Multi-Objective Instance Weighting-Based Deep Transfer Learning Network for Intelligent Fault Diagnosis

Kihoon Lee [1], Soonyoung Han [1], Van Huan Pham [1], Seungyon Cho [1], Hae-Jin Choi [1,2,*], Jiwoong Lee [3], Inwoong Noh [3] and Sang Won Lee [3]

[1] School of Mechanical Engineering, Chung-Ang University, 84 Heukseok-Ro, Dongjak-Gu, Seoul 06974, Korea; nukoori1212@cau.ac.kr (K.L.); hsy4190@cau.ac.kr (S.H.); pvh134@cau.ac.kr (V.H.P.); chosy28@cau.ac.kr (S.C.)

[2] Department of Computer Science and Engineering, Chung-Ang University, 84 Heukseok-Ro, Dongjak-Gu, Seoul 06974, Korea

[3] School of Mechanical Engineering, Sungkyunkwan University, 2066 Seobu-Ro, Jangan-Gu, Suwon 16419, Korea; ljwg3000@skku.edu (J.L.); freedomniw@skku.edu (I.N.); sangwonl@skku.edu (S.W.L.)

* Correspondence: hjchoi@cau.ac.kr

Abstract: Fault diagnosis is a top-priority task for the health management of manufacturing processes. Deep learning-based methods are widely used to secure high fault diagnosis accuracy. Actually, it is difficult and expensive to collect large-scale data in industrial fields. Several prerequisite problems can be solved using transfer learning for fault diagnosis. Data from the source domain that are different but related to the target domain are used to increase the diagnosis performance of the target domain. However, a negative transfer occurs that degrades diagnosis performance due to the transfer when the discrepancy between and within domains is large. A multi-objective instance weighting-based transfer learning network is proposed to solve this problem and successfully applied to fault diagnosis. The proposed method uses a newly devised multi-objective instance weight to deal with practical situations where domain discrepancy is large. It adjusts the influence of the domain data on model training through two theoretically different indicators. Knowledge transfer is performed differentially by sorting instances similar to the target domain in terms of distribution with useful information for the target task. This domain optimization process maximizes the performance of transfer learning. A case study using an industrial robot and spot-welding testbed is conducted to verify the effectiveness of the proposed technique. The performance and applicability of transfer learning in the proposed method are observed in detail through the same case study as the actual industrial field for comparison. The diagnostic accuracy and robustness are high, even when few data are used. Thus, the proposed technique is a promising tool that can be used for successful fault diagnosis.

Keywords: deep learning; fault diagnosis; industrial robot; prognostics and health management (PHM); spot welding; transfer learning

1. Introduction

Fault diagnosis is one of the significant tasks covered by PHM (Prognostics and Health Management). It includes the process of monitoring mechanical equipment to determine the current health state and predict when and what failures occur. It enables decision-making for the maintenance and efficient management of the production process. Various techniques have been developed to increase the reliability of fault diagnosis. Among them, fault diagnosis methods that use machine learning are primarily used. Intelligent fault diagnosis refers to the use of machine learning to detect and diagnose faults, and to manage health information in mechanical equipment [1]. Performing intelligent fault diagnosis effectively is one of the foundations for smart manufacturing [2,3].

Traditional fault diagnosis using machine learning consists of three stages: data acquisition, artificial feature extraction and selection, and fault classification.

In the data acquisition step, sensor signal data are collected following the attachment of various sensors to mechanical equipment to diagnose problems. In general, signals such as acoustic, vibration, current, and thermal are collected for fault diagnosis. Each signal has advantages and disadvantages and is chosen to be appropriate for the subject being diagnosed. Acoustic emission data can detect incipient and hidden faults, and are used to diagnose bearings [4], gears [5], and induction motors [6]. It is particularly useful for low speed operation conditions or low-frequency noise environments. The thermal signal is collected in several forms, and there is a possibility that it can be applied noninvasively without contacting the driving part, such as a thermal image. These signals are difficult to process and are used for fault diagnosis in induction motors [7,8] and electric impact drills [9]. In the case of electric current signal, it can be easily collected from the current transformer and has a great advantage in diagnosing electric-driven machines [10,11]. Among these signals, the vibration signal is a useful and effective tool widely used for fault diagnosis of various kinds of machines. In order to remove the influence of external interference and noise in the operating environments and to be used as a low noise level, various signal processing is performed [12,13]. It is used to diagnose mechanical and electrical failures of various equipment such as bearings [14], gearboxes [15], commutator motors [16].

In the next step, various features, including the time domain, frequency domain, and time-frequency domain, are artificially extracted from the collected signals through signal processing methods. Then, specific feature selection methods are used to select only important features with information about health states. Subsequently, health states, such as failure, are predicted using the conventional machine learning. A number of machine learning techniques were used for fault diagnosis such as artificial neural network (ANN) [17], support vector machine (SVM) [18], and classification and regression trees (CART) ensemble [19].

Traditional fault diagnosis has the following problems. First, the feature extraction and selection processes are passive and inefficient. Feature extraction relies on the choice of methods based on experience or knowledge. Each handcrafted feature tends to be suitable only for fault diagnosis under certain circumstances, requiring the operator's judgment in many situations. Additionally, a specialized extraction method should be used according to the shape and characteristics of the sensor signal. As the type of collected sensor signal diversifies and the volume increases, it becomes impossible. As a result, the process reliability reduces, and traditional techniques become inappropriate for the fault diagnosis task. Second, these problems make it difficult to generalize and automate the fault diagnosis procedure. Generalizing and automating the entire procedure for use in diverse circumstances is necessary to increase the work efficiency and diagnosis accuracy.

Therefore, deep learning-based methods are used as an alternative. Deep learning-based methods are used for various purposes in several fields and are constantly evolving. In the PHM field, deep learning-based methods are also dominant in data-driven methods for fault diagnosis and prediction. Vibration signal data are also widely used for fault diagnosis using deep learning. After converting vibration signal into a spectrogram that can visualize the frequency band and time change of original signal, diagnosis performance can be dramatically improved by deep learning modeling such as convolutional neural network (CNN), a deep neural network widely used for visual image analysis. CNN is useful for time-frequency characteristic analysis because it extracts spatial information using image-specialized convolution calculations. By extracting high-dimensional information that cannot be distinguished by previous techniques through deep learning models, different health conditions can be easily distinguished, and end-to-end feature learning and classification are performed simultaneously. However, in the case of vibration signal data, there is a disadvantage in that many sensors must be attached to acquire data according to the characteristics of the object to be applied. In addition, problems such as optimization of the number of sensors and installation location may occur. Therefore, there are some studies that performed fault diagnosis and condition monitoring by building image-based

deep learning models using thermal image data. The deep learning-based diagnosis model varies in its structure. Based on this, it can predict the health state of mechanical equipment. Ma et al. [20] used time-frequency analysis and a deep residual network to perform fault diagnosis of a planetary gearbox under nonstationary running conditions. Zhang et al. [21] conducted a fault diagnosis of rolling bearings using a deep residual network.

The accuracy of such deep learning-based diagnosis models is closely related to the number of collected data. When sufficient training data are available, deep learning-based diagnosis models with complex structures outperform other diagnosis models. However, if a small number of valid training data are collected directly from the task to diagnose faults, the reliability and accuracy of the diagnosis model inevitably decrease. In addition, training data and test data used for deep learning-based diagnosis models must have the same distribution. In actual industrial fields, the cost required to collect quality data is very high. Therefore, few data are actually collected from machine equipment under the specific work environment for diagnosis.

The application of transfer learning can solve these problems in deep learning-based diagnosis models. Even if the labeled data are insufficient in the target domain to be diagnosed, there are data collected under different operating conditions. These different but related data can be used as the source domain for training a diagnosis model. The feature learning ability can be obtained through the source domain with sufficient labeled data and transfer learning. High diagnosis accuracy and stability are obtained by using source domain data for the target task. Transfer learning is essential for comprehensive fault diagnosis in industrial fields where identical process work is performed under various operating conditions. In recent years, studies using transfer learning have been actively conducted primarily for semi-supervised or unsupervised learning tasks, and few studies have conducted supervised learning tasks. Shao et al. [22] transferred the structure and parameters of the VGG16 network trained from the ImageNet dataset for image recognition and used them for fault diagnosis of an induction motor, bearing, and planetary gearbox. Yang et al. [23] used a CNN and transfer learning for fault diagnosis of rotating machinery under different working conditions. In addition, Cao et al. [24] performed gear fault diagnosis using a deep CNN and transfer learning.

The performance of transfer learning is determined by the size of the labeled target data and the discrepancy between the source domain and target domain [25]. Therefore, the process evaluating the attributes of the source domain instances that are useful for target domain task and using them as additional resources for model training effectively improves the performance of transfer learning. This is the main concept of instance-based transfer learning.

Various optimization methods have been developed and used for conducting instance-based transfer learning. Wang et al. [26] also conducted instance-based transfer learning using the dropout algorithm by removing target instances that are not well classified by a classifier. Dropout is a useful tool for reducing dissimilarity between domains when enough labeled data are available for each source or target domain. Zhang et al. [27] carried out the dropout of unsuitable source instances up to a certain number for a situation in which the number of target domain data is much smaller than that of the source domain. The combination of target dropout optimization, SAE-SVM, and ensemble learning resulted in the successful fault diagnosis of the ball screw. Dropout is a domain optimization technique suitable for situations where sufficient labeled data of the domain exist to drop out. If dropout is used when insufficient labeled data exist in each domain, model performance decreases due to a lack of data required for training model. Moreover, this is unsuitable for dealing with situations in which the dissimilarity between domains is large because all instances that are not removed have the same influence on model training.

The instance-weighting strategy can solve these problems. It is noted that the accuracy of the method can be improved by focusing on the ensemble technique such as boosting. For instance, Bustillo et al. [28] performed AdaBoost using a small size dataset collected from various operating conditions to predict the quality of friction drilling and to find

the optimum parameters. Especially, Dai et al. [29] proposed TrAdaBoost, which grafts AdaBoost applying instance-weighting strategy on to the transfer learning concept. It includes updating the weight vector due to the distribution difference between the source domain and target domain. Yao et al. [30] expanded TrAdaBoost to deal with multi-source cases. The key idea is that, among the diff-distribution training instances from different domains, instances that are not well fitted to the target domain have low training weights and become less influential in later training stages.

In this paper, we propose the multi-objective instance weighting-based deep transfer learning network (MOTL). The proposed method is applicable not only for fault diagnosis of general cases but also for fault diagnosis when data are collected under different operating conditions in a practical range. A significant distribution difference between source and target domains exists, and only a small number of labeled data are collected from each domain. In this method, source instance weights calculated using Kullback–Leibler divergence (KLD) [31] and the maximum mean discrepancy (MMD) [32] (two indicators evaluating the distribution difference between domains) and a large volume of source domain data are used as auxiliary training resources to train a deep residual network. Afterward, the structure and parameters of the pre-trained model are transferred, and fine-tuning is conducted using a small number of labeled target data.

In addition, a spot-weld case study using an industrial robot is conducted to verify the applicability to actual industrial fields, which is one of the significant advantages of transfer learning. The main contributions of this paper are summarized as follows.

1. We present a fault diagnosis framework based on instance-based transfer learning using multi-objective instance weighting to diagnose faults when few labeled target data are available in the target task. This framework helps achieve high diagnosis accuracy and robustness in high dissimilarity situations between and within domains due to distinct operating conditions. The proposed method uses instance weights obtained from the two complementary dissimilarity indicators to minimize the dissimilarity between domains that affect model training. The knowledge of source instances suitable for the target task can be transferred through the domain optimization process. It results in improved performance of the target diagnosis model.

2. According to various instance optimization techniques used in instance-based transfer learning, the diagnosis accuracy is compared in detail. Through this comparison, the domain optimization process and effectiveness of the proposed method are confirmed.

3. The accuracy of the diagnosis model using the proposed method and transfer learning is monitored in detail, changing the number of labeled target data. The case study is also conducted through the testbed identical to the actual industrial field, verifying the applicability of the proposed method for diagnosis when the target labeled data are less available at the actual industrial field.

2. Proposed Method

Figure 1 illustrates the overall progress of the proposed MOTL method. First, signal processing is performed to convert the collected vibration signals into an image representing a time-frequency distribution. An initial pre-trained model is obtained using the source domain data and fine-tuned once with the target domain data. ResNet50 [33] is used as the model structure for all model training. Then, the KLD and MMD are calculated from the source domain data and the initial deep transfer model. Multi-objective instance weights are computed from these two indicators. The instance-weight vector is used to train the optimized pre-trained model as a training weight. The final target fault diagnosis model is obtained by transferring the optimized pre-trained model and fine-tuning it. The two-stage transfer process is performed to derive the optimal multi-objective instance weights and effectively minimize the influence of domain discrepancy. A detailed flowchart of the proposed method is presented in Figure 2.
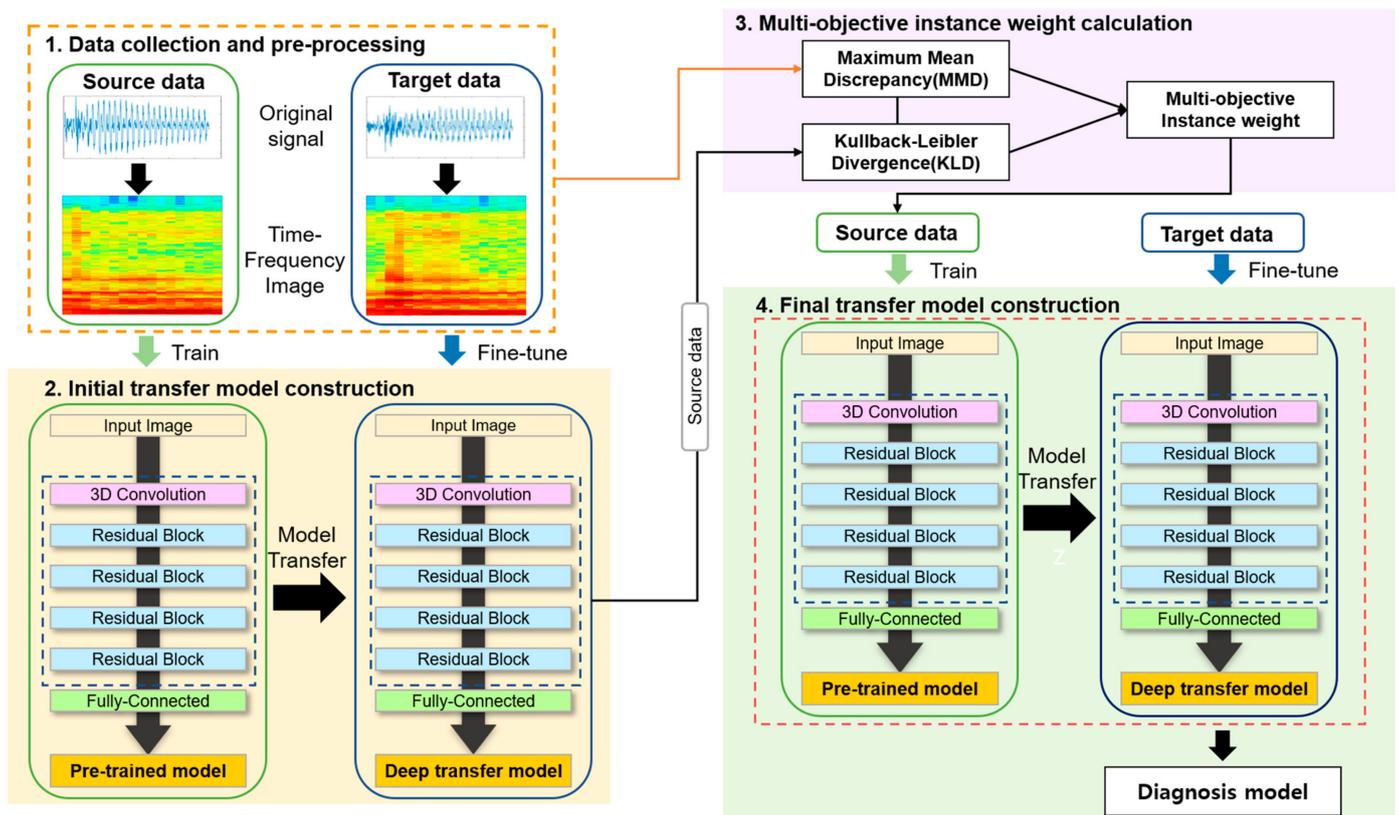
Figure 1. Detailed procedure of the proposed multi-objective instance weighting-based deep transfer learning network (MOTL) method.

2.1. Data Preprocessing (Time-Frequency Domain Imaging)

First, an accelerometer is used to collect vibration signals as work progresses under various circumstances with several different process conditions. The raw vibration signal is converted into a time-frequency domain through various signal processing methods to obtain useful information. The purpose is to extract the characteristics of the frequencies in a specific time domain from time-series sensor data composed of signals from various kinds of components, which are used for diagnosis. The short-time Fourier transformation (STFT), wavelet packet decomposition (WPD), and WPD with spectral subtraction are used to convert signals into the time-frequency domain [34]. The parameters are adjusted in detail using a 50% overlap to increase the resolution of the time domain. Each of these different spectrogram images extracted by the signal processing methods is treated as input to the model.
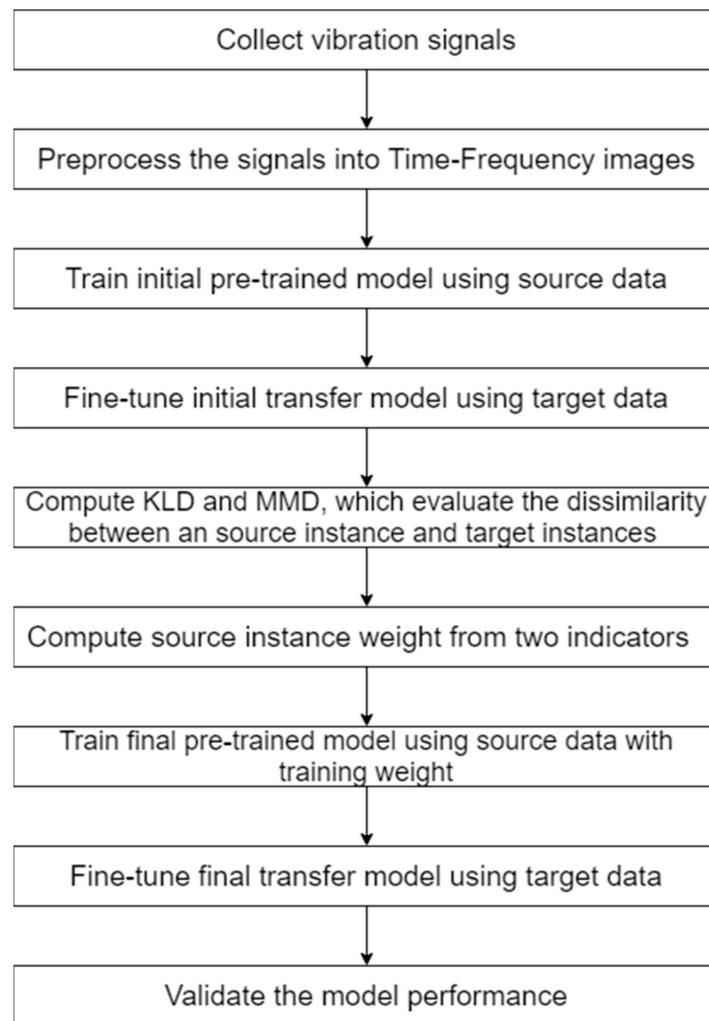
Figure 2. Flowchart of the proposed multi-objective instance weighting-based transfer learning network (MOTL) method.

2.2. Deep Residual Learning Network

Residual network builds up the residual block based on the convolutional layer to form a deep learning model. The residual block consists of the general forward channel and shortcut connections that perform identity mapping. The forward channel has the same output as the common stacked convolutional layer. The shortcut connection performs residual mapping, which adds input features to the forward channel output to derive the final residual block output. The degradation problem of the deep network model can be solved without increasing the number of parameters or model complexity using these residual blocks. A deep residual network is obtained by building these structural blocks. The ResNet50 model used for training consists of a total of 34 parameter layers.

Table 1 describes the detailed structure of ResNet50, which has been used in various areas, including PHM, and is more accurate than the CNN-based diagnosis model [1].

Table 1. Detailed configuration of ResNet50 architecture.

Layer	Type	Data Dimension
	Input	#,224,224,3
Conv1	ConvolutionalLayer	#,112,112,64
bn_conv1	BatchNormalizationLayer	#,112,112,64
conv1_relu	Ramp	#,112,112,64
pool1_pad	PaddingLayer	#,113,113,64
pool1	PoolingLayer	#,56,56,64
2a	ResidualBlock2 (12 nodes)	#,56,56,256
2b	ResidualBlock2 (10 nodes)	#,56,56,256
2c	ResidualBlock2 (10 nodes)	#,56,56,256
3a	ResidualBlock3 (12 nodes)	#,28,28,512
3b	ResidualBlock3 (10 nodes)	#,28,28,512
3c	ResidualBlock3 (10 nodes)	#,28,28,512
3d	ResidualBlock3 (10 nodes)	#,28,28,512
4a	ResidualBlock4 (12 nodes)	#,14,14,1024
4b	ResidualBlock4 (10 nodes)	#,14,14,1024
4c	ResidualBlock4 (10 nodes)	#,14,14,1024
4d	ResidualBlock4 (10 nodes)	#,14,14,1024
4e	ResidualBlock4 (10 nodes)	#,14,14,1024
4f	ResidualBlock4 (10 nodes)	#,14,14,1024
5a	ResidualBlock5 (12 nodes)	#,7,7,2048
5b	ResidualBlock5 (10 nodes)	#,7,7,2048
5c	ResidualBlock5 (10 nodes)	#,7,7,2048
pool5	PoolingLayer	#,1,1,2048
flatten_0	FlattenLayer	#,2048
Fc2	LinearLayer	#,2
prob	SoftmaxLayer	#,2
	Output	class

2.3. Transfer Learning and Fine-Tuning Strategy

This section describes the notation and definitions about transfer leaning. The necessary notation and definitions refer to S. J. Pan's Survey of Transfer Learning [35]. First, domain $\mathcal{D}$ and task $\mathcal{T}$ are defined. Domain $\mathcal{D}$ consists of the collected feature space $\mathcal{X}$ and its marginal probability distribution $P(X)$, where $X = \{x_1, \cdots, x_n\} \in \mathcal{X}$, and n is the number of instances in each domain ($\mathcal{D} = \{\mathcal{X}, P(X)\}$). The symbol X denotes an instance set, which is specific training sample, and x_i is the ith instance in particular domain. Task $\mathcal{T}$ consists of label space $\mathcal{Y}$ and prediction model $f(\cdot)$ ($\mathcal{T} = \{\mathcal{Y}, f(\cdot)\}$). The symbol $Y = \{y_1, \cdots, y_n\} \in \mathcal{Y}$ denotes the label set corresponding to an instance set X, and y_i is a label corresponding to the instance x_i. Prediction model $f(\cdot)$ is not observed and collected but is trained using collected training data. In this paper, the trained prediction model $f(\cdot)$ predicts the corresponding health state label $f(x)$, of an input instance x. In terms of probabilistic method, $f(x)$ can be denoted as $P(y|x)$.

In transfer learning, two types of domains exist: source domain and target domain. The target domain is the domain to proceed with the diagnosis task, whereas the source domain is a domain different but related to the target domain, which has useful information for the target domain. Transfer learning can provide useful knowledge in $\mathcal{D}_S$ and $\mathcal{T}_S$ to improve the performance of target prediction model $f_T(\cdot)$, where $\mathcal{D}_S \neq \mathcal{D}_T$ or $\mathcal{T}_S \neq \mathcal{T}_T$. Thus, a relation that is visibly obvious or inherent must exist between the source and target domains. When this condition is satisfied, the two domains are related. If the source domain is not related to the target domain, the information held by the source domain does not improve the performance of the target prediction model; instead, it causes a negative transfer that reduces performance [26].

We denote source domain data as $D_S = \left\{ \left(x_{S_1}, y_{S_1}\right), \ldots, \left(x_{S_{n_S}}, y_{S_{n_S}}\right) \right\}$, where x_{S_i} and y_{S_i} are i^{th} source instance and class label, respectively. n_S is the number of instances in the source domain. Source domain data D_S are employed for training a pre-trained model. In

the same manner, target domain data is denoted as $D_T = \left\{ (x_{T_1}, y_{T_1}), \ldots, \left(x_{T_{n_T}}, y_{T_{n_T}}\right) \right\}$, where x_{T_i} and y_{T_i} are i^{th} target instance and heath state label in the target domain. n_T is the number of target domain instances. Target domain data D_T is divided into target training data D_T^{train} and target test data D_T^{test}. Target training data D_T^{train} are employed in fine-tuning the pre-trained model, whereas target test data D_T^{test} are used only for validation. Both D_S and D_T are sets of vectors composed of domain instances' features and corresponding true health state labels. A large labeled dataset is available in the source domain, whereas only a small labeled dataset is available in the target domain ($n_S \gg n_T$). In addition, data belonging to each domain follow different marginal probability distributions ($P_S(X) \neq P_T(X)$, where $P_S(X)$, and $P_T(X)$ are marginal probability distributions of source and target domain, respectively).

A fine-tuning strategy is used to train the target diagnosis model. When the deep model is trained with sufficient source domain data, a pre-trained model with feature extraction and classification layers suitable for source domain is obtained. The structure and parameters of the pre-trained model are transferred to target domain and fine-tuned with the target domain data. After these steps, a deep transfer learning model suitable for the task of the target domain is obtained. By using this strategy, incomplete training and overfitting can be avoided, which occur when only few target data are used for training. In addition, the performance of the fine-tuning strategy is determined by the dissimilarity between the source and target domains. Therefore, if the source and target domains are significantly similar, only the fully-connected layer can be fine-tuned, whereas the range of the structure carrying out fine-tuning should increase when the dissimilarity between source and target domain increase.

2.4. Instance-Based Transfer Learning

Instance-based transfer learning focuses on using proper parts of the source domain data for training in the target domain by re-weighting. Instance-based transfer learning approaches are mainly based on the instance weighting strategy. In transfer learning, the source and target domains are used as training data consequently. Both domains have the same conditional probability distributions but follow different marginal probability distributions:

$$P_S(X) \neq P_T(X), \ P_S(Y|X) = P_T(Y|X). \tag{1}$$

The performance of transfer learning depends on the relationship between the source and target domain instances, which determines how much useful knowledge for diagnosing the target domain can be transferred from the source to target domain. The amount of transferred knowledge or distributional characteristics of the instances is crucial factor in determining the transfer learning performance either. However, some instances of the source domain data are not suitable for the target prediction model. Therefore, directly transferring all the large source domain data may not be useful for the task of the target domain. Thus, it is necessary to measure the dissimilarity and distribution difference between the target and source domain instances and use it as a weight that influences model training to improve the transfer learning performance. This is the main concept of instance-based transfer learning using instance weighting. The main objective of instance weighting strategy is to reduce dissimilarity between source and target domain instances. Our instance weighting approach is based on the following equation [36,37]:

$$\begin{aligned} \mathcal{R}[P_T, \ \ell(x,y;f)] &= \mathbb{E}_{(x,y)\sim P_T}[\ell(x,y;f)] = \mathbb{E}_{(x,y)\sim P_S}\left[\frac{P_T(x,y)}{P_S(x,y)}\ell(x,y;f)\right] \\ &= \mathbb{E}_{(x,y)\sim P_S}\left[\frac{P_T(x)}{P_S(x)}\ell(x,y;f)\right] = \mathbb{E}_{(x,y)\sim P_S}[\beta(x,y)\ell(x,y;f)], \end{aligned} \tag{2}$$

where $\mathcal{R}$ is the regularized risk and $\ell(x,y;f)$ is the loss function in the prediction model f using x and y. $\mathbb{E}$ is the expected risk of the loss function, ℓ, and β is instance weight for source domain. The prediction model f is trained to minimize the expected risk $\mathbb{E}$ of the loss function. From the above Equation (2), theoretically, the value of the instance weight

β is the same as $P_T(x) / P_S(x)$. However, the exact ratio of $P_T(x)$ to $P_S(x)$ is usually not known [36].

Therefore, several methods to estimate the weight have been introduced. For example, kernel mean matching (KMM) procedure [37], which is presented by Huang et al., deals with weight estimation by matching the means between source domain and target domain instances in a reproducing kernel Hilbert space (RKHS):

$$\underset{\beta_i}{\text{argmin}} \parallel \frac{1}{n_S} \sum_{i=1}^{n_S} \beta_i \Phi(x_{S_i}) - \frac{1}{n_T} \sum_{j=1}^{n_T} \Phi(x_{T_j}) \parallel_{\mathcal{H}}^2, \tag{3}$$

where β_i is a weighting parameter of ith instance, and Φ is the kernel mapping function. We can obtain instance weight β_i using KMM procedure and apply to model training process. In addition, Sugiyama et al. [38] proposed an approach called the Kullback–Leibler importance estimation procedure (KLIEP) to estimate the weight. It estimates the importance, which means the weight, through a variant of likelihood CV (cross validation) by minimizing the Kullback–Leibler divergence (KLD). These two methods were basically devised to solve the dataset shift problem. It is one of the important assumptions in machine learning that the training data and test data should be drawn from the same distribution. A situation in which the joint distribution of inputs and outputs is different in training and test data is referred to as a dataset shift. It appears when training and test data are collected from different distributions. In the case of transfer learning, the source domain is composed of several kinds of data collected from different situations. Essentially, similar phenomena with dataset shift occur in transfer learning. Therefore, the above methods can be extended and used to solve similar problems occurring in transfer learning.

Once instance weights are derived using specific methods, training proceeds using weighted source domain instances. The objective function for training the pre-trained model using the instance weight is as follows:

$$\underset{f}{\min} \frac{1}{n_s} \sum_{i=1}^{n_s} \mathcal{W}_i \ell(f(x_{S_i}), y_{S_i}), \tag{4}$$

where $\mathcal{W}_i$ ($i = 1, 2, \cdots, n_S$) is the weighting parameter of the source domain instances. Through this, training is conducted to minimize the weighted average of the loss function values. Instances with a large weight have a significant influence on model training.

2.5. Multi-Objective Instance-Weighting Strategy

The main objective of the instance weighting strategy is to reduce distribution dissimilarity between the source and target domain instances. Therefore, it is important to decide how to effectively evaluate the dissimilarity between instances from different domains. Two indicators, KLD and MMD, measure the distribution dissimilarity between domain instances using different theoretical methods. These two indicators are suitable to be used to achieve the objective of instance weighting. However, instance weighting using a specific indicator may not be consistently effective. Their performance varies according to the type or number of domain data used for model training. Even if the data are collected in the same environment, suitability of the indicator in the process varies inevitably depending on the number of training data. Therefore, we use multi-objective instance weighting to minimize the two metrics concurrently. Domain optimization using multi-objective instance weights is conducted to achieve two objectives: (1) to make the target and source domain instances have similar distributions and (2) to transfer useful information to the target domain task.

2.5.1. Kullback–Leibler Divergence

The KLD measures how different a particular probability distribution is from the reference probability distribution based on information theory. Given two probability distributions p and q, the KLD of p and q can be defined as follows:

$$KLD(p \parallel q) = \sum_{j=1}^{n} p(j) \log\left(\frac{p(j)}{q(j)}\right), \tag{5}$$

where n is the number of class label, $p(j)$ refers to the true probability of class j, and $q(j)$ refers to the predicted class probability of class j. It is the expectation of the logarithmic difference between the probabilities $p(j)$ and $q(j)$, where the expectation is taken using the probability $p(j)$. KLD has a nonnegative value. $KLD(p|q) = 0$ if and only if $p = q$. Because KLD is a nonsymmetric measure, it does not directly indicate the distance between two probability distributions.

In our case, we calculated the KLD for each source domain instance to be used as weight for that instance. The KLD of ith source domain instance is formulated as follows:

$$KLD\left(y_{S_i} \parallel \hat{y}_{S_i}\right) = \sum_{j=1}^{n} Pr(j|y_{S_i}) \log\left(\frac{Pr(j|y_{S_i})}{Pr(j|\hat{y}_{S_i})}\right), \tag{6}$$

where $\hat{y}_{S_i}$ denotes the expected probability distribution of ith source domain instance predicted from the initially trained transfer learning network. The KLD measures the amount of helpful information of each source domain instance for the classification task of the target domain in terms of the classification model. Instances accurately predicted from the target prediction model have small KLD values and are suitable for auxiliary training data of the target diagnosis model.

2.5.2. Maximum Mean Discrepancy

The MMD is the distance between the two domain distributions calculated under the RKHS. Note that the above-mentioned KMM procedure derives the instance weights by minimizing the MMD between domains. The MMD is also used to measure domain discrepancy in transfer learning and is formulated as follows:

$$MMD(X_S, X_T) = \parallel \frac{1}{n_S} \sum_{i=1}^{n_S} \psi(x_{S_i}) - \frac{1}{n_T} \sum_{j=1}^{n_T} \psi\left(x_{T_j}\right) \parallel_{\mathcal{H}}^{2}, \tag{7}$$

where ψ is the kernel mapping function, which converts original feature space into RKHS. We derive the MMD using the Gaussian radial basis function (RBF) kernel in the form of a nonlinear kernel. Reference paper has confirmed that this kernel function is suitable for the fault diagnosis, which is the goal of this research [39]. The MMD is used to measure the dissimilarity between domain instances because it can obtain the distance between two probability distributions in a nonparametric method without dimension constraints. Intuitively, because it is the squared distance of the mean of domain instances, it has a value close to 0 if the data are drawn from the same distribution. We derive the MMD of each source instance with the entire target domain data in order to use it as the weight. The MMD between i^{th} source domain instance and entire target domain instance is calculated as follows [40,41]:

$$MMD_i\left(x_{S_i}, X_T\right) = \parallel \psi\left(x_{S_i}\right) - \frac{1}{n_T} \sum_{j=1}^{n_T} \psi\left(x_{T_j}\right) \parallel_{\mathcal{H}}^{2} \tag{8}$$

Source domain instances which have a distribution similar to the target distribution have low MMD values. The source domain instance with low MMD is suitable for an auxiliary training sample for the target domain task.

2.5.3. Multi-Objective Instance Weighting

Two indicators, KLD and MMD, measure the dissimilarity between a source domain instance and target domain. The KLD measures how suitable the source domain is for the target domain task, and MMD more intuitively measures the distance between the two domain instances. Although both indicators measure domain discrepancy, they demonstrate inconsistent results. Since two indicators exhibit a low positive correlation, those indicators are combined as the multi-objective instance weight to play complementary roles in transfer learning.

Both KLD and MMD of certain source instance have positive values. These two indicators were standardized and converted to be used as weights $\mathcal{W}_i^{KLD}$ and $\mathcal{W}_i^{MMD}$, respectively. Instances with a high value in a specific indicator should have a low weight value calculated from that indicator. The two weights have a value between 0 and 1, and the larger the indicator value, the smaller the weight value. In this process, the instance weight derived from each indicator is set to be 0 if corresponding indicator of that instance is larger than a standard deviation from domain mean value. Afterward, the final weight of ith source domain instance $\mathcal{W}_i$ is derived by the weighted sum of the two weights:

$$\mathcal{W}_i = w_1 * \mathcal{W}_i^{KLD} + w_2 * \mathcal{W}_i^{MMD}, \tag{9}$$

where w_1 and w_2 are weights of the KLD and MMD-based weight. In this research, two weights are equal at 0.5, because the importance of KLD and MMD are even.

The multi-objective instance weights are used as training weights for the source domain in the training stage of the pre-trained model as shown in Equation (4). Through this process, source instances that are not useful for the target task are assigned very low training weights, so they are removed from the training stage, or their influence is significantly reduced. As a result, the effect of the dissimilarity between the two domains on model training is minimized so that transfer learning can proceed effectively.

2.6. Detailed Procedure of the Proposed Method

Step 1. The vibration signals are collected under various operating conditions for each domain using an accelerometer. The target domain has fewer labeled data than the source domain. There are labeled training and test data in the target domain. The labeled training data is used to fine-tune the pre-trained model to obtain the target diagnosis model, and the test data is not used for training but to evaluate the model performance.

Step 2. The acquired acceleration signals are converted into spectrogram images in the time-frequency domain through signal processing methods. Spectrogram images are used as input for a deep residual network. A pre-processed input image has the form of a [224, 224, 3] three-channel RGB.

Step 3. A deep residual network (ResNet50) is trained using source domain data as an initial pre-trained model. The last fully connected layer is initialized to be suitable for the task class, and training starts with random initial weights. The initial pre-trained model has feature extraction layers and parameters suitable for fault diagnosis in the source domain.

Step 4. The initial pre-trained model is transferred to target domain and fine-tuned using a small number of labeled target domain data. The initial deep transfer residual network is obtained. Detailed procedure of this step is presented in Figure 3.

Step 5. The MMD between the source domain instance and target domain are calculated. In addition, the KLD of the source domain instances are derived from the initial deep transfer residual network. Multi-objective source instance weights are derived from the calculated MMD and KLD.

Step 6. A final pre-trained model is trained using source domain data and multi-objective instance weights. By assigning instance weights, each instance in the source domain has different influence on the subsequent model training process. The learning rate was fixed at 0.001, and the training process was controlled so that no overfit occurs. After training a specific epoch, the structure of the trained model and all parameters are saved.

Step 7. The optimized pre-trained model is fine-tuned using target domain training data, which generates a final instance-based deep transfer learning network. Detailed procedure of this step is presented in Figure 3. To fine-tune the pre-trained model, we freeze beginning layers before residual blocks and train the parameters of the remainder layers including the fully-connected layer. During the fine-tuning procedure, we use Adam optimizer, and the batch size is set to be 32. The initial value of the learning rate is 0.001, and it is gradually decreased by multiplying by 0.5 to find the optimal parameters when the loss value over 50 epochs does not improve.

Step 8. The instance-based deep transfer learning network is applied to predict the label of the target test data, which is used to validate the performance of the designed model on the target fault diagnosis task.
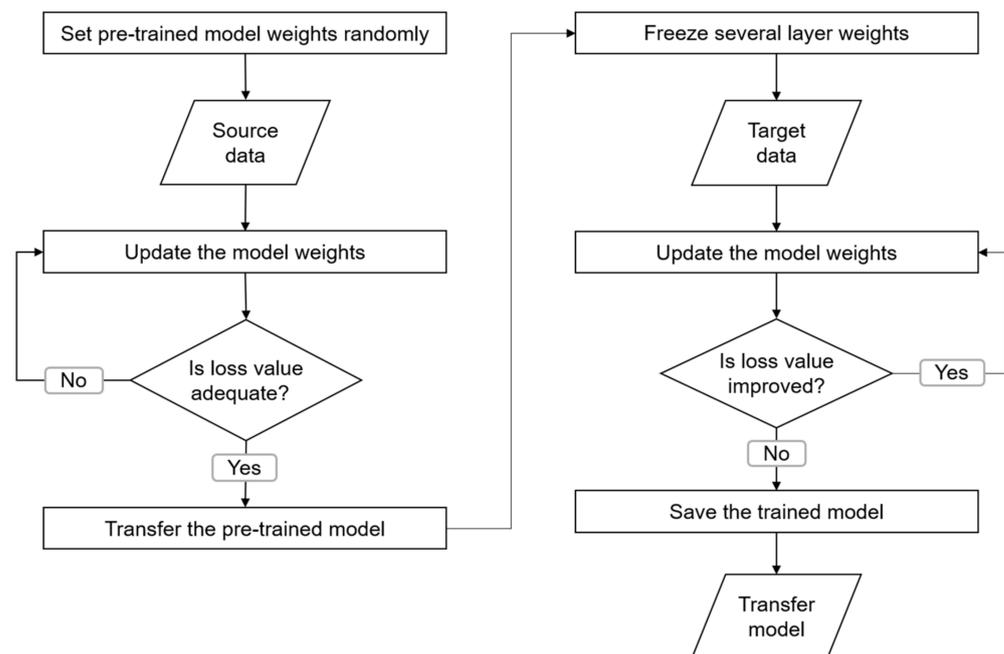

Figure 3. Flowchart of the model transfer and the fine-tuning procedure.

3. Case Study: Experimental Verification and Comparison Results

3.1. Experimental Setup Description

A case study was completed to verify the effectiveness of the proposed method for the fault diagnosis task and its applicability to actual industrial fields. Spot welding is one of the joining methods broadly used in various manufacturing processes and is performed in automotive body manufacturing processes. Welding faults are directly related to product quality degradation. However, it is difficult to identify a defect in spot welding without damaging the specimen. Several nondestructive tests exist, but it is impossible to do this for all products. Therefore, as an alternative, we performed a defect diagnosis of welding quality using only the accelerometer. The spot-welding process using the industrial robot imitated in the case study is straightforward to automate. Therefore, transfer learning can be particularly effective for diagnosing failures of different welding products in automated welding processes.

A testbed that mimics the spot-welding process using an industrial six-axis articulated robot used in the actual automotive body manufacturing process was installed. Machinery, including manufacturing robots, was investigated, and similar components were selected as much as possible. The HS-180 manufactured by Hyundai Robotics [42] was used as the manufacturing robot. The robot's payload is up to 180 kg, the maximum height is 2088 mm, and the maximum length is 3128 mm. A spot-welding servo gun made by Obara [43]

was used for spot welding. An experimental jig and other facilities were also designed as in the field.

As depicted in Figure 4, the accelerometer PCB 353B03 model was installed perpendicular to the electrode at the bottom of the spot-welding gun frame. Vibration signals were collected when welding was repeatedly performed. The experiment was conducted under a sampling frequency of 12.8 kHz and a sampling time of 35 s using the NI9232 DAQ.
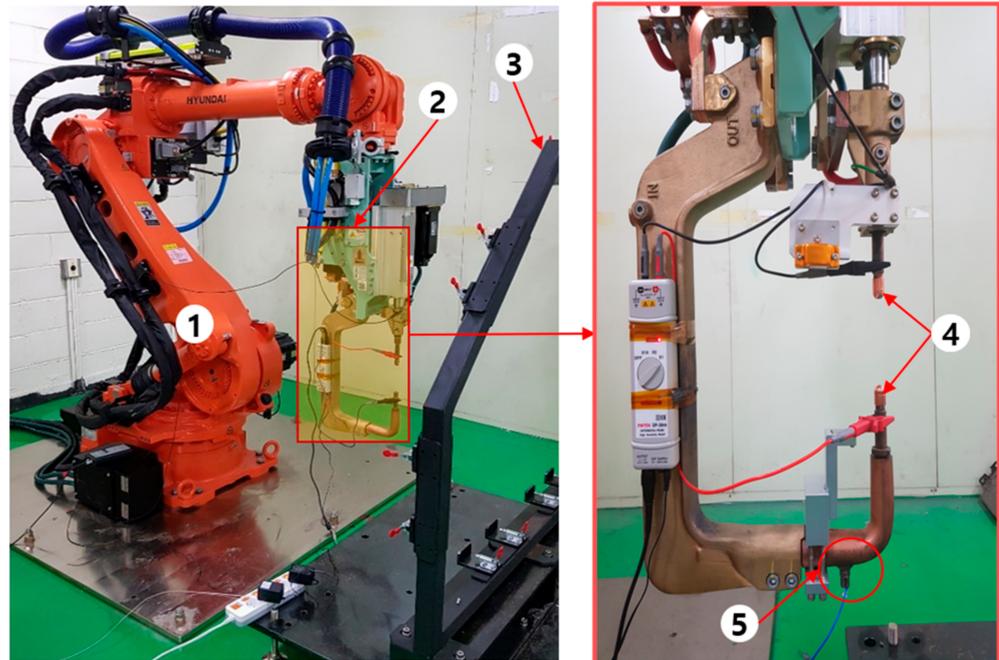

Figure 4. Experimental facility setup: (**1**) industrial robot, (**2**) spot-welding servo gun, (**3**) jig, (**4**) electrode, and (**5**) accelerometer.

In this case study, the occurrence of a small nugget was selected as the major failure state to observe. If the welding nugget is not stably formed, the joint does not proceed correctly. This is a typical welding quality fault, which causes a defect in the finished product. In terms of process variables, such a failure occurs when the weld current or the weld time is low or when the electrode pressing force is excessive compared to the weld current. In addition, small nugget often occurs when the dimension of the electrode tip is large due to wear. It can be caused by problems with various detailed components of the industrial robot, such as reducers, input gears, and motors. A cross-section test-based quality inspection was performed on the welded specimen to identify the failure. Cross-section test images are illustrated in Figure 5. Data were collected in three cases according to the type of welding specimen: mild steel, galvanized (GI) steel, and galvannealed (GA) steel. Primarily, the mechanical properties are different because the specimen materials are different for each welding specimen. The three types of specimens used in the case study have a common size of 300 mm in width, 100 mm in length, and 0.8 mm in thickness. The chemical composition of the mild steel specimen is C: 0.018%, Si: 0.002%, Mn: 0.201%, P: 0.009%, and S: 0.005%. The zinc coating mass of the GI steel specimen is 18.3 g/mm^2 and 17.8 g/mm^2, and the chemical composition is C: 0.02%, P: 0.013%, S: 0.011%, Mn: 0.13%, and Sol-Al: 0.03%. The GI steel specimen is zinc alloy steel with a zinc mass of 120 g/mm^2, and its chemical composition is C: 0.25%, P: 0.10%, S: 0.04%, and Mn: 1.35%. Therefore, the welding conditions, such as the electrode pressing force and weld current, were also set differently according to the properties of each specimen material. Table 2 lists the welding conditions for each specimen material in detail.
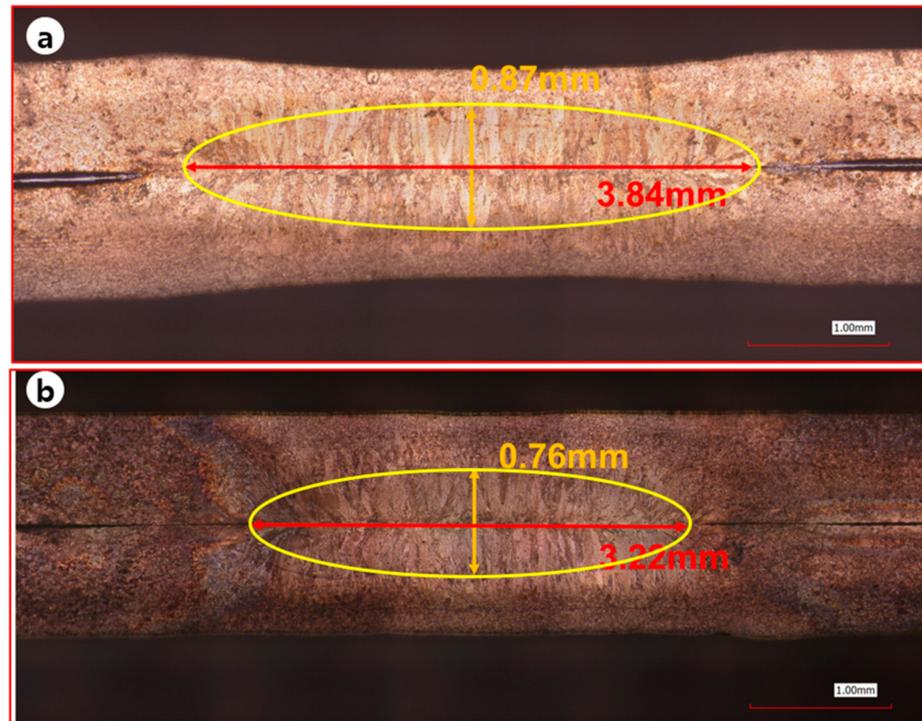
Figure 5. Cross-section test images of the welded specimen: (**a**) normal state, (**b**) small nugget.

Table 2. Weld conditions for each specimen.

Specimen Material	Electrode Force	Weld Current	Weld Time	Hold Time	Hold Time
Mild steel	250 kgF	5.0 kA	0.167 s	0.250 s	0.250 s
GA steel	385 kgF	5.5 kA	0.167 s	0.250 s	0.250 s
GI steel	360 kgF	6.0 kA	0.167 s	0.250 s	0.251 s

From the experiment, a total of 2160 data were collected. For each of the three specimens, the angle between the welding specimen and ground had two different cases, $0°$ or $45°$. Welding was performed 180 times for each normal and fault state in each angle case. In our comparison studies, the target domain data were defined as 360 data collected from GA steel with the $0°$ welding angle in order to make the target domain as similar as possible. The source domain data were defined as 1440 data collected from two different specimen materials, mild steel and GI steel. In the source domain, unlike with the target domain, all data from two materials were used regardless of the welding angle, because the source domain is not significantly affected by the data identity.

3.2. Comparison Studies

To evaluate the proposed method, the data from GA steel were set as the target domain, because it has the highest diagnosis difficulty. Among 360 target data collected from the GA steel welding under the same conditions, the randomly selected 180 data of 90 normal and 90 fault were used as target test data. The remaining 180 data were used for model training. For training the pre-trained model, 1440 data in the entire source domain were used. All result comparisons were derived by repeating the same model training process five times and calculating the arithmetic mean. In the following three comparison studies, we observed the model performances (i.e., prediction accuracy) depending on the employed (1) signal processing methods and learning algorithms, (2) the proposed method and the non-transfer learning method, (3) the instance optimization methods.

3.2.1. Comparison of Model Performance by Signal Processing Methods and the Learning Algorithms

The accuracy of the proposed method was compared with that of using only a small number of target domain data and using the source and target domain data simply together without transfer learning. In addition, the comparison of accuracy when using different signal processing methods is also presented in each case. We also used a fully-connected neural network with 100-50-25 neurons. A total of 12 handcrafted features extracted from each frequency band through wavelet decomposition were used to train the fully-connected neural network: minimum, maximum, mean, absolute mean, root mean square (RMS), peak-peak, peak to RMS, variance, kurtosis, skewness, entropy, and RMS error of estimation.

The results are listed in Table 3. In most cases, the diagnosis accuracy was improved by additionally using the source domain data compared to only using the target domain data. By comparing the diagnosis accuracy of the models trained using same ResNet50 structure and different training algorithms, it was verified that the accuracy can be increased through proposed method. On average, when the source and target data were treated as a single training dataset for model training, the average accuracy increases by 1.34% compared to when only using target data. However, when the proposed method is used, the average accuracy increases by 4.33% compared to when only using target data. From this result, it is possible to confirm the excellent effectiveness of the transfer learning and the fine-tuning strategy to use the knowledge of the source domain systematically. In addition, the performance of the deep learning-based method is quite superior to the traditional machine learning method using handcrafted features, and transfer learning can be a good application method to develop deep learning-based methods.

Table 3. Comparison results of each method.

Input Shape	Algorithm	Training Dataset		Classification Accuracy (%)
		Labeled Target Data	**Source Data**	
Handcrafted features	DNN (fully-connected)	O	X	73.80
		O	O	50.10
STFT image	ResNet	O	X	92.67
	ResNet	O	O	94.44
	Proposed method	O	O	98.33
WPD image	ResNet	O	X	94.00
	ResNet	O	O	95.00
	Proposed method	O	O	97.22
WPD image (spectral subtraction)	ResNet	O	X	92.56
	Proposed method	O	O	96.67

The highest accuracy of 98.33% was derived using the proposed method and STFT, which is 5.66% higher than 92.67% when only target data are used. The average accuracies of the proposed method using WPD and WPD with spectral subtraction are 97.22% and 96.67% respectively, which decrease 1.11% and 1.67% compared with the STFT. It is inferred that the time-frequency image created using STFT is the most suitable to be used in the proposed method in this case study.

3.2.2. Comparison between the Proposed Method and Non-Transfer Learning Method

The advantage of the proposed method is that high diagnosis accuracy can be obtained even when the labeled data of the target domain for diagnosis are few. To verify this advantage, we observed the change in accuracy through a scenario in which the number of target domain training data is reduced by 10. In addition, it was compared with the accuracy when transfer learning was not used. The comparison of these results confirmed how much the diagnosis accuracy improves by using the source domain and proposed

method. Table 4 presents the average value calculated by repeating the same process five times. Table 4 confirms that the accuracy improvement using the proposed method is more prominent when the target training data are few. We observed the average accuracy by dividing the entire range according to the number of training data into three ranges:

- S1 for 130 to 180 data ranges with an average accuracy of 90% or more;
- S2 for 70 to 120 ranges with an average accuracy of 80%;
- S3 for 20 to 60 ranges;
- S4 for the entire 20 to 180 range.

Table 4. Comparison results of the proposed and non-transfer learning methods.

Number of Target Training Data	Proposed Method		ResNet50 (Non-Transfer Learning)	
	Classification Accuracy (%)	Max.Accuracy Epoch	Classification Accuracy (%)	Max.Accuacry Epoch
180	98.33	148	92.67	497
170	98.89	212	93.78	382
160	96.44	96	91.78	253
150	97.78	138	91.11	200
140	97.33	237	91.78	218
130	96.22	173	88.67	403
120	96.00	277	89.22	232
110	95.56	211	90.00	237
100	95.44	230	86.22	218
90	91.89	169	87.11	181
80	91.11	282	84.67	193
70	93.11	121	82.33	309
60	88.56	191	79.44	319
50	85.11	295	78.11	235
40	83.22	186	74.22	353
30	75.56	62	61.67	431
20	75.00	90	61.00	484

In S1, with many training data, the average accuracy of the proposed method was 97.5%. The average accuracy increased by 5.87% compared with 91.63% when transfer learning was not performed. In S2, the average accuracy increased by 7.26%, from 86.59% to 93.85%. In S3, with few training data, the average accuracy was 81.49% and 70.89%, respectively, and an increase of 10.6% was observed. A detailed difference of accuracy between the proposed method and non-transfer learning method is presented in Figure 6. The accuracy difference usually increases as the number of target training data decreases. Especially in cases where the number of training data is 20, 30, and 40, it is possible to observe an increase in accuracy of 14%, 13.89%, and 9%, respectively. When the transfer learning was not used, the deep learning model was not adequately trained. It was confirmed that the use of the proposed method increases the diagnosis accuracy regardless of the number of training data. The efficiency also increases when the target domain data are few.
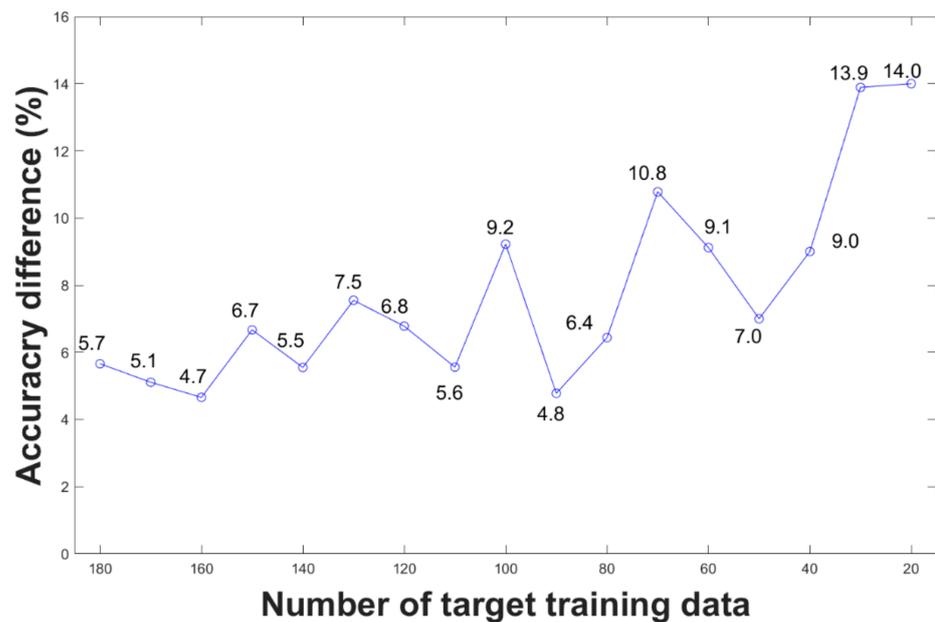
Figure 6. Accuracy difference between the proposed and non-transfer learning methods.

3.2.3. Comparison of Model Performance According to Domain Optimization Methods

The most critical part of instance-based transfer learning is the domain optimization process using instance weighting to minimize dissimilarity between domain instances. To demonstrate the effectiveness of the multi-objective instance weight of the proposed method, we compared the results according to the five domain optimization methods by performing transfer learning in the same process. In addition, each optimization method has different characteristics and merits, and the performance is greatly affected by the number of target domain data. Therefore, we compared the results through some scenarios in which a number from 20 to 180 of the target training data were randomly removed. The test set used to calculate the accuracy was fixed at the initial 180 to remove the randomness and observe only the effect of the number of target training data.

In Table 5, in all cases, the diagnosis accuracy of instance-based transfer learning with domain optimization was higher than that of the transfer learning without the domain optimization process. If domain optimization is not performed, it was confirmed that negative transfer occurs in some scenarios. The proposed method had the highest diagnosis accuracy in all cases. The other three optimization methods have similar results over the entire range but have different pros and cons.

These results are presented in Table 6. The highest diagnosis accuracy was obtained using the proposed method like other accuracy comparisons. The proposed method showed 97.5%, 93.85%, 81.49%, and 91.5% accuracies for S1, S2, S3, and S4, respectively.

The target dropout-based method demonstrated a low variance in the average accuracy with 94.95%, 89.39%, 78.84%, and 88.25% accuracy for S1, S2, S3, and S4, respectively. The KLD weight-based method has 95.32%, 89.02%, 75.29%, and 87.2% accuracy, respectively. The MMD weight-based method has 95.76%, 90.85%, 79.96%, and 89.38% accuracy, respectively, showing higher diagnostic accuracy than the other two domain optimization methods. A detailed comparison of accuracy over the entire range is presented in Figure 7. The results confirmed that excellent domain optimization effect can be obtained by complementarily using the advantages of the two indicators for weighting.

Table 5. Accuracy comparison results of different domain optimization methods.

Scenario	Number of Training Data	No Optimization	Target Dropout	KLD Instance Weights	MMD Instance Weights	Multi-Objective Instance Weights (Ours)
S1	180	92.11	96.33	97.11	97.33	98.33
	170	96.33	97.11	97.44	97.44	98.89
	160	89.33	92.78	95.22	92.56	96.44
	150	92.56	93	95.89	95.89	97.78
	140	94.11	94.67	95.67	96.56	97.33
	130	93.33	95.78	90.56	94.78	96.22
S2	120	88.89	92.11	93	91.78	96
	110	92.33	93.44	94.56	92.78	95.56
	100	90.44	92.67	90.67	93	95.44
	90	86.67	86.22	86.78	90.45	91.89
	80	88.22	85	83.89	87.11	91.11
	70	89.22	86.89	85.22	90	93.11
S3	60	83.22	85.33	81.33	86.44	88.56
	50	77.78	81.44	77.44	82.56	85.11
	40	78.78	81.89	75.44	81.89	83.22
	30	69	75.56	72.78	73.89	75.56
	20	65.56	70	69.44	75	75

Table 6. Average accuracy according to the transfer scenario.

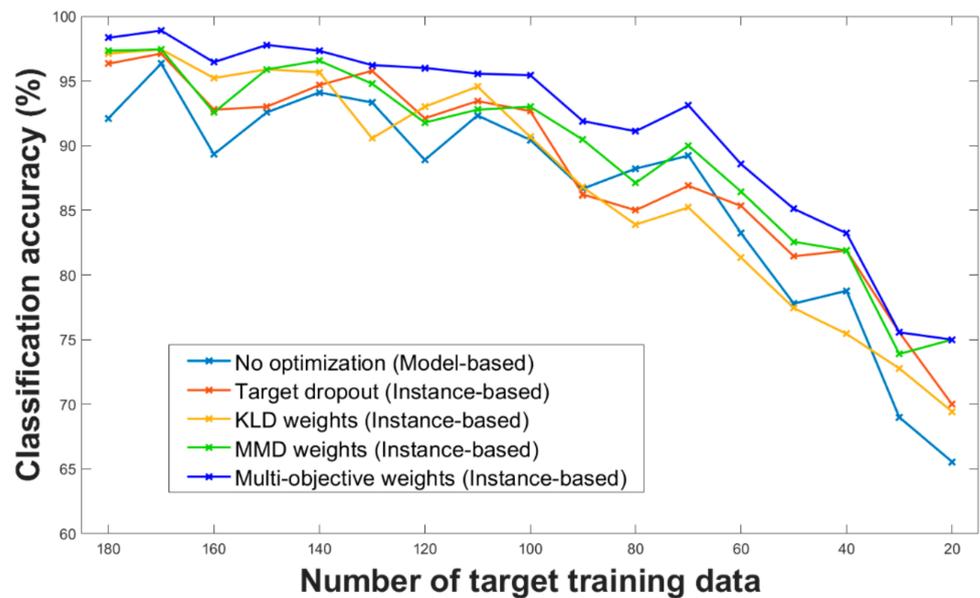| Transfer Scenario (Number of Training Data)	No Optimization	Target Dropout	KLD Instance Weights	MMD Instance Weights	Multi-Objective Instance Weights (Ours)
S1 (from 130 to 180)	92.96	94.95	95.32	95.76	97.5
S2 (from 70 to 120)	89.3	89.39	89.02	90.85	93.85
S3 (from 20 to 60)	74.87	78.84	75.29	79.96	81.49
S4 (from 20 to 180)	86.35	88.25	87.2	89.38	91.5

Figure 7. Comparison results between different domain optimization methods.

4. Discussion

(1)   The proposed method performs instance weighting using multi-objective instance weights for effective transfer learning to minimize the discrepancy between the target and source domain instances. Each source instance is assigned a weight that evaluates the relation to the target domain to curtail the negative transfer effect resulting from source instances with a large domain discrepancy. The MMD and KLD, which are indicators measuring dissimilarity, are used to obtain these weights. This method considers the discrepancy between the source and target domains and the discrepancy within the source domain. It includes the process of finding the optimal multi-objective instance weights and the final transfer learning process. The effect of knowledge transfer between the source and target domains is maximized through this method.

(2)   It was confirmed that the target task performance increased using a different but related source domain. However, some instances are not related to the source domain and have a large discrepancy. Therefore, performance improvement varies greatly depending on the technique of using the source domain. In this paper, we compared the performance of the domain optimization techniques used for this purpose. Our method of using two different indicators as instance weights causes a complementary effect of combining the advantages of the two indicators.

(3)   When performing intelligent fault diagnosis, a lack of labeled target data is a prevalent situation. However, data from similar work environments are usually available. When one has few labeled target data, the proposed method and transfer learning can be an excellent alternative. From the comparison results, robust performance was found even when few data exist. Through this, it was confirmed that the applicability in actual industrial fields is promising.

5. Conclusions

Intelligent fault diagnosis for practical machine equipment is one of the most important tasks in the industrial field. In this paper, a multi-objective instance weighting-based transfer learning network is proposed for successful fault diagnosis. The proposed method is based on the transfer learning and instance weighting strategy that complementarily utilize two indicators, KLD and MMD, to minimize discrepancy between the two domains used for the transfer learning model. Through this, the accuracy of target diagnosis is improved by using data with different conditions under a general situation where only a small number of data were collected from the target system condition. Following conclusions were drawn according to the case study and comparison results.

(1)   The proposed method outperforms the standard diagnosis models without transfer learning method.

(2)   It is verified that our multi-objective instance weighting strategy has higher performance than other optimization strategies used in transfer learning. The multi-objective instance weighting strategy can cope with deterioration of model performance due to dissimilarity between and within domains inherent in transfer learning process.

(3)   In particular, as the number of the training data in the target domain decreases, the improvement of the diagnosis accuracy and stability due to use of the proposed method increases.

(4)   The experimental setup of industrial spot welding that is actually carried out in automobile factories was used. Case study was conducted through the data collected from one accelerometer, with the realistic experimental conditions, such as difference in operating conditions and the number of data instances. It is confirmed that the proposed method has remarkable applicability for fault diagnosis in industrial sites.

Because of the above advantages, the proposed method can apply to actual fault diagnosis where the collected data is limited. This means that high diagnostic performance can be achieved with low data collection cost. Furthermore, if the weight parameters for

each indicator is not the equal value, but is given according to the characteristics of the domain data to be applied, it can be widely used for other applications. In addition, through the process of repeatedly updating the weight parameters like the ensemble method, there is a possibility that it can be used for comprehensive failure diagnosis in industrial sites where many units of similar facilities exist.

References

1. Lei, Y.; Yang, B.; Jiang, X.; Jia, F.; Li, N.; Nandi, A.K. Applications of machine learning to machine fault diagnosis: A review and roadmap. *Mech. Syst. Signal. Process.* **2020**, *138*, 106587. [CrossRef]
2. Kusiak, A. Smart manufacturing. *Int. J. Prod. Res.* **2018**, *56*, 508–517. [CrossRef]
3. Tao, F.; Qi, Q.; Liu, A.; Kusiak, A. Data-Driven smart manufacturing. *J. Manuf. Syst.* **2018**, *48*, 157–169. [CrossRef]
4. Pandya, D.H.; Upadhyay, S.H.; Harsha, S.P. Fault diagnosis of rolling element bearing with intrinsic mode function of acoustic emission data using APFKNN. *Expert Syst. Appl.* **2013**, *40*, 4137–4145. [CrossRef]
5. Li, C.; Sanchez, R.V.; Zurita, G.; Cerrada, M.; Cabrera, D.; Vasquez, R.E. Gearbox fault diagnosis based on deep random forest fusion of acoustic and vibratory signals. *Mech. Syst. Signal Process.* **2016**, *76–77*, 283–293. [CrossRef]
6. Glowacz, A.; Glowacz, W.; Glowacz, Z.; Kozik, J. Early fault diagnosis of bearing and stator faults of the single-phase induction motor using acoustic signals. *Measurement* **2018**, *113*, 1–9. [CrossRef]
7. Lopez-Perez, D.; Antonino-Daviu, J. Application of infrared thermography to failure detection in industrial induction motors: Case stories. *IEEE Trans. Ind. Appl.* **2017**, *53*, 1901–1908. [CrossRef]
8. Garcia-Ramirez, A.G.; Morales-Hernandez, L.A.; Osornio-Rios, R.A.; Benitez-Rangel, J.P.; Garcia-Perez, A.; Romero-Troncoso, R.D. Fault detection in induction motors and the impact on the kinematic chain through thermographic analysis. *Electrt. Power Syst. Res.* **2014**, *114*, 1–9. [CrossRef]
9. Glowacz, A. Fault diagnosis of electric impact drills using thermal imaging. *Measurement* **2021**, *171*, 108815. [CrossRef]
10. Naha, A.; Samanta, A.K.; Routray, A.K.; Deb, A.K. Low complexity motor current signature analysis using sub-nyquist strategy with reduced data length. *IEEE Trans. Instrum. Meas.* **2017**, *66*, 3249–3259. [CrossRef]
11. Yang, T.; Pen, H.; Wang, Z.; Chang, C. Feature knowledge based fault detection of induction motors through the analysis of stator current data. *IEEE Trans. Instrum. Meas.* **2016**, *65*, 549–558. [CrossRef]
12. Urbikain, G.; Campa, F.J.; Zulaika, J.J.; Lopez de Lacalle, L.N.; Alonso, M.A.; Collado, V. Preventing chatter vibrations in heavy-duty turning operations in large horizontal lathes. *J. Sound Vib.* **2015**, *340*, 317–330. [CrossRef]
13. Oleaga, I.; Pardo, C.; Zulaika, J.J.; Bustillo, A. A machine-learning based solution for chatter prediction in heavy-duty milling machines. *Measurement* **2018**, *128*, 34–44. [CrossRef]
14. Rai, A.; Upadhyay, S.H. A review on signal processing techniques utilized in the fault diagnosis of rolling element bearings. *Tribol. Int.* **2016**, *48*, 289–306. [CrossRef]
15. Goyal, D.; Vanraj; Pabla, B.S.; Dhami, S.S. Condition monitoring parameters for fault diagnosis of fixed axis gearbox: A review. *Arch. Comput. Methods Eng.* **2017**, *24*, 543–556. [CrossRef]
16. Glowacz, A.; Glowacz, W. Vibration-Based fault diagnosis of commutator motor. *Shock Vib.* **2018**, *2018*, 7460419. [CrossRef]
17. Palacios, R.H.C.; Goedtel, A.; Godoy, W.F.; Fabri, J.A. Fault identification in the stator winding of induction motors using PCA with artificial neural networks. *J. Control Autom. Electr. Syst.* **2016**, *27*, 406–418. [CrossRef]
18. Kang, M.; Kim, J.; Kim, J.M.; Tan, A.C.C.; Kim, E.Y.; Choi, B.K. Reliable fault diagnosis for low-speed bearings using individually trained support vector machines with kernel discriminative feature analysis. *IEEE Trans. Power Electron.* **2015**, *30*, 2786–2797. [CrossRef]

19. Han, S.; Choi, H.J.; Choi, S.K.; Oh, J.S. Fault diagnosis of planetary gear carrier packs: A class imbalance and multiclass classification problem. *Int. J. Precis. Eng.* **2019**, *20*, 167–179. [CrossRef]

20. Ma, S.; Chu, F.; Han, Q. Deep residual learning with demodulated time-frequency features for fault diagnosis of planetary gearbox under nonstationary running conditions. *Mech. Syst. Signal. Process.* **2019**, *127*, 190–201. [CrossRef]

21. Zhang, W.; Li, X.; Ding, Q. Deep residual learning-based fault diagnosis method for rotating machinery. *ISA Trans.* **2019**, *95*, 295–305. [CrossRef] [PubMed]

22. Shao, S.; McAleer, S.; Yan, R.; Baldi, P. Highly accurate machine fault diagnosis using deep transfer learning. *IEEE Trans. Industr. Inform.* **2019**, *15*, 2446–2455. [CrossRef]

23. Yang, F.; Zhang, W.; Tao, L.; Ma, J. Transfer learning strategies for deep learning-based PHM algorithms. *Appl. Sci.* **2020**, *10*, 2361. [CrossRef]

24. Cao, P.; Zhang, S.; Tang, J. Preprocessing-Free gear fault diagnosis using small datasets with deep convolutional neural network-based transfer learning. *IEEE Access.* **2018**, *6*, 26241–26253. [CrossRef]

25. Wang, Z.; Dai, Z.; Póczos, B.; Carbonell, J. Characterizing and avoiding negative transfer. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 11293–11302.

26. Wang, T.; Huan, J.; Zhu, M. Instance-Based deep transfer learning. In Proceedings of the 2019 IEEE Winter Conference on Applications of Computer Vision (WACV), Waikoloa Village, HI, USA, 7–11 January 2019; pp. 367–375.

27. Zhang, L.; Guo, L.; Gao, H.; Dong, D.; Fu, G.; Hong, X. Instance-Based ensemble deep transfer learning network: A new intelligent degradation recognition method and its application on ball screw. *Mech. Syst. Signal. Process.* **2020**, *140*, 106681. [CrossRef]

28. Bustillo, A.; Urbikain, G.; Perez, J.M.; Pereira, O.M.; Lopez de Lacalle, L.N. Smart optimization of a friction-drilling process based on boosting ensembles. *J. Manuf. Syst.* **2018**, *48*, 108–121. [CrossRef]

29. Dai, W.; Yang, Q.; Xue, G.-R.; Yu, Y. Boosting for transfer learning. In Proceedings of the 24th International Conference on Machine Learning, New York, NY, USA, 20–24 June 2007; pp. 193–200.

30. Yao, Y.; Doretto, G. Boosting for transfer learning with multiple sources. In Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Francisco, CA, USA, 13–18 June 2010; pp. 1855–1862.

31. Kullback, S.; Leibler, R.A. On information and sufficiency. *Ann. Math. Stat.* **1951**, *22*, 79–86. [CrossRef]

32. Borgwardt, K.M.; Gretton, A.; Rasch, M.J.; Kriegel, H.P.; Scholkopf, B.; Smola, A.J. Integrating structured biological data by Kernel Maximum Mean Discrepancy. *Bioinformatics* **2006**, *22*, e49–e57. [CrossRef]

33. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.

34. Huh, J.; Pham Van, H.; Han, S.; Choi, H.J.; Choi, S.K. A data-driven approach for the diagnosis of mechanical systems using trained subtracted signal spectrograms. *Sensors* **2019**, *19*, 1055. [CrossRef]

35. Pan, S.J.; Yang, Q. A survey on transfer learning. *IEEE Trans. Knowl. Data. Eng.* **2009**, *22*, 1345–1359. [CrossRef]

36. Zhuang, F.; Qi, Z.; Duan, K.; Xi, D.; Zhu, Y.; Zhu, H.; Xiong, H.; He, Q. A comprehensive survey on transfer learning. *Proc. IEEE* **2021**, *109*, 43–76. [CrossRef]

37. Huang, J.; Gretton, A.; Borgwardt, K.; Schölkopf, B.; Smola, A. Correcting sample selection bias by unlabeled data. *Adv. Neural. Inf. Process. Syst.* **2006**, *19*, 601–608.

38. Sugiyama, M.; Suzuki, T.; Nakajima, S.; Kashima, H.; von Bünau, P.; Kawanabe, M. Direct importance estimation for covariate shift adaptation. *Ann. Inst. Stat.* **2008**, *60*, 699–746. [CrossRef]

39. Wang, Q.; Michau, G.; Fink, O. Domain adaptive transfer learning for fault diagnosis. In Proceedings of the 2019 Prognostics and System Health Management Conference (PHM-Paris), Paris, France, 2–5 May 2019; pp. 279–285.

40. Pan, S.J.; Tsang, I.W.; Kwok, J.T.; Yang, Q. Domain adaptation via transfer component analysis. *IEEE Trans. Neural. Netw.* **2011**, *22*, 199–210. [CrossRef] [PubMed]

41. Gretton, A.; Borgwardt, K.M.; Rasch, M.J.; Schölkopf, B.; Smola, A. A kernel two-sample test. *J. Mach. Learn. Res.* **2012**, *13*, 723–773.

42. Hyundai Robotics. Available online: https://www.hyundai-robotics.com/english/index.html (accessed on 9 July 2020).

43. Obara. Available online: http://www.obara.co.jp/en/ (accessed on 12 July 2020).