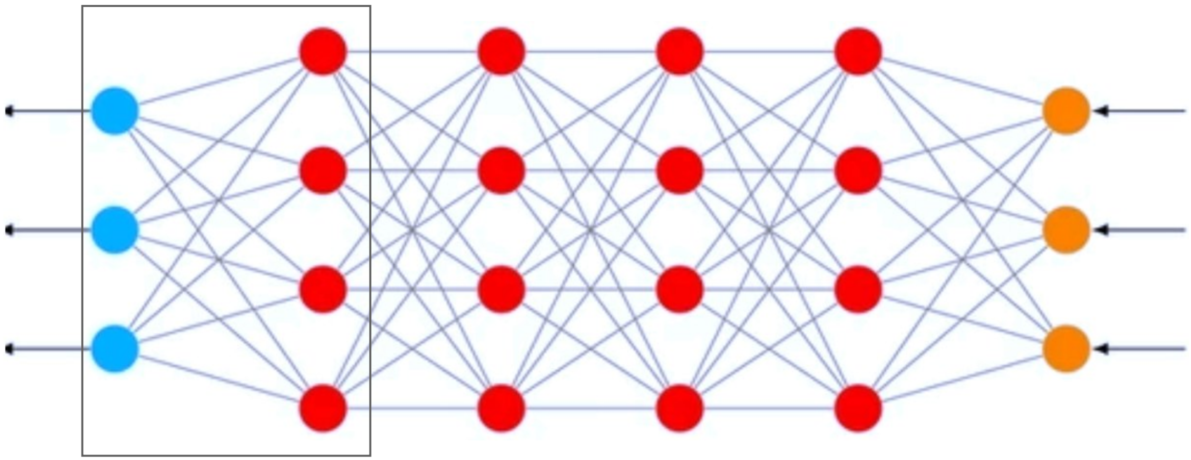
Libraries

- building blocks
- modelling, learning, inference



vir: https://medium.com/analytics-vidhya/not-torturing-in-learning-pytorch-b2f7f169923a

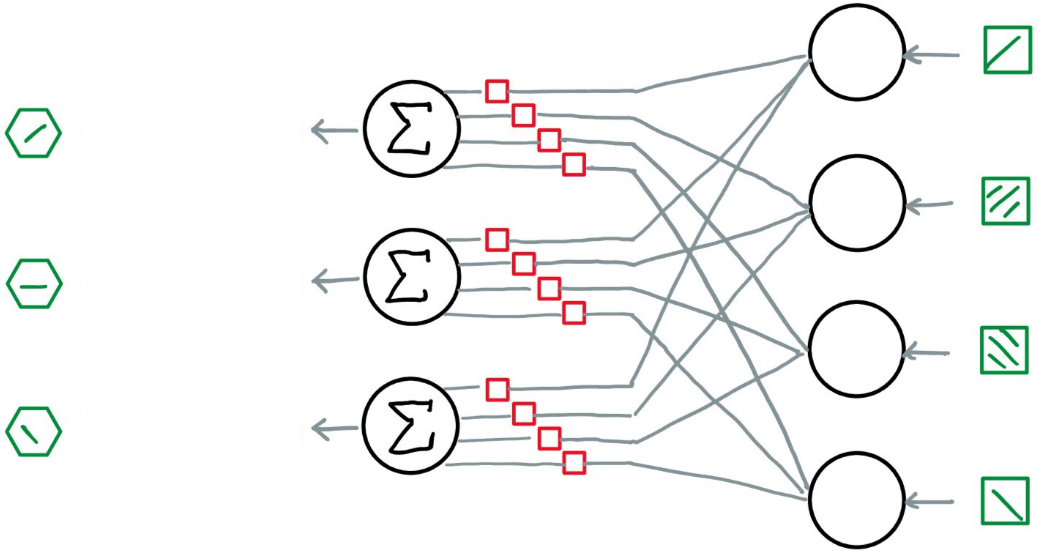
Neural networks: Multilayered perceptron



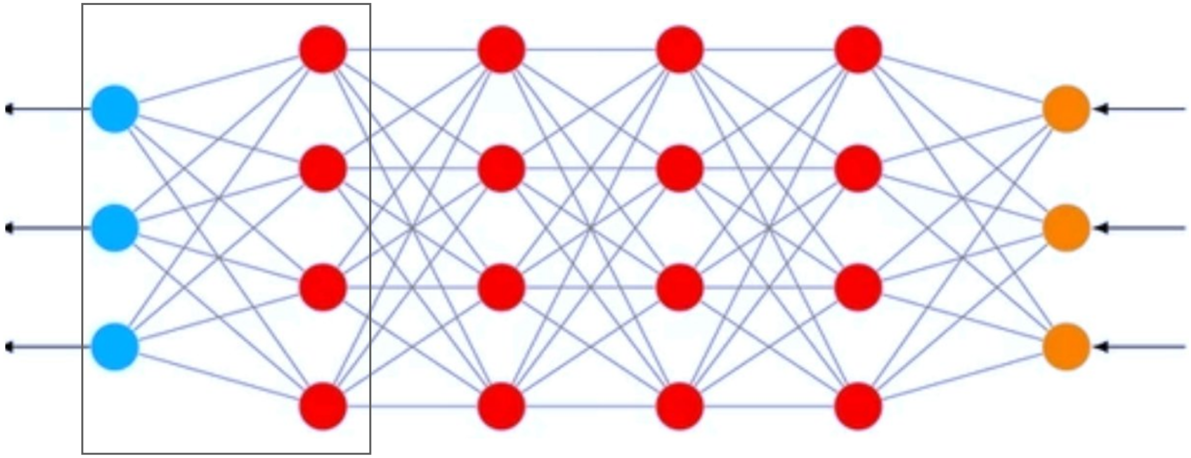
Fully connected model

Inference at one layer

- input values are multiplied by synaptic weights
- a sum of values on all synapses
- activation function gives output



Neural networks: Multilayered perceptron



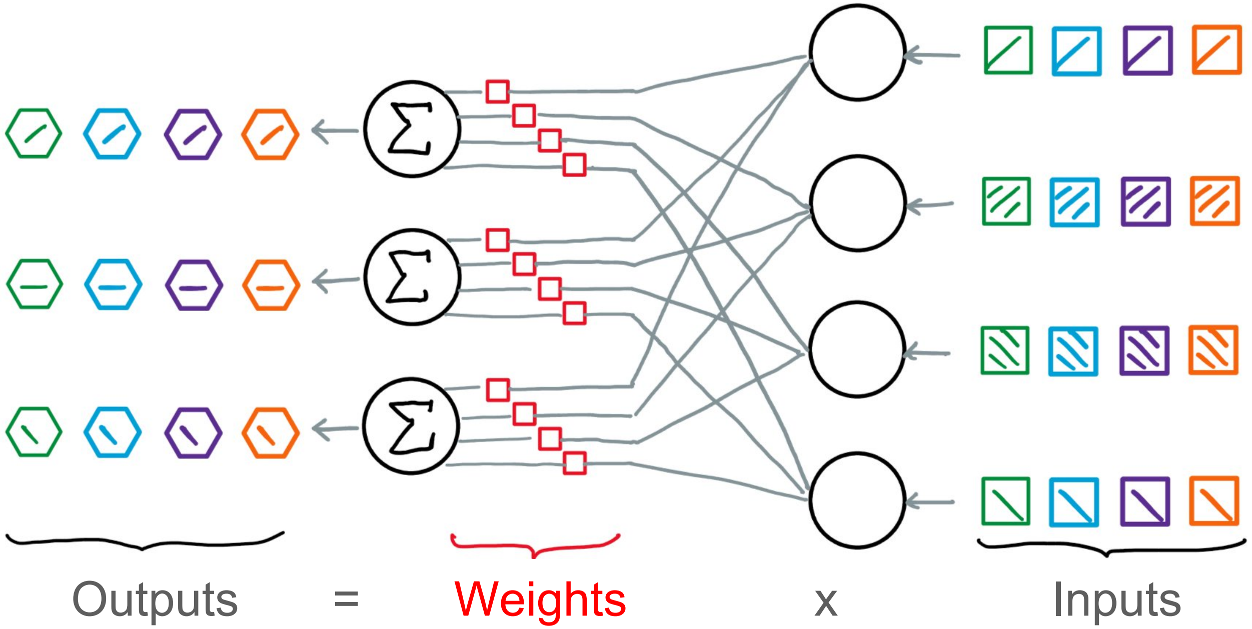
Fully connected model

Inference at one layer

- input values are multiplied by synaptic weights
- a sum of values on all synapses
- activation function gives output

Mathematical description:

- matrix multiplication
- the product of the input vector (matrix) and weight matrix gives the output vector (matrix)

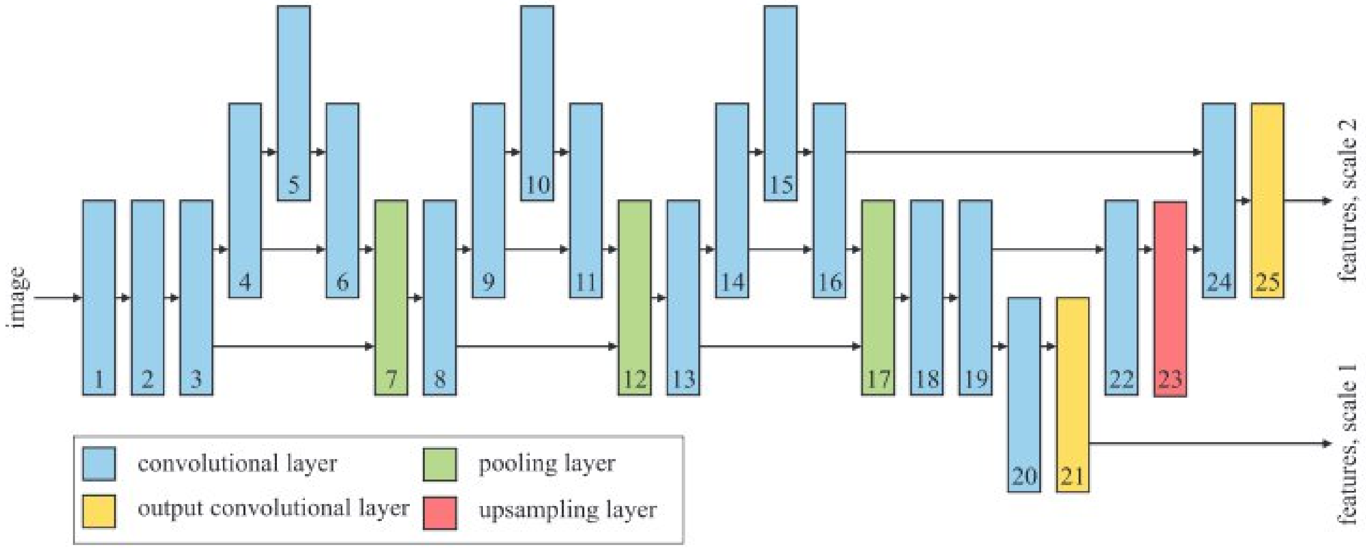


Neural networks: YOLO-v4 tiny backbone

Image processing

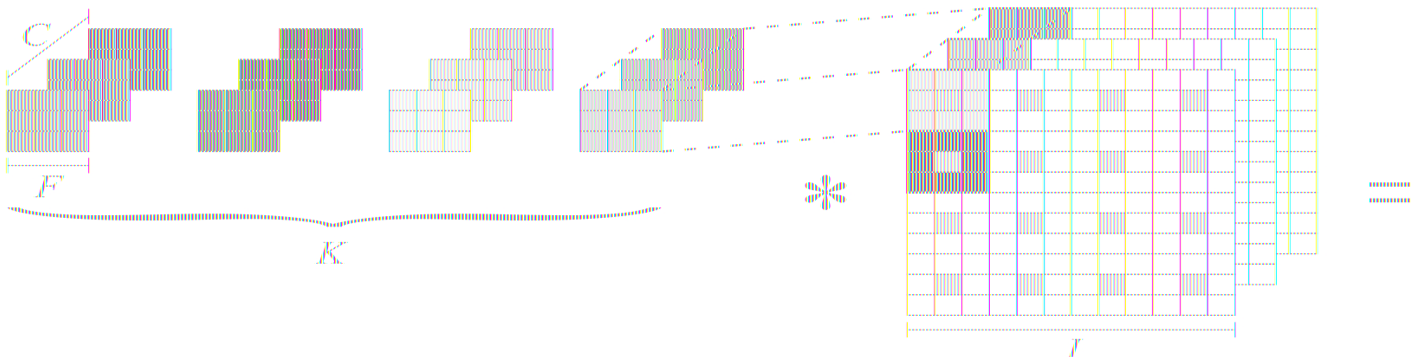
Many convolutional layers

Inference



Pilipović *et. al.* An approximate GEMM unit for energy-efficient object detection, Sensors 21 (12), 4195, 2021

Neural networks:
YOLO-v4 tiny backbone



Convolution of filters and image

Tensors

Pilipović *et. al*. An approximate GEMM unit for energy-efficient object detection, Sensors 21 (12), 4195, 2021

Neural networks:
YOLO-v4 tiny backbone

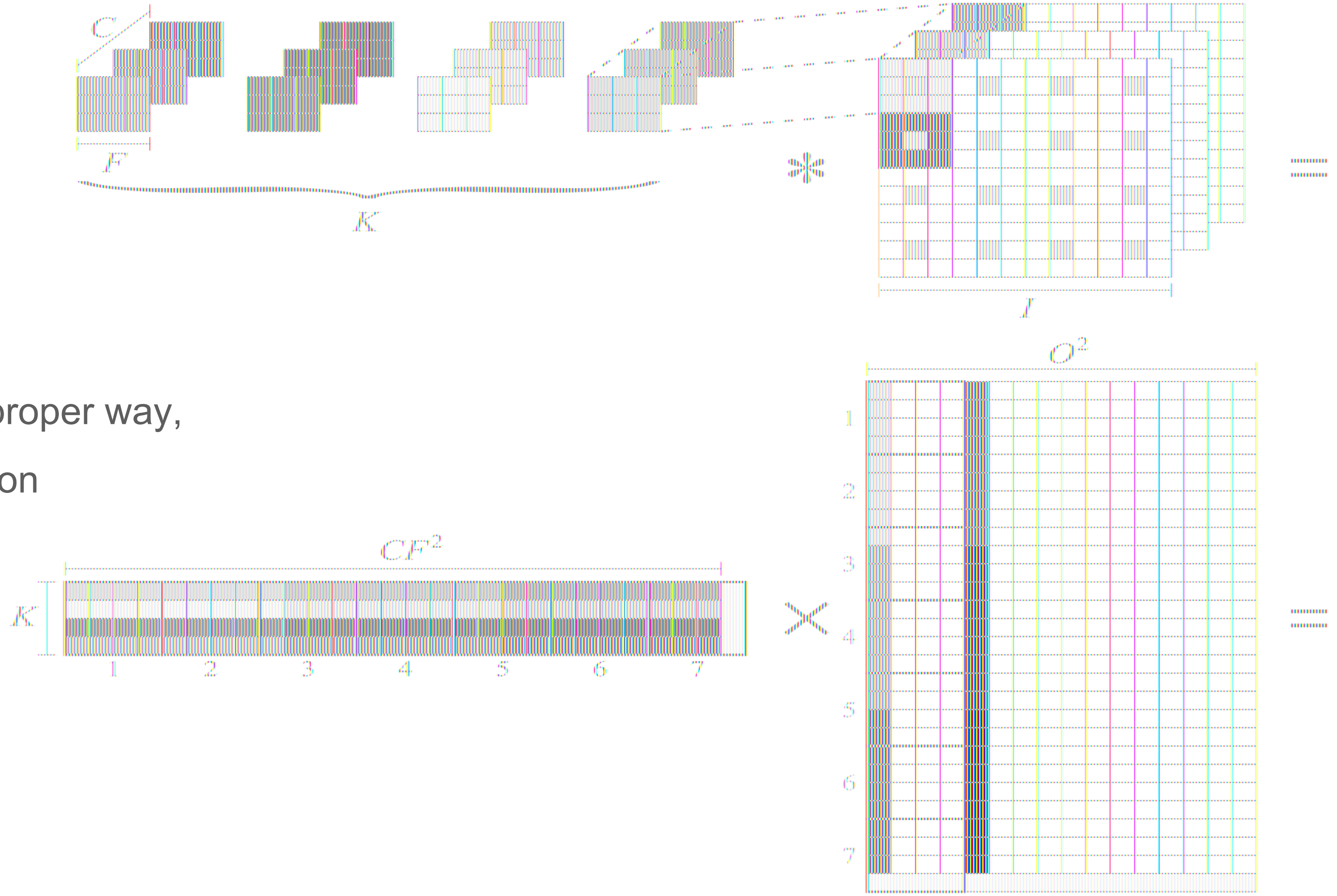
Convolution of filters and image

Tensors

Matrix multiplication

By unfolding filter and image data in a proper way,

convolution becomes matrix multiplication



Pilipović *et. al*. An approximate GEMM unit for energy-efficient object detection, Sensors 21 (12), 4195, 2021

Neural networks:
YOLO-v4 tiny backbone

Training

presenting the neural network with a set of pairs

(inputs, correct outputs)

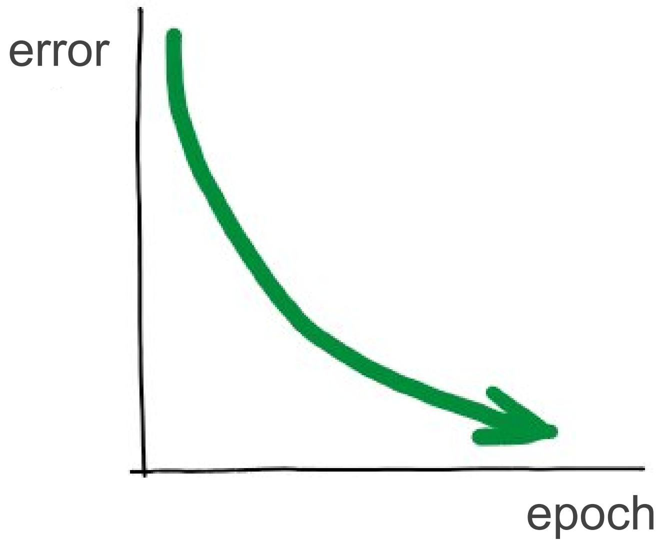
inference + weight adaptation

model error decreases

Large language models use similar ideas

For fast computation, we must try keeping model parameters

(and data) in graphics accelerators' memory



Matrix operations

Matrix multiplication: FMA operation

Fused multiply and add

d = c + a x b

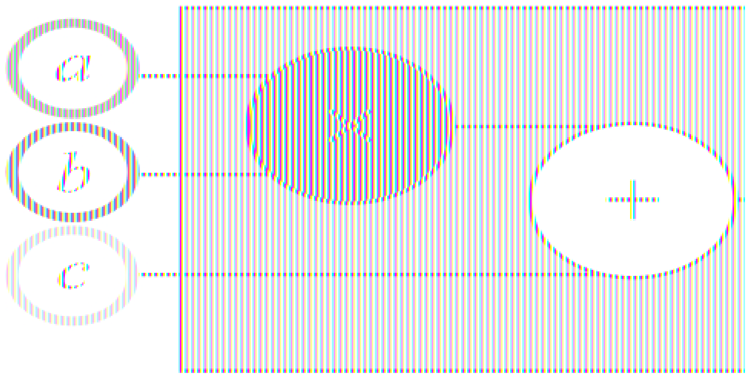
multiplier, adder, rounding

Addition

an accumulator to store intermediate result

output from the accelerator goes to the adder input

one addition in each clock cycle



Matrix multiplication:
FMA and matrices

C = **A** x **B**

To get one element in C

walking a row of A and column of B

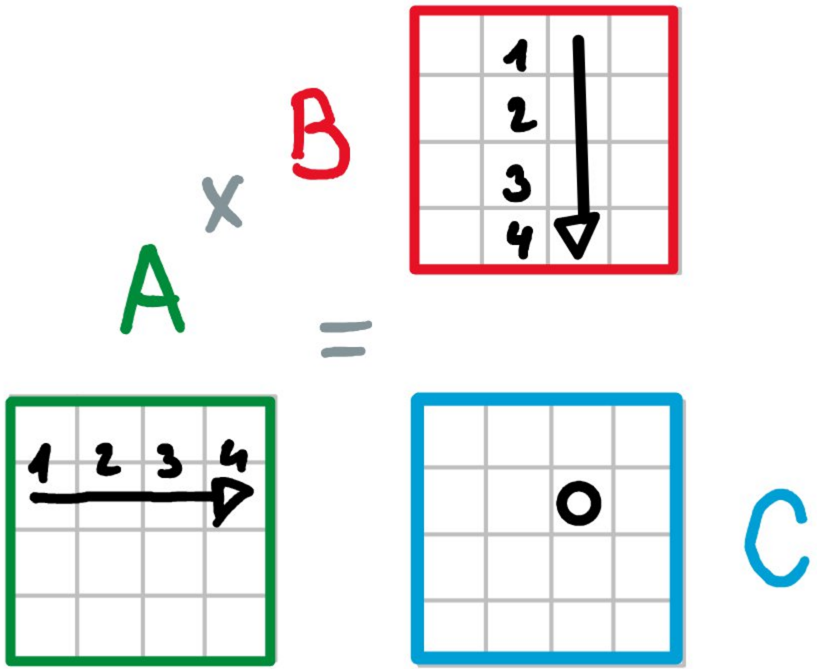
in each step:

multiplication of elements

addition of the product to the current sum

Can do all elements of C in parallel

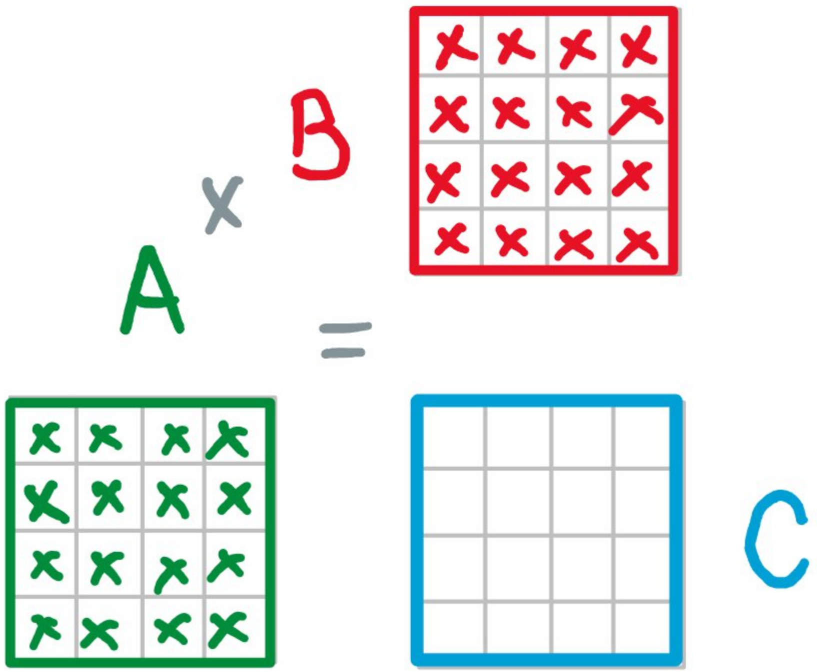
The larger is C, the more graphics accelerators excel



Matrix multiplication:
FMA and matrices

Getting data

- naïve approach: each thread reads data from memory
- optimized approach:
  - caching
  - first, threads read all data
  - second, they compute their elements in C
  - Each element is transferred only once



Matrix multiplication: GEMM unit

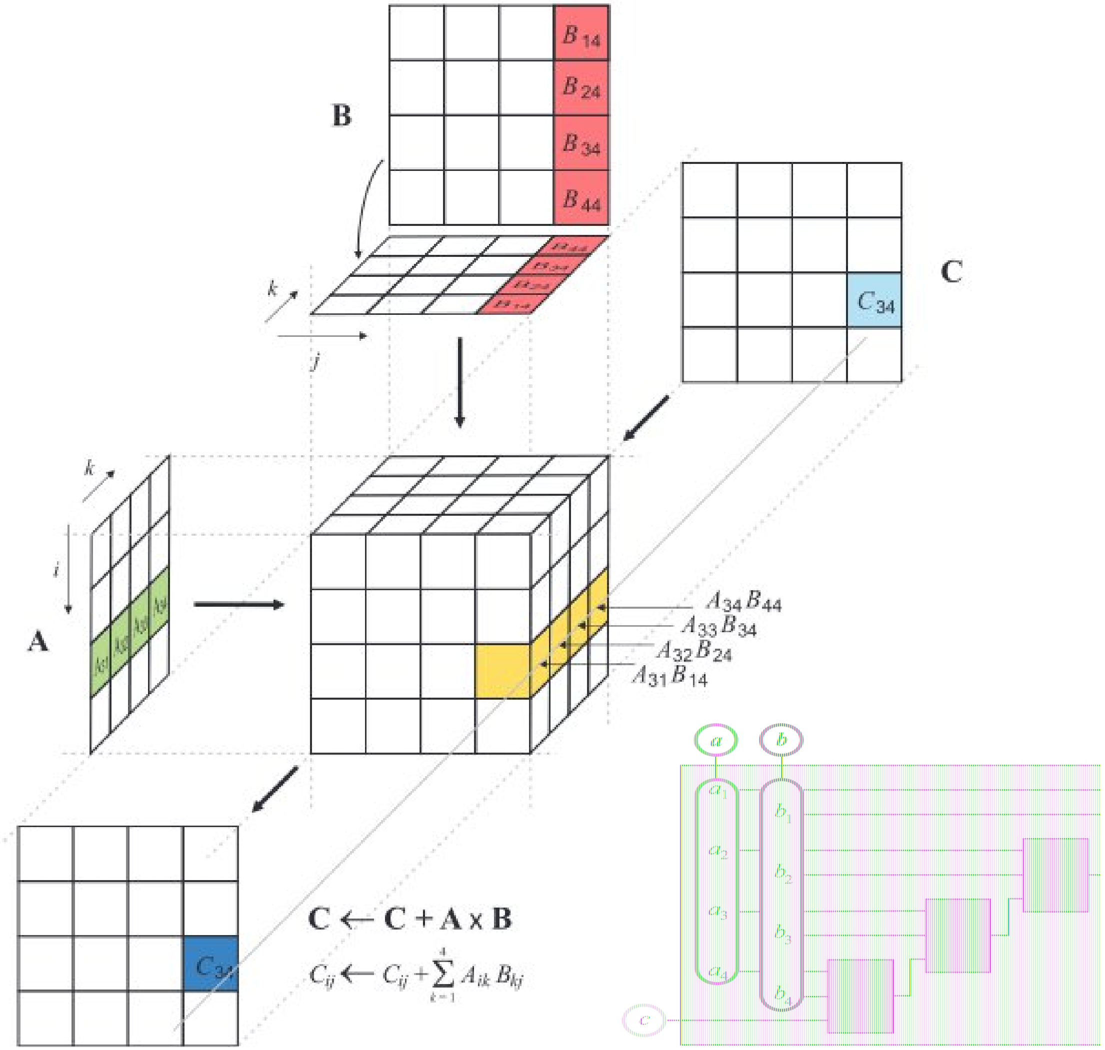
D = C + A x B

We can do 4 x 4 matrix multiplication in one clock cycle

- multipliers and adders are decision circuits
- we can combine them into a GEMM unit
- a circuit with 64 multipliers

Commercial names

- TPU, tensor core, neural processing unit,
- neural engine, matrix core



Pilipović *et. al*. An approximate GEMM unit for energy-efficient object detection, Sensors 21 (12), 4195, 2021

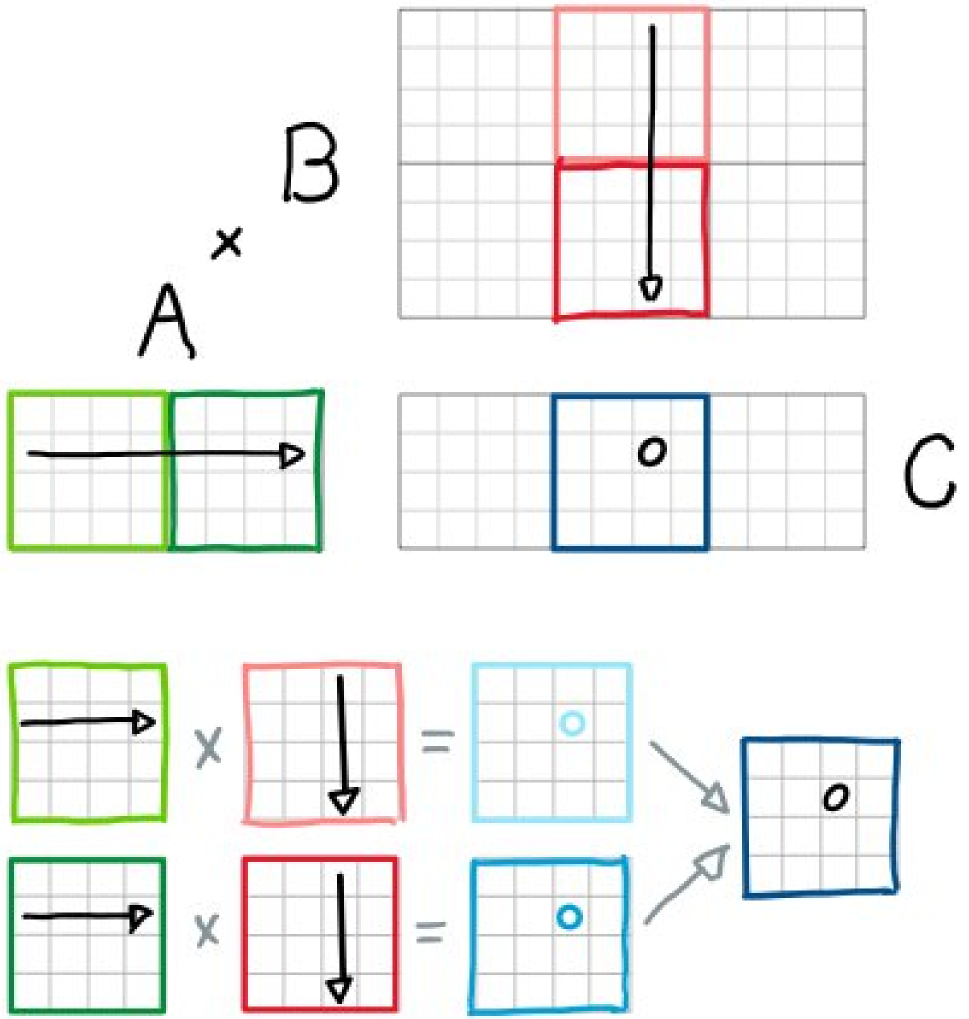
Matrix multiplication: Laying tiles

We divide each matrix into submatrices of size 4 x 4

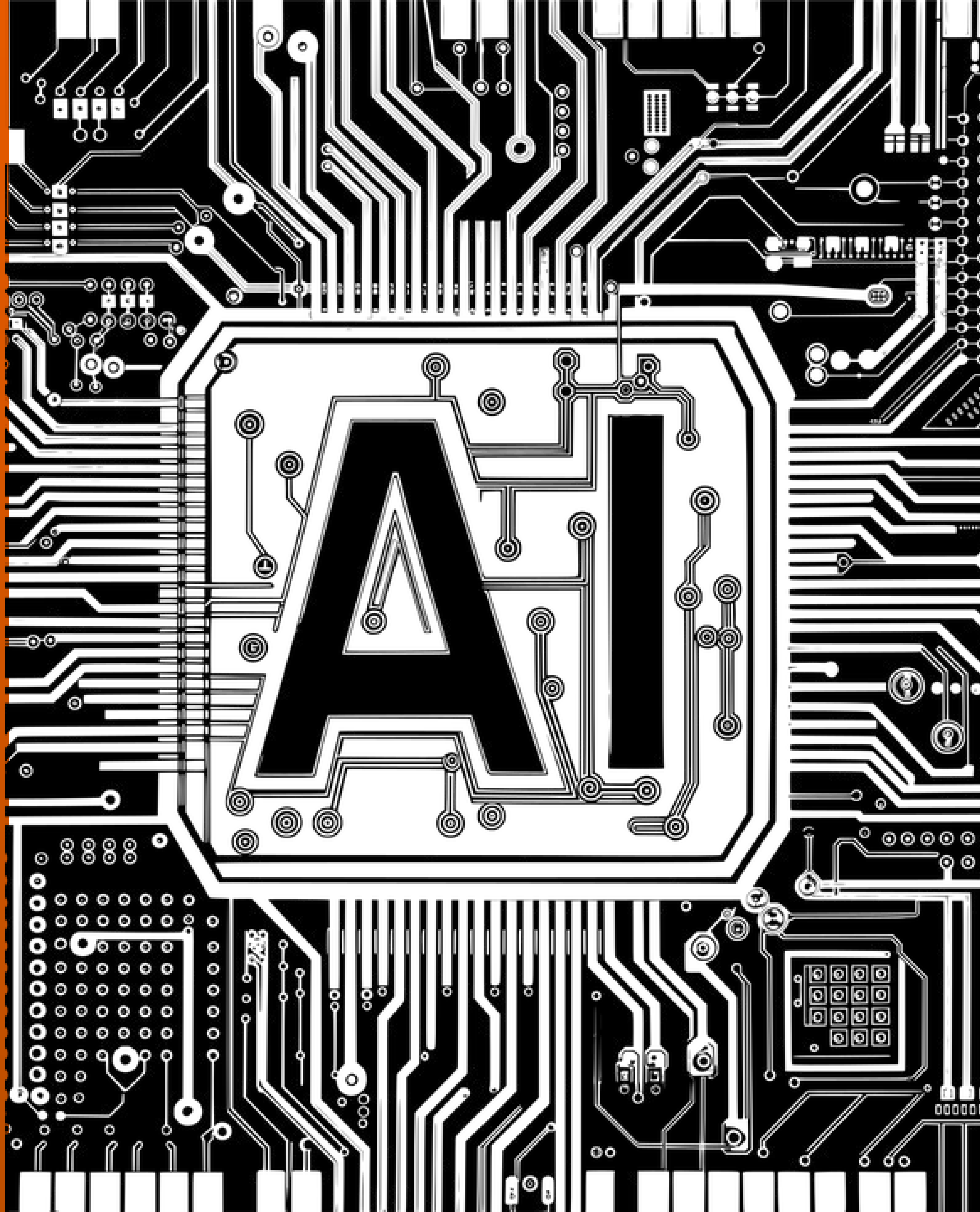
The size of submatrices corresponds to the GEMM unit size

Multiplication

- load a submatrix of A and a submatrix of B
- multiply and add to the corresponding submatrix of C



Conclusion

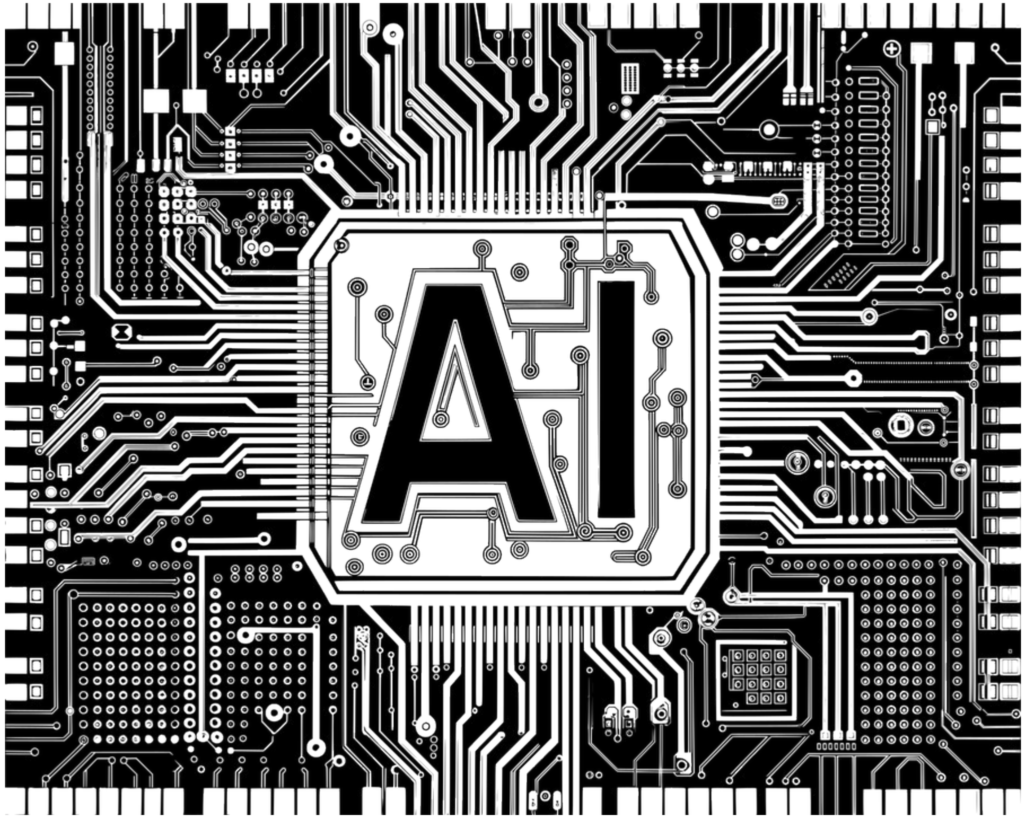


Conclusion

New complex hardware units to speed inference and training of new neural network models

Number representation

- double-precision, single-precision,
- half-precision, quarter-precision
- betting on adaptive capabilities of neural network models



Funding

This project has received funding from the European High-Performance Computing Joint Undertaking (JU) under grant agreement No 101101903. The JU receives support from the Digital Europe Programme and Germany, Bulgaria, Austria, Croatia, Cyprus, Czech Republic, Denmark, Estonia, Finland, Greece, Hungary, Ireland, Italy, Lithuania, Latvia, Poland, Portugal, Romania, Slovenia, Spain, Sweden, France, Netherlands, Belgium, Luxembourg, Slovakia, Norway, Türkiye, Republic of North Macedonia, Iceland, Montenegro, Serbia.
The SLING National Competence Centre is co-funded by the Ministry of Higher Education, Science and Innovation.

Co-funded by the European Union

REPUBLIKA SLOVENIJA
MINISTRSTVO ZA VISOKO ŠOLSTVO, ZNANOST IN INOVACIJE

Media sponsor

Računalniške novice
www.racunalniske-novice.com