Compression Image Dataset Based on Multiple Matrix Product States

Ze-Feng Gao[1,2⋆], Peiyu Liu[1,3⋆], Wayne Xin Zhao[1,3⋆⋆], Zhi-Yuan Xie[2⋆⋆], Ji-Rong Wen[1,3] and Zhong-Yi Lu[2]

[1] Gaoling School of Artificial Intelligence, Renmin University of China,
[2] Department of Physics, Renmin University of China,
[3] Beijing Key Laboratory of Big Data Management and Analysis Methods
{zfgao,liupeiyustu,qingtaoxie,jrwen,zlu}@ruc.edu.cn,
batmanfly@qq.com

Abstract. Large-scale datasets have made impressive progress in deep learning. However, storing datasets and training neural network models on large datasets have become increasingly expensive. In this paper, we present an effective dataset compression approach based on the matrix product states (short as MPS) and knowledge distillation. MPS can decompose image samples into a sequential product of tensors to achieve task-agnostic image compression by preserving the low-rank information of the images. Based on this property, we use multiple MPS to represent the image datasets samples. Meanwhile, we also designed a task-related component based on knowledge distillation to enhance the generality of the compressed dataset. Extensive experiments have demonstrated the effectiveness of the proposed approach in image datasets compression, especially obtaining better model performance (2.26% on average) than the best baseline method on the same compression ratio.

Keywords: dataset compression, matrix product state, deep learning

1 Introduction

Large-scale datasets consisting of millions of samples are becoming the norm to obtain state-of-the-art machine learning models in several fields including speech enhancement and recognition [33,37], computer vision [32] and natural language processing [7]. At such a scale, the resources needed to store datasets and train neural networks grow extremely large, and training machine learning models on it requires the specialized equipment and infrastructure. Therefore, it is the critical problem in machine learning that effectively reduces the size of the datasets as well as maintains the model performance.

An intuitive way is to compress the samples one by one through low-rank approximation [4, 22], in order to keep the maximum possible information in each sample.

⋆ Authors contributed equally.
⋆⋆ Corresponding author.

However, the compressed sample may consist of noise that is irrelevant to the downstream task. Motivated by this shortcoming, there comes another research line that focuses on compressing datasets for both high compression ratio and comparable performance with a full set of downstream tasks. There are two widely used methods,*i.e.*, *data selection* and *dataset distillation*. They compress the dataset by identifying the most representative training samples in the dataset and generating a small training set, respectively. Nevertheless, the selection and generation of small sets are task-specific and require additional computational costs when retraining for different tasks. It is a crucial problem that implements an effective image dataset compression method that has a high compression ratio while reducing the computational cost required for task migration.

In this paper, we introduce a novel matrix product states (MPS) [8] based technique for compressing image datasets. MPS is a low-rank decomposition method that was originally used to describe “short-range correlation” information - similar to the "locality of pixel dependencies" in an image - in the study of quantum many-body physics problems. Technically, the MPS can factorize a matrix into a sequential product of local tensors (*i.e.,* a multi-way array). An important merit of MPS decomposition is that it can effectively represent the low-rank information of the matrix [4]. Furthermore, it is more efficient to use multiple MPS to represent the same matrix than a single MPS. This is because the ability to represent low-rank information using multiple MPS tensors is superior to that of a single MPS [43]. Moreover, dataset distillation can synthesize a small dataset to represent task-relevant features, which leads to a compressed dataset. Such properties motivate us to propose a better dataset compression approach, which bases on multiple MPS representations and knowledge distillation. We can compress the dataset by multiple MPS to represent low-rank information in task-agnostic dataset compression, while use the knowledge distillation to supplement the information in task-specific training.

To this end, we propose a Multiple MPS-based dataset compression approach, called *MMPS*, to compress the image dataset. The MMPS approach not only enables deep neural networks to obtain similar performance as on the original dataset but also can be used for different models as well as different types of tasks. We have made two import technical contributions for image dataset compression based on MPS and knowledge distillation. First, we introduce a new task-agnostic dataset compression approach that efficiently represents low-rank information among pixels. We formulate this goal as the problem of minimizing the difference between multiple low-rank tensors with constraints and the original image samples. We present both theoretical discussion and experimental verification for the effectiveness of this dataset compression approach. Second, we propose a new task-specific component for information supplementation, tailored for the machine learning model. In general, different downstream tasks have different information for image datasets, the offline dataset compression does not contain task-specific information. Thus, we propose a module based on knowledge distillation to make the compressed datasets adaptable to different tasks.

To the best of our knowledge, it is the first time that multiple MPS is applied to the dataset compression, which is well suited for model training and image dataset storage. We construct experiments to evaluate the effectiveness of the proposed compression ap-

proach for CIFAR, FashinMNSIT, and ImageNet, respectively. Extensive experiments have demonstrated the effectiveness of the proposed approach in dataset compression, especially obtaining better model performance (2.26% on average) than similar methods for the same compression ratio.

In the rest of the paper, we first briefly describe the MPS decomposition and the process of knowledge distillation in Section 3. Then we introduce our proposed MMPS approach for image dataset compression in Section 4. We report experimental results in Section 5, review the related work in in Section 2 and conclude the paper in Section 6.

2 Related Work

We review the related works in three aspects.

Data selection. The data selection technique of selecting the valid knowledge through an illuminating or a priori approach [2, 6, 35, 39], either by giving illuminating knowledge about the task or by finding representative samples. The data selection define representative criterion in the first (*e.g.,* compactness [6, 29], forgetfulness [39], diversity [2, 35]), then select representative samples from original dataset based on the criterion, finally use the selected small dataset to train the machine learning model for a downstream task. In contrast, our approach does not require the presence of a representative sample and is a more general approach.

Knowledge distillation. Knowledge distillation is a technique of transferring knowledge from a collection of models into a single model [3, 5, 17, 30]. While network distillation aims to distill the knowledge of multiple networks into a single model, dataset distillation models network parameters as a function of synthetic training data and learn their synthetic data by minimizing the training loss on the original training data and the synthetic training data [41]. We use the idea of knowledge distillation to complement the learning of task-relevant information under different tasks. In other words, our goal is to capture the portion of information in the dataset sample that is valid for training deep neural networks and to perform a "selection" of information in the dataset sample.

Tensor-based matrix representation. Tensor-based method of matrices is a technique that allows representing dataset samples in the tensor form such that quantum entanglement corresponds to classical correlations between different coarse-grained textures [22]. Another application is the compression of neural networks. Matrix product operators have been used to compress linear layers of deep neural networks [9, 11, 25, 38]. Then, MPOP was proposed to fine-tune the pre-trained language model (PLM) effecitively [24], and OPF proposed to narrow the performance gap between small and large PLM [12]. Moreover, MPOE combined the tensor-based matrix representation and Mixture-of-Experts (MoE) effectively enlarge the PLM scalability [10]. In contrast, we represent a dataset sample jointly with multiple low-rank tensors, each low-rank tensor describing the difference in information between the previous other tensors and the original graph (*i.e.,* residual information).

Our work is highly built on these studies, while we have a new perspective by designing the dataset compression algorithm which enables extracted low-rank information in the image. It is the first time that multiple MPS is applied to image dataset compression, and we make contributions for a novel approach to dataset compression.

3 Preliminary

In this paper, scalars are denoted by lowercase letters (*e.g.,* a), matrices are denoted by boldface capital letters (*e.g.,* $\mathbf{M}$), and high-order (order three or higher) tensors are denoted by boldface Euler script letters (*e.g.,* $\mathcal{T}$). A 3-order tensor $\mathcal{T}_{i_1,i_2,i_3}$ can be considered as a (potentially multi-dimensional) array with 3 indices $\{i_1, i_2, i_3\}$.

3.1 Matrix Product State

Originating from quantum many-body physics, matrix product states (MPS) is a standard algorithm to factorize a matrix into a sequential product of multiple local tensors (*i.e.*, a multi-way array) [4, 22, 28]. MPS decomposition is generally divided into two parts: coarse-grained process and low-rank truncation approximation.

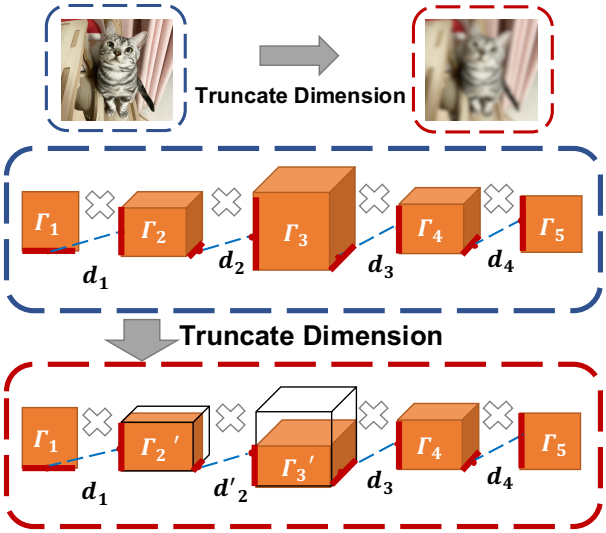

Fig. 1: MPS decomposition with five tensors. Dash line denotes *bond* of MPS tensors.

Coarse-grained process. Formally, given a matrix $\mathbf{S} \in \mathbb{R}^{I \times J}$, its MPS decomposition into a product of n local tensors can be represented as:

$$\text{MPS}\,(\mathbf{S}) = \prod_{k=1}^{n} \mathcal{T}_{(k)}[d_{k-1}, j_k, d_k], \quad d_k = \min\left(\prod_{m=1}^{k} j_m, \prod_{m=k+1}^{n} j_m\right), \tag{1}$$

where the $\mathcal{T}_{(k)}[d_{k-1}, j_k, d_k]$ is a 3-order tensor with size $d_{k-1} \times j_k \times d_k$ in which $\prod_{k=1}^{n} j_k = I \times J$ and $d_0 = d_n = 1$. We use the concept of *bond* to connect two adjacent tensors [8]. The bond dimension d_k is defined by:

$$d_k = \min\left(\prod_{m=1}^{k} i_m, \prod_{m=k+1}^{n} i_m\right), \tag{2}$$

we can see from Equation (2) that the d_k is large in the middle and small on both sides. We present a detailed algorithm for MPS decomposition in Algorithm 1. Figure 2 presents the illustration of MPS decomposition, and we use $n = 5$ in this paper.

Algorithm 1 MPS decomposition for a matrix.

Require: matrix $\mathbf{S}$, the number of local tensors n
Ensure: : MPS tensor list $\{\mathcal{T}_{(k)}\}_{k=1}^{n}$
for $k = 1 \rightarrow n-1$ **do**
  Perform coarse-grained process: $\mathbf{S}[I, J] \longrightarrow \mathbf{S}[d_{k-1} \times i_k, -1]$
  Perform SVD: $\mathbf{U}\lambda\mathbf{V}^{\top} = \mathrm{SVD}\ (\mathbf{S})$
  Reshape matrix to 3-order tensor: $\mathbf{U}[d_{k-1} \times i_k, d_k] \longrightarrow \mathcal{U}[d_{k-1}, i_k, d_k]$
  Save the decomposed tensor: $\mathcal{T}^{(k)} := \mathcal{U}$
  Merge λ and $\mathbf{V}^{\top}$: $\mathbf{S} := \lambda\mathbf{V}^{\top}$
end for
Save the decomposed tensor: $\mathcal{T}^{(n)} := \mathbf{S}$
Perform the normalization procedure.
return MPS tensor list $\{\mathcal{T}_{(k)}\}_{k=1}^{n}$

Low-rank truncation approximation. With the MPS decomposition describe in Equation (1), we can exactly decompose a matrix by MPS into the form of a series of products of local tensors and multiply these tensors together to completely reconstruct the original matrix $\mathbf{M}$. We can truncate the k-th bond dimension d_k (see Equation (1)) of local tensors to $d_k^{'}$ for low-rank approximation ($d_k > d_k^{'}$). Different values for $\{d_k\}_{k=1}^{n}$ can be set to control the low-rank information. Let $\{\lambda_j\}_{j=1}^{d_k}$ are the singular values of $M[j_1, \ldots, j_k, j_{k+1}, \ldots, j_n]$. We define the truncation error induced by the k-th bond dimension d_k local truncation error ϵ_k, which can be efficiently computed as $\epsilon_k = \sum_{j=d_k-d_k^{'}}^{d_k} \lambda_j$. After defining the local truncation error in Definition, we can derive the upper exact bound of the truncation error of the MPS decomposition by iteration. Let ϵ_k denoted the local truncation error of k-th bond dimension. The upper exact bound of the truncation error with MPS decomposition can be caclulated by:

$$||\mathbf{M} - \mathrm{MPS}(\mathbf{M})||_F \leq \sqrt{\sum_{k=1}^{n-1} \epsilon_k^2}. \tag{3}$$

Suppose that we have truncated the dimensions of local tensors from $\{d_k\}_{k=1}^n$ to $\{d_k^{'}\}_{k=1}^n$, the compression ratio can be computed by :

$$\rho = \frac{\sum_{k=1}^n d_{k-1}^{'} j_k d_k^{'}}{\prod_{k=1}^n j_k}. \tag{4}$$

The smaller the compression ratio, the fewer parameters are kept in MPS representation.

3.2 Knowledge Distillation

Knowledge distillation is a method of transferring knowledge from a collection of many individually trained networks into a single, typically compact network, performing a kind of model compression [16]. Specifically, dataset distillation is where we keep the model fixed, but encapsulate the knowledge of the entire training dataset (which typically contains thousands to millions of images) in a small number of synthetic training images [41].

Suppose we are given a large dataset consisting of $|\mathcal{S}|$ training images and its category label $\mathcal{S} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^{|\mathcal{S}|}$, where $\boldsymbol{x} \in \mathcal{X} \subset \{0, \ldots, C-1\}$, $\mathcal{X}$ is a d-dimensional input space and C is the number of class. We want to learn a deep neural network ϕ with parameter θ that correctly predicts the labels of previously unseen images, *i.e.*, $y = \phi_\theta(x)$. We can learn the parameters of this function by minimizing an empirical loss term on the training set, *i.e.*, $\theta^{\mathcal{S}} = \operatorname{argmin}_\theta \mathcal{L}^{\mathcal{S}}(\theta)$, where $\mathcal{L}^{\mathcal{S}}(\theta) = \frac{1}{\|\mathcal{S}\|} \sum_{(\boldsymbol{x},y)\in\mathcal{S}} l(\phi_\theta(\boldsymbol{x}, y))$, $l(\cdot,\cdot)$ is a task-specific loss and $\theta^{\mathcal{S}}$ is minimizer of $\mathcal{L}^{\mathcal{S}}$. The goal of dataset distillation is to generate a small set of condensed synthetic samples with their labels, $\mathcal{B} = (\boldsymbol{b}_i, y_i)_{i=1}^{\mathrm{B}}$ where $\boldsymbol{b} \in \mathcal{R}^d$ and $y \in \mathcal{Y}$. And this motivates us to distill task-specific knowledge from real datasets to supplement compressed datasets based on task-agnostic compression methods.

4 Approach

MPS decomposition, a tensor representation commonly used in quantum many-body physics, has been indicated for compression of image datasets [4]. However, this direct truncation of tensor for compression results in significant information loss. Inspired by the fact that a set of tensors orthogonal to each other in tensor decomposition is more expressive than a single tensor [43], we propose a method to approximate a picture using multiple MPS tensors, called *MMPS*. In particular, we propose two main improvements for MPS-based dataset compression, which can efficiently compress the image dataset and effectively complementary task-specific information.

4.1 Task-agnostic Dataset Compression

In this subsection, we aim to introduce our proposed MMPS approach. Suppose we are given a large dataset consisting of $|\mathcal{S}|$ training samples $\mathcal{S} = \{(\mathbf{S}_i)\}_{i=1}^{|\mathcal{S}|}$ where $\mathbf{S}_i \in \mathcal{S} \subset \mathbb{R}^d$, $\mathcal{S}$ is a d-dimensional input space. The $\mathbf{S}_i \in \mathcal{S}$ as the original image

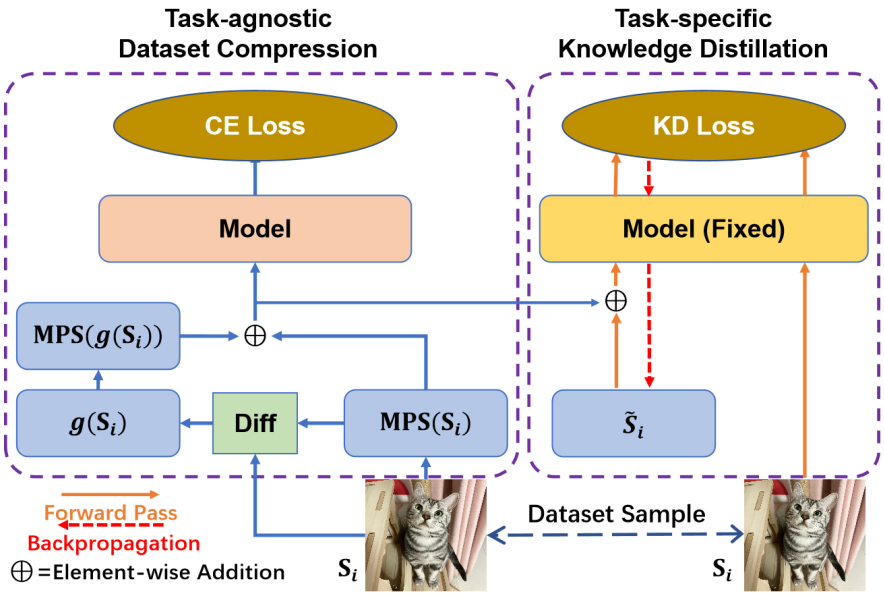

Fig. 2: Illustration of the proposed MMPS strategy. $\mathbf{S}_i$ denotes the original image dataset sample. $\mathrm{MPS}(\mathbf{S}_\mathrm{i})$ denotes the MPS decomposed tensor set. $g(\mathbf{S}_\mathrm{i})$ denotes the difference between $\mathbf{S}_i$ and $\mathrm{MPS}(\mathbf{S}_\mathrm{i})$. $\tilde{\mathbf{S}}_i$ denotes the trainable matrix for distillation. CE loss and KD loss denote the cross-entropy loss function and the knowledge distillation loss function, respectively.

dataset sample. We denote the $\mathrm{MPS}(\mathbf{S}_\mathrm{i})$ as the truncated tensor set with MPS decomposition on $\mathbf{S}_i$, which was proposed to use MPS for image datasets compression [4]. Compressing image datasets directly using MPS decomposition is effective on some simpler datasets (*e.g.* MNIST [23], COIL [27]). However, this approach brings a significant degradation of model performance when dealing with some complex datasets (*e.g.* CIFAR [21], ImageNet [31]). To address this problem, inspired by [43], we intend to introduce multiple MPS tensors to represent an image.

Here we discuss arbitrary image $\mathbf{S}_i$ with m-dimensional in the image dataset, and this can be eminently extended to the entire image dataset as well. First, we can use MPS decomposition to represent $\mathbf{S}_i$ as a series of products of local tensors of the form $\{\mathcal{T}_1, \mathcal{T}_2, \mathcal{T}_3, \mathcal{T}_4, \mathcal{T}_5\}^{(\mathbf{S}_i)}$, where $\{\mathcal{T}_i\}_{i=1}^5$ are third-order tensors. After obtaining this tensor group $\{\mathcal{T}_i\}_{i=1}^5$, we can truncate the k-th bond dimension between the tensors from $\{d_k\}_{k=1}^5$ to $\{d'_k\}_{k=1}^5$. Then we can use tensordot to reconstruct these tensors into a vector $\mathbf{S}'_i$ with the m-dimensions, which denote as MPS($\mathbf{S}_i$). We denote $g(\mathbf{S}_i) = \mathbf{S}_i - \mathrm{MPS}(\mathbf{S}_i)$ as the vector of the difference between the original image $\mathbf{S}_i$ and the reconstructed image $\mathbf{S}'_i$, which contains the information that is lost due to truncation approximation. Subsequently, we decompose $g(\mathbf{S}_i)$ by MPS and perform the same truncation operation as the previous. Finally, we will get two sets of tensors (*i.e.*, $\mathrm{MPS}(g(\mathbf{S}_i))$ and $\mathrm{MPS}(\mathbf{S}_\mathrm{i})$) for the image $\mathbf{S}_i \in \mathbb{R}^m$. The image obtained by MMPS

representation is as follows:

$$\mathrm{MMPS}(\mathbf{S}_i) = \mathrm{MPS}(\mathbf{S}_i) + \mathrm{MPS}(g(\mathbf{S}_i)). \tag{5}$$

Since the number of parameters contained in these tensor is smaller than the number of original image pixels, we can obtain the compressed image dataset. The compression ratio η of the obtained image dataset can be calculated as following:

$$\eta = \frac{2\sum_{k=1}^{5} d'_{k-1} j_k d'_k}{\prod_{k=1}^{5} j_k}, \tag{6}$$

From Equation (6) we can see that if d'_k is larger, then more parameters of ResMPS will be used and the compression rate ρ will become larger. Conversely, the smaller d_k is, the smaller the compression rate will be. We will store the parameters of these tensors in hardware instead of the pixels of the original image. Note that when training the neural network using the MPS dataset, we need to reconstruct the two MPS tensor sets as vectors separately. Then we do an element-wise add for these two vectors, which is the image sample used to train the network. Algorithm 2 presents a complete procedure for our approach.

Algorithm 2 Task-agnostic Dataset Compression Procedure.

Require: Image training dataset with N samples ($\mathbf{S}$).
Ensure: : Compressed training dataset.
1: **for** $i = 1 \rightarrow N$ **do**
2: Perform MPS decomposition: $\mathrm{MPS}(\mathbf{S}_\mathrm{i}) = \prod_{\mathrm{k}=1}^{5} \mathcal{T}_{(\mathrm{k})}[\mathrm{d}_{\mathrm{k}-1}, \mathrm{j}_\mathrm{k}, \mathrm{d}_\mathrm{k}]$
3: Compress MPS tensors by trucating $\{d_k\}_{k=1}^{5} \longrightarrow \{d'_k\}_{k=1}^{5}$
4: Computing residual information: $\mathrm{Res}(\mathbf{S}_i) = \mathbf{S}_i - \mathrm{MPS}(\mathbf{S}_i)$
5: Perform MPS decomposition: $\mathrm{MPS}(\mathrm{Res}(\mathbf{S}_\mathrm{i})) = \prod_{\mathrm{k}=1}^{5} \mathcal{R}_{(\mathrm{k})}[\mathrm{d}_{\mathrm{k}-1}^{(\mathrm{r})}, \mathrm{j}_\mathrm{k}, \mathrm{d}_\mathrm{k}^{(\mathrm{r})}]$
6: Compress MPS tensors by trucating $\{d_k^{(r)}\}_{k=1}^{5} \longrightarrow \{d_k^{'(r)}\}_{k=1}^{5}$
7: **end for**
8: **return** Compressed dataset$\{\mathrm{MPS}(\mathbf{S}_\mathrm{i}); \mathrm{MPS}(\mathrm{Res}(\mathbf{S}_\mathrm{i})\}_{\mathrm{i}=1}^{\mathrm{N}}$

We show the empirically results (Table 1) that our proposed approach can improve the model performance significantly than MPS [4, 22] on the same compression rate.

4.2 Task-specific Knowledge Distillation

Image datasets are often used to accomplish different computational tasks. In such cases, *task-specific* knowledge [26] is the aspects of the images that are relevant to the specific task. Therefore, we adopt knowledge distillation to complement ResMPS with task-specific information according to different task types. Specifically, we initialize trainable matrix $\tilde{\mathbf{S}}_i$ as implicit bias between $\mathrm{ResMPS}(\mathbf{S}_\mathrm{i})$ and real dataset sample $\mathbf{S}_i$. Then we introduce the knowledge distillation loss function to learn $\tilde{\mathbf{S}}_i$.

Knowledge distillation is used to distill the knowledge from a large training dataset into a small one [41]. They synthesize data matrix as training data to approximate models trained on the original data. Inspired by data distillation, we initialize trainable matrix $\tilde{\mathbf{S}}_i$ with zeros as implicit bias so that adding $\tilde{\mathbf{S}}_i$ would not hurt the model performance at the first step of training. Similarly, in the context of information supplementation, the synthetic dataset is calculated by $\mathbf{S_i}^* = \mathrm{ResMPS}(\mathbf{S_i}) + \tilde{\mathbf{S}}_\mathrm{i}$. Then the synthetic dataset $\mathbf{S_i}^*$ is trained to mimic the behaviors of the real dataset $\mathbf{S}_i$ with the model fixed. Formally, This training process can be modeled as minimizing the following objective function:

$$\mathcal{L}_{\mathrm{KD}} = \sum_{\mathbf{S}_i \in \mathcal{S}} \mathcal{L}(f(\mathbf{S_i}^*), f(\mathbf{S}_i)), \tag{7}$$

where $\mathcal{L}(\cdot)$ is a loss function that evaluates the difference between outputs of real and synthetic datasets. Without loss of generality, we use the Mean Squared Error (MSE) between the logit vectors as the $\mathcal{L}(\cdot)$ inspired by the existing experience [19]. Finally, our approach provides a more principle way of information supplementation. By updating implicit bias $\tilde{\mathbf{S}}_i$, the synthetic dataset sample $\mathbf{S_i}^*$ can better adapt to a specific task or network architecture and thus achieve better performance.

To this end, we use $\tilde{\mathbf{S}}_i$ to represent task-specific knowledge to supplement the task-agnostic compression method and boost the downstream performance. And the number of parameters in $\tilde{\mathbf{S}}_i$ can be further reduced by low-rank approximation to alleviate huge computation costs when transferring to other downstream tasks.

4.3 The Overall Procedure

In general, our approach can compress any given image dataset and make the model trained on the compressed dataset match the test accuracy of the model trained on the original dataset. Here, we choose CIFAR [4] and FashionMNIST [5] as representative image datasets and use our algorithm for these datasets.

The procedure can be simply summarized as follows. First, we perform the task-independent dataset compression process, which can get multiple MPS local tensors, and reconstruct each of the tensor sets as a vector with the same dimensionality as the original image sample. Then, we add the two vectors to obtain the image samples based on the ResMPS approximation. Next, we perform the task-specific knowledge distillation process. The distillation information matrix $\tilde{\mathbf{S}}_i$ specific to different tasks can be obtained. Finally, the matrix $\tilde{\mathbf{S}}_i$ obtained by distillation is summed with the image samples of ResMPS as the sample input to the model. Furthermore, we demonstrate in Section 5.1 through sufficient experiments that our proposed ResMPS approach outperforms existing tensor decomposition-based compression methods [4] for image datasets.

4.4 Discussion

The existing image dataset compression is divided into three approaches, *i.e.*, low-rank approximation of images [4], data selection [13], and dataset distillation [41] (also

[4] Available at https://www.cs.toronto.edu/~kriz/cifar.html

[5] Available at https://www.worldlink.com.cn/en/osdir/fashion-mnist.html

know as dataset condensation [44]). In particular, the low-rank approximation of images is task-agnostic compression, while core-set selection, as well as dataset distillation, are task-specific dataset compressions. However, the data selection typically relies on heuristics (*e.g.* picking cluster centers) that does not guarantee any optimal solution for the downstream task. The low-rank approximation based method performs well on small datasets but causes the model to perform much worse on large datasets. Our proposed MMPS can provide a more efficient representation of the low-rank information of the data under the same compression rate condition by introducing a set of low-rank approximation tensors. We also introduce a task-specific knowledge distillation procedure to complement the application weakness of compressed datasets under different tasks, using the idea of knowledge distillation to introduce task-specific information into the dataset compression. Conclusively, our proposed MMPS approach achieves a more efficient representation based on a low-rank approximation, while introducing task-specific information to improve the adaptability of the compressed dataset to different tasks.

In practice, we do not need to strictly follow the original image size. Instead, it is easy to pad additional zero entries to enlarge matrix rows or columns, so that we can obtain different MPS decomposition results. Another note is that the MPS-based approach can work with other compression methods: it can compress datasets condensed by previous methods even more.

5 Experiments

In this section, we first evaluate data selection [1], SVD-based method, MPS-based method [4] and our MMPS approach on CIFAR10, CIFAR100 and FashionMNIST datasets. Next, we investigate the proposed approach by performing ablation analysis and controlled experiments. Finally, we validated the effectiveness of our MMPS approach over the MPS approach on ResNet18, VGG, and MobileNETV2.

Datasets. We first evaluate classification performance with compressed images on three standard benchmark datasets: CIFAR10, CIFAR100 and FashionMNIST. In particular, the FashionMNIST dataset has 60,000 training and 10,000 testing images of 10 classes, while CIFAR10 and CIFAR100 both have 50,000 training and 10,000 testing images from 10 and 100 object categories, respectively. Then we use AutoMobile for ablation experiments since it consists of images from 1985 ward's automotive yearbook with two kinds of labels *w.r.t.* two multi-label classification tasks (*i.e.,* categories and colors classification). In all experiments, we use the standard train/test splits of the datasets and finally report the accuracy of the testing dataset.

Baselines. Our baseline methods include:

• MPS [4]: It first transforms images into MPS representation and truncates the dimensions only once for compression.

• Data Selection: It reduces the large dataset into a small equally informative portion of data, including Random and K-Center [42]. In Random, the training samples are randomly selected as the core set. K-Center picks multiple center points such that the largest distance between a data point and its nearest center is minimized.

Table 1: The performance comparison to data selection method and tensor-based methods. This table shows the testing accuracies (%) of different methods on three compression ratios,*i.e.* 30%, 50% and 70%. "↑" indicates average improvement for the dataset comparing MMPS with the best baseline method. ResNet18 is used for training and testing. Bold fonts indicate the best results in each block.

Dataset	CIFAR10 (↑ 0.91)			CIFAR100 (↑ 2.26)			FashionMNIST (↑ 0.45)		
	30%	50%	70%	30%	50%	70%	30%	50%	70%
Random	91.67	91.19	92.27	60.15	65.31	68.76	91.42	91.83	92.55
K-Center	77.42	78.56	79.93	47.63	52.31	54.27	83.36	85.85	86.02
SVD	87.92	89.88	90.47	57.30	59.65	62.14	91.26	91.75	91.98
CP	83.94	86.65	87.21	51.77	53.54	54.24	89.01	89.93	90.11
MPS	91.49	91.27	92.67	60.89	66.46	68.96	91.45	92.12	92.78
MMPS	**92.16**	**92.67**	**93.32**	**64.03**	**68.77**	**70.29**	**91.46**	**92.89**	**93.35**

• Tensor Based Methods: It compresses the dataset by applying low-rank approximation to each image, including SVD and CP decomposition.

Implementations. For main results in Table 1, we reproduce baseline methods on the datasets. As for the implementation of the compression rate, the core-set construction-based methods (*i.e.,* Random and K-Center) use a portion of the images, and the matrix decomposition-based methods (*i.e.,* SVD, CP, and MPS) achieve by a low-rank approximation. To ensure fairness of the comparison, we specify three levels of compression ratio, *i.e.,* 30%, 50%, and 70%. We conducted experiments on A100 with 40G memory. For all models, we chose the most appropriate learning rate among $\{0.1, 0.01, 0.001\}$ and selected the most appropriate checkpoint for testing based on the accuracy of the validation set.

Note that the task-specific knowledge distillation module is represented as a linear layer closest to the data so that it can be easily removed without affecting other components in the network architecture. Thus, we may directly integrate the task-specific knowledge into the MMPS as a whole for storage and migration. In other words, the final network architecture is unaffected, and we only get a compressed dataset. Compared with other methods, our approach significantly improves the accuracy of downstream tasks due to the task-specific information.

5.1 Experimental Results

Comparison to image compression. We first compare MMPS with the closest baseline, *i.e.,* MPS, in the three image classification benchmarks. As shown in Table 1, our approach outperforms MPS in all tasks. We note that MPS, benefiting from its effective retention of important information in images, achieves the best performance among all the baseline methods. Then MMPS can further boost the performance of MPS on downstream tasks, which verifies the superiority of our proposal. By zooming in on a specific dataset, the performance of all methods on CIFAR100 is relatively lower than that on other datasets. This can be attributed to the fact that CIFAR100 is more challenging, as

recognizing 10 times more categories with $\frac{1}{10}$ fewer images per class. But MMPS, on the contrary, obtains the most obvious improvement, especially with the lowest compression ratio (*i.e.,* 64.03% vs. 60.89% for MMPS and MPS under a compression ratio of 30%). The MMPS dataset seems to work better with few shot tasks, which enhances the data efficiency of the training dataset.

Comparison to data selection methods. To demonstrate the strength of image compression with MMPS over the data selection, we perform experiments on CIFAR10 and use Random and K-Center for comparison. Table 1 summarizes the results. Overall, our MMPS approach achieves competitive results over data selection methods, especially for the K-Center method. This is due to the fact that MMPS can effectively represent low-rank information in image datasets. To compare the quality of truncated MPS representation for CIFAR10, we visualize images from five categories with different dimensions $d_k^{'}$ in Figure 3. We observe that it is impossible to see the difference before and after compression if ρ is larger than 36%. Compared to losing some images by the data selection, reserving low-rank information by our MMPS approach can minimize the damage to the dataset.

Comparison to tensor based methods. As discussed in Section 4.4, we use other tensor-based methods (*i.e.,* SVD [15], CP [18]) for comparison to demonstrate the effectiveness of dataset compression. From the Tabel 1, we can observe that SVD and CP decomposition failed to preserve useful information for model performance especially when the compression ratio is less than 30% (at a maximum of 8.22 compared with MMPS and CP on CIFAR10). In particular, we note that MPS has a significant advantage over other low-rank approximation methods, and this verifies that MPS works better than others for the problem of reserving important information of images. And the result that MMPS reaches higher scores than MPS is a further indication of the superiority of our method.

Comparison to different models. In general, our approach can be applied to any kind of network architecture. We have evaluated its performance with ResNet18. In this section, we continue to test our approach using another two standard deep network architectures: VGG-16 [36] and MobileNetV2 [34]. These famous pre-trained models showed state-of-the-art accuracy for several challenging recognition tasks on ImageNet and competitions. Table 2 presents the comparison of the testing accuracy with three network architectures. We find that there exists an obvious variance of performance due to difference of architecture design and number of parameters in these models and ResNet18 achieves the best in CIFAR10 and FashionMNIST. Despite the differences, MMPS can cooperate with all kinds of network architectures and outperforms MPS.

Ablation results. Our approach has incorporated two major contributions: task-agnostic dataset compression and task-specific knowledge distillation. To further demonstrate the effects, we conduct ablation experiments on a multi-label classification task. Here we consider two variants for comparison based on a low-rank approximation baseline [4]: (1) "w/o TS,TA" uses MPS representation . This comparison is to examine

Table 2: Evaluations on different network architectures. This table shows the testing accuracy (%) of different methods at a compression ratio of 70%.

Datasets	ResNet18			VGG			MobileNetV2		
	SVD	MPS	MMPS	SVD	MPS	MMPS	SVD	MPS	MMPS
CIFAR10	86.73	89.49	**92.04**	85.99	88.07	**92.21**	85.16	85.91	**89.56**
CIFAR100	63.39	64.06	**70.29**	59.67	60.80	**67.71**	59.14	61.05	**66.39**
FashionMNIST	91.57	92.55	**93.35**	91.28	92.37	**93.21**	91.17	92.73	**93.93**

Datasets	Automobile		Avg.
	Cat.	Col.	
Origin	82.1	92.6	87.4
w/o TS,TA	80.5	94.0	86.1
w/o TS	83.7	94.2	87.6
w/o TA	83.8	94.6	87.9
MMPS	**85.8**	**95.8**	**90.8**

Table 3: Ablation results of ResMPS and task-specific knowledge distillation.

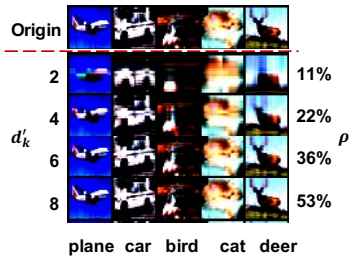

Fig. 3: Illustration of low-rank approximation for MPS to CIFAR10 images.

whether introducing residual information on image representation would lead to a performance improvement. (2) "w/o TA" remove task-specific knowledge distillation components. The goal of this variant is to demonstrate the improvements when considering the difference in task types. Table 3 shows the results when we ablate these. Comparing with "w/o TS,TA" and "w/o TS" in the table, we find that MMPS plays a key role in maintaining important information of images (losing too much information at once for MPS seriously damages the model performance, *i.e.,* 80.5 vs. 82.1 for MPS and Original images in "Cat."). Considering the characteristics of task types, we notice a significant variance in the performance of different tasks due to the different difficulties. We notice that MMPS can boost the performance in both task types based on MMPS (90.8 vs. 87.6 for average scores), which demonstrates the requirement for task-specific information.

5.2 Evaluation on More Tasks

As introduced in Section 4, our approach contains task-agnostic dataset compression and task-specific information supplementation. Due to task-agnostic compression, MPS representation can be applied in other computer vision tasks (*i.e., Pedestrian Detection*, *Visual Question Answering* and *Large-scale Image Classification*).

Pedestrian detection. First, we apply our approach to a pedestrian detection scenario where the goal is to accurately locate pedestrians in an image. We build our model on Mask R-CNN [14] method and fine-tune a pre-trained Mask R-CNN model in the Penn-Fudan Database [40] for Pedestrian Detection and Segmentation task. The dataset

contains 170 images with 345 instances of pedestrians and we decompose the original images with MMPS and use a trainable matrix to supplement information. The desired outcome is to obtain a high mean of average precision over all the classes. Finally, we report a COCO-style mAP score after 10 epochs of training, and the result is shown in Table 4. The result indicates that the MPS dataset can achieve competitive model performance while reducing 25% parameters of the dataset.

Visual question answering. The visual question answering task typically uses paired images and text to bridge vision and language respectively. Current approaches to this heavily rely on image feature extraction processes. Here we explore the use of our approach on VQAv2. The VQAv2 task asks for answers given pairs of an image and a question in natural language. Test-dev score is calculated by comparing the inferred answer to the 10 ground-truth answers. Our goal is to verify that short-range correlation in images can be used to model the cross-modal interaction between image-text pairs efficiently. To this end, we replace the image with an MPS representation in the image-text pairs. Following [20], we use a pre-trained ViLT model and fine-tune the model on the MPS dataset. Finally, from Tabel 4 we observe that our approach achieves comparable testing performance, and meanwhile significantly decreases the size of the dataset.

Large-scale image classification Pre-trained deep learning models (ResNet, VGG) learned on large-scale datasets have shown their effectiveness over conventional methods. Instead of training a model from scratch, one can fine-tune a pre-trained model to solve some specific task. To demonstrate the effectiveness of low-rank information on transfer learning, we apply our approach to the ImageNet dataset. To this end, we observe that the total dataset size is significantly reduced due to the compression of each image in the dataset. Furthermore, we evaluate the performance of the pre-training ResNet18 model on both the original ImageNet and the MPS compressed dataset. We observe that the MMPS dataset achieves comparable accuracy to the original ImageNet dataset. This result shows that the MMPS dataset with low-rank information can support large-scale pre-training.

Table 4: The performance comparison with the MPS method, both our proposed MMPS approach and the MPS method have a compression ratio of 75%.

Experiments	**Pedestrian Detection** (mAP)	**VQAv2** (test-dev score)	**ImageNet** (acc)
Origin	79.90	70.33	64.21
SVD	66.59	55.64	46.58
MPS	67.32	57.27	47.13
MMPS	**78.45**	**68.78**	**63.10**

6 Conclusion

In this paper, we propose a novel dataset compression approach based on multiple MPS decomposition and knowledge distillation. With MPS decomposition, it is able to efficiently reorganize and decouple low-rank information in local tensors. Since the low-rank information in images is important for training, we design a novel dataset compression approach that achieves effective compression of the dataset by performing multiple MPS decomposition for images in the task-agnostic scenario, while using distillation to complement task-relevant information. Extensive experiments have demonstrated the effectiveness of our MMPS approach, especially in that the compressed dataset using the MMPS can be directly applied to a variety of different neural network tasks. To the best of our knowledge, this is the first application of multiple MPS for dataset compression. In future work, we will consider exploring more decomposition structures for MPS.

Acknowledgments

This work was partially supported by National Natural Science Foundation of China under Grants No. 62206299 and 62222215, Beijing Outstanding Young Scientist Program under Grant No. BJJWZYJH012019100020098 and CCF-Zhipu AI Large Model Fund. Xin Zhao and Zhi-Yuan Xie are the corresponding authors.

References

1. Agarwal, P. K., Har-Peled, S., and Varadarajan, K. R. Approximating extent measures of points. *J. ACM 51*, 4 (2004), 606–635.
2. Aljundi, R., Lin, M., Goujaud, B., and Bengio, Y. Gradient based sample selection for online continual learning. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada* (2019), H. M. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. B. Fox, and R. Garnett, Eds., pp. 11816–11825.
3. Ba, J., and Caruana, R. Do deep nets really need to be deep? In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada* (2014), Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, Eds., pp. 2654–2662.
4. Bengua, J. A., Phien, H. N., and Tuan, H. D. Optimal feature extraction and classification of tensors via matrix product state decomposition. In *2015 IEEE International Congress on Big Data* (2015), IEEE, pp. 669–672.
5. Buciluǎ, C., Caruana, R., and Niculescu-Mizil, A. Model compression. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining* (2006), pp. 535–541.
6. Castro, F. M., Marín-Jiménez, M. J., Guil, N., Schmid, C., and Alahari, K. End-to-end incremental learning. In *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part XII* (2018), V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds., vol. 11216 of *Lecture Notes in Computer Science*, Springer, pp. 241–257.

7. DEVLIN, J., CHANG, M., LEE, K., AND TOUTANOVA, K. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)* (2019), J. Burstein, C. Doran, and T. Solorio, Eds., Association for Computational Linguistics, pp. 4171–4186.
8. FANNES, M., NACHTERGAELE, B., AND WERNER, R. F. Finitely correlated states on quantum spin chains. *Communications in mathematical physics 144*, 3 (1992), 443–490.
9. GAO, Z.-F., CHENG, S., HE, R.-Q., XIE, Z. Y., ZHAO, H.-H., LU, Z.-Y., AND XIANG, T. Compressing deep neural networks by matrix product operators. *Phys. Rev. Research 2* (Jun 2020), 023300.
10. GAO, Z.-F., LIU, P., ZHAO, W. X., LU, Z.-Y., AND WEN, J.-R. Parameter-efficient mixture-of-experts architecture for pre-trained language models. In *Proceedings of the 29th International Conference on Computational Linguistics* (2022), pp. 3263–3273.
11. GAO, Z.-F., SUN, X., GAO, L., LI, J., AND LU, Z.-Y. Compressing lstm networks by matrix product operators. *arXiv preprint arXiv:2012.11943* (2020).
12. GAO, Z.-F., ZHOU, K., LIU, P., ZHAO, W. X., AND WEN, J.-R. Small pre-trained language models can be fine-tuned as large models via over-parameterization. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (2023), pp. 3819–3834.
13. HAR-PELED, S., AND MAZUMDAR, S. On coresets for k-means and k-median clustering. In *Proceedings of the thirty-sixth annual ACM symposium on Theory of computing* (2004), pp. 291–300.
14. HE, K., GKIOXARI, G., DOLLÁR, P., AND GIRSHICK, R. B. Mask R-CNN. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017* (2017), IEEE Computer Society, pp. 2980–2988.
15. HENRY, E., AND HOFRICHTER, J. [8] singular value decomposition: Application to analysis of experimental data. *Methods in enzymology 210* (1992), 129–192.
16. HINTON, G. E., OSINDERO, S., AND TEH, Y. W. A fast learning algorithm for deep belief nets. *Neural Computation 18* (2006), 1527–1554.
17. HINTON, G. E., VINYALS, O., AND DEAN, J. Distilling the knowledge in a neural network. *CoRR abs/1503.02531* (2015).
18. HITCHCOCK, F. L. The expression of a tensor or a polyadic as a sum of products. *Journal of Mathematics and Physics 6*, 1-4 (1927), 164–189.
19. KIM, T., OH, J., KIM, N., CHO, S., AND YUN, S. Comparing kullback-leibler divergence and mean squared error loss in knowledge distillation. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021, Virtual Event / Montreal, Canada, 19-27 August 2021* (2021), Z. Zhou, Ed., ijcai.org, pp. 2628–2635.
20. KIM, W., SON, B., AND KIM, I. Vilt: Vision-and-language transformer without convolution or region supervision. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event* (2021), M. Meila and T. Zhang, Eds., vol. 139 of *Proceedings of Machine Learning Research*, PMLR, pp. 5583–5594.
21. KRIZHEVSKY, A., HINTON, G., ET AL. Learning multiple layers of features from tiny images.
22. LATORRE, J. I. Image compression and entanglement. *CoRR abs/quant-ph/0510031* (2005).
23. LECUN, Y. Mnist handwritten digit database, yann lecun, corinna cortes and chris burges. *URL: http://yann. lecun. com/exdb/mnist [Online* (2013).
24. LIU, P., GAO, Z., ZHAO, W. X., XIE, Z., LU, Z., AND WEN, J. Enabling lightweight fine-tuning for pre-trained language model compression based on matrix product operators. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics*

and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021 (2021), C. Zong, F. Xia, W. Li, and R. Navigli, Eds., Association for Computational Linguistics, pp. 5388–5398.

25. LIU, P., GAO, Z.-F., ZHAO, W. X., AND WEN, J.-R. Scaling pre-trained language models to deeper via parameter-efficient architecture. *arXiv preprint arXiv:2303.16753* (2023).
26. NEIFELD, M. A., ASHOK, A., AND BAHETI, P. K. Task-specific information for imaging system analysis. *J. Opt. Soc. Am. A 24*, 12 (Dec 2007), B25–B41.
27. NENE, S. A., NAYAR, S. K., MURASE, H., ET AL. Columbia object image library (coil-100).
28. PEREZ-GARCIA, D., VERSTRAETE, F., WOLF, M., AND CIRAC, J. Matrix product state representations. *QUANTUM INFORMATION & COMPUTATION 7*, 5-6 (2007), 401–430.
29. REBUFFI, S., KOLESNIKOV, A., SPERL, G., AND LAMPERT, C. H. icarl: Incremental classifier and representation learning. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017* (2017), IEEE Computer Society, pp. 5533–5542.
30. ROMERO, A., BALLAS, N., KAHOU, S. E., CHASSANG, A., GATTA, C., AND BENGIO, Y. Fitnets: Hints for thin deep nets. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings* (2015), Y. Bengio and Y. LeCun, Eds.
31. RUSSAKOVSKY, O., DENG, J., SU, H., KRAUSE, J., SATHEESH, S., MA, S., HUANG, Z., KARPATHY, A., KHOSLA, A., BERNSTEIN, M., BERG, A. C., AND FEI-FEI, L. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV) 115*, 3 (2015), 211–252.
32. RUSSAKOVSKY, O., DENG, J., SU, H., KRAUSE, J., SATHEESH, S., MA, S., HUANG, Z., KARPATHY, A., KHOSLA, A., BERNSTEIN, M., ET AL. Imagenet large scale visual recognition challenge. *International journal of computer vision 115*, 3 (2015), 211–252.
33. SAINATHTN, M., ET AL. Deep convolutionalneuralnetworksforlvcsr. *IEEE International-Conferenceon Acoustics, Speechand SignalProcessing 8614* (2013), 8618.
34. SANDLER, M., HOWARD, A. G., ZHU, M., ZHMOGINOV, A., AND CHEN, L. Mobilenetv2: Inverted residuals and linear bottlenecks. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018* (2018), Computer Vision Foundation / IEEE Computer Society, pp. 4510–4520.
35. SENER, O., AND SAVARESE, S. Active learning for convolutional neural networks: A core-set approach. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings* (2018), OpenReview.net.
36. SIMONYAN, K., AND ZISSERMAN, A. Very deep convolutional networks for large-scale image recognition. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings* (2015), Y. Bengio and Y. LeCun, Eds.
37. SUN, X., GAO, Z.-F., LU, Z.-Y., LI, J., AND YAN, Y. A model compression method with matrix product operators for speech enhancement. *IEEE/ACM Transactions on Audio, Speech, and Language Processing 28* (2020), 2837–2847.
38. SUN, X., GAO, Z.-F., LU, Z.-Y., LI, J., AND YAN, Y. A model compression method with matrix product operators for speech enhancement. *IEEE/ACM Transactions on Audio, Speech, and Language Processing 28* (2020), 2837–2847.
39. TONEVA, M., SORDONI, A., DES COMBES, R. T., TRISCHLER, A., BENGIO, Y., AND GORDON, G. J. An empirical study of example forgetting during deep neural network learning. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019* (2019), OpenReview.net.

40. WANG, L., SHI, J., SONG, G., AND SHEN, I. Object detection combining recognition and segmentation. In *Computer Vision - ACCV 2007, 8th Asian Conference on Computer Vision, Tokyo, Japan, November 18-22, 2007, Proceedings, Part I* (2007), Y. Yagi, S. B. Kang, I. Kweon, and H. Zha, Eds., vol. 4843 of *Lecture Notes in Computer Science*, Springer, pp. 189–199.
41. WANG, T., ZHU, J., TORRALBA, A., AND EFROS, A. A. Dataset distillation. *CoRR abs/1811.10959* (2018).
42. WOLF, G. W. Facility location: concepts, models, algorithms and case studies. series: Contributions to management science. *Int. J. Geogr. Inf. Sci. 25*, 2 (2011), 331–333.
43. XIE, H., HUANG, R., HAN, X., YAN, X., ZHAO, H., XIE, Z., LIAO, H., AND XIANG, T. Reorthonormalization of chebyshev matrix product states for dynamical correlation functions. *Physical Review B 97*, 7 (2018), 075111.
44. ZHAO, B., MOPURI, K. R., AND BILEN, H. Dataset condensation with gradient matching. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021* (2021), OpenReview.net.