applied sciences

MDPI

Article

Adversarial Optimization-Based Knowledge Transfer of Layer-Wise Dense Flow for Image Classification

Doyeob Yeo [1], Min-Suk Kim [2,*] and Ji-Hoon Bae [3,*]

[1] KSB Convergence Research Department, Electronics and Telecommunications Research Institute, Daejeon 34129, Korea; yeody@etri.re.kr
[2] Department of Human Intelligence and Robot Engineering, Sangmyung University, Cheonan 03016, Korea
[3] Department of AI and Big Data Engineering, Daegu Catholic University, Gyeongsan-si 38430, Korea
* Correspondence: minsuk.kim@smu.ac.kr (M.-S.K.); jihbae@cu.ac.kr (J.-H.B.)

Abstract: A deep-learning technology for knowledge transfer is necessary to advance and optimize efficient knowledge distillation. Here, we aim to develop a new adversarial optimization-based knowledge transfer method involved with a layer-wise dense flow that is distilled from a pre-trained deep neural network (DNN). Knowledge distillation transferred to another target DNN based on adversarial loss functions has multiple flow-based knowledge items that are densely extracted by overlapping them from a pre-trained DNN to enhance the existing knowledge. We propose a semi-supervised learning-based knowledge transfer with multiple items of dense flow-based knowledge extracted from the pre-trained DNN. The proposed loss function would comprise a supervised cross-entropy loss for a typical classification, an adversarial training loss for the target DNN and discriminators, and Euclidean distance-based loss in terms of dense flow. For both pre-trained and target DNNs considered in this study, we adopt a residual network (ResNet) architecture. We propose methods of (1) the adversarial-based knowledge optimization, (2) the extended and flow-based knowledge transfer scheme, and (3) the combined layer-wise dense flow in an adversarial network. The results show that it provides higher accuracy performance in the improved target ResNet compared to the prior knowledge transfer methods.

Keywords: adversarial optimization; layer-wise dense flow; knowledge transfer; image classification

Citation: Yeo, D.; Kim, M.-S.; Bae, J.-H. Adversarial Optimization-Based Knowledge Transfer of Layer-Wise Dense Flow for Image Classification. *Appl. Sci.* **2021**, *11*, 3720. https://doi.org/10.3390/app11083720

Academic Editor: Myo-Taeg Lim

Received: 21 March 2021
Accepted: 19 April 2021
Published: 20 April 2021

1. Introduction

In the past few years, as deep-learning technology has advanced dramatically, state-of-the-art deep neural network (DNN) models find applications in several fields, ranging from computer vision to natural language processing [1–10]. Modern DNNs are based on the convolutional neural network (CNN) structure [11], such as AlexNet [12], GoogleNet [13], VGGNet [14], the residual network (ResNet) [15,16], a densely connected convolutional network (DenseNet) [17], and EfficientNet [18], that has achieved increased accuracy by expanding more layers. Therefore, generally, top-performing DNNs have deep and wide neural network architectures with enormous parameters, significantly increasing the training time at high computational costs. Moreover, it is challenging to achieve global or local optimization for a complex DNN with an extended dataset, such as ImageNet data [19], used for training from scratch. Transfer learning [20] can be a reasonable candidate to address this limitation. This is because it leverages the knowledge gained from solving a task when applied to other similar tasks. When the wide and deep DNNs are successfully trained, they usually contain a wealth of knowledge within the learning parameters. Therefore, well-known transfer learning based on CNN structures [21,22] directly reuses most pre-trained convolutional layers that automatically learn hierarchical feature representations for knowledge formation. Notably, CNN-based transfer learning allows us to quickly and easily build some of the accurate network models by taking advantage of the previous learning without beginning from scratch. In addition, although transfer

learning provides a generalized network model with high performance, the training data are insufficient for a new domain. However, notably, the DNN trained from the existing CNN-based transfer learning becomes a complex neural network structure with numerous parameters because it reuses the entire pre-trained convolution-based structure. This complex structure can lead to high memory demands and increased inference time, which is not well suited for applications with limited computing resources, such as the "Internet of Things" (IoT) environment [23–25]. Therefore, more efficient knowledge distillation (KD) and knowledge transfer techniques in transfer learning are essentially extended to the application of DNNs for improved accuracy, fast inference time, including training time, and lightweight network structures suitable for restricted computing environments.

To achieve these requirements, in this study, we consider a knowledge transfer framework (KTF)-based training approach using a pre-trained DNN (as the source network model) and target DNN need to be trained using pre-trained knowledge. The concept of the KTF was first introduced in [26] by minimizing the Kullback–Leibler (KL) divergence of the output distribution between the pre-trained and target network models. Based on [26], Hinton et al. [27] proposed KD terminology from the KTF by considering the relaxed output distribution of the pre-trained DNN as distilled knowledge. According to [27], softening the output layer's neural response in the pre-trained DNN provided more information to the target DNN during training. Therefore, the main feature of KD can define the best knowledge that represents the useful information of the pre-trained DNN. Research on KD has been further extended by introducing an intermediate representation of the pre-trained DNN [28] and the flow of the solution procedure (FSP) [29] for knowledge expression. While Ref. [28] used the intermediate hidden information in the middle layer of the pre-trained DNN, the flow across two different layers in [29] was devised to represent the direction between the features of the two layers. Compared with [28], FSP-based distilled knowledge [29] provided better performance in classification accuracy over several benchmark datasets. Notably, the aforementioned methods are adopted by the Euclidean distance-based similarity measure to calculate the cost function of transferring distilled knowledge. There have been other studies in terms of knowledge transfer [30,31], attempting to transfer knowledge extracted by the pre-trained DNN to other target DNNs using a network structure similar to the generative adversarial network (GAN) [32,33]. In their studies, a target DNN was modeled as a generator, and a discriminator tried to distinguish the output results created by the pre-trained DNN from the result provided by the target DNN. The aforementioned knowledge transfer methods adopting the GAN structure experimentally showed that the target DNN using the adversarial optimization-based approach would be better captured into the pre-trained knowledge distributions than the l^2-norm-based knowledge transfer methods, as mentioned above. Therefore, it is crucial to extract useful knowledge from the pre-trained DNN and efficiently transfer the extracted knowledge to the target DNN.

In this study, we propose a layer-wise dense flow (LDF)-based knowledge transfer technique coupled with an adversarial network to generate low complexity DNN models with high accuracy performance that can be adaptively applied to target domains with limited computing resources. First, the proposed method introduces densely overlapped flow using FSP matrices as distilled knowledge of the pre-trained DNN. To rephrase, the multiple-overlapped flow-based knowledge is densely distilled, such that each piece of flow-based knowledge extracted between two different corresponding layers is superimposed. Second, KTF using the adversarial network transfers densely extracted knowledge with layer-wise concurrent training between the pre-trained and target DNNs. In this stage, we designed multiple independent discriminators for adversarial optimization-based knowledge transfer using multiple pairs of dense flows, where each discriminator is assigned to compare a pair of flow-based features between the pre-trained and target DNNs. Owing to the proposed LDF and its concurrent transfer for the adversarial optimization process, the target DNN can efficiently and accurately learn a plethora of information from the pre-trained DNN.

In contrast to previous work on the dense flow-based knowledge transfer [34], there are major differences between [34] and the proposed method. First, flow-based knowledge in [34] is transferred based on l^2-norm-based training when the dense flow extracted from layers is transferred and trained to another target model. However, knowledge in our study is transferred in the GAN structure-based adversarial optimization manner. Second, the densely extracted flow-based knowledge in [34] is sequentially transferred step-by-step to a target model, but this study deals with layer-wise concurrent training when transferring the dense flow.

2. Related Works

2.1. Generative Adversarial Networks

The GAN, comprising generator and discriminator, was proposed in [32] to capture the given data distributions. The role of the generator is to generate newly synthesized images or fake data, and that of the discriminator is to determine whether an input sample is the given real data or fake data from the generator. These two are designed to compete in an adversarial optimization manner such that the generator captures the distributions of the given data, and the discriminator makes the right decision as to whether the sample is real or fake. Let G and D be the sets of weights of the generator and discriminator, respectively. Then, the min-max optimization problem to train G and D can be defined as follows:

$$\min_{G} \max_{D} \mathcal{L}_{\mathrm{GAN}}(G, D) \tag{1}$$

In (1), $\mathcal{L}_{\mathrm{GAN}}$ is defined as

$$\mathcal{L}_{\mathrm{GAN}}(G, D) = \mathop{\mathbb{E}}_{x \sim P_{\mathrm{data}}(x)} [\log D(x)] + \mathop{\mathbb{E}}_{z \sim P_{\mathrm{z}}(z)} [1 - \log D(G(z))], \tag{2}$$

where P_{data} and P_z denote the distributions of the real data and input noise of the generator, respectively. To derive a well-optimized solution using the GAN, (1) is usually solved with min-max optimization iteratively. However, it is challenging to find the optimal solution for (1) because of some of the undesired saddle point problems. Therefore, several existing studies have been conducted to stabilize the convergence of the GAN algorithm, such as unrolled GANs [35], Wasserstein GANs [36], and least-squares GANs [37]. In addition, to improve the convergence and robustness for learning optimization of (1), heuristic techniques such as feature matching, minibatch discrimination, and one-sided label smoothing were introduced in [38]. Another type of prior advanced GAN architecture, robust deep convolutional generative adversarial networks (DCGANs) [39], have been successfully applied to image processing tasks such as object removal and vector arithmetic [39], super-resolution [40,41], and denoising [42,43].

2.2. Output-Distribution-Based Knowledge Transfer Using Adversarial Networks

Based on the general GAN-based architecture, an adversarial network-based knowledge transfer approach adopted by the KTF was first introduced in [30]. The relaxed output probability of neural networks described in [27] was used as the pre-trained knowledge in their method. GAN-based previous work in [30] considered couple of different optimization procedures for knowledge transfer method such as solving a discriminator's maximization problem and solving a minimization problem for a target DNN model. To update the discriminator and target DNN for adversarial training in the KTF. First, the discriminator's goal is required to distinguish whether a relaxed output distribution is provided by a pre-trained DNN or target DNN. In contrast, the target DNN, which plays the same role as the generator of the original GAN-based structure [32], is adversarially trained, similar to the relaxed output probability of the pre-trained DNN. Therefore, the knowledge transmitted from the discriminator leads the target DNN to provide the probability result of the output layer, similar to the pre-trained DNN. In addition, according to [27], the cross-entropy loss for the supervised training approach is considered in the

KTF. Experimentally, their method proved that the adversarial network-based approach could effectively transfer output-distribution-based knowledge from the pre-trained DNN to the target DNN, compared to the l^2-norm-based knowledge transfer [27] without using adversarial training loss. However, the optimized target DNN using [30] is usually inferior to the pre-trained original DNN, especially when considering wider and deeper neural networks as a source DNN.

2.3. FSP-Based Knowledge Transfer Using Adversarial Network

An adversarial network-based KTF technique using FSP-based knowledge distillation was proposed in [31] to improve the performance of the traditional adversarial network-based KTF using relaxed output distribution-based knowledge distillation [30]. For distilled knowledge, three flow-based items of source knowledge in the pre-trained DNN are extracted in the form of FSP matrices based on the input and output of the residual block in the ResNet structure. In [29], the FSP matrix was mathematically devised for flow-based knowledge distillation across two layers. Let $F(x;W) \in \mathbb{R}^{h\times w\times m}$ and $H(x;W) \in \mathbb{R}^{h\times w\times n}$ be two different feature maps with an input x and weights W in a ResNet. Then, the FSP matrix $G^W(x;W) = (g_{ij}^W) \in \mathbb{R}^{m\times n}$ between F and H is defined by

$$g_{ij}^W = \frac{1}{h\times w}\langle F_{:,:,i}, H_{:,:,j}\rangle_F \tag{3}$$

where $F_{:,:,i}$ and $H_{:,:,i}$ denote i-th $h \times w$ matrices of F and H, respectively, and $\langle\cdot,\cdot\rangle_F$ denotes the Frobenius inner product between two matrices of the same size.

Next, for knowledge transfer, multiple flow-based source knowledge is transferred using the adversarial optimization procedure between the target ResNet and the three discriminators. The target ResNet was trained to build its FSP matrices to deceive the discriminators. Simultaneously, the discriminators were trained to distinguish FSP matrices created by the pre-trained ResNet from those created by the target ResNet. Therefore, the adversarial optimization-based KTF approach [31] was implemented in semi-supervised learning such that the target ResNet can (i) capture the distribution of the FSP-based source knowledge and (ii) simultaneously use a known dataset with true labels. According to the results of [31], the classification accuracy of the target ResNet trained using [31] is better than the existing adversarial optimization-based knowledge transfer method [30] because of the FSP-based rich source knowledge. In addition, a target ResNet using [31] can accurately capture better knowledge from the original pre-trained knowledge distribution than the l^2-norm-based knowledge transfer method using FSP-based distilled knowledge [29].

3. Proposed Method

The proposed method is an adversarial training scheme using densely distilled flow-based knowledge based on the pre-trained DNN approach, which can efficiently optimize the KTF network for image classification tasks. The pre-trained information for dense flow is fully generated by converting the detailed features from the lower layers into abstracted features in the higher layers. This process requires efficient transmission of dense flow-based information to the target DNN module through multiple discriminators to optimize the proposed distilled-knowledge transfer. In this section, we present the proposed methods involving the main concepts to improve the classification performance over the prior KTF methods in terms of the knowledge transfer scheme.

3.1. Adversarial-Based Knowledge Optimization

Figure 1 shows the adversarial-based KTF architecture, where flow-based knowledge is extracted from a pre-trained network. This knowledge can be transferred to target and discriminator networks for updating them in an adversarial-optimization manner. The flow-based knowledge considered in this study is represented as an FSP matrix [29] based on the direction between the input and output results of the residual module of a pre-trained ResNet. Specifically, the adversarial-based KTF, as shown in Figure 1, describes

how the target network can be trained and built to deceive the discriminator network in its flow-based result. In addition, the discriminator network is required to optimize and distinguish the flow-based knowledge created by the pre-trained network from the flow-based result of the target network.

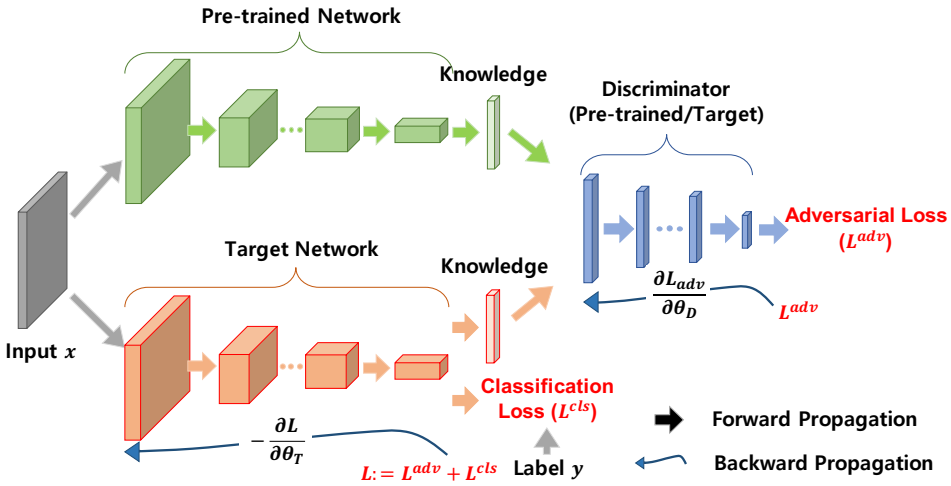


Figure 1. Adversarial-based knowledge transfer framework architecture.

As shown in Figure 1, mathematical notations for the adversarial-based knowledge optimization can be represented and derived as follows: Let T, R, and D be the weights of each target, pre-trained, and discriminator networks, respectively. Let G^T and G^R denote the flow-based FSP matrices of the target and pre-trained networks, respectively. $D(\cdot)$ represents the output of probability between zero and one by the discriminator, and the input of $D(\cdot)$ is assumed to be the FSP matrix of the target or pre-trained networks. When the value of $D(\cdot)$ is one, it implies that the input is an FSP matrix created by the pre-trained network, and, conversely, zero implies that the target network generates an FSP matrix rather than a pre-trained network. Then, the loss function of the adversarial-based knowledge optimization for positive parameters α, β, and γ is given below:

$$\mathcal{L}(T, D) = \alpha\Big\{\mathcal{L}^{\text{adv}}(T, D) + \gamma\mathcal{L}^{\text{FSP}}(T)\Big\} + \beta\mathcal{L}^{\text{cls}}(T), \tag{4}$$

where

$$\mathcal{L}^{\text{adv}}(T, D) = \mathop{\mathbb{E}}_{x\sim P_{\text{data}}(x)}\Big[\log D\Big(G^R(x; R)\Big)\Big] + \mathop{\mathbb{E}}_{x\sim P_{\text{data}}(x)}\Big[\log\Big(1 - D\Big(G^T(x; T)\Big)\Big)\Big], \tag{5}$$

$$\mathcal{L}^{\text{cls}}(T) = -\mathop{\mathbb{E}}_{y\sim P_{\text{data}}(y|x)}\Big[\log P^T(y \mid x)\Big], \tag{6}$$

and

$$\mathcal{L}^{\text{FSP}}(T) = \mathop{\mathbb{E}}_{x\sim P_{\text{data}}(x)}\Big[\|G^R(x; R) - G^T(x; T)\|_F^2\Big]. \tag{7}$$

Here, $\|\cdot\|_F$ denotes the Frobenius norm of the matrix. Then, an adversarial-based optimization problem for knowledge transfer is given as

$$\min_T \max_D \mathcal{L}(T, D). \tag{8}$$

To address the optimization problem of (8) with respect to the mini-batch $\mathcal{B} = \{(x_i, y_i)\}_{i=1}^N$ of size N, we reconstruct (8) as (9):

$$\min_T \max_D \mathcal{L}(\mathcal{B}; T, D). \tag{9}$$

According to (9), (5) and (6) can be represented as (10) and (11) below, respectively:

$$\mathcal{L}^{\text{adv}}(\mathcal{B};T,D)=\frac{1}{N}\sum_{i=1}^{N}\log D\Big(G^{R}(x_i;R)\Big)+\frac{1}{N}\sum_{i=1}^{N}\log\Big(1-D\Big(G^{T}(x_i;T)\Big)\Big). \tag{10}$$

and

$$\mathcal{L}^{\text{cls}}(\mathcal{B};T)=-\frac{1}{N}\sum_{i=1}^{N}\log P^{T}(y_i\mid x_i)=-\frac{1}{N}\sum_{i=1}^{N}y_i\cdot\log T(x_i), \tag{11}$$

where $T(\cdot)$ denotes the output probability of the target network. For a mini-batch size of N, (7) can be rewritten as

$$\mathcal{L}^{\text{FSP}}(\mathcal{B};T)=\frac{1}{N}\sum_{i=1}^{N}\|G^{R}(x_i;R)-G^{T}(x_i;T)\|_{F}^{2}. \tag{12}$$

Regarding (12), we consider adopting the l^2-norm-based loss term representing a blurring effect [31] of constructing the adversarial optimization-based loss function of (4). It is anticipated to provide more information about pre-trained source knowledge to the target network than flow-based knowledge without softening.

Based on the aforementioned loss functions, we can fully and simultaneously train both numerous discriminators and the target DNN of the adversarial network-based KTF to optimize the densely distilled knowledge transfer.

3.2. Knowledge Transfer Scheme for Densely Distilled Flow-Based Knowledge

In our previous work [34], we introduced a dense flow-based knowledge transfer learning scheme with a deep neural network, where flow-based knowledge was densely overlapped and extracted from a pre-trained ResNet. Compared to the original flow-based knowledge [29], the target ResNet obtained using dense flow-based training yielded higher performance owing to the rich information of the extended flow-based features for dense learning. Notably, when training a target ResNet in [34], the densely extracted knowledge was sequentially delivered step-by-step to the target ResNet. In this regard, the knowledge transfer of dense flow is performed using the l^2-norm-based loss function between a pre-trained and target ResNets. According to [34], concurrent flow-based training yielded inferior accuracy to the bottom-up sequential training scheme when transferring the densely extracted pre-trained knowledge in a KTF. Therefore, there is a limitation in simultaneously transmitting several densely extracted information using the traditional l^2-distance-based similarity measure.

Applying an adversarial network-based architecture that uses discriminator networks rather than the l^2-norm-based training approach to knowledge transfer of dense flow can be an effective solution to address this limitation. This explains why a typical target network using the adversarial-based training method can more accurately capture the distribution of pre-trained knowledge than the transfer learning method using the l^2-norm that usually produces blurriness in image restoration. Therefore, even considering densely extracted knowledge items, the adversarial training method can handle concurrent transference of the densely distilled knowledge, whereas the traditional l^2-norm-based method cannot.

Figure 2 shows the proposed adversarial network-based structure for concurrent knowledge transfer of the densely distilled flow-based knowledge when considering the popular ResNet model with three residual blocks in a KTF. To rephrase, six FSP matrices $G_{i,j}^{R}$ ($i=0,1,2$ and $j=1,2,3$) from the pre-trained ResNet are extracted from a dense overlap, and, similarly, the same number of FSP matrices $G_{i,j}^{T}$ from the target ResNet are generated. Then, the target ResNet is trained using the same number of discriminators $D_{i,j}$ ($i=0,1,2$ and $j=1,2,3$) such that the target ResNet's flow-based features are formed as close as possible to the actual features of the pre-trained ResNet by deceiving the discriminators. Meanwhile, the discriminators are trained to distinguish FSP matrices extracted by the pre-trained ResNet from those generated by the target ResNet. Notably, a single discriminator

is assigned to compare a pair of FSP matrices between the pre-trained and target ResNets. In discriminator's architecture, a multi-layer perceptron (MLP)-based discriminator with M linear units [31] rather than the popular CNN-based discriminator [39] was adopted in this study, considering computational bottleneck for the LDF-based knowledge transfer scheme. Here, a single linear unit comprises a fully connected layer, a batch normalization layer, and a leak rectified linear unit [31]. Thus, the flow-based ResNet layers are densely trained, as more enhanced information can be transmitted to the target ResNet fully and simultaneously using the overlapping flow-based features and densely designed discriminators.

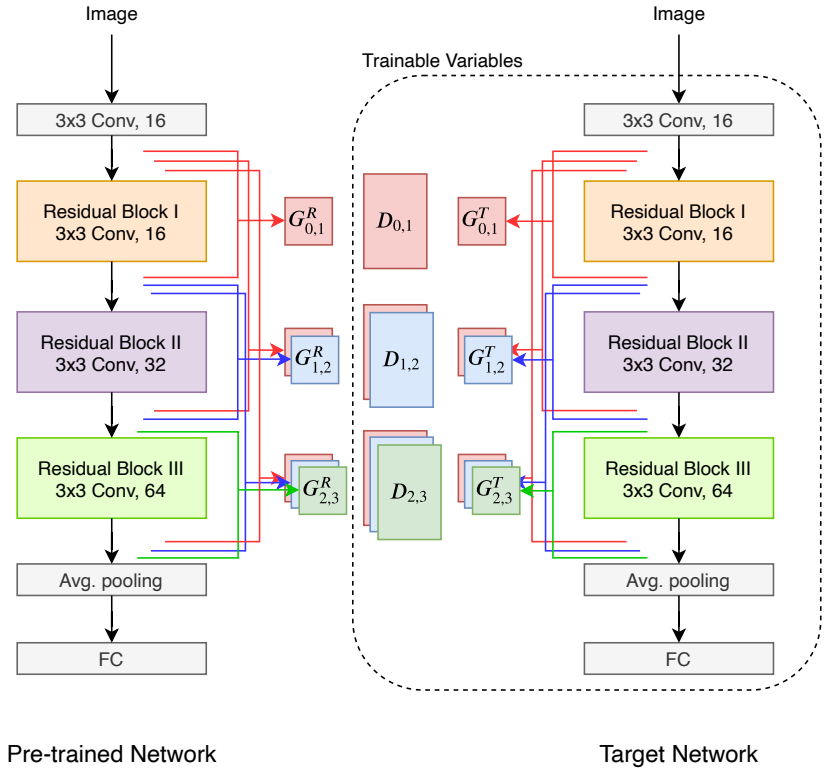


Figure 2. Adversarial concurrent knowledge transfer structure using the layer-wise dense flow. In this figure, $G^R_{i,j}$, $G^T_{i,j}$, and $D_{i,j}$ refer to the FSP matrix of pre-trained ResNet, the FSP matrix of target ResNet, and the discriminator, respectively. Only the variables in the dotted box marked "Trainable Variables" are used for training.

3.3. Adversarial-Based Loss Functions for Knowledge Transfer Using Layer-Wise Dense Flow

In Section 3.1, we present the loss functions of the adversarial-based optimization for distilled-knowledge-based transfer. Furthermore, dense flow-based feature extraction can enhance the original flow-based knowledge distillation, as described in Section 3.2. Therefore, by applying LDF-based knowledge transfer to the adversarial-based knowledge optimization, the proposed loss functions for the dense flow can be derived as follows: First, the adversarial loss function consists of M residual blocks of the pre-trained and target ResNets. Let $G^T_{i,j}$ and $G^R_{i,j}$ be the FSP matrix between the input feature of the $(i+1)$-th residual block and the output feature of the j-th residual block in the target and pre-trained ResNets, respectively, and let $D_{i,j}$ be a discriminator for $G^T_{i,j}$ and $G^R_{i,j}$, where $i = 0, 1, \cdots, M-1$ and $j = 1, 2, \cdots, M$ such that $i < j$. Then, we define $\mathcal{L}^{\text{adv}}_{i,j}$ and $\mathcal{L}^{\text{FSP}}_{i,j}$ for $i = 0, 1, \cdots, M-1$ and $j = 1, 2, \cdots, M$ as follows:

$$\mathcal{L}^{\text{adv}}_{i,j}(T, D_{i,j}) = \mathop{\mathbb{E}}_{x \sim P_{\text{data}}(x)} \left[\log D_{i,j}\left(G^R_{i,j}(x; R) \right) \right] + \mathop{\mathbb{E}}_{x \sim P_{\text{data}}(x)} \left[\log\left(1 - D_{i,j}\left(G^T_{i,j}(x; T) \right) \right) \right]. \tag{13}$$

and

$$\mathcal{L}_{i,j}^{\mathrm{FSP}}(T) = \mathop{\mathbb{E}}_{x \sim P_{\mathrm{data}}(x)} \left[\| G_{i,j}^{R}(x;R) - G_{i,j}^{T}(x;T) \|_{F}^{2} \right]. \tag{14}$$

Then, together with the supervised cross-entropy loss function with true labels, we define a final loss function $\mathcal{L}^{\mathrm{LDF}}$ of the LDF-based KTF for semi-supervised knowledge transfer as follows:

$$\mathcal{L}^{\mathrm{LDF}}(T,D) = \beta \mathcal{L}^{\mathrm{cls}}(T) + \sum_{i=0}^{M-1} \sum_{j=i+1}^{M} \left(\alpha \mathcal{L}_{i,j}^{\mathrm{adv}}(T, D_{i,j}) + \alpha\gamma \mathcal{L}_{i,j}^{\mathrm{FSP}}(T) \right), \tag{15}$$

where α, β, and γ denote positive control parameters for the adversarial-based knowledge optimization.

First, for the optimization of densely designed discriminators, we set the variables T and D according to the well-known Gaussian distribution-based weight initialization. Then, we simultaneously update the discriminators with D by maximizing the adversarial loss of (15) while freezing the variables of T in the target ResNet. In this stage, each discriminator makes an optimal binary decision of whether the flow-based feature is generated by the target ResNet or the pre-trained ResNet.

Next, we update the target ResNet with T by applying a stochastic gradient descent to (15) with respect to T for fixed variables of D. The target ResNet tries to generate LDF-based features similar to the real LDF-based features of the pre-trained ResNet. Simultaneously, the target ResNet is trained to perform an ordinary classification task using real labels.

Therefore, adversarial-based knowledge transfer using dense flow is implemented alternatively to update D and T until the number of iterations reaches a predefined threshold. The entire learning procedure for the proposed method is summarized in Algorithm 1.

Algorithm 1 Adversarial-based knowledge transfer procedure with LDF

1: Load the weights of a pre-trained network R
2: Initialize the variables of a target network T
3: Initialize the variables of a discriminator network D
4: **while** does not converge **do**
5: Choose a minibatch $\mathcal{B}$ in a given dataset
6: For fixed D, update T by descending its stochastic gradient: $\nabla_T \mathcal{L}^{\mathrm{LDF}}(\mathcal{B};T,D)$
7: For fixed T, update D by descending its stochastic gradient: $-\nabla_D \mathcal{L}^{\mathrm{LDF}}(\mathcal{B};T,D)$
8: **end while**
9: **return** T

4. Experiments

In this section, we analyze the proposed method using reliable benchmark datasets: CIFAR-10 and CIFAR-100 [44]. First, for CIFAR-10, we considered adapting a ResNet structure with three residual modules with {16,32,64} filters [29] for the pre-trained and target DNNs in a KTF. Second, for CIFAR-100, we used a wide ResNet structure with {64,128,256} four times more than those in the CIFAR-10, considering the small number of training images per class. In this experiment, there are six discriminators: $D_{0,1}$, $D_{0,2}$, $D_{0,3}$, $D_{1,2}$, $D_{1,3}$, and $D_{2,3}$. Each discriminator structure is based on multilayer perceptron [31]. Here, the number of linear units with each discriminator of CIFAR-10 and CIFAR-100 is configured, as shown in Table 1. When each discriminator structure of $D_{i,j}$ is designed with the number of MLP-based linear units, the number is determined by the spatial size of the corresponding $G_{i,j}$. For example, in CIFAR-10, sorting by the number of elements constituting the Gramian matrix is as follows: $G_{0,1} < G_{0,2} = G_{1,2} < G_{0,3} = G_{1,3} < G_{2,3}$. Notably, this is because the dimensions of the Gramian matrices are represented as $G_{0,1} \in \mathbb{R}^{16\times16}$, $G_{0,2}, G_{1,2} \in \mathbb{R}^{16\times32}$, $G_{0,3}, G_{1,3} \in \mathbb{R}^{16\times64}$, and $G_{23} \in \mathbb{R}^{32\times64}$. For this reason, we set the number of linear units of the discriminators in the following order: $D_{0,1} = D_{0,2} = D_{1,2} < D_{0,3} = D_{1,3} = D_{2,3}$. There-

fore, based on several experiments, the final number of linear units of the discriminators is determined in Table 1. Similar to CIFAR-10, in CIFAR-100, the number of linear units of the discriminators was set experimentally, as shown in Table 1, according to the Gramian matrix dimension.

The experimental conditions for the proposed method were as follows: The loss function using (15) has α and β, as shown in Table 2. γ was used in all experiments, 0.01. Both optimizers for the target ResNet and the discriminator used the same RMSProp optimization algorithm [45]. In addition, 64,000 iterations were performed, and a batch size of 256 was used when one target ResNet was trained in the KTF. Notably, lr_T and lr_D have an initial learning rate for training each target ResNet and discriminator. In our experiment, both lr_T and lr_D were trained by applying variable learning rates, where the learning rate changed 0.1 times after 32,000 iterations and 0.01 times after 48,000 iterations.

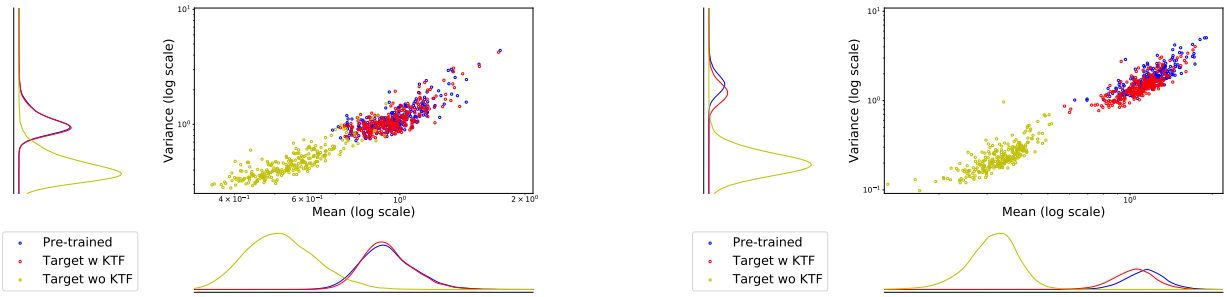
Table 1. The number of linear units in the discriminator.

Discriminators	CIFAR-10	CIFAR-100
$D_{0,1}$	6	14
$D_{0,2}$	6	14
$D_{0,3}$	8	15
$D_{1,2}$	6	14
$D_{1,3}$	8	15
$D_{2,3}$	8	15

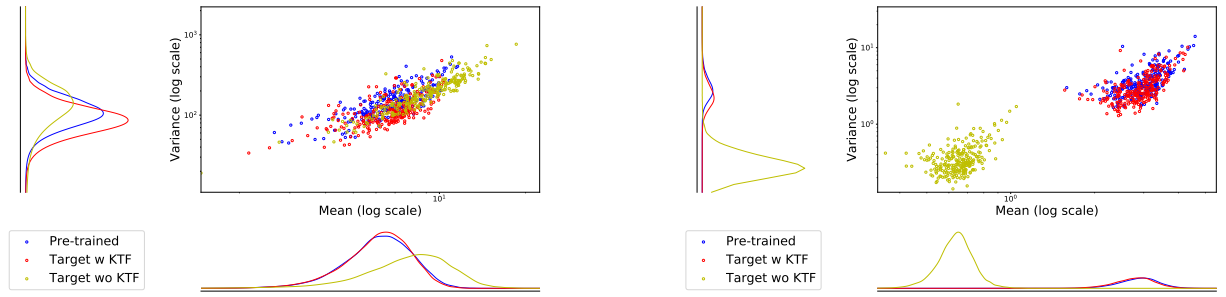
4.1. Dense Flow-Based Knowledge Distribution

This section discusses the ability of the proposed method for delivering LDF-based distilled knowledge in the KTF. To evaluate how well the target ResNet learned pre-trained knowledge, we used the LDF-based knowledge distribution as a performance metric. In Figure 3, the proposed method shows that the Gramian matrix distribution results in (i) the pre-trained ResNet to transmit LDF-based knowledge and (ii) the target ResNet to receive the transferred knowledge. In addition, to derive more specific results, we experimented with simple Gramian distributions of the original ResNet without pre-trained knowledge. In Figure 3, we used CIFAR-100 as the training dataset and adopted a 32-layer ResNet and an 8-layer ResNet, respectively, as the pre-trained and target DNNs in the KTF.

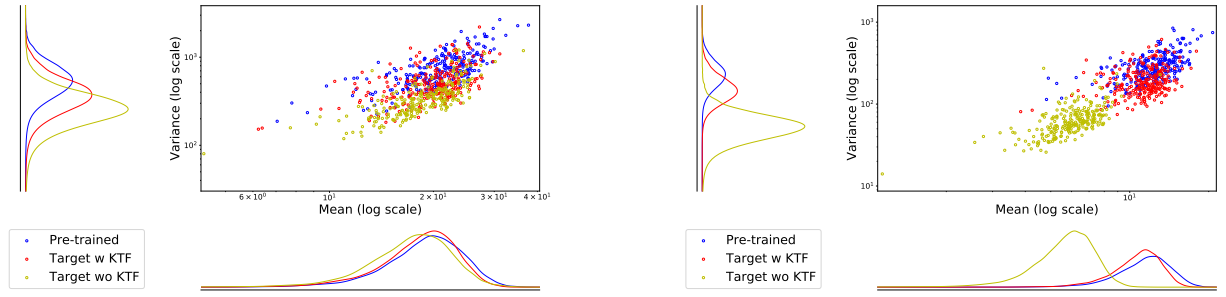
We have observed that all Gramian matrix distributions of the original 8-layer ResNet, which does not take knowledge transfer for the learning process, are significantly different from the distributions of the pre-trained ResNet. However, using the proposed method, the target ResNet can yield distributions with a higher learning performance, and it is largely similar to the pre-trained ResNet. Furthermore, as shown in the distribution table in Figure 3, the obtained knowledge involved with low-level features has significant training results for the distributed information from the pre-trained ResNet. However, although the distributions between the pre-trained and target ResNets are generally in agreement, the knowledge based on high-level features, such as $G_{0,3}$, $G_{1,3}$, and $G_{2,3}$, yields slightly lower learning performance, compared to the distributions for $G_{0,1}$, $G_{0,2}$, and $G_{1,2}$.



(**a**) $G_{0,1}$ (**b**) $G_{0,2}$



(**c**) $G_{0,3}$ (**d**) $G_{1,2}$



(**e**) $G_{1,3}$ (**f**) $G_{2,3}$

Figure 3. Distributions of FSP matrices for the 32-layer pre-trained, 8-layer target, and original 8-layer ResNets without knowledge. In this figure, the blue, red, and yellow points denote the mean and variables of the FSP matrices for the pre-trained, target, and original ResNets, respectively.

4.2. Evaluation of the Proposed Method for Dense Flow

In these experiments, we compare the performance of the proposed adversarial-based method with that of the existing l^2-norm-based method from a knowledge transfer perspective for the LDF-based distilled knowledge. The related parameters in (15) and hyper-parameters related to learning rates are given in Table 2.

Table 2. Hyper-parameters in experiments for training datasets and target networks.

CIFAR-10	**8-layer**	**14-layer**	**20-layer**
α	0.005	0.1	1.0
β	1.0	0.01	0.01
lr_T	0.01	0.01	0.01
lr_D	0.01	0.0125	0.001
CIFAR-100	**8-layer**	**14-layer**	**20-layer**
α	1.0	1.0	0.1
β	0.01	0.01	0.01
lr_T	0.005	0.01	0.01
lr_D	0.001	0.005	0.001

For the same pre-trained ResNet, Table 3 shows the training results based on the CIFAR-10 dataset using a 14-layer target ResNet. In addition, Table 4 presents the training results with the same dataset using a 20-layer target ResNet to solve a classification problem. In addition, Table 5 shows the results of using a 32-layer pre-trained and a 14-layer target ResNets for the CIFAR-100 dataset. In Table 3–5, we can observe that the proposed method has better accuracy than the prior l^2-norm-based methods [34] that follow both sequential and concurrent training approaches mainly based on a dense flow-based scheme. Here, the sequential training involves repetitive sequential knowledge transfer of dense flow from bottom to top between pre-trained and target DNNs, whereas the concurrent approach involves the simultaneous transmission of dense flow into a target DNN. Furthermore, we compared the relative difference between the performance of the target and pre-trained DNNs owing to the difference in performance between the pre-trained DNN model used in the experiment [34] and the pre-trained DNN model used in our experiment. The relative difference is calculated as

$$\text{Relative difference} = \frac{\text{Acc}^T - \text{Acc}^P}{\text{Acc}^P} \times 100,$$

where Acc^P and Acc^T denote the accuracies of the pre-trained and target DNNs, respectively. In CIFAR-10, there was no significant difference in classification accuracy performance between the 26-layer pre-trained ResNet adopted in [34] (Acc^P=91.91%) and the pre-trained ResNet used in our experiment (Acc^P=91.79%). In contrast, we can observe that the 32-layer pre-trained ResNet in [34] provides a lower performance for CIFAR-100 than the 32-layer pre-trained ResNet used in our experiment, although the two pre-trained ResNets have the same number of layers. Notably, this is because the pre-trained ResNet model used in [34] did not adopt any data pre-processing, resulting in Acc^P=64.69%. To rephrase, we adopted the pre-trained ResNet with data pre-processing in our experiment [15], resulting in Acc^P=74.70%. According to [15], we used a random crop of size 32×32 after 4×4 padding with a pre-processing of random flip, and the same pre-processing method was used for knowledge transfer. Tables 3–5 show that the relative difference using the proposed adversarial training can produce better results compared to the previous l^2-norm-based training experiments [34]. In addition, it can be observed that more complex and deeper ResNet structures can yield better accuracy in terms of knowledge transfer in the proposed method.

Table 3. Accuracy (%) of CIFAR-10 for the 26-layer pre-trained and 14-layer target ResNets.

Methods	Pre-Trained	1st	2nd	3rd	Average
Sequential dense flow training (l^2) [34]	91.91	91.8	91.35	91.61	91.58 (0.36% ↓)
Concurrent dense flow training (l^2) [34]		91.16	90.9	90.98	91.01 (0.98% ↓)
Adversarial-based concurrent LDF training (ours)	91.79	91.6	91.45	91.47	91.51 (0.31% ↓)

Table 4. Accuracy (%) of CIFAR-10 for the 26-layer pre-trained and 20-layer target ResNets.

Methods	Pre-Trained	1st	2nd	3rd	Average
Sequential dense flow training (l^2) [34]	91.91	92.29	92.19	92.20	92.22 (0.34% ↑)
Adversarial-based concurrent LDF training (ours)	91.79	92.50	92.33	92.33	92.39 (0.65% ↑)

Table 5. Accuracy (%) of CIFAR-100 for the 32-layer pre-trained and 14-layer target ResNets.

Methods	Pre-Trained	1st	2nd	3rd	Average
Sequential dense flow training (l^2) [34]	64.69	64.95	65.06	65.03	65.01 (0.49% ↑)
Concurrent dense flow training (l^2) [34]		63.46	63.98	64.15	63.86 (1.28% ↓)
Adversarial-based concurrent LDF training (ours)	74.70	75.92	75.85	75.89	75.89 (1.59% ↑)

4.3. Comparison of Knowledge Transfer Performance

In addition, as shown in Tables 6 and 7, we compare the results between the existing knowledge transfer methods and the proposed method. In both existing methods, flow-based knowledge was chosen as the distilled knowledge of the pre-trained DNNs. Conversely, the difference between these two techniques is the loss function design used for knowledge transfer in a KTF. In essence, knowledge transfer using l^2-loss was performed in [3] to calculate the cost function of the flow-based distilled knowledge. In the previous method [31], the adversarial loss was used in the cost function to transfer the flow-based knowledge. In contrast to these two methods, we proposed an adversarial-based knowledge transfer method coupled with the layer-wise overlapping dense flow.

The performance shown in Tables 6 and 7 is the average of the three high values extracted from five experiments. The results indicate that both of the existing methods of [3] and [31] have better classification accuracy than all original ResNets trained without knowledge transfer approach. However, we can observe that the performance of the obtained target ResNet using the proposed approach outperforms the two methods. As mentioned in Section 4.2, a deeper and more complex network structure can obtain better performance enhancement.

In Table 8, the experimental results represent the total number of floating-point operations (FLOPs) required to infer pre-trained and target ResNets. FLOPs ratio (T/R) in Table 8 represents the ratio between the total number of FLOPs in the target ResNet and that of FLOPs in the pre-trained ResNet. First, the CIFAR-10 results in Table 6 show that the performance of the pre-trained and target DNNs in the KTF is largely similar to each other when the target DNN is a 14-layer ResNet. However, the number of FLOPs for inference is 50% or less as shown in Table 8. In addition, the 20-layer target ResNet obtained using the proposed method is superior to the pre-trained 26-layer ResNet, which has more layers and provides improved accuracy compared to the two existing knowledge transfer methods. Subsequently, for CIFAR-100, as shown in Table 7 and 8, the 14-layer target ResNet performs 1.2% higher than the 32-layer pre-trained ResNet but only 37.6% of inference complexity. In particular, the classification accuracy of the 20-layer ResNet in Table 7 is 77.32%, which is slightly higher or similar to 77.29% of the 1001-layer ResNet performance [46].

Table 6. Averaged accuracy (%) of CIFAR-10.

Pre-Trained (26-Layer)		**91.79**	
	8-Layer	**14-Layer**	**20-Layer**
Original ResNet	88.06	90.22	91.06
FSP [3]	88.7	90.92	92.14
Adversarial KTF [31]	88.78	91.35	91.78
Adversarial-based concurrent LDF training (ours)	89.19	91.51	92.39

Table 7. Averaged accuracy (%) of CIFAR-100.

Pre-Trained (32-Layer)		**74.7**	
	8-Layer	**14-Layer**	**20-Layer**
Original ResNet	69.02	73.16	73.54
FSP [3]	71.95	74.81	75.29
Adversarial KTF [31]	72	75.14	75.69
Adversarial-based concurrent LDF training (ours)	73.35	75.89	77.32

Table 8. Comparison results of total number of FLOPs for inference in the proposed method on CIFAR-10 and CIFAR-100 between the target networks and the pre-trained network.

CIFAR-10	**8-layer**	**14-layer**	**20-layer**	**26-layer (Pre-trained)**
# Inference FLOPs	392,927	880,799	1,368,671	1,856,543
FLOPs ratio (T/R)	21.16%	47.44%	73.72%	100.00%
CIFAR-100	**8-layer**	**14-layer**	**20-layer**	**32-layer (Pre-trained)**
# Inference FLOPs	6,263,951	14,021,519	21,779,087	37,294,223
FLOPs ratio (T/R)	16.80%	37.60%	58.40%	100.00%

5. Conclusions

In this study, we proposed an adversarial-based knowledge transfer approach using densely distilled layer-wise flow-based knowledge of a pre-trained deep neural network for

image classification tasks. The proposed knowledge transfer framework was composed of a pre-trained ResNet to extract LDF-based knowledge, a given target ResNet to receive extracted knowledge, and densely placed discriminators to transfer adversarial optimization-based knowledge. In particular, to process LDF-based knowledge distilled from the pre-trained ResNet, the proposed framework was implemented by a semi-supervised learning technique using numerous discriminators for adversarial training and true labels for conventional training. In addition, we designed several adversarial-based loss functions suitable for densely distilled flow-based knowledge transfer. Regarding the loss functions, the l^2 distance-based loss function using densely generated FSP matrices was considered in the proposed framework to deliver more LDF-based feature information to a target ResNet while maintaining stability through adversarial optimization-based knowledge transfer. According to the devised loss functions and adversarial-based knowledge transfer scheme, the proposed method can concurrently update the numerous discriminators and target ResNet.

To validate the performance of the proposed method in terms of knowledge transfer accuracy, we used reliable benchmark datasets such as CIFAR-10 and CIFAR-100 and considered various ResNet architectures with different numbers of layers for a pre-trained source and target models. For all LDF distributions, the results demonstrated that the proposed approach more accurately transferred pre-trained rich information of dense flow between low-level detailed and high-level abstract knowledge compared to the existing l^2-norm-based approach. Furthermore, the small target ResNet obtained from the proposed layer-wise concurrent training yielded higher accuracy than the existing knowledge transfer methods considered in this study or even the original complex pre-trained ResNet. In future work, we plan to use more complicated CNN-based architectures to further analyze the effect of knowledge distributions so that the parameters of the discriminators can be dynamically optimized in the adversarial learning process for a flow-based feature that has a two-dimensional image shape. We will also apply and analyze knowledge transfer proposed in this study to other DNN models that have a different form from the ResNet in future research.

Author Contributions: Conceptualization, D.Y., M.-S.K., and J.-H.B.; Methodology, D.Y.; Visualization, D.Y.; Writing—original draft, M.-S.K. and J.-H.B.; Writing—review and editing, D.Y., M.-S.K., and J.-H.B.; Supervision, J.-H.B. and M.-S.K. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by a National Research Council of Science & Technology (NST) grant from the Korean government (MSIP) (No. CRC-15-05-ETRI).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Simonyan, K.; Zisserman, A. Two-stream convolutional networks for action recognition in videos. In *Advances in Neural Information Processing Systems*; MIT Press: Cambridge, MA, USA, 2014; pp. 568–576.
2. Girshick, R. Fast r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Las Condes, Chile, 11–18 December 2015; pp. 1440–1448.
3. Yim, J.; Jung, H.; Yoo, B.; Choi, C.; Park, D.; Kim, J. Rotating your face using multi-task deep neural network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 676–684.
4. Donahue, J.; Anne Hendricks, L.; Guadarrama, S.; Rohrbach, M.; Venugopalan, S.; Saenko, K.; Darrell, T. Long-term recurrent convolutional networks for visual recognition and description. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 2625–2634.
5. Brahimi, M.; Boukhalfa, K.; Moussaoui, A. Deep learning for tomato diseases: Classification and symptoms visualization. *Appl. Artif. Intell.* **2017**, *31*, 299–315. [CrossRef]

6. Roy, S.; Das, N.; Kundu, M.; Nasipuri, M. Handwritten isolated bangla compound character recognition: A new benchmark using a novel deep learning approach. *Pattern Recognit. Lett.* **2017**, *90*, 15–21. [CrossRef]
7. Liu, G.; Bao, H.; Han, B. A stacked autoencoder-based deep neural network for achieving gearbox fault diagnosis. *Math. Probl. Eng.* **2018**, *2018*, 1–10. [CrossRef]
8. Bae, J.H.; Yeo, D.; Yoon, D.B.; Oh, S.W.; Kim, G.J.; Kim, N.S.; Pyo, C.S. Deep-Learning-Based Pipe Leak Detection Using Image-Based Leak Features. In Proceedings of the 2018 25th IEEE International Conference on Image Processing (ICIP), Athens, Greece, 7–10 October 2018; pp. 2361–2365.
9. Wang, P.; Wang, H.; Zhang, H.; Lu, F.; Wu, S. A Hybrid Markov and LSTM Model for Indoor Location Prediction. *IEEE Access* **2019**, *7*, 185928–185940. [CrossRef]
10. Wan, J.; Chen, B.; Xu, B.; Liu, H.; Jin, L. Convolutional neural networks for radar HRRP target recognition and rejection. *EURASIP J. Adv. Signal Process.* **2019**, *2019*, 5. [CrossRef]
11. LeCun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. *Proc. IEEE* **1998**, *86*, 2278–2324. [CrossRef]
12. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*; MIT Press: Cambridge, MA, USA, 2012; pp. 1097–1105.
13. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9.
14. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
15. 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 (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016; pp. 770–778.
16. Veit, A.; Wilber, M.; Belongie, S. Residual networks are exponential ensembles of relatively shallow networks. *arXiv* **2016**, *1*, 3, arXiv:1605.06431
17. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4700–4708.
18. Tan, M.; Le, Q.V. Efficientnet: Rethinking model scaling for convolutional neural networks. *arXiv* **2019**, arXiv:1905.11946.
19. Zeiler, M.D.; Fergus, R. Visualizing and understanding convolutional networks. In *European Conference on Computer Vision*; Springer: Berlin/Heidelberg, Germany, 2014; pp. 818–833.
20. Tan, C.; Sun, F.; Kong, T.; Zhang, W.; Yang, C.; Liu, C. A survey on deep transfer learning. In *International Conference on Artificial Neural Networks*; Springer: Berlin/Heidelberg, Germany, 2018; pp. 270–279.
21. 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. *Int. J. Comput. Vis.* **2015**, *115*, 211–252. [CrossRef]
22. Yosinski, J.; Clune, J.; Bengio, Y.; Lipson, H. How transferable are features in deep neural networks? In *Advances in Neural Information Processing Systems*; MIT Press: Cambridge, MA, USA, 2014; pp. 3320–3328.
23. Jeon, D.; Kim, M.S.; Ryu, S.J.; Lee, D.H.; Kim, J.K. Fully Printed Chipless RFID Tags Using Dipole Array Structures with Enhanced Reading Ranges. *J. Electromagn. Eng. Sci.* **2017**, *17*, 159–164. [CrossRef]
24. Lee, S.; Yoon, H.; Baik, K.J.; Jang, B.J. Emulator for Generating Heterogeneous Interference Signals in the Korean RFID/USN Frequency Band. *J. Electromagn. Eng. Sci.* **2018**, *18*, 254–260. [CrossRef]
25. Yoon, Y.S.; Zo, H.; Choi, M.; Lee, D.; Lee, H.w. Exploring the dynamic knowledge structure of studies on the Internet of things: Keyword analysis. *ETRI J.* **2018**, *40*, 745–758. [CrossRef]
26. Li, J.; Zhao, R.; Huang, J.T.; Gong, Y. Learning small-size DNN with output-distribution-based criteria. In Proceedings of the Fifteenth Annual Conference of the International Speech Communication Association, Singapore, 14–18 September 2014.
27. Hinton, G.; Vinyals, O.; Dean, J. Distilling the knowledge in a neural network. *arXiv* **2015**, arXiv:1503.02531.
28. Romero, A.; Ballas, N.; Kahou, S.E.; Chassang, A.; Gatta, C.; Bengio, Y. Fitnets: Hints for thin deep nets. *arXiv* **2014**, arXiv:1412.6550.
29. Yim, J.; Joo, D.; Bae, J.; Kim, J. A Gift From Knowledge Distillation: Fast Optimization, Network Minimization and Transfer Learning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 4133–4141.
30. Xu, Z.; Hsu, Y.C.; Huang, J. Training Shallow and Thin Networks for Acceleration via Knowledge Distillation with Conditional Adversarial Networks. *arXiv* **2017**, arXiv:1709.00513.
31. Yeo, D.; Bae, J.H. Multiple flow-based knowledge transfer via adversarial networks. *Electron. Lett.* **2019**, *55*, 989–992. [CrossRef]
32. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial nets. In *Advances in Neural Information Processing Systems*; MIT Press: Cambridge, MA, USA, 2014; pp. 2672–2680.
33. Hu, C.; Wu, X.J.; Shu, Z.Q. Bagging deep convolutional autoencoders trained with a mixture of real data and GAN-generated data. *KSII Trans. Internet Inf. Syst.* **2019**, *13*, 5427–5445.
34. Bae, J.H.; Yeo, D.; Yim, J.; Kim, N.S.; Pyo, C.S.; Kim, J. Densely distilled flow-Based knowledge transfer in teacher-student framework for image classification. *IEEE Trans. Image Process.* **2020**, *29*, 5698–5710. [CrossRef]
35. Metz, L.; Poole, B.; Pfau, D.; Sohl-Dickstein, J. Unrolled generative adversarial networks. *arXiv* **2016**, arXiv:1611.02163.
36. Arjovsky, M.; Chintala, S.; Bottou, L. Wasserstein gan. *arXiv* **2017**, arXiv:1701.07875.

37. Mao, X.; Li, Q.; Xie, H.; Lau, R.Y.; Wang, Z.; Paul Smolley, S. Least squares generative adversarial networks. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2794–2802.
38. Salimans, T.; Goodfellow, I.; Zaremba, W.; Cheung, V.; Radford, A.; Chen, X. Improved techniques for training gans. In *Advances in Neural Information Processing Systems*; MIT Press: Cambridge, MA, USA, 2016; pp. 2234–2242.
39. Radford, A.; Metz, L.; Chintala, S. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv* **2015**, arXiv:1511.06434.
40. Ledig, C.; Theis, L.; Huszar, F.; Caballero, J.; Cunningham, A.; Acosta, A.; Aitken, A.; Tejani, A.; Totz, J.; Wang, Z.; et al. Photo-realistic single image super-resolution using a generative adversarial network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 4681–4690.
41. Wei, Z.; Bai, H.; Zhao, Y. Stage-GAN with Semantic Maps for Large-scale Image Super-resolution. *KSII Trans. Internet Inf. Syst.* **2019**, *13*, 3942–3961.
42. Chen, J.; Chao, H.; Yang, M. Image blind denoising with generative adversarial network based noise modeling. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018; pp. 3155–3164.
43. Cheng, J.; Yang, Y.; Tang, X.; Xiong, N.; Zhang, Y.; Lei, F. Generative Adversarial Networks: A Literature Review. *KSII Trans. Internet Inf. Syst.* **2020**, *14*.
44. Krizhevsky, A.; Nair, V.; Hinton, G. The CIFAR-10 Dataset and CIFAR-100 Dataset. 2014 Available online: http://www.cs.toronto.edu/~kriz/cifar.html (accessed on 20 April 2021).
45. Ruder, S. An overview of gradient descent optimization glgorithms. *arXiv* **2016**, arXiv:1609.04747.
46. He, K.; Zhang, X.; Ren, S.; Sun, J. Identity mappings in deep residual networks. In *European Conference on Computer Vision*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 630–645.