Source-Free Multi-Domain Adaptation with Fuzzy Rule-based Deep Neural Networks

Keqiuyin Li, Jie Lu*, *Fellow, IEEE,* Hua Zuo, *Member, IEEE,* and Guangquan Zhang

***Abstract*—Unsupervised domain adaptation deals with a task from an unlabeled target domain by leveraging the knowledge gained from labeled source domain(s). Fuzzy system is adopted in domain adaptation to better tackle the uncertainty caused by information scarcity in the transfer. Most existing fuzzy and non-fuzzy domain adaptation methods depend on data-level distribution matching to eliminate domain shift. However, data sharing can trigger privacy concerns. This situation results in the unavailability of source data, wherein most domain adaptation methods cannot be applied. Source-free domain adaptation is then proposed to handle this problem. But existing source-free domain adaptation methods rarely deal with any soft information component due to data imprecision. Besides, fewer methods handle multiple source domains which provide richer transfer information. Thus, in this paper, we propose source-free multi-domain adaptation with fuzzy rule-based deep neural networks (SF-FDN), which takes advantage of a fuzzy system to handle data uncertainty in domain adaptation without source data. To learn source private models with high generality, which is important to collect low noisy pseudo target labels, auxiliary tasks are designed by jointly training source models from multiple domains which share source parameters and fuzzy rules while protecting source data. To transfer fuzzy rules and fit source private parameters to the target domain, self-supervised learning and anchor-based alignment are built to force target data to source feature spaces. Experiments on real-world datasets under both homogeneous and heterogeneous label space scenarios are carried out to validate the proposed method. The results indicate the superiority of the proposed fuzzy rule-based source-free multi-domain adaptation method.**

***Index Terms*—Domain adaptation, transfer learning, classification, machine learning, fuzzy rules.**

I. Introduction

UNSUPERVISED domain adaptation shows the advantage of leveraging knowledge from a label-rich source domain to an unlabeled target domain, given its ability to handle training (source) and testing (target) data from different distributions [1]. To overcome data shift between source and target domains, reducing the distribution discrepancy is a widely explored solution [2]. Another popular method to reduce data bias is adversarial learning, which is introduced through the application of deep neural networks in domain adaptation [3]. Adversarial learning matches source and target distributions by building a two-player game, including a domain discriminator that distinguishes source and target samples, and a generator extracting invariant feature or generating fake data which confuses the domain discriminator.

This work was supported by the Australian Research Council under grant FL190100149.

*Corresponding author. The authors are with the Australian Artificial Intelligence Institute, University of Technology Sydney, Sydney, NSW 2007, Australia (e-mail: {keqiuyin.li; jie.lu; hua.zuo; guangquan.zhang}@uts.edu.au).

Adapting source and target data provides viability of knowledge transfer across domains. To transfer knowledge from the source domain(s) to the target domain, the problem falls on what and how to transfer. Instance-based [4], feature-based [5], parameter-based [6] and relationship-based [7] methods are developed to deal with what and how issues. Feature-based domain adaptation is currently widely investigated. Aligned with the feature dimension, it can be divided into homogeneous [8] and heterogeneous [9] domain adaptation. Corresponding to the label space, it can be divided into closed-set [10], open-set [11], partial [12] and universal [13] domain adaptation. The main problem is how to collect transfer knowledge to over label gap. Enriching the transfer knowledge means exploiting multi-view feature or multi-source domain adaptation [14]. Multi-source domain adaptation faces several challenges such as the increased size of combined source datasets and the lack of mixture parameters of target distribution. Dynamic classifier alignment proposes an automatic merging method over the sample to learn the mixture parameters of source classifiers, and investigates the contribution of multi-view features simultaneously by an importance learning strategy that can handle both homogeneous and heterogeneous features [15].

Most previous domain adaptation methods rely on their access to source data, a factor that might trigger privacy concerns. To solve this problem, source-free domain adaptation is proposed [16]. Two approaches commonly employed to transfer knowledge across domains without source data, include data generation [17] and model adaptation [18]. Data generation methods are developed based on generative adversarial network [19]. The central idea is constructing source-like or target-like images using a generator, then reducing the difference between the generated and real images by a discriminator to adapt the source model to the target domain. A recent study- source data free domain adaptation- learns joint distribution of source domain by producing source-style proxy samples from the pre-trained source classifier, the learned distribution is then used to extract invariant features of the unlabeled target domain to fine-tune the pre-trained model [20]. Model adaptation methods depend mainly on a pseudo-labeling strategy [21]. Unsupervised learning techniques, such as clustering, are employed to provide pseudo target labels, the target model is then trained based on the source model in a self-supervised way. A previous multi-source data free domain adaptation method adopts weighted information maximization and weighted pseudo-labeling to combine source predictions automatically and collect target labels, the target model is trained by jointly optimizing the source feature encoders with corresponding weights [22].

However, neither the aforementioned domain adaptation methods with source data nor the methods without source data consider the soft information caused by uncertain data during transfer. To solve this problem, fuzzy domain adaptation attracts attention in light of its advantages in building soft information to handle data uncertainty [23]. A theory-based study-learning from imprecise observations- investigates multi-class classification with fuzzy observations and protects data privacy by transferring original data into concepts, thereby considering both data uncertainty and security [24]. It creates fuzzy vectors from real observations and provides an estimation error bound learned from fuzzy random variables. Fuzzy multi-source transfer learning focuses on selecting and merging fuzzy rules from multiple domains to generate target rules under both homogeneous and heterogeneous domain adaptation scenarios [25]. Interactive transfer learning distills useless source information by a knowledge filter, and designs a self-balancing mechanism to learn the scene difference and inherent uncertainty, which are used match source and target domains by reducing unbalanced diversity [26]. Fuzzy multi-output transfer learning considers the degree to which the qualities of multiple outputs being either shared or unique of multiple outputs are reflected by source fuzzy rules constructed from both output-input dependencies and inter-output correlations [27]. Fuzzy transfer representation learning transforms data into fuzzy space by TSK fuzzy rules, which adopts linear discriminant analysis and principal component analysis to protect the discriminant information and geometric properties of data [28].

Existing fuzzy domain adaptation methods focus on transferring the invariant knowledge extracted from data, but fuzzy domain adaptation without source data remains unsolved. In addition, few domain adaptation studies extend fuzzy model to deep structure. Many deep fuzzy methods can only handle tasks where the training and testing data follow the same distribution [29], [30]. In transfer learning, source (training) and target (testing) domains have different distributions, where existing fuzzy deep methods can fail to apply. Most source-free domain adaptation methods rarely take the inherent soft information into account, especially in deep neural networks, where the data is trained over the batch by extracting region information using convolution kernels. The extracted regions and samples belonging to the same category but from different views contain multiple information levels, while samples from different categories can contain similar regions, which contribute differently to its category and the whole classifier. In this situation, dividing samples into multiple groups according to their information levels might benefit the classification. The fuzzy model has the advantage of describing the degrees of multi-level information belonging to multiple categories. Hence, in this paper, we propose source-free multi-domain adaptation with fuzzy rule-based deep neural networks (SF-FDN) to extract soft information from precise data, which introduces fuzzy C-means clustering and Takagi-Sugeno fuzzy rules to source-free domain adaptation.

The proposed method improves the generality of a source private model on multiple domains by establishing auxiliary tasks, which derive benefit from similar tasks and preserve the data privacy in source domains simultaneously. When transferring source rules and parameters to a target domain, to guarantee the accuracy of pseudo labels, a target sample selection strategy is adopted to collect pseudo labels with low noise. We use the pseudo labels to supervise the training of the target model on the label-level, and develop anchor-based alignment to reduce data bias between domains on the distribution-level. This approach allows us to extract both invariant and specific information of target domains to parameterize the target model. Our contributions are summarized as follows:

- We propose source-free multi-domain adaptation with fuzzy rule-based deep neural networks. To the best of our knowledge, this is the first work adopting fuzzy rules to deal with source-free transfer learning. The proposed method deals with soft information to enrich transferable knowledge among both classes and domains, which most non-fuzzy methods rarely consider. It develops fuzzy C-means clustering in a deep structure to construct fuzzy rules and introduces the Takagi-Sugeno model to solve domain adaptation without source data. Based on experiments, the fuzzy source-free domain adaptation method is superior to non-fuzzy methods by transferring knowledge on multiple information levels;
- We develop an auxiliary learning mechanism to enhance the multi-domain performance of the private source models. Few existing source-free methods handle multiple source domains. The proposed method takes advantage of category information from other source domains by jointly training source parameters. Compared with existing source-free multi-domain adaptation methods which train source models independently, by doing this, multiple source domains can now share invariant knowledge without sharing data;
- We generate source anchors from source fuzzy rules to collect highly representative class features, which are employed to define an anchor-based alignment strategy to fit the pre-trained source model to the target domain while protecting the source data. By reducing the distance between source and target anchors which highly represent class information, the target feature extractor is forced to transform target data into the latent feature space which is closer to source distribution. Compared with existing source data generation methods, the proposed source anchors based on fuzzy rules can collect more usable knowledge on multiple information levels;
- We build a selection strategy in assistance with fuzzy outputs and nearest clustering to collect strong target samples to calculate clustering centers which we employ to predict pseudo labels with high confidence. In comparison to existing methods computing clustering centers using all pseudo target labels, the proposed strategy can reduce the label noise which is known to result in negative transfer.

The remainder of this paper is organized as follows. Section II briefly describes the related work on domain adaptation with and without source data. Section III provides the details of the proposed source-free multi-domain adaptation method, which generates and transfers fuzzy rules across domains by anchor-based alignment and self-supervised learning. The experiment results on real-world visual datasets and subsequent analysis are presented in Section IV. Section V summarizes the research and proposes potential directions for future study.

II. Related Work

This section introduces previous studies on fuzzy and non-fuzzy unsupervised domain adaptation with and without source data. Popular techniques in non-fuzzy and fuzzy domain adaptation are reviewed.

A. Domain Adaptation

A popular technique for source-free domain adaptation is self-supervised learning based on pseudo labels. Commonly used pseudo-labeling techniques includes clustering and K-nearest neighbors. Domain consensus clustering constructs discriminative clusters by exploring domain consensus knowledge, that identifies target clusters optimally by leveraging cycle-consistent matching. Both semantic-level and sample-level consensus knowledge are extracted to estimate the consensus score that permits us to predict the degrees of samples belonging to clusters, by which the number of target clusters can be defined [31]. To perform source model on a target domain without source data, robust adaptation is proposed to preserve the robustness and performance of the pre-trained source model, where standard models are employed to provide pseudo target labels with less noise, while robust models are used to generate adversarial target samples which expect to enhance the domain alignment [32]. Casting a BAIT deals with both online and offline source-free domain adaptation by building a two-step optimization policy, where an extra classifier that identifies certain and uncertain features is introduced to find misalignment samples, while the multi-class source classifier is used to provide class anchors [33].

B. Fuzzy Domain Adaptation

Fuzzy domain adaptation bridges source and target domains by constructing fuzzy rules or fuzzy relations, both single source and multi-source domain adaptation with homogeneous and heterogeneous feature spaces are explored. Transfer learning based on fuzzy residual adopts a residual function to generate target rules from the learned source hypothesis, where TSK fuzzy rules are used to describe the marginal distribution of data, which treat the target model as a combination of source tasks, making it possible to update the target model in a model-agnostic way [34]. Fuzzy-relation neural networks are exploited in multi-source heterogeneous domain adaptation, which extracts shared fuzzy information from multiple feature spaces and constructs a new latent feature space to minimize the discrepancy of multiple sources and the target branches produced by shared-fuzzy-equivalence-relations [35]. Fuzzy rule-based deep neural networks for multi-source domain adaptation introduce the Takagi–Sugeno fuzzy model to deep networks by grouping source samples into fuzzy sets according to different information levels, domain and cluster discriminators are designed to measure the membership of a sample being clustered to multiple classes [36].

Most existing fuzzy domain adaptation methods assume source data is available, but transfer without source data remains unsolved. In this paper, we design a new fuzzy rule-based deep neural network to tackle data-free domain adaptation with multiple source domains. In the proposed method, source private models under fuzzy rules of each domain are learned by jointly training other source models using an auxiliary learning strategy, where source parameters are shared while source data is preserved. Furthermore, anchor-based alignment is designed to match target samples to the source anchors according to the agreements of clustering a target sample to a source category. Since source data is unavailable, to fit source models better, self-supervised learning based on pseudo labels is employed to train the target feature extractor which transforms target data into a latent feature space close to the source space. To reduce the influence of noisy target labels, a sample selection strategy is designed by combining the predictions of the source model and deep clustering to identify strong target samples, which are then used to update clustering centers that renew pseudo labels with a high level of certainty.

III. The Proposed Source-Free Multi-Domain Adaptation Method

We focus on multi-source data-free domain adaptation. The proposed method is illustrated in Fig. 1. The top figure displays the process of source private model training with auxiliary tasks, while the bottom figure indicates the model adaptation based on self-supervised training. As shown in Fig. 1(a), for each source domain, the original data is first transformed into a latent feature space by the feature extractor. To leverage soft information, we adopt fuzzy C-means clustering to calculate prototypes of each source domain and memberships of samples to define fuzzy rules and predict the final outputs. Source anchors are generated based on the fuzzy model to describe class information distinctly and preserve data privacy for other users. The error between fuzzy outputs and the ground-truth labels are employed to parameterize the training. To provide better generality of source private models in predicting target task, source parameters and fuzzy rules are shared among domains as auxiliary models for each other using a joint training method. In Fig. 1(b), given the pre-trained source models based on fuzzy rules, pre-learned source rules are frozen to match target data to source distribution where source models can be transferred, while feature extractors are re-trained. Anchor-based alignment is designed to force the target data to source feature space by extracting invariant information. Besides, to extract specific information from target domain, self-supervision is constructed to parameterize the re-training. The success of self-supervision relies on the high quality of pseudo labels. To guarantee these pseudo labels provided by deep clustering come with low noise, a sample selection strategy based on fuzzy outputs is built to select target labels confidently and generate reliable clustering centers. The crossentropy loss between pseudo labels and the fuzzy outputs is employed to fine-tune the feature extractors.

Table I describes the symbols in this paper.

TABLE I
SYMBOL DESCRIPTIONS.

Notation	Description
$\mathcal{D}_{s_k}$, $\mathcal{D}_t$	the private source/target domain
n_{s_k}, n_t	number of source/target samples
$\boldsymbol{x}_{s_k}$, $\boldsymbol{x}_t$	source/target sample
$\boldsymbol{y}_{s_k}$	source label of $\boldsymbol{x}_{s_k}$
ϕ	pre-trained deep-structured backbone
ϕ_k	the kth source private feature extractor
$\boldsymbol{v}^l_{s_k}$, $\boldsymbol{v}^l_{t_k}$	clustering prototype from the source/target domain
$u^l_{s_k}$, $u^l_{t_k}$	membership of the source/target sample belonging to lth fuzzy set
$\boldsymbol{f}^c_{s_k}$, $\boldsymbol{f}^c_{t_k}$	class anchor from the source/target domain
$\boldsymbol{w}_c$	deep clustering center from target domain
$\boldsymbol{r}$	probability vector estimated by classifier

A. Source Private Model Training

In this paper, we employ Takagi-Sugeno fuzzy rules to build the source model. Given source domains $\{\mathcal{D}_{s_k}\}_{k=1}^K$, for input data $\boldsymbol{x}_{s_k} \in \mathbb{R}^s$ and the corresponding output $\boldsymbol{y}_{s_k} \in \mathbb{R}^c$ in the kth source domain, a rule can be described as:

$$\begin{aligned}&\text{if } \boldsymbol{x}_{s_k} \ is \ A_{kl}(\phi_k(\phi(\boldsymbol{x}_{s_k}))),\\ &\text{then } \boldsymbol{y}_{s_k} \ is \ P_{kl}(\phi_k(\phi(\boldsymbol{x}_{s_k}))),\\ &l = 1, 2, \cdots, L_k.\end{aligned}$$

ϕ_k is the private feature extractor in the kth source domain, while ϕ is a pre-trained deep-structured backbone on a very large dataset. Parameters of ϕ are shared among all domains. Feature extractors transform original data to feature space $\mathbb{R}^d$. A_{kl} represents the fuzzy condition of the lth rule, P_{kl} is a function transforming data from $\mathbb{R}^d$ to $\mathbb{R}^c$. L_k represents the number of rules in the kth source domain.

The final prediction of the Takagi-Sugeno fuzzy model in each source domain is the linear combining of the outputs of all rules, which is:

$$\boldsymbol{y}_{s_k} = \sum_{l=1}^{L_k} u^l_{s_k} \cdot P_{kl}(\phi_k(\phi(\boldsymbol{x}_{s_k}))),$$

$u^l_{s_k}$ is the membership of data $\boldsymbol{x}_{s_k}$ belonging to the lth fuzzy set.

There are three problems to be solved to build the source model: first, how to define fuzzy rule number L_k; second, how to learn function P_{kl}; third, how to measure the membership $u^l_{s_k}$ to calculate the final prediction.

To solve the first problem, we design a class grouping strategy based on the similarities among each pair of classes. Here, the correlation coefficient is employed to measure the similarity between every two classes. Denote any class pair as $(\boldsymbol{x}_{c_i}, \boldsymbol{x}_{c_j})$, the correlation coefficient between each two classes is calculated as:

$$\rho_{ij} = \frac{\mathbb{E}(\boldsymbol{x}_{c_i}\boldsymbol{x}_{c_j}) - \mathbb{E}(\boldsymbol{x}_{c_i})\mathbb{E}(\boldsymbol{x}_{c_j})}{\sqrt{\mathbb{E}(\boldsymbol{x}^2_{c_i}) - (\mathbb{E}(\boldsymbol{x}_{c_i}))^2}\sqrt{\mathbb{E}(\boldsymbol{x}^2_{c_j}) - (\mathbb{E}(\boldsymbol{x}_{c_j}))^2}}, \quad (1)$$

where

$$\boldsymbol{x}_c = \frac{\sum_{i=1}^{n^c_{s_k}} \mathbb{1}_{\boldsymbol{y}^i_{s_k}=c} \cdot \phi(\boldsymbol{x}^i_{s_k})}{\sum_{i=1}^{n^c_{s_k}} \mathbb{1}_{\boldsymbol{y}^i_{s_k}=c}},$$

$n^c_{s_k}$ denotes the number of source samples in the cth class. When $\rho_{ij} > a_\rho$, where a_ρ is a threshold, we think classes c_i and c_j are similar, and they can share the same rule. Here, we use $\boldsymbol{Y}_l$ to denote the label set containing similar classes.

To solve the second problem, we apply a structural risk minimization principle [37] to learn the function. In the classification task, P_{kl} is a classifier parameterized by minimizing the assumption between the prediction and the ground-truth source labels, which can be expressed as:

$$P_{kl} = \underset{\substack{P_{s_{kl}} \\ (\boldsymbol{x}_{s_k}, \boldsymbol{y}_{s_k}) \in \mathcal{D}_{s_k}}}{\arg\min} \ \mathcal{L}(P_{kl}(\phi_k(\phi(\boldsymbol{x}_{s_k}))), \boldsymbol{y}_{s_k}), \quad (2)$$

where

$$\mathcal{L} = -\frac{1}{n_{s_k}} \sum_{i=1}^{n_{s_k}} \boldsymbol{y}^i_{s_k} \log(P_{kl}(\phi_k(\phi(\boldsymbol{x}^i_{s_k})))).$$

To improve the training speed and prevent source model parameters from overfitting which may fail the transfer, a label smoothing strategy is adopted to transform hard labels to soft labels [16], [38], which is:

$$\tilde{\boldsymbol{y}}_{s_k} = (1 - \mu)\boldsymbol{y}_{s_k} + \mu / C,$$

where μ is the smoothing parameter, and C is the number of source classes. Classifier P_{kl} in equation (2) with smooth label is:

$$P_{kl} = \underset{\substack{P_{kl} \\ (\boldsymbol{x}_{s_k}, \tilde{\boldsymbol{y}}_{s_k}) \in \mathcal{D}_{s_k}}}{\arg\min} \ \mathcal{L}(P_{kl}(\phi_k(\phi(\boldsymbol{x}_{s_k}))), \tilde{\boldsymbol{y}}_{s_k}), \quad (3)$$

where $\mathcal{L}$ is re-written as:

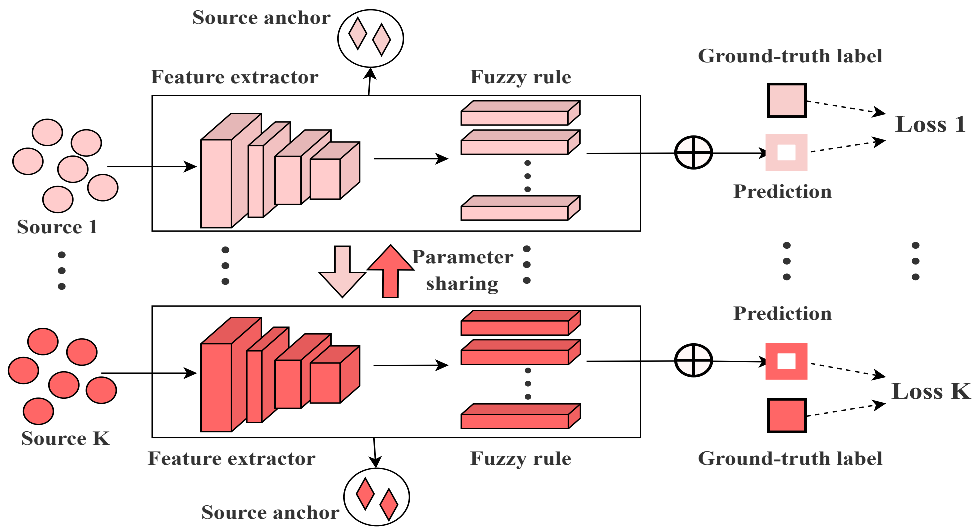
$$\mathcal{L} = -\frac{1}{n_{s_k}} \sum_{i=1}^{n_{s_k}} \tilde{\boldsymbol{y}}^i_{s_k} \log(P_{kl}(\phi_k(\phi(\boldsymbol{x}^i_{s_k})))).$$

To solve the third problem, fuzzy C-mean clustering is a popular technique for calculating the memberships. Setting the cluster number as the rule number defined using equation (1), it calculates a prototype in every cluster to estimate data membership. Generally, the cluster prototypes and data memberships are updated alternately by fixing the other. In this work, cluster prototypes are initialized as the mean values of samples from the same cluster grouped according to their labels, expressed as:

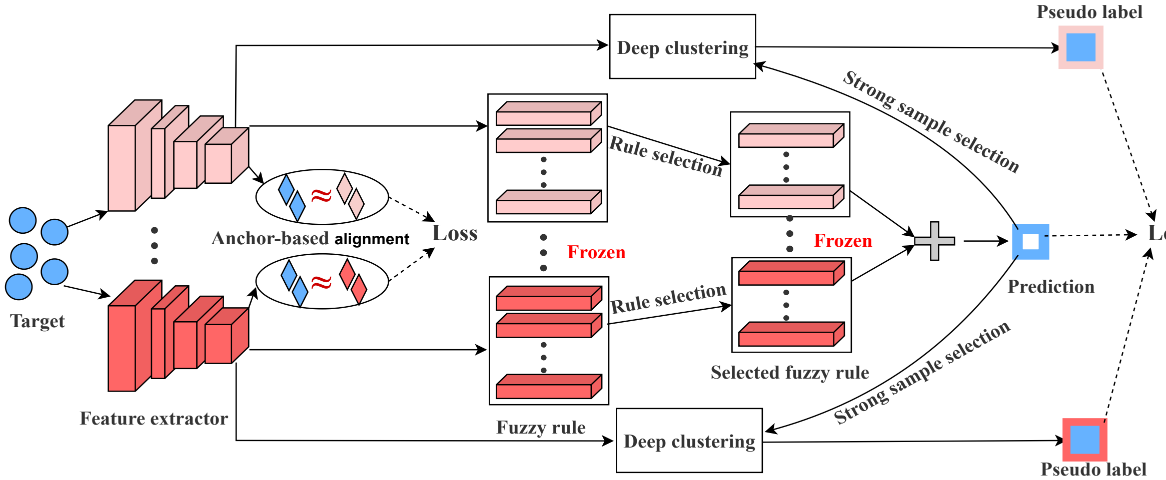
$$\boldsymbol{v}^l_{s_k} = \frac{\sum_{i=1}^{n^l_{s_k}} \mathbb{1}_{\boldsymbol{y}^i_{s_k} \in \boldsymbol{Y}_l} \cdot \phi_k(\phi(\boldsymbol{x}^i_{s_k}))}{\sum_{i=1}^{n^l_{s_k}} \mathbb{1}_{\boldsymbol{y}^i_{s_k} \in \boldsymbol{Y}_l}}, \quad (4)$$

$\boldsymbol{Y}_l$ is a label set containing similar classes. $n^l_{s_k}$ is the number of samples in the label set $\boldsymbol{Y}_l$. Given cluster prototypes $\{\boldsymbol{v}^l_{s_k}\}_{l=1}^{L_k}$, the membership of data $\boldsymbol{x}_{s_k} \in A_{kl}$ is generally defined as:

$$u^l_{s_k} = \frac{1}{\sum_{i=1}^{L_k} \left(\frac{\|\boldsymbol{v}^l_{s_k} - \phi_k(\phi(\boldsymbol{x}_{s_k}))\|}{\|\boldsymbol{v}^i_{s_k} - \phi_k(\phi(\boldsymbol{x}_{s_k}))\|}\right)^{\frac{2}{m-1}}}. \quad (5)$$



(a) Source private model training with auxiliary learning.



(b) Model adaption with self-supervised learning.

Fig. 1. The procedure of the proposed method. The solid arrow means data-flow, the dashed arrow means loss computing. Figure (a) indicates source model training. Auxiliary tasks are constructed by sharing source parameters. Fuzzy C-means clustering is employed to learn the prototypes and memberships to build the fuzzy rules. Source anchors are extracted to describe the source class information without referring to the original data. Figure (b) demonstrates domain adaptation. By freezing source rules, self-supervised learning is employed to fine-tune feature extractors. Anchor-based alignment is built to match domains on the data-level by extracting invariant information. Deep clustering and a sample selection strategy are designed to predict pseudo labels with low noise which learn specific target information.

Using the membership calculated via equation (5), the cluster prototypes are updated with training processing as:

$$\boldsymbol{v}_{s_k}^l = \frac{\sum_{i=1}^{n_{s_k}} (u_{s_k}^{li})^m \cdot \phi_k(\phi(\boldsymbol{x}_{s_k}^i))}{\sum_{i=1}^{n_{s_k}} (u_{s_k}^{li})^m}. \tag{6}$$

The loss function of training classifiers defined by fuzzy rules in each source domain is:

$$\begin{aligned}\mathcal{L}_k =& \mathcal{L}(P_{kl}(\phi_k(\phi(\boldsymbol{x}_{s_k}))), \tilde{\boldsymbol{y}}_{s_k}) + \\ & \mathcal{L}(\sum_{l=1}^{L_k} u_{s_k}^l \cdot P_{kl}(\phi_k(\phi(\boldsymbol{x}_{s_k}))), \tilde{\boldsymbol{y}}_{s_k}).\end{aligned}$$

To enhance the performed generality across tasks of the private source models, the auxiliary learning strategy is designed to training multiple source models jointly, and is expected to make full use of the classification information from other source domains. Not all fuzzy rules from a source domain can be performed on other different source domains because of data shift. Hence, for the kth source domain, we choose half nearest rules (denote as L_{near}) from different source domains as auxiliary tasks to improve the generality of source models. The auxiliary tasks are trained as:

$$\mathcal{L}_{aux} = \frac{1}{K-1} \sum_{k' \neq k}^{K} \mathcal{L}(\sum_{l \in L_{near}} u_{s_{k'}}^l \cdot P_{k'l}(\phi_{k'}(\phi(\boldsymbol{x}_{s_k}))), \tilde{\boldsymbol{y}}_{s_k}).$$

Then the overall objective of the kth source model is:

$$\mathcal{L}_s = \mathcal{L}_k + \alpha \mathcal{L}_{aux}. \tag{7}$$

Grouping similar classes to construct fuzzy rules can enrich information of every fuzzy set but degrade the representation of each class. To preserve the identification of each class, source class anchors of each domain are generated to extract present features that highly reflect class information. By this, the anchors can describe classes without referring to the original data. When fuzzy rule number is equal to the class number, which means each class has its individual rule, the clustering prototypes in equation (6) will act as source anchors. Otherwise, the averaged mean values of normalized classifier weight vectors are adopted as source anchors. By learning the anchors based on the source private model parameters, source data will not be leaked by decoding these anchors. Thus, employing these anchors does not harm data privacy. The anchor of the cth class is calculated as:

$$\boldsymbol{f}_{s_k}^c = \begin{cases} \frac{1}{L_k} \sum_{l=1}^{L_k} \mathrm{Norm}(P_{kl}) \text{ if } L_K \neq C; \\ \frac{\sum_{i=1}^{n_{s_k}} (u_{s_k}^{li})^m \cdot \phi_k(\phi(\boldsymbol{x}_{s_k}^i))}{\sum_{i=1}^{n_{s_k}} (u_{s_k}^{li})^m}, \text{ if } L_K = C; \end{cases} \tag{8}$$
$$c = l, c = 1, 2, \cdots, C$$

B. Pseudo Target Label Collection

Given target domain $\mathcal{D}_t = \{\boldsymbol{x}_t^j\}_{j=1}^{n_t}$, without access to source data, traditional domain adaptation methods relying on matching source and target samples cannot be adopted. To tackle the target task, we employ a pseudo labelling strategy to generate the target model from source models. As source models are available, we feed target data to the kth source model, and select most half nearest rules (denote as L_{near}) to predict target labels. At the very beginning, initializing the target clustering prototypes $\{v_{t_k}^l\}_{k=1}^{L_k}$ as source clustering prototypes $\{v_{s_k}^l\}_{k=1}^{L_k}$, the membership is calculated as:

$$u_{t_k}^l = \frac{1}{\sum_{i=1}^{L_k} (\frac{\|\boldsymbol{v}_{t_k}^l - \phi_k(\phi(\boldsymbol{x}_t))\|}{\|\boldsymbol{v}_{t_k}^i - \phi_k(\phi(\boldsymbol{x}_t))\|})^{\frac{2}{m-1}}}. \tag{9}$$

The prediction of applying source classifiers is expressed as:

$$\hat{\boldsymbol{y}}_{t_p} = \sum_{l \in L_{near}} u_{t_k}^l \cdot P_{kl}(\phi_k(\phi(\boldsymbol{x}_t))); \tag{10}$$

Target prototypes and memberships are then updated alternately with the process of training by fixing the other, which can be expressed as:

$$\begin{aligned} \boldsymbol{v}_{t_k}^l &= \frac{\sum_{i=1}^{n_t} (u_{t_k}^{li})^m \cdot \phi_k(\phi(\boldsymbol{x}_t^i))}{\sum_{i=1}^{n_t} (u_{t_k}^{li})^m}; \\ u_{t_k}^l &= \frac{1}{\sum_{i=1}^{L_k} (\frac{\|\boldsymbol{v}_{t_k}^l - \phi_k(\phi(\boldsymbol{x}_t))\|}{\|\boldsymbol{v}_{t_k}^i - \phi_k(\phi(\boldsymbol{x}_t))\|})^{\frac{2}{m-1}}}. \end{aligned} \tag{11}$$

The pseudo target labels provided by the pre-trained source model could be noisy due to the data bias between domains. To reduce the label noise, the distillation strategy is designed to collect high-confident target labels assumed to be correct. We call these target samples strong samples. The strong samples are used to further update the pseudo labels of all target samples, which is expected to improve the accuracy of target predictions.

First, denote $\boldsymbol{r} = [r_1, r_2, ..., r_C]$ as the probability vector returned by source classifiers $\{P_{kl}\}_{l=1}^{L_k}$ as in equation (10) which indicates the probability of a target sample belonging to the source classes. A threshold a_c of the cth class is defined to identify the potential of a target label being correct. For a pseudo label $\hat{\boldsymbol{y}}_{t_p} = c$, if $r_c \geqslant a_c$, we think this target label is correct with a high probability.

In addition, based on deep clustering [16], [39], we adopt the nearest neighbor to estimate the target labels. Since target data is unlabeled, to estimate the clustering center in each class of target domain, target samples are fed to source classifiers to calculate the probability vectors. The initial clustering center by applying the kth source model then can be written as:

$$\boldsymbol{w}_{ck}^0 = \frac{\sum_{i=1}^{\hat{n}_{t_p}^c} r_c^i \cdot \phi_k(\phi(\boldsymbol{x}_t^i))}{\sum_{i=1}^{\hat{n}_{t_p}^c} r_c^i}, \tag{12}$$

$\hat{n}_{t_p}^c$ is the number of samples in the cth class predicted by equation (10). The clustering label of target sample is then estimated by:

$$\hat{\boldsymbol{y}}_{t_d} = \underset{\boldsymbol{x}_t \sim \mathcal{D}_t}{\arg\max^C} \frac{1}{\sum_{i=1}^{C} (\frac{\|\boldsymbol{w}_{ck}^0 - \phi_k(\phi(\boldsymbol{x}_t))\|}{\|\boldsymbol{w}_{ik}^0 - \phi_k(\phi(\boldsymbol{x}_t))\|})^2}. \tag{13}$$

The target domain is unlabeled with higher data uncertainty, but we hope to collect accurate class information to predict its labels. Thus, the soft class information reflected by probability vector $\boldsymbol{r}$ is transformed into hard class information by replacing the probability vector $\boldsymbol{r}$ with the predicted label $\hat{\boldsymbol{y}}_{t_d}$. This means the initial cluster centers and labels in equations (12) and (13) are upgraded as:

$$\begin{aligned} \boldsymbol{w}_{ck}^1 &= \frac{\sum_{i=1}^{\hat{n}_{t_d}^c} \mathbb{1}_{\hat{\boldsymbol{y}}_{t_d}^i = c} \cdot \phi_k(\phi(\boldsymbol{x}_t^i))}{\sum_{i=1}^{\hat{n}_{t_d}^c} \mathbb{1}_{\hat{\boldsymbol{y}}_{t_d}^i = c}}, \\ \hat{\boldsymbol{y}}_{t_d} &= \underset{\boldsymbol{x}_t \sim \mathcal{D}_t}{\arg\max^C} \frac{1}{\sum_{i=1}^{C} (\frac{\|\boldsymbol{w}_{ck}^1 - \phi_k(\phi(\boldsymbol{x}_t))\|}{\|\boldsymbol{w}_{ik}^1 - \phi_k(\phi(\boldsymbol{x}_t))\|})^2}. \end{aligned} \tag{14}$$

$\hat{n}_{t_d}^c$ is the number of cluster samples predicted by equation (13).

When $\hat{\boldsymbol{y}}_{t_p} = \hat{\boldsymbol{y}}_{t_d}$ and $r_{\hat{\boldsymbol{y}}_{t_p} = c} \geqslant a_c$, we select the corresponding target sample as a strong sample. After collecting strong samples, we update the clustering centers in equation (14) using the selected target samples, and then renew the pseudo labels by combining information from multiple models, which can be expressed as:

$$\begin{aligned} \boldsymbol{w}_{ck}^2 &= \frac{\sum_{i=1}^{\hat{n}_{sel}^c} \mathbb{1}_{\hat{\boldsymbol{y}}_{t_p}^i = c} \cdot \phi_k(\phi(\boldsymbol{x}_t^i))}{\sum_{i=1}^{\hat{n}_{sel}^c} \mathbb{1}_{\hat{\boldsymbol{y}}_{t_p}^i = c}}, \\ \hat{\boldsymbol{y}}_t &= \underset{\boldsymbol{x}_t \sim \mathcal{D}_t}{\arg\max^C} \frac{1}{\sum_{i=1}^{C} (\frac{\| \sum_{k=1}^{K} \gamma_k (\boldsymbol{w}_{ck}^2 - \phi_k(\phi(\boldsymbol{x}_t)))\|}{\| \sum_{k=1}^{K} \gamma_k (\boldsymbol{w}_{ik}^2 - \phi_k(\phi(\boldsymbol{x}_t)))\|})^2}. \end{aligned} \tag{15}$$

where γ_k is the combination weight satisfied $\gamma_k > 0$, $\sum_{k=1}^{K} \gamma_k = 1$, $\hat{n}_{sel}^c$ denotes the number of selected samples in the cth class.

C. Model Adaptation and Target Task Prediction

When predicting the target task, to fit source models to the target domain, we design a self-supervised strategy to train the target model using the collected pseudo labels. Anchor-based alignment is built to force target data to the source feature spaces.

In freezing source classifiers, only feature extractors are fine-tuned to extract invariant information. When applying the kth fuzzy model, the corresponding generated target model is trained by minimizing the errors between the predictions and the pseudo labels, which is:

$$P_{t_k} = \underset{\substack{\phi_k,\phi \\ \boldsymbol{x}_t \sim \mathcal{D}_t}}{\arg\min} \mathcal{L}(\sum_{k}^{K} \gamma_k P_{t_k}(\phi_k(\phi(\boldsymbol{x}_t))), \hat{\boldsymbol{y}}_t),$$

where

$$\mathcal{L} = -\frac{1}{n_t}\sum_{i=1}^{n_t} \hat{\boldsymbol{y}}_t \log(\sum_{k}^{K} \gamma_k P_{t_k}(\phi_k(\phi(\boldsymbol{x}_t^i)))).$$

P_{t_k} is a linear combination of source classifiers under fuzzy rules, which is:

$$P_{t_k} = \sum_{l \in L_{near}} u_{t_k}^l \cdot P_{kl}.$$

$u_{t_k}^l$ is calculated as in equation (11).

To reduce the domain shift between source and target domains on the label-level, information maximization loss is employed to parameterize the target outputs being individually certain and globally diverse by encoding the target outputs to one-hot vectors, which is:

$$L_{div}^k = \sum \bar{\boldsymbol{p}}_t \log(\bar{\boldsymbol{p}}_t),$$

$\bar{\boldsymbol{p}}_t = \frac{1}{n_t}\sum_{i=1}^{n_t} P_{t_k}(\phi_k(\phi(\boldsymbol{x}_t^i)))$ is a C-dimension vector.

To transform target data to source feature space on the data-level, anchor-based alignment is designed to reduce the data bias. Source anchors $\{f_{s_k}^c\}_{c=1}^C$ are generated from source data as in equation (8), which highly represent the class feature information. These anchors will not weaken the data privacy as they are transformations of the original data. Other users (e.g. target domain) cannot estimate source data by decoding the anchors. Target anchors are calculated according to the soft class information returned by applying source fuzzy rules, we still denote it as vector $\boldsymbol{r}$, the target anchor is:

$$\boldsymbol{f}_{t_k}^c = \frac{\sum_{i=1}^{n_b} r_c^i \cdot \phi_k(\phi(\boldsymbol{x}_t^i))}{\sum_{i=1}^{n_b} r_c^i}, \tag{16}$$

n_b is the batch size. The advantage of calculating target anchors over the batch is that when there is no sample of class c in the randomly selected batch, the anchor can still be generated according to the probabilities of samples from other classes belonging to class c.

The loss function of matching source and target anchors is:

$$\mathcal{L}_{anc}^k = \sum_{c=1}^{C} \|\boldsymbol{f}_{t_k}^c - \boldsymbol{f}_{s_k}^c\|^2.$$

The total loss of training for the target model is:

$$\mathcal{L}_t = \mathcal{L}(\sum_{k}^{K} \gamma_k P_{t_k}(\phi_k(\phi(\boldsymbol{x}_t))), \hat{\boldsymbol{y}}_t) + \sum_{k=1}^{K}(\beta L_{div}^k + \lambda \mathcal{L}_{anc}^k). \tag{17}$$

The target label is a combination of the predictions provided by all source classifiers:

$$\boldsymbol{y}_t = \sum_{k=1}^{K} \gamma_k P_{t_k}(\phi_k(\phi(\boldsymbol{x}_t))). \tag{18}$$

The processing of the proposed source-free multi-domain adaptation with fuzzy rule-based deep neural networks (SF-FDN) are described in Algorithms 1 and 2.

Algorithm 1 SF-FDN: Source private model training.

Input: Source domains $\{\mathcal{D}_{s_k}\}_{k=1}^K = \{\boldsymbol{x}_{s_k}, \boldsymbol{y}_{s_k}\}_{k=1}^K$;
Initialization: Backbone ϕ, feature extractors $\{\phi_k\}_{k=1}^K$;
Define $L_k \leftarrow$ group similar classes via $\rho_{ij} > a_\rho$ as in equations (1);
Initialize $\{P_{kl}\}_{k,l=1}^{K,L_k}$;
// In each source domain:
Initialize $\boldsymbol{v}_{s_k}^l \leftarrow \phi_k(\phi(\boldsymbol{x}_{s_k})), \boldsymbol{y}_{s_k}$ as in equation (4);
for $\epsilon = 1$, $\epsilon < \mathcal{I}_s$, $\epsilon + +$, **do**
    Calculate $u_{s_k}^l \leftarrow \boldsymbol{v}_{s_k}^l, \phi_k(\phi(\boldsymbol{x}_{s_k}))$ as in equation (5);
    Update $\boldsymbol{v}_{s_k}^l \leftarrow u_{s_k}^l, \phi_k(\phi(\boldsymbol{x}_{s_k}))$ as in equation (6);
    Calculate loss $\mathcal{L}_s$ as in equation (7);
    Update ϕ, $\{\phi_k\}_{k=1}^K$, $\{P_{kl}\}_{k,l=1}^{K,L_k}$ by $\mathcal{L}_s$;
end for
// Generate source anchors as in equation (8):
if $L_k \neq C$ **then**
    $\boldsymbol{f}_{s_k}^c \leftarrow \{P_{kl}\}_{l=1}^{L_k}$;
else
    $\boldsymbol{f}_{s_k}^c \leftarrow u_{s_k}^l, \phi_k(\phi(\boldsymbol{x}_{s_k})), l = c$;
end if
Output: Source models $\{\phi, \phi_k, \{P_{kl}\}_{l=1}^{L_k}\}_{k=1}^K$, source anchors $\{\boldsymbol{f}_{s_k}^c\}_{k,c=1}^{K,C}$.

The time complexity of the whole network is $32.8 \times 10^8 + 5.2 \times 10^5 + (256 \cdot C) \times L_k$, while the space complexity is $109 \times 10^6 + 5.3 \times 10^5 + (256 \cdot (C+1)) \times L_k$.

IV. Experiments

In this section, the proposed fuzzy rule-based source-free multi-domain adaptation method is validated on popular real-world visual datasets. Both homogeneous and heterogeneous label spaces are applied to validate the proposed method. For closed-set and partial domain adaptation, accuracy of the whole domain is used to evaluate the performance, while for open-set, per-class accuracy is used.

In the following, section IV-A introduces the datasets, compared methods and parameter settings. Experiment results and analysis are displayed in section IV-B. Section IV-C analyzes the generality of the source-only model. Section IV-D analyzes the influence of rule numbers. The ablation study is carried out in section IV-E. Section IV-G validates the proposed method under partial and open-set domain adaptation scenarios. Section IV-H displays the data visualization. Section

Algorithm 2 SF-FDN: Target model adaptation.

Input: Source models $\{\phi, \phi_k, \{P_{kl}\}_{l=1}^{L_k}\}_{k=1}^{K}$, source anchors $\{\boldsymbol{f}_{s_k}^{c}\}_{k,c=1}^{K,C}$, target domain $\mathcal{D}_t = \{\boldsymbol{x}_t\}$;
//Feed target samples to each source model:
Initialize $\boldsymbol{v}_{t_k}^{l} \leftarrow \boldsymbol{v}_{s_k}^{l}$;
Initialize $u_{t_k}^{l} \leftarrow \boldsymbol{v}_{t_k}^{l}, \phi_k(\phi(\boldsymbol{x}_t))$ as in equation (9);
for $\epsilon = 1, \epsilon < \mathcal{I}_t, \epsilon + +,$ **do**
    Predict $\hat{\boldsymbol{y}}_{t_p}, r_c \leftarrow u_{t_k}^{l}, P_{kl}(\phi_k(\phi(\boldsymbol{x}_t)))$ as in equation (10);
    Update $\boldsymbol{v}_{t_k}^{l} \leftarrow u_{t_k}^{l}, \phi_k(\phi(\boldsymbol{x}_t))$;
    Update $u_{t_k}^{l} \leftarrow \boldsymbol{v}_{t_k}^{l}, \phi_k(\phi(\boldsymbol{x}_t))$ as in equation (11);
    Calculate $\boldsymbol{w}_{ck}^{0} \leftarrow r_c, \phi_k(\phi(\boldsymbol{x}_t))$ where $r_c \geqslant a_c$ as in equation (12);
    Predict $\hat{\boldsymbol{y}}_{t_d}, r_{\hat{\boldsymbol{y}}_{t_p}=c} \leftarrow \boldsymbol{w}_{ck}^{0}, \phi_k(\phi(\boldsymbol{x}_t))$ as in equation (13);
    Update $\boldsymbol{w}_{ck}^{1} \leftarrow \hat{\boldsymbol{y}}_{t_d}, \phi_k(\phi(\boldsymbol{x}_t))$;
    Update $\hat{\boldsymbol{y}}_{t_d} \leftarrow \boldsymbol{w}_{ck}^{1}, \phi_k(\phi(\boldsymbol{x}_t))$ as in equation (14);
    Update $\boldsymbol{w}_{ck}^{2} \leftarrow \hat{\boldsymbol{y}}_{t_p}, \phi_k(\phi(\boldsymbol{x}_t))$ where $\hat{\boldsymbol{y}}_{t_p} = \hat{\boldsymbol{y}}_{t_d}$ and $r_c, r_{\hat{\boldsymbol{y}}_{t_p}=c} \geqslant a_c$;
    Collect $\hat{\boldsymbol{y}}_t \leftarrow \gamma_k, \boldsymbol{w}_{ck}^{2}, \phi_k(\phi(\boldsymbol{x}_t))$ as in equation (15);
    Generate $\boldsymbol{f}_{t_k}^{c} \leftarrow r_c, \phi_k(\phi(\boldsymbol{x}_t))$ over the batch as in equation (16);
    Calculate loss $\mathcal{L}_t$ as in equation (17);
    Fine-tune ϕ, $\{\phi_k\}_{k=1}^{K}$ by $\mathcal{L}_t$;
end for
Predict $\boldsymbol{y}_t$ as in equation (18);
Output: Target labels.

IV-I analysis the convergence of the proposed model. Section IV-J tests the application of fuzzy rules on segmentation task.

A. Datasets and Baselines

The proposed method is tested on four real-world datasets. Datasets details are listed in Table II.

TABLE II
CLASSES, DOMAINS AND SAMPLES IN EXPERIMENT DATASETS.

Dataset\Classes	Domain\Samples	Total Samples	Tasks
Office-31\31	Amazon\2817 Webcam\795 DSLR\498	4110	WD-A AD-W AW-D
ImageCELF-DA\12	Caltech\600 ImageNet\600 Pascal\600	1800	IP-C PC-I IC-P
Office-Caltech\10	Amazon\958 Webcam\295 DSLR\157 Caltech\1123	2533	CDW-A ACD-W ACW-D ADW-C
Office-Home\65	Art\2427 Clipart\4365 Product\4439 RealWorld\4357	15588	CPR-A APR-C ACR-P ACP-R

Office-31 and ImageCLEF-DA include three domains sharing 31 and 12 categories, respectively. Three tasks of each dataset can be built: $AW-D$; $AD-W$; $WD-A$ from Office-31 and $IC-P$; $IP-C$; $CP-I$ from ImageCLEF-DA.

Office-Caltech10 and Office-Home contain four domains sharing 10 and 65 categories, respectively. Each of them has four tasks: $ADW-C$; $CDW-A$; $ACD-W$, $ACW-D$ from Office-Caltech10 and $ACP-R$; $ACR-P$; $APR-C$; $CPR-A$ from Office-Home.

Domain adaptation with heterogeneous label spaces are validated on Office-Home.

The baselines include related domain adaptation methods with and without source data under both homogeneous and heterogeneous settings which employ the same learning schemes such as self-supervision to adapt the target model. For fair comparison, the baseline methods are trained based on ResNet. For single-source methods, we report average performance from all adapted source models, where the source model is trained based on single source. Comparison with single-source methods aims to prove the superiority of learning from multiple domains. Comparison with source available methods aims to validate that the proposed method can handle more complex domain adaptation where source-available baselines cannot be applied with less information but achieve better performance simultaneously. Comparison with non-fuzzy methods not only indicates the advantage of fuzzy model, but also shows the superiority of the proposed techniques in data-matching and pseudo label selection. Source data available methods include:

- TransN: Transferable Normalization [40];
- MDD: Margin disparity discrepancy [2];
- JUMBOT: Joint unbalanced minibatch optimal transport [41];
- RBDA: Reducing bias to source samples [3];
- RWOT: Reliable weighted optimal transport [8];
- LtC-MSDA: Learning to combine [5];
- MSCLDA: Learning source contribution for multi-domain adaptation [14];
- DCA: Multi-domain adaptation with dynamic classifier alignment [15];
- SAN: Partial transfer learning with selective adversarial networks [42];
- ETN: Learning to transfer examples for partial domain adaptation [43];
- SAFN: Adaptive feature norm approach for unsupervised domain adaptation [44];
- DARL: Domain adversarial reinforcement learning for partial domain adaptation [45];
- MSAN: Attention guided for partial domain adaptation [12];
- OSBP: Open set domain adaptation by backpropagation [46];
- STA: Separate to adapt: Open set domain adaptation via progressive separation [47];
- DAOD: Open set domain adaptation: theoretical bound and algorithm [48];
- LtGUR: Learning to generate the unknowns as a remedy for open-set adaptation [49].

Source free methods include:

- BAIT: Domain adaptation without source data by casting a bait [33];

- PrDA: Progressive domain adaptation [50];
- SHOT: Source hypothesis transfer with information maximization [16];
- SDDA: Source data free domain adaptation- domain impression [51];
- G-SFDA: Generalized source-free domain adaptation [18];
- AAN: Adaptive adversarial network [52];
- NRC: Intrinsic neighborhood structure for source-free domain adaptation [53];
- JNUSF: Source-free domain adaptation with Jacobian Norm [54];
- CDCL: Cross-domain contrastive learning for unsupervised domain adaptation [21];
- PGL: Source-free progressive graph learning for open-set domain adaptation [55];

ETN, SAFN, DARL and MSAN are compared under a partial domain adaptation setting, while OSBP, STA, DAOD, LtGUR and PGL are compared under an open-set domain adaptation setting. All compared results are collected from previous publications. For single source-free domain adaptation methods, we take the average predictions from all source domains as the multi-source results. Source model only indicates predicting target labels using source model directly without adaptation (ResNet with an extra feature extraction layer).

$ResNet$ is employed as the backbone complemented by Pytorch. Parameters are updated based on back-propagation with Stochastic Gradient Descent (SGD) [56]. The details of parameter are listed in Table III. Most parameter values used are experience values. Threshold a_c is defined as medium value of the predicted probabilities in each category. This is following the previous studies [15], [57]. Fuzzy rule number is affected by the threshold a_ρ, the value is defined based on experiments. For datasets ImageCLEF-DA and Office-Caltech10, we set the rule numbers as their class numbers. For datasets Office-31 and Office-Home, domains containing few samples applies a small value while domains containing a large number of samples take on a greater value of a_ρ. For dataset Office-31, ImageCLEF-DA, Office-Caltech10, the value of a_ρ is between $[0.4, 0.5]$, for Office-Home, the value is between $[0.45, 0.65]$. Domains A, D and W have 15, 16 and 15 rules respectively, domains A, C, P and R have 5, 6, 7 and 7 rules respectively. Under heterogeneous label space settings on dataset Office-Home, for partial domain adaptation, domains A, C, P and R have 5, 6, 7 and 7 rules respectively, while for open-set domain adaptation, domains A, C, P and R have 5, 6, 8 and 8 rules respectively.

TABLE III
PARAMETERS AND VALUES

Parameter	Relavant Component	Value
Learning rate η	SGD,Classifier	0.01
Learning rate of Backbone	SGD, ResNet	0.001
Momentum	Training process	0.9
Training progress ϵ	Training process	[0,1]
Batch size n_b	Training process	64
Smoothing parameter μ	Source label	0.1
Threshold a_c	Target pseudo label	medium value of predictions
Correlation coefficient threshold a_ρ	Fuzzy rule	[0.4,0.5] in Office-31, ImageCLEF-DA, Office-Caltech10 [0.45, 0.65] in Office-Home
Fuzzy rule	Classifier	ImageCLEF-DA: class number Office-Caltech10: class number Office-31: A:15, D: 16, W: 15 Office-Home: A:5, C:6, P:7, R:7

B. Results and Analysis

Results reported include both mean averaged combination (SF-FDNm) and weighted average combination (SF-FDNw) (in equation (18)). Without extra explanation, ablation experiments in the following parts are carried out using mean averaged combination.

Tables IV, V, VI and VII show the results of the proposed method and the baselines under closed-set domain adaptation. We compare the proposed method SF-FDN with a fuzzy rule-based baseline MDAFuz, and other non-fuzzy baselines. It indicates that the proposed method performs the best on most tasks and achieve the highest average performance on four datasets.

To compare the proposed SF-FDN method with the source-free domain adaptation methods, the average accuracy is improved by 1.8% on dataset Office-31, 0.6% on dataset ImageCLEF-DA, 0.8% on dataset Office-Caltech10 and 1.6% on dataset Office-Home, respectively. This indicates that introducing a fuzzy system to handle soft information among samples from different categories can leverage richer transfer knowledge across domains. The proposed SF-FDN improves the average accuracy of baselines with source data by 0.7% on dataset Office-31, 0.6% on dataset Office-Caltech10 and 1.1% on dataset Office-Home compared with the latest domain adaptation method with source data. On dataset ImageCLEF-DA, the proposed SF-FDN and baseline MDAFuz, another method based on fuzzy rules, achieve the same average performance. It means extracting soft information is more suitable for this dataset. Even though the two methods gain the same average performance, we deal with domain adaptation under a more difficult setting, and obtain a superior performance on most tasks in this dataset. It also shows that, except for the proposed method, several source-free methods based on pseudo-labelling, such as SHOT and BAIT, produce a similar performance compared with other methods with source data, meaning that self-supervised training is advantageous in taking usable information in the target domain to help the transfer.

C. Generality Analysis of Source Private Model

When handling source-free domain adaptation employing a self-training strategy. there are two main questions: how to improve the generality of the source only model and how to collect pseudo labels with low noise. In this paper, we expect fuzzy rules to have the superiority to take full use of class information to learn classifiers with high cross-domain ability, and design auxiliary tasks to enhance the generality of source-only models. The high cross-domain performance of the source model is beneficial to collecting low noisy pseudo target labels

TABLE IV
COMPARISON (%) OF THE PROPOSED FUZZY RULE-BASED DEEP NETWORK AND THE BASELINES ON DATASET OFFICE-31

Standards	Method	AW-D	AD-W	WD-A	Avg
Source data	ResNet	99.3	96.7	62.5	86.2
	TransN	97	97.2	73.8	89.3
	RBDA	**100.0**	99.0	74.2	91.1
	MDD	96.8	96.6	73.4	88.9
	RWOT	97.3	97.3	**77.7**	90.8
	MSCLDA	99.8	98.8	73.7	90.8
	MDAFuz	99.7	99.0	74.0	90.9
	DCA	99.6	98.9	75.1	91.2
Source free	Source model only	97.5	95.4	60.2	84.4
	BAIT	98.8	98.5	71.1	89.5
	PrDA	96.7	93.8	73.2	87.9
	SHOT	94.9	97.8	75.0	89.2
	SDDA	99.8	99.0	67.7	88.8
	AAN	97.3	96.6	76.1	90.1
	NRC	97.9	94.9	75.2	89.3
	JNUSF	97.9	95.5	76.4	89.9
	CDCL	97.2	95.3	75.3	89.3
	SF-FDNm	**100.0**	99.2	75.8	91.6
	SF-FDNw	**100.0**	**99.4**	**76.3**	**91.9**

TABLE V
COMPARISON (%) OF THE PROPOSED FUZZY RULE-BASED DEEP NETWORK AND THE BASELINES ON DATASET IMAGECLEF-DA

Standards	Method	IC-P	IP-C	PC-I	Avg
Source data	ResNet	74.8	91.5	83.9	83.4
	RBDA	78.5	**98.0**	91.4	89.3
	RWOT	80.2	97.3	92.8	**90.1**
	MSCLDA	79.5	95.9	94.3	89.9
	MDAFuz	79.4	96.3	**94.5**	**90.1**
	DCA	78.9	96.2	93.9	89.7
Source free	Source model only	77.0	93.3	90.3	86.9
	SHOT	79.2	96.2	93.2	89.5
	SF-FDNm	80.2	97.3	92.7	**90.1**
	SF-FDNw	**80.3**	97.2	92.7	**90.1**

TABLE VI
COMPARISON (%) OF THE PROPOSED FUZZY RULE-BASED DEEP NETWORK AND THE BASELINES ON DATASET OFFICE-CALTECH10

Standards	Method	ADW-C	CDW-A	ACD-W	ACW-D	Avg
Source data	ResNet	82.5	91.2	98.9	99.2	93.0
	MSCLDA	94.1	95.3	99.1	98.5	96.8
	DCA	94.7	96.0	99.7	99.1	97.4
Source free	Source model only	92.1	96.3	98.0	99.5	96.5
	BAIT	95.7	**97.5**	98.0	97.5	97.2
	PrDA	94.6	97.3	97.6	97.1	96.7
	SHOT	**95.8**	95.7	99.6	96.8	97.0
	SF-FDNm	94.9	95.9	**100.0**	**100.0**	97.7
	SF-FDNw	95.3	96.7	**100.0**	**100.0**	**98.0**

TABLE VII
COMPARISON (%) OF THE PROPOSED FUZZY RULE-BASED DEEP NETWORK AND THE BASELINES ON DATASET OFFICE-HOME

Standards	Method	ACP-R	ACR-P	APR-C	CPR-A	Avg
Source data	ResNet	67.8	71.3	51.8	53.4	61.1
	TransN	76.7	75.7	54.1	64.1	67.6
	MDD	75.9	75.8	56.2	64.6	68.1
	RWOT	77.3	75.8	52.8	64.5	67.7
	JUMBOT	78.3	77.8	55.9	67.0	70.0
	MSCLDA	80.6	79.9	61.4	71.6	73.4
	LtC-MSDA	80.1	79.2	**64.1**	67.4	72.7
	DCA	81.4	80.5	63.6	72.1	74.4
Source free	Source model only	76.3	78.8	50.1	50.9	64.0
	BAIT	77.2	79.4	59.6	71.1	71.8
	PrDA	76.8	79.1	57.5	69.3	70.7
	SHOT	81.5	83.0	57.2	72.1	73.5
	G-SFDA	82.2	83.4	57.9	72.0	73.9
	AAN	81.4	81.1	58.4	69.9	73.9
	NRC	81.2	81.9	57.6	68.1	72.2
	JNUSF	81.2	81.1	56.8	70.8	72.5
	SF-FDNm	**82.7**	**83.7**	60.7	73.7	75.2
	SF-FDNw	**82.7**	83.5	61.5	**74.2**	**75.5**

at the very beginning. This section demonstrates experiments on a source-only model for analyzing the performance of the proposed enhancement strategy.

Taking datasets Office-31 and Office-Home as examples, Tables VIII and IX show the performance of source-only models trained without fuzzy rules and auxiliary tasks. The results are returned by applying source models on the target domain directly without fine-tuning, which indicates the ability across tasks of source models. Method “S” represents predictions of single source domain, “M” indicates the performance of multi-source domains. “Non-fuzzy” means the models are trained with auxiliary tasks but without fuzzy rules, “Non-auxiliary” means training with fuzzy rules but without auxiliary tasks, “Proposed” means both fuzzy rules and auxiliary tasks are used.

It can be seen that the multi-source model outperforms the single source model, indicating that enriching source knowledge is helpful in learning a classifier that can perform on multiple domains. The proposed method performs best on two datasets compared with models trained without fuzzy rules or auxiliary tasks. For a dataset with fewer categories and samples (Office-31), models trained with fuzzy rules but without auxiliary tasks achieve greater accuracy than those trained with auxiliary tasks but without fuzzy rules. It means fuzzy rules have a significant advantage to improve the generality of source models. While for a dataset with more categories and samples (Office-Home), auxiliary learning is more important to leveraging source knowledge as the models trained with auxiliary tasks outperform those without. This is because Office-Home contains more domains than Office-31, and some domains with low relatedness are learned together, degrading the performance of classifiers since a classifier fitting all domains well may not exist. This encourages us to explore which tasks should be learned together to reduce the negative transfer and improve future positive transfer.

TABLE VIII
ACCURACY (%) ON DATASET OFFICE-31 OF SOURCE ONLY MODELS.

Standards	Method	AW-D	AD-W	WD-A	Avg
Non-fuzzy	S	98.6	96.2	64.7	86.5
		98.6	96.2	64.5	
	M	98.8	96.4	65.8	87.0
Non-auxiliary	S	99.0	95.2	63.5	86.6
		99.6	97.2	65.0	
	M	99.4	**97.4**	66.0	87.6
Proposed	S	99.6	95.2	65.0	87.1
		99.8	97.2	65.9	
	M	**99.8**	**97.4**	**66.4**	**87.9**

TABLE IX
ACCURACY (%) ON DATASET OFFICE-HOME OF SOURCE ONLY MODELS.

Standards	Method	ACP-R	ACR-P	APR-C	CPR-A	Avg
Non-fuzzy	S	81.5	76.9	51.7	68.2	69.7
		80.8	77.3	52.2	68.2	
		81.1	78.5	51.4	68.8	
	M	**81.6**	78.1	52.0	**69.6**	70.3
Non-auxiliary	S	76.0	71.7	49.7	61.4	65.1
		73.5	72.2	47.2	60.0	
		75.3	77.2	50.4	66.1	
	M	79.7	76.9	53.1	66.7	69.1
Proposed	S	80.4	76.5	53.5	67.2	69.8
		79.5	77.6	54.0	68.2	
		80.6	**78.7**	53.1	67.8	
	M	80.7	**78.7**	**54.1**	68.7	**70.6**

D. *Influence of Rule Number*

To explore how the number of rules affects the performance of the transfer, this section shows the results of the proposed method trained with different rule numbers. Tables X and XI show the classification results of the proposed fuzzy rule-based method with different rule numbers on datasets Office-31 and Office-Home. Standard "1" indicates only one rule is used. Standard "$C/2$" means the rule number is half of the class number, while "C" means the rule number is set as a class number. The results show that rule numbers defined by the proposed grouping strategy based on correlation coefficient achieve the highest performance on target domain. It indicates that too few or too many rules can result in degradation of the transfer.

Fewer rules might fail to discover the specific information among different classes. Setting the rule number as a value equal to or greater than the number of classes should be advantageous in extracting class information and learning high-performance classifiers, but experiments reveal different assumptions on datasets containing many categories. We think this is caused by the data unbalance of categories and domains. For a classifier under a rule that contains fewer highly representative samples, the learning may fail to provide correct predictions because other class samples occupying a large proportion will dominate the training. Besides, having too many rules requires a high computing environment such as computer memory and compute units. Thus, defining appropriate fuzzy rules is beneficial in overcoming the problems mentioned above.

TABLE X
ACCURACY (%) ON DATASET OFFICE-31 WITH DIFFERENT RULE NUMBERS.

Standards	AW-D	AD-W	WD-A	Avg
1	99.5	98.7	75.5	91.3
C	99.8	98.7	75.3	91.2
Proposed	**100.0**	**99.2**	**75.8**	**91.6**

TABLE XI
ACCURACY (%) ON DATASET OFFICE-HOME WITH DIFFERENT RULE NUMBERS.

Standards	ACP-R	ACR-P	APR-C	CPR-A	Avg
1	**83.1**	83.1	60.1	73.3	74.9
C/2	82.0	82.4	**61.1**	72.6	74.5
C	82.8	79.7	56.5	66.0	71.3
Proposed	82.7	**83.7**	60.7	**73.7**	**75.2**

E. *Ablation Study*

To validate the performance of different modules used in the proposed method, Tables XIII and XIV display the ablation study on datasets Office-31 and Office-Home. Three modules affect the training of the target model:(1) selecting strong target samples to predict low noisy pseudo labels (denote as $\mathcal{L}_{sel}$); (2) balancing domain shift using information maximization loss (reflected by $\mathcal{L}_{div}$), and (3) forcing target data to source feature space using anchor-based alignment (reflected by $\mathcal{L}_{anc}$), the setting of ablation study is detailed in Tabel XII, "×" means training without the module, while "✓" means training with the module.

TABLE XII
SETTING OF ABLATION STUDY.

Standards	Target sample selection	Information maximization loss	Anchor-based alignment
L_{div}	✓	×	✓
L_{anc}	✓	✓	×
L_{sel}	×	✓	✓
Proposed	✓	✓	✓

It indicates that information maximization loss L_{div} is more important than other modules as the model trained without it produces a lower performance on both datasets. This is because L_{div} can circumvent the trivial solution for unlabeled target samples with the same one-hot encoding. Anchor-based alignment L_{anc} contributes more than sample selection L_{sel}. The difference is due to the quality of source anchors and target pseudo labels. The power of target sample selection relies on how good the pseudo labels are. Since target domain is unlabeled, and there is data shift, when the dataset is complex, the confidence pseudo labels can be collected are fewer. That is why the influence of sample selection seems small. Even in supervised learning, predicting a dataset containing a large number of samples and categories is more difficult than that in a dataset with fewer categories. Thus, the anchors and pseudo labels from a small dataset can describe class information more accurately than those from a large dataset.

TABLE XIII
ACCURACY (%) ON DATASET OFFICE-31 OF ABLATION STUDY.

Standards	AW-D	AD-W	WD-A	Avg
$\mathcal{L}_{div}$	99.9	98.4	73.1	90.5
$\mathcal{L}_{anc}$	99.6	98.4	72.9	90.3
$\mathcal{L}_{sel}$	99.9	98.6	74.8	91.1
Proposed	**100.0**	**99.2**	**75.8**	**91.6**

TABLE XIV
ACCURACY (%) ON DATASET OFFICE-HOME OF ABLATION STUDY.

Standards	ACP-R	ACR-P	APR-C	CPR-A	Avg
$\mathcal{L}_{div}$	82.0	81.8	55.4	70.6	72.4
$\mathcal{L}_{anc}$	**82.8**	**83.7**	60.3	73.0	75.0
$\mathcal{L}_{sel}$	**82.8**	**83.7**	60.4	73.3	75.1
Proposed	82.7	**83.7**	**60.7**	**73.7**	**75.2**

F. Trade-off Parameter Sensitivity Analysis

This section analyzes the sensitivity of trade-off parameter. Trade-off parameters control the contribution level of auxiliary task $\mathcal{L}_{aux}$, information maximization loss $\mathcal{L}_{div}$ and anchor-based alignment $\mathcal{L}_{anc}$. α ($\mathcal{L}_{aux}$) and β ($\mathcal{L}_{div}$) are experience values in previous domain adaptation methods [14], [16]. Here we provide the experiment on λ, which refects the term $\mathcal{L}_{anc}$. Taking datasets Office-31 and Office-Home as examples, the results are shown in Tables XV and XVI. It can been seen that when $\lambda = 0.5$, the proposed method achieves the highest performance.

TABLE XV
ACCURAY (%) ON DATASET OFFICE-31 WITH DIFFERENT VALUES OF PARAMETER λ.

λ	AW-D	AD-W	WD-A	Avg
0.3	99.8	98.5	75.2	91.2
0.5	**100.0**	**99.2**	**75.8**	**91.6**
0.7	99.9	98.4	75.3	91.2
1	99.8	98.6	75.2	91.2

TABLE XVI
ACCURAY (%) ON DATASET OFFICE-HOME WITH DIFFERENT VALUES OF PARAMETER λ.

λ	ACP-R	ACR-P	APR-C	CPR-A	Avg
0.3	82.0	83.6	60.4	73.0	74.8
0.5	**82.7**	**83.7**	**60.7**	**73.7**	**75.2**
0.7	82.6	82.9	60.8	73.5	75.0
1	82.1	83.6	60.6	73.2	74.9

G. Validation under Heterogeneous Label Space Setting

This section describes the experiments on the dataset Office-Home under heterogeneous label spaces settings, including partial and open-set domain adaptation. In partial domain adaptation where target label space is a proper subset of source label space, we choose 25 classes as target domain, and source domains include all classes. In open-set domain adaptation, source label space is contained inside target label space. We select 25 classes to build source domains while target domain includes all classes. The results are shown in Tables XVII and XVIII.

Compared with non-fuzzy baselines, the proposed method achieves the highest accuracy under both partial and open-set scenarios, meaning the proposed method based on fuzzy rules has superiority over other methods. In addition, it indicates that the target model generated from multiple source domains with joint training takes advantage of similar tasks to improve the transfer across domains.

TABLE XVII
COMPARISON (%) OF THE PROPOSED FUZZY RULE-BASED DEEP NETWORK AND THE BASELINES ON DATASET OFFICE-HOME UNDER PARTIAL DOMAIN ADAPTATION.

Standards	Method	ACP-R(25)	ACR-P(25)	APR-C(25)	CPR-A(25)	Avg
Source data	ResNet	71.2	67.2	45.4	61.6	61.4
	SAN	77.5	70.8	46.4	66.5	65.3
	ETN	79.6	75.7	57.4	69.0	70.4
	SAFN	79.9	76.4	58.2	72.9	71.9
	DARL	84.2	77.5	54.5	72.0	72.1
	JUMBOT	83.3	78.2	63.3	77.0	75.5
	MSAN	80.4	76.2	56.7	67.2	70.1
Source free	SHOT	**88.4**	82.4	64.0	77.6	78.1
	SF-FDN	88.3	**83.3**	**66.3**	78.2	**79.0**

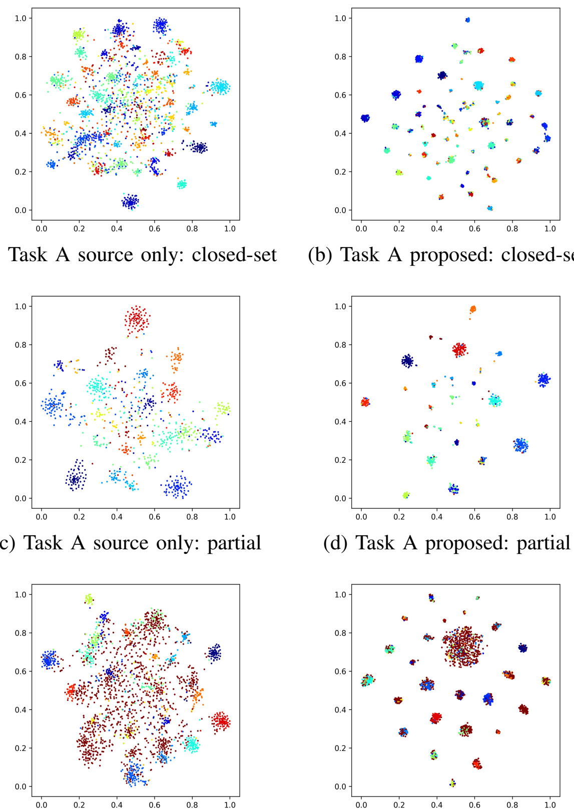
TABLE XVIII
COMPARISON (%) OF THE PROPOSED FUZZY RULE-BASED DEEP NETWORK AND THE BASELINES ON DATASET OFFICE-HOME UNDER OPEN-SET DOMAIN ADAPTATION.

Standards	Method	ACP(25)-R	ACR(25)-P	APR(25)-C	CPR(25)-A	Avg
Source data	ResNet	65.3	63.1	62.8	69.8	65.3
	OSBP	64.2	62.8	65.4	70.2	65.7
	STA	70.4	67.7	66.4	73.3	69.5
	DAOD	79.8	73.5	58.6	67.2	69.8
	LGUR	82.3	78.5	58.8	71.3	72.7
Source free	SHOT	82.4	79.3	61.4	64.5	71.9
	PGL	**86.1**	79.2	63.8	**75.1**	76.1
	SF-FDN	83.1	**83.5**	**66.5**	73.2	**76.6**

H. Visualization Analysis

Fig. 2 shows the data visualization of target domain A from Office-Home under both homogeneous and heterogeneous domain adaptation settings. We can see target classes distinctly separate from each other distinctly after adaptation. In partial domain adaptation, the target domain contains 25 classes, and quiet large distances can be seen within each pair of classes. In open-set domain adaptation, there are 65 classes in the target domain. However, source classifiers can only identify

25 classes, the other unshared 40 classes in the target domain are treated as unknown classes (samples in very deep red color). We can see the unknown classes are grouped after fine-tuning, while the share classes are divided clearly into different classes.



(a) Task A source only: closed-set (b) Task A proposed: closed-set

(c) Task A source only: partial (d) Task A proposed: partial

(e) Task A source only: open-set (f) Task A proposed: open-set

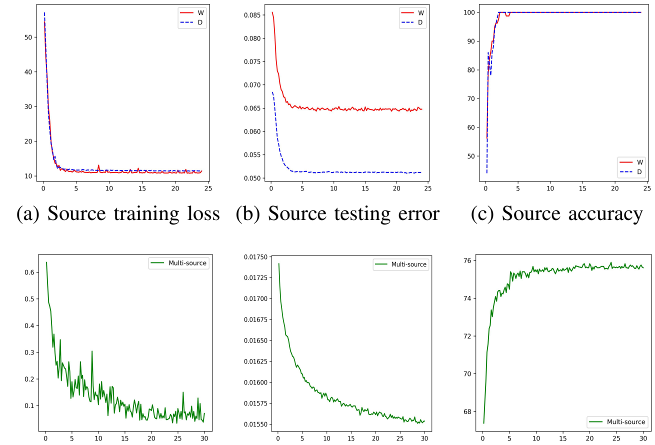
Fig. 2. T-SNE visualization on target domain A from dataset Office-Home.

I. Convergence Graph of Training and Adaptation Procedures

Fig. 3 indicates the convergence of the proposed SF-FDN model during training and adaptation procedures on source and target domains. It can be seen that the training loss decreases sharply at start and keeps stable after about 5 epochs. Testing error changes pretty similarly to the training loss. The accuracy of source and adapted target models increases with the training processing. For source domain, the training is convergent at about 5 epochs, while for target, the model is convergent at about 10 epochs.

J. Fuzzy Rule-Based Deep Neural Networks for Segmentation

To further validate the influence of fuzzy rules, we carry out experiments on a more complex task- image segmentation of city views. Two rules are manually designed to deal with dynamic stuffs and static stuffs. Table. XIX shows the results of applying FDN to segmentation task $GTA5 \rightarrow Cityscapes$. By simply introducing fuzzy rules to ResNet in the last classification layer, it can be seen the model with fuzzy rules achieves the highest average MIoU (mean intersection-over-union) compared with several typical non-fuzzy baselines.



(a) Source training loss (b) Source testing error (c) Source accuracy

(d) Target training loss (e) Target testing error (f) Target accuracy

Fig. 3. Training and testing loss of source private and target adapted models.

Fuzzy rules can significant improve the performance especially on dynamic stuffs such as car and bus.

V. Conclusion and Further Study

This paper proposes a fuzzy rule-based deep structure for source-free multi-domain adaptation. It is an early study of fuzzy domain adaptation without source data. The proposed method introduces the Takagi-Sugeno fuzzy model to source-free domain adaptation. It defines source fuzzy rule numbers by grouping similar classes into the same cluster according to the correlation coefficient within each pair of classes. To train source models with high generality, which is of advantage in predicting target labels with low noise at the beginning, auxiliary tasks are designed by jointly training fuzzy rules from other source domains. The auxiliary training strategy shares source parameters without referring to the original data from other domains, which can protect data privacy. To collect high confident target pseudo labels, a samples selection strategy is built by combining the predictions of source classifiers and deep clustering. Experiments on real-world datasets validate the superiority of the proposed method. The proposed method based on fuzzy rules results in higher performance than baselines trained with and without source data.

Some questions remain unsolved. For example, multiple source domains have their individual models, which requires a large memory of computer memory due to the sizeble number of parameters needed during training and transferring. Furthermore, similarities among each pair of source and target domains are different, meaning multiple source domains contribute to target domain differently. However, without access to source data, it is difficult to measure the similarities between source and target domains. In the future studies, we will try to solve the problems we have illustrated to reduce the computing complexity and learn the contribution of source domains in improving the transfer performance, also more complex task such as segmentation in non-satisfied situation based on fuzzy model.

TABLE XIX
SEGMENTATION PERFORMANCE (MIOU (%)) ON TASK $GTA5 \rightarrow Cityscapes$ OF FDN AND BASELINES.

Method	road	sidewalk	building	wall	fence	pole	light	sign	veg.	sky	person	rider	car	bus	m-cycle	bike	Avg
ResNet	74.2	27.5	69.9	10.5	8.7	23	0.2	0.3	77.9	78.6	45.3	12.3	74.6	26.1	16.2	28.5	35.9
MDAN [58]	80.6	**34.3**	73.9	15.9	1.9	22.9	0.1	0	73.6	58.9	48.4	12.2	78.8	36.8	14.2	23.7	36.0
AdaSeg [59]	81.6	26.6	79.5	20.7	20.5	23.7	29.9	**22.6**	81.6	**81.2**	52.4	20.2	79.1	28.8	24.7	26.2	43.7
MinEnt [60]	80.2	31.9	**81.4**	25.1	20.8	**24.6**	**30.2**	17.5	83.2	76.2	55.2	24.6	75.5	31.2	**27.4**	22.9	44.2
FDN	**87.7**	26.4	79.9	**26.2**	19.2	24	30	17.1	**85.1**	76.6	**58.4**	**25.0**	**82.6**	**40.1**	22.7	21.9	**45.2**

REFERENCES

[1] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, 2009.

[2] Y. Zhang, T. Liu, M. Long, and M. Jordan, "Bridging theory and algorithm for domain adaptation," in *Proc. Int. Conf. Mach. Learn. (ICML)*. Long beach, USA: PMLR, June 9 - 15 2019, pp. 7404–7413.

[3] Y. Ye, Z. Huang, T. Pan, J. Li, and H. T. Shen, "Reducing bias to source samples for unsupervised domain adaptation," *Neural. Netw.*, vol. 141, pp. 61–71, 2021.

[4] A. Sharma, T. Kalluri, and M. Chandraker, "Instance level affinity-based transfer for unsupervised domain adaptation," in *Proc. IEEE. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Virtual online, June 19 - 25 2021, pp. 5361–5371.

[5] H. Wang, M. Xu, B. Ni, and W. Zhang, "Learning to combine: Knowledge aggregation for multi-source domain adaptation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*. Springer, 2020, pp. 727–744.

[6] A. Rozantsev, M. Salzmann, and P. Fua, "Residual parameter transfer for deep domain adaptation," in *Proc. IEEE. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Salt Lake City, Utah, USA, June 18 - 22 2018, pp. 4339–4348.

[7] J. Davis and P. Domingos, "Deep transfer via second-order markov logic," in *Proc. Int. Conf. Mach. Learn. (ICML)*, Montreal, Canada, July 14 - 18 2009, pp. 217–224.

[8] R. Xu, P. Liu, L. Wang, C. Chen, and J. Wang, "Reliable weighted optimal transport for unsupervised domain adaptation," in *Proc. IEEE. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Virtual online, June 14 - 19 2020, pp. 4394–4403.

[9] Q. Tian, H. Sun, C. Ma, M. Cao, Y. Chu, and S. Chen, "Heterogeneous domain adaptation with structure and classification space alignment," *IEEE Trans. Cybern.*, 2021.

[10] J. Li, M. Jing, H. Su, K. Lu, L. Zhu, and H. T. Shen, "Faster domain adaptation networks," *IEEE Trans. Knowl. Data Eng.*, 2021. [Online]. Available: 10.1109/TKDE.2021.3060473

[11] Z. Fang, Y. Li, J. Lu, J. Dong, B. Han, and F. Liu, "Is out-of-distribution detection learnable?" in *Proc. Int. Conf. Neural Inf. Process. Syst. (NeurIPS)*, 2022.

[12] C. Zhang and Q. Zhao, "Attention guided for partial domain adaptation," *Inf. Sci.*, vol. 547, pp. 860–869, 2021.

[13] K. Saito, D. Kim, S. Sclaroff, and K. Saenko, "Universal domain adaptation through self supervision," in *Proc. Int. Conf. Neural Inf. Process. Syst. (NeurIPS)*, Virtual online, December 6 - 12 2020, pp. 16 282–16 292.

[14] K. Li, J. Lu, H. Zuo, and G. Zhang, "Multi-source contribution learning for domain adaptation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 4, 2021. [Online]. Available: 10.1109/TNNLS.2021.3069982

[15] K. Li, J. Lu, H. Zuo, and G. Zhang, "Dynamic classifier alignment for unsupervised multi-source domain adaptation," *IEEE Trans. Knowl. Data Eng.*, 2022. [Online]. Available: 10.1109/TKDE.2022.3144423

[16] J. Liang, D. Hu, and J. Feng, "Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation," in *Proc. Int. Conf. Mach. Learn. (ICML)*. Virtual online: PMLR, July 12 - 18 2020, pp. 6028–6039.

[17] R. Li, Q. Jiao, W. Cao, H.-S. Wong, and S. Wu, "Model adaptation: Unsupervised domain adaptation without source data," in *Proc. IEEE. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Virtual online, June 14 - 19 2020, pp. 9641–9650.

[18] S. Yang, Y. Wang, J. van de Weijer, L. Herranz, and S. Jui, "Generalized source-free domain adaptation," in *Proc. Int. Conf. Comput. Vis. (ICCV)*, Virtual online, October 11 - 17 2021, pp. 8978–8987.

[19] Y. Hou and L. Zheng, "Source free domain adaptation with image translation," *arXiv preprint arXiv:2008.07514*, 2020.

[20] V. K. Kurmi, V. K. Subramanian, and V. P. Namboodiri, "Domain impression: A source data free domain adaptation method," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Virtual online, January 5 - 9 2021, pp. 615–625.

[21] R. Wang, Z. Wu, Z. Weng, J. Chen, G.-J. Qi, and Y.-G. Jiang, "Cross-domain contrastive learning for unsupervised domain adaptation," *IEEE Transactions on Multimedia*, 2022.

[22] S. M. Ahmed, D. S. Raychaudhuri, S. Paul, S. Oymak, and A. K. Roy-Chowdhury, "Unsupervised multi-source domain adaptation without access to source data," in *Proc. IEEE. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Virtual online, June 19 - 25 2021, pp. 10 103–10 112.

[23] J. Lu, V. Behbood, P. Hao, H. Zuo, S. Xue, and G. Zhang, "Transfer learning using computational intelligence: a survey," *Knowl. Based. Syst.*, vol. 80, pp. 14–23, 2015.

[24] G. Ma, F. Liu, G. Zhang, and J. Lu, "Learning from imprecise observations: An estimation error bound based on fuzzy random variables," in *Proc. IEEE. Int. Conf. Fuzzy Syst. (FUZZ-IEEE)*. IEEE, 2021, pp. 1–8.

[25] J. Lu, H. Zuo, and G. Zhang, "Fuzzy multiple-source transfer learning," *IEEE Trans. Fuzzy Syst.*, vol. 28, no. 12, pp. 3418–3431, 2020.

[26] H. Han, H. Liu, Z. Liu, and J. Qiao, "Interactive transfer learning-assisted fuzzy neural network," *IEEE Trans. Fuzzy Syst.*, 2021.

[27] X. Che, H. Zuo, J. Lu, and D. Chen, "Fuzzy multi-output transfer learning for regression," *IEEE Trans. Fuzzy Syst.*, 2021.

[28] P. Xu, Z. Deng, J. Wang, Q. Zhang, K.-S. Choi, and S. Wang, "Transfer representation learning with tsk fuzzy system," *IEEE Trans. Fuzzy Syst.*, vol. 29, no. 3, pp. 649–663, 2021.

[29] T. K. B. Mudiyanselage, X. Xiao, Y. Zhang, and Y. Pan, "Deep fuzzy neural networks for biomarker selection for accurate cancer detection," *IEEE Trans. Fuzzy Syst.*, vol. 28, no. 12, pp. 3219–3228, 2019.

[30] G. Wang and J. Qiao, "An efficient self-organizing deep fuzzy neural network for nonlinear system modeling," *IEEE Trans. Fuzzy Syst.*, 2021.

[31] G. Li, G. Kang, Y. Zhu, Y. Wei, and Y. Yang, "Domain consensus clustering for universal domain adaptation," in *Proc. IEEE. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Virtual online, June 19 - 25 2021, pp. 9757–9766.

[32] P. Agarwal, D. P. Paudel, J.-N. Zaech, and L. Van Gool, "Unsupervised robust domain adaptation without source data," *arXiv preprint arXiv:2103.14577*, 2021.

[33] S. Yang, Y. Wang, J. van de Weijer, L. Herranz, and S. Jui, "Casting a bait for offline and online source-free domain adaptation," *arXiv preprint arXiv:2010.12427*, vol. 1, no. 2, p. 3, 2021.

[34] G. Chen, Y. Li, and X. Liu, "Transfer learning under conditional shift based on fuzzy residual," *IEEE Trans. Cybern.*, vol. 52, no. 2, pp. 960–971, 2022.

[35] F. Liu, G. Zhang, and J. Lu, "Multi-source heterogeneous unsupervised domain adaptation via fuzzy-relation neural networks," *IEEE Trans. Fuzzy Syst.*, vol. 29, no. 11, pp. 3308–3322, 2021.

[36] K. Li, J. Lu, H. Zuo, and G. Zhang, "Multi-source domain adaptation with fuzzy-rule based deep neural networks," in *Proc. IEEE. Int. Conf. Fuzzy Syst. (FUZZ-IEEE)*. Virtual Online: IEEE, July 11 - 14 2021, pp. 1–6.

[37] V. N. Vapnik, "An overview of statistical learning theory," *IEEE Trans. Neural Netw.*, vol. 10, no. 5, pp. 988–999, 1999.

[38] R. Müller, S. Kornblith, and G. Hinton, "When does label smoothing help?" in *Proc. Int. Conf. Neural Inf. Process. Syst. (NeurIPS)*, Vancouver, Canada, December 8 - 14 2019.

[39] M. Caron, P. Bojanowski, A. Joulin, and M. Douze, "Deep clustering for unsupervised learning of visual features," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Munich, Germany, September 8 - 14 2018, pp. 132–149.

[40] X. Wang, Y. Jin, M. Long, J. Wang, and M. I. Jordan, "Transferable normalization: Towards improving transferability of deep neural networks," in *Proc. Int. Conf. Neural Inf. Process. Syst. (NeurIPS)*, vol. 32, Vancouver, Canada, December 8 - 14 2019, pp. 1953–1963.

[41] K. Fatras, T. Séjourné, R. Flamary, and N. Courty, "Unbalanced minibatch optimal transport; applications to domain adaptation," in *Proc. Int. Conf. Mach. Learn. (ICML)*. Virtual Online: PMLR, July 18 - 24 2021, pp. 3186–3197.

[42] Z. Cao, M. Long, J. Wang, and M. I. Jordan, "Partial transfer learning with selective adversarial networks," in *Proc. IEEE. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Salt Lake City, Utah, USA, June 18 - 22 2018, pp. 2724–2732.

[43] Z. Cao, K. You, M. Long, J. Wang, and Q. Yang, "Learning to transfer examples for partial domain adaptation," in *Proc. IEEE. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, California, USA, June 16 - 20 2019, pp. 2985–2994.

[44] R. Xu, G. Li, J. Yang, and L. Lin, "Larger norm more transferable: An adaptive feature norm approach for unsupervised domain adaptation," in *Proc. Int. Conf. Comput. Vis. (ICCV)*, California, USA, June 16 - 20 2019, pp. 1426–1435.

[45] J. Chen, X. Wu, L. Duan, and S. Gao, "Domain adversarial reinforcement learning for partial domain adaptation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 2, pp. 539–553, 2022.

[46] K. Saito, S. Yamamoto, Y. Ushiku, and T. Harada, "Open set domain adaptation by backpropagation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Munich, Germany, September 8 - 14 2018, pp. 153–168.

[47] H. Liu, Z. Cao, M. Long, J. Wang, and Q. Yang, "Separate to adapt: Open set domain adaptation via progressive separation," in *Proc. IEEE. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, California, USA, June 16 - 20 2019, pp. 2927–2936.

[48] Z. Fang, J. Lu, F. Liu, J. Xuan, and G. Zhang, "Open set domain adaptation: Theoretical bound and algorithm," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 10, pp. 4309–4322, 2021.

[49] M. Baktashmotlagh, T. Chen, and M. Salzmann, "Learning to generate the unknowns as a remedy to the open-set domain shift," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Hawaii, USA, January 4 - 8 2022, pp. 207–216.

[50] Y. Kim, S. Hong, D. Cho, H. Park, and P. Panda, "Progressive domain adaptation from a source pre-trained model," *arXiv preprint arXiv:2007.01524*, 2020.

[51] V. K. Kurmi, V. K. Subramanian, and V. P. Namboodiri, "Domain impression: A source data free domain adaptation method," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Virtual online, January 5 - 9 2021, pp. 615–625.

[52] H. Xia, H. Zhao, and Z. Ding, "Adaptive adversarial network for source-free domain adaptation," in *Proc. IEEE. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Virtual online, June 19 - 25 2021, pp. 9010–9019.

[53] S. Yang, J. van de Weijer, L. Herranz, S. Jui *et al.*, "Exploiting the intrinsic neighborhood structure for source-free domain adaptation," in *Proc. Int. Conf. Neural Inf. Process. Syst. (NeurIPS)*, Virtual online, December 6 - 14 2021.

[54] W. Li, M. Cao, and S. Chen, "Jacobian norm for unsupervised source-free domain adaptation," *arXiv preprint arXiv:2204.03467*, 2022.

[55] Y. Luo, Z. Wang, Z. Chen, Z. Huang, and M. Baktashmotlagh, "Source-free progressive graph learning for open-set domain adaptation," *arXiv preprint arXiv:2202.06174*, 2022.

[56] Y. Ganin and V. Lempitsky, "Unsupervised domain adaptation by backpropagation," in *Proc. Int. Conf. Mach. Learn. (ICML)*. JMLR, July 6 - 11 2015, pp. 1180–1189.

[57] K. Li, J. Lu, H. Zuo, and G. Zhang, "Source-free multi-domain adaptation with generally auxiliary model training," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*. Padua, Italy: IEEE, July 18 - 23 2022, pp. 1–8.

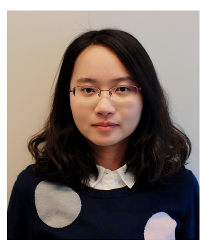
[58] H. Zhao, S. Zhang, G. Wu, J. M. Moura, J. P. Costeira, and G. J. Gordon, "Adversarial multiple source domain adaptation," in *Proc. Int. Conf. Neural Inf. Process. Syst. (NeurIPS)*, Montreal, Canada, December 2 - 8 2018, pp. 8559–8570.

[59] Y.-H. Tsai, W.-C. Hung, S. Schulter, K. Sohn, M.-H. Yang, and M. Chandraker, "Learning to adapt structured output space for semantic segmentation," in *Proc. IEEE. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Salt Lake City, Utah, USA, June 18 - 22 2018, pp. 7472–7481.

[60] T.-H. Vu, H. Jain, M. Bucher, M. Cord, and P. Pérez, "Advent: Adversarial entropy minimization for domain adaptation in semantic segmentation," in *Proc. IEEE. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, California, USA, June 16 - 20 2019, pp. 2517–2526.

Keqiuyin Li is a research associate working on an Australian research council laureate project in Decision Systems and e-Service Intelligent (DeSI) lab in the Australian Artificial Intelligence Institute (AAII) at University of Technology Sydney, Australia. Her research interest is mainly in transfer learning. She has published papers in leading journals such as IEEE TNNLS, IEEE TKDE and leading conferences such as WCCI.

Jie Lu (F'18) is an Australian Laureate Fellow, IFSA Fellow, Distinguished Professor and the Director of Australian Artificial Intelligence Institute (AAII) at the University of Technology Sydney, Australia. She received a PhD degree from Curtin University in 2000. Her main research expertise is in transfer learning, concept drift, fuzzy systems, decision support systems and recommender systems. She has been awarded 12 Australian Research Council (ARC) discovery and linkage grants and led 20 industry projects. She has published over 500 papers in IEEE Transactions and other leading journals and conferences. Many of her research results have been successfully transferred to industry. She has delivered 40 keynote speeches at international conferences, supervised 50 PhD students to completion, serves as Editor-In-Chief for Knowledge-Based Systems and received national and international awards including the IEEE Transactions on Fuzzy Systems Outstanding Paper Awards (2018 and 2022).



Hua Zuo is an Australian DECRA Fellow and Senior Lecturer with the Faculty of Engineering and Information Technology, University of Technology Sydney, Australia. She received her PhD from the University of Technology Sydney, Sydney, Australia, in 2018. Her research interests include transfer learning and fuzzy systems. She is a Member of the Decision Systems and e-Service Intelligence (DeSI) Research Laboratory at the Australian Artificial Intelligence Institute (AAII), University of Technology Sydney.

Guangquan Zhang is an Australian Research Council (ARC) QEII Fellow, Associate Professor and the Director of the Decision Systems and e-Service Intelligent (DeSI) Research Laboratory at the Australian Artificial Intelligence Institute, University of Technology Sydney, Australia. He received his PhD in applied mathematics from Curtin University, Australia, in 2001. From 1993 to 1997, he was a full Professor in the Department of Mathematics, Hebei University, China. His main research interests lie in fuzzy multi-objective, bilevel and group decision making, fuzzy measures, transfer learning and concept drift adaptation. He has published six authored monographs and over 500 papers including some 300 articles in leading international journals. He has supervised 40 PhD students to completion and mentored 15 Postdoc fellows. Prof Zhang has won ten very competitive ARC Discovery grants and many other research projects. His research has been widely applied in industries.