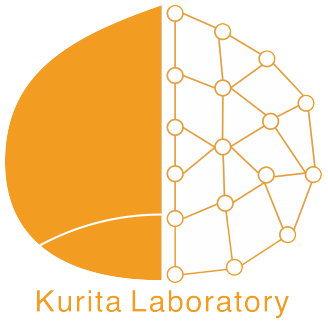
# Channel Planting for Deep Neural Networks using Knowledge Distillation

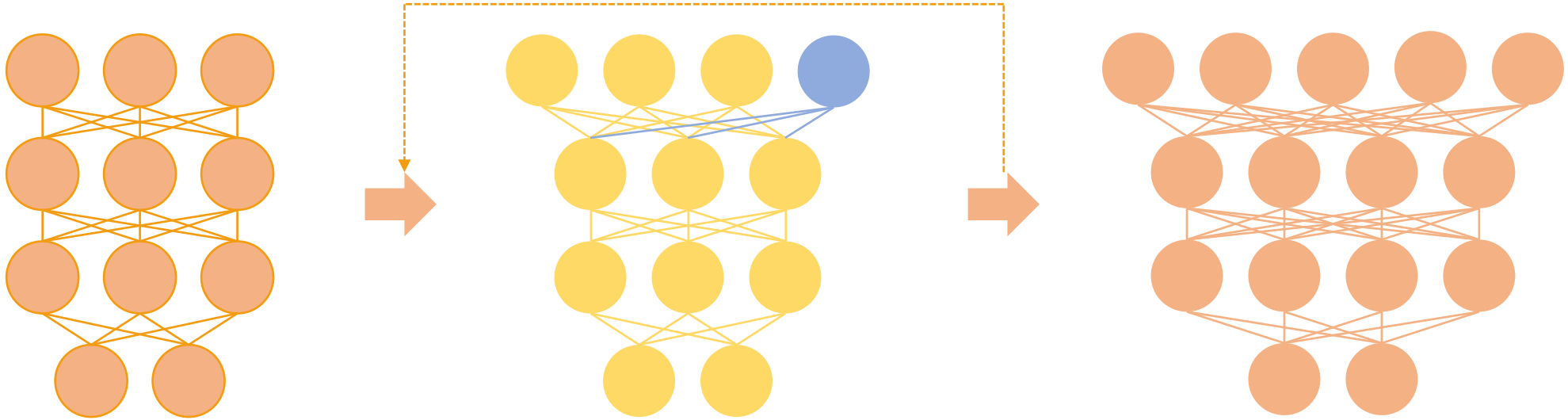
Kakeru Mitsuno, Yuichiro Nomura and Takio Kurita

Hiroshima University, Japan

# Introduction

**Planting** can search the optimal network architecture with smaller number of parameters for improving the network performance by augmenting channels incrementally to layers of the initial networks while keeping the earlier trained parameters fixed



Initial network

Fix the trained weight
Add a channel

Final network

HIROSHIMA UNIVERSITY

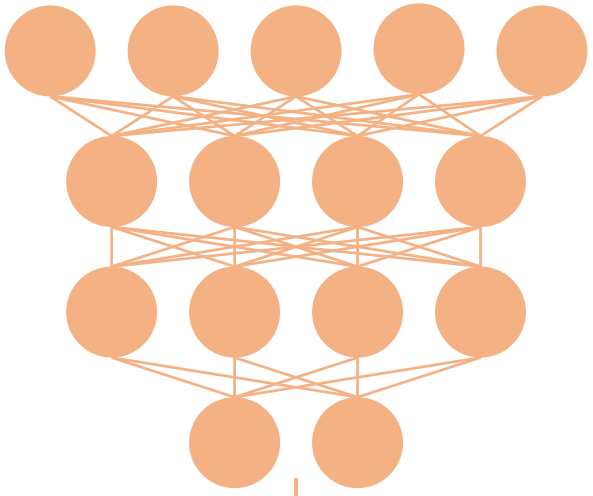
Kurita Laboratory

# Continued…

**Knowledge distillation** can transfer the knowledge of DNNs with a large number of parameter (teacher networks) to a smaller shallow networks (student networks)

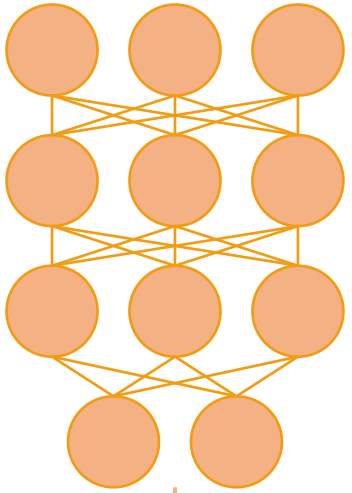
$$\mathbb{L}_{\mathbb{KL}}(z^L||z^S) = \sum_i \frac{\exp z_i^L}{\sum_j \exp z_j^L} log\left(\frac{\exp z_i^S}{\sum_j \exp z_j^S}\right)$$

Teacher Network



$z^L$

Student Network



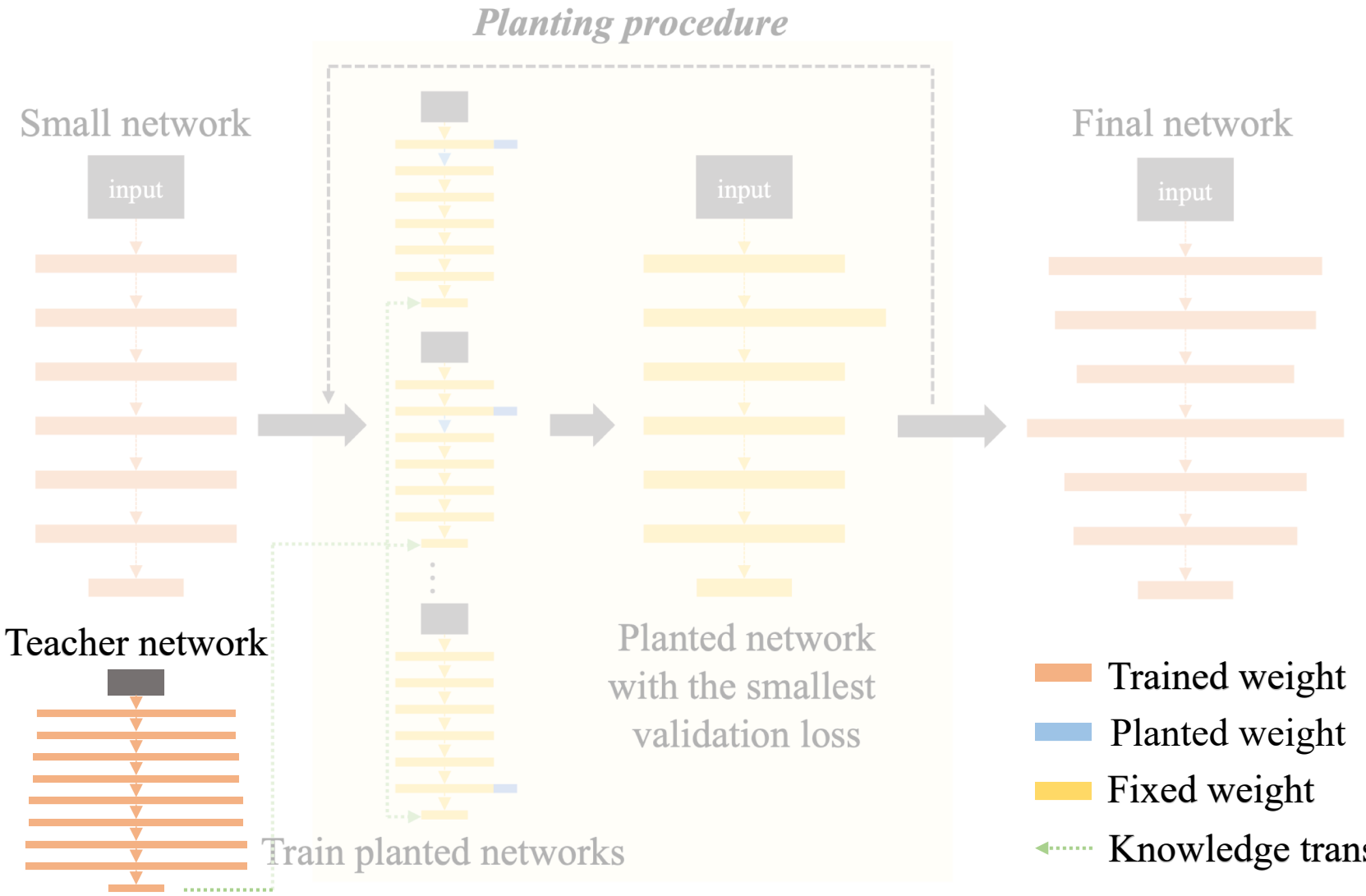
$z^S$

HIROSHIMA UNIVERSITY

Kurita Laboratory

# Proposed Method

1. Train a teacher network.

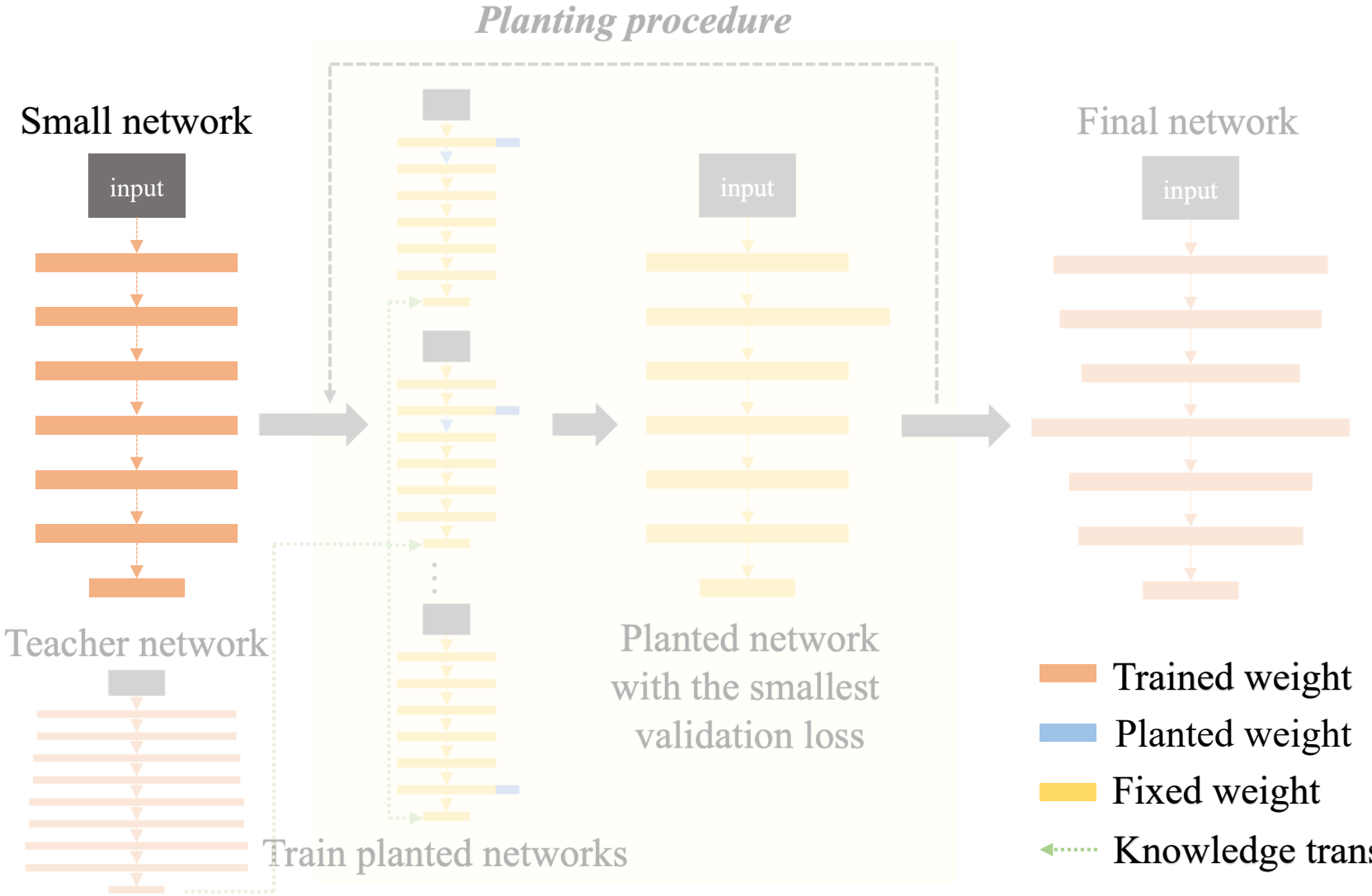



HIROSHIMA UNIVERSITY

Kurita Laboratory

# Continued…

2. Train a small network with fewer channels at each layer

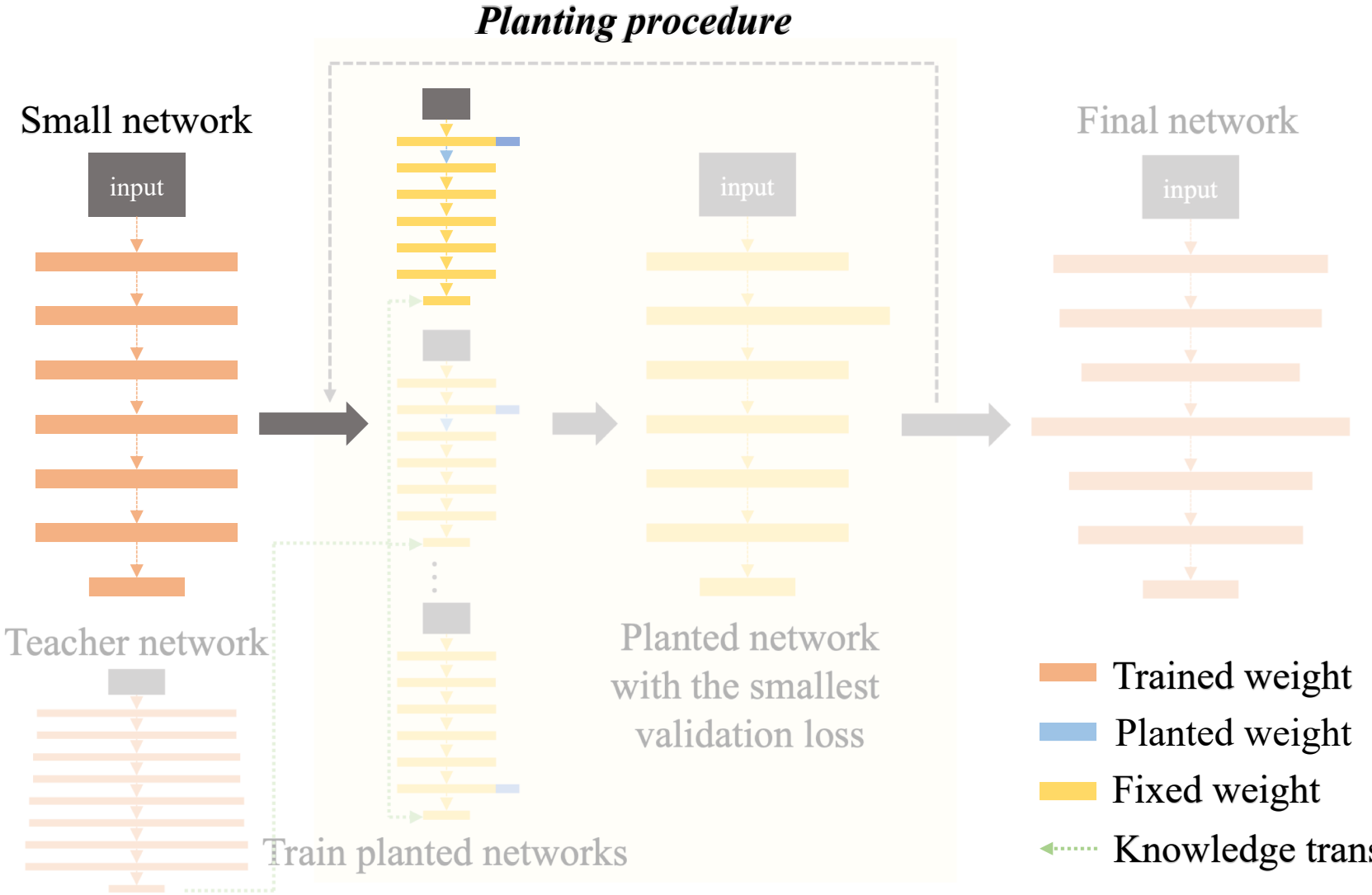



HIROSHIMA UNIVERSITY

Kurita Laboratory

# Continued…

3. Add channels to a layer of the small network

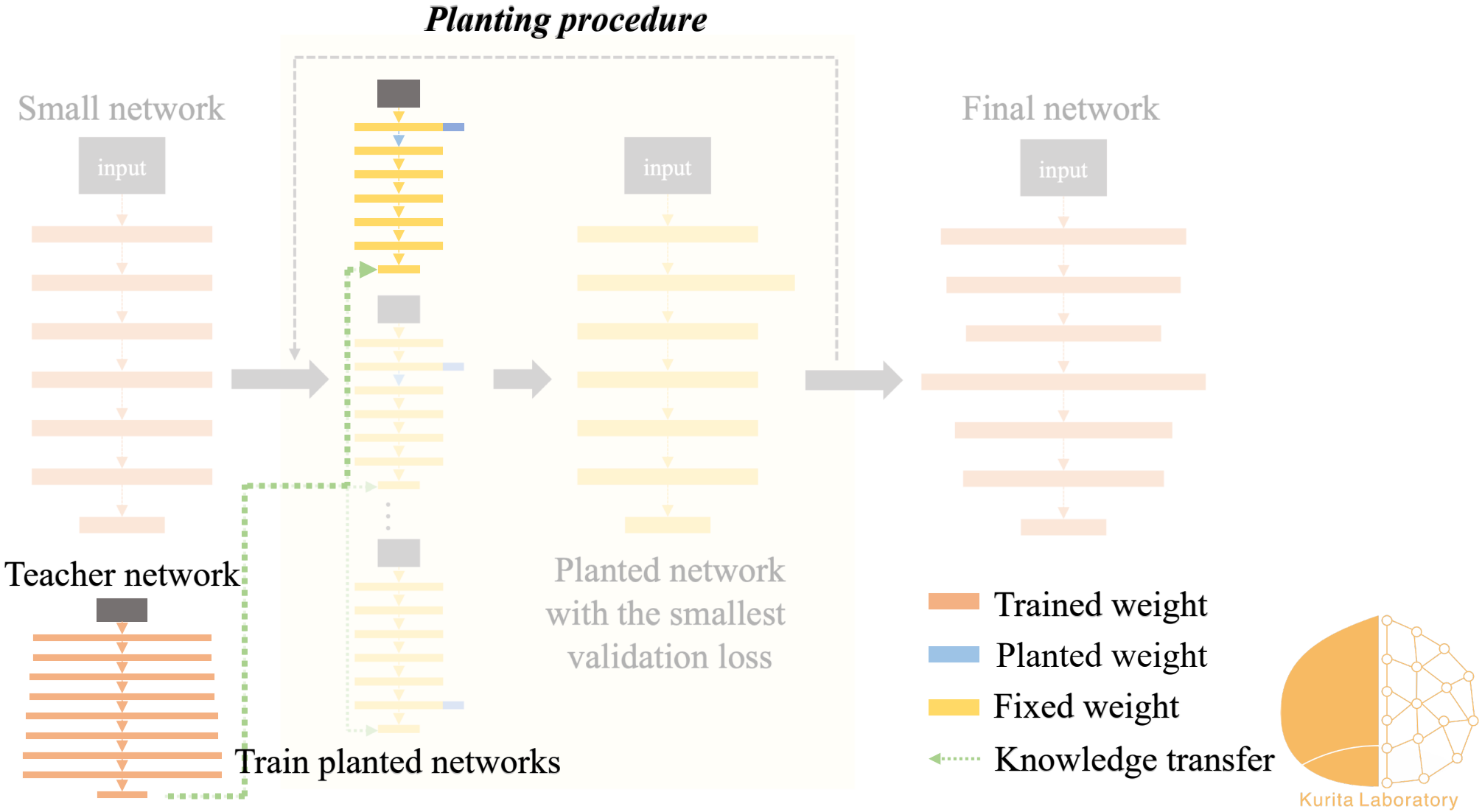



HIROSHIMA UNIVERSITY

Kurita Laboratory

# Continued…

4. Train the augmented channels by using knowledge distillation with the teacher network.

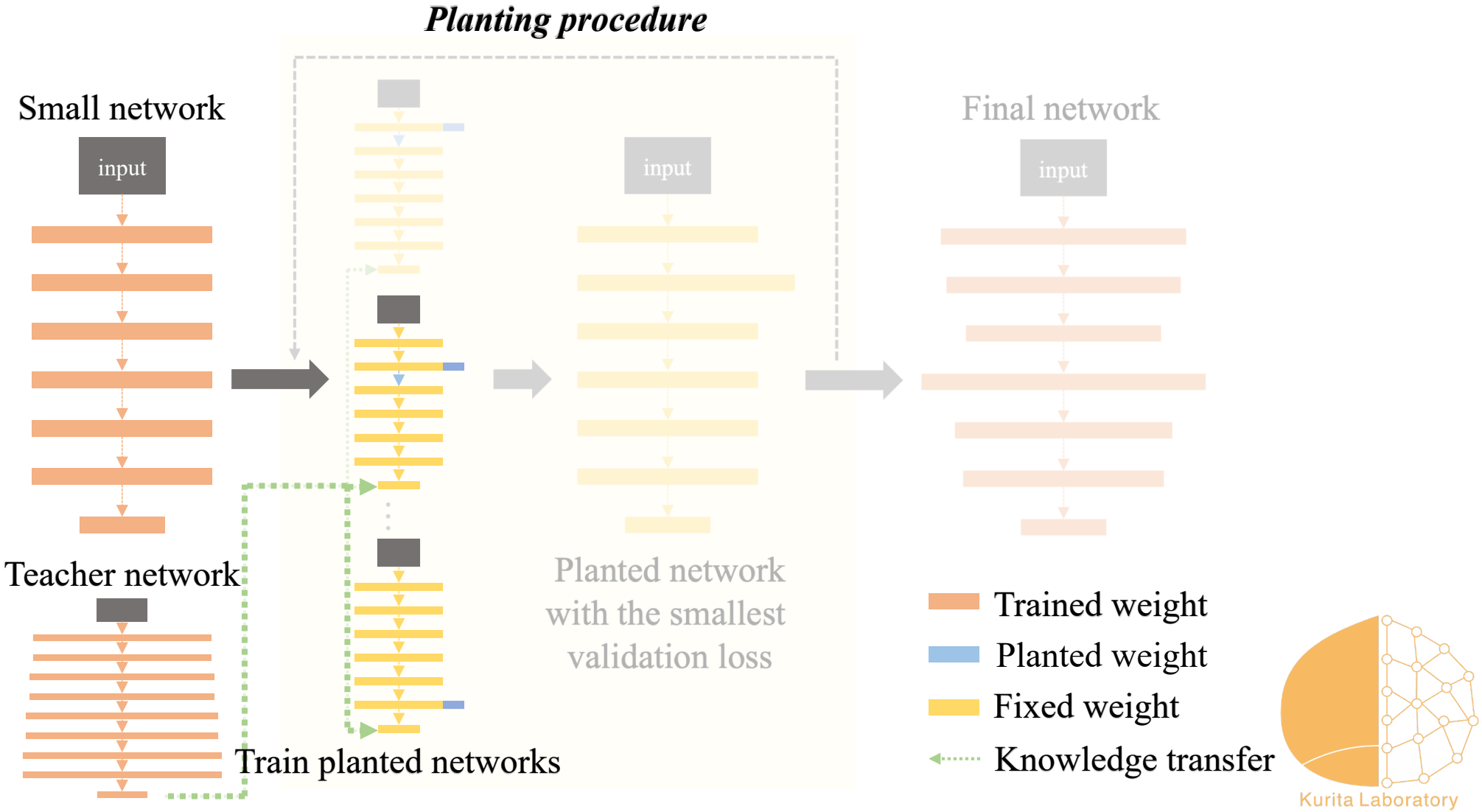



HIROSHIMA UNIVERSITY

Kurita Laboratory

# Continued…

5. Repeat the 3. and 4. operation for each layer in the network and obtain multiple networks.

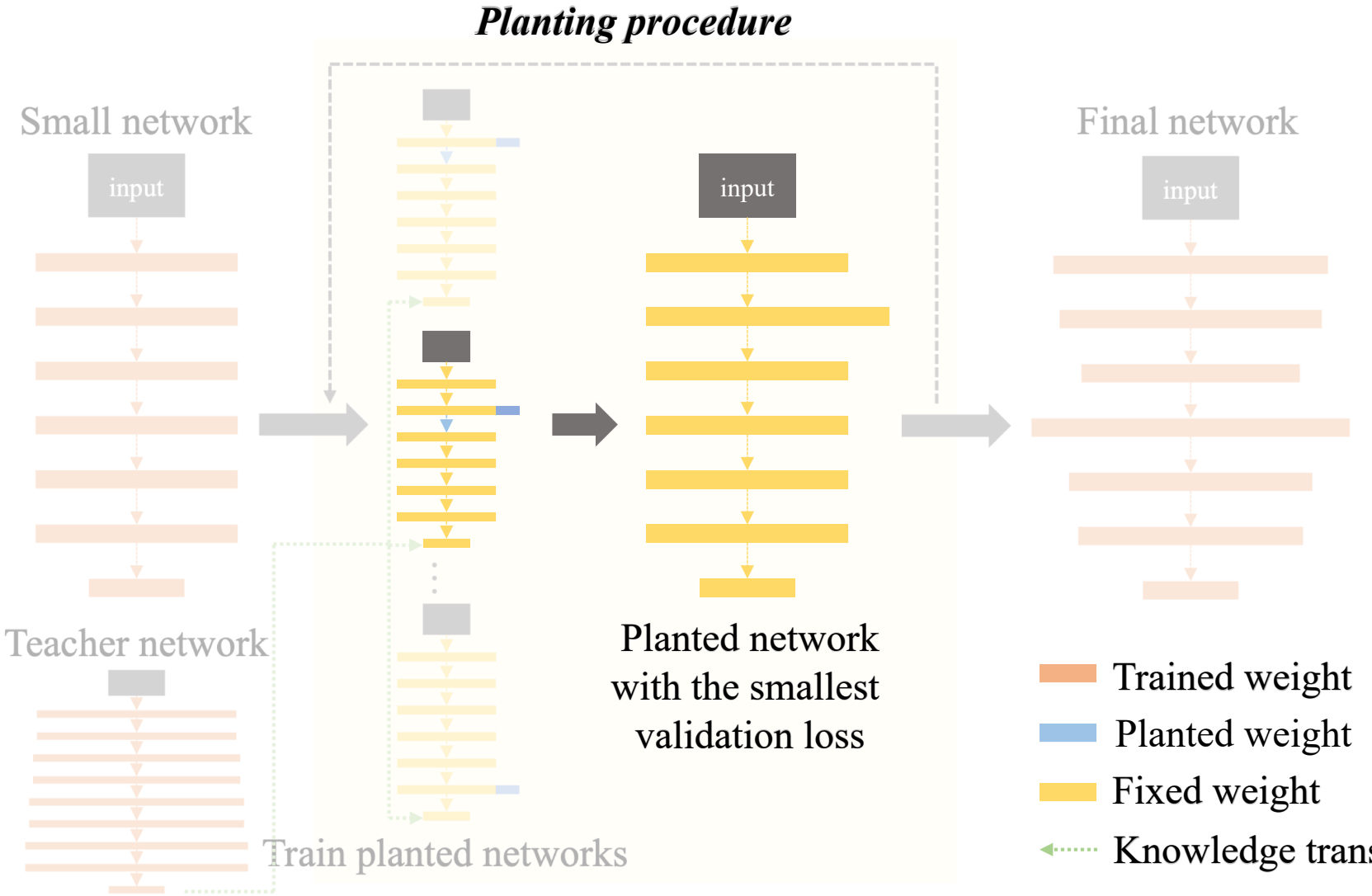



HIROSHIMA UNIVERSITY

Kurita Laboratory

# Continued…

6. Select a planted network with the smallest validation loss

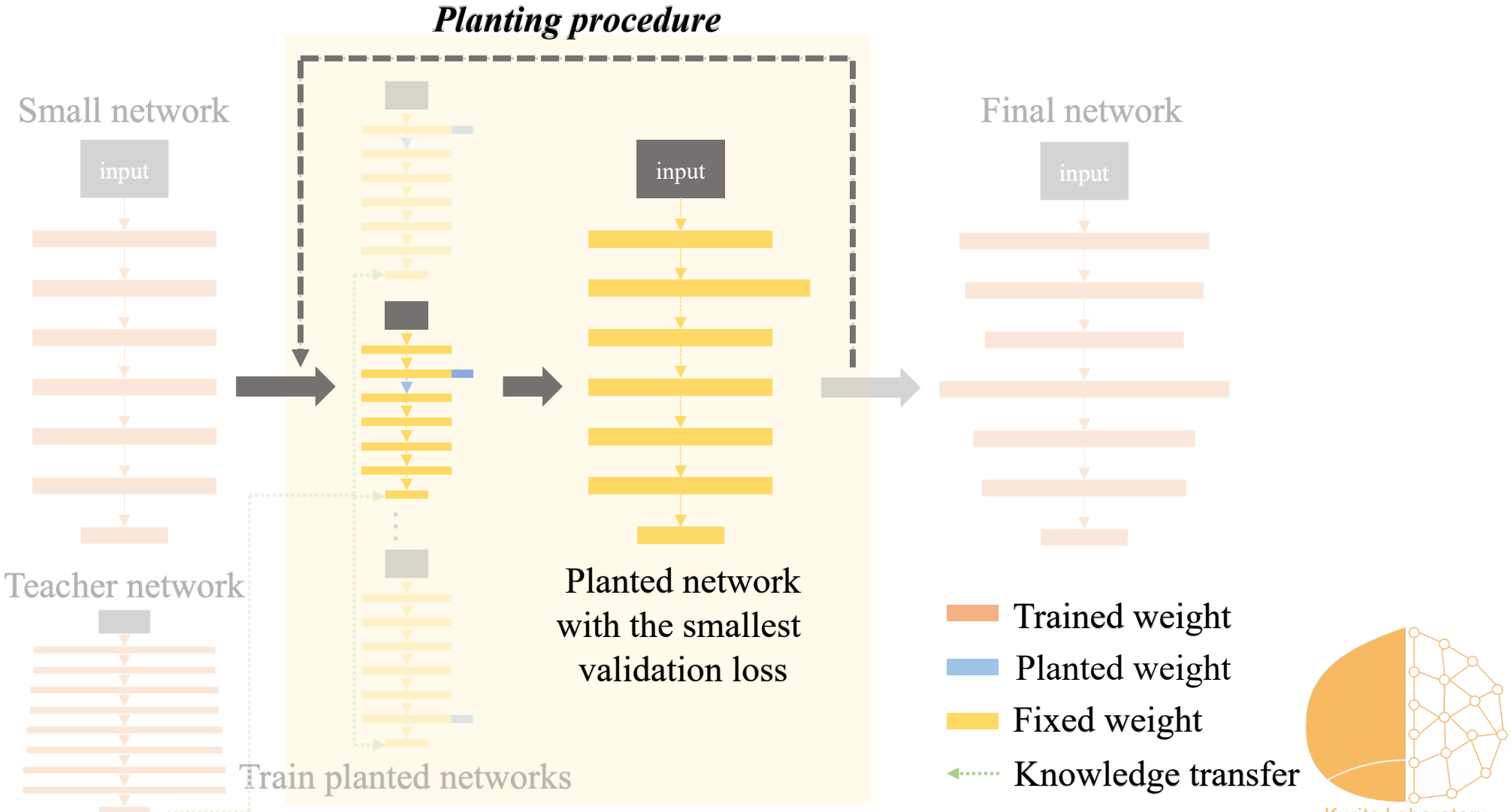



HIROSHIMA UNIVERSITY

Kurita Laboratory

# Continued…

7. Repeat 5. and 6. while reducing the classification loss than the previous network

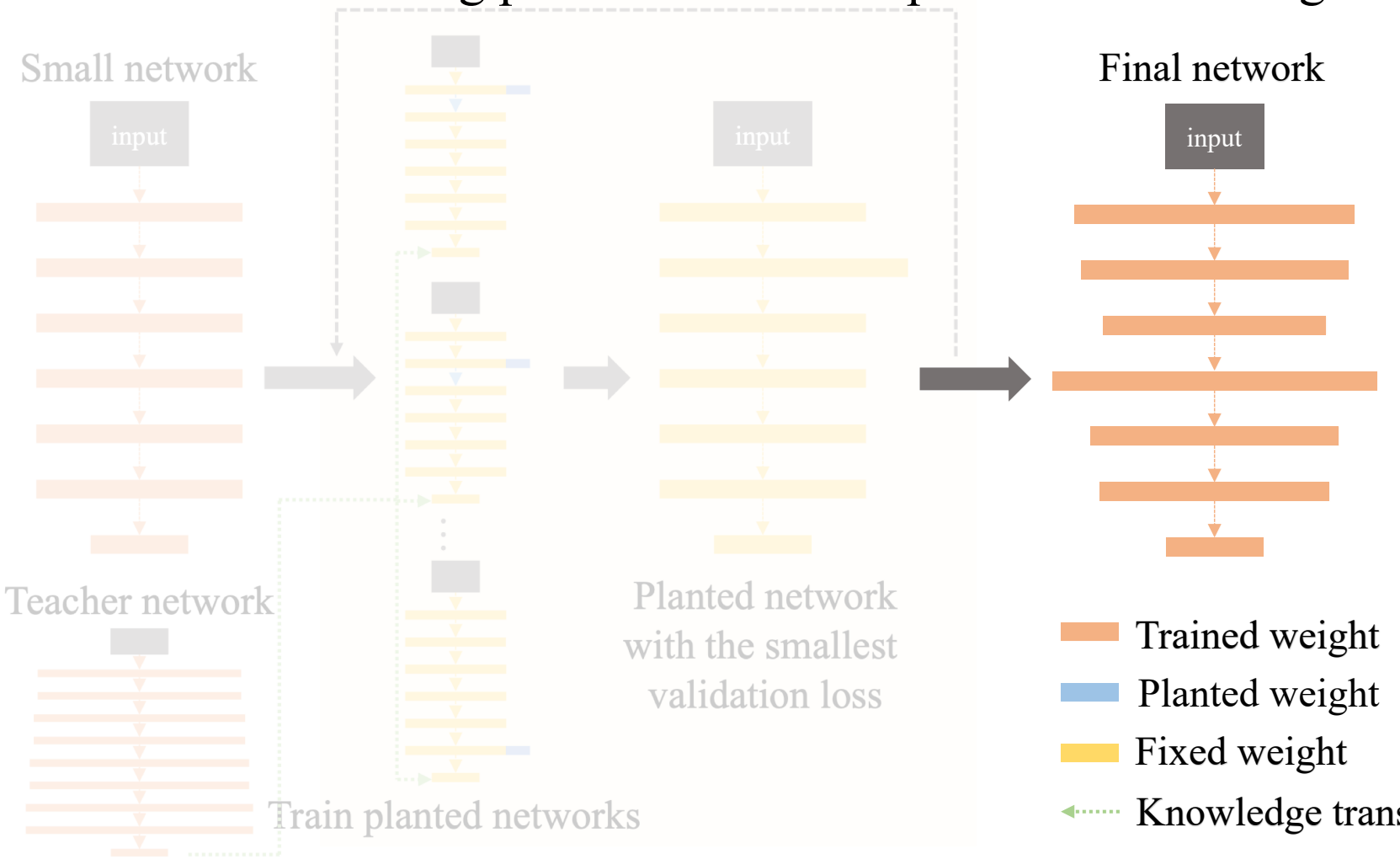



HIROSHIMA UNIVERSITY

Kurita Laboratory

# Continued…

8. Obtaining a small network with fewer channels, which has higher performance than the networks obtained in a standard training procedure and can prevent over-fitting.




HIROSHIMA UNIVERSITY

Kurita Laboratory

# Experiments

## Network architecture and datasets

- CIFAR-10, CIFAR-100 and STL-10

THE STRUCTURE OF NETWORKS

| For CIFAR-10/100 | For STL-10 |
|---|---|
| ReLU(conv1(kernel=3)) | ReLU(conv1(kernel=3)) |
| max pooling(2*2) | max pooling(2*2) |
| ReLU(conv2(kernel=3)) | ReLU(conv2(kernel=3)) |
| max pooling(2*2) | max pooling(2*2) |
| ReLU(conv3(kernel=3)) | ReLU(conv3(kernel=3)) |
| ReLU(conv4(kernel=3)) | max pooling(2*2) |
| ReLU(conv5(kernel=3)) | ReLU(conv4(kernel=3)) |
| max pooling(2*2) | ReLU(conv5(kernel=3)) |
| ReLU(fc1()) | max pooling(2*2) |
| output=fc2() | ReLU(fc1()) |
| | output=fc2() |

# Results

## On CIFAR-10

| Network | Params | Test Err. | Test Acc. | Loss func |
|---|---|---|---|---|
| Teacher[128] | 857.5K | 0.5007 | 88.10% | CELoss |
| Student[128] | | 0.3823 | 88.51% | KLLoss |
| Initial Network (Student[8]) | 20.4K | 0.8300 | 71.55% | CELoss |
| | | 0.8245 | 71.69% | KLLoss |
| Student[16] | 43.9K | 0.6071 | 79.42% | CELoss |
| | | 0.6108 | 79.23% | KLLoss |
| Student[32] | 104.8K | 0.4898 | 84.03% | CELoss |
| | | 0.4791 | 84.02% | KLLoss |
| Student[64] | 282.0K | 0.4431 | 86.83% | CELoss |
| | | 0.4103 | 86.80% | KLLoss |
| Ours | 40.6K | 0.4825 | 84.35% | KLLoss |

# Results

## On CIFAR-100

| Network | Params | Test Err. | Test Acc. | Loss func |
|---|---|---|---|---|
| Teacher[128] | 869.1K | 2.5010 | 57.76% | CELoss |
| Student[128] | | 1.6232 | 60.05% | KLLoss |
| Student[8] | 32.0K | 2.5280 | 36.53% | CELoss |
| | | 2.5053 | 36.90% | KLLoss |
| Initial Network (Student[16]) | 55.5K | 2.1190 | 45.45% | CELoss |
| | | 2.0679 | 46.66% | KLLoss |
| Student[32] | 116.5K | 1.9022 | 52.15% | CELoss |
| | | 1.7805 | 53.72% | KLLoss |
| Student[64] | 293.6K | 1.9510 | 55.74% | CELoss |
| | | 1.6707 | 57.71% | KLLoss |
| Ours | 78.5K | 1.7584 | 54.31% | KLLoss |

# Results

## On STL-10

| Network | Params | Test Err. | Test Acc. | Loss func |
|---|---|---|---|---|
| Teacher[64] | 445.8K | 1.5360 | 66.33% | CELoss |
| Student[64] | | 1.1807 | 66.47% | KLLoss |
| Initial Network (Student[8]) | 40.8K | 1.2776 | 55.55% | CELoss |
| | | 1.2682 | 54.99% | KLLoss |
| Student[16] | 84.9K | 1.2924 | 59.34% | CELoss |
| | | 1.1998 | 61.10% | KLLoss |
| Student[32] | 186.8K | 1.2213 | 64.57% | CELoss |
| | | 1.1712 | 64.07% | KLLoss |
| Student[128] | 1.2M | 1.7612 | 67.04% | CELoss |
| | | 1.1643 | 67.71% | KLLoss |
| Ours | 82.6K | 1.0772 | 67.12% | KLLoss |

# Conclusion

- Proposed a novel incremental training method for DNNs called ***planting***, that can train smaller network with excellent performance and find the optimal network architecture automatically.

- Introduced the **knowledge transfer** to train planted channels.

- We confirmed that the proposed approach was able to achieve comparable performance with smaller number of parameters compared to the larger network.

HIROSHIMA UNIVERSITY

Kurita Laboratory

# Thank you

HIROSHIMA UNIVERSITY

Kurita Laboratory