# Towards Model-Agnostic Dataset Condensation by Heterogeneous Models

Jun-Yeong Moon, Jung Uk Kim*, and Gyeong-Moon Park*

Kyung Hee University, Yongin, Republic of Korea
{moonjunyyy, ju.kim, gmpark}@khu.ac.kr

**Abstract.** The advancement of deep learning has coincided with the proliferation of both models and available data. The surge in dataset sizes and the subsequent surge in computational requirements have led to the development of the Dataset Condensation (DC). While prior studies have delved into generating synthetic images through methods like distribution alignment and training trajectory tracking for more efficient model training, a significant challenge arises when employing these condensed images practically. Notably, these condensed images tend to be specific to particular models, constraining their versatility and practicality. In response to this limitation, we introduce a novel method, Heterogeneous Model Dataset Condensation (HMDC), designed to produce universally applicable condensed images through cross-model interactions. To address the issues of gradient magnitude difference and semantic distance in models when utilizing heterogeneous models, we propose the Gradient Balance Module (GBM) and Mutual Distillation (MD) with the Spatial-Semantic Decomposition method. By balancing the contribution of each model and maintaining their semantic meaning closely, our approach overcomes the limitations associated with model-specific condensed images and enhances the broader utility. The source code is available in https://github.com/KHU-AGI/HMDC.

**Keywords:** Dataset condensation · Model agnostic · Heterogeneous

## 1 Introduction

In recent years, deep learning [25, 26, 32] has demonstrated a remarkable surge in both effectiveness and applicability across diverse domains [8, 11, 17, 33]. With the increasing depth and complexity of models, the need for substantial datasets has become imperative to sustain their performance and forestall overfitting [5, 19]. Yet, the challenges extend beyond the mere acquisition of huge datasets to management and efficient utilization. In this context, techniques such as dataset distillation (DD) [35] or dataset condensation (DC) [38] have emerged, aiming to address these challenges by offering more efficient data management strategies. These methods not only enable the selection of core-sets [1, 9, 12] capable of maintaining the original performance using a whole dataset but also facilitate a

* Corresponding authors

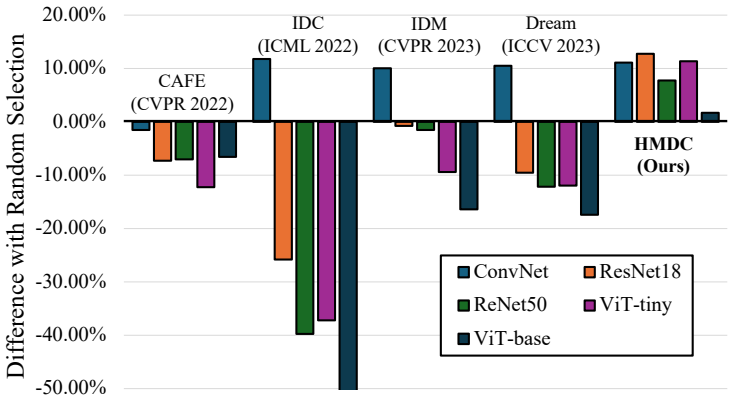



**Fig. 1:** Accuracy plots illustrating the performance of different models trained using images generated by recent dataset condensation methods on the CIFAR-10 dataset with an IPC10 setting. Each bar signifies a performance comparison relative to randomly selected images on 10 images per class, with the initial state of each method identical to that of the random image. Notably, the methods exhibit over-condensation on ConvNet, resulting in performance degradation on other models.

dramatic reduction in dataset size through the creation of synthetic data [2, 3, 21, 28, 30, 34, 35, 38, 40, 41] that accurately represents the original dataset.

Traditionally, dataset condensation methods have employed a compact 3-layered model named ConvNet [38] for the condensation process [2, 28, 34, 40]. The standard evaluation method for assessing the performance of condensed images has been the *ConvNet-to-ConvNet* test, where condensation and evaluation are conducted on ConvNet. Some studies have delved into assessing the general performance of condensed images on other shallow models, such as 3-layered MLP or AlexNet [24]. However, as illustrated in Figure 1, the effectiveness of generated images diminishes when applied to widely used models like ResNet [13] and Vision Transformer (ViT) [8]. This indicates that synthetic images are over-condensed on ConvNet, showing a high model dependence. This dependency significantly constrains the versatility of condensed images. Introducing a completely new model necessitates training a new model and generating a new condensed image, implying that training data must be stored in some way. Consequently, there is a need for a model-agnostic benchmark, not limited to *ConvNet-to-ConvNet*, that operates independently of the specific model.

The primary challenge in achieving model-agnostic dataset condensation lies in identifying the common characteristics within a model and devising an effective method for their extraction. It is difficult to distinguish between features with general recognition information and those excessively tailored to a particular model. To address this challenge, for the first time, we introduce a novel approach that utilizes two models to extract generalized knowledge without bias towards any specific model. To this end, we propose a novel dataset condensation method, Heterogeneous Model Dataset Condensation (HMDC), leveraging

heterogeneous models to extract common features that are more universally applicable.

When naively employing two heterogeneous models simultaneously for dataset condensation, two difficult issues arise. Firstly, there is a problem of gradient magnitude difference, characterized by significant differences in the size of the gradient provided to the synthetic image due to structural or depth variations between the two models. This discrepancy can lead to the disregard of one model or the failure of the image to converge. To alleviate this, we present a Gradient Balance Module (GBM), which accumulates the gradient magnitude of each optimization target, to control the magnitude of the loss. Thus, even if two models have different structures, they can have a similar impact on the synthetic image.

Another challenge arises in the semantic distance resulting from different knowledge between models. As the two models undergo learning, they converge towards optimal points specific to other models, leading to a growing of semantic distance and instability in image convergence. To tackle this, we propose a Mutual Distillation (MD) by Spatial-Semantic Decomposition of the two models and feature matching throughout the process. This process enables consistent updates of synthetic images regardless of the model, avoiding over-condensation on any particular model by obtaining information from different models. This characteristic makes our method effective in a model-agnostic setting.

Our main contributions can be summarized as follows:

- For the first time, we present Heterogeneous Model Dataset Condensation (HMDC) for model-agnostic dataset condensation, which resolves the over-condensation issue to a specific model.
- To facilitate the convergence of synthetic images, we propose the Gradient Balance Module to control a gap between the gradient magnitudes of heterogeneous models.
- We propose Mutual Distillation by Spatial-Semantic Decomposition feature matching of heterogeneous models to fill in the semantic distance between models.
- From the extensive experiments, we demonstrate that our condensed images consistently show great performances from shallow models to widely used large models.

## 2 Related Work

### 2.1 Dataset Condensation

Through the exploration of various core-set selection methodologies [1, 9, 12], it has become evident that synthesized images generated through optimization procedures exhibit greater efficacy compared to direct utilization of real images. Optimization techniques for generating synthetic images primarily fall into two categories: those concerned with tracking training trajectories and those focused on aligning feature distributions. The trajectory tracking approach involves aligning the gradients [21, 28, 38, 41] between real and synthetic images

during the training process, or adapting the weights of the model trained on synthetic images to resemble the model trained on real images [2]. These techniques aim to generate synthetic images based on their influence on the model's learning process. Another stream of condensation methods emphasizes feature distribution matching at intermediate layers [34], or output distribution alignment with the synthetic images [30,35,40,41]. These approaches aim to generate synthetic images by emphasizing feature similarity within the synthetic image but they still depend on utilizing an intermediate training state of the model. In this work, we use the gradient matching method, which has shown better performance in previous studies [21,28,38,41]. In the existing studies [2,28,34,40], synthetic images were typically generated using small 3-layer models, resulting in images that exhibited limited compatibility with other models. In contrast to conventional approaches, our method employs two distinct models to generate a balanced condensed image that avoids being overly biased toward either model, addressing limitations observed in earlier approaches.

### 2.2 Knowledge Distillation

Knowledge Distillation (KD), as introduced by Hinton et al. [16], is a technique in machine learning where a smaller model known as the student model trains to replicate the behavior of a larger model which is the teacher model. This process is done to transfer the knowledge and generalization capabilities of the teacher model to the smaller and more efficient student model. This transfer of knowledge in KD is achieved by aligning what is often referred to as dark knowledge, which can manifest as either logits [10,29,36,39] or features [14,15,18,22].

While the primary objective of KD is to create a more efficient smaller model that performs similarly to a larger one, in this study, we leverage KD to specifically reduce the semantic distance between models. This approach enables learning from a single image through the knowledge of two distinct models, thereby ensuring stable learning without the risk of collapse.

### 2.3 Utilization of Heterogeneity

Previous methods for adjusting loss or gradient have primarily focused on using the uncertainty [20] or norm of the gradient [4] to balance multi-task learning or employing multiple adaptors for domain-robust models [27]. In this work, we claim that while there is a single task, heterogeneity is necessary to solve it effectively. We decompose the features of the image model into spatial and semantic information, allowing us to simultaneously leverage the knowledge of two models with distinct features. By accumulating the gradient norm, we can identify differences in the average gradient and appropriately scale it to inject more general features into the synthetic images, thereby compensating for the imbalance in learning caused by the structures of models.

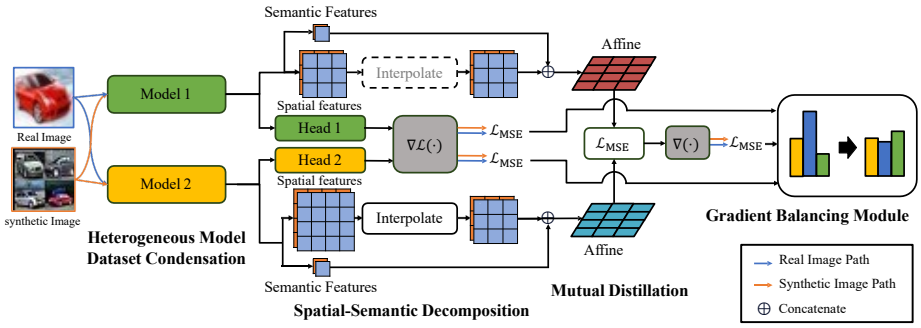



**Fig. 2:** Diagram of Heterogeneous Model Dataset Condensation (HMDC), where two distinct models are employed for feature extraction. These features undergo dimension adjustment through Spatial-Semantic Decomposition, a critical step facilitating Mutual Distillation, and enhancing knowledge sharing between the two models. Throughout the dataset condensation process, the compensatory Gradient Balance Module comes into play, mitigating gradient variations inherent to different models. This module ensures the extraction of general knowledge by harmonizing gradient magnitudes, thus contributing to a more universally applicable condensation process.

## 3 Method

### 3.1 Problem Formulation

Dataset Condensation (DC) [38] is an approach that aims at creating a synthetic dataset denoted as $\mathcal{S} = (\mathbf{x}_i, y_i)_{i=1}^{|\mathcal{S}|}$ from the complete training data $\mathcal{T} = (\mathbf{x}_i, y_i)_{i=1}^{|\mathcal{T}|}$, where $|\mathcal{S}| \ll |\mathcal{T}|$. This synthetic dataset, $\mathcal{S}$, is designed to train a model to a performance level comparable to what could be achieved with the original data. The dataset $\mathcal{S}$ can encompass a subset of $\mathcal{T}$. However, recent research has unveiled that the utilization of synthetic data yields superior performance [2, 3, 21, 28, 30, 34, 35, 38, 40, 41]. In many cases, the central challenge in dataset condensation revolves around the determination of the optimization target, denoted as $\phi(\mathbf{x}, y)$, for the condensed image set $\mathcal{S}$. This can be formally expressed as an optimization problem:

$$\mathcal{S} = \arg\min_{\mathcal{S}} \sum_{i=1}^{|\mathcal{B}|} \mathcal{D}(\phi(\mathbf{x}_i^t, y_i^t), \phi(\mathbf{x}_i^s, y_i^s)), \ (\mathbf{x}_i^t, y_i^t)_{i=1}^{|\mathcal{B}|} \sim \mathcal{T}, \ (\mathbf{x}_i^s, y_i^s)_{i=1}^{|\mathcal{B}|} \sim \mathcal{S}, \quad (1)$$

where $\mathcal{D}(\cdot, \cdot)$ represents a matching function, such as Euclidean or cosine distance; commonly, Mean Squared Error (MSE) is used, and $\mathcal{B}$ represents a mini-batch. $\phi$ is typically associated with features, gradients, or weights of the model. For model-agnostic DC, we utilize a gradient-based method that involves two heterogeneous models, enabling the condensed image to acquire more generalized knowledge. In the following sections, we introduce Heterogeneous Model Dataset Condensation (HMDC) along with methods to find an appropriate optimization target.

### 3.2 Heterogeneous Model Dataset Condensation

Traditional methods for gradient-based dataset condensation typically utilize the gradient of the model's cross-entropy loss as the optimization target [38, 40], denoted as $\phi(\mathbf{x}, y)$. This is expressed mathematically as:

$$\phi(\mathbf{x}, y) = \nabla \mathcal{L}_{\text{CE}}(f_\theta(\mathbf{x}), y). \tag{2}$$

Here, $f_\theta$ represents a model parameterized by $\theta$ and $\mathcal{L}_{\text{CE}}$ is the cross-entropy loss. However, these approaches exhibit a model-dependent nature since they focus on training the model's path on the image. In contrast, our HMDC seeks to extract the common features by concurrently considering the training paths of two models, $f_{\theta_1}$ and $f_{\theta_2}$, where two models complement their features each other as illustrated in Figure 2. This is expressed mathematically as:

$$\begin{aligned}
\mathcal{S} = \arg\min_{\mathcal{S}} \Bigg\{ & \sum_{i=1}^{|\mathcal{B}|} \mathcal{D}\left(\nabla \mathcal{L}_{\text{CE}}\left(f_{\theta_1}\left(\mathbf{x}_i^t\right), y_i^t\right), \nabla \mathcal{L}_{\text{CE}}\left(f_{\theta_1}\left(\mathbf{x}_i^s\right), y_i^s\right)\right) \\
& + \sum_{i=1}^{|\mathcal{B}|} \mathcal{D}\left(\nabla \mathcal{L}_{\text{CE}}\left(f_{\theta_2}\left(\mathbf{x}_i^t\right), y_i^t\right), \nabla \mathcal{L}_{\text{CE}}\left(f_{\theta_2}\left(\mathbf{x}_i^s\right), y_i^s\right)\right) \Bigg\} \\
= \arg\min_{\mathcal{S}} & \left\{ \sum_{i=1}^{|\mathcal{B}|} \mathcal{L}^1 + \sum_{i=1}^{|\mathcal{B}|} \mathcal{L}^2 \right\},
\end{aligned} \tag{3}$$

where $\mathcal{L}^1$ and $\mathcal{L}^2$ mean the optimization target about $f_{\theta_1}$ and $f_{\theta_2}$ respectively, in this case, the mean squared error between gradients generated by synthetic and real images in each model. To enhance performance, additional regularization terms can be introduced as $\mathcal{L}^3, ..., \mathcal{L}^k$. This formulation envisions a dataset condensation method that straightforwardly utilizes two models. However, two key challenges arise the gradient magnitude difference and semantic distance between the models. We address these challenges with our novel Gradient Balance Module (GBM) and Mutual Distillation (MD) with Spatial-Semantic Decomposition (SSD), which are described next.

### 3.3 Gradient Balance Module

The optimization of Eq. 3 faces new challenges due to a significant disparity in the gradient magnitude among $\nabla \mathcal{L}^1, \nabla \mathcal{L}^2, ..., \nabla \mathcal{L}^k$. This discrepancy can stem from various factors, such as differences in model structure, depth, and feature dimensionality. Furthermore, if there are additional optimization targets, a hyperparameter search becomes necessary. The varying magnitudes in these gradients could potentially lead to the neglect of one side or hinder convergence.

To address this, we propose the gradient balance module as illustrated in Figure 2, which sets up an accumulator to store the size of the gradient from

each optimization target. The accumulator $\mathcal{A} = \{a_1, a_2, ..., a_k\} \in \mathbb{R}^k$ is expressed as:

$$\mathcal{A} = \left[ \sum_{s=1}^{S} \max\left(\left|\nabla \mathcal{L}^1(s)\right|\right), \sum_{s=1}^{S} \max\left(\left|\nabla \mathcal{L}^2(s)\right|\right), ..., \sum_{s=1}^{S} \max\left(\left|\nabla \mathcal{L}^k(s)\right|\right) \right], \quad (4)$$

where $S$ represents the total number of optimization steps, an aforementioned $k$ is the number of the optimization targets, $\max(\cdot)$ represents the maximum scalar value in a given tensor, and there are $k$ optimization targets. During the optimization, the reciprocal of the normalized accumulator is multiplied to ensure a similar gradient amplitude for each element:

$$\mathcal{S} = \arg\min_{\mathcal{S}} \left\{ \left[\mathcal{L}^1, \mathcal{L}^2, ..., \mathcal{L}^k\right] \cdot \min(\mathcal{A})\mathcal{A}^{\mathrm{R}} \right\}, \quad (5)$$

where $\min(\cdot)$ represents the minimum scalar value in a given tensor, $\mathcal{A}^{\mathrm{R}}$ represents element-wise reciprocal of vector $\mathcal{A}$. In our study, we utilize three distinct losses as optimization targets. To address the imbalance between images in the update and prevent image collapse, we normalize the gradient of each synthetic image. To manage the computational intensity associated with additional gradient computations, we apply an accumulation strategy once in a while, *e.g.*, by sampling once every 10 steps.

### 3.4 Mutual Distillation with Spatial-Semantic Decomposition

Eq. 3 designed to track the learning path of each model through the gradient matching. However, if the meanings of the two features differ significantly, it impedes the effective learning of synthetic images and hinders their convergence. To mitigate this, we propose a new mutual distillation loss that restricts the semantic divergence between the two models in training processes. On top of that, since two different models vary in depth, dimensionality, and the number of features, effective distillation between them requires careful consideration.

To address this challenge, we introduce a Spatial-Semantic Decomposition (SSD) to preserve the semantics of the features to the maximum extent possible. It decomposes the features of a model into spatial and semantic parts and transforms them accordingly, allowing you to compare the semantics of two models with a linear projection. We designate the features related to classification as semantic features, representing the entire image, and those characterizing each spatial location as spatial features. As spatial features can be represented in an image-like form, features of varying sizes can be aligned using bilinear interpolation, simplifying the alignment process without requiring complex transformations. For easy understanding, here we explain our proposed method using two example models below Vision Transformer (ViT) [8] and Convolutional Neural Network (CNN) [26]. Note that two heterogeneous models can vary, not limited to them.

**ViT Architecture.** First, in the ViT architecture, we utilize a class token (CLS) attached to the front of input tokens as semantic features, and the other image tokens as spatial features. Given the feature from every layer of ViT in the dimension of $\mathcal{F}_{\text{ViT}} \in \mathbb{R}^{n\times(w_{\text{ViT}}h_{\text{ViT}}+1)\times d_{\text{ViT}}}$:

$$\begin{aligned}\mathcal{F}_{\text{ViT}}^{\text{sementic}} &= \mathcal{F}_{\text{ViT}}\left[:,0,:\right] \quad \in \mathbb{R}^{n\times 1\times d_{\text{ViT}}},\\ \mathcal{F}_{\text{ViT}}^{\text{spatial}} &= \mathcal{F}_{\text{ViT}}\left[:,1:,:\right] \in \mathbb{R}^{n\times(w_{\text{ViT}}\times h_{\text{ViT}})\times d_{\text{ViT}}},\end{aligned} \tag{6}$$

where $n$ means the number of layers, $w_{\text{ViT}}$ and $h_{\text{ViT}}$ means the number of tokens generated by the ViT when it performs patch embedding to generate image tokens in the horizontal and vertical directions, respectively. The CLS token is attached to the front, and $d_{\text{ViT}}$ is the dimension of each token. Given the variations in the size of spatial features across layers and models, standardization is achieved through bilinear interpolation. To address differences in feature dimensionality, we apply a learnable affine transformation. The dimension-aligned feature is expressed as:

$$\begin{aligned}\mathcal{F}_1 = \Bigg[ & W_1^1 \left[\mathcal{F}_{\text{ViT}}^{\text{sementic}}[1]; \underset{w\times h}{\text{I}}\left(\mathcal{F}_{\text{ViT}}^{\text{spatial}}[1]\right)\right] + \mathbf{b}_1^1;\\ & ...; W_n^1 \left[\mathcal{F}_{\text{ViT}}^{\text{sementic}}[n]; \underset{w\times h}{\text{I}}\left(\mathcal{F}_{\text{ViT}}^{\text{spatial}}[n]\right)\right] + \mathbf{b}_n^1\Bigg],\end{aligned} \tag{7}$$

where $n$ is the number of layers in ViT, $W_1^1, ..., W_n^1 \in \mathbb{R}^{d\times d_{\text{ViT}}}$, $\mathbf{b}_1^1, ..., \mathbf{b}_n^1 \in \mathbb{R}^d$, and $\underset{w\times h}{\text{I}}$ refers to bilinear interpolation into the size of $w \times h$. As a result, we get the feature of $\mathbb{R}^{n\times(wh+1)\times d}$. $w$, $h$, and $d$ mean a target interpolation width, height, and dimension respectively. We use dimension and interpolation size into smaller values between two models, which is found to be empirically better. For computational convenience, it is transposed to $\mathbb{R}^{(wh+1)\times d\times n}$.

**CNN Architecture.** In the case of CNN features, inherently semantic features can be generated through mean pooling of spatial features. This process is mathematically expressed as:

$$\mathcal{F}_{\text{CNN}} = \{\mathcal{F}_{\text{CNN}}^1, ..., \mathcal{F}_{\text{CNN}}^m\}, \quad \mathcal{F}_{\text{CNN}}^l \in \mathbb{R}^{d_l\times w_l\times h_l}. \tag{8}$$

$$\begin{aligned}\mathcal{F}_{\text{CNN}}^{\text{sementic}} &= \left\{\frac{1}{w_1h_1}\sum_{w=1}^{w_1}\sum_{h=1}^{h_1}\mathcal{F}_{\text{CNN}}^1\left[:,w,h\right], ..., \frac{1}{w_mh_m}\sum_{w=1}^{w_m}\sum_{h=1}^{h_m}\mathcal{F}_{\text{CNN}}^m\left[:,w,h\right]\right\},\\ \mathcal{F}_{\text{CNN}}^{\text{spatial}} &= \mathcal{F}_{\text{CNN}},\end{aligned} \tag{9}$$

where m is the number of layers in CNN. As observed in the case of ViT, we standardize dimension and feature size via bilinear interpolation and affine transformation. This process is expressed as:

$$\begin{aligned}\mathcal{F}_2 = \Bigg[ & W_1^2 \left[\mathcal{F}_{\text{CNN}}^{\text{sementic}}[1]; \underset{w\times h}{\text{I}}\left(\mathcal{F}_{\text{CNN}}^{\text{spatial}}[1]\right)\right] + \mathbf{b}_1^2;\\ & ...; W_m^2 \left[\mathcal{F}_{\text{CNN}}^{\text{sementic}}[m]; \underset{w\times h}{\text{I}}\left(\mathcal{F}_{\text{CNN}}^{\text{spatial}}[m]\right)\right] + \mathbf{b}_m^2\Bigg],\end{aligned} \tag{10}$$

where $W_1^2 \in \mathbb{R}^{d\times d_1}, ..., W_m^2 \in \mathbb{R}^{d\times d_m}$, and $\mathbf{b}_1^2, ..., \mathbf{b}_m^2 \in \mathbb{R}^d$. Finally we get the feature of $\mathbb{R}^{m\times(wh+1)\times d}$. And, it is transposed to $\mathbb{R}^{(wh+1)\times d\times m}$.

The aligned dimensionality and the number of features are currently the same, but the number of layers in the two models is still different, *i.e.*, $\mathcal{F}_1 \in \mathbb{R}^{(wh+1)\times d\times n}$, $\mathcal{F}_2 \in \mathbb{R}^{(wh+1)\times d\times m}$. We further match the number of layers by using an $n \times m$ matrix $M_{\text{layer}} \in \mathbb{R}^{n\times m}$, where $n$ represents the number of layers in one model and $m$ denotes the number of layers in the other. Softmax is applied to this matrix for layer selection. We call this matching process as Spatial-Semantic Decomposition (SSD), where the matched features can be expressed as follows:

$$\begin{cases} \mathcal{F}_1, \mathcal{F}_2 \cdot \text{softmax}(M_{\text{layer}}^{\text{T}}) & m > n \\ \mathcal{F}_1 \cdot \text{softmax}(M_{\text{layer}}), \mathcal{F}_2 & \text{otherwise} \end{cases}. \tag{11}$$

Here, we align the number of layers to the smaller one, which is found to be empirically better. This gives us two $\mathbb{R}^{(wh+1)\times d\times n}$ of dimension-aligned features when $m > n$ or $\mathbb{R}^{(wh+1)\times d\times m}$ dimension of feature otherwise. Both the feature affine matrixes ($W^1$ and $W^2$) and the layer matching matrix ($M_{\text{layer}}$) undergo training to ensure feature alignment between the models at each step. Note again that we give an example using CNN and ViT, but our method can be extended to any model with spatial features.

The Spatial-Semantic Decomposition method enables a comparison of the distinct features of two models. Throughout the training process, we propose Mutual Distillation (MD) to align the meanings of these features, aiming to make them similar knowledge across both models. The training loss for each model $f_1$ and $f_2$ is expressed as:

$$\begin{aligned} \mathcal{L}_{\text{MD}}(\mathbf{x}) &= \text{MSE}(\text{SSD}(f_{\theta_1}(\mathbf{x}), f_{\theta_2}(\mathbf{x}))), \\ \mathcal{L}_{f_1} &= \mathcal{L}_{\text{CE}}(f_{\theta_1}(\mathbf{x}), y) + \mathcal{L}_{\text{MD}}(\mathbf{x}), \\ \mathcal{L}_{f_2} &= \mathcal{L}_{\text{CE}}(f_{\theta_2}(\mathbf{x}), y) + \mathcal{L}_{\text{MD}}(\mathbf{x}). \end{aligned} \tag{12}$$

To guide the image in learning intermediate information, the Mutual Distillation loss serves as an additional regularization term for the synthetic image. Through this process, each semantic aspect of the model and the learning path of the synthetic image are guided, achieving a balance between the two models and enhancing generality in the results. Consequently, the number of optimization target $k$ for condensed images is 3 in this case. Total loss function for condensed images can be defined as a vector inner product as follows:

$$\begin{aligned} \mathbf{L} &= \left[\mathcal{L}^1, \mathcal{L}^2, \text{MSE}\left(\nabla\mathcal{L}_{\text{MD}}(\mathbf{x}^t), \nabla\mathcal{L}_{\text{MD}}(\mathbf{x}^s)\right)\right], \\ \mathcal{L}_{\text{target}} &= \mathbf{L} \cdot \min(\mathcal{A})\,\mathcal{A}^{\text{R}} \\ &= \frac{\min(\mathcal{A})}{a_1}\mathcal{L}_1 + \frac{\min(\mathcal{A})}{a_2}\mathcal{L}_2 + \frac{\min(\mathcal{A})}{a_3}\text{MSE}\left(\nabla\mathcal{L}_{\text{MD}}(\mathbf{x}^t), \nabla\mathcal{L}_{\text{MD}}(\mathbf{x}^s)\right), \end{aligned} \tag{13}$$

where $a_1, a_2$, and $a_3$ is current value in the accumulator $\mathcal{A}$ part of the Gradient Balance Module. This allows us to learn by considering the semantic distance

between two models when training a model and when training an image, and to extract more general knowledge from it.

## 4 Experiments

### 4.1 Implementation Details

In this study, we performed a comprehensive comparative analysis, assessing the effectiveness of our proposed method against cutting-edge gradient matching techniques, IDC [21] and DREAM [28]. To broaden our evaluation scope to distribution matching, we selected CAFE [34] and IDM [40] as benchmark methodologies.

To ensure a fair and standardized comparison, we employed a consistent augmentation strategy across all methods. This strategy encompassed a sequence of color modifications, cropping, and either Cutout [7] or CutMix [37], aligning with the recommended practices outlined in IDC [40]. Also, we use multi-formation which is used in IDC and IDM [40] for every method. Nonetheless, it appears that while this robust augmentation technique enhances the performance of the gradient-based method, it adversely affects the distribution matching method, resulting in a decline in performance. In the images generated during the process, a noticeable inclination to produce corrupted images under intense augmentation was observed.

To measure how the condensed image effectively trains the large model, We conducted experiments on CIFAR10 [23] on Images Per Class (IPC) 1, 10, 50. Our assessment followed the experimental procedures detailed in each referenced paper. For evaluation, we utilized ConvNet, ViT-tiny [8], ResNet18 [13], ViT-small, ResNet50, ResNet101, and ViT-base models, each with specific learning rates (0.01, 0.001, 0.001, 0.001, 0.001, 0.0001, and 0.0001, respectively). The best scores were measured as performance metrics during the training of the models for a consistent duration of 2,000 epochs. Because training a large model on a small dataset makes it difficult to compare performance, every model in Table 1 except ConvNet was pre-trained on the ImagiNet-1K [6] dataset.

In our approach, we adopt ConvNet and ViT-tiny as heterogeneous models. We configure the iteration parameter to 100 and set the loop to iterate 100 times for each iteration. Within each iteration, we perform updates to the image, update the model, and adjust the affine layer along with the layer-matching matrix. The learning rate of each model is 0.001, and the affine layer and layer-matching matrix are 0.01. Both use SGD optimizer [31] as follows prior works. We set the batch size to 128.

### 4.2 Results

Table 1 presents a comprehensive performance comparison of the condensed images generated by each technique on CIFAR10. The proposed HMDC demonstrates commendable performance across most models, except ConvNet. Notably,

**Table 1:** Experimental results of dataset condensation methods on CIFAR-10. HMDC is the best or second-best performer in most cases.

| IPC | Methods | Models (#Params) | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | ConvNet (0.3M) | ResNet18 (11M) | ResNet50 (22M) | ResNet101 (43M) | CNN Average | ViT-tiny (5.5M) | ViT-small (21M) | ViT-base (86M) | ViT Average | Average |
| 1 | Random | 22.51±0.30 | 39.39±3.43 | 39.95±8.40 | 51.17±1.04 | 38.26±3.29 | 32.49±3.59 | **58.33±1.42** | 41.05±5.99 | 43.96±3.66 | 40.70±3.45 |
| | CAFE | 25.87±0.84 | 23.61±2.27 | 27.07±3.15 | 24.42±1.28 | 25.24±1.89 | 18.43±3.37 | 22.80±3.00 | 23.84±7.77 | 21.69±2.65 | 23.72±3.10 |
| | IDM | 32.86±0.24 | 18.92±1.69 | 19.20±1.60 | 15.83±7.37 | 21.70±2.72 | 18.01±1.57 | 15.90±2.78 | 12.81±4.90 | 15.58±1.68 | 19.08±2.88 |
| | IDC | 37.48±0.29 | 18.81±2.44 | 16.85±0.96 | 11.15±0.58 | 21.07±1.07 | 10.82±1.81 | 10.82±1.81 | 14.93±1.74 | 12.72±1.01 | 12.72±1.01 |
| | DREAM | 37.85±0.35 | 22.50±2.89 | 19.47±3.71 | 8.82±1.24 | 22.16±2.04 | 14.53±1.26 | 10.93±1.32 | 11.93±4.23 | 12.46±2.27 | 18.00±2.14 |
| | HDMC | **38.74±0.37** | **52.45±2.78** | **57.40±4.40** | **51.32±0.88** | **49.98±2.11** | **39.25±6.05** | 53.50±7.22 | **49.43±10.9** | **47.39±2.51** | **48.87±4.65** |
| 10 | Random | 36.45±0.12 | 56.59±4.86 | 69.55±9.69 | **82.66±1.82** | 61.31±4.12 | 59.36±9.19 | **90.11±1.25** | 81.26±5.66 | 76.91±5.37 | 68.00±4.66 |
| | CAFE | 34.92±0.63 | 49.30±5.24 | 62.51±4.20 | 67.91±4.28 | 53.66±3.59 | 47.10±5.89 | 89.22±2.48 | 74.69±11.80 | 70.33±4.72 | 60.81±4.93 |
| | IDM | 46.49±0.17 | 55.82±1.16 | 68.01±2.41 | 60.42±6.39 | 57.68±2.53 | 49.96±2.75 | 68.61±9.74 | 64.87±13.67 | 61.15±5.53 | 59.17±5.18 |
| | IDC | **48.22±0.34** | 30.75±1.78 | 29.79±5.94 | 25.71±2.58 | 33.62±2.66 | 22.14±1.20 | 29.79±2.92 | 27.10±4.90 | 26.35±1.85 | 30.50±2.81 |
| | DREAM | 47.62±0.53 | 44.26±1.78 | 50.70±6.11 | 45.68±1.97 | 47.07±2.60 | 36.26±2.62 | 55.95±15.59 | 56.69±14.05 | 49.64±10.76 | 48.17±6.09 |
| | HDMC | 47.96±0.09 | **69.87±0.12** | **77.29±1.73** | 82.25±0.93 | **69.34±0.72** | **73.55±4.24** | 89.01±1.42 | **85.38±1.45** | **82.64±1.62** | **75.04±1.43** |
| 50 | Random | 45.47±0.39 | 73.55±1.02 | 80.96±6.48 | 91.33±0.57 | 72.83±2.11 | 71.68±5.66 | **96.19±0.10** | 94.68±1.88 | 87.52±2.54 | 79.13±2.30 |
| | CAFE | 43.70±0.08 | 75.86±0.28 | **85.47±1.59** | **92.15±0.22** | 74.29±0.54 | **77.77±2.33** | 89.63±1.07 | **94.77±1.84** | **96.35±0.29** | **80.86±0.94** |
| | IDM | 49.43±0.08 | 73.51±1.27 | 82.98±3.03 | 91.39±0.15 | 74.33±1.14 | 75.18±3.12 | 95.66±0.30 | 94.06±1.26 | 88.30±1.56 | 80.32±1.32 |
| | IDC | **52.90±0.23** | 63.69±0.69 | 72.60±0.49 | 67.15±12.6 | 64.09±3.50 | 53.53±3.87 | 77.39±15.5 | 74.41±12.0 | 68.44±5.97 | 65.95±6.48 |
| | DREAM | 52.63±0.34 | **76.00±3.05** | 85.04±0.69 | 90.11±0.89 | **75.95±1.24** | 69.25±5.03 | 79.75±1.92 | 92.02±2.57 | 84.81±2.10 | 93.17±0.84 |
| | HDMC | 51.94±0.48 | 75.87±0.56 | 83.75±0.96 | 88.91±0.36 | 75.12±0.59 | 74.76±4.32 | 91.03±0.45 | 91.34±0.68 | 85.71±2.17 | 79.66±1.12 |

other techniques generally exhibit inferior performance compared to Random across all models, except for ConvNet. The described methods collaboratively yield a condensed image that is impactful without inducing over-condensation, particularly evident in the case of ConvNet. This trend becomes more pronounced as IPC decreases. Simultaneously, our method competes favorably with other methods on ConvNet. Noteworthy is that HMDC performs the best on IPC 1 for all models. This suggests that the proposed HMDC effectively captures general features and incorporates them into a limited synthetic image. Unlike previous methods, HMDC shows promise for training large models from images compressed from relatively small models, aligning with the goal of dataset condensation. Despite utilizing two models, HMDC requires only 100 iterations and consumes less time than other models, typically using 1,200 to 20,000 iterations.

### 4.3 Ablation Studies

We conducted an ablation study to evaluate the impact of the proposed features on performance. Specifically, we examined performance by removing Mutual Distillation by Spatial-Sementic Decomposition and the Gradient Balance Module. Table 2 presents the experimental result of ablation. The experimental results demonstrate that each element contributes to performance, and when both methods are employed, they exhibit synergy.

In the Ablation Study, the results demonstrate that the isolated application of the Gradient Balance Module (GBM) and Mutual Distillation (MD) can be beneficial in certain scenarios, though they tend to favor either Convolutional Neural Network (CNN) model. This leads to good performance on certain models and poor performance on others. The isolated use of the GBM tends to favor

**Table 2:** Table illustrating the outcomes of the ablation study. GBM means Gradient Balance Module and MD means Mutual Distillation by Spatial-Semantic Decomposition. The results demonstrate the individual contributions of the presented factors to performance enhancements, revealing a synergistic effect when employing them simultaneously.

| GBM | MD | CIFAR-10 (IPC 10) | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | ConvNet (0.3M) | ResNet18 (11M) | ResNet50 (22M) | ResNet101 (43M) | CNN Average | ViT-tiny (5.5M) | ViT-small (21M) | ViT-base (86M) | ViT Average | Average |
| | | 46.80±0.16 | **72.33±3.82** | 76.00±3.62 | 79.51±0.78 | 68.66±2.09 | 73.05±0.78 | **89.58±1.89** | 81.12±1.10 | 81.25±0.57 | 74.05±1.74 |
| ✓ | | 47.13±0.48 | 72.19±1.66 | 74.50±3.87 | 76.19±8.40 | 67.50±3.60 | 69.72±10.0 | 83.15±9.50 | 78.90±13.4 | 77.26±2.13 | 71.68±6.76 |
| | ✓ | 47.30±0.41 | 71.74±1.30 | **78.05±2.23** | 80.61±2.93 | **69.43±1.71** | 71.06±3.89 | 86.18±5.22 | 80.24±5.54 | 79.16±0.87 | 73.60±3.07 |
| ✓ | ✓ | **47.54±0.73** | 69.34±0.91 | 77.29±1.73 | **81.82±0.99** | 69.00±1.09 | **73.55±4.24** | 89.01±1.42 | **85.38±1.45** | **82.64±1.62** | **74.85±1.64** |

**Fig. 3:** Comparision of condensed images between DREAM [28] and HMDC(Ours)

**((a))** Condensed image generated by DREAM [28] method, with CIFAR10, 10 images per class.

**((b))** Condensed image generated by Heterogeneous Model Dataset Condensation (HMDC) method, with CIFAR10, 10 images per class.

smaller models because the semantic distance between the two models fails to encapsulate the complex patterns necessary for larger models. Conversely, when only MD is employed, it extracts more generalized features, performing well with larger models. However, the results are biased due to the gradient difference in the condensed model. By combining these methods, a balanced gradient is achieved for each model, significantly reducing the semantic distance between them. This synergy not only enhances the overall performance but also ensures a more model-agnostic improvement.

### 4.4 Qualitative Results

Figure 3(a) presents a condensed image generated using the DREAM [28] method. In contrast, Figure 3(b) depicts a condensed image created using the Heterogeneous Model Dataset Condensation approach. Overall, we observe that objects exhibit the same characteristics, such as increased contrast and sharpened

**Table 3:** The performance variations across model combinations, depicting results for pairs of identical models (CNN + CNN) and pairs involving larger models than those employed in the experiment.

| ConvNet | ResNet18 | ViT-tiny | ViT-small | CIFAR-10 (IPC 10) | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | ConvNet (0.3M) | ResNet18 (11M) | ResNet50 (22M) | ResNet101 (43M) | CNN Average | ViT-tiny (5.5M) | ViT-small (21M) | ViT-base (86M) | ViT Average | Average |
| ✓ | | ✓ | | **47.96±0.09** | 69.87±0.12 | 77.29±1.73 | 82.25±0.93 | 69.34±0.72 | 73.55±4.24 | 89.01±1.42 | 85.38±1.45 | 82.64±1.62 | 75.04±1.43 |
| ✓ | ✓ | | | 47.47±0.08 | 45.83±2.56 | 52.15±1.20 | 38.81±9.78 | 46.07±3.41 | 38.71±5.54 | 50.89±13.08 | 43.99±16.42 | 44.53±5.57 | 45.41±6.95 |
| | ✓ | | ✓ | 38.35±0.62 | **76.38±0.55** | **85.62±0.28** | **89.05±0.28** | **72.35±0.43** | **82.83±1.65** | **93.71±0.57** | **90.42±2.26** | **88.99±0.85** | **79.48±0.89** |

**Table 4:** Comparison table between simply using ViT-Tiny and using the presented method.

| Methods | ConvNet | ViT-tiny | CIFAR-10 (IPC 10) | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | ConvNet (0.3M) | ResNet18 (11M) | ResNet50 (22M) | ResNet101 (43M) | CNN Average | ViT-tiny (5.5M) | ViT-small (21M) | ViT-base (86M) | ViT Average | Average |
| Random | | | 36.45±0.12 | 56.59±4.86 | 69.55±9.69 | **82.66±1.82** | 61.31±4.12 | 59.36±9.19 | **90.11±1.25** | 81.26±5.66 | 76.91±5.37 | 68.00±4.66 |
| Dream | ✓ | | 47.62±0.53 | 44.26±1.78 | 50.70±6.11 | 45.68±1.97 | 47.07±2.60 | 36.26±2.62 | 55.95±15.59 | 56.69±14.05 | 49.64±10.76 | 48.17±6.09 |
| Dream | | ✓ | 25.69±0.58 | 48.38±1.72 | 54.27±6.59 | 60.57±2.73 | 47.23±2.91 | 56.32±5.17 | 60.00±12.63 | 51.80±22.49 | 56.04±8.69 | 51.00±7.42 |
| HMDC | ✓ | ✓ | **47.96±0.09** | **69.87±0.12** | **77.29±1.73** | 82.25±0.93 | **69.34±0.72** | **73.55±4.24** | 89.01±1.42 | **85.38±1.45** | **82.64±1.62** | **75.04±1.43** |

edges. While the visual disparities are minimal, it is noteworthy that the image generated by our method exhibits fewer artifacts and distortions compared to DREAM. These distortions appear to be caused by over-condensation, improving performance on certain models but degrading performance on others.

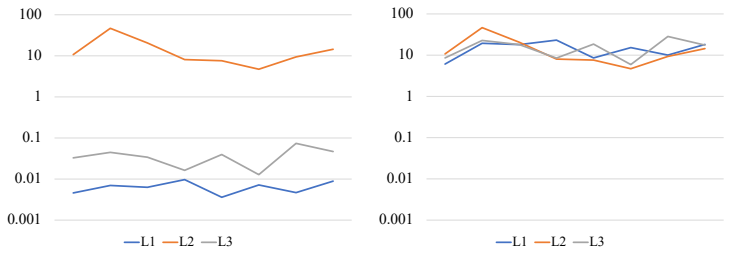
### 4.5 Analysis

Table 3 illustrates the performance outcomes of Heterogeneous Model Dataset Condensation across various model combinations. Significantly, there is an evident decrease in overall performance when the CNN is used uniformly, contrary to the intended objective of the method. In the third row, we adjusted the learning rate of the affine layer and layer-matching matrix to 0.001. This suggests that with minimal modification, larger models can be accommodated using HMDC.

Table 4 presents a comparative analysis of our experimental results with the state-of-the-art method, DREAM, changing the model into ViT-Tiny. The results reveal an overall enhancement in performance for ViT-Tiny attributed to an increased understanding of the model. However, performance drops on ConvNet and it is notably modest compared to using random images. This trend is further emphasized by the large gap between HMDC in combination with ConvNet. Conversely, in ConvNet, overall performance is diminished due to the model's limited capacity, with a pronounced decline in the ViT series. This observation underscores the model dependency of the condensed image and the condensed image is over-condensed.

Figure 4 depicts the maximum gradient magnitude of the synthetic image after the learning step. On the left, before integrating the Gradient Balance Module, there is a substantial disparity in the gradient magnitudes, leading to the neglect of other losses. Following the inclusion of the Gradient Balance Module, the gradient magnitudes from each loss function become uniform. This ensures

**Fig. 4:** A logarithmic plot depicting the gradient magnitude evolution of a synthetic image throughout the training process. L1 and L2 refer to the optimization targets in Eq. 3. L3 is MSE $\left(\nabla\mathcal{L}_{\text{MD}}(\mathbf{x}^t), \nabla\mathcal{L}_{\text{MD}}\right)$.




balanced dataset condensation irrespective of the model, underscoring the essential role of the Gradient Balance Module in Heterogeneous Model Dataset Condensation.

## 5 Conclusion

We identified a model dependency issue in existing dataset condensation methods and proposed a remedy by balancing different heterogeneous models. However, employing two distinct models for dataset condensation introduces two challenges: bias arising from differences in gradient magnitude between the models and semantic distance resulting from the models converging to their respective optimal points. To address the gradient magnitude disparity, we introduced the Gradient Balance Module. To tackle the semantic distance issue, we proposed Mutual Distillation by Spatial-Semantic Decomposition. The combination of these components effectively alleviated the model dependency problem.

However, there are still some limitations. Firstly, it is impossible to surpass the model's capacity limits, as evidenced by the performance gap for ViT-Base due to a fifteenfold difference in parameter number. Furthermore, due to the continued use of floating points in condensed images, they occupy four times the capacity of the equivalent number of real images. From this standpoint, there is a need for quantization-aware condensation that enables the synthetic image to be treated like a real image.

## Acknowledgement

This work was supported by MSIT (Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2024-RS-2023-00258649) supervised by the IITP (Institute for Information & Communications Technology Planning & Evaluation), and in part by the IITP grant funded by the Korea Government (MSIT) (Artificial Intelligence Innovation Hub) under Grant 2021-0-02068, and by the IITP grant funded by the Korea

government (MSIT) (No.RS-2022-00155911, Artificial Intelligence Convergence Innovation Human Resources Development (Kyung Hee University)).

## References

1. Agarwal, P.K., Har-Peled, S., Varadarajan, K.R.: Approximating extent measures of points. J. ACM **51**(4), 606–635 (jul 2004). https://doi.org/10.1145/1008731.1008736, https://doi.org/10.1145/1008731.1008736
2. Cazenavette, G., Wang, T., Torralba, A., Efros, A.A., Zhu, J.Y.: Dataset distillation by matching training trajectories. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4750–4759 (2022)
3. Cazenavette, G., Wang, T., Torralba, A., Efros, A.A., Zhu, J.Y.: Generalizing dataset distillation via deep generative prior. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3739–3748 (2023)
4. Chen, Z., Badrinarayanan, V., Lee, C.Y., Rabinovich, A.: Gradnorm: Gradient normalization for adaptive loss balancing in deep multitask networks. In: International conference on machine learning. pp. 794–803. PMLR (2018)
5. Cubuk, E.D., Zoph, B., Mane, D., Vasudevan, V., Le, Q.V.: Autoaugment: Learning augmentation policies from data. arXiv e-prints p. arXiv preprint arXiv:1805.09501 (2018)
6. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 248–255. Ieee (2009)
7. DeVries, T., Taylor, G.W.: Improved regularization of convolutional neural networks with cutout. arXiv preprint arXiv:1708.04552 (2017)
8. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. In: International Conference on Learning Representations (2020)
9. Feldman, D., Schmidt, M., Sohler, C.: Turning big data into tiny data: Constant-size coresets for k-means, pca, and projective clustering. SIAM Journal on Computing **49**(3), 601–657 (2020)
10. Furlanello, T., Lipton, Z., Tschannen, M., Itti, L., Anandkumar, A.: Born again neural networks. In: International Conference on Machine Learning. pp. 1607–1616. PMLR (2018)
11. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. Advances in neural information processing systems **27** (2014)
12. Har-Peled, S., Mazumdar, S.: On coresets for k-means and k-median clustering. In: Proceedings of the Thirty-Sixth Annual ACM Symposium on Theory of Computing. p. 291–300. STOC '04, Association for Computing Machinery, New York, NY, USA (2004). https://doi.org/10.1145/1007352.1007400, https://doi.org/10.1145/1007352.1007400
13. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
14. Heo, B., Kim, J., Yun, S., Park, H., Kwak, N., Choi, J.Y.: A comprehensive overhaul of feature distillation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 1921–1930 (2019)

15. Heo, B., Lee, M., Yun, S., Choi, J.Y.: Knowledge transfer via distillation of activation boundaries formed by hidden neurons. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 33, pp. 3779–3787 (2019)
16. Hinton, G., Vinyals, O., Dean, J.: Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531 (2015)
17. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. Advances in neural information processing systems **33**, 6840–6851 (2020)
18. Huang, Z., Wang, N.: Like what you like: Knowledge distill via neuron selectivity transfer. arXiv preprint arXiv:1707.01219 (2017)
19. Karystinos, G.N., Pados, D.A.: On overfitting, generalization, and randomly expanded training sets. IEEE Transactions on Neural Networks **11**(5), 1050–1057 (2000)
20. Kendall, A., Gal, Y., Cipolla, R.: Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7482–7491 (2018)
21. Kim, J.H., Kim, J., Oh, S.J., Yun, S., Song, H., Jeong, J., Ha, J.W., Song, H.O.: Dataset condensation via efficient synthetic-data parameterization. In: International Conference on Machine Learning. pp. 11102–11118. PMLR (2022)
22. Kim, J., Park, S., Kwak, N.: Paraphrasing complex network: Network compression via factor transfer. Advances in neural information processing systems **31** (2018)
23. Krizhevsky, A., Hinton, G., et al.: Learning multiple layers of features from tiny images (2009)
24. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems **25** (2012)
25. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. nature **521**(7553), 436–444 (2015)
26. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proceedings of the IEEE **86**(11), 2278–2324 (1998)
27. Li, W.H., Liu, X., Bilen, H.: Universal representation learning from multiple domains for few-shot classification. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 9526–9535 (2021)
28. Liu, Y., Gu, J., Wang, K., Zhu, Z., Jiang, W., You, Y.: Dream: Efficient dataset distillation by representative matching. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 17314–17324 (October 2023)
29. Mirzadeh, S.I., Farajtabar, M., Li, A., Levine, N., Matsukawa, A., Ghasemzadeh, H.: Improved knowledge distillation via teacher assistant. In: Proceedings of the AAAI conference on artificial intelligence. vol. 34, pp. 5191–5198 (2020)
30. Nguyen, T., Novak, R., Xiao, L., Lee, J.: Dataset distillation with infinitely wide convolutional networks. Advances in Neural Information Processing Systems **34**, 5186–5198 (2021)
31. Robbins, H., Monro, S.: A stochastic approximation method. The annals of mathematical statistics pp. 400–407 (1951)
32. Rumelhart, D.E., Hinton, G.E., Williams, R.J., et al.: Learning internal representations by error propagation (1985)
33. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017)
34. Wang, K., Zhao, B., Peng, X., Zhu, Z., Yang, S., Wang, S., Huang, G., Bilen, H., Wang, X., You, Y.: Cafe: Learning to condense dataset by aligning features.

In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12196–12205 (2022)
35. Wang, T., Zhu, J.Y., Torralba, A., Efros, A.A.: Dataset distillation. arXiv preprint arXiv:1811.10959 (2018)
36. Yang, C., Xie, L., Su, C., Yuille, A.L.: Snapshot distillation: Teacher-student optimization in one generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2859–2868 (2019)
37. Yun, S., Han, D., Oh, S.J., Chun, S., Choe, J., Yoo, Y.: Cutmix: Regularization strategy to train strong classifiers with localizable features. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 6023–6032 (2019)
38. Zhao, B., Mopuri, K.R., Bilen, H.: Dataset condensation with gradient matching. In: International Conference on Learning Representations (2020)
39. Zhao, B., Cui, Q., Song, R., Qiu, Y., Liang, J.: Decoupled knowledge distillation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11953–11962 (2022)
40. Zhao, G., Li, G., Qin, Y., Yu, Y.: Improved distribution matching for dataset condensation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7856–7865 (2023)
41. Zhou, Y., Nezhadarya, E., Ba, J.: Dataset distillation using neural feature regression. Advances in Neural Information Processing Systems **35**, 9813–9827 (2022)