2023 IEEE International Conference on Robotics and Automation (ICRA 2023)
May 29 - June 2, 2023. London, UK

Multi-source Domain Adaptation for Unsupervised Road Defect Segmentation

Jongmin Yu[1], Hyeontaek Oh[2], Sebastiano Fichera[3,4], Paolo Paoletti[3,4], and Shan Luo[1,†]

***Abstract*— The performance of road defect segmentation (*a.k.a.* pixel-level road defect detection) has been improved alongside with remarkable achievement of deep learning. Those improvements need a large-scale and well-constructed dataset. However, road surface materials or designs vary from country to country, and the patterns of defects are hard to pre-define. In this paper, we propose a novel multi-source domain adaptation method to boost the performance of road defect segmentation on an unlabelled dataset. The proposed method generates multi-source ensembled labels using transferred information from models trained with multiple labelled source domains, which are utilised as supervisory signals for the unlabelled target domain. Furthermore, to reduce the domain gap between each source domain and a target domain, these domains are re-aligned with outlier repositioning to improve the defect segmentation performance. We demonstrate the effectiveness of our proposed method on Cracktree200, CRACK500, CFD, and Crack360 datasets. Experimental results show that the proposed method outperforms the existing unsupervised road defect segmentation methods and achieves competitive performance compared with recent supervised methods. The source code is publicly available on** https://github.com/andreYoo/MSDA_RDS.git**.**

I. Introduction

Defects in the road surface are a significant cause of vehicle damage and sometimes accidents, posing a threat to road traffic safety. According to the Department for Transport of United Kingdom [1], road environmental conditions, including poor or defective road surfaces, caused 12.6% of car accidents in 2020. In addition, recent studies on the links between road surface quality and carbon emissions showed that road defects cause more carbon emissions [2], [3]. The studies showed that bad surface quality increases about 2% of carbon emission and 5% of fuel consumption. In the future era of autonomous vehicles, it would be essential to develop robots that can detect road defects and repair them autonomously [4], [5].

Road defect segmentation methods can be classified into supervised learning-based and unsupervised learning-based approaches, depending on if explicit annotation is used. In general, supervised methods [6]–[12] show better performance than unsupervised methods [13]–[17] since it is possible to present a clear criterion for distinguishing defects. However, data labelling is labour-intensive, and collecting sufficient amounts of data can be problematic.

In this paper, we propose a novel Multi-source Domain Adaptation (MDA) method to boost road defect segmentation performance on an unlabelled dataset using multiple labelled datasets. To the best of the authors' knowledge, there have been no prior works on road defect segmentation that consider such a practical setting where labelled datasets from multiple sources are available. Compared with using a single labelled dataset to boost the model's performance on an unlabelled dataset, MDA is more intractable: though simply combining different sources into one source and directly employing single domain adaptation could improve the segmentation performance, but it may not perform well since images from different source domains may interfere with each other during the learning process [18], [19].

The proposed MDA first performs domain adaptation on the trained models for each of the given multiple source domains. The MDA generates highly reliable supervisory signals for the target domain by aggregating information transferred from each model and using them to improve the performance of each model on the target domain. These domains are re-aligned with outlier repositioning, reducing gaps between each source domain and the target domain. To our best knowledge, this is the first paper on the multi-source domain adaptation for road defect segmentation.

Our method is tested on Cracktree200 [20], CRACK500 [9], CFD [21], and Crack360 [22] datasets. We compare our method with various supervised methods and unsupervised methods including domain adaptation-based methods [23], [24]. The proposed method achieves 31% and 17% better performances on average than the AAE [25], the ADA [24], and the ProDA [26] methods. Those figures are the best performance among the unsupervised and domain adaptation-based methods, respectively. In addition, the proposed method also achieves competitive performance compared with some supervised methods [6], [7]. Consequently, the proposed method demonstrates that using multiple source domains can significantly improve road defect segmentation performance.

II. Related works

Road defect segmentation methods can be categorised into supervised learning-based and unsupervised learning-based approaches. The supervised learning approach can be considered as an end-to-end learning framework between

[1]Department of Engineering, King's College London, Strand, London, WC2R 2LS, United Kingdom, {jongmin.yu, shan.luo}@kcl.ac.uk.

[2]Institute for Information Technology Convergence, Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Republic of Korea, hyeontaek@kaist.ac.kr

[3]Robotiz3D, Sci-Tech Daresbury Keckwick Lane, WA4 4FS, United Kingdom, {sebastiano.fichera, paolo.paoletti}@robotiz3d.com

[4]School of Engineering, University of Liverpool, Brownlow Hill, L69 3GH, Liverpool, United Kingdom, {sebastiano.fichera, paoletti}@liverpool.ac.uk

This work was supported by the Innovate UK SMART grant "ARRES PREVENT: The World-First Autonomous Road Repair Vehicle" (10006122).

† represents the corresponding author.

input images and the corresponding ground truth labels for road defects. Therefore, the labelled dataset is essential. Based on the remarkable feature extraction performance of deep learning, supervised learning-based methods [6], [7], [9]–[12] showed outstanding performance compared with previous hand-crafted feature-based methods [4], [5], [27]. However, unfortunately, there are various causes of road defects, and the patterns of defects are hard to predict. Therefore, it is almost impossible to define the patterns of road defects before they appear. Furthermore, making a well-composed dataset for supervised learning is labour-intensive and expensive.

To overcome the dependency on a labelled dataset, various unsupervised methods for road defect segmentation have been developed [13]–[17]. The dominant way for unsupervised road defect segmentation methods is to make a model to represent normal road surfaces. Reconstruction-based [13], [14] and generative model-based methods [15]–[17] have been proposed. Those approaches assume that their training dataset is only composed of normal road surface images and expect that when a defective image is given, the output of models, *e.g.,* reconstruction error or likelihood, would be larger than normal ones [28].

The unsupervised learning-based method, free from the necessity of a labelled dataset, has a considerable advantage in real-world applications. However, various studies have still shown that the supervised learning-based methods can achieve better performance compared with the existing unsupervised learning-based methods [22], [28], [29]. These results showed that the use of both labelled and unlabelled datasets helps improve road defect segmentation performance. However, although there are many advantages in adopting labelled datasets in unsupervised learning methods, only a few approaches leverage labelled datasets to improve road defect segmentation performance on unlabelled datasets [23], [30], and those methods utilise a single labelled dataset only.

III. The Proposed Method

A. Method Overview

The general process of domain adaptation for road defect segmentation is to use a pre-trained model using a labelled source domain to generate a pseudo label for the target domain and retrain the model using it [23], [30]. The pseudo label is generated by simply comparing the output with a manually pre-defined threshold; as a result, the accuracy of the predicted pseudo label is varied depending on the threshold setting. Furthermore, a domain gap between the source and target domains also can affect the accuracy of the pseudo label. As a result, the pseudo label may contain false positives which degrade the performance of the target domain significantly.

To address this issue, we proposed the MDA. Based on the pre-trained model using the multiple source domains, the proposed MDA first generates a pseudo label called a multi-source ensemble label by aggregating outputs of the models on the target domains and conducts a supervised learning and distribution refinement using multi-source ensemble labels for the target domain. In addition, to reduce the domain gap between each source domain and the target domain, '*Domain Alignment with Outlier Repositioning*' (DAOR) is proposed. Fig. 1 shows the workflow of the MDA and the DAOR for unsupervised road defect segmentation. Detailed explanations about the MDA and the DAOR are described in further sections. We define some notations for representation efficiency, with a summarised description for those notations shown in Table I.

Notation	Name	Description
S	Source domain set	$\mathrm{S} = \{\mathrm{S}_i\}_{i=1:n^{\mathrm{S}}}$, where n^{S} is the number of the source domains. $\mathrm{S}_i = \{(x_j^{\mathrm{S}_i}, \bar{y}_j^{\mathrm{S}_i})\}_{j=1:n^{\mathrm{S}_i}}$, where n^{S_i} is the number of image x and label $\bar{y}$ pairs.
T	Target domain	$\mathrm{T} = \{x_j^{\mathrm{T}}\}_{j=1:n^{\mathrm{T}}}$, where x_j^{T} and n^{T} denote images and the number of images in the domain.
Φ^{S}	Defect segmentation model	$\Phi^{\mathrm{S}} = \{\Phi^{\mathrm{S}_i}\}_{i=1:n^{\mathrm{S}}}$. Φ^{S_i} indicates a segmentation model for S_i. Φ^{S_i} consists of an encoder f^{S_i} and a decoder g^{S_i}
f^{S_i}	Encoder for S_i	$f^{\mathrm{S}_i} : x^{\mathrm{S}_i} \rightarrow z^{\mathrm{S}_i}$
g^{S_i}	Decoder for S_i	$g^{\mathrm{S}_i} : z^{\mathrm{S}_i} \rightarrow y^{\mathrm{S}_i}$
z^*	Latent feature	z^{S_i} and z^{T} denote latent feature extracted from S_i and T, respectively.
c^*	Centroids of latent features	c^{S_i} and c^{T} indicate latent feature centroids of S_i and T, respectively.
y^*	Output of segmentation models	y^{S_i} and $y^{\mathrm{S}_i \rightarrow \mathrm{T}}$ denote outputs of the segmentation model Φ^{S_i} for the source domain S_i and the target domain T, respectively.
$\hat{y}^D$	Multi-source ensemble label	A pseudo label for the target domain T defined by Eq. 1
m^*	Masks	m^{E} and m^D are the masks to compute Eq. 2 and Eq. 3. The values of the masks are defined by $\hat{y}^D$.

TABLE I: Notations used in this paper grouped by categories.

B. Multi-source Domain Adaptation

In the beginning, the MDA leverages information transferred from multiple source domains. The MDA generates supervisory signals called multi-source ensemble label $\hat{y}^{\mathrm{T}}$ by jointly considering the multiple outputs obtained from the prediction models trained with multiple source domains. The multi-source ensemble label is designed to provide more precise and diverse supervisory signals for distinguishing whether a pixel of an image represents a defect or normal. $\hat{y}^{\mathrm{T}}$ is defined as follows:

$$\hat{y}^{\mathrm{T}} = \bigwedge_{i=1}^{n^{\mathrm{S}}} y^{\mathrm{S}_i \rightarrow \mathrm{T}} \oplus \bigvee_{i=1}^{n^{\mathrm{S}}} y^{\mathrm{S}_i \rightarrow \mathrm{T}}, \tag{1}$$

where $y^{\mathrm{S}_i \rightarrow \mathrm{T}}$ indicates the output of the models trained by S_i for the sample of target domain x^{T}: $\Phi^{\mathrm{S}_i}(x^{\mathrm{T}}) = y^{\mathrm{S}_i \rightarrow \mathrm{T}}$. $\bigwedge$ and $\bigvee$ denote sequential '*and*' and '*or*' operations, respectively. $\oplus$ represents element-wise addition for segmentation maps. Since the output of each model is taking binary value (0 is normal and 1 is defect), $\hat{y}^{\mathrm{T}}$ would be 1 (uncertain) if some models predict different results than others. If all models recognise a pixel as a normal surface or defect, $\hat{y}^{\mathrm{T}}$ would be 0 (certainly normal) or 2 (certainly defect), respectively.

After $\hat{y}^{\mathrm{T}}$ is generated, the MDA conducts supervised learning for the target domain. The supervised learning loss $\mathcal{L}^{\mathrm{seg}}_{\mathrm{S}_i \rightarrow \mathrm{T}}$ using $\hat{y}^{\mathrm{T}}$ is defined as follows:

$$\mathcal{L}^{\mathrm{seg}}_{\mathrm{S}_i \rightarrow \mathrm{T}} = -\sum_{i,j=1}^{\mathrm{W,H}} m^{\mathrm{E}}_{ij}(\hat{y}^{\mathrm{T}}_{ij} \log(y^{\mathrm{S}_i \rightarrow \mathrm{T}}_{ij}) + (2 - \hat{y}^{\mathrm{T}}_{ij}) \log(2 - y^{\mathrm{S}_i \rightarrow \mathrm{T}}_{ij})), \tag{2}$$

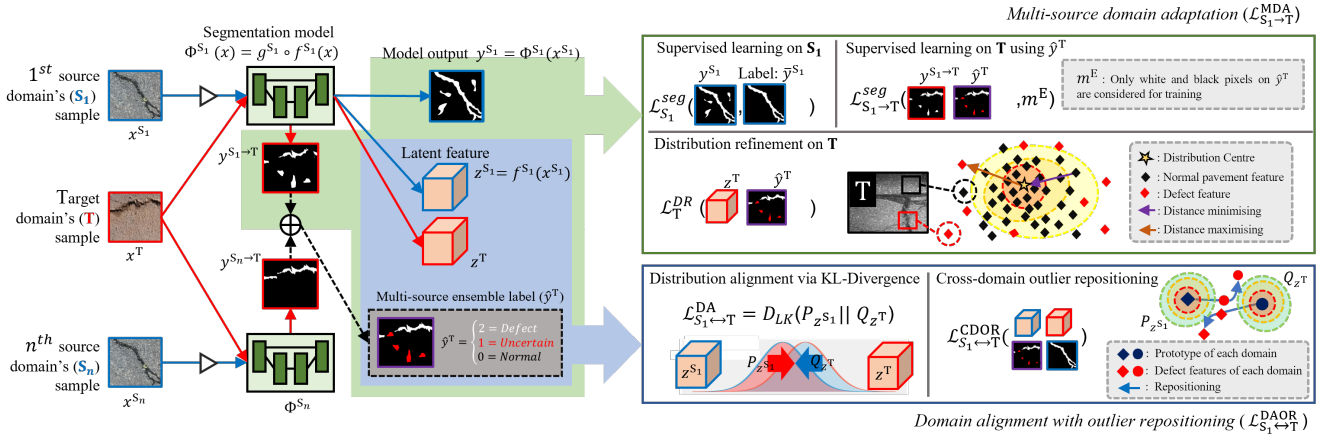

Fig. 1: Workflow of the proposed Multi-source Domain Adaptation (MDA) and the Domain Alignment with Outlier Repositioning (DAOR) for unsupervised road defect segmentation. The blue and red lines denote the process for the two source domains (S_1 and S_n) and the unlabelled target domain (T), respectively. The black lines represent the process to generate the multi-source ensemble label $\hat{y}^{\text{T}}$. The components in the green box are used in computing the MDA, and the components in the blue box are applied to compute the DAOR.

where W and H indicate the width and the height of $\hat{y}^{\text{T}}$, respectively. m^{E}_{ij} is the mask for $\mathcal{L}^{\text{seg}}_{\text{S}_i\to\text{T}}$ generated from $\hat{y}^{\text{T}}$. If $\hat{y}^{\text{T}}$ is 0 or 2, which means the results are certain, m^{E}_{ij} is set by 1; otherwise, it is set as 0.

$\mathcal{L}^{\text{seg}}_{\text{S}_i\to\text{T}}$ provides more accurate supervised learning for unlabelled target domain compared with the previous studies [23], [30] as it jointly considers the outputs of the models trained by multiple source domains and ignores uncertain prediction results (*i.e.,* $\hat{y}^{\text{T}}_{ij} = 1$) by using the mask m^{E}_{ij}. However, if too many prediction results are ignored, then the learnt features for the target domain may not provide good generalisation performance for the target domain.

To address this issue, the MDA additionally conducts the target domain distribution refinement using the latent features extracted from the target domain sample $f^{\text{S}_i}(x^{\text{T}}) = z^{\text{T}}$ and $\hat{y}^{\text{T}}$, simultaneously. We assume that latent features are generated in one-to-one correspondence for each pixel of the input image and the output segmentation map. The purpose of distribution refinement is to improve the discriminative power on the latent feature space by clustering latent features for normal road surfaces that account for most of the input sample (*i.e.,* to take uncertain results as normal since the portion of the normal is much greater than that of the defect in samples) and by pushing back the latent features for defect patterns from the centre of the latent feature distribution. The loss for the distribution refinement $\mathcal{L}^{\text{DR}}_{\text{T}}$ is defined as follows:

$$\mathcal{L}^{\text{DR}}_{\text{T}} = \sum_{i,j=1}^{\text{W,H}} ||c^{\text{T}} - z^{\text{T}}_{ij}||^{2*m^{\text{D}}_{ij}}, \tag{3}$$

where z^{T}_{ij} indicates the latent feature corresponding to the detection results y^{T}_{ij}. m^{D}_{ij} denotes the mask, and its value is decided by $\hat{y}^{\text{T}}$: The value of m^{D}_{ij} is set as -1 when $\hat{y}^{\text{T}}$ is 2, which means the pixel is certainly from defects. Otherwise, m^{D}_{ij} is set by 1. c^{T} denotes the centre of latent features of the normal surface pixels, and it is initialised by: $c^{\text{T}} = \frac{1}{\text{N}^{\text{T}}\cdot W\cdot H}\sum_t^{\text{N}^{\text{T}}}\sum_{i,j}^{\text{W,H}} z^{\text{T}}_{tij}$, where N^{T} is the number of samples in the target domain. z^{T}_{tij} denotes the latent feature extracted from t^{th} sample's (i,j) coordinate. c^{T} is updated at each training step as follow:

$$c^{\text{T}} := c^{\text{T}} + \frac{\sigma}{2}\sum_{i,j}^{\text{W,H}} \left(1 + m^{\text{D}}_{ij}\right) z^{\text{T}}_{ij}, \tag{4}$$

where σ defines a learning rate. In the above update scheme, the latent features classified by the defect patterns are automatically ignored since m^{D}_{ij} becomes -1; therefore, latent features only for normal road surface or uncertain conditions are applied to update c^{T}. Since m^{D}_{ij} for normal and uncertain signals are set as 1, *i.e.,* $(1 + m^{D}_{ij})$ becomes 2, we use $\sigma/2$ to properly calculate c^{T}.

Eq. (3) is formulated based on Mean Square Error (MSE) which is more robust to noisy labels than cross-entropy-based losses [31]. Furthermore, the false positive results on pseudo labels can be decayed when a small amount of labelled information can be provided [32], [33]. In the general case of input for road defect segmentation, the number of normal road surface pixels is much larger than that for defects. As a result, even if some false positives are included in the predicted labels, their effect would be reduced.

The total objective function of the MDA is defined by combining a supervised learning loss for each labelled source domain $\mathcal{L}^{\text{seg}}_{\text{S}_i}$ and the above two objective functions $\mathcal{L}^{\text{seg}}_{\text{S}_i\to\text{T}}$ and $\mathcal{L}^{\text{DR}}_{\text{T}}$ with a balancing weight α, as follows:

$$\mathcal{L}^{\text{MDA}}_{\text{S}_i\to T} = \mathcal{L}^{\text{seg}}_{\text{S}_i} + \mathcal{L}^{\text{seg}}_{\text{S}_i\to\text{T}} + \alpha\mathcal{L}^{\text{DR}}_{\text{T}}. \tag{5}$$

Intuitively, the MDA can provide more reliable supervisory signals for the unlabelled target domain compared with the previous domain adaptation-based methods using a single source domain. However, the MDA does not consider domain gaps which are intrinsic distributional differences of domains on latent feature spaces. If the domain gap is too large, the multi-source ensemble label will contain a lot of false positives even though it is predicted by information transferred from multiple source domains, which will greatly degrade the performance of the road defect segmentation model trained

through the MDA. To explicitly reduce the domain gaps, we propose a domain alignment with outlier repositioning.

C. Domain Alignment with Outlier Repositioning

To reduce the domain gap between each source domain S_i and the target domain T, we propose a domain alignment with outlier repositioning (DAOR). At first, the DAOR conducts the domain alignment for explicitly minimising the difference between two latent feature distributions. The objective loss for the domain alignment $\mathcal{L}^{\mathrm{DA}}_{\mathrm{S}_i \leftrightarrow \mathrm{T}}$ is formulated as follows:

$$\mathcal{L}^{\mathrm{DA}}_{\mathrm{S}_i \leftrightarrow \mathrm{T}} = \sum D_{KL}\left(P_{z^{\mathrm{S}_i}} || Q_{z^{\mathrm{T}}}\right) \tag{6}$$

where D_{KL} denotes the Kullback–Leibler (KL) divergence [34] which is a metric to measure the probabilistic difference between the two distributions. $P_{z^{\mathrm{S}_i}}$ and $Q_{z^{\mathrm{T}}}$ indicate the distributions of latent features of the source domain and the target domain, respectively. By minimising Eq. (6), the stochastic distance between the two latent feature distributions of the two domains can be minimised.

However, when we simply minimise the distance between two distributions, a set of latent features of defect pixels (which are originally positioned at the edge of each distribution) may be positioned in the centre of the newly merged domain. It may degrade the entire performance. To address this issue, DAOR conducts Cross-Domain Outlier Handling (CDOH). The CDOH is presented to cluster the latent features of normal surface pixels to each centroid and to push the defect features away from the centre of each distribution (as shown in the bottom right side of Fig. 1). The objective function of the CDOH is formulated as follows:

$$\mathcal{L}^{\mathrm{CDOH}}_{\mathrm{S}\leftrightarrow\mathrm{T}} = \sum_{i,j=1}^{\mathrm{W,H}} \left(\frac{1 - m^{\mathrm{D}}_{ij}}{||c^{\mathrm{S}_i} - z^{\mathrm{T}}_{ij}||}\right)^2 + \left(\frac{\bar{y}^{\mathrm{S}_i}_{ij}}{||c^{\mathrm{T}} - z^{\mathrm{S}_i}_{ij}||}\right)^2, \tag{7}$$

where c^{S_i} and c^{T} are the centroids of the latent features of the source and target domains, respectively. In updating c^{S_i}, latent features corresponding to the normal road surfaces are used only. When latent features related to normal road surface are given (*i.e.,* $m^{\mathrm{D}}_{ij} = 1$ or $\bar{y}^{\mathrm{S}_i}_{ij} = 0$), each term on $\mathcal{L}^{\mathrm{CDOH}}_{\mathrm{S}\leftrightarrow\mathrm{T}}$ would be 0 so that only latent features of defect patterns are considered for computing $\mathcal{L}^{\mathrm{CDOH}}_{\mathrm{S}\leftrightarrow\mathrm{T}}$.

As a result, the total objective function of the DAOR can be defined as

$$\mathcal{L}^{\mathrm{DAOR}}_{\mathrm{S}_i\leftrightarrow\mathrm{T}} = \mathcal{L}^{\mathrm{DA}}_{\mathrm{S}_i\leftrightarrow\mathrm{T}} + \mathcal{L}^{\mathrm{CDOH}}_{\mathrm{S}_i\leftrightarrow\mathrm{T}}. \tag{8}$$

D. Learning and Inference

The entire objective function is defined as follows:

$$\mathcal{L}^{\mathrm{total}}_{\mathrm{S}\rightarrow\mathrm{T}} = \sum_{i=1}^{n^{\mathrm{S}}} \mathcal{L}^{\mathrm{MDA}}_{\mathrm{S}_i\rightarrow T} + \beta\mathcal{L}^{\mathrm{DAOR}}_{\mathrm{S}_i\leftrightarrow\mathrm{T}}, \tag{9}$$

where β indicates the balancing weight for $\mathcal{L}^{\mathrm{DAOR}}_{\mathrm{S}_i\leftrightarrow\mathrm{T}}$.

The training of the proposed multi-domain adaptation method is to minimise $\mathcal{L}^{\mathrm{total}}_{\mathrm{S}\rightarrow\mathrm{T}}$ with respect to the entire segmentation models $\Phi^{\mathrm{S}} = \{\Phi^{\mathrm{S}_i}\}_{i=1:n^{\mathrm{S}}}$ for given multiple source domains S and an unlabelled target domain T, represented by

$$\arg\min_{\Phi^{\mathrm{S}}_\theta} \mathcal{L}^{\mathrm{total}}_{\mathrm{S}\rightarrow\mathrm{T}},$$

where $\Phi^{\mathrm{S}}_\theta = \{\Phi^{\mathrm{S}_i}_\theta\}_{i=1:n^{\mathrm{S}}}$ denotes the set of trainable parameters of the entire segmentation models Φ^{S} for the given multiple source domains.

After the training for multiple models is over, two strategies for generating the segmentation results on the target domain can be considered in the inference step. The model that achieves the best performance on the target domain in the training phase can be selected as a representative model for the target domain. Alternatively, the outputs of all models can be ensembled. Unless specified in the rest of the paper, we use the latter strategy.

IV. Experiment setup

Datasets: Cracktree200 [20], CRACK500 [9], CFD [21], and Crack360 [22] datasets are used to demonstrate the effectiveness of the proposed MDA and compare the performance of the proposed MDA with existing road defect segmentation methods. Cracktree200 contains 206 surface images with 800×600 resolution, which have various types of surface defects captured with various lighting conditions. CRACK500 is composed of 500 images of 2,000×1,500 resolution. CFD contains 118 images with 480×320 resolution. Crack360 contains 360 images, which consists of 260 images with 800×600 resolution and 100 images with 512×512 resolutions.

Experimental protocol: For experimental efficiency, only three datasets (*i.e.,* Cracktree200, CRACK500, and Crack360) are leveraged for the ablation studies. In ablation studies, we monitor the effect of hyper-parameters of our method and verify the effectiveness of the key components of the proposed method. For the performance comparison with existing road defect segmentation methods, we compare the performance with the result of performing domain adaptation using all datasets except the comparison target dataset. The above protocol is inspired by the multi-source domain adaptation for image segmentation [18], [19], [35].

Evaluation metrics: Three different F-measure-based metrics are employed in the evaluation: 1) the best F-measure on the data set for a fixed threshold Optimal Dataset Scale (ODS), 2) the aggregate F-measure on the data set for the best threshold on each image Optimal Image Scale (OIS), and 3) the Average Intersection over Union (AIU) [9]. In particular, AIU provides more general information about segmentation performance by taking the width information into consideration to evaluate detection results and illustrates the overall overlap extent between detection results and ground truth with various threshold settings. When the values of each metric are closer to 1, it can be interpreted as a better road defect segmentation method.

Hyper-parameter settings: We employ U-Net [10] as our backbone model for each source domain. The number of epochs and the batch size used in the experiments are 100 and 4, respectively. The two balancing weights α and β are set

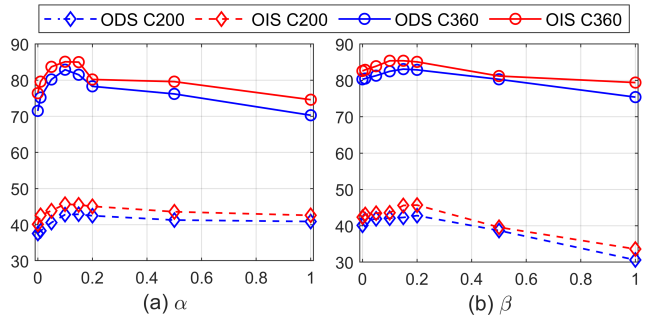

Fig. 2: Trends of ODS and OIS depending on the settings of α and β. (a) and (b) represent the trends of ODS and OIS with respect to the setting of α and β, respectively. Cracktree200 (C200) and Crack360 (C360) are employed as the unlabelled target domain.

DA method	Source	C200		C500		C360	
		ODS	OIS	ODS	OIS	ODS	OIS
Without the DAOR $\mathcal{L}_{S\leftrightarrow T}^{DAOR}$ (Eq. (8))							
Single DA	C200		-	41.4	44.7	63.2	70.3
	C500	18.6	20.1	-	-	62.6	68.3
	C360	20.5	21.7	49.6	53.5	-	-
MDA (Eq. (5))	C500+C360	40.1	42.4	-	-	-	-
	C200+C360	-	-	52.7	54.6	-	-
	C200+C500	-	-	-	-	80.3	82.6
With the DAOR $\mathcal{L}_{S\leftrightarrow T}^{DAOR}$ (Eq. (8))							
Single DA	C200		-	52.8	55.0	68.7	71.6
	C500	23.6	25.0	-	-	61.8	66.5
	C360	30.2	32.7	50.5	54.3	-	-
MDA (Eq. (5))	C500+C360	**42.8**	**45.7**	-	-	-	-
	C200+C360	-	-	**57.1**	**60.1**	-	-
	C200+C500	-	-	-	-	**82.9**	**85.1**

TABLE II: Quantitative performance analysis depending on various settings of objective functions for domain adaptation. C200, C500, and C360 represent the Cracktree200, CRACK500, and Crack360 datasets, respectively.

as 0.1 and 0.2 for the best performance based on our ablation studies. The networks are optimised by Stochastic Gradient Descent (SGD). The initial learning rate is set by 0.01, and we apply exponential learning late decay by multiplying 0.95 by the learning rate at every epoch.

V. Experimental results

A. Ablation study

We conduct an ablation study to demonstrate the effectiveness of the MDA and the DAOR. Also, we observe the performances depending on the settings of α and β to provide insights to decide α and β for the best performance.

Parameter analysis on α: α in Eq. (5) determines the weight of the distribution refinement term that diverges the defect feature from the centre of the distribution in the target domain distribution during the training process. The experimental results in Fig. 2(a) show that when α increases from 0 to 0.1, the curves of ODI and OIS increase steeply, and after that, those are gradually decreased. These trends show that distribution refinement can generally improve road segmentation detection performance. However, since there is still a false positive result on the multi-source ensemble label, it can negatively affect the performance. The best performance is achieved by α of 0.1.

Parameter analysis on β: β in Eq. (9) is the balancing weight for the DAOR term $\mathcal{L}^{DAOR}$. The results in Fig. 2(b) show that when β increases from 0 to 0.2, the curves of ODI and OIS are increased rapidly, and after that, those are also steeply decreased. The experimental results show that the DAOR helps improve road defect segmentation performance but it is very sensitive to the value of β. The best performance is achieved by β of 0.2.

Effectiveness of MDA: We compare the MDA (Eq. (5)) with the single domain adaptation approach. In the cases using the single domain adaptation approach, the DAOR is naively applied to train a model. The experimental results in Table II show that the MDA helps improve road defect segmentation performance. In contrast to the performances of the models applied to the single adaptation approach that shows great volatility depending on a source domain, the experimental results with MDA show better performance. The MDA produces the best performances of each dataset, and those are at least 6% better than the best performance of the model trained with the single domain adaptation approach. The visualisation results in Fig. 3 also justify that the proposed method achieves better accuracy and stability.

Effectiveness of DAOR: Additionally, Table II also shows the performance depending on the existence of the domain alignments with outlier repositioning term $\mathcal{L}_{S\leftrightarrow T}^{DAOR}$. The results show that when the DAOR is applied in the training step, the performances are improved obviously. The models that applied the DAOR produce an average of 7.1% better performance than those without it. These results demonstrate that the DAOR improves the road defect segmentation performance by pushing out the latent features corresponding to the defect road surfaces of each domain in reducing the domain gap.

B. Comparison with existing methods

We have compared the road defect segmentation performance of our method with various existing unsupervised [14], [21], [25] and domain adaptation-based [23], [24], [26], [36] road defect segmentation methods. Among the domain adaptation-based methods, Fan *et al.* [23], and Zhang *et al.* [26] released their source code. In addition, we implemented a gradient-based domain adaptation [36] and Adversarial Domain Adaptation (ADA) [24]. Those domain adaptation methods are applied to the U-Net [10] and evaluated for performance comparison. Also, the proposed method is compared with some supervised methods [6], [7], [9], [10]. Table III contains AIUs, OISs, and ODSs on Cracktree200, CRACK500, CFD, and Crack360 datasets.

Our method produces the state-of-the-art performance compared with unsupervised methods, including domain adaptation-based methods. Our method achieves 2.3 of AIU, 43.1 of ODS, 46.2 of OIS on Cracktree200 and 46.3 of AIU, 58.0 of ODS, and 60.2 of OIS on CRACK500. In addition, our method produces 17.4 of AIU, 69.9 of ODS, 71.5 of OIS on CFD and 19.4 of AIU, 83.7 of ODS, and 85.8 of OIS on Crack360. Those figures perform best among the unsupervised and domain adaptation-based methods. Additionally, the comparison between our method and various supervised

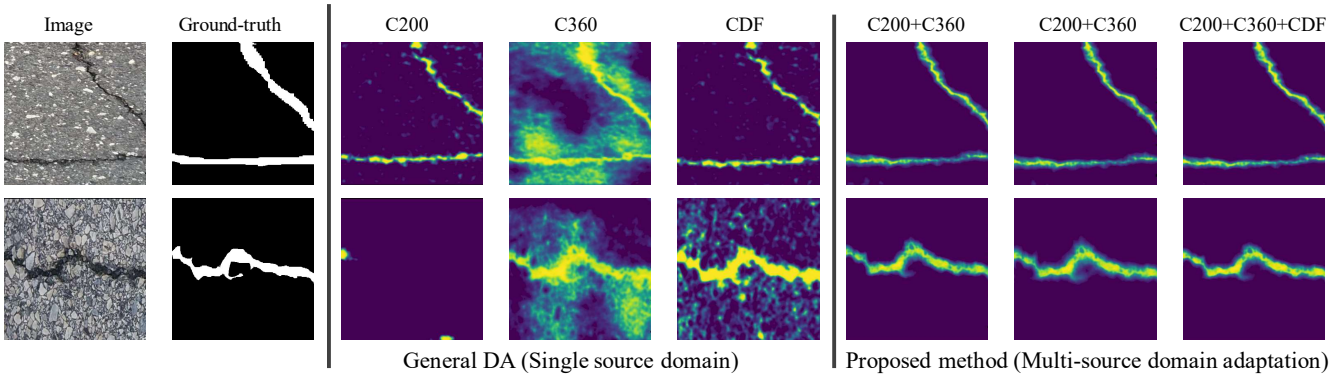

Fig. 3: Defect segmentation results of different Domain Adaptation (DA) settings. Images of CRACK500 are applied as an unlabelled target domain. Cracktree200 (C200), CDF, and Crack360 (C360) are used as the source domains. Detection results using general domain adaptation (between the third to fifth images from the left) show the images containing a lot of noise. On the other hand, the results of our method using multiple source domains (more than two) show more precise detection results.

Methods	S/DA/U	Source	Cracktree200 [20]			Source	CRACK500 [9]			Source	CFD [21]			Source	Crack360 [22]			ES (sec.)
			AIU	ODS	OIS		AIU	ODS	OIS		AIU	ODS	OIS		AIU	ODS	OIS	
RCF [6]	S	×	3.2	25.5	48.7	×	40.3	49.0	58.6	×	10.5	54.2	60.7	×	-	92.7	94.9	0.079
FCN [7]	S	×	0.8	33.4	33.3	×	37.9	51.3	57.7	×	21.0	58.5	60.9	×	-	91.2	93.6	0.114
FPHBN [9]	S	×	4.10	51.7	55.3	×	48.9	60.4	63.5	×	17.3	68.3	70.5	-	-	92.3	93.9	0.237
U-Net [10]	S	×	3.80	46.9	47.7	×	47.1	59.2	61.3	×	18.6	71.2	73.5	-	-	89.5	93.2	0.125
DeepCrack [9]	S	×	4.30	52.3	56.4	×	50.2	63.2	65.1	×	19.2	74.3	76.4	×	-	94.5	96.8	0.125
CrackForest [21]	U	×	-	8.0	8.0	×	-	19.9	19.9	×	-	10.4	10.4	×	-	-	-	0.981
AE [14]	U	×	0.9	21.4	22.1	×	27.1	42.1	45.9	×	6.1	43.2	47.2	×	9.8	58.2	59.4	**0.066**
AAE [25]	U	×	1.7	26.3	28.1	×	30.5	46.2	51.8	×	8.2	51.0	58.2	×	10.9	72.6	74.1	0.721
Ganin *et al.* [36]	DA	CRACK500	1.8	26.9	28.3	Crack360	33.0	49.3	50.7	Crack360	8.1	51.4	54.1	CRACK500	11.3	75.4	78.3	0.125
ADA [24]	DA	CRACK500	2.1	30.2	32.4	Crack360	38.2	53.5	55.0	CRACK500	10.4	53.1	58.2	CRACK500	13.2	79.4	82.6	0.125
Fan *et al.* † [23]	DA	CRACK500	1.4	20.6	21.5	Crack360	34.7	51.8	52.8	CRACK500	6.4	49.9	53.1	CDF	0.8	63.7	65.4	0.125
ProDA† [26]	DA	CRACK500	1.6	23.1	27.8	Crack360	40.1	56.3	54.9	CRACK500	13.7	64.2	65.9	CRACK500	12.4	77.5	78.9	0.125
Our method	DA	Multiple	**2.3**	**43.1**	**46.2**	Multiple	**46.3**	**58.0**	**60.2**	Multiple	**17.4**	**69.9**	**71.5**	Multiple	**19.4**	**83.7**	**85.8**	0.167

TABLE III: Quantitative comparison for the road defect segmentation performance using Cracktree200, CRACK500, CFD, and Crack360 datasets. 'S/DA/U' denotes whether a model is derived by '*Supervised*', '*Domain Adaptation*', or '*Unsupervised*' based approaches. 'Source' defines the source domain used to achieve the corresponding performance. 'ES' represents the Execution Speed (in seconds) defined by the time to process one image. "×" denotes that a source domain was not given. "-" means the results are not provided. "†" denotes the results obtained by our experiment with the official source code released by the authors. The underlined figures represent the best performance among the supervised methods. The **bolded** numbers indicate the best performance among the unsupervised methods.

methods shows that our method outperforms some supervised methods [6], [7]. However, the best performance is obtained by DeepCrack [9], showing at least 8.3% better results than our method.

Overall, the performance comparison results demonstrate that our multi-source domain adaptation can improve road defect segmentation performance. In particular, the performance gap between our method and other domain adaptation-based methods can be interpreted as an advantage of using multiple source domains. Moreover, the experimental results show that if multiple source domains can be suitably leveraged, we can achieve outstanding road defect segmentation performance, which is better than some supervised methods.

C. Computational complexity and execution speed

The computational complexity of our method will be varied depending on the complexity of the baseline method and the number of source domains. The computational complexity of our method, which is estimated based on the model complexity study of convolutional neural networks [37], is $O(n^{s}\sum_{i=1}^{d} W_i H_i D_i w_i h_i d_i)$, where n^{s} denotes the number of source domains. d is the number of convolutional layers. W_i, H_i, and D_i indicate the width, height, and depth of input data and the outputs of each convolutional layer. w_i, h_i, and d_i define the width, height, and the using kernel of each convolutional layer. In our experiments, our method takes 0.167 seconds to process one image. It is 0.042 seconds longer than that of the U-Net [10], which is the baseline model of our method. This figure is obtained by applying parallel processing, and the processing speed of the proposed method is proportional to the number of source domains used.

VI. Conclusion

A precise road defect segmentation method is desired in developing automated road repair robots. In this paper, we proposed a novel multi-source domain adaptation method for unsupervised road defect segmentation. The proposed method comprises Multi-source Domain Adaptation (MDA) and Domain Alignment with Outlier Repositioning (DAOR). The MDA conducts joint transfer learning and distribution refinement to improve discriminative power for road defect segmentation about the unlabelled target domain by using multi-source ensemble labels. The DAOR reduces domain gaps between the source and target domains. The experimental results demonstrated the effectiveness of our proposed method in boosting unsupervised road defect segmentation performance. In future work, we will deploy and test our method on an automated road repair robot.

REFERENCES

[1] Department of Transport (United Kingdom), "Road safety statistics," May 25, 2022. [Online]. Available: https://www.gov.uk/government/statistical-data-sets/reported-road-accidents-vehicles-and-casualties-tables-for-great-britain

[2] H. Wang, I. Al-Saadi, P. Lu, and A. Jasim, "Quantifying greenhouse gas emission of asphalt pavement preservation at construction and use stages using life-cycle assessment," *International Journal of Sustainable Transportation*, vol. 14, no. 1, pp. 25–34, 2020.

[3] I. Al-Saadi, H. Wang, X. Chen, P. Lu, and A. Jasim, "Multi-objective optimization of pavement preservation strategy considering agency cost and environmental impact," *International Journal of Sustainable Transportation*, vol. 15, no. 11, pp. 826–836, 2021.

[4] R. S. Lim, H. M. La, Z. Shan, and W. Sheng, "Developing a crack inspection robot for bridge maintenance," in *2011 IEEE International Conference on Robotics and Automation*. IEEE, 2011, pp. 6288–6293.

[5] P. Prasanna, K. J. Dana, N. Gucunski, B. B. Basily, H. M. La, R. S. Lim, and H. Parvardeh, "Automated crack detection on concrete bridges," *IEEE Transactions on automation science and engineering*, vol. 13, no. 2, pp. 591–599, 2014.

[6] Y. Liu, M.-M. Cheng, X. Hu, K. Wang, and X. Bai, "Richer convolutional features for edge detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 3000–3009.

[7] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431–3440.

[8] L. Pauly, D. Hogg, R. Fuentes, and H. Peel, "Deeper networks for pavement crack detection," in *Proceedings of the 34th ISARC*. IAARC, 2017, pp. 479–485.

[9] F. Yang, L. Zhang, S. Yu, D. Prokhorov, X. Mei, and H. Ling, "Feature pyramid and hierarchical boosting network for pavement crack detection," *IEEE Transactions on Intelligent Transportation Systems*, 2019.

[10] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

[11] R. Fan, M. J. Bocus, Y. Zhu, J. Jiao, L. Wang, F. Ma, S. Cheng, and M. Liu, "Road crack detection using deep convolutional neural network and adaptive thresholding," in *2019 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2019, pp. 474–479.

[12] S. Li and X. Zhao, "Convolutional neural networks-based crack detection for real concrete surface," in *Sensors and Smart Structures Technologies for Civil, Mechanical, and Aerospace Systems 2018*, vol. 10598. SPIE, 2018, pp. 955–961.

[13] A. Mujeeb, W. Dai, M. Erdt, and A. Sourin, "One class based feature learning approach for defect detection using deep autoencoders," *Advanced Engineering Informatics*, vol. 42, p. 100933, 2019.

[14] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *Journal of machine learning research*, vol. 11, no. Dec, pp. 3371–3408, 2010.

[15] Z. Gao, B. Peng, T. Li, and C. Gou, "Generative adversarial networks for road crack image segmentation," in *2019 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2019, pp. 1–8.

[16] K. Zhang, Y. Zhang, and H.-D. Cheng, "Crackgan: Pavement crack detection using partially accurate ground truths based on generative adversarial learning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 2, pp. 1306–1319, 2020.

[17] A. Sekar and V. Perumal, "Cfc-gan: Forecasting road surface crack using forecasted crack generative adversarial network," *IEEE Transactions on Intelligent Transportation Systems*, 2022.

[18] S. Zhao, B. Li, X. Yue, Y. Gu, P. Xu, R. Hu, H. Chai, and K. Keutzer, "Multi-source domain adaptation for semantic segmentation," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[19] J. He, X. Jia, S. Chen, and J. Liu, "Multi-source domain adaptation with collaborative learning for semantic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 11 008–11 017.

[20] Q. Zou, Y. Cao, Q. Li, Q. Mao, and S. Wang, "Cracktree: Automatic crack detection from pavement images," *Pattern Recognition Letters*, vol. 33, no. 3, pp. 227–238, 2012.

[21] Y. Shi, L. Cui, Z. Qi, F. Meng, and Z. Chen, "Automatic road crack detection using random structured forests," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 12, pp. 3434–3445, 2016.

[22] K. Li, B. Wang, Y. Tian, and Z. Qi, "Fast and accurate road crack detection based on adaptive cost-sensitive loss function," *IEEE Transactions on Cybernetics*, 2021.

[23] R. Fan, H. Wang, M. J. Bocus, and M. Liu, "We learn better road pothole detection: from attention aggregation to adversarial domain adaptation," in *European Conference on Computer Vision*, 2020, pp. 285–300.

[24] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, "Adversarial discriminative domain adaptation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 7167–7176.

[25] A. Makhzani, J. Shlens, N. Jaitly, I. Goodfellow, and B. Frey, "Adversarial autoencoders," *arXiv preprint arXiv:1511.05644*, 2015.

[26] P. Zhang, B. Zhang, T. Zhang, D. Chen, Y. Wang, and F. Wen, "Prototypical pseudo label denoising and target structure learning for domain adaptive semantic segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 12 414–12 424.

[27] S. Chambon, C. Gourraud, J. M. Moliard, and P. Nicolle, "Road crack extraction with adapted filtering and markov model-based segmentation: introduction and validation," 2010.

[28] J. Yu, D. Y. Kim, Y. Lee, and M. Jeon, "Unsupervised pixel-level road defect detection via adversarial image-to-frequency transform," in *2020 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2020, pp. 1708–1713.

[29] Z. Qu, C. Cao, L. Liu, and D.-Y. Zhou, "A deeply supervised convolutional neural network for pavement crack detection with multiscale feature fusion," *IEEE transactions on neural networks and learning systems*, 2021.

[30] H. Liu, C. Yang, A. Li, Y. Ge, S. Huang, X. Feng, and Z. Ruan, "Deep domain adaptation for pavement crack detection," *arXiv preprint arXiv:2111.10101*, 2021.

[31] A. Ghosh, H. Kumar, and P. S. Sastry, "Robust loss functions under label noise for deep neural networks," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 31, no. 1, 2017.

[32] L. Ruff, R. A. Vandermeulen, N. Görnitz, A. Binder, E. Müller, K.-R. Müller, and M. Kloft, "Deep semi-supervised anomaly detection," *arXiv preprint arXiv:1906.02694*, 2019.

[33] J. Yu, H. Oh, M. Kim, and J. Kim, "Normality-calibrated autoencoder for unsupervised anomaly detection on data contamination," *arXiv preprint arXiv:2110.14825*, 2021.

[34] S. Kullback and R. A. Leibler, "On information and sufficiency," *The annals of mathematical statistics*, vol. 22, no. 1, pp. 79–86, 1951.

[35] O. Tasar, Y. Tarabalka, A. Giros, P. Alliez, and S. Clerc, "Standardgan: Multi-source domain adaptation for semantic segmentation of very high resolution satellite images by data standardization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2020, pp. 192–193.

[36] Y. Ganin and V. Lempitsky, "Unsupervised domain adaptation by backpropagation," in *International conference on machine learning*. PMLR, 2015, pp. 1180–1189.

[37] K. He and J. Sun, "Convolutional neural networks at constrained time cost," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 5353–5360.