SISSA, 2 December 2024
Junior Math Days 2024

A Reduced Order Approach for Artificial Neural Networks Applied to Object Recognition

Laura Meneghetti

Joint work with:
Nicola Demo and Gianluigi Rozza

Finanziato dall'Unione europea
NextGenerationEU

Research Experience

UNIVERSITÀ DEGLI STUDI DI PADOVA

Mathematics at University of Padova (Bachelor and Master)

2012-2018

1st edition Junior Math Days - May 2018

SISSA

PhD at SISSA in collaboration with Electrolux Professional

2018-2022

The Research Hub By ELECTROLUX PROFESSIONAL

Electrolux PROFESSIONAL

Post-Doctoral Researcher at SISSA

2022-now

Some of our interests

- **Model Order Reduction** (studying also integration with Deep Learning)
- **Computational Fluid Dynamics** (enhance broader applications in multiphysics and coupled settings, such as aeronautical, mechanical, naval, cardiovascular surgery, ...)
- **Digital Twins**
- **Machine Learning and Deep Learning**
- **Industrial and Medical Applications** (application of ROM and Deep Learning techniques for demanding applications)
- **Scientific Computing**

Our libraries

- ITHACA-DG
- WaveBEM
- ITHACA-SEM
- Athena
- Gea
- PyGeM
- EZyRB
- ITHACA - FV
- BladeX
- ATLAS
- PyDMD
- multiphenics
- RBniCS
- PINA
- ARGOS

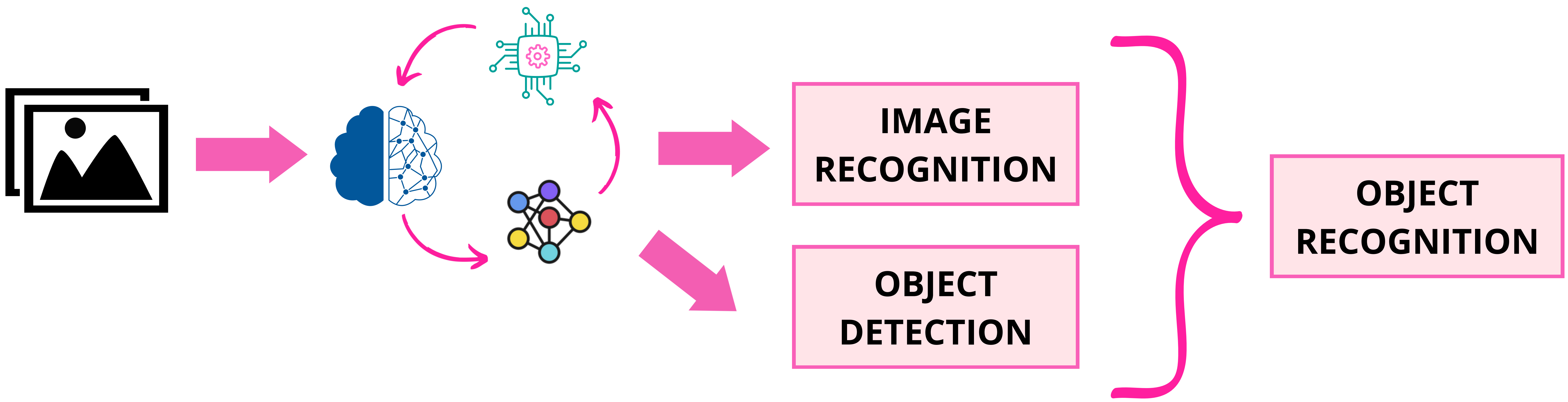
Libraries: platforms gathering functions to solve scientific problems quickly.

Open-source philosophy to facilitate sharing with other developers.

MOTIVATIONS

The Problem

Build a model based on Artificial Neural Networks (ANNs) able to recognize and detect the position of different types of objects in order to be included inside a vision embedded system.

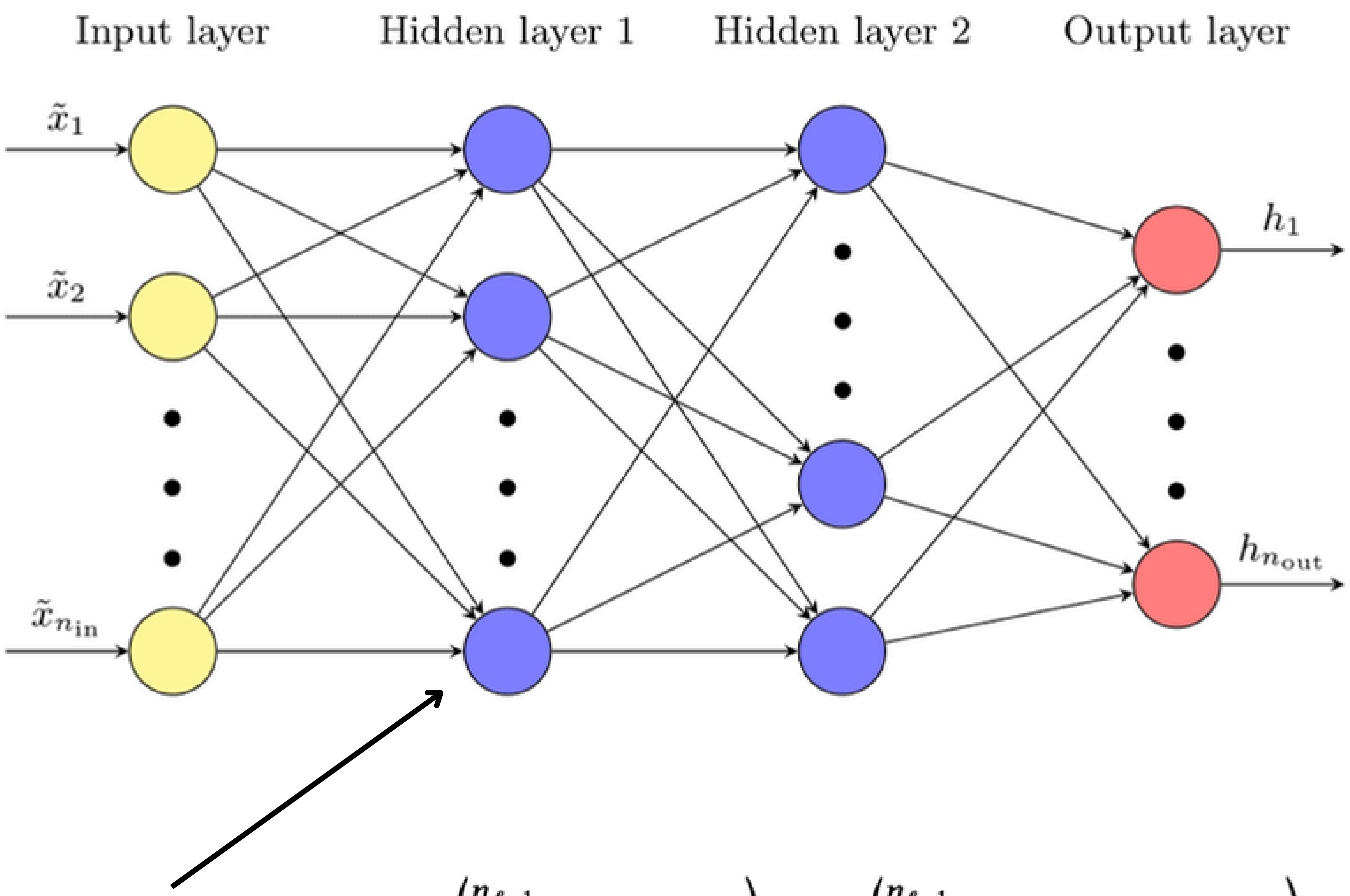


MOTIVATIONS

Artificial Neural Networks

Artificial Neural Networks (ANNs) are computational models inspired by the human brain, designed to recognize patterns and solve complex problems. Their main components are:

- **neurons**: basic units
- **layers**: input, output or hidden
- **weights**: connections between neurons from different layers



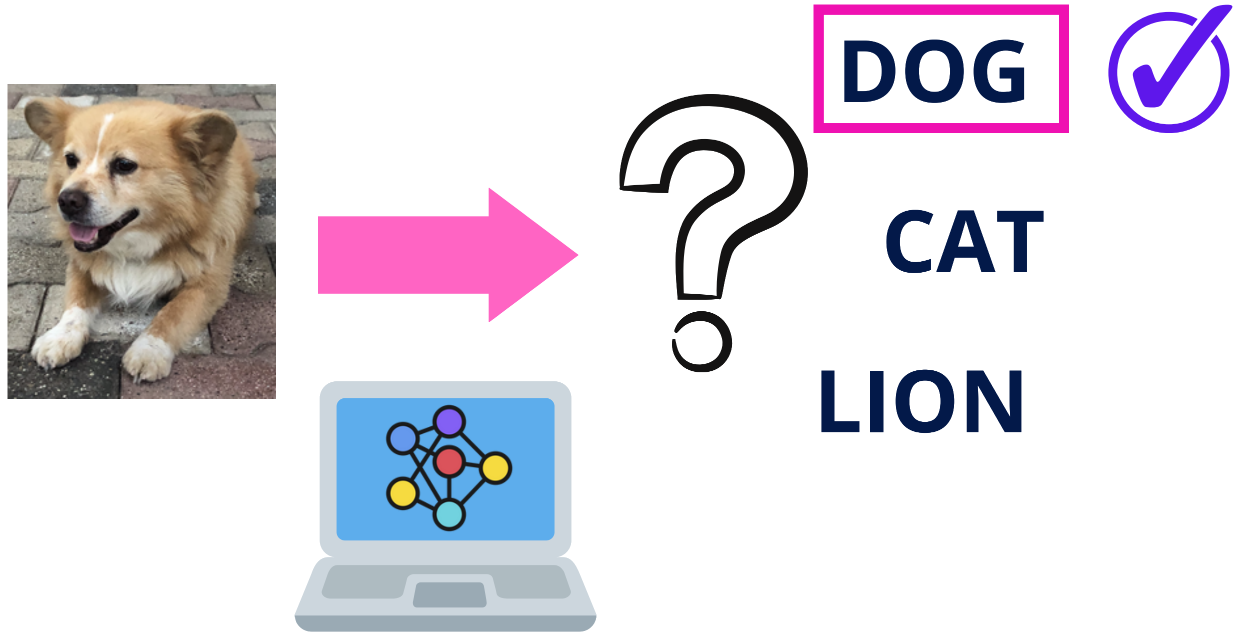
$$x_j^{(\ell)} = \sigma(h_j^{(\ell)}) = \sigma\left(\sum_{i=0}^{n_{\ell-1}} w_{ji}^{(\ell)} x_i^{(\ell-1)}\right) = \sigma\left(\sum_{i=1}^{n_{\ell-1}} w_{ji}^{(\ell)} x_i^{(\ell-1)} - b_j^{(\ell)}\right)$$

ANNs are widely used in tasks like image recognition, natural language processing, and decision-making.

MOTIVATIONS

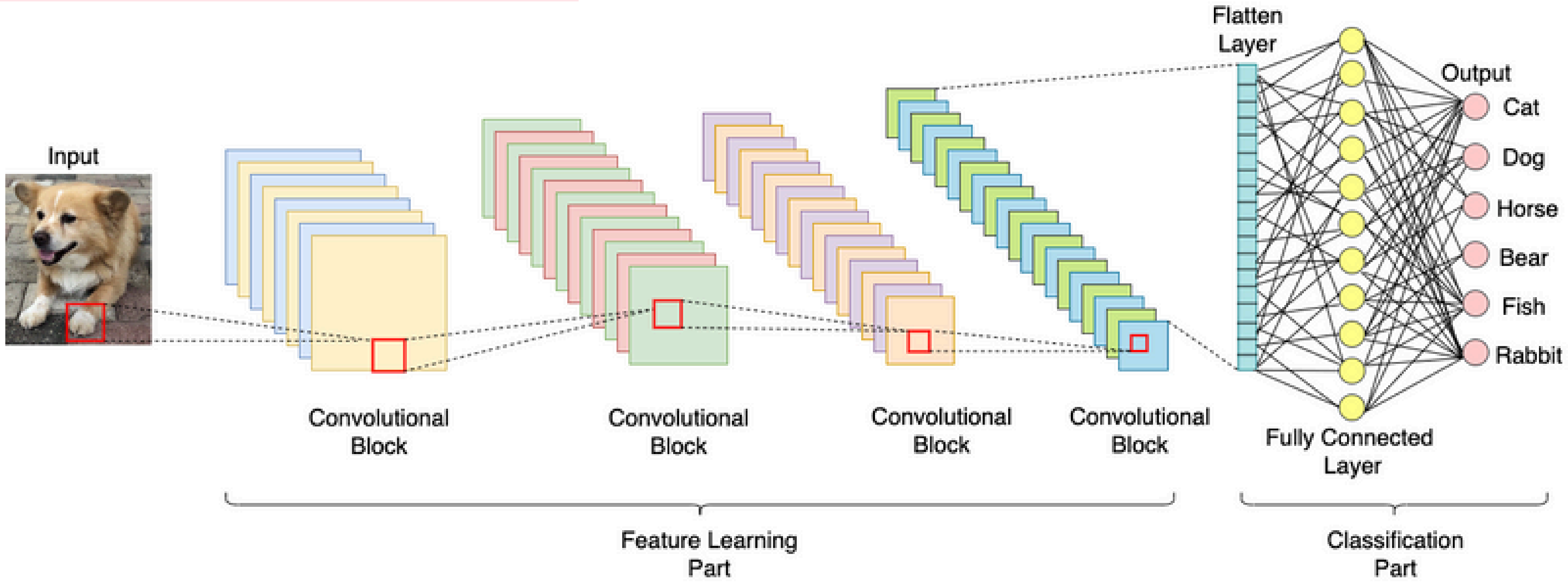
Image Recognition & CNNs

Given a picture, classify the depicted objects.
A **Convolutional Neural Network** (CNN) is a Deep Learning algorithm which can take as input an image, assign importance (learnable weights and biases) to various aspects/objects in the image and be able to differentiate one from the other.



CNN architecture can be mainly subdivided in two parts:

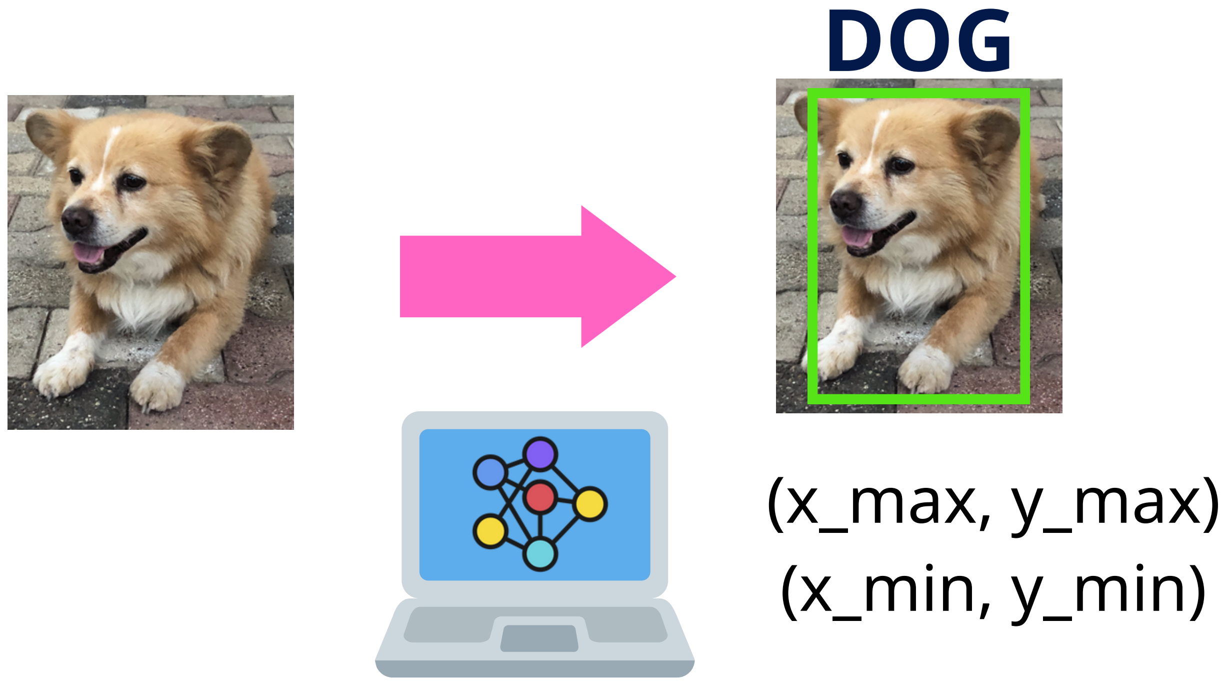
- **Feature Learning Part** responsible for detecting the objects features.
- **Classification Part** providing the final classification.



MOTIVATIONS

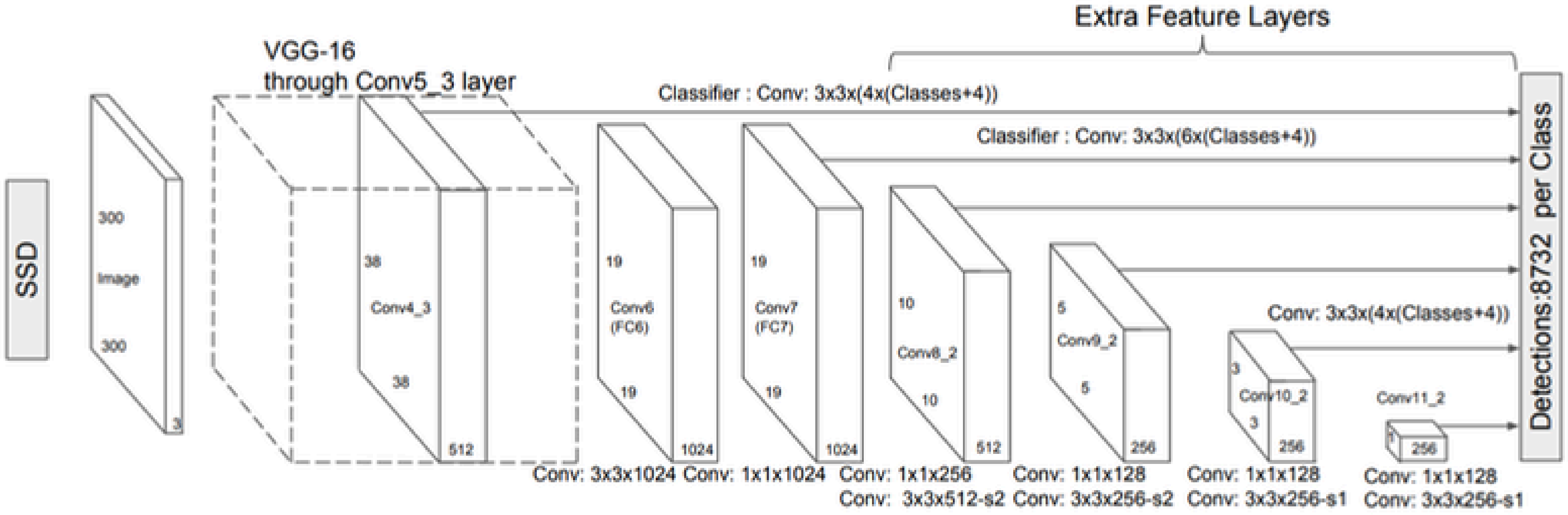
Object Detection

Given a picture, classify and localize the depicted objects. Object Detectors are Deep Learning algorithms developed to solve this task.



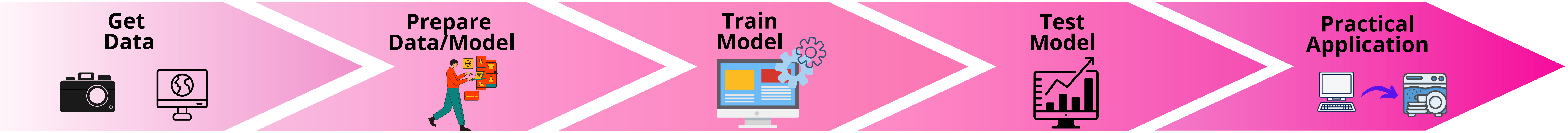
We focus on **SSD-type architectures**, such as SSD300, composed of:

- a **base net** (CNN), extracting the low-level features;
- some **auxiliary layers**, responsible for capturing the high-level features;
- **two siblings predictors**, one for the classification and one for the localization of the objects.



MOTIVATIONS

Development pipeline of an Artificial Neural Network for the problem of Object Recognition to be later deployed in vision embedded systems.



Get Data
- Benchmark Dataset
- Custom Dataset

Prepare Data/Model
- Convolutional Neural Networks
- Object Detectors

↓

- high number of parameters (space required)
- need GPU for training
- demanding hardware resources

→ DIMENSIONALITY PROBLEMS ←

Practical Application

Embedded Systems

↓

- limited hardware resources
- strict memory constraints
- low CPU performance

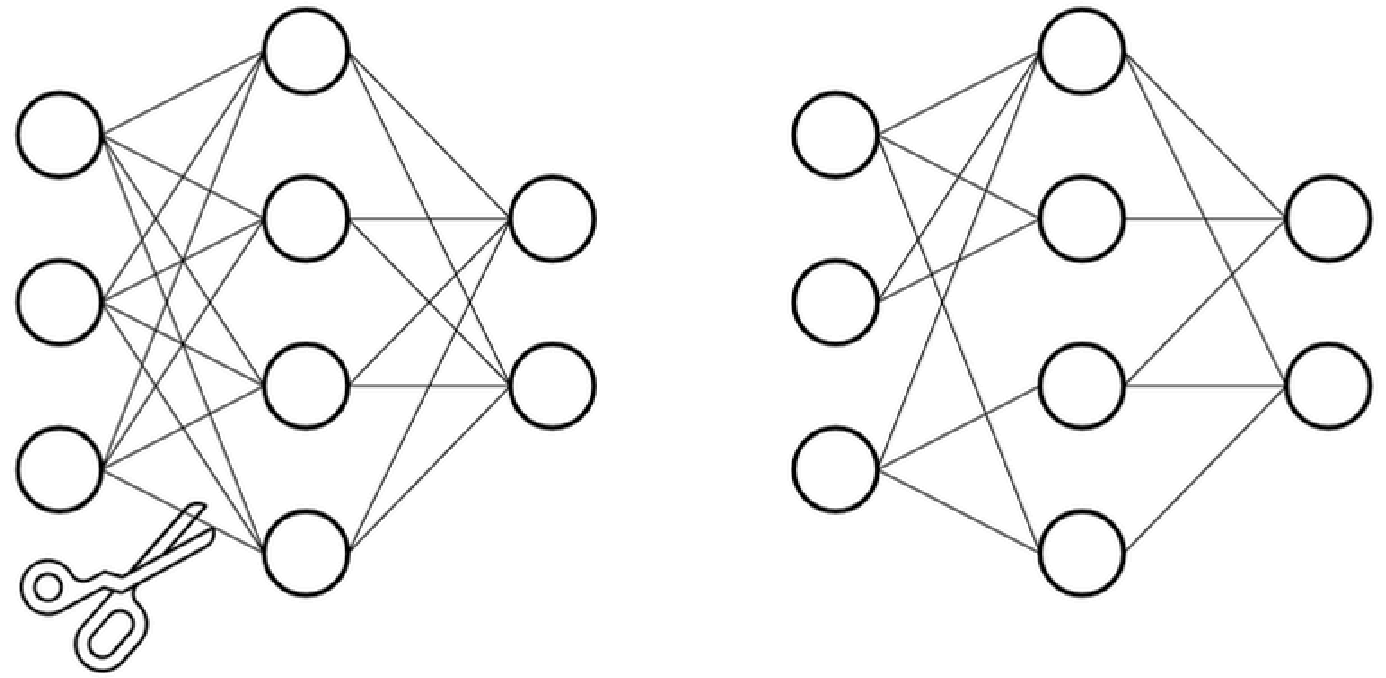
A REDUCED APPROACH

DIMENSIONALITY PROBLEMS

SOLUTION

There exists different approaches to compress an Artificial Neural Network:

- **Network Pruning**
- **Parameter Quantization**
- **Development of Efficient Neural Networks Architectures**
- **Low-rank and Tensor Factorization**
- **Knowledge Distillation**
- **Manually designing convolutional layers**

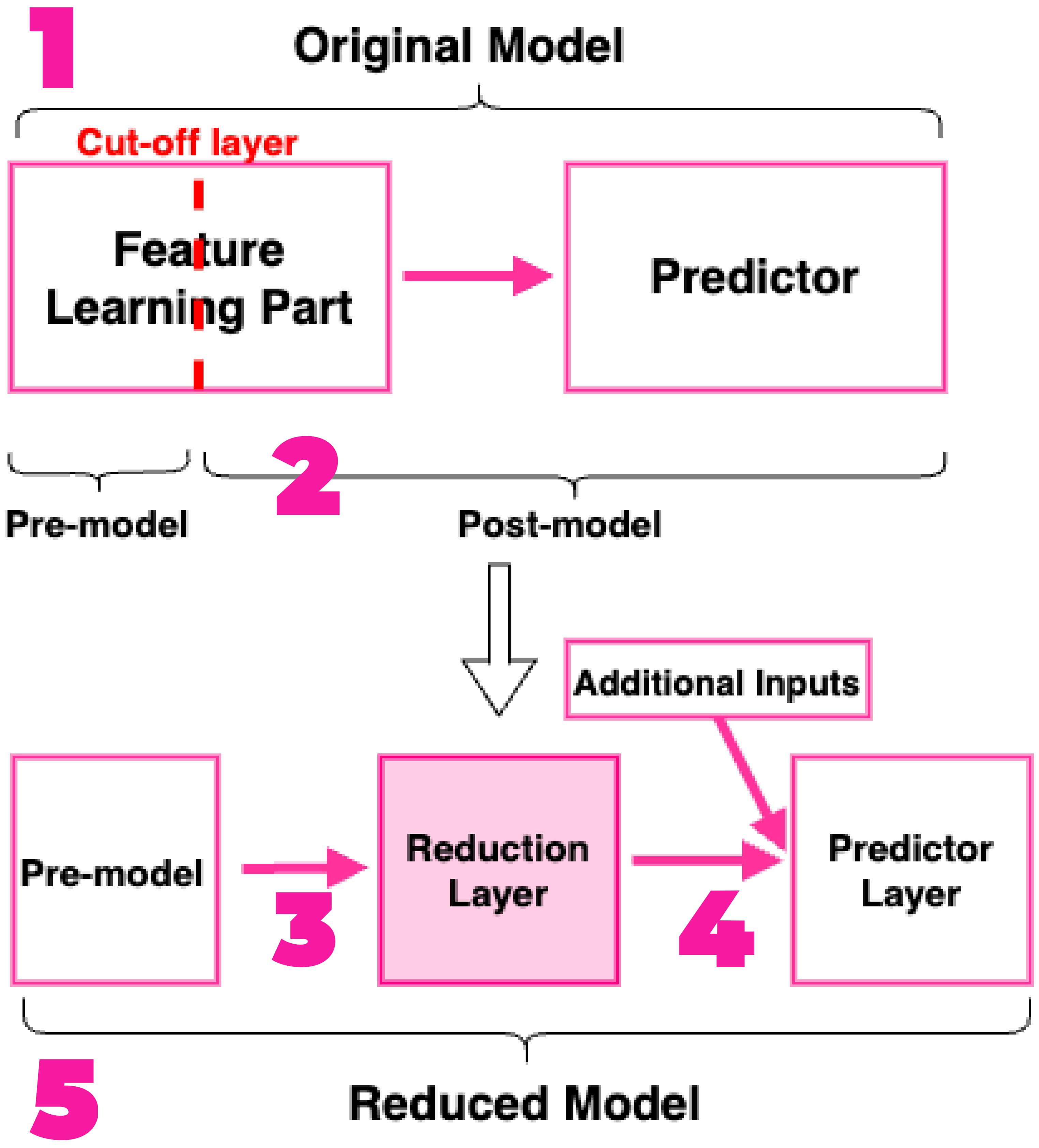


Before pruning

After pruning

A REDUCED APPROACH

1. Original Neural Network
2. Network Splitting
3. Dimensionality Reduction
4. Predictor
5. Compressed Neural Network



A REDUCED APPROACH

1 Original Neural Network

Take an Artificial Neural Network. It can be described as the compositions of its layers: $\mathcal{ANN} \equiv f_L \circ f_{L-1} \circ \cdots \circ f_1$.

2 Network Splitting

Choose a cut-off layer l (based on L1-norm analysis). Define:

- the **pre-model** as : $\mathcal{ANN}^l_{\text{pre}} = f_l \circ f_{l-1} \circ \cdots \circ f_1$,
- the **post-model** as: $\mathcal{ANN}^l_{\text{post}} = f_L \circ f_{L-1} \circ \cdots \circ f_{l+1}$

Discard the post-model.

PAY ATTENTION

The role of cut-off index is crucial to obtain a good level of accuracy of the reduced network.
It is chosen based on an L1-norm analysis and on considerations about the network and the dataset at hand, balancing the final accuracy and the compression ratio.

A REDUCED APPROACH

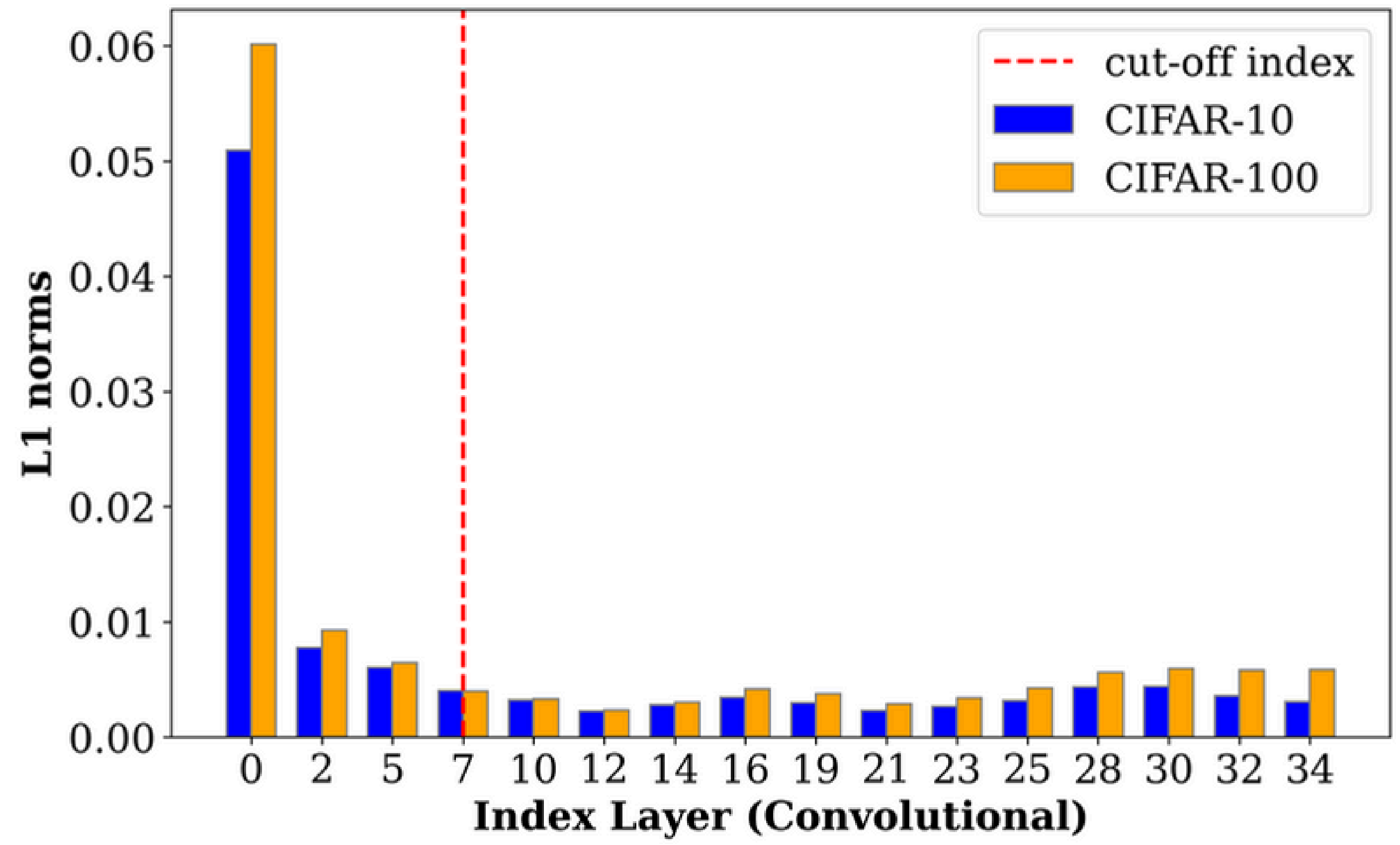
Computation of cut-off index

- Evaluate the L1 norm, normalized by parameter count, across each trainable layer (convolutional and linear)
- Find the layer index at which the L1-norm stabilizes, with subsequent values remaining comparable or lower

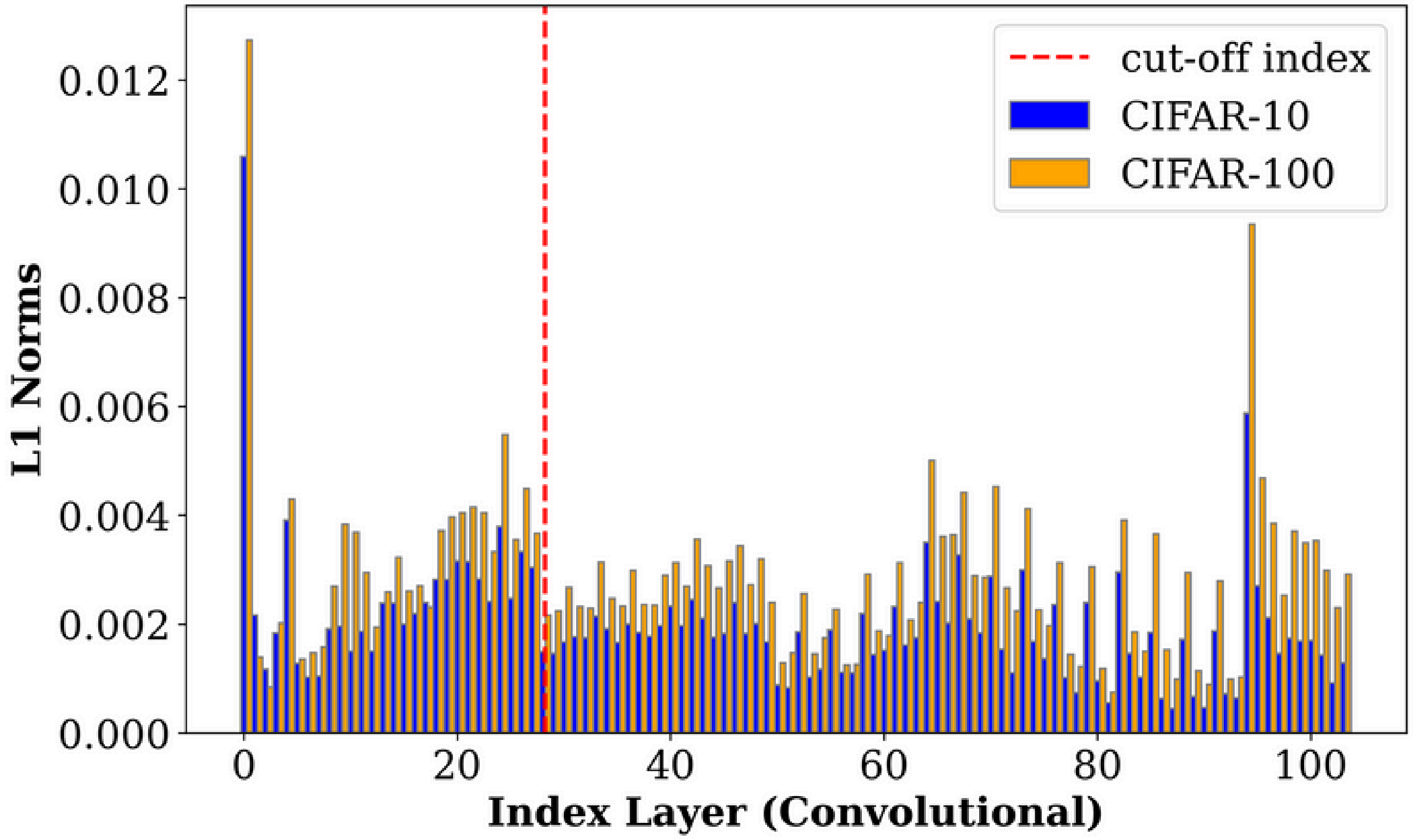
Tested with:

- several datasets: CIFAR10, CIFAR100, STL10
- two CNNs: ResNet-110, VGG-16

VGG-16



ResNet-101



A REDUCED APPROACH

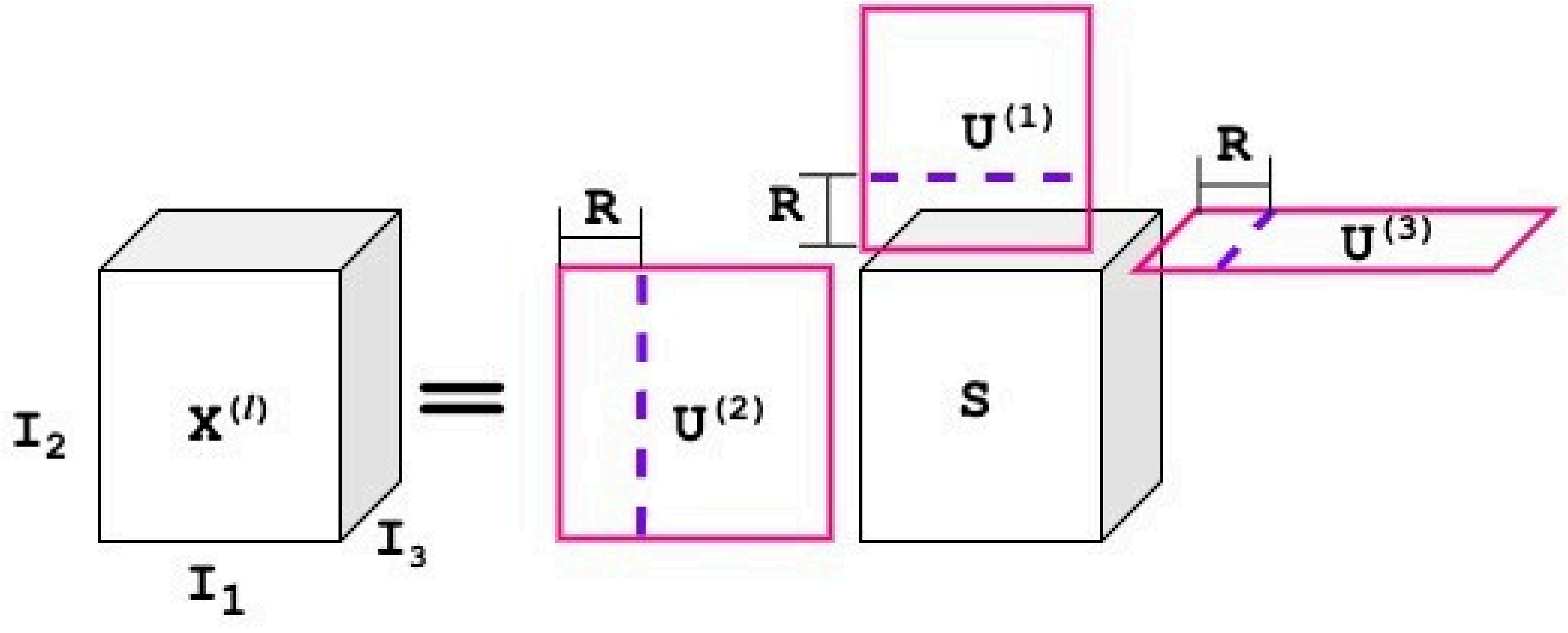
3 **Dimensionality Reduction**

- Compute the pre-model outputs for each input in the train dataset.
- Pre-model outputs lies in a high dimensional space.
- **IDEA**: Project them into a low-dimensional space.

Tested Reduction Methods:

- **POD**: Compute the trunctated Singular Value Decomposition (SVD) of the matrix containing the linearized pre-model outputs.
- **HOSVD**(generalization in higher dimensions of SVD): Take into account the tensorial structure of pre-model outputs by computing the truncated SVD in each tensor direction.

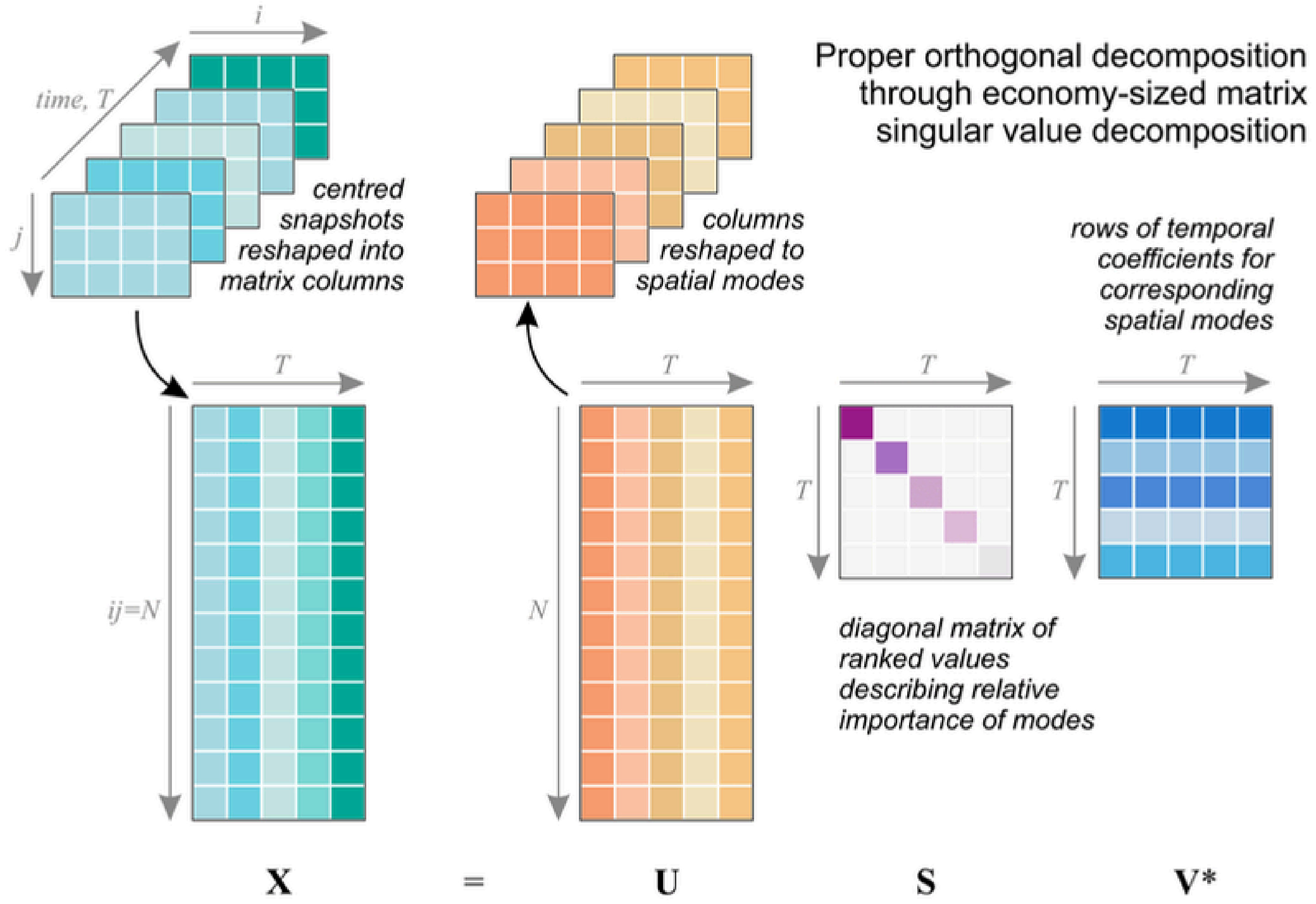
Goal of the reduction methods: determine and retain the most important parameters (eingenvalue analysis).



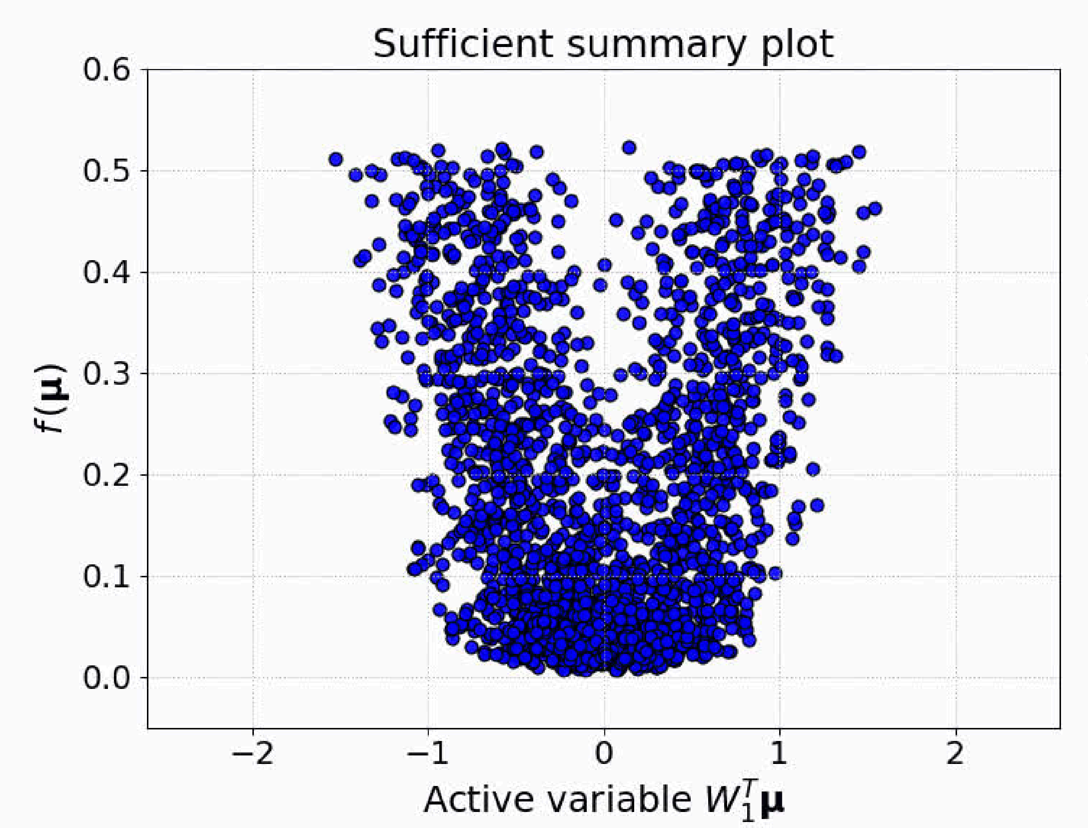
A REDUCED APPROACH

Idea Reduced Order Methods

- Construct a matrix with your inputs (snapshots)
- Compute the SVD of this matrix
- Perform an eigenvalue analysis to study the decay of the singular values.
- Retain only the first r singular values (most important ones) and so the first r columns of U ("eigenvectors")



➡ Retain only the most important directions in the parameter space, determined by the singular value analysis.



A REDUCED APPROACH

POD

- Flatten of the pre-model outputs.
- Snapshot matrix $\mathsf{S} = [\mathsf{x}^{(\ell),1} \dots \mathsf{x}^{(\ell),n_{\text{train}}}]$
- Compute the SVD of S: $\mathsf{S} = \Psi\Sigma\Theta^T$
- Perform an eigenvalue analysis to study the decay of the singular values.
- Retain only the first r eigenvalues and so the first r columns of Ψ
- Use the reduced version of Ψ as projection matrix for the pre-model outputs:

$$\mathsf{z} = \Psi_r^T \mathsf{x}^{(\ell)}.$$

HOSVD

- i-th pre-model output: $A = \mathsf{x}^{(\ell),i} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$
- Compute the (three) matrix unfoldings of A
- Compute the SVD of each unfolding:

$$U^{(j)}, \Sigma^{(j)}, V^{(j)} = \text{SVD}(A_{(j)})$$

- Perform an eigenvalue analysis along each direction.
- Retain the first R_1, R_2, R_3 columns of the U matrices, obtaining thus the projection matrices along each direction.
- The reduced tensor is obtained as follows:

$$S_R = A \times_1 U_{R_1}^{(1),T} \times_2 U_{R_2}^{(2),T} \times_3 U_{R_3}^{(3),T} \in \mathbb{R}^{R_1 \times R_2 \times R_3}$$

4-th dimensional tensors (batch size) → **Averaged HOSVD AHOSVD**

A REDUCED APPROACH

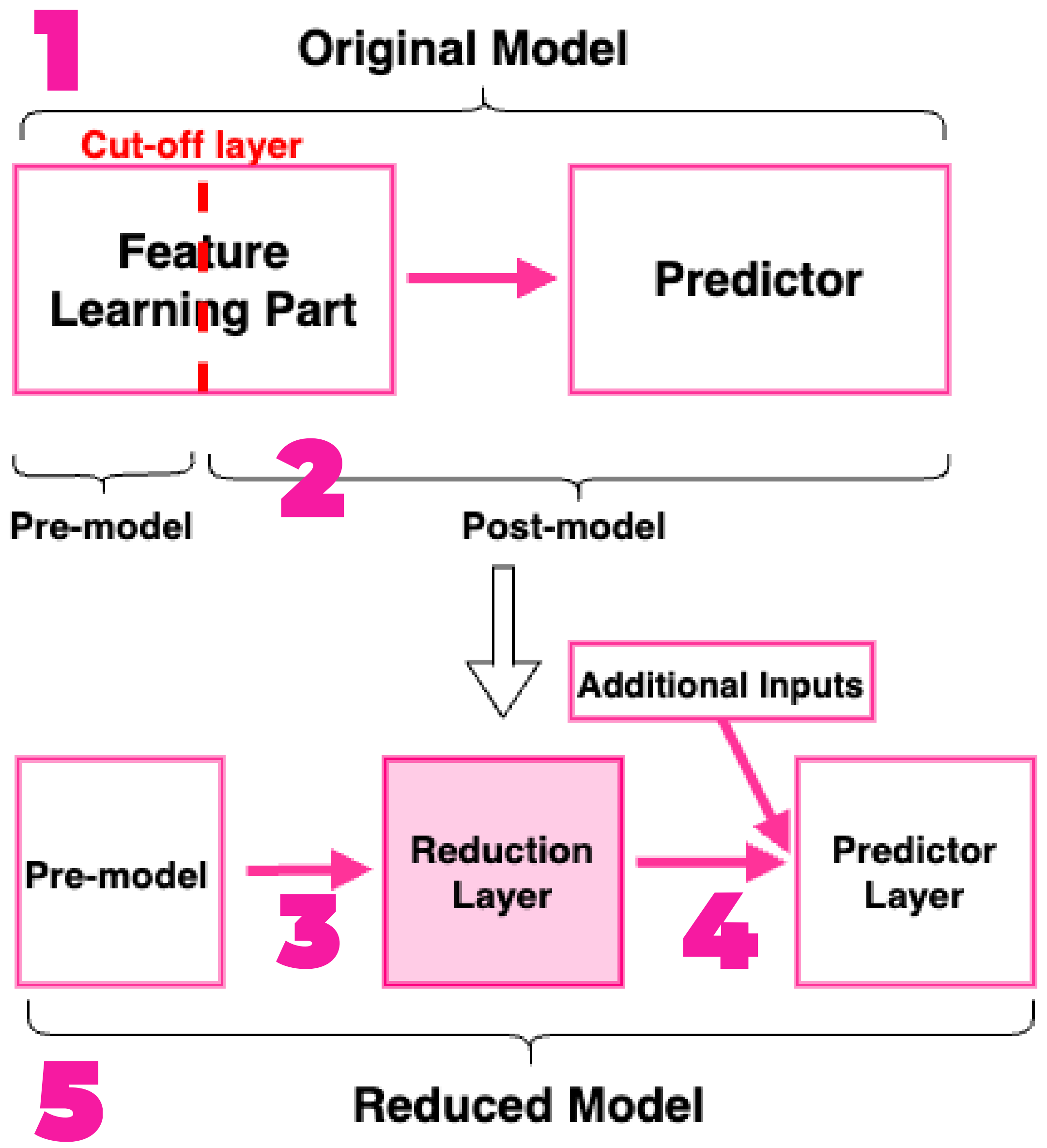
4 **Predictor**

- The output of Step 3 becomes the input for the predictor, together with some additional inputs.
- Maintain the predictor's architecture of the original model, adjusting it for the variation in size of its inputs.

5 **Compressed Neural Network**

Re-training of the constructed Reduced Artificial Neural Network to achieve a good level of accuracy, comparable to the original model.

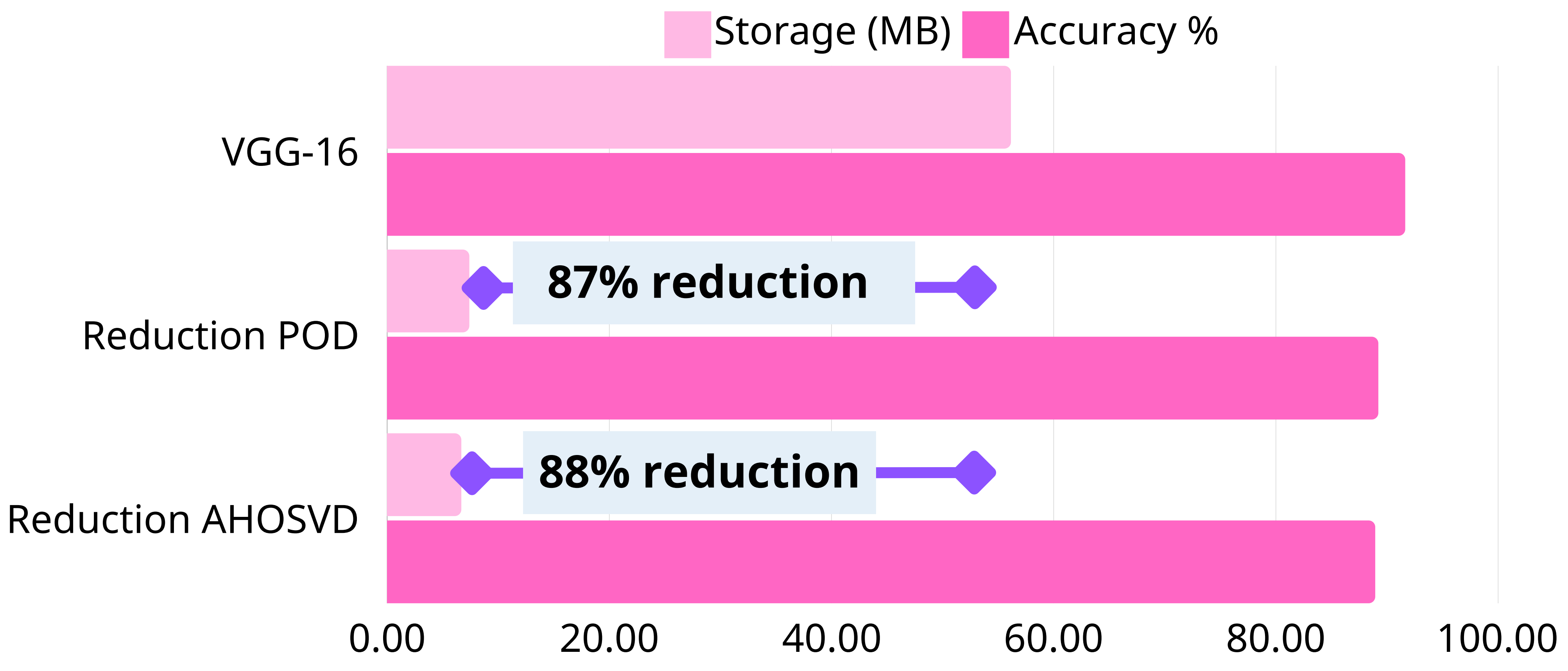
Used: Knowledge Distillation for the Image Classification task.



RESULTS

Image Recognition

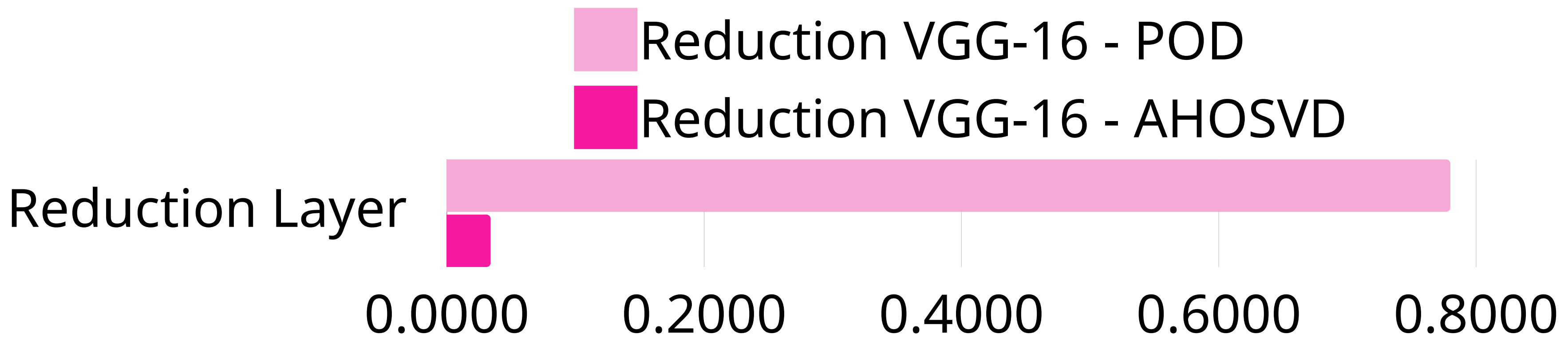
- Original Model: VGG-16
- Dataset: CIFAR10
- Cut-off layer: 7
- Reduced dimension R: 50 for POD, 3x3x35 for AHOSVD



Storage comparison for the reduction layers using POD or AHOSVD.

Other tests done with:

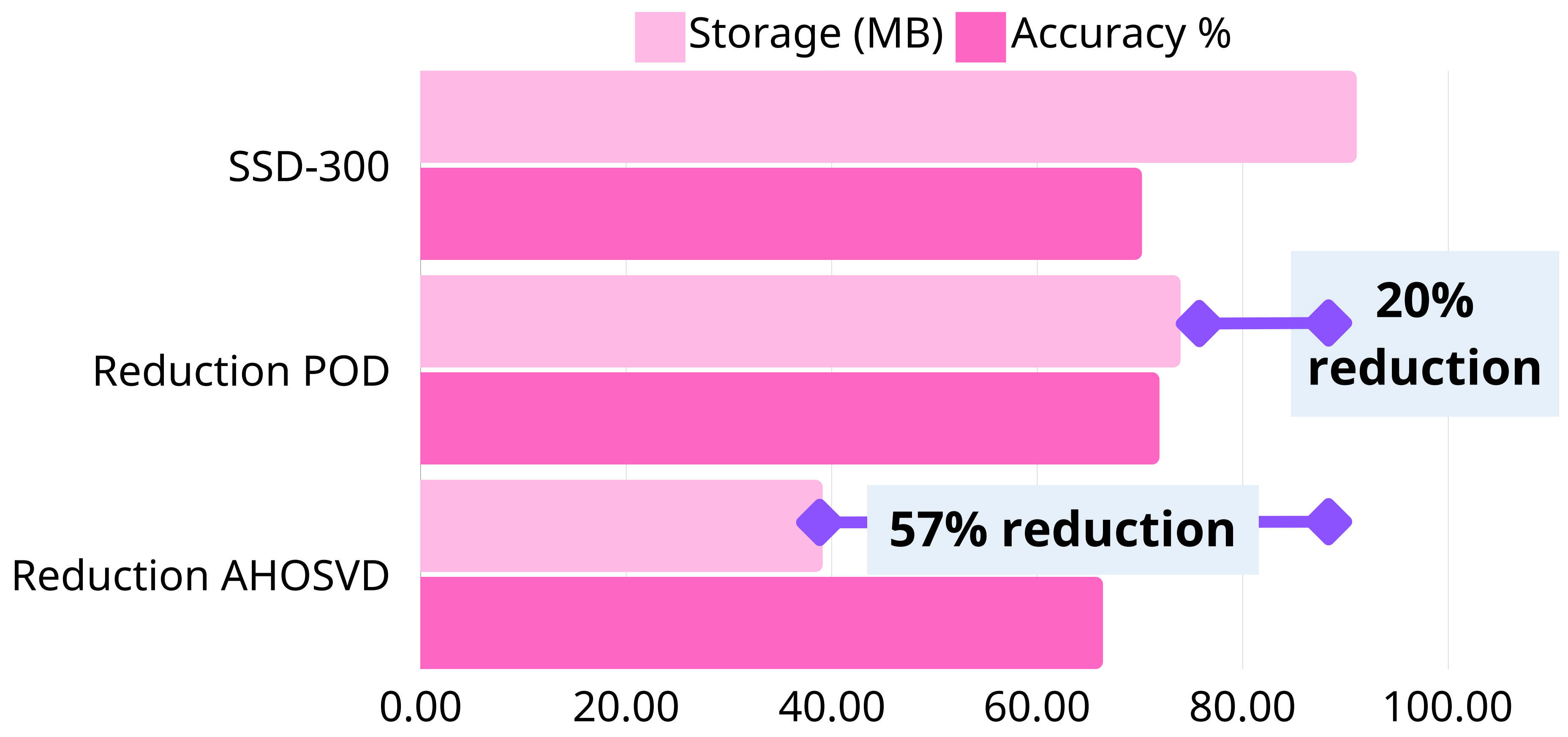
- Original Model: ResNet-110 (cut-off 28)
- Dataset: CIFAR-100, STL10
- Different cut-off layers and different reduced dimensions



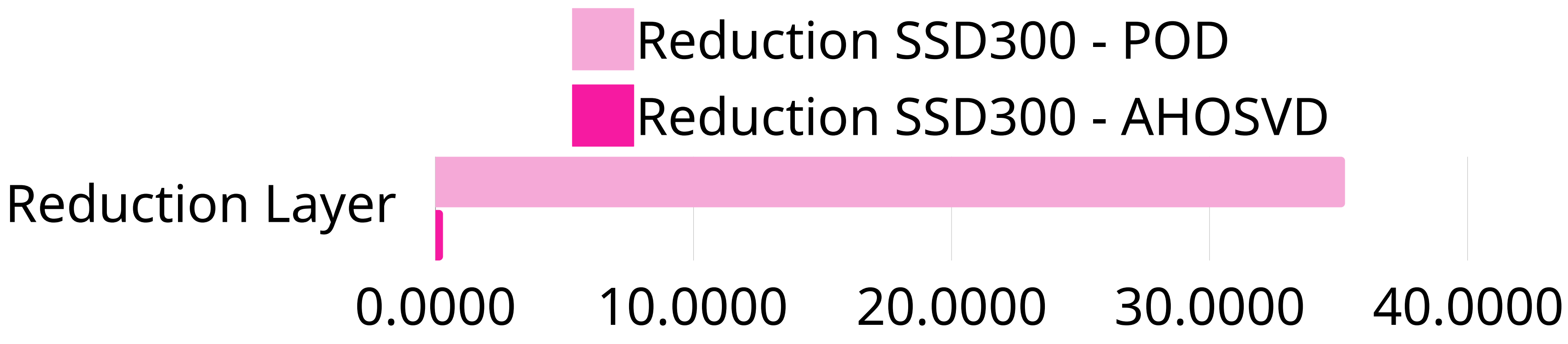
RESULTS

Object Detection

- Original Model: SSD300
- Dataset: Cats &Dogs Dataset, a smaller dataset (300 images and two categories) extracted from PASCALVOC
- Cut-off layer: 11
- Reduced dimension R: 50 for POD, 3x3x150 for AHOSVD



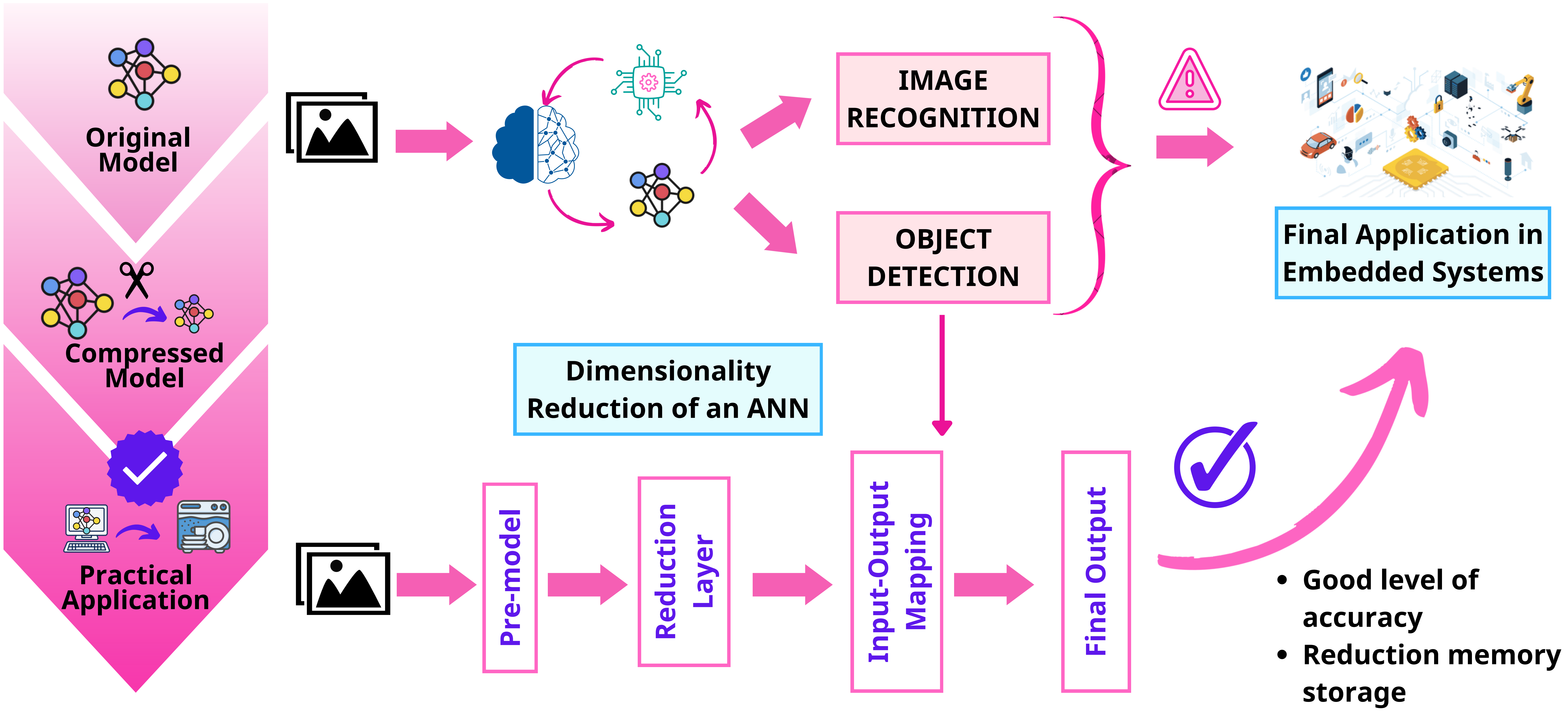
Storage comparison for the reduction layers using POD or AHOSVD.



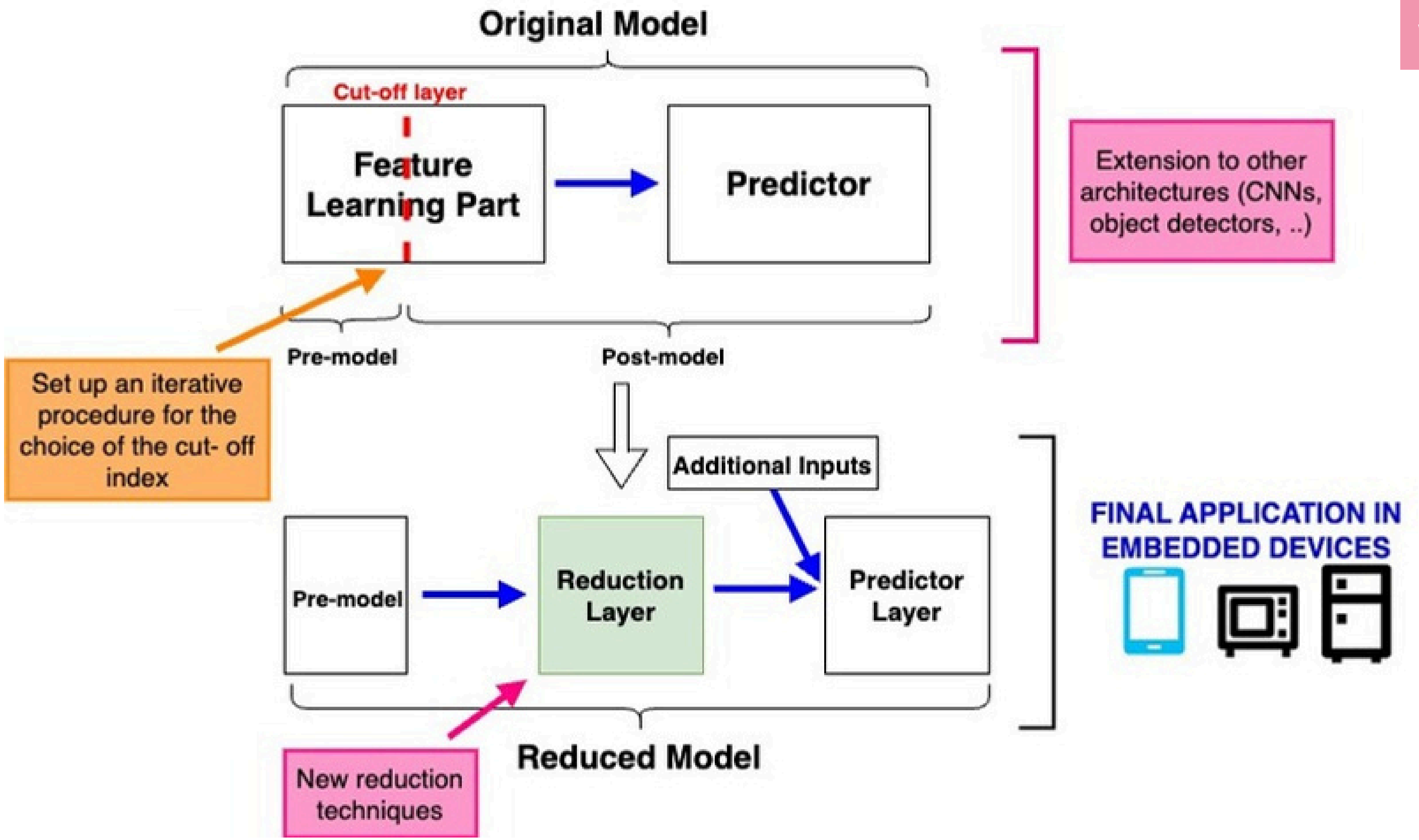
Other tests done with:

- Different cut-off layers and different reduced dimensions

CONCLUSIONS



CONCLUSIONS



Some Future Developments

Generalizability of the approach
more tasks, more architectures, more datasets

Criteria for cut-off index
Information theory notions (e.g. entropy) to understand the most important neurons/layers

More reduction tecniques
e.g. non linear ones

Comparison and integration of other compression methods
e.g. pruning, quantization,..

Thank you for your Attention!

Any Questions?

REFERENCES

- Meneghetti, L., Demo, N., Rozza, G.: A dimensionality reduction approach for convolutional neural networks. Applied Intelligence 53(19), 22818–22833 (2023). https://doi.org/10.1007/s10489-023-04730-1
- Meneghetti, L., Demo, N., Rozza, G.: A Proper Orthogonal Decomposition Approach for Parameters Reduction of Single Shot Detector Networks. In: 2022 IEEE ICIP. pp. 2206–2210 (2022). https://doi.org/10.1109/ICIP46576.2022.9897513
- Meneghetti, L., Bianchi, E., Demo, N., Rozza,: KD-AHOSVD: Neural Network Compression via Knowledge Distillation and Tensor Decomposition (2024) submitted

Check our GitHub page for the code!