

Paper Review

# **EfficientAD: Accurate Visual Anomaly Detection at Millisecond-Level Latencies**

**YeongHyeon Park**

Department of Electrical and Computer Engineering

SungKyunKwan University



# **EfficientAD: Accurate Visual Anomaly Detection at Millisecond-Level Latencies**

Kilian Batzner<sup>1</sup>

Lars Heckler<sup>1,2</sup>

Rebecca König<sup>1</sup>

<sup>1</sup>MVTec Software GmbH, <sup>2</sup>Technical University of Munich

{kilian.batzner, lars.heckler, rebecca.koenig}@mvtec.com



**Thursday, January 4**

Thursday, January 4

**1300–1700 Registration** (Naupaka Terrace)**1345–1400 Exhibits & Demos** (Paniolo & Paniolo Terrace)**1345–1400 Refreshments** (Paniolo)**1400–1430 Welcome & Paper Awards** (Naupaka)**1430–1545 Orals 1.1: Machine Learning: Architectures, Formulations, Algorithms** (Naupaka)

Paper number represents poster number in today's poster session.

Format (10 min. presentation; 2 min. questions)

1. [1430] Asymmetric Image Retrieval With Cross Model Compatible Ensembles, *Alon Shoshan, Ori Linial, Nadav Honker, Elad Hirsch, Lior Zamir, Igor Kviatkovsky, Gérard Medioni*
  2. [1442] Cross-Feature Contrastive Loss for Decentralized Deep Learning on Heterogeneous Data, *Sai Aparna Aketi, Kaushik Roy*
  3. [1454] Learning Generalizable Perceptual Representations for Data-Efficient No-Reference Image Quality Assessment, *Suhash Srinath, Shankhanil Mitra, Shikha Rao, Rajiv Soundararajan*
  4. [1506] Location-Aware Self-Supervised Transformers for Semantic Segmentation, *Mathilde Caron, Neil Houlsby, Cordelia Schmid*
  5. [1518] Robust Feature Learning and Global Variance-Driven Classifier Alignment for Long-Tail Class Incremental Learning, *Jayateja Kalla, Soma Biswas*
  6. [1530] Wino Vidi Vici: Conquering Numerical Instability of 8-Bit Winograd Convolution for Accurate Inference Acceleration on Edge, *Pierpaolo Mori, Lukas Frickenstein, Shambhavi Balamuthu Sampath, Moritz Thoma, Nael Fasfous, Manoj Rohit Vempala, Alexander Frickenstein, Christian Unger, Walter Stechele, Daniel Mueller-Gritschneider, Claudio Passerone*
- Format (Virtual papers: see the WACV 2024 online interface)
- Data-Centric Debugging: Mitigating Model Failures via Targeted Image Retrieval, *Sahil Singla, Atoosa Malemir Chegini, Mazda Moayeri, Soheil Feizi*
  - Distortion-Disentangled Contrastive Learning, *Jinfeng Wang, Sifan Song, Jionglong Su, S. Kevin Zhou*
  - GTP-ViT: Efficient Vision Transformers via Graph-Based Token Propagation, *Xuwei Xu, Sen Wang, Yudong Chen, Yanping Zheng, Zhewei Wei, Jiajun Liu*
  - SequenceMatch: Revisiting the Design of Weak-Strong Augmentations for Semi-Supervised Learning, *Khanh-Binh Nguyen*
  - Stochastic Binary Network for Universal Domain Adaptation, *Saurabh Kumar Jain, Sukhendu Das*
  - PECoP: Parameter Efficient Continual Pretraining for Action Quality Assessment, *Aminrhosein Dadashzadeh, Shuchao Duan, Alan Whone, Majid Mirmehdi*

**1545–1600 Exhibits & Demos** (Paniolo & Paniolo Terrace)**1545–1600 Break** (Paniolo)**Program****1600–1700 Plenary 1** (Naupaka)**Keynote:** Dima Damen (*Bristol University*)**Title:** Opportunities in Egocentric Video Understanding**Abstract:** Forecasting the rise of wearable devices equipped with audio-visual feeds, this talk will present opportunities for research in egocentric video understanding. This talk aims to argue for a fine(r)-grained perspective on human-object interactions, from video sequences, captured in an egocentric perspective (i.e. first-person footage). I will argue for the need to tackle new approaches of supervision [CVPR 2023, ongoing], object transformations [NeurIPS 2022, ongoing], domain adaptation [ICCV 2023], modalities [ICASSP 2023], and reconstruction [NeurIPS 2023] as we attempt to learn from longer videos captured in an unscripted manner. Additionally, I'll review well-known (EPIC-KITCHENS, Ego4D) and new (Ego-Exo4D) resources for Egocentric Videos.**1700–1830 Orals 1.2: Image/Video Recognition & Understanding; Low-Level & Physics-Based Vision** (Naupaka)

Paper number represents poster number in today's poster session.

Format (10 min. presentation; 2 min. questions)

7. [1700] EfficientAD: Accurate Visual Anomaly Detection at Millisecond-Level Latencies, *Kilian Batzner, Lars Heckler, Rebecca König*
  8. [1712] Efficient Semantic Matching With Hypercolumn Correlation, *Seungwook Kim, Juhong Min, Minsu Cho*
  9. [1724] ArcGeo: Localizing Limited Field-of-View Images Using Cross-View Matching, *Maxim Shugaev, Ilya Semenov, Kyle Ashley, Michael Klaczynski, Naresh Cuntoor, Mun Wai Lee, Nathan Jacobs*
  10. [1736] Contextual Affinity Distillation for Image Anomaly Detection, *Jie Zhang, Masanori Suganuma, Takayuki Okatani*
  11. [1748] Offline-to-Online Knowledge Distillation for Video Instance Segmentation, *Hojin Kim, Seunghun Lee, Hyeon Kang, Sunghoon Im*
  12. [1800] Video-kMaX: A Simple Unified Approach for Online and Near-Online Video Panoptic Segmentation, *Inkyu Shin, Dahyun Kim, Qihang Yu, Jun Xie, Hong-Seok Kim, Bradley Green, In So Kweon, Kuk-Jin Yoon, Liang-Chieh Chen*
  13. [1812] Conditional Velocity Score Estimation for Image Restoration, *Ziqiang Shi, Ruijie Liu*
  14. [1824] ARNIQA: Learning Distortion Manifold for Image Quality Assessment, *Lorenzo Agnolucci, Leonardo Galteri, Marco Bertini, Alberto Del Bimbo*
- Format (Virtual papers: see the WACV 2024 online interface)
- Hard Sample-Aware Consistency for Low-Resolution Facial Expression Recognition, *Bokyung Lee, Kyungdeuk Ko, Jonghwan Hong, Hanseok Ko*
  - Open-Set Object Detection by Aligning Known Class Representations, *Hiran Sarkar, Vishal Chudasama, Naoyuki Onoe, Pankaj Wasnik, Vineeth N. Balasubramanian*
  - DeVos: Flow-Guided Deformable Transformer for Video Object Segmentation, *Volodymyr Fedynyak, Yaroslav Romanus, Bohdan Hlavatskyi, Bohdan Sydor, Oles Dobosevych, Igor Babin, Roman Razantsev*
  - Disentangled Pre-Training for Image Matting, *Yanda Li, Zilong Huang, Gang Yu, Ling Chen, Yunchao Wei, Jianbo Jiao*

# **Summary of EfficientAD**

# EfficientAD

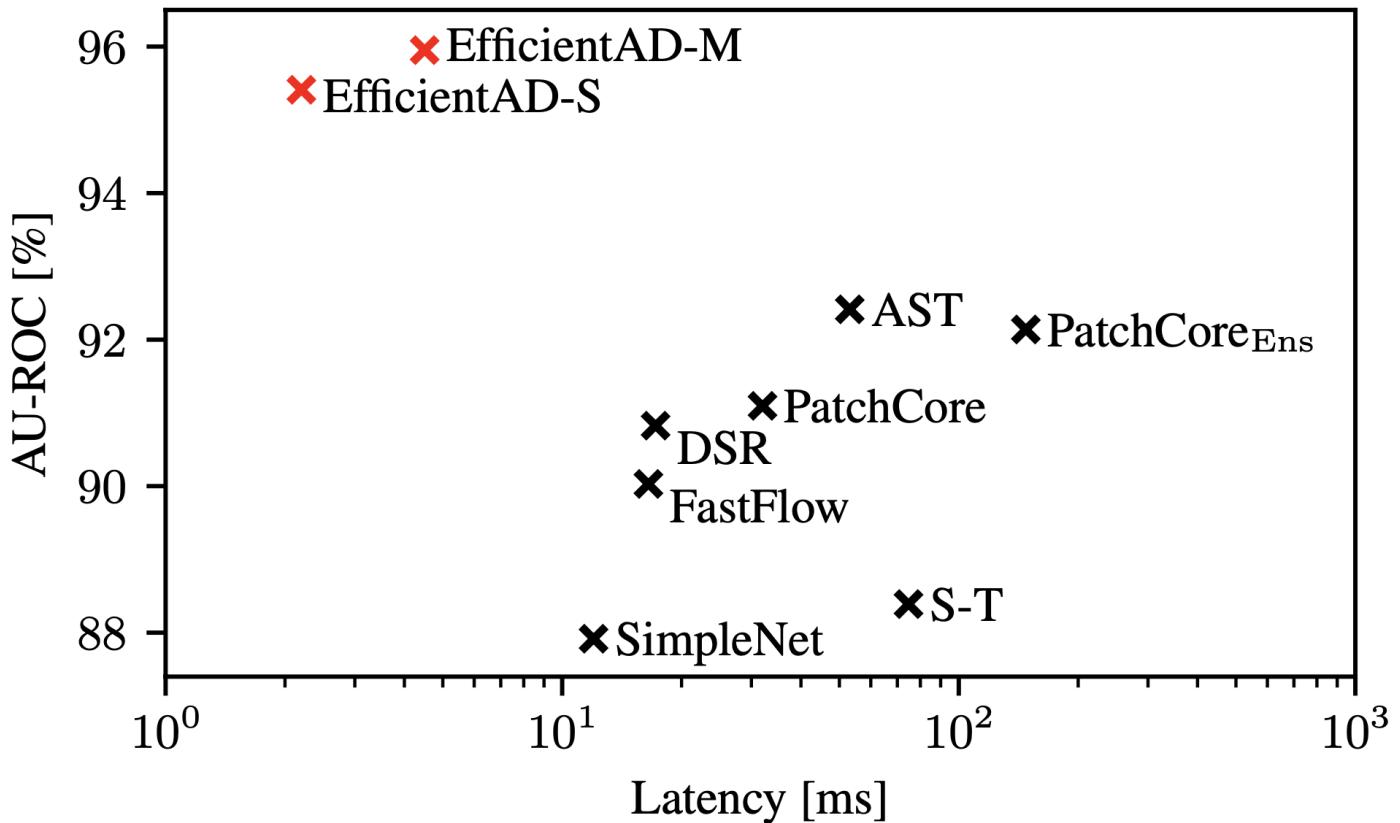


Figure 1. Anomaly detection performance vs. latency per image on an NVIDIA RTX A6000 GPU. Each AU-ROC value is an average of the image-level detection AU-ROC values on the MVTec AD [7, 9], VisA [69], and MVTec LOCO [8] dataset collections.

# Motivation (1)

## 1. Introduction

In the past years, deep learning methods have continued to improve the state of the art across a wide range of computer vision applications. This progress has been accompanied by advances in making neural network architectures faster and more efficient [43, 59, 61, 63]. Modern classification architectures, for example, focus on characteristics such as latency, throughput, memory consumption, and the number of trainable parameters [32, 33, 54, 59, 60, 63]. This ensures that as networks become more capable, their computational requirements remain suitable for real-world applications. The field of visual anomaly detection has also seen rapid progress in the recent past, especially on industrial anomaly detection benchmarks [7, 9, 47, 50]. State-of-the-art anomaly detection methods, however, often sacrifice computational efficiency for an increased anomaly detection performance. Common techniques are ensembling, the use of large backbones, and increasing the input image resolution to up to  $768 \times 768$  pixels.

Real-world anomaly detection applications frequently put constraints on the computational requirements of a method. There are cases where detecting an anomaly too late can cause substantial economic damage, such as metal objects in a crop field entering the interior of a combine harvester. In other cases, even human health is at risk, for example, if a limb of a machine operator approaches a blade. Furthermore, industrial settings commonly involve strict runtime limits caused by high production rates [4]. Not adhering to these limits would decrease the production rate of the respective application and thus its economic viability. It is therefore essential to pay attention to the computational and economic cost of anomaly detection methods to keep them suitable for real-world applications.

# Motivation (2)

## 3. Method

We describe the components of EfficientAD in the following subsections. It begins with the efficient extraction of features from a pretrained neural network in Sec. 3.1. We detect anomalous features at test time using a lightweight student–teacher model, as described in Sec. 3.2. A key challenge is to achieve a competitive anomaly detection performance while keeping the overall runtime low. To this end, we introduce a novel loss function for the training of a student–teacher model. In Sec. 3.3, we explain how to efficiently detect logical anomalies with an autoencoder-based approach. Finally, we provide a solution for calibrating and combining the detection results of the autoencoder with those of the student–teacher model in Sec. 3.4.

# In-depth details on EfficientAD

# Overview

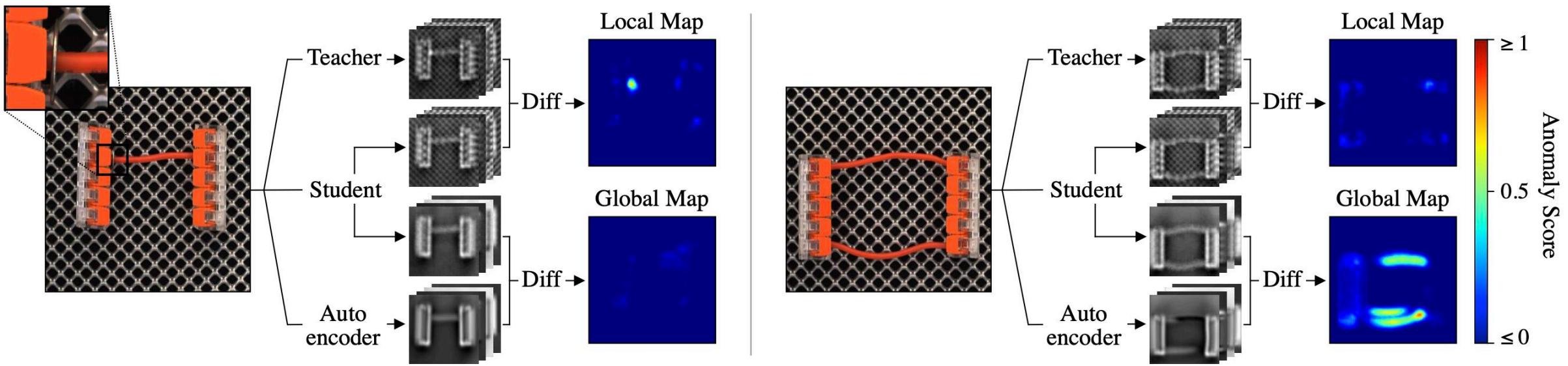
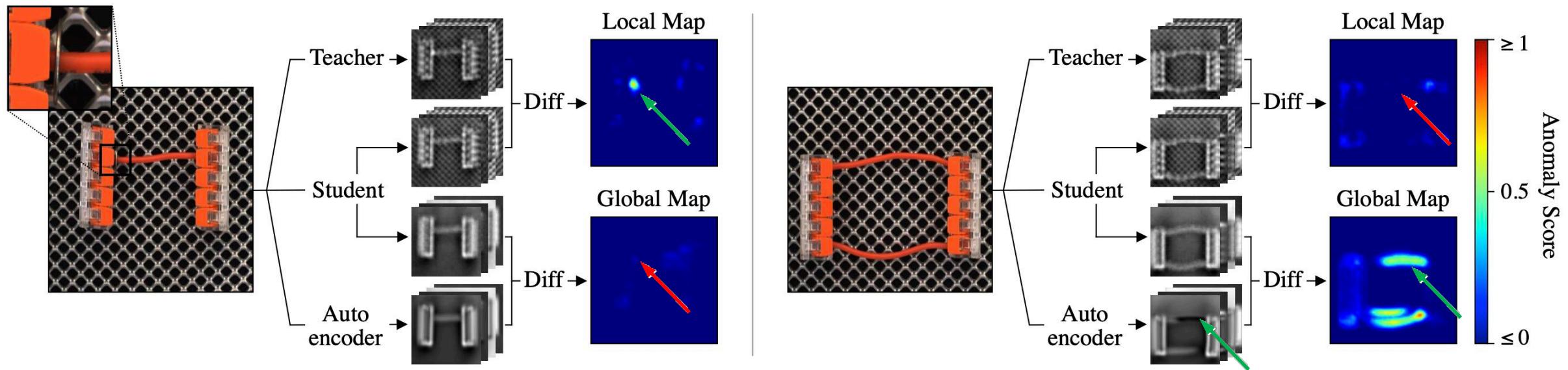


Figure 5. EfficientAD applied to two test images from MVTec LOCO. Normal input images contain a horizontal cable connecting the two splicing connectors at an arbitrary height. The anomaly on the left is a foreign object in the form of a small metal washer at the end of the cable. It is visible in the local anomaly map because the outputs of the student and the teacher differ. The logical anomaly on the right is the presence of a second cable. The autoencoder fails to reconstruct the two cables on the right in the feature space of the teacher. The student also predicts the output of the autoencoder in addition to that of the teacher. Because its receptive field is restricted to small patches of the image, it is not influenced by the presence of the additional red cable. This causes the outputs of the autoencoder and the student to differ. “Diff” refers to computing the element-wise squared difference between two collections of output feature maps and computing its average across feature maps. To obtain pixel anomaly scores, the anomaly maps are resized to match the input image using bilinear interpolation.

# Overview



## Student-Teacher pair

- Allows to detect local anomalies
- Cannot detect global anomalies

## Student-Autoencoder pair

- Enables to detect global and logical anomalies

# Patch Description Network (1)

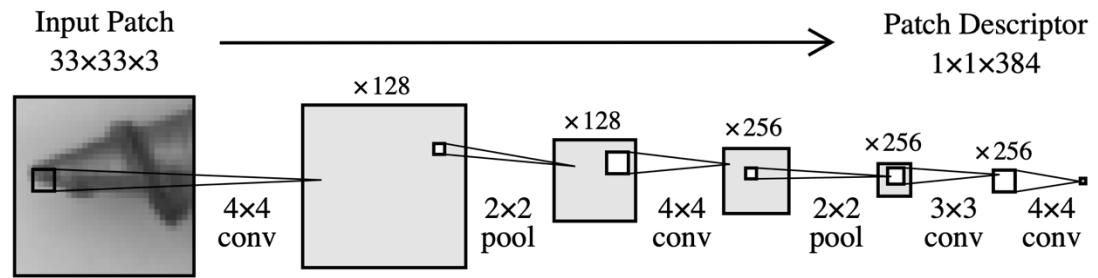


Figure 2. Patch description network (PDN) architecture of EfficientAD-S. Applying it to an image in a fully convolutional manner yields all features in a single forward pass.

Local anomalous features cannot affect local normal neighbors by a well-defined receptive field of PDN.

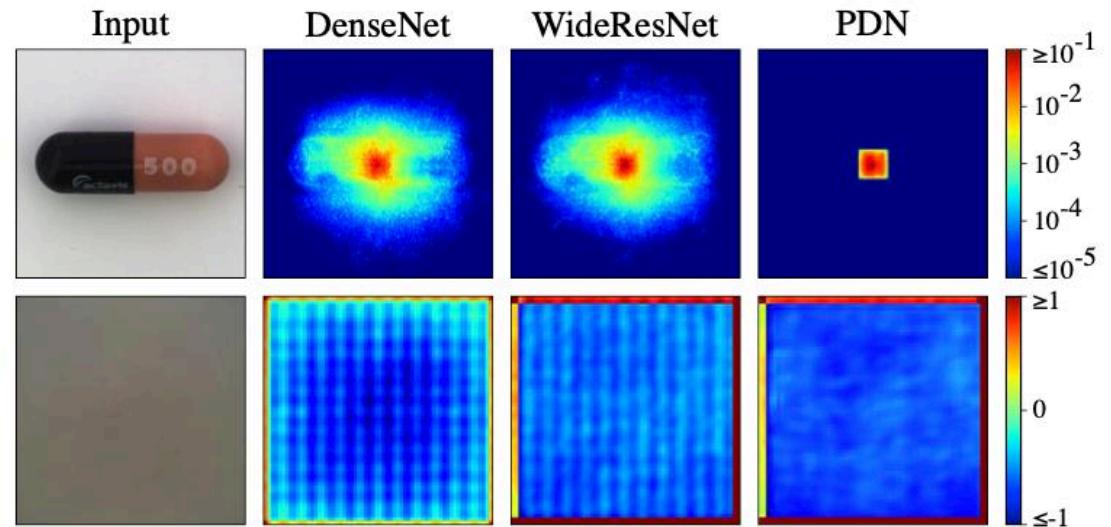


Figure 3. Upper row: absolute gradient of a single feature vector, located in the center of the output, with respect to each input pixel, averaged across input and output channels. Lower row: Average feature map of the first output channel across 1000 randomly chosen images from ImageNet [51]. The mean of these images is shown on the left. The feature maps of the DenseNet [24] and the WideResNet exhibit strong artifacts.

## Patch Description Network (2)

Layer Name	Stride	Kernel Size	Number of Kernels	Padding	Activation
Conv-1	$1 \times 1$	$4 \times 4$	128	3	ReLU
AvgPool-1	$2 \times 2$	$2 \times 2$	128	1	-
Conv-2	$1 \times 1$	$4 \times 4$	256	3	ReLU
AvgPool-2	$2 \times 2$	$2 \times 2$	256	1	-
Conv-3	$1 \times 1$	$3 \times 3$	256	1	ReLU
Conv-4	$1 \times 1$	$4 \times 4$	384	0	-

Table 1. Patch description network architecture of the teacher network for EfficientAD-S. The student network has the same architecture, but 768 kernels instead of 384 in the Conv-4 layer. A padding value of 3 means that three rows, or columns respectively, of zeros are appended at each border of an input feature map.

Layer Name	Stride	Kernel Size	Number of Kernels	Padding	Activation
Conv-1	$1 \times 1$	$4 \times 4$	256	3	ReLU
AvgPool-1	$2 \times 2$	$2 \times 2$	256	1	-
Conv-2	$1 \times 1$	$4 \times 4$	512	3	ReLU
AvgPool-2	$2 \times 2$	$2 \times 2$	512	1	-
Conv-3	$1 \times 1$	$1 \times 1$	512	0	ReLU
Conv-4	$1 \times 1$	$3 \times 3$	512	1	ReLU
Conv-5	$1 \times 1$	$4 \times 4$	384	0	ReLU
Conv-6	$1 \times 1$	$1 \times 1$	384	0	-

Table 2. Patch description network architecture of the teacher network for EfficientAD-M. The student network has the same architecture, but 768 kernels instead of 384 in the Conv-5 and Conv-6 layers. A padding value of 3 means that three rows, or columns respectively, of zeros are appended at each border of an input feature map.

# Detail

## Training Phase1 - teacher network ( $T$ )

- Section1.2 of Supplementary Material
- Algorithm 3 of Supplementary Material
- Table 1 and Table 2 of Supplementary Material
- Section 3.1 of Main paper
- Figure2 and Figure 4 of Main paper

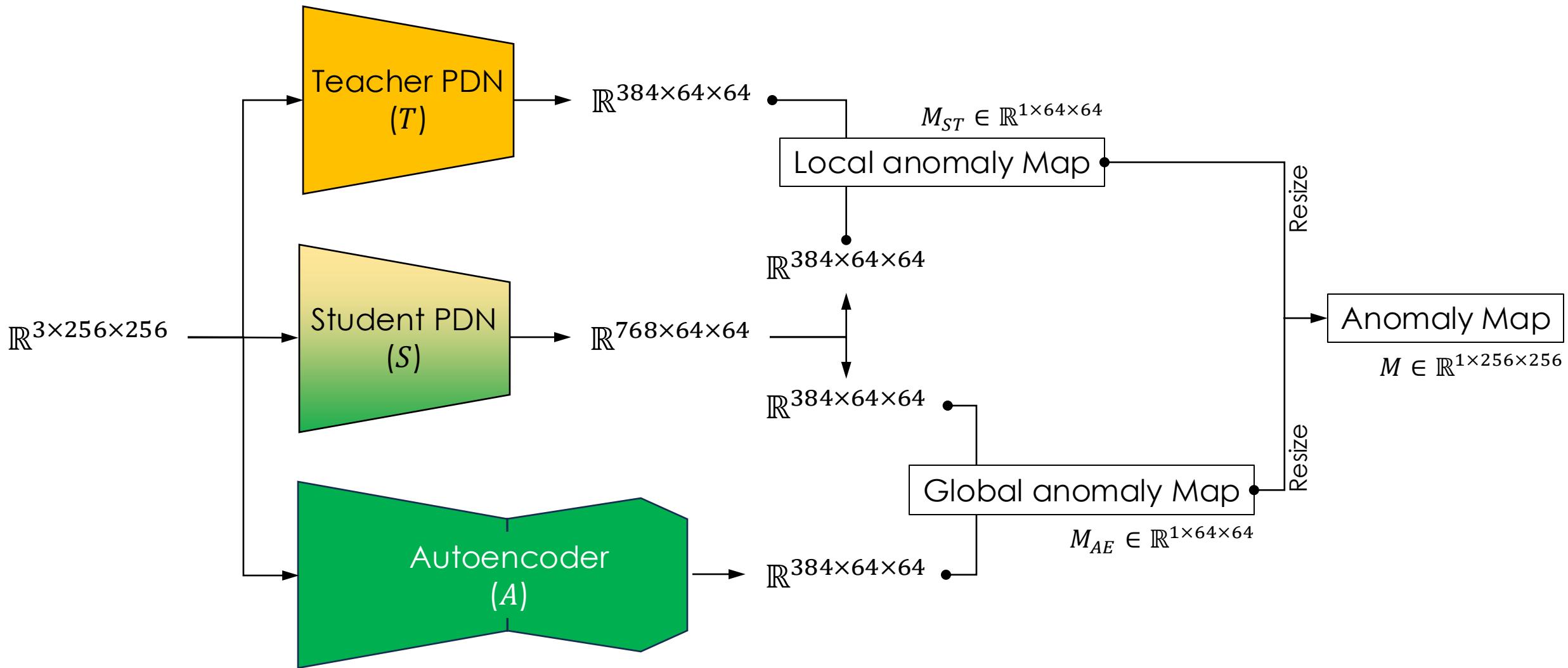
## Training Phase2 - student network ( $S$ ) and autoencoder ( $A$ )

- Section1.1 of Supplementary Material
- Algorithm 1 of Supplementary Material
- Table 1, Table 2, and Table 3 of Supplementary Material
- Section 3.2 and Section 3.3 of Main paper
- Figure2 and Figure 4 of Main paper

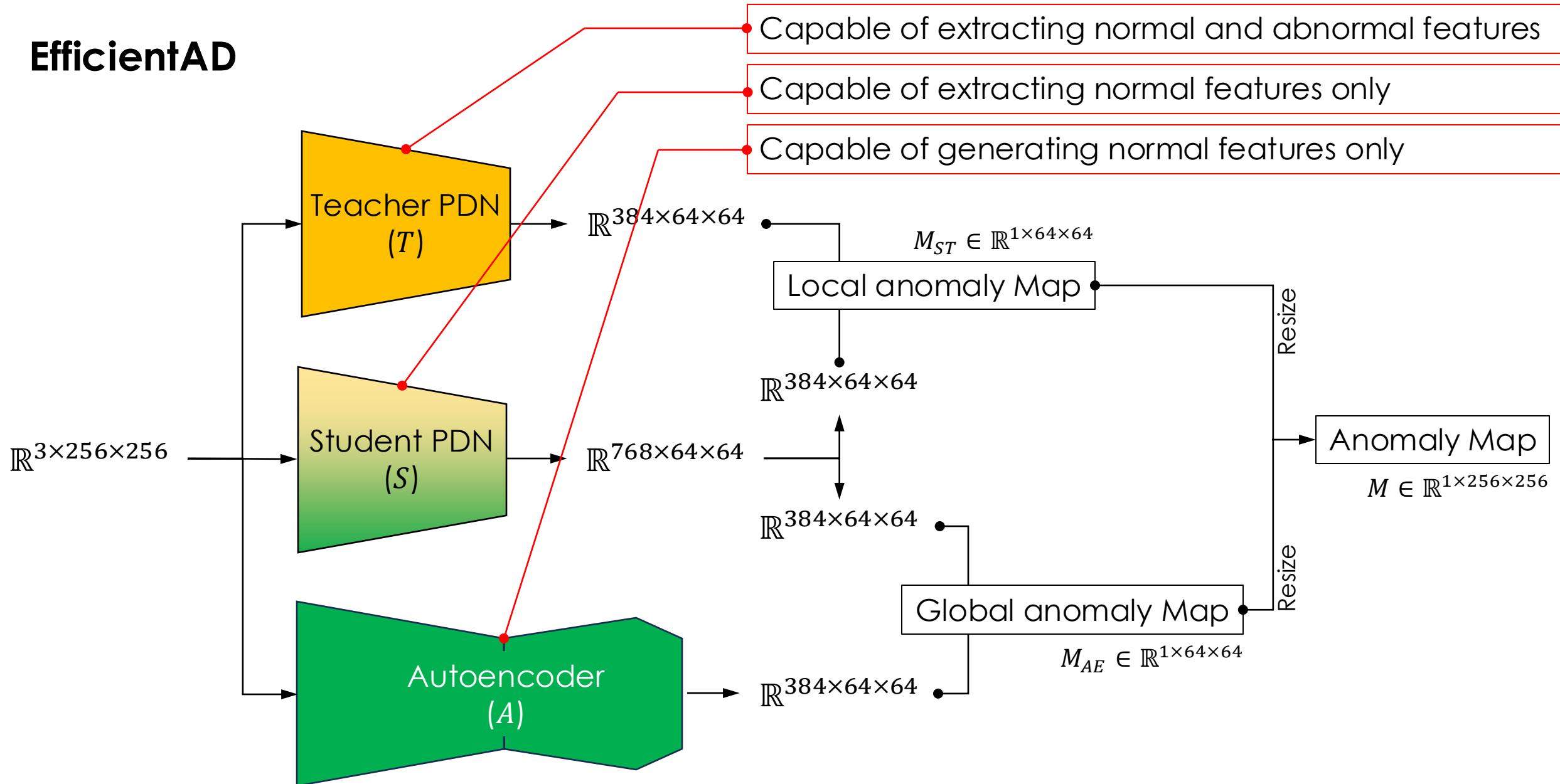
## Inference for anomaly detection ( $T$ , $S$ , and $A$ )

- Section1.1 of Supplementary Material
- Algorithm 2 of Supplementary Material
- Section 3.4 of Main paper
- Figure5 of Main paper

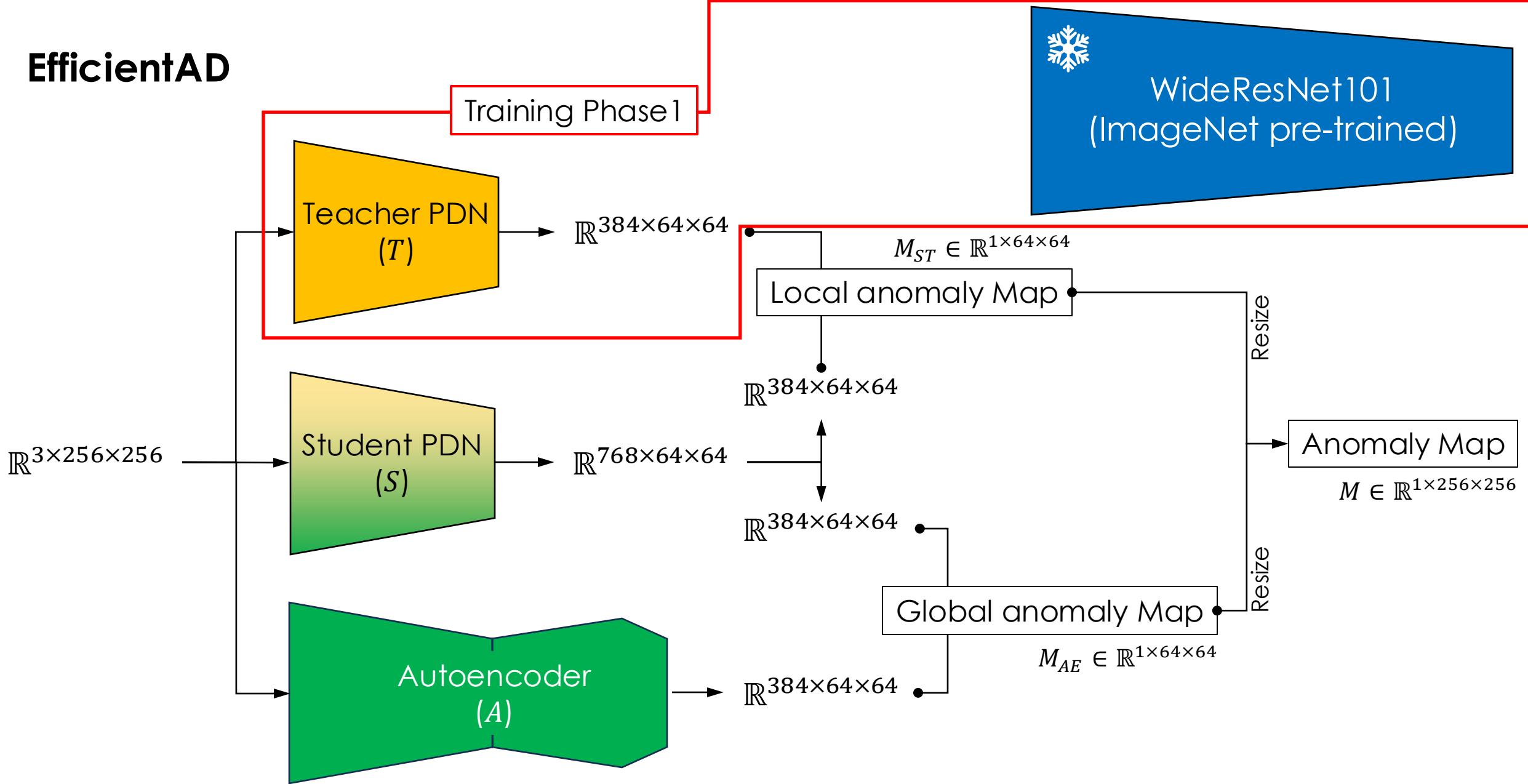
# EfficientAD



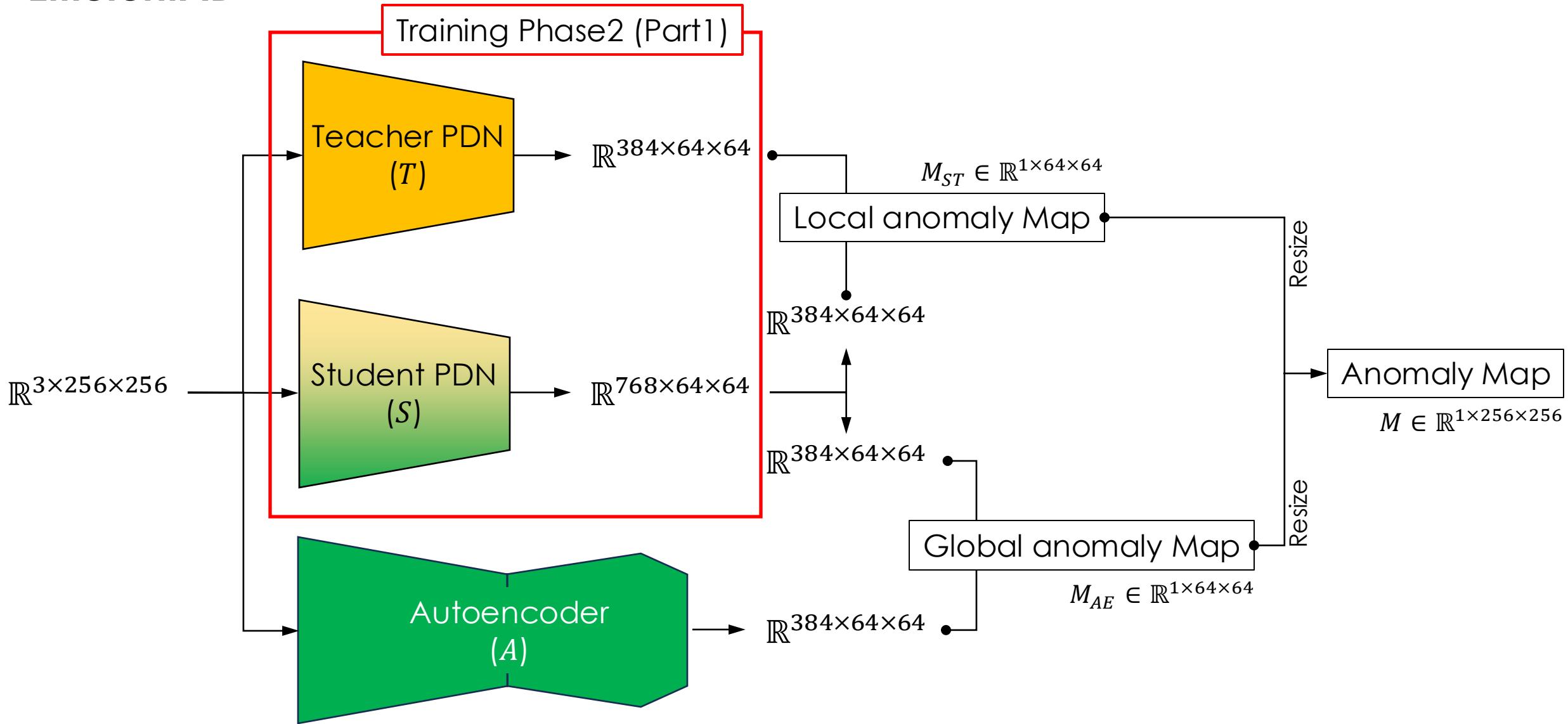
# EfficientAD



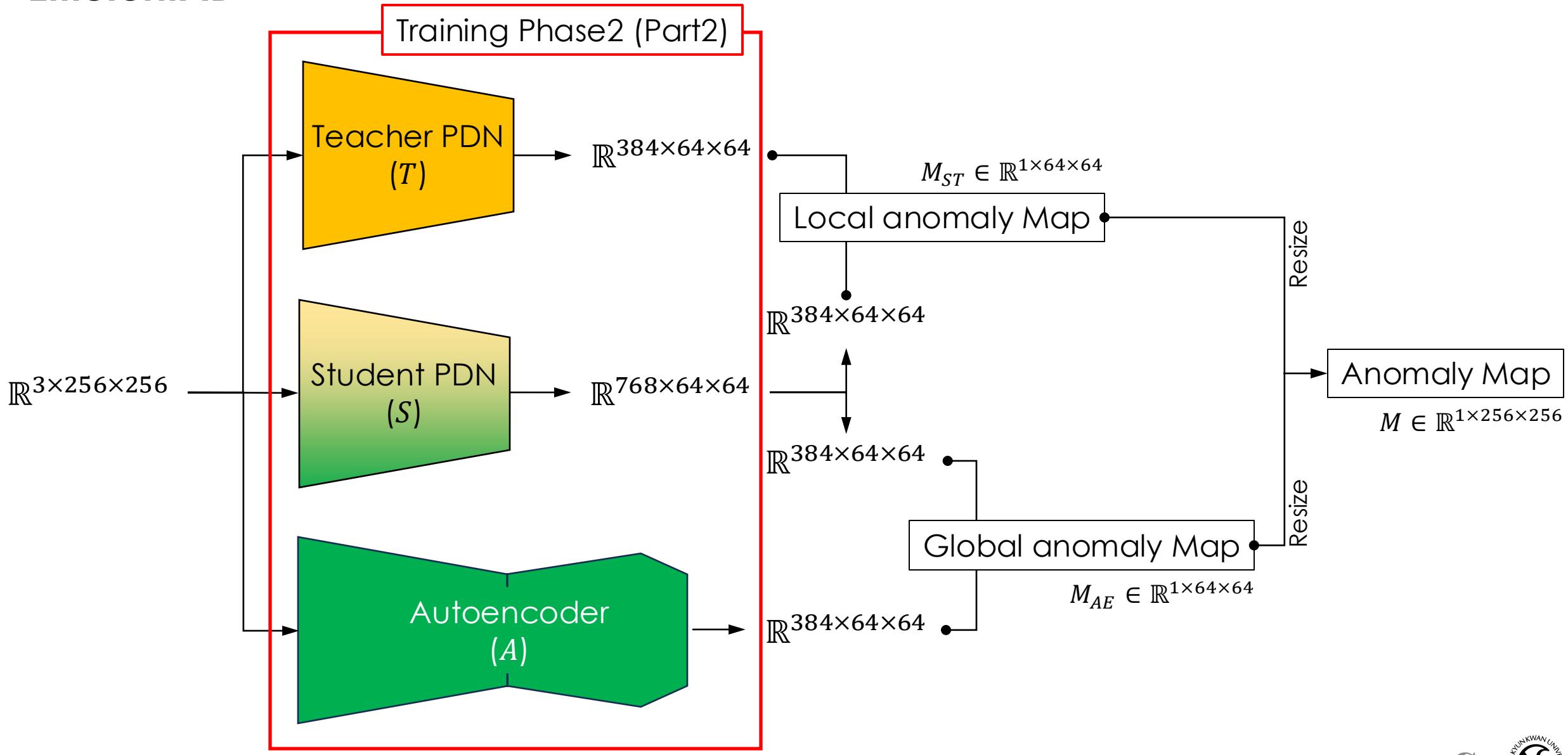
# EfficientAD



# EfficientAD



# EfficientAD

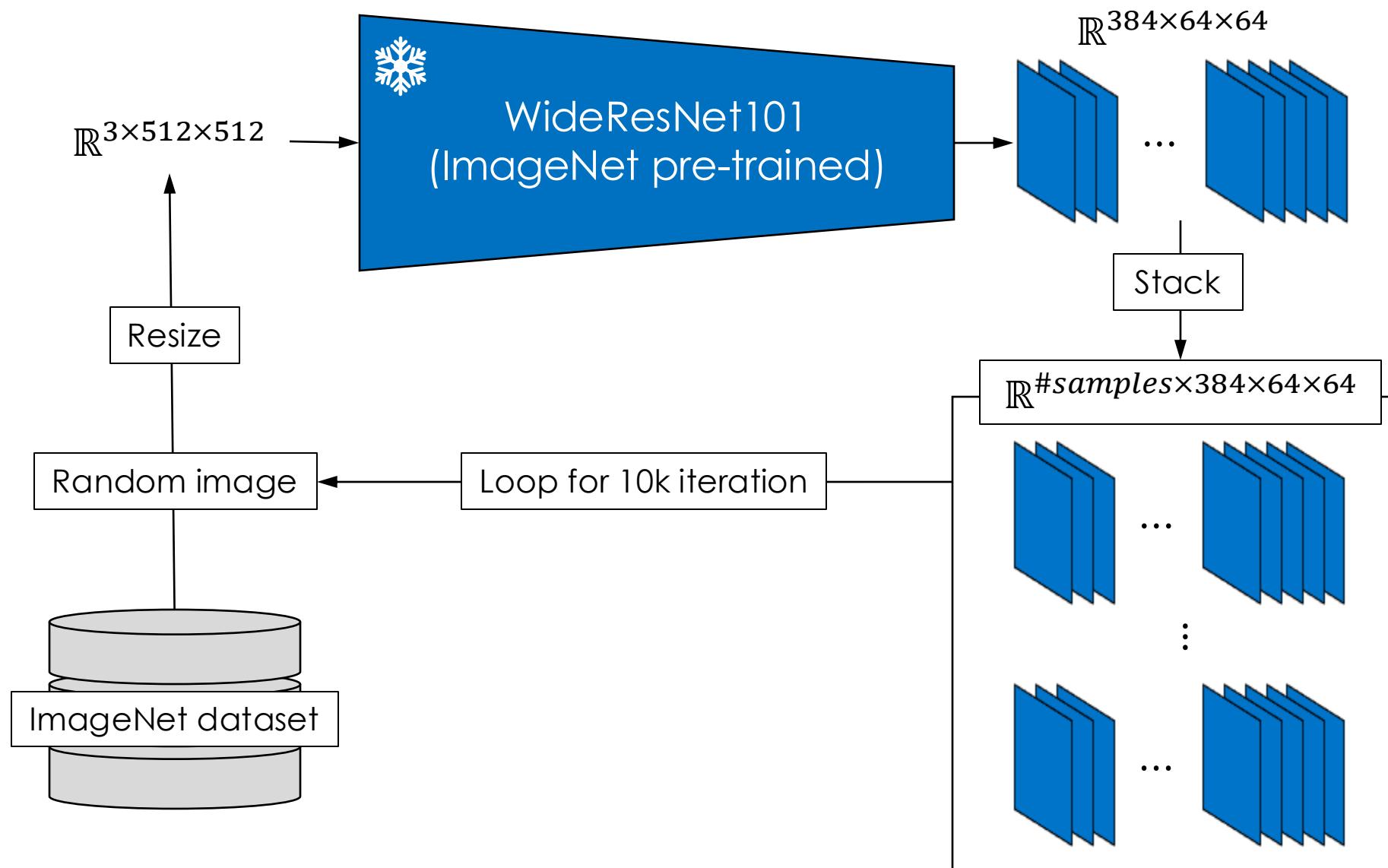


# Training Procedure

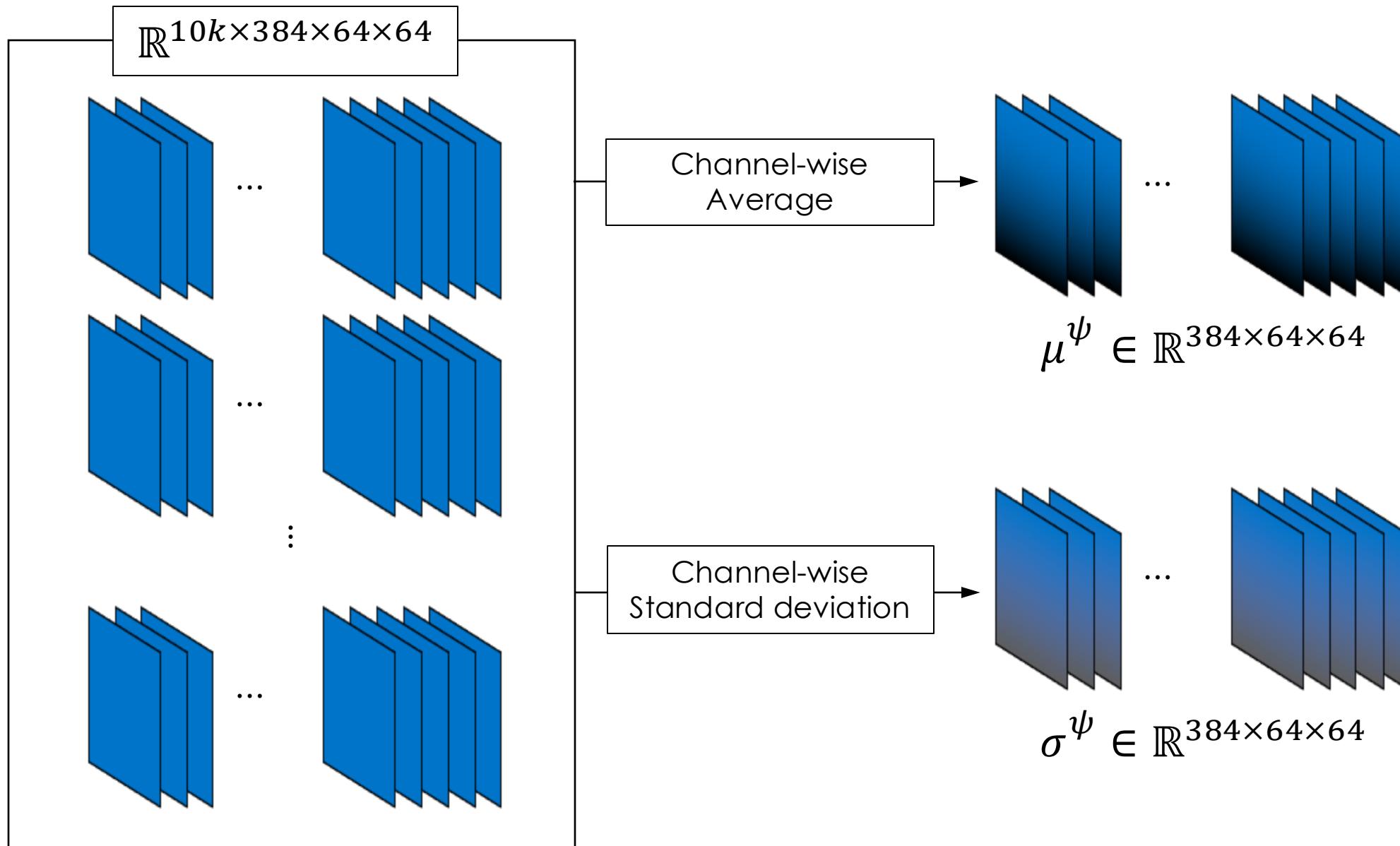
## Phase1

- Section1.2 of Supplementary Material
- Algorithm 3 of Supplementary Material
- Table 1 and Table 2 of Supplementary Material
- Section 3.1 of Main paper
- Figure2 and Figure 4 of Main paper

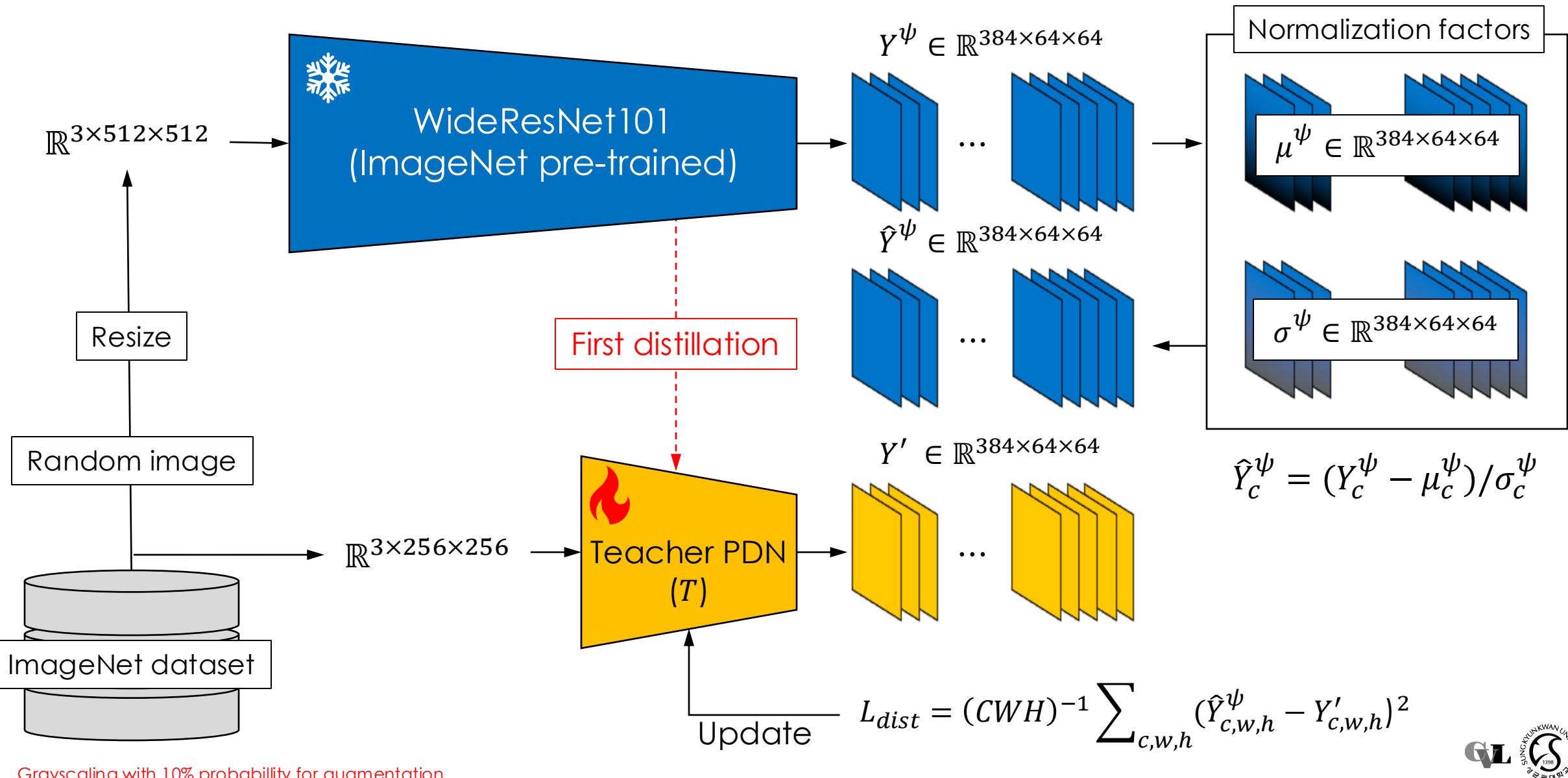
# Training Phase1 - Teacher Network (1)



## Training Phase1 - Teacher Network (2)



# Training Phase1 - Teacher Network (3)

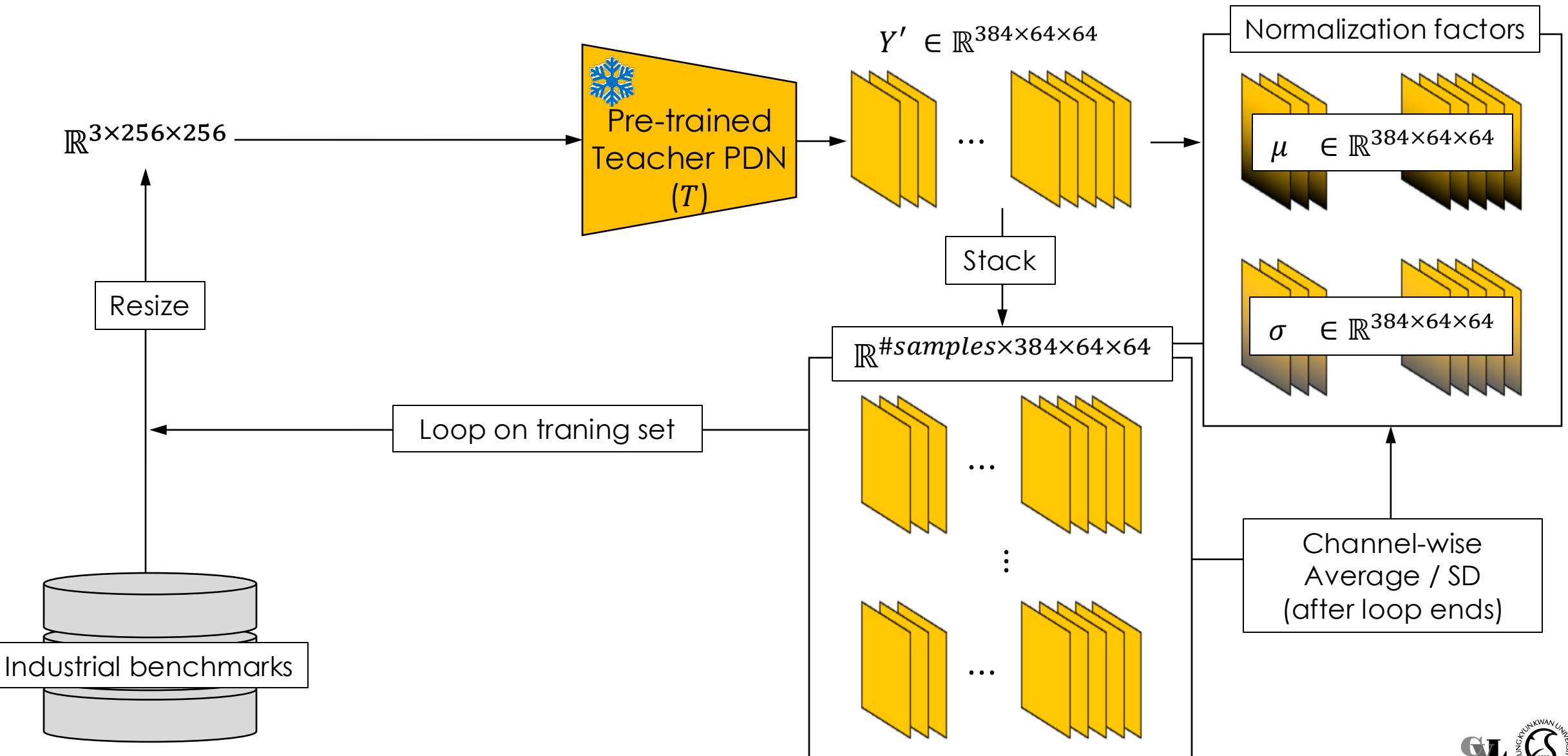


# Training Procedure

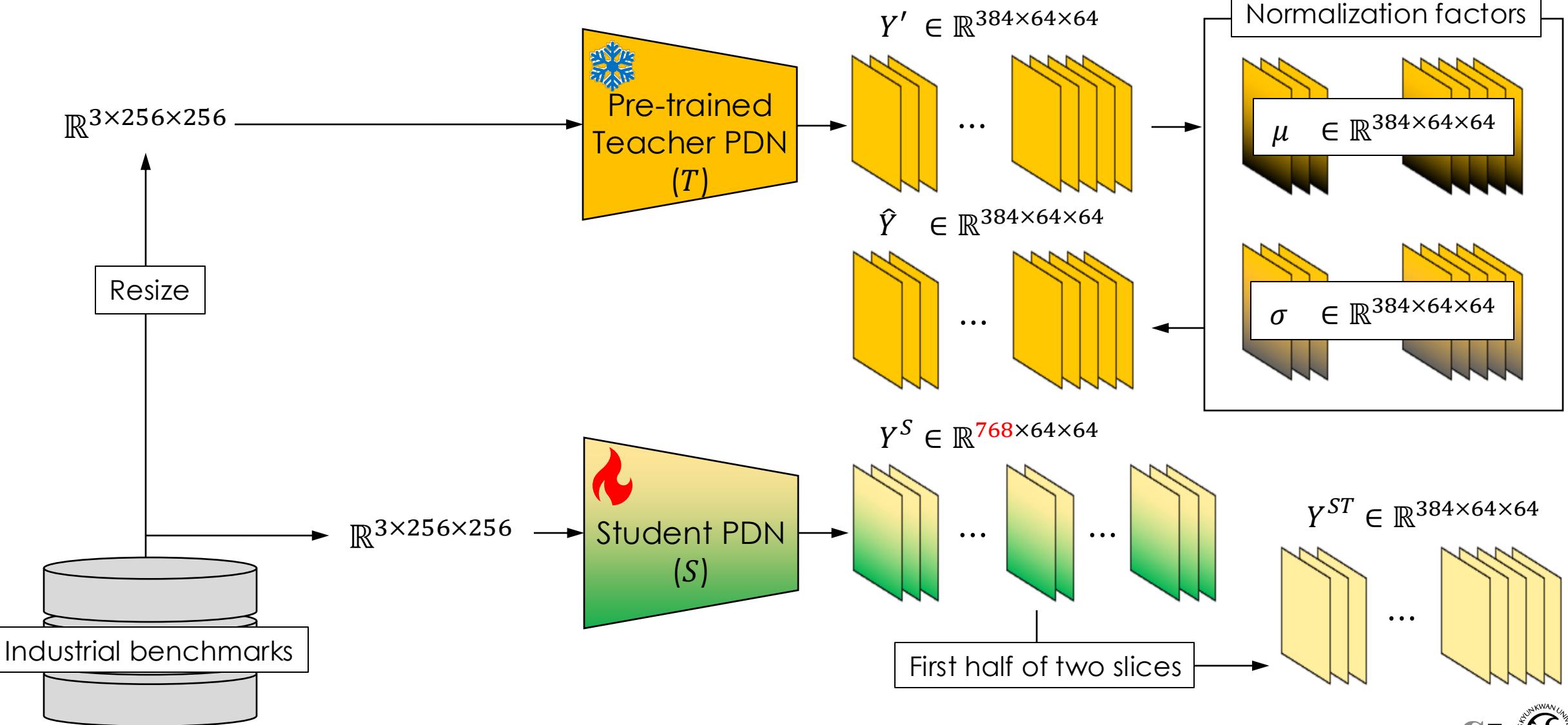
## Phase2

- Section1.1 of Supplementary Material
- Algorithm 1 of Supplementary Material
- Table 1, Table 2, and Table 3 of Supplementary Material
- Section 3.2 and Section 3.3 of Main paper
- Figure2 and Figure 4 of Main paper

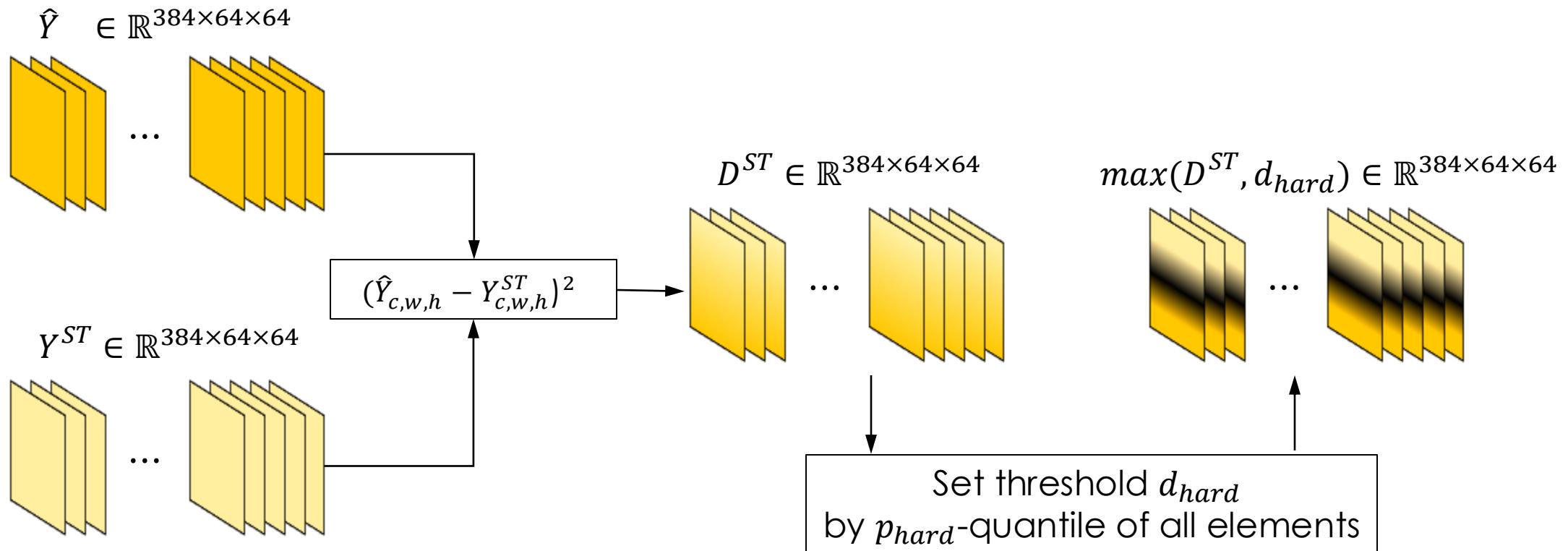
# Training Phase2 - Student Network Part1 (1)



# Training Phase2 - Student Network Part1 (2)



## Training Phase2 - Student Network Part1 (3)



$$L_{hard} = (CWH)^{-1} \sum_{c,w,h} (\hat{Y}_{c,w,h} - \max(D^{ST}, d_{hard}))^2$$

# Hard Feature Loss

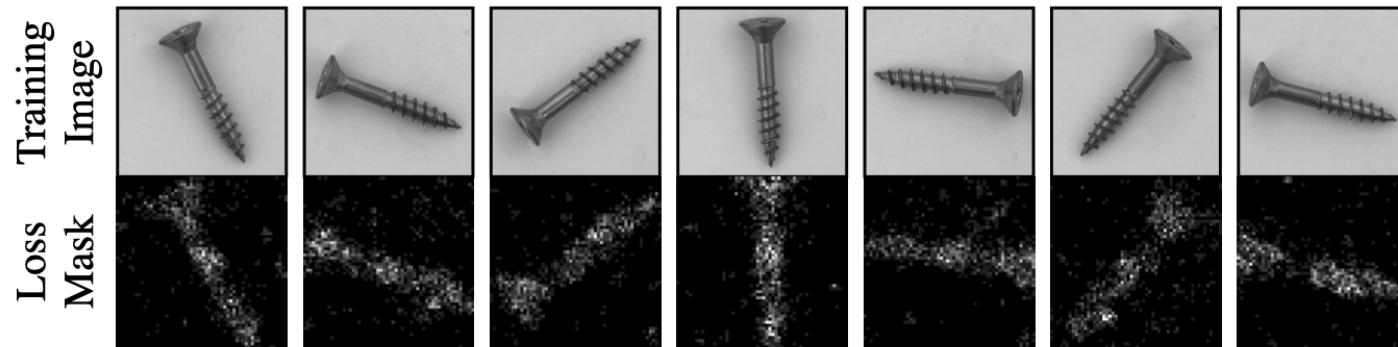
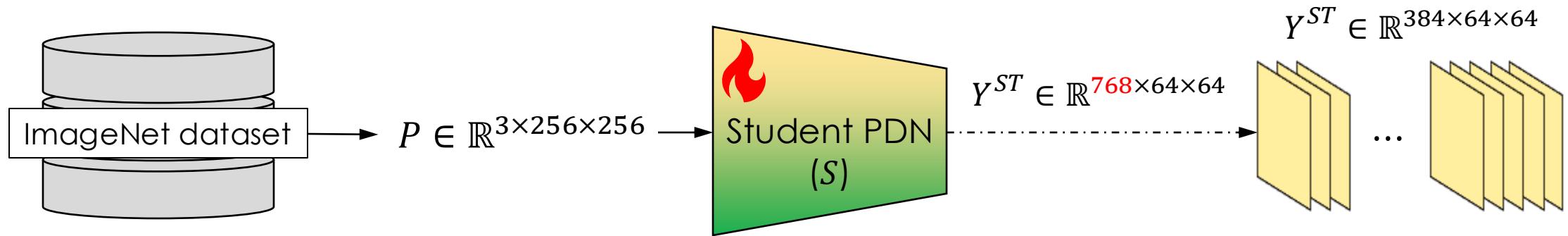


Figure 4. Randomly picked loss masks generated by the hard feature loss during training. The brightness of a mask pixel indicates how many of the dimensions of the respective feature vector were selected for backpropagation. The student network already mimics the teacher well on the background and thus focuses on learning the features of differently rotated screws.

Only **pixels of outlier-level errors** are exploited for backpropagation.

## Training Phase2 - Student Network Part1 (4)

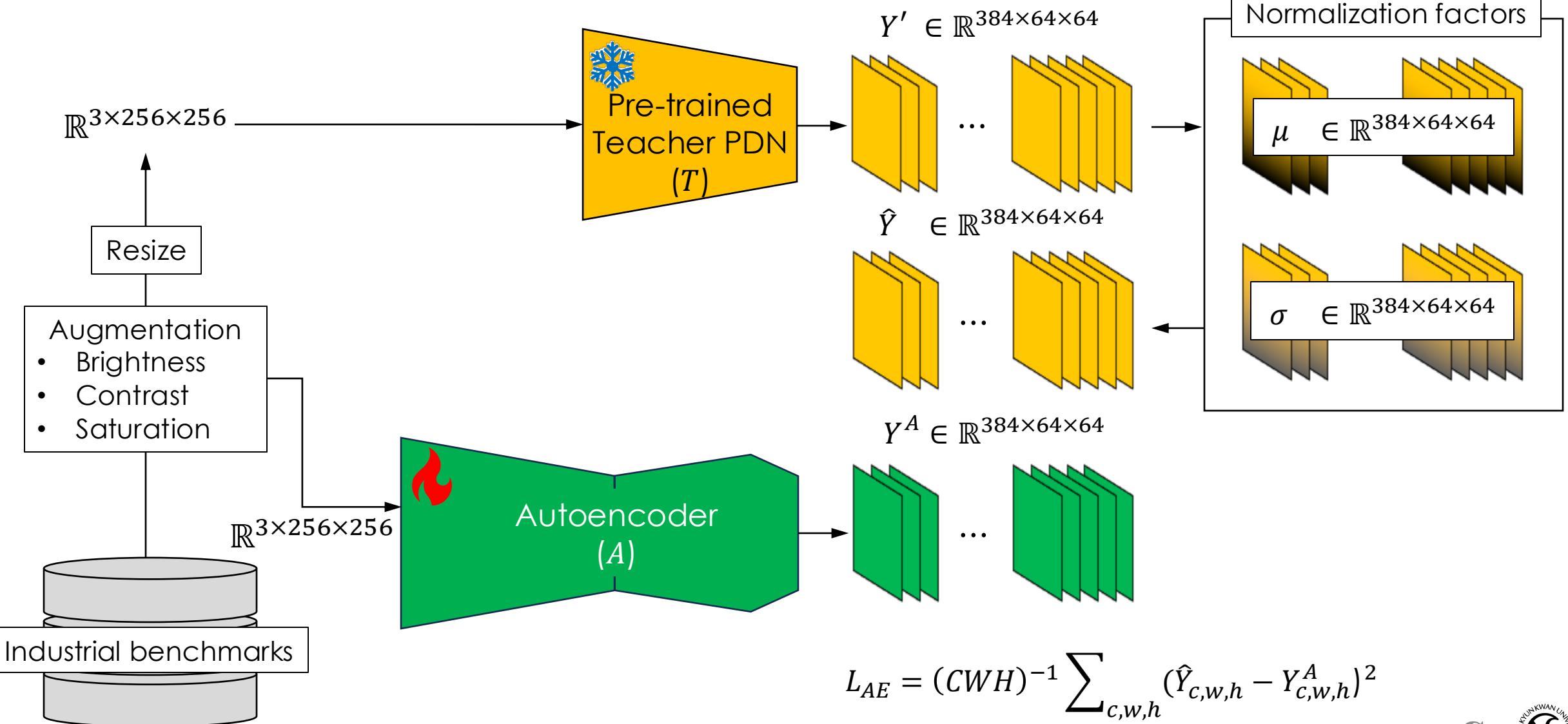


$$L_{penalty} = (CWH)^{-1} \sum_{c,w,h} \|S(P)\|_F^2$$

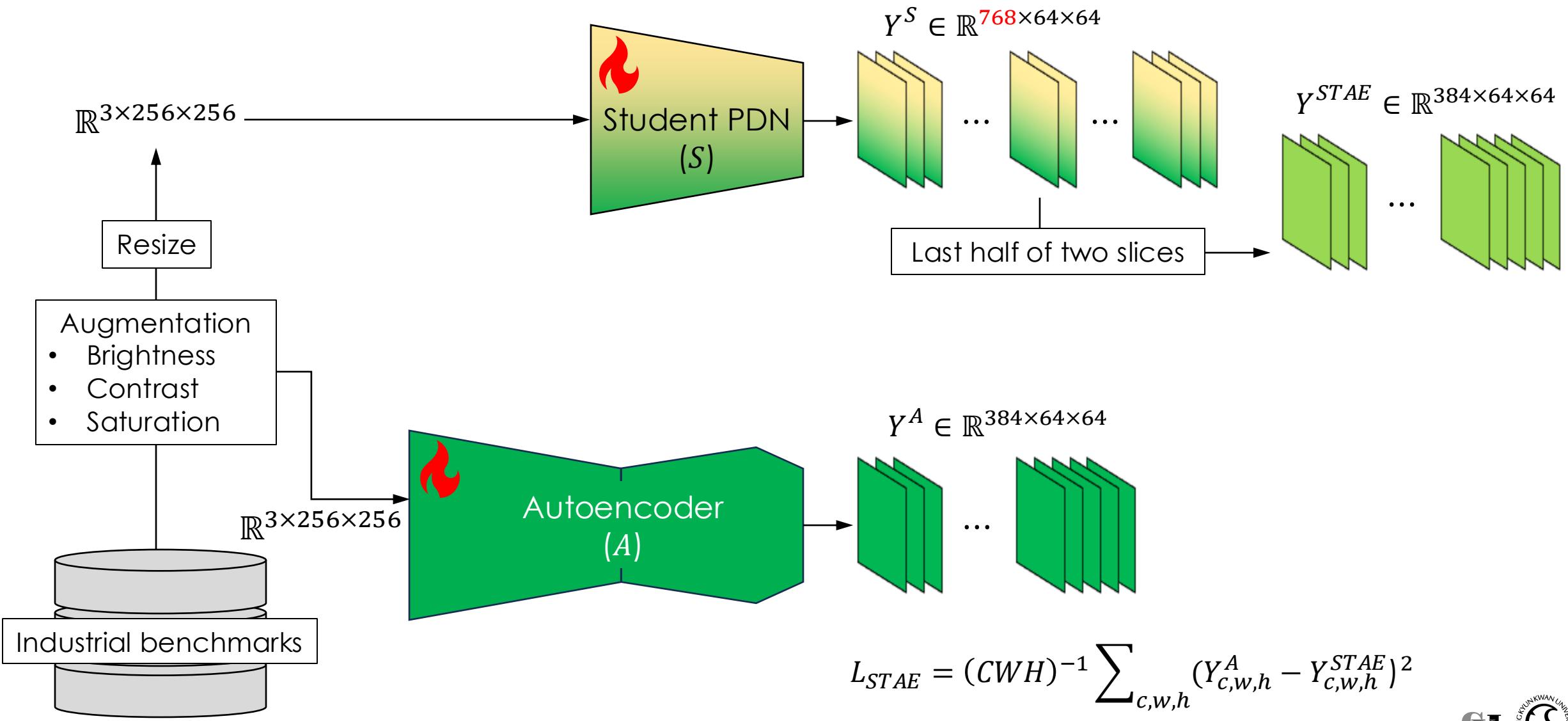
Trace of squared error

$$L_{ST} = L_{part} + L_{penalty}$$

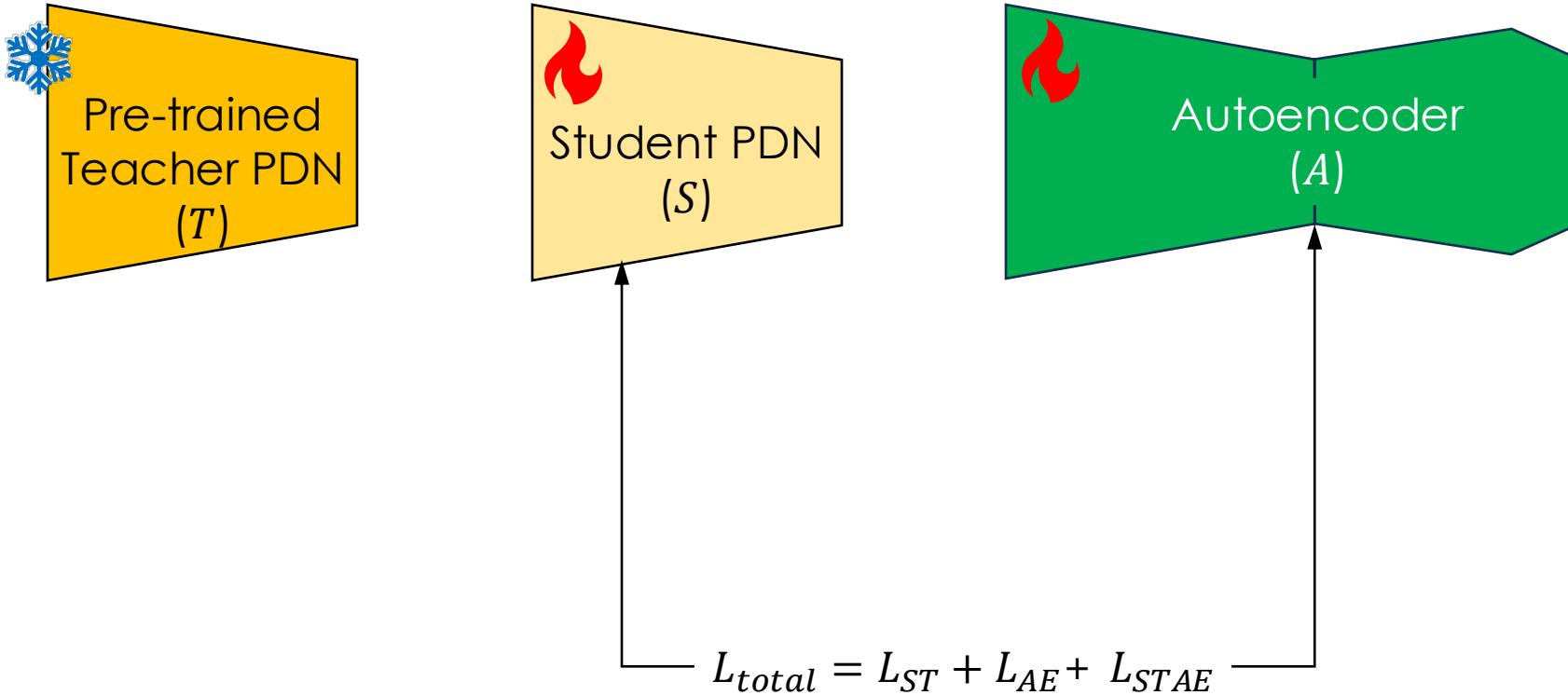
# Training Phase2 - Student Network Part2 (1)



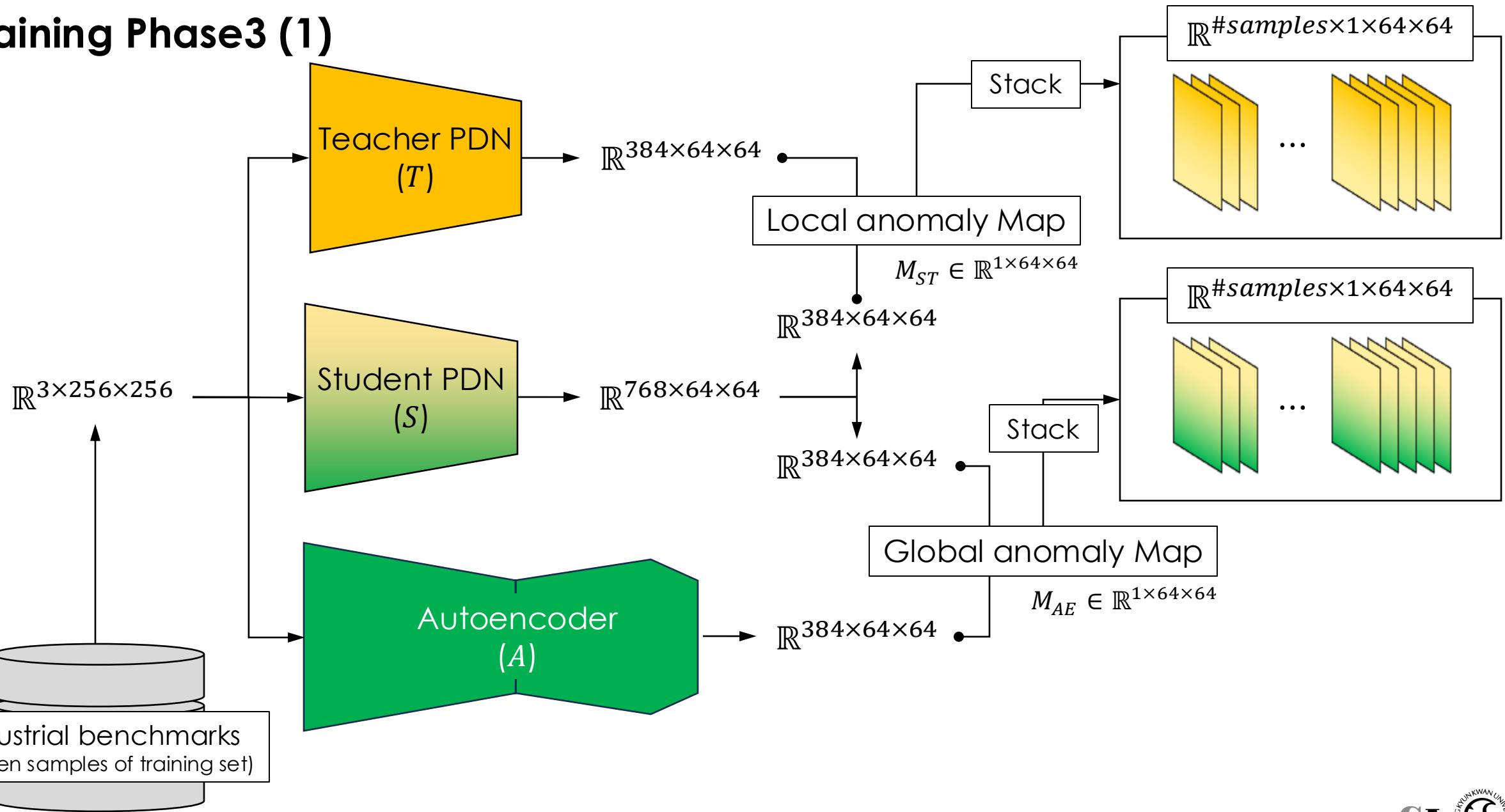
## Training Phase2 - Student Network Part2 (2)



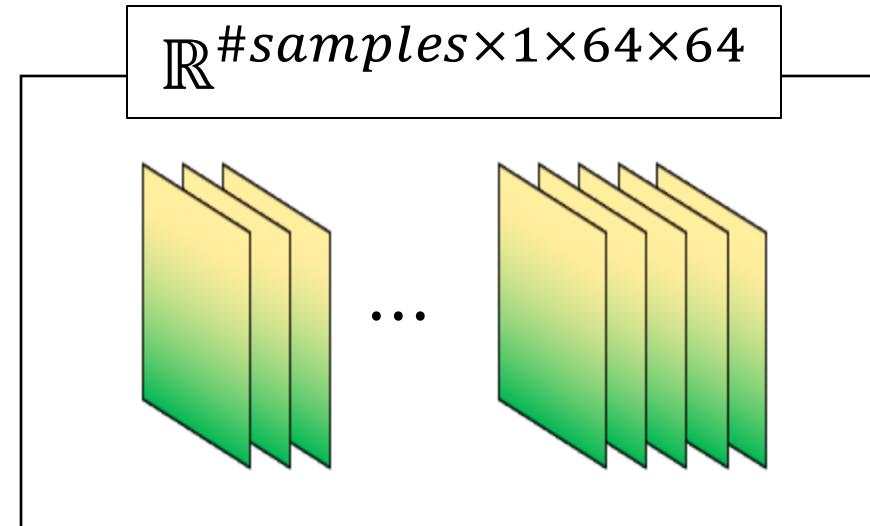
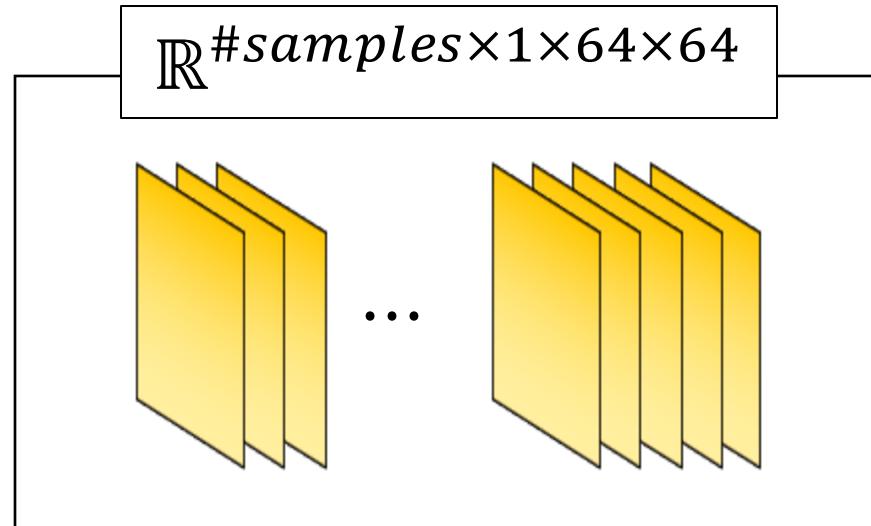
# Training Phase2 - Parameter Update



## Training Phase3 (1)



## Training Phase3 (2)

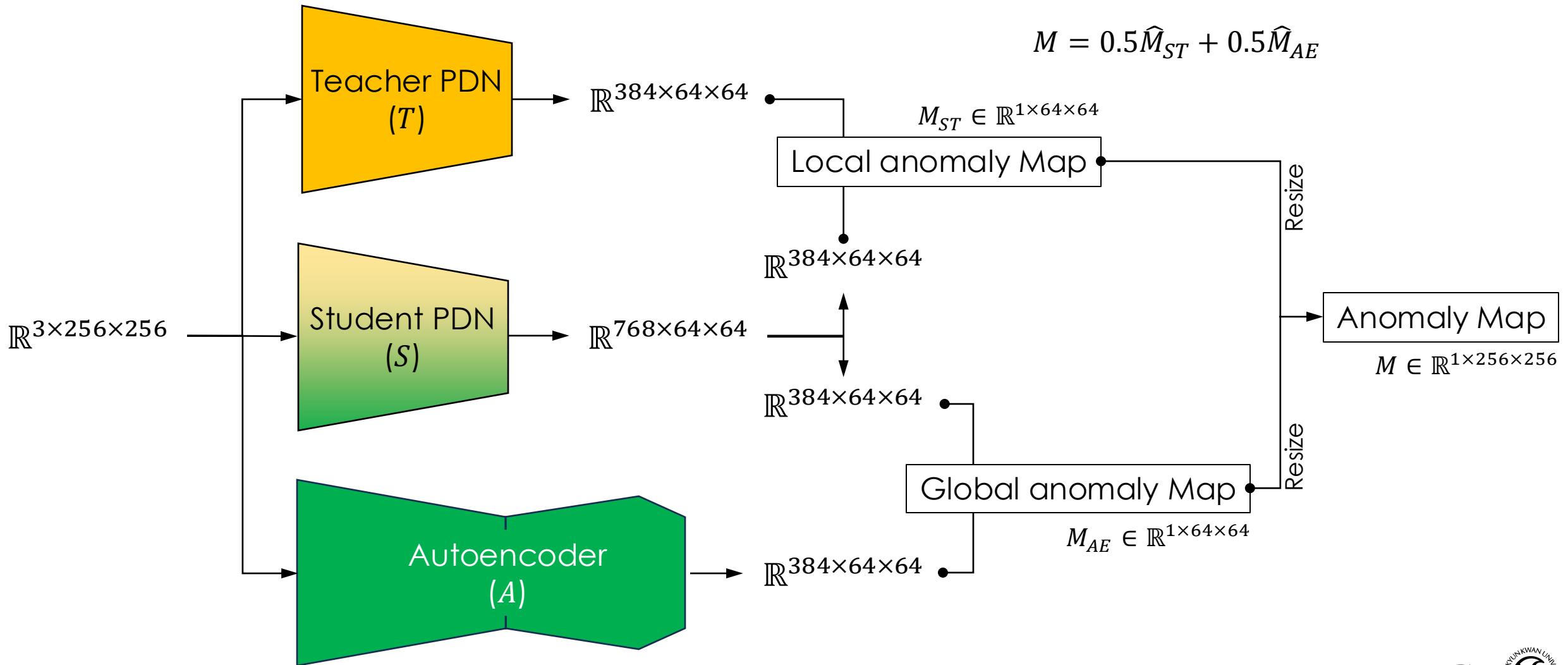


- 0.9-quantiile  $q_a^{ST}$
- 0.999-quantiile  $q_b^{ST}$
- 0.9-quantiile  $q_a^{AE}$
- 0.999-quantiile  $q_b^{AE}$

# Inference Procedure

- Section 1.1 of Supplementary Material
- Algorithm 2 of Supplementary Material
- Section 3.4 of Main paper
- Figure 5 of Main paper

# Inference



$$\hat{M}_{ST} = 0.1(M_{ST} - q_a^{ST})(q_b^{ST} - q_a^{ST})^{-1}$$

$$\hat{M}_{AE} = 0.1(M_{AE} - q_a^{AE})(q_b^{AE} - q_a^{AE})^{-1}$$

$$M = 0.5\hat{M}_{ST} + 0.5\hat{M}_{AE}$$

# Experiments

# Introduction of Experiments

## 4. Experiments

We compare EfficientAD to AST [50], DSR [66], Fast-Flow [64], GCAD [8], PatchCore [47], SimpleNet [34], and S-T [10], using official implementations where available. We provide configuration details for all evaluated methods in the supplementary material. GCAD consists of an ensemble of two anomaly detection models that use different feature extractors. We find that one of the two ensemble members performs better on average than the combined ensemble and therefore report the results for this member. This reduces the latency reported for GCAD by a factor of two. For SimpleNet, we are able to reproduce the official results but find that SimpleNet tunes the training duration on the test images of a scenario. During training, the model is repeatedly evaluated on all test images and the maximum of all obtained test scores is reported after training. We disable this technique, since it overestimates the actual performance of the model on unseen images. In practice, it would furthermore require a validation set with anomalous images. MVTec AD, VisA, and MVTec LOCO do not include anomalous images in their training and validation sets to avoid defect-type-specific tuning of hyperparameters. For SimpleNet, we therefore evaluate the final trained model, following common practice.

For PatchCore, we include two variants: the default single model variant, for which the authors report the lowest latency, and the ensemble variant, denoted by PatchCore<sub>Ens</sub>. We are able to reproduce the official results but disable the cropping of the center 76.6 % of input images for a fair comparison. In the case of MVTec AD, 99.9 % of the defects lie fully or partially within this cropped area. In real-world applications, anomalies can occur outside of this area as well. We disable custom cropping, as it implies knowledge about the anomalies in the test set.

### 4.1. Experimental Details

**Datasets.** To study industrial anomaly detection performance, the majority of our experiments are performed on the MVTec Anomaly Detection benchmark [5].

MVTec AD contains 15 sub-datasets with a total of 5354 images, 1725 of which are in the test set. Each sub-dataset is divided into nominal-only training data and test sets containing both nominal and anomalous samples for a specific product with various defect types as well as respective anomaly ground truth masks. As in [10, 14, 56], images are resized and center cropped to  $256 \times 256$  and  $224 \times 224$ , respectively. No data augmentation is applied, as this requires prior knowledge about class-retaining augmentations.

# Efficiency and Anomaly Detection Performance

Method	Detect. AU-ROC	Segment. AU-PRO	Latency [ms]	Throughput [img / s]
GCAD	85.4	88.0	11	121
SimpleNet	87.9	74.4	12	194
S-T	88.4	89.7	75	16
FastFlow	90.0	86.5	17	120
DSR	90.8	78.6	17	104
PatchCore	91.1	80.9	32	76
PatchCore <sub>Ens</sub>	92.1	80.7	148	13
AST	92.4	77.2	53	41
EfficientAD-S	95.4 ( $\pm$ 0.06)	92.5 ( $\pm$ 0.05)	<b>2.2</b> ( $\pm$ 0.01)	<b>614</b> ( $\pm$ 2)
EfficientAD-M	<b>96.0</b> ( $\pm$ 0.09)	<b>93.3</b> ( $\pm$ 0.04)	4.5 ( $\pm$ 0.01)	269 ( $\pm$ 1)

Table 1. Anomaly detection and anomaly localization performance in comparison to the latency and throughput. Each AU-ROC and AU-PRO percentage is an average of the mean AU-ROCs and mean AU-PROs, respectively, on MVTec AD, VisA, and MVTec LOCO. For EfficientAD, we report the mean and standard deviation of five runs.

Method	MAD	LOCO	VisA	Mean	LOCO Logic.	LOCO Struct.
GCAD	89.1	83.3	83.7	85.4	83.9	82.7
SimpleNet	98.2	77.6	87.9	87.9	71.5	83.7
S-T	93.2	77.4	94.6	88.4	66.5	88.3
FastFlow	96.9	79.2	93.9	90.0	75.5	82.9
DSR	98.1	82.6	91.8	90.8	75.0	90.2
PatchCore	98.7	80.3	94.3	91.1	75.8	84.8
PatchCore <sub>Ens</sub>	<b>99.3</b>	79.4	97.7	92.1	71.0	87.7
AST	98.9	83.4	94.9	92.4	79.7	87.1
EfficientAD-S	98.8	90.0	97.5	95.4	85.8	94.1
EfficientAD-M	99.1	<b>90.7</b>	<b>98.1</b>	<b>96.0</b>	<b>86.8</b>	<b>94.7</b>

Table 2. Mean anomaly detection AU-ROC percentages per dataset collection (left) and on the logical and structural anomalies of MVTec LOCO (right). For EfficientAD, we report the mean of five runs. Performing method development solely on MVTec AD (MAD) becomes prone to overfitting design choices to the few remaining misclassified test images.

# Efficiency and Anomaly Detection Performance

Method	Detect. AU-ROC	Segment. AU-PRO	Latency [ms]	Throughput [img / s]	Number of Parameters [ $\times 10^6$ ]	FLOPs [ $\times 10^9$ ]	GPU Memory [MB]
GCAD	85.4	88.0	11	121	65	416	555
SimpleNet	87.9	74.4	12	194	73	38	508
S-T	88.4	89.7	75	16	26	4468	1077
FastFlow	90.0	86.5	17	120	92	85	404
DSR	90.8	78.6	17	104	40	267	314
PatchCore	91.1	80.9	32	76	83 + 3	41 + kNN	637 + kNN
PatchCore <sub>Ens</sub>	92.1	80.7	148	13	150 + 8	159 + kNN	1335 + kNN
AST	92.4	77.2	53	41	154	199	618
EfficientAD-S	95.4 ( $\pm 0.06$ )	92.5 ( $\pm 0.05$ )	2.2 ( $\pm 0.01$ )	614 ( $\pm 2$ )	8 ( $\pm 0$ )	76 ( $\pm 0$ )	100 ( $\pm 0$ )
EfficientAD-M	96.0 ( $\pm 0.09$ )	93.3 ( $\pm 0.04$ )	4.5 ( $\pm 0.01$ )	269 ( $\pm 1$ )	21 ( $\pm 0$ )	235 ( $\pm 0$ )	161 ( $\pm 0$ )

Table 11. Extension of Table 1 in the main paper by additional computational efficiency metrics measured on a NVIDIA RTX A6000 GPU. For EfficientAD, we report the mean and standard deviation of five runs. For PatchCore, we report the computational requirements of the feature extraction during inference separately from the nearest neighbor search.

# Hyperparameter tuning

$a$ (for $q_a$ )	0.5	0.8	<b>0.9</b>	0.95	0.98	0.99
AU-ROC	95.9	95.9	96.0	95.9	95.9	95.8
$b$ (for $q_b$ )	0.95	0.98	0.99	<b>0.995</b>	0.998	0.999
AU-ROC	95.8	95.9	96.0	96.0	95.9	95.9
$p_{\text{hard}}$	0	0.9	0.99	<b>0.999</b>	0.9999	0.99999
AU-ROC	94.9	94.9	95.7	96.0	95.8	95.7

Table 3. Mean anomaly detection AU-ROC of EfficientAD-M on MVTec AD, VisA, and MVTec LOCO when varying the locations of quantiles. These are the two sampling points  $a$  and  $b$  of the quantile-based map normalization and the mining factor  $p_{\text{hard}}$ . Setting  $p_{\text{hard}}$  to zero disables the proposed hard feature loss. Default values used in our experiments are highlighted in bold.

# Ablation study

	Detection AU-ROC	Diff.	Latency [ms]
PDN	93.2		2.2
↪ with map normalization	94.0	+ 0.8	2.2
↪ with hard feature loss	95.0	+ 1.0	2.2
↪ with pretraining penalty	95.4	+ 0.4	2.2
EfficientAD-S	95.4		2.2
EfficientAD-M	96.0	+ 0.6	4.5

Table 4. Cumulative ablation study in which techniques are gradually combined to form EfficientAD. Each AU-ROC percentage is an average of the mean AU-ROCs on MVTec AD, VisA, and MVTec LOCO.

	Detection AU-ROC	Diff.	Latency [ms]
EfficientAD-S	95.4		2.2
Without map normalization	94.7	- 0.7	2.2
Without hard feature loss	94.7	- 0.7	2.2
Without pretraining penalty	95.0	- 0.4	2.2

Table 5. Isolated ablation study in which techniques are separately removed from EfficientAD-S.

- We introduce a training loss that significantly improves the anomaly detection performance of a student–teacher model without affecting its inference runtime.