kcg.edu MPSoC 2024, Kanazawa, Japan

Proposal of a Hyperparameter Optimization Method for Neural Networks

July 8, 2024

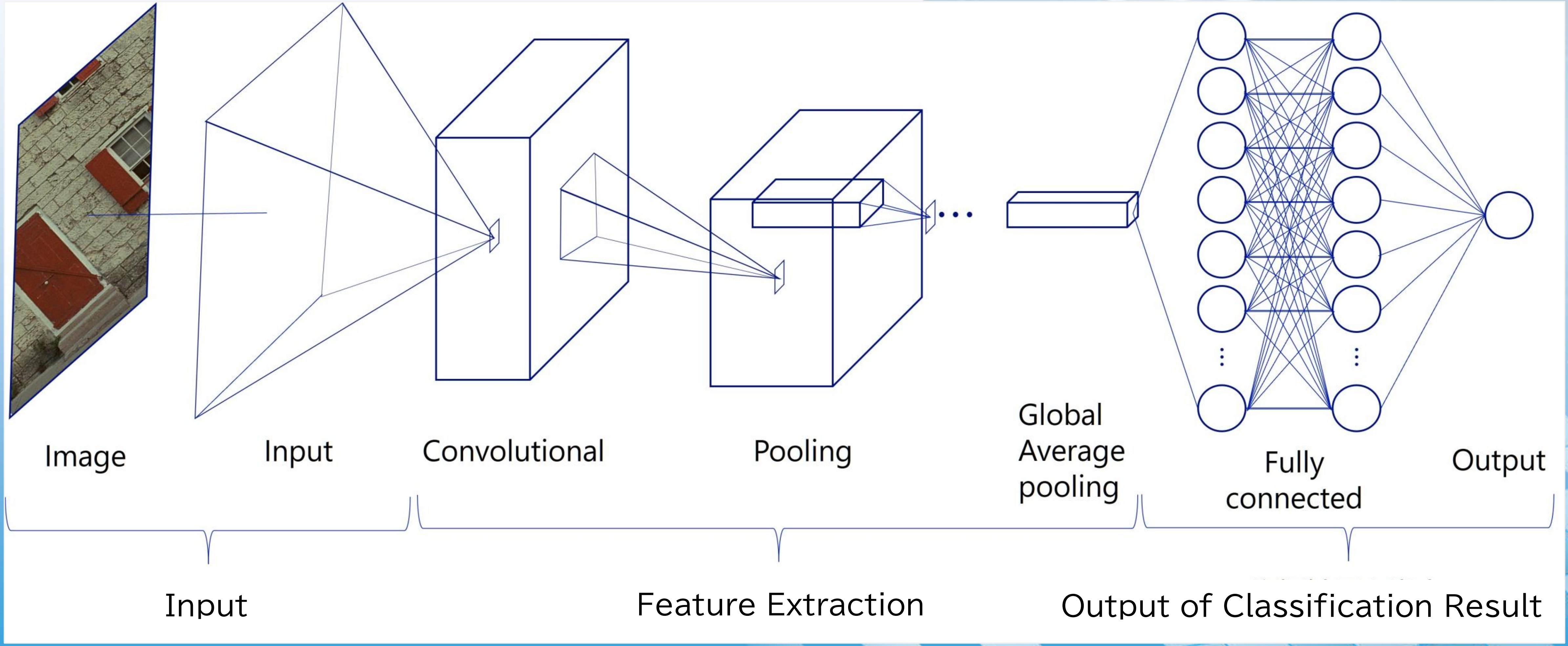
Masaharu Imai (m_imai@kcg.edu)

The Kyoto College of Graduate Studies for Informatics (KCGI), and
ASIP Solutions, Inc.

Agenda

- Introduction
- Hyperparameters of CNN
- Hyperparameter Optimization
- Proposal of Cross-Search Method
- Experimental Results
- Conclusion & Future Work
- References

Convolutional Neural Network



Motivation for this research

- Neural networks for machine learning are becoming larger and deeper (such as LLM)
- Cloud servers consume large amounts of electric energy
- Hyperparameter (HP) Optimization is crucial to obtain <u>more intelligent ML systems with less energy consumption</u>

Agenda

- Introduction
- Hyperparameters of CNN
- Hyperparameter Optimization
- Proposal of Cross-Search Method
- Experimental Results
- Conclusion & Future Work
- References

Hyperparameters (HP) of CNN

- In Convolutional Neural Networks (CNNs), there are many hyperparameters that you can tune to optimize the performance of the model.
- Kind of parameters
  - Hard type
  - Soft type

Hard Type HP Examples

- Number of Filters (Kernels)
- Filter Size (Dimensions)
- Stride
- Padding
- Pooling Size and Types
- Architecture

Soft Type HP Examples

- Learning Rate
- Number of Epochs
- Batch Size
- Dropout Rate
- Regularization Parameters
- Activation Function
- Optimizer, etc.

HP Optimization

- Challenging Issue due to the following reasons:
- Vast Parameter Space
  - Numerous Combinations
  - Interdependencies of Parameters
- Execution Time and Cost
  - Expensive and Time-consuming Experiments
  - Probabilistic Behavior of Component
- Complex Interactions
  - Parameter Interactions
  - Global Optimization Challenge

Agenda

- Introduction
- Hyperparameters of CNN
- Hyperparameter Optimization
- Proposal of Cross-Search Method
- Experimental Results
- Conclusion & Future Work
- References

HP Optimization Methods

- Blackbox
  - Optimization using objective function values only (current mainstream)
- Gray box
  - Utilize auxiliary information useful for optimization derived from the characteristics of the target problem(current trend)
- Others
  - Gradient method, Reinforcement Learning (not major)

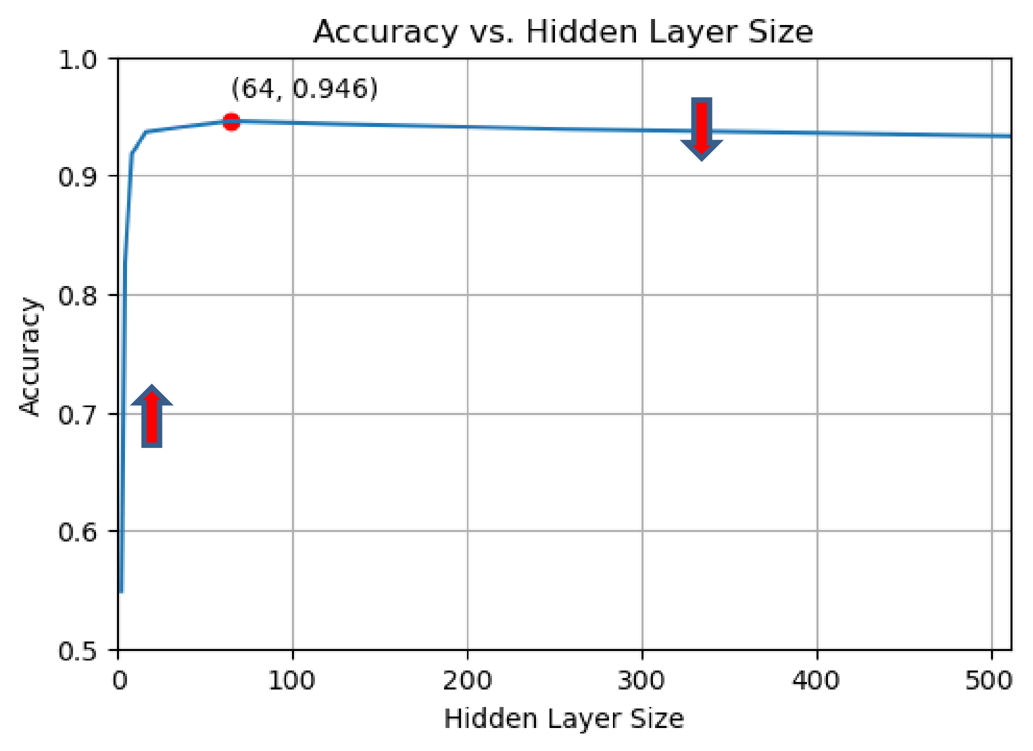
Agenda

- Introduction
- Hyperparameters of CNN
- Hyperparameter Optimization
- Proposal of Cross-Search Method
- Experimental Results
- Conclusion & Future Work
- References

Observations (1)

- Optimizing the number of neurons in each layer of a Neural Network has the following ease and difficulties
- Ease
  - The shape of learning curves (Loss, Accuracy) are roughly unimodal
- Difficulties
  - Computation time for the learning is dominant
  - The learning curves are superimposed on the stochastic error

Accuracy vs Hidden Layer Size

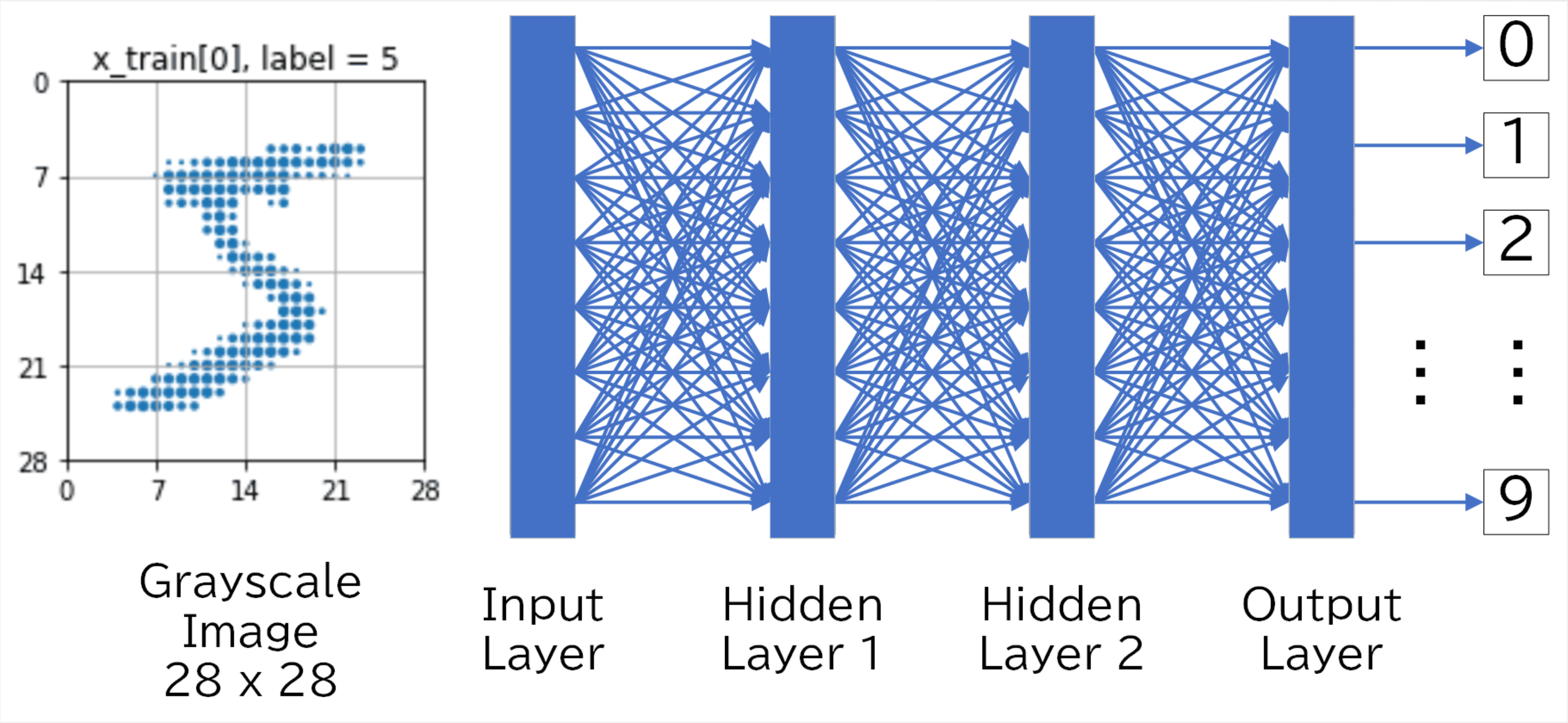


- Initially, Accuracy increases as the number of neurons increases.
- When the number of neurons exceeds a certain value, Accuracy decreases. (Overlearning)
- The graph is roughly unimodal

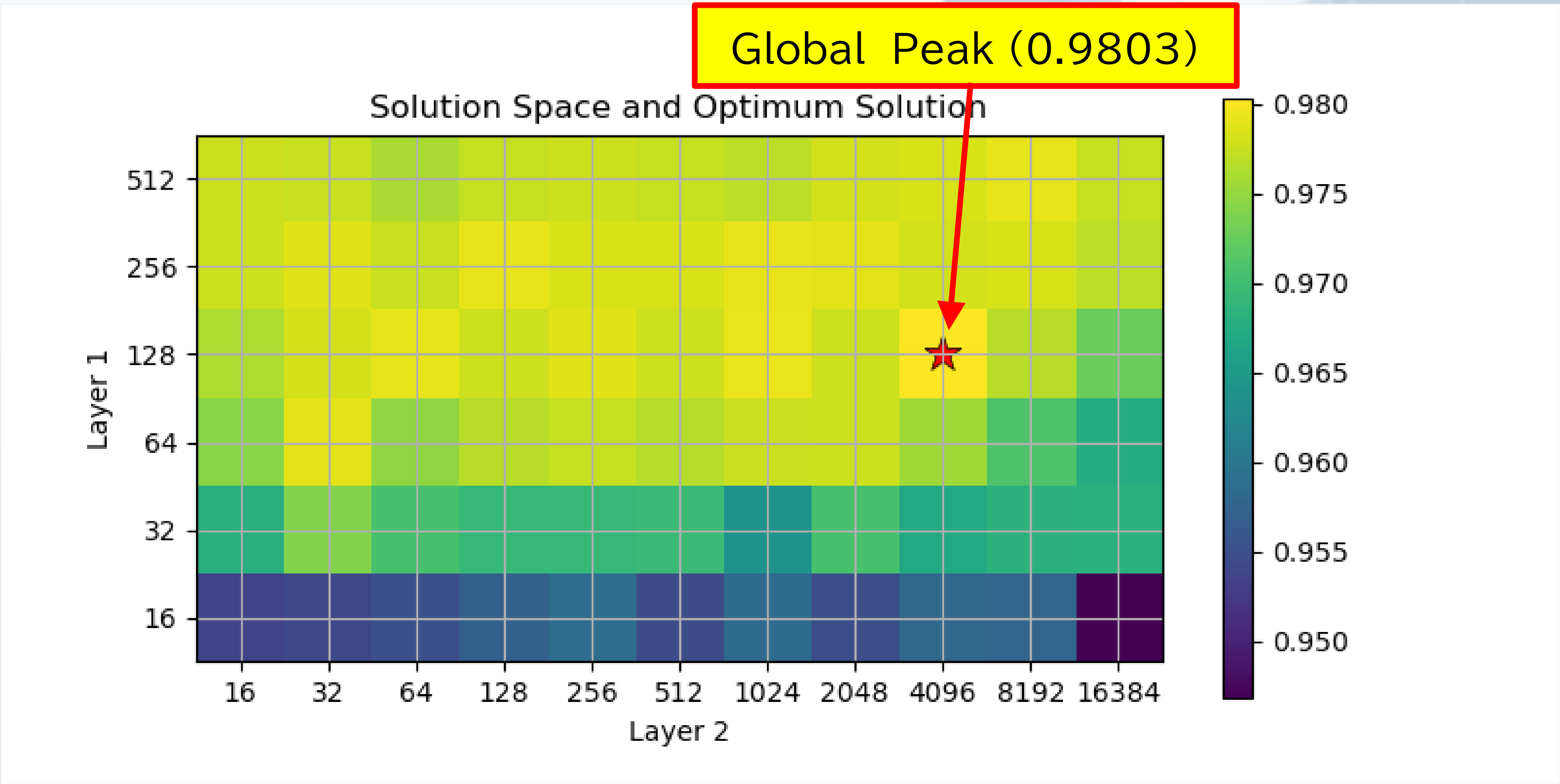
Observations (2)

- Full Grid Search can find the global optimum HP combination, but time-consuming and not practical
- Hill climbing search is one of the best and easiest methods to understand, but it may fail to find the Global Optimum Solution
- Any good idea?

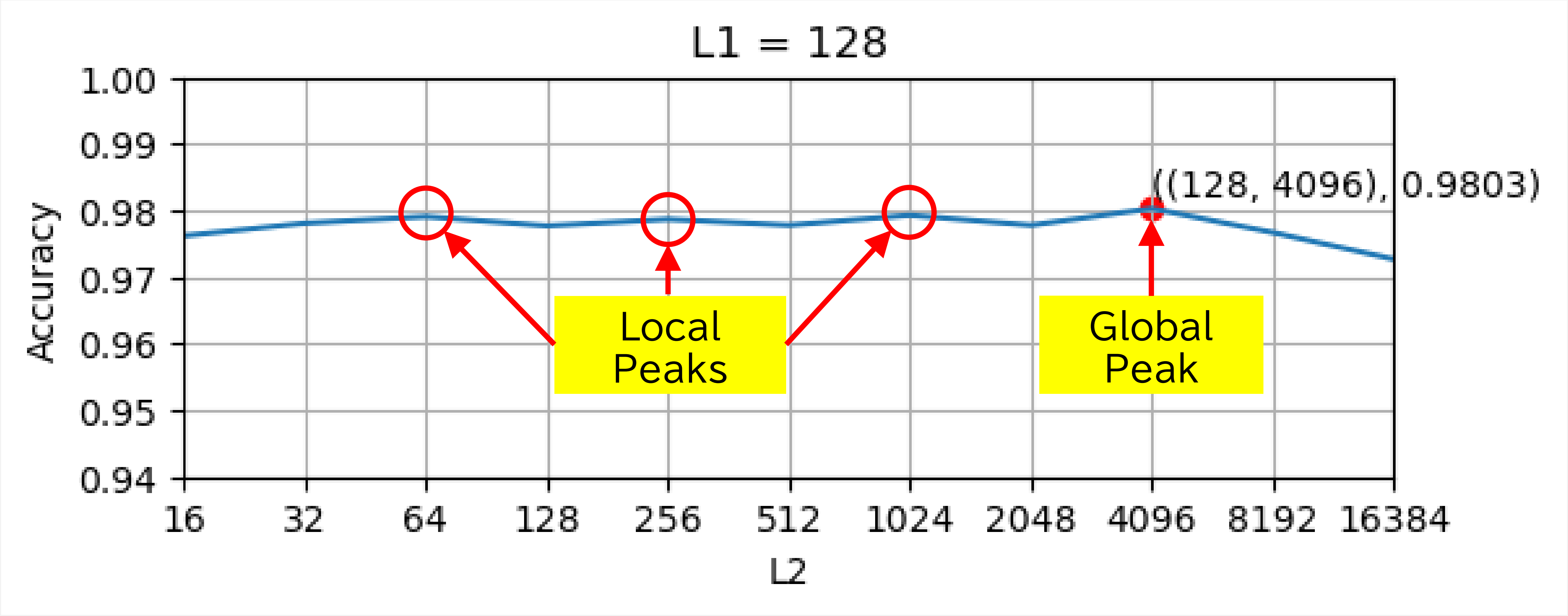
3 Layer Neural Network



Solution Space and Global Peak



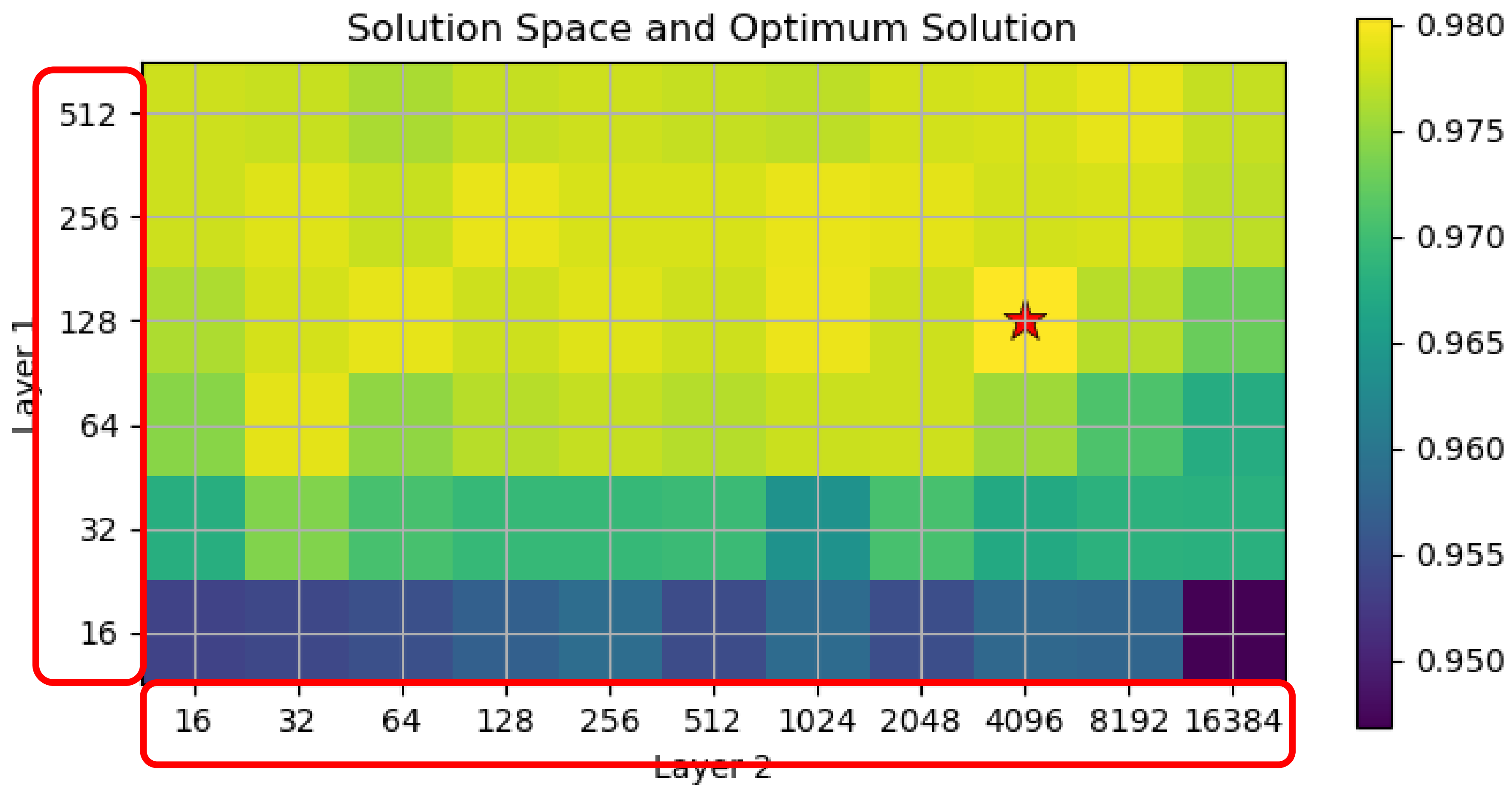
“Noise” makes Local Peaks



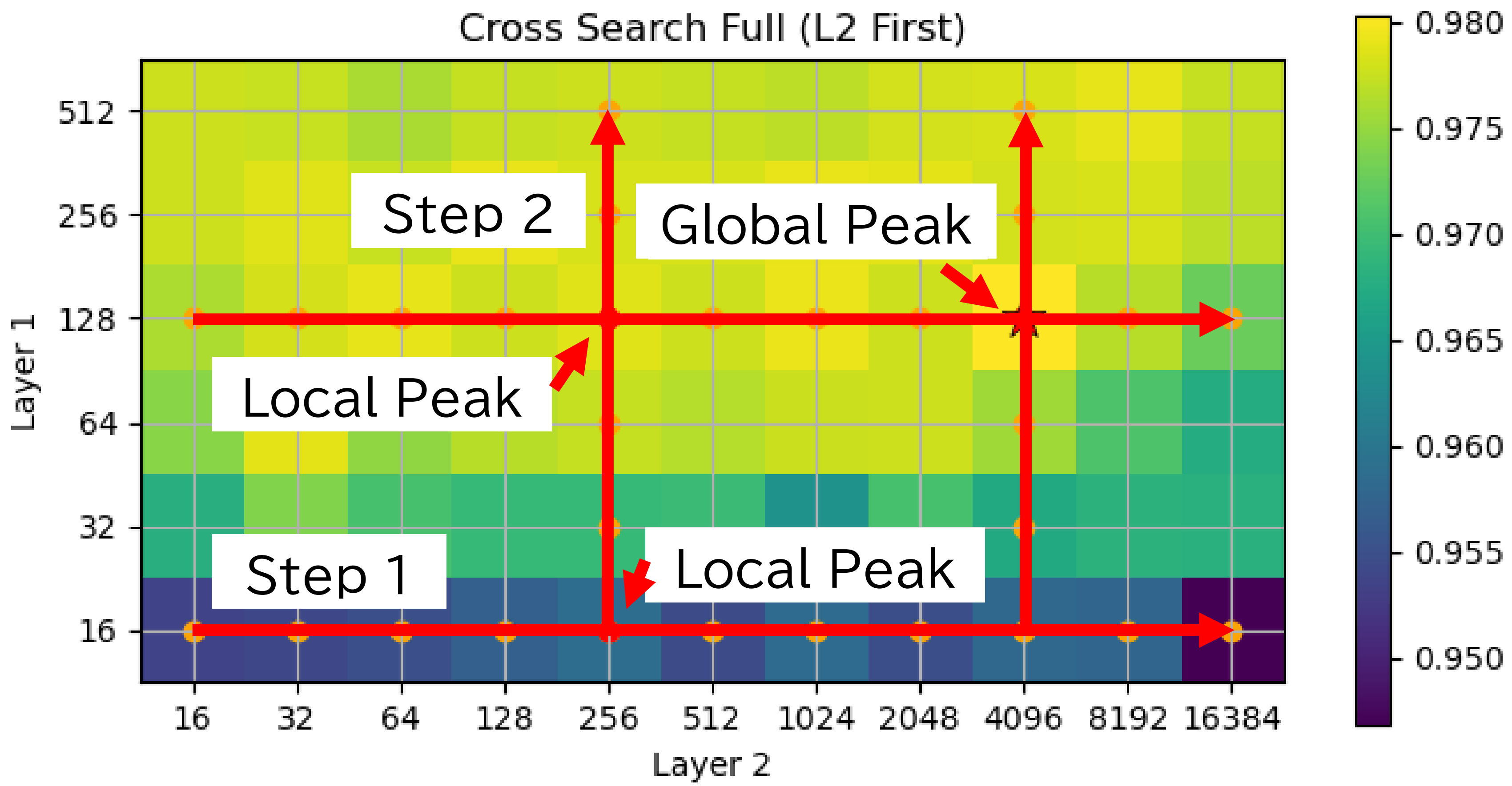
Proposals

- Use of the Logarithmic Spacing Grid
- Use straight forward search method
  - Full Cross-Search Method
  - Partial Cross-Search Method

Logarithmic Spacing Grid



Full Cross-Search Example



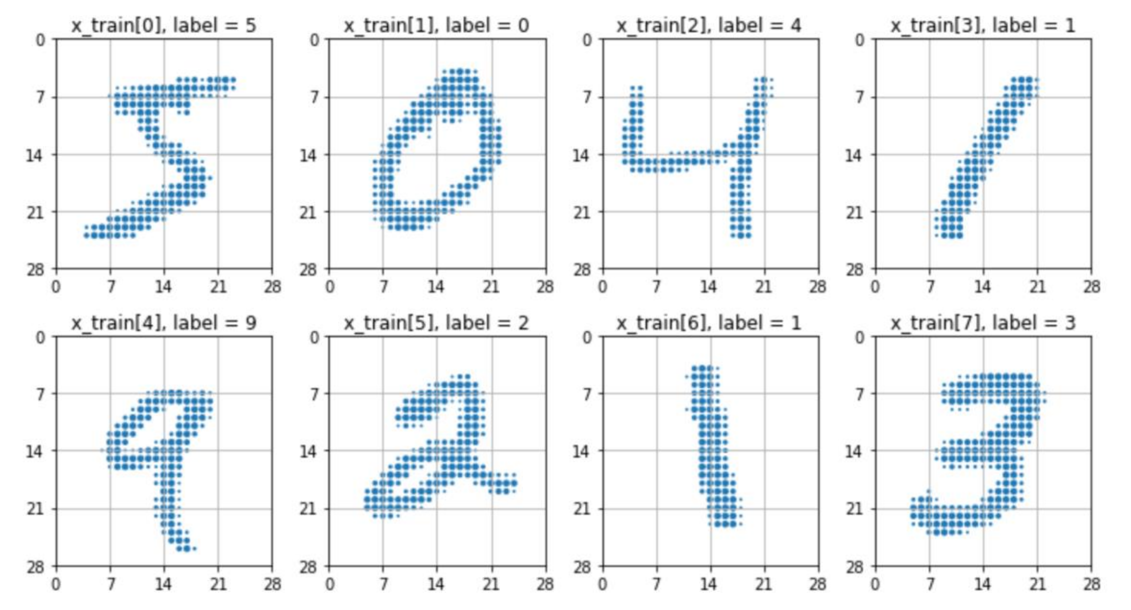
Agenda

- Introduction
- Hyperparameters of CNN
- Hyperparameter Optimization
- Proposal of Cross-Search Method
- Experimental Results
- Conclusion & Future Work
- References

Target CNN

- Application: Hand Written Digits recognition
- Dataset: MNIST
- Input: 28 x 28 x 8bit grayscale image
- Output: 0, 1, 2, ⋯, 9
- 60,000 learning data + 10,000 test data

Part of MNIST Dataset



- Grayscale image
- 28 x 28 bytes
- Data type: uint8

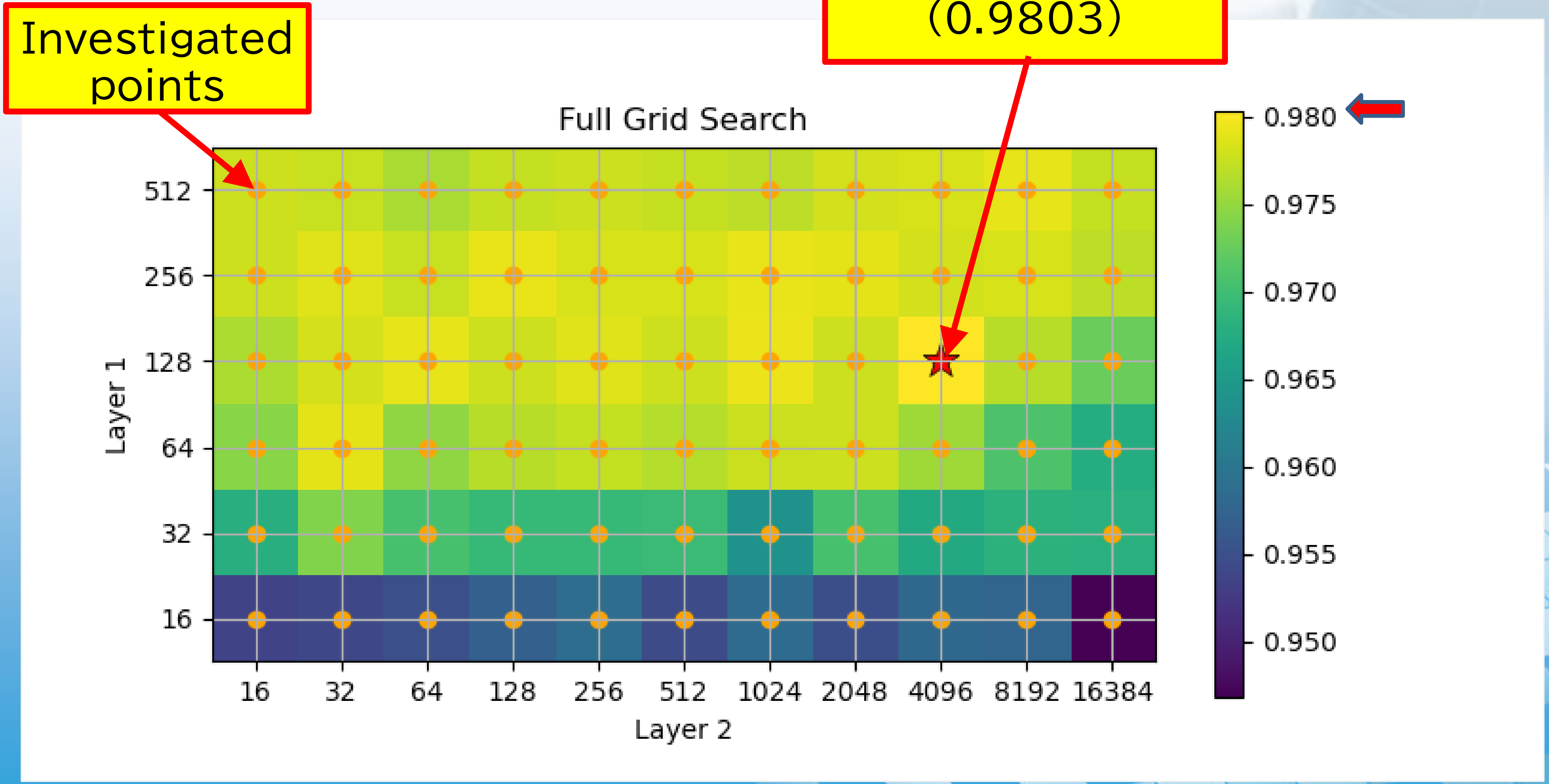
Outline of Optimum Solution

- Known Optimum Solution
  - #neurons in L1 = 128
  - #neurons in L2 = 4,096
  - Accuracy = 0.9803
- Relative Computation Time
  - #of multiplication op's / learning cycle

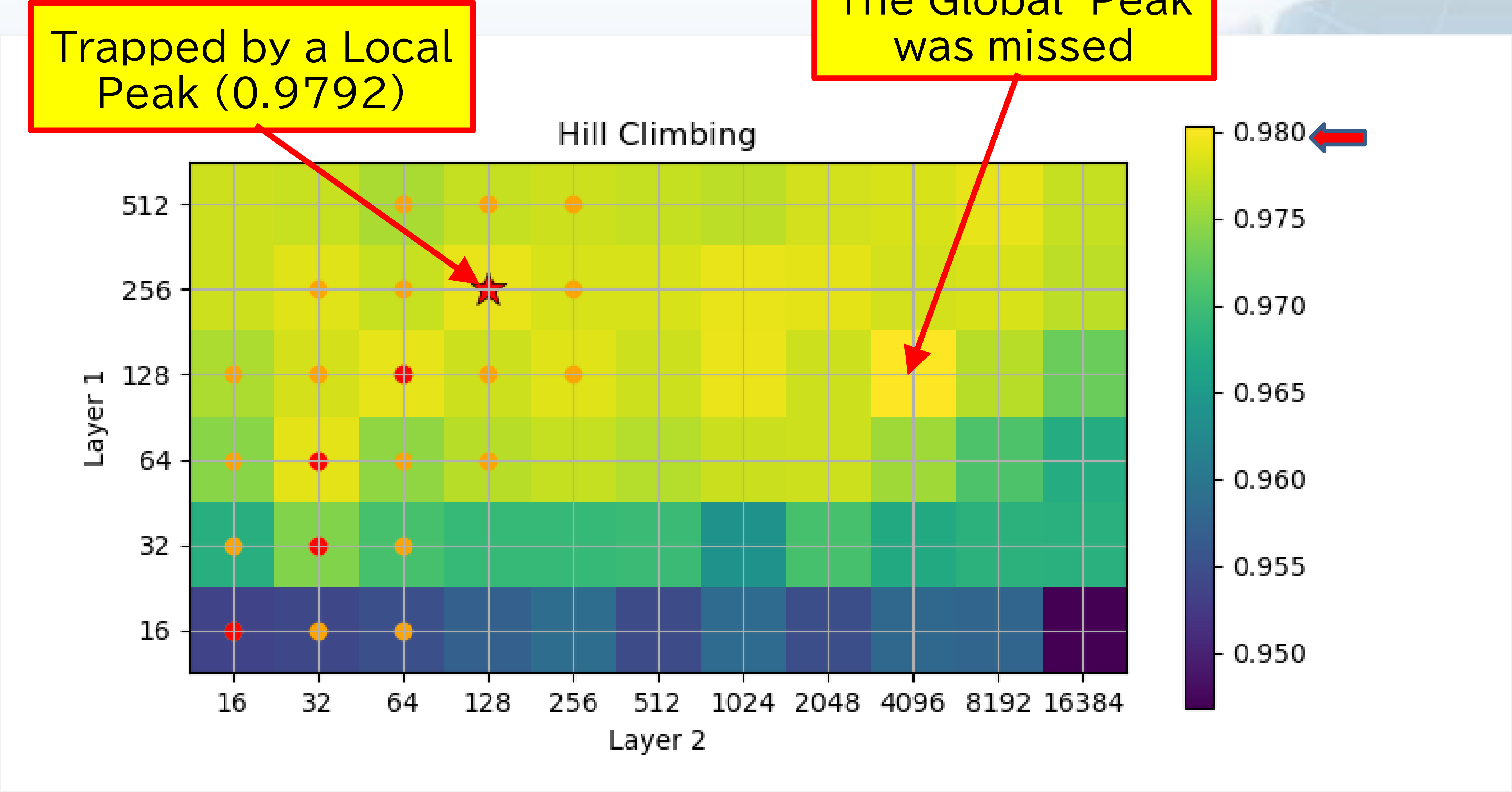
$$T(N_{in}, N_1, N_2, N_{out}) = N_{in} \times N_1 + N_1 \times N_2 + N_2 \times N_{out}$$

  - $T(28 \times 28, 128, 4096, 10) = 43{,}672{,}128$

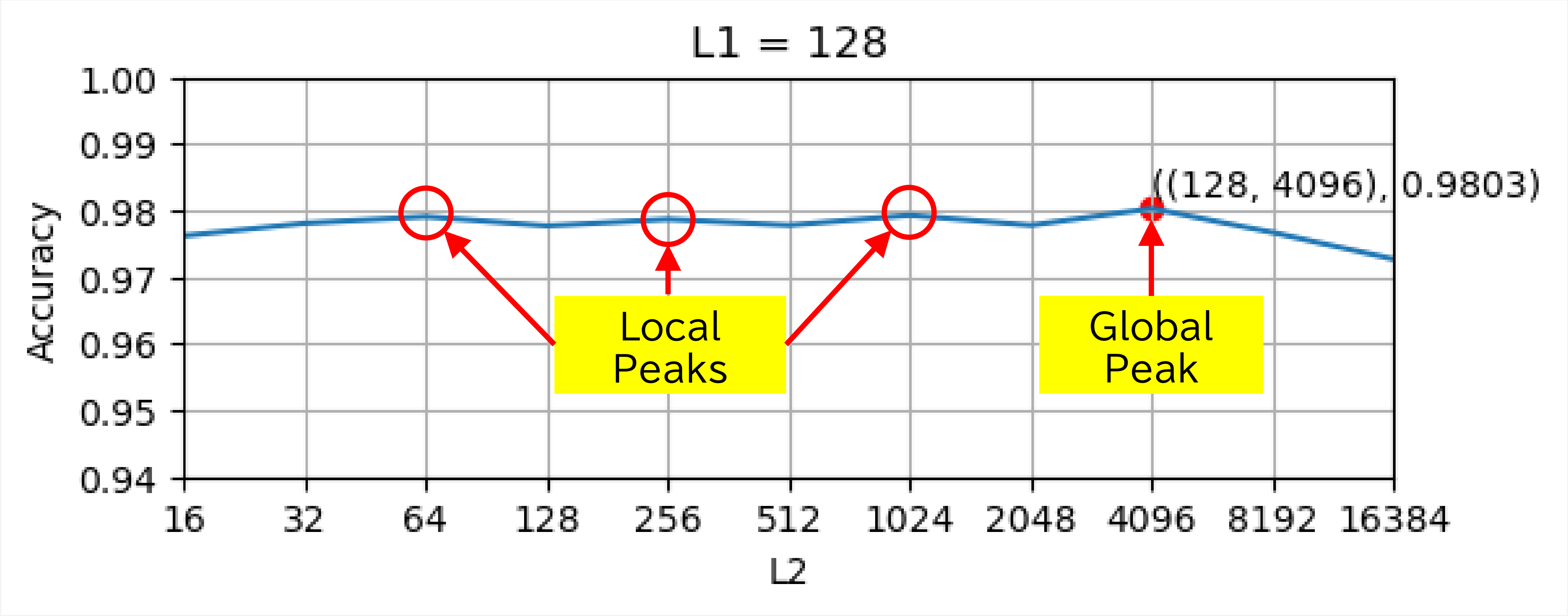
Full Grid Search



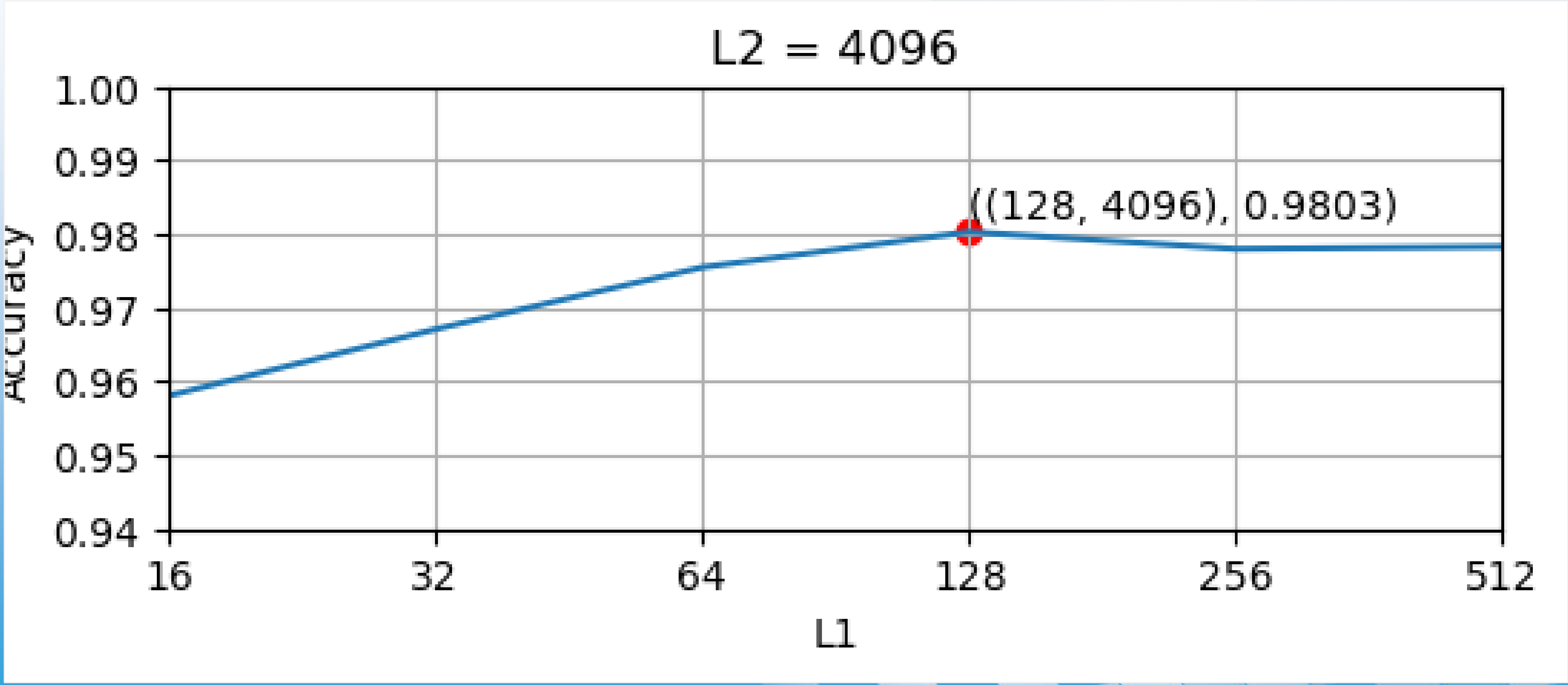
Hill Climbing



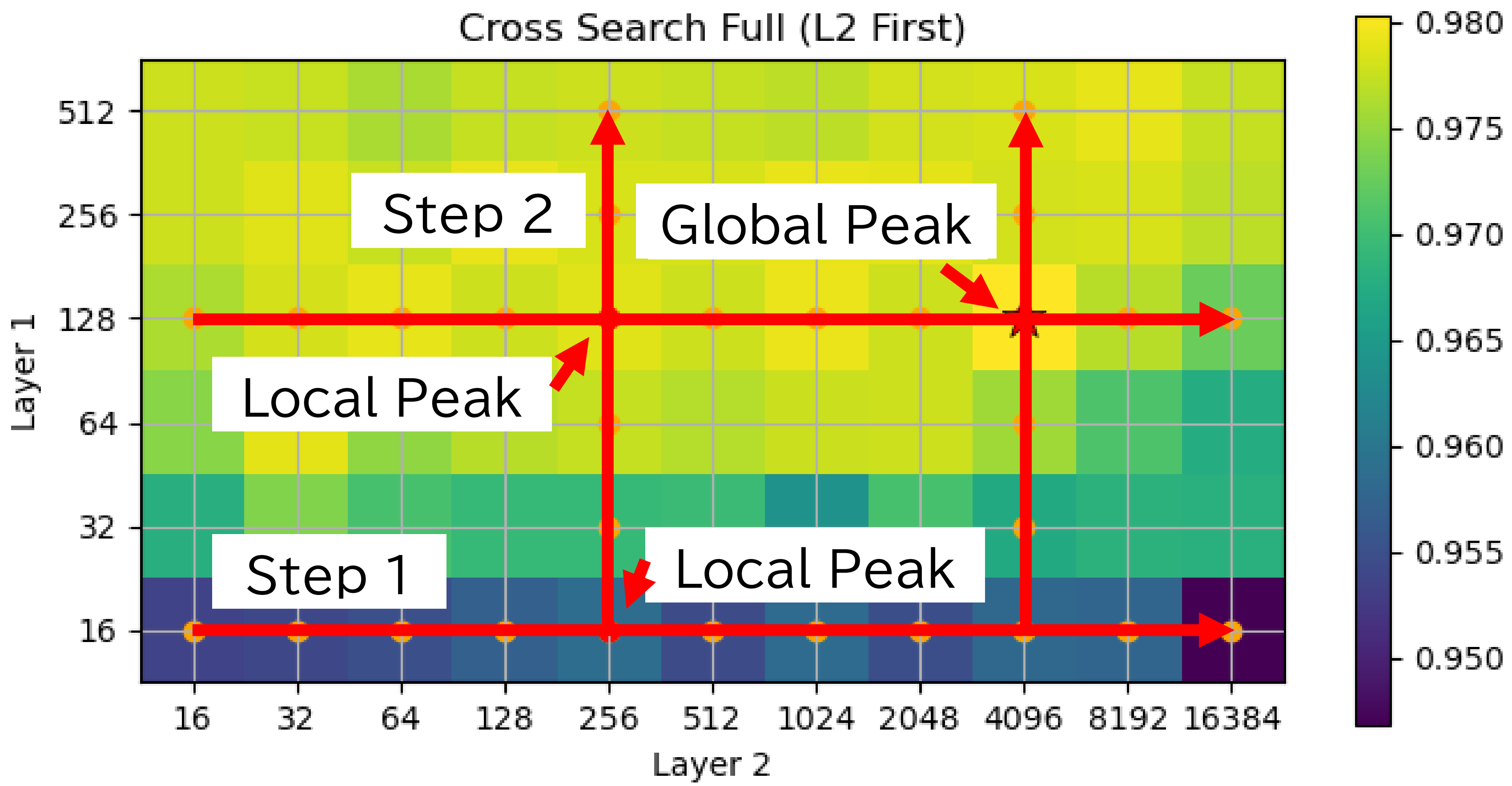
Global and Local peaks



Global Optimum Solution



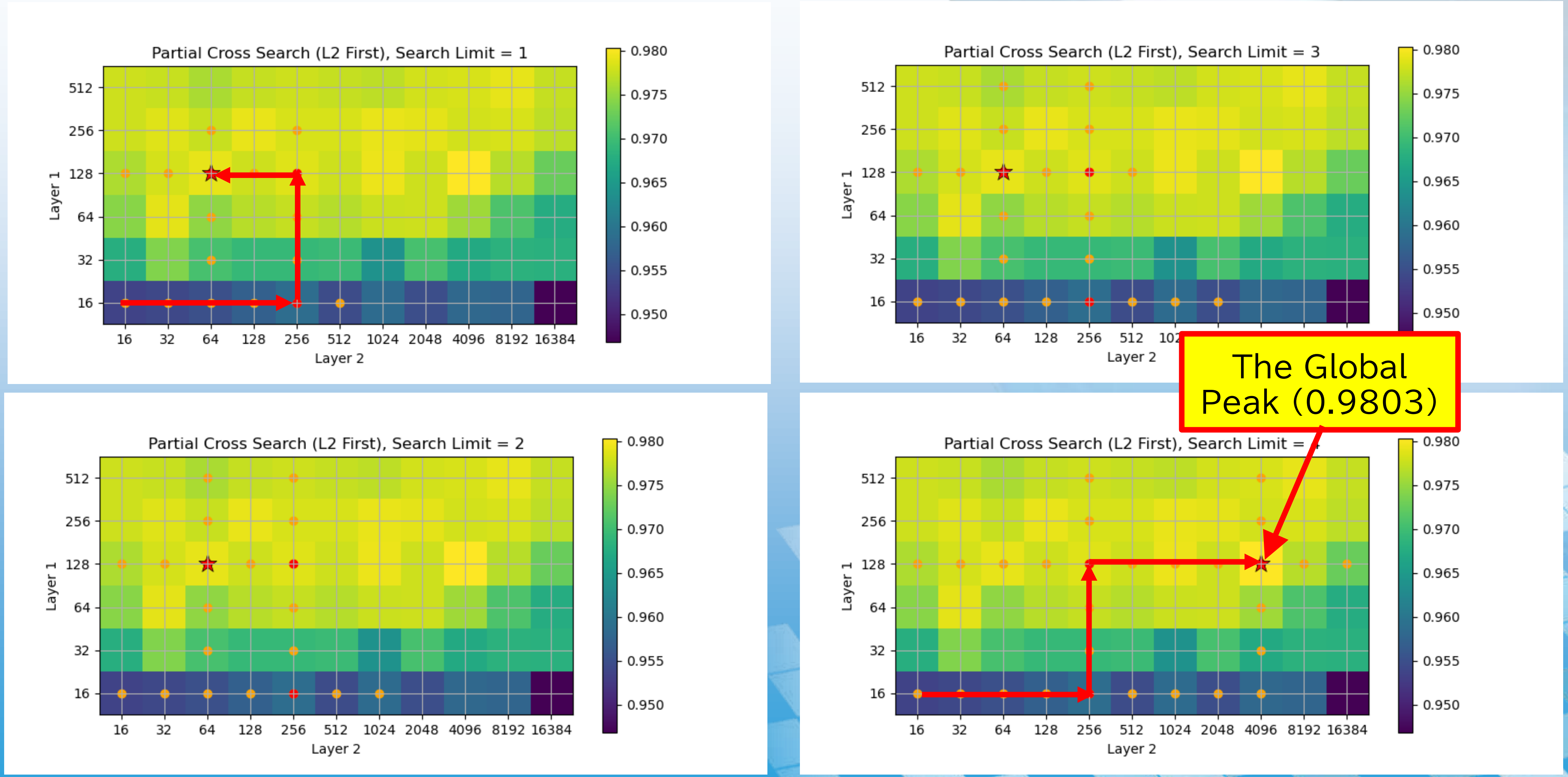
Full Cross-Search Example



Partial Cross-Search

- Parameter "Limit" (= 1, 2, 3, ⋯)
- Continues search Limit more steps after a local peak was found
- As the Limit increases, Partial Cross-Search returns a better solution, but longer computation time is required

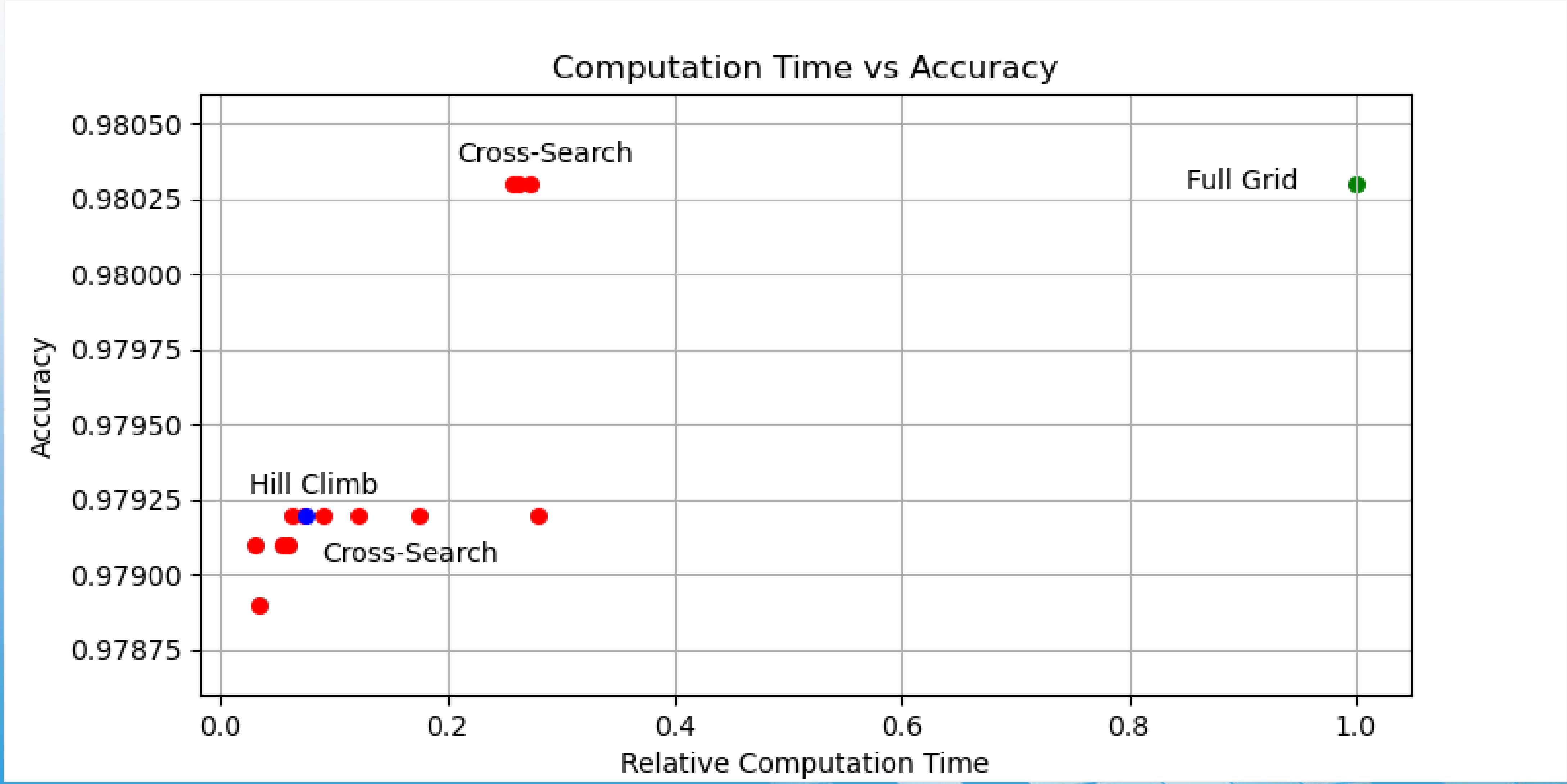
Partial Cross (Limit=1～4)



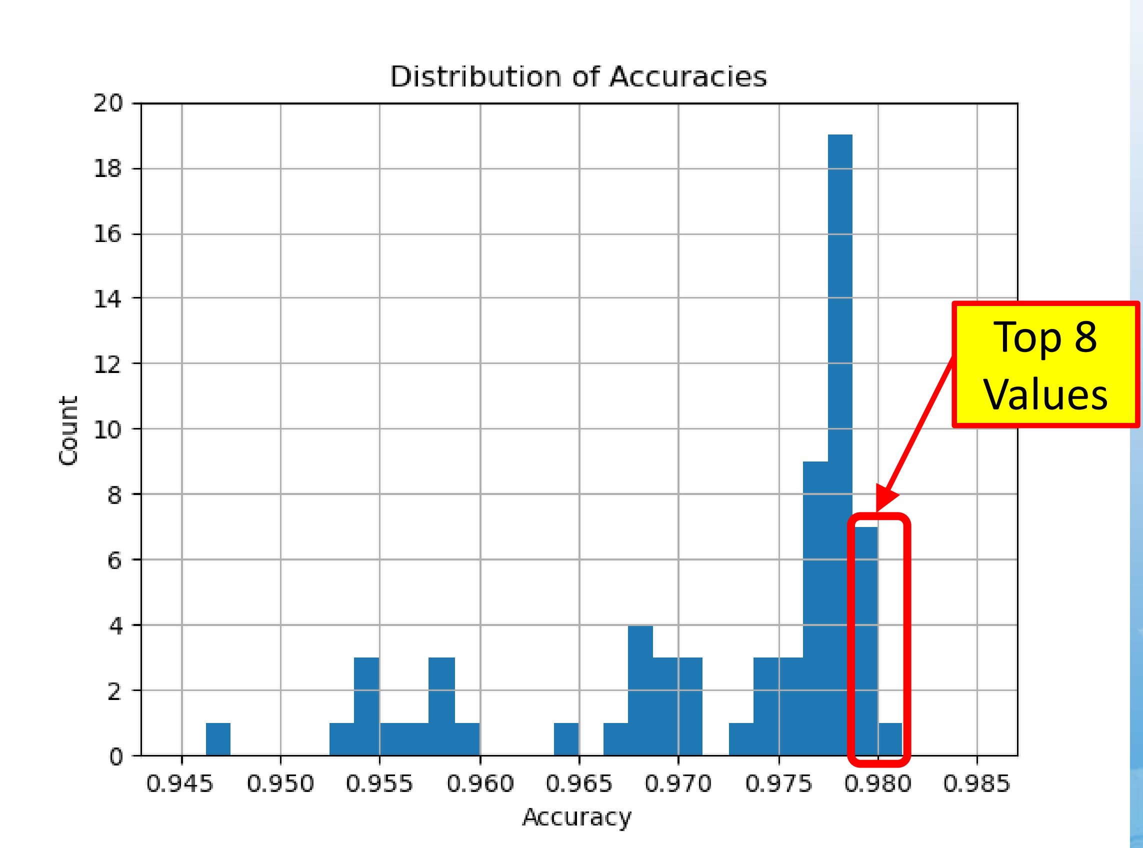
Experimental Results

Search Algorithm	Hyper Param	Accuracy	Diff.	Relative Comp. Time
Full Search	(128, 4096)	0.9803	0.0000	1.0
Hill Climbing	(256, 128)	0.9792	0.0011	0.075
Full Cross (L2) Partial Cross (L2)	(128, 4096)	0.9803	0.0000	0.273 0.257
Full Cross (L1) Partial Cross (L1)	(256, 128)	0.9792	0.0011	0.280 0.063
Partial Cross (L2)	(128, 64)	0.9791	0.0012	0.031
Partial Cross (L)	(64, 32)	0.9789	0.0014	0.034

Computation Time vs Accuracy



Distribution of Accuracy



- Top 8 Acc values
  - 0.9803 ← Full Grid Full Cross
  - 0.9793
  - 0.9792 ← Hill Climb Full Cross
  - 0.9792
  - 0.9791 ← Partial Cross
  - 0.9790
  - 0.9789 ← Partial Cross
  - 0.9789

Agenda

- Introduction
- Hyperparameters of CNN
- Hyperparameter Optimization
- Proposal of Cross-Search Method
- Experimental Results
- Conclusion & Future Work
- References

Conclusion (1)

- Accuracy
  - Accuracy of Full Cross-Search is comparable to Full Grid
  - Accuracy of Partial Cross-Search is comparable to Hill Climb
- Computation Time
  - Full Cross-Search is about 3 times as fast as Full Grid
  - Partial Cross-Search with limit=1 is the fastest, about 2.5 times as fast as Hill Climb
- The accuracy of the solutions obtained by the Partial Cross-Search is high enough
- Partial Cross-Search is <u>scalable and cost-effective</u>

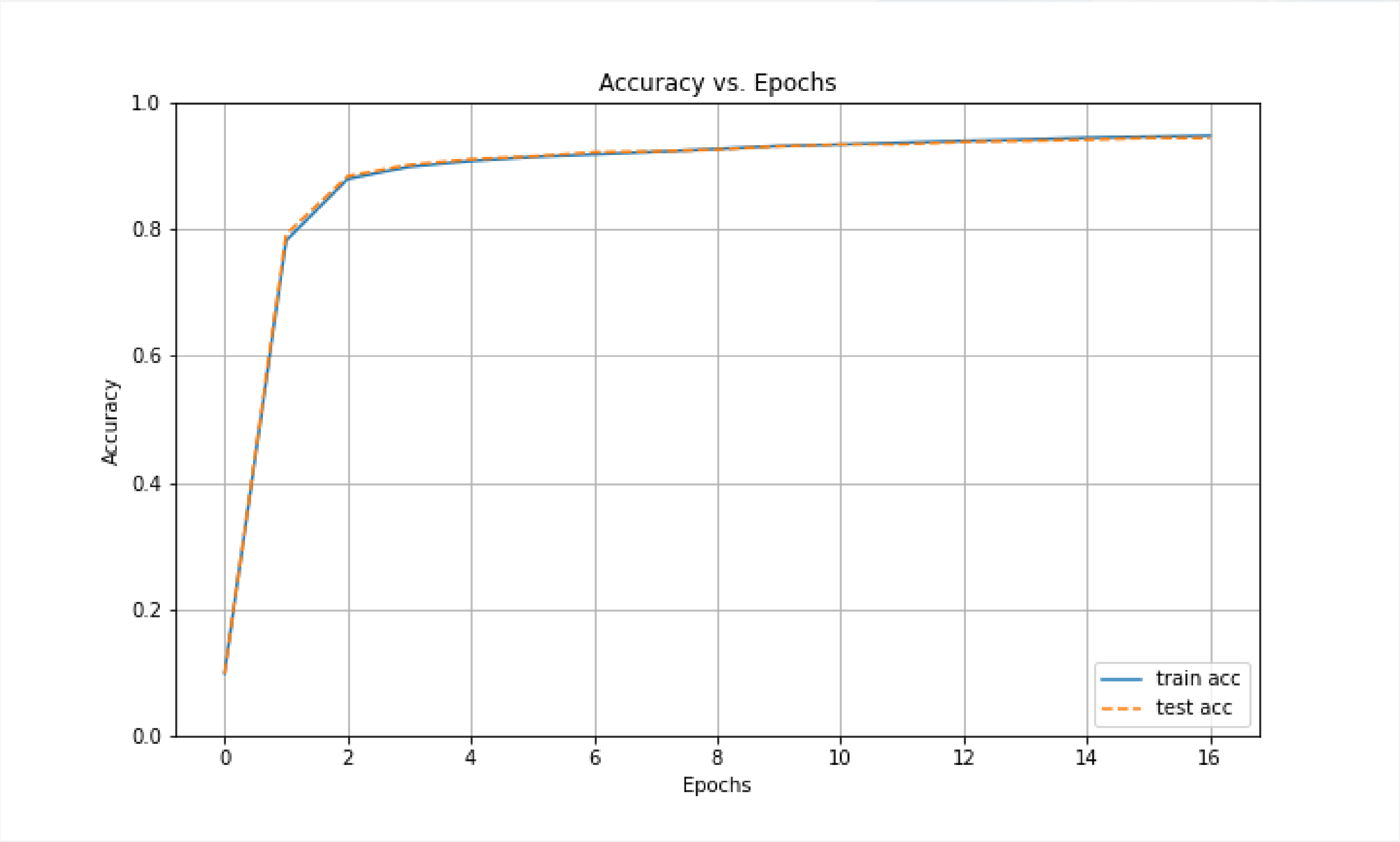
Conclusion (2)

- Full / Partial Cross-Search are suitable for Optimization of HPs with continuous quantity
- Suitability for Parallel Computation
  - Full Grid and Full Cross-Search have good Parallelism
  - Hill Climb Search has limited parallelism (only “neighbors” of candidate combination of HP can be computed in parallel)

Future Work

- Effect of the search Start Point assessment
- Application of Full / Partial Cross to other HPs and larger neural network
- Comparison with other HP optimization methods (Random Search, Bayesian, etc.)
- Validity of learning assessment results (accuracy, loss, etc.) using early learning epochs

Accuracy vs. Epochs



Agenda

- Introduction
- Hyperparameters of CNN
- Hyperparameter Optimization
- Proposal of Cross-Search Method
- Experimental Results
- Conclusion & Future Work
- References

References

- Y. Ozaki, M. Nomura, and M. Onishi, “Hyperparameter Optimization Methods: Overview and Characteristics,” IEICE Trans. D, Vol. J103-D, No. 9, pp.615-631, 2020.
- Yann LeCun, Corinna Cortes, Christopher J.C. Burges, THE MNIST DATABASE of handwritten digits; http://yann.lecun.com/exdb/mnist/