

Minimizing Human Labeling in Training Deep Models for Pedestrian Intention Prediction

Muhammad Naveed Riaz^{ID}, Maciej Wielgosz^{ID}, and Antonio M. López^{ID}, *Member, IEEE*

Abstract—Accurately predicting whether pedestrians will cross in front of an autonomous vehicle is essential for ensuring safe and comfortable maneuvers. However, developing models for this task remains challenging due to the limited availability of diverse datasets containing both crossing (C) and non-crossing (NC) scenarios. Therefore, we propose a procedure that leverages synthetic videos with C/NC labels and an untrained model whose architecture is designed for C/NC prediction to automatically produce C/NC labels for a set of real-world videos. Thus, this procedure performs a synth-to-real unsupervised domain adaptation for C/NC prediction, so we term it S2R-UDA-CP. To assess the effectiveness of S2R-UDA-CP in self-labeling, we utilize two state-of-the-art models, PedGNN and ST-CrossingPose, and we rely on the publicly-available PedSynth dataset, which consists of synthetic videos with C/NC labels. Notably, once the real-world videos are self-labeled, they can be used to train models different from those used in S2R-UDA-CP. These models are designed to operate onboard a vehicle, whereas S2R-UDA-CP is an offline procedure. To evaluate the quality of the C/NC labels generated by S2R-UDA-CP, we also employ PedGraph+ (another literature referent) as it is not used in S2R-UDA-CP. Overall, the results show that training models to predict C/NC using videos labeled by S2R-UDA-CP achieves performance even better than models trained on human-labeled data. Our study also highlights different discrepancies between automatic and human labeling. To the best of our knowledge, this is the first study to evaluate synth-to-real self-labeling for C/NC prediction.

Index Terms—Vision-based autonomous driving, pedestrian intention prediction, semi-supervised learning (SSL), unsupervised domain adaptation (UDA).

I. INTRODUCTION

AS EVIDENCED in an early Google self-driving car report [29], 10% of their self-driving malfunctions on streets were due to incorrect behavior predictions from other road users, including pedestrians. In fact, it is important to know whether pedestrians plan to cross in front of the self-driving vehicle to perform safe and comfortable maneuvers (e.g., see Figure 1). However, creating fast and accurate models to perform such a prediction from sequential images is challenging [3], [6], [16], [17], [21], [26], [34], [47], [52], [53], [54], [56]. One of the factors that makes it difficult is the lack of datasets with diverse crossing (C) and non-crossing (NC) scenarios. Currently, two datasets, JAAD [33] and PIE [32], are widely used to benchmark such prediction models. In these datasets, the core ground truth consists of manually provided labels indicating if pedestrians are crossing or will cross in front of the ego vehicle. This human labeling process tends to be time-consuming and subjective.

In fact, the course of human data labeling inherently poses challenges to various computer vision tasks that are at the core of many autonomous driving systems. To address this challenge, self-labeling procedures have been developed for tasks such as object detection [31] and semantic segmentation [41]. An effective approach involves using synthetic images with automatically generated labels as initial training data for the respective visual models. Subsequently, an incremental procedure retrains these models using both the synthetic data and real-world images, where the latter are self-labeled by successive versions of the models, until some convergence criteria is fulfilled. These procedures fall into the realm of *synth-to-real unsupervised domain adaptation* (S2R-UDA). The goal is to develop models trained without manual intervention that perform as well as or better than those trained with manually labeled images.

With this aim, we propose an S2R-UDA procedure inspired by *semi-supervised learning* (SSL) [44], [45] to automatically assign C/NC labels to real-world videos. For brevity, we refer to it as S2R-UDA-CP. To our knowledge, this is the first application of synth-to-real self-labeling to the task of predicting C/NC intentions. Unlike object detection [31], semantic segmentation [41], and monocular depth estimation [19], where S2R-UDA is applied to still images, pedestrian C/NC

Received 26 July 2024; revised 15 March 2025 and 23 April 2025; accepted 23 April 2025. Date of publication 26 May 2025; date of current version 16 September 2025. The work of Muhammad Naveed Riaz was supported in part by MCIN/AEI/10.13039/501100011033 for the development of PedSynth, PedGNN, and S2R-UDA-CP under Spanish Grant PID2020-115734RB-C21; in part by the FI Fellowship Application AGAUR through the Secretaria d'Universitats i Recerca of the Generalitat de Catalunya and the Fons Social Europeu for his Ph.D. Research under Grant 2021 FI-SDUR 00281; and in part by the Departament de Recerca i Universitats of the Generalitat de Catalunya during the FI Fellowship Application by SGR 2021 ADAS under Grant 2021 SGR 01621. The work of Maciej Wielgosz was supported in part by European Union's Horizon 2020 Research and Innovation Programme (Tecniospring Industry) under Marie Skłodowska-Curie Grant 801342 and in part by the Government of Catalonia's Agency for Business Competitiveness (ACCIO) for the development of AdveRsarial Cases for AutoNymous vEhicles (ARCANE) key to later develop the PedSynth Dataset. The work of Antonio M. López was supported in part by MCIN/AEI/10.13039/501100011033 for the development of PedSynth, PedGNN, and S2R-UDA-CP under Spanish Grant PID2020-115734RB-C21; in part by the Departament de Recerca i Universitats of the Generalitat de Catalunya during the FI Fellowship Application by SGR 2021 ADAS under Grant 2021 SGR 01621; and in part by ICREA under the ICREA Academia Program for his general research activities. The Associate Editor for this article was M. Brackstone. (*Corresponding author: Muhammad Naveed Riaz.*)

Muhammad Naveed Riaz and Antonio M. López are with the Department Ciències de la Computació and the Computer Vision Center (CVC), Universitat Autònoma de Barcelona (UAB), 08193 Barcelona, Spain (e-mail: Muhammad.Naveed@uab.cat).

Maciej Wielgosz is with the Norwegian Institute of Bioeconomy (NIBIO), 1433 Ås, Norway.

Digital Object Identifier 10.1109/TITS.2025.3565667



Fig. 1. Left: a distracted pedestrian intends to cross the road through a non-zebra crossing while listening to music, as a vehicle approaches. Right: an image captured by the vehicle's onboard camera. Autonomous vehicles must detect the pedestrian's intention to cross before it occurs to execute a safe and comfortable braking, similar to how a human driver would react.

intentions must be labeled accurately in terms of their start and end frames in a video sequence. This is challenging because there is inherent uncertainty in determining those frames. For instance, a human labeler can navigate forward and backward through a video to review a pedestrian C/NC action. The labeler may determine that it is a C action. However, suppose the first frame labeled as C is set too early relative to the actual onset of the C action. In that case, this training sample may introduce causal confusion during the training of C/NC prediction models, as there might not yet be a clear reason in the initial frames to predict the C label. Conversely, if the first frame labeled as C is set too late, the observations to predict the C action may be too short to be an effective training sample. A similar reasoning can be drawn for an NC case. This uncertainty can be further increased if different labelers follow inconsistent rules due to ambiguous labeling instructions. Implementing an automatic labeling protocol is expected to improve consistency in C/NC labeling. Moreover, we believe that programmatically generated synthetic C/NC samples can serve as a powerful prior to achieve the desired consistency. Thus, this paper presents:

1) The S2R-UDA-CP procedure for C/NC self-labeling of real-world videos. It takes as input a dataset of synthetic videos with C/NC labels, $\mathcal{V}^{SL}$, an untrained deep model designed for C/NC prediction, $\mathcal{M}$, real-world videos to be labeled, $\mathcal{V}^{RU}$, and a small set of videos with human-provided C/NC labels, $\mathcal{V}^{RL}$. Unlike traditional SSL procedures, the latter is not included in the training set. Instead, it is used as a validation set to establish a criterion for stopping the self-labeling process and set hyper-parameters. S2R-UDA-CP returns $\mathcal{V}^{RL}$, which is $\mathcal{V}^{RU}$ with added C/NC pseudo-labels. Moreover, $\mathcal{M}$ becomes trained for C/NC prediction, $\mathcal{M}^*$, and its performance computed on $\mathcal{V}^{RL}$, p . Thus, formally speaking $\text{S2R-UDA-CP}(\mathcal{V}^{SL}, \mathcal{V}^{RL}, \mathcal{V}^{RU}, \mathcal{M}) \rightarrow \langle \mathcal{V}^{RL}, \mathcal{M}^*, p \rangle$.

2) Currently, the only publicly available dataset of synthetic videos with C/NC labels we identified was our PedSynth [36]. Consequently, we use it to demonstrate the effectiveness of S2R-UDA-CP. For real-world videos, following common practice in the literature, we utilize the PIE dataset [32]. As untrained models for pedestrian C/NC prediction, we employ two from the state-of-the-art that are based on pedestrian pose estimation: ST-CrossingPose [53] and our PedGNN [36].

3) The self-labeled videos can be used to train models different from those employed in S2R-UDA-CP. Thus, to

further evaluate the effectiveness of the C/NC *pseudo-labels*, we employ another state-of-the-art reference, PedGraph+ [6], which is not used in the S2R-UDA-CP procedure.

4) The quantitative results on PIE dataset demonstrate that S2R-UDA-CP achieves competitive self-labeling performance, even surpassing human labeling.

5) Through qualitative examples, we also highlight different discrepancies between automatic and human labeling.

The remainder of the paper is organized as follows. Section II summarizes the main related works. Section III presents our S2R-UDA-CP proposal. In Section IV, we outline our experimental protocol, including the corresponding results and discussion. Finally, Section V summarizes the main outcomes of this work and highlights potential future work.

II. RELATED WORK

A. Pedestrian Intention Prediction (PIP)

Pioneering approaches cast C/NC prediction as a trajectory prediction problem, which requires the explicit estimation of the future location, speed, and acceleration of the observed pedestrians [24], [40]. In practice, the corresponding dynamic models were difficult to adjust and required to extract the silhouette of the pedestrians, dense depth, and dense optical flow with ego-motion compensation. On the other hand, Schneemann and Heinemann [39] concluded that pedestrians' posture and body movement are essential to making faster C/NC predictions. Consequently, methods relying on the temporal evolution of pedestrian skeletons gained popularity, especially due to the increasing accuracy of the deep models adjusting them in 2D images, *e.g.*, see [9] (becoming OpenPose [8]) and [15] (becoming AlphaPose [14]). For example, a sequence of pedestrian skeletons was used as input for classical machine learning models, such as a random forest C/NC classifier [16], as well as for more computationally intensive yet more accurate deep learning models, including PedGraph [7] and ST-CrossingPose [53], both based on Graph Convolutional Networks (GNNs). ST-CrossingPose demonstrated competitive performance on JAAD while analyzing factors such as pedestrian size, time to event, and occlusion. PedGraph was further developed into PedGraph+ [6] by incorporating the ego-vehicle's speed into the C/NC prediction, positioning PedGraph+ as one of the state-of-the-art models.

Skeleton extraction and C/NC prediction have also been tackled as a joint multitask problem in [35]. Semantic and contextual information is also considered in different works. In [26], a deep model based on recurrent neural networks (RNN) and attention modules is proposed, which takes as input the speed of the ego-vehicle, the bounding box (BB), the skeleton and the local context (RGB crops) of pedestrians. In [51], RNNs and attention modules are used to train two intermediate deep architectures whose output is fused (mid/late fusion) to provide C/NC predictions. One architecture considers the speed of the ego-vehicle, the BB, and the skeleton of pedestrians. The other considers scene semantic segmentation.

In [34], a dual encoding strategy was proposed to process different input-data modalities both independently and jointly, enabling the extraction of shared and unique features

to represent pedestrian intentions. In [52], the concepts of intention-driven actions and actions-informed intentions are introduced, where multimodal data is used to predict pedestrian actions, which are then employed to detect intentions. In [21], input-data modalities are categorized as visual or non-visual, with each category processed separately and combined through a late-fusion C/NC prediction. While multimodal approaches have enhanced prediction reliability, they often prioritize adding new data types and designing fusion modules, overlooking challenges such as feature redundancy and reduced computational efficiency. Other methods, such as PIT [56] and TReP [54], integrate deep Transformers to improve performance at the cost of increased computational time.

In this paper: Our S2R-UDA-CP procedure requires a (untrained) model capable of per-frame C/NC prediction. Given that our PedGNN model [36] is lightweight, fast, and achieves state-of-the-art performance on standard datasets, we employ it within S2R-UDA-CP. To demonstrate that S2R-UDA-CP is compatible with other models as well, we also utilize ST-CrossingPose [53], as it can similarly provide per-frame C/NC predictions. To evaluate the effectiveness of the pseudo-labels generated by S2R-UDA-CP, we rely on a state-of-the-art model not involved in this procedure—specifically, PedGraph+, which serves as a benchmark capable of predicting C/NC 1 to 2 seconds ahead (Time-to-Event).

B. Synthetic Datasets Focusing on C/NC Prediction

C/NC prediction research started with relatively small datasets [16], [40]. Fortunately, larger and naturalistic datasets such as JAAD [33] and PIE [32] appeared progressively, helping to accelerate this research. Using synthetic data to support C/NC prediction research was also a matter of time. Onboard pedestrian detection was one of the first tasks for which a model was trained on synthetic images to perform later in real-world images; this was done more than a decade ago [30]. Since then, there have been many works that use synthetic data to support the training of perception models or performing simulations [5], [12], [18], [37], [38], [42], [50].

Focusing on pedestrians, synthetic data has been mainly generated and used for the tasks of detection and tracking either onboard or from static infrastructure locations [10], [13], [22], [25], [43]; where the required ground truth for each pedestrian consists of a 2D/3D BB, pixel-level segmentation and depth, an ID, and, eventually, a body skeleton. In addition to this kind of ground truth, for collecting samples to develop C/NC prediction models, we must control pedestrian behavior in the simulator, *e.g.*, to force C/NC situations as we wish, and we must label each pedestrian track accordingly as in Figure 2. Recent attempts to do so [1], [2] rely on the CARLA simulator [12]. In [1], the CP2A dataset was introduced with 220K video clips with per-frame C/NC labels, where 25% of the clips contain crossing (C) examples. In [2], the Virtual-Pedcross-4667 dataset was introduced with 4,667 video clips specially prepared to cover a variety of weather and lighting conditions and per-frame C/NC labels, where 61% of the clips contain crossing (C) examples. On the other hand, these



Fig. 2. Summary of two videos from PedSynth. Top rows: a pedestrian crosses the road perpendicularly to the ego-vehicle moving direction. Bottom rows: a pedestrian changes the intention of crossing the road at mid-lane. In both examples, the pedestrians enter the road at locations that are not enabled for crossing. GT stands for ground truth (*i.e.*, C/NC label).

datasets lack some corner cases like the one shown as the bottom example in Figure 2. Overall, the experiments shown in [1] and [2] encourage using synthetic data to train C/NC prediction models.

In this paper: S2R-UDA-CP requires a set of synthetic videos with C/NC labels. Given that our PedSynth dataset [36] has proven to be an excellent complement to human-labeled datasets such as PIE and JAAD, we employ it in S2R-UDA-CP. Moreover, CP2A and Virtual-Pedcross-4667 are not publicly available, whereas PedSynth is.

C. Unsupervised Domain Adaptation

The use of automatically labeled synthetic data to train deep models that need to work with real-world unlabeled data falls within the realm of unsupervised domain adaptation (UDA) [11], [48], [49], [55]. In this context, UDA proposals have mainly focused on visual tasks involving isolated images, such as object detection [31], semantic segmentation [41], monocular depth estimation [19], and animal pose estimation [27]. Such UDA proposals may focus on adapting specific deep models to work on different domains (*e.g.*, alignment of feature space [28], online self-labeling and model weights adaptation [23]) or they may focus on self-labeling data coming from new domains using a semi-supervised learning (SSL) procedure (*e.g.*, co-training is used in [46] for object detection and in [20] for semantic segmentation). A primary advantage of the latter is that the deep model used to support self-labeling is treated as a close box and the produced pseudo-labels can be used to train future (eventually improved) deep models even based on quite different architectures. Analogously, since it is treated as a close box, the model used for self-labeling the data can be replaced by future (eventually improved) deep models too, *i.e.*, using the same self-labeling procedure. On the other hand, in general terms, one must pay attention to

self-labeling error drifts when using SSL procedures [44], [45]. However, an advantage of using an SSL-inspired procedure to address synth-to-real UDA is that one can leverage a relatively large amount of synthetic labeled data, which acts as strong regularize provided these data are of sufficient quality. Conversely, in traditional SSL settings (without domain shift), a relatively small amount of labeled data is assumed.

In this paper: Building on the complementarity of PedSynth and human-labeled data we draw in [36], here we go beyond by introducing S2R-UDA-CP, an SSL-inspired self-labeling procedure to automatically provide the C/NC labels on a real-world training set, *i.e.*, avoiding human labeling. It is important to note that we are not labeling isolated images. Instead, from a large video, we must identify continuous segments (start and end frames) that display relevant C/NC events.

III. METHODS

This paper focuses on the use of synthetic videos with C/NC labels to perform an SSL-inspired self-labeling procedure on real-world videos. As mentioned, this is a type of *synth-to-real unsupervised domain adaptation* (S2R-UDA) applied to C/NC prediction; thus, we call it S2R-UDA-CP. A high-level conceptualization of S2R-UDA-CP is presented in Algorithm 1, which we detail in the rest of this section.

Inputs. Our self-labeling procedure requires a set of synthetic videos with C/NC labels, we call it $\mathcal{V}^{SL}$. This set is the starting point for self-labeling real-world videos. Thus, $\mathcal{V}^{SL}$ is critical for the convergence of the underlying domain adaptation and so for the production of accurate C/NC labels in real-world videos. For instance, as we will see through our experiments (Sect. IV), PedSynth allows us to complete the task.

We term $\mathcal{V}^{RU}$ the real-world videos to be self-labeled. For instance, we utilize the PIE training set while disregarding its C/NC labels. Self-labeling implies processing videos to locate segments with C/NC cases, *i.e.*, to identify their start-end frames. It is worth noting that both $\mathcal{V}^{SL}$ and $\mathcal{V}^{RU}$ come with pedestrian tracks (2D bounding box and ID), thus, start-end frames of a C/NC case are per pedestrian track. Eventually, we can designate one pedestrian track with a C label and another with an NC label within the same frame. Thus, the C/NC self-labeling task differs from simply classifying full short video clips with C/NC/other labels. The hard task is to automatically identify the start-end frames of a C/NC case, if this is provided by hand (*e.g.*, in the form of a short video clip), then the C/NC label could be added effortlessly. In the case of $\mathcal{V}^{SL}$, pedestrian tracks are automatically generated. For $\mathcal{V}^{RU}$, these tracks can be obtained by applying state-of-the-art pedestrian detection and tracking procedures [4] in offline videos, or they can be provided manually. In any case, we need to add the appropriate C/NC labels afterward.

In practice, to develop a deep model, we need data to perform quantitative evaluations; from hyperparameter tuning at training time to regression tests in the software engineering sense. Thus, some manually labeled data is expected to be available. Using procedures such as our S2R-UDA-CP, we can maximize the use of the manually labeled data for model

Algorithm 1 S2R-UDA-CP (See Main Text for details)

Inputs :

$\mathcal{V}^{SL}$: Synthetic videos with C/NC labels (*e.g.*, PedSynth).
 $\mathcal{V}^{RL}$: Real videos with labels (*e.g.*, PIE validation set).
 $\mathcal{V}^{RU}$: Real unlabeled videos (*e.g.*, PIE training set w/o labels).
 $\mathcal{M}$: Untrained deep model for C/NC prediction (*e.g.*, PedGNN).
 $\mathcal{H}^{\mathcal{M}}$: Deep model training hyper-parameters.
 $\mathcal{H}^{SL}$: Self-labeling hyper-parameters.
 T_{max} : Maximum number of cycles.

Comments :

% # denotes number of C/NC cases (w or w/o label).
 SSL assumption: $\#\mathcal{V}^{RL} \ll \#\mathcal{V}^{RU}$.
 Usual in S2R-UDA literature: $\#\mathcal{V}^{SL} \geq \#\mathcal{V}^{RU}$.

Outputs :

$\mathcal{V}^{RL}$: $\mathcal{V}^{RU}$ with C/NC pseudo-labels.
 The input model $\mathcal{M}$ trained for C/NC prediction.
 p : Performance of $\mathcal{M}$ in $\mathcal{V}^{RL}$.

Self-Labeling:

```
% Factor out data computations common to all cycles.
PreProcessing( $\mathcal{V}^{SL}$ ,  $\mathcal{V}^{RU}$ ,  $\mathcal{V}^{RL}$ )
% The initial model is trained with synthetic data only.
 $\mathcal{M}_0$ ,  $p_0 \leftarrow \text{TrainInitialModel}(\mathcal{V}^{SL}, \mathcal{H}^{\mathcal{M}}, \mathcal{V}^{RL})$ 
 $\mathcal{V}_t^{RL} \leftarrow \emptyset$ ;  $\text{SelfTrain} \leftarrow \text{True}$ ;  $t \leftarrow 0$ 
while  $\text{SelfTrain}$  and  $t < T_{max}$  do
    % Obtain C/NC pseudo-labels by applying the
    current
    % C/NC prediction model to unlabeled real-world
    videos.
     $\mathcal{V}_t^{RL} \leftarrow \text{GeneratePseudoLabels}(\mathcal{M}_t, \mathcal{V}^{RU}, \mathcal{H}^{SL})$ 
    % Retrain the C/NC prediction model by mixed
    batches of
    % labeled synthetic videos and self-label real-world
    videos.
     $\mathcal{M}_{t+1}$ ,  $p_{t+1} \leftarrow \text{TrainModel}(\mathcal{V}^{SL}, \mathcal{V}_t^{RL}, \mathcal{H}^{\mathcal{M}}, \mathcal{H}^{SL}, \mathcal{V}^{RL})$ 
    % Use a labeled real-world validation set to decide
    % if the C/NC prediction model keeps improving.
     $\text{SelfTrain} \leftarrow \text{PerformanceIncreases}(p_t, p_{t+1}, \mathcal{H}^{SL})$ 
    if  $\text{SelfTrain}$  then
        |  $t \leftarrow t + 1$ 
    end
end
return  $\mathcal{V}_t^{RL}$ ;  $\mathcal{M}_t$ ;  $p_t$ 
```

assessment tasks since training data comes from self-labeling. For instance, in machine learning terms, part of the manually labeled data can be used as the validation set (helping in hyperparameter tuning) and the rest as the testing set. This is what we do in this paper since our experiments will utilize the standard PIE validation and testing sets. S2R-UDA-CP can use a C/NC human-labeled validation set to define a stopping criterion in this context. We term such a set as $\mathcal{V}^{RL}$.

In addition to the datasets used in S2R-UDA-CP as input (*i.e.*, $\mathcal{V}^{SL}$, $\mathcal{V}^{RU}$ and $\mathcal{V}^{RL}$), we need a deep architecture trainable to perform the C/NC prediction task on video streams. We term

it as $\mathcal{M}$. At input time, $\mathcal{M}$ is not trained for the task. It can be any proposal from the literature with reasonable performance. For instance, to show the effectiveness of S2R-UDA-CP, in this paper we utilize ST-CrossingPose and our PedGNN. They provide state-of-the-art performance in C/NC prediction by processing video streams using a temporal sliding window technique. This allows us to know the starting and ending frames of the C/NC cases.

As inputs to S2R-UDA-CP, we also have $\mathcal{H}^{\mathcal{M}}$ and $\mathcal{H}^{SI}$. The former refers to the usual hyper-parameters involved in training deep models (learning rate protocol and settings, number of epochs, mini-batch size, etc.) and model-specific parameters (e.g., the number of consecutive frames, N_F , considered to gather input information for the model). In other words, these hyper-parameters are not specific for S2R-UDA-CP but for the training of $\mathcal{M}$. The latter refers to hyper-parameters that are specific for S2R-UDA-CP. For instance, the balance between synthetic and real-world self-labeled data (β_{SR}) used in each mini-batch, a threshold (τ_v) on model performance improvement according to evaluations on the validation set, and a parameter used in a post-self-labeling step (N_W). We will refer to these parameters in the functions using them, explained in the rest of this section.

Finally, we can also set a maximum number of cycles, T_{max} , to stop S2R-UDA-CP.

Outputs. The main output of S2R-UDA-CP is the set of training videos, $\mathcal{V}^{RU}$, augmented with C/NC pseudo-labels. We term the new set as $\mathcal{V}^{RL}$. In addition, the procedure can output the model $\mathcal{M}$ used to produce $\mathcal{V}^{RL}$ and its performance on the validation set $\mathcal{V}^{RL}$. To avoid biases due to different amounts of C and NC labels, we use the standard F1-score classification metric.

PreProcessing($\mathcal{V}^{SL}, \mathcal{V}^{RU}, \mathcal{V}^{RL}$). The execution of a C/NC prediction model may require the computation of input features for each pedestrian track. If the model is applied in offline tasks that imply several passes over the same datasets, we can compute such features once and use them in all passes. For instance, ST-CrossingPose and PedGNN models utilize per-frame pedestrian skeletons as input. Then, for a procedure such as S2R-UDA-CP, we can compute all the skeletons of the datasets of interest, $\mathcal{V}^{SL}$, $\mathcal{V}^{RU}$ and $\mathcal{V}^{RL}$, as part of this function and use them each time the C/NC prediction model is applied to these datasets.

TrainInitialModel($\mathcal{V}^{SL}, \mathcal{H}^{\mathcal{M}}, \mathcal{V}^{RL}$): $\mathcal{M}, p$. Initially, we lack C/NC labels from real-world videos for training. Therefore, we train an initial model, $\mathcal{M}$ given $\mathcal{H}^{\mathcal{M}}$, using the synthetic data $\mathcal{V}^{SL}$. The performance of $\mathcal{M}$ on $\mathcal{V}^{RL}$ is computed using F1-score and informed as p .

GeneratePseudoLabels($\mathcal{M}, \mathcal{V}^{RU}, \mathcal{H}^{SI}$): $\mathcal{V}^{RL}$. This function uses $\mathcal{M}$ to generate C/NC pseudo-labels for $\mathcal{V}^{RU}$, which we denote as $\mathcal{V}^{RL}$. It is worth noting that $\mathcal{V}^{RU}$ consists of one or multiple videos. If there are several videos, we apply the same processing iterating over each one. For instance, this is the case of the PIE training set. As mentioned before, $\mathcal{M}$ predicts C/NC cases by analyzing the current and previous frames following a temporal sliding window technique. Since this self-labeling is done during training, we can consider previous and future frames. We take advantage of this by applying

a post-processing step to remove spurious C/NC labels in each pedestrian track. In particular, we consider a time window of N_W frames to do so, where $N_W \in \mathcal{H}^{SI}$, $N_W > 0$. Self-labeling errors may be common in the initial cycles of S2R-UDA-CP as $\mathcal{M}$ transitions from the synthetic to the real domain by considering real-world pseudo-labels, which are expected to improve gradually throughout the cycles. In this type of incremental procedure, error drifts can appear. However, the synthetic data (e.g., PedSynth) is expected to act as a strong regularizer during the mixed training of $\mathcal{M}$ (TrainModel() function) and we have not observed such potential drifts in practice. Additionally, we do not carry over the generated pseudo-labels from cycle to cycle, which helps to mitigate potential error drifts.

TrainModel($\mathcal{V}^{SL}, \mathcal{V}^{RL}, \mathcal{H}^{\mathcal{M}}, \mathcal{H}^{SI}, \mathcal{V}^{RL}$): $\mathcal{M}, p$. It trains $\mathcal{M}$ given $\mathcal{H}^{\mathcal{M}}$, using $\mathcal{V}^{SL}$ and the self-labeled set $\mathcal{V}^{RL}$. C/NC cases from $\mathcal{V}^{SL}$ and $\mathcal{V}^{RL}$ are mixed as training samples at the mini-batch level during the training of $\mathcal{M}$. A parameter β_{SR} determines the percentage of synthetic samples used in the mini-batch, where $\beta_{SR} \in \mathcal{H}^{SI}$, $|\beta_{SR}| \leq 1$. The performance of $\mathcal{M}$ on the validation set $\mathcal{V}^{RL}$ is computed using F1-score and reported as p .

PerformanceIncreases($p_c, p_n, \mathcal{H}^{SI}$): **Boolean**. This function returns *True* if $p_n - p_c > \tau_v$, and *False* otherwise; where $\tau_v \in \mathcal{H}^{SI}$, $\tau_v > 0$. Note that p_n is supposed to be a new performance value and p_c the current one. Thus, we check if the new performance significantly improves the current one.

Hyper-parameter tuning. To determine the values of the hyper-parameters in $\mathcal{H}^{\mathcal{M}}$ and $\mathcal{H}^{SI}$, we begin with $\mathcal{H}^{\mathcal{M}}$. We can invoke S2R-UDA-CP by setting $T_{max} = 0$, which allows us to train $\mathcal{M}$ using only the synthetic data $\mathcal{V}^{SL}$ given $\mathcal{H}^{\mathcal{M}}$. By empirically exploring the suitable range of values for the variables in $\mathcal{H}^{\mathcal{M}}$ while observing the corresponding returned performances p , we can establish a suitable $\mathcal{H}^{\mathcal{M}}$. Then, with a fixed $\mathcal{H}^{\mathcal{M}}$, we can follow the same exploratory approach for the variables in $\mathcal{H}^{SI}$. We can use $T_{max} = 1$ so that we run one self-labeling cycle. As mentioned before, we remark that we use F1-score to assess C/NC prediction performance.

IV. EXPERIMENTAL RESULTS

A. Datasets, Evaluation Metrics, and Self-Labeling Models

The standard real-world datasets for C/NC prediction are JAAD [33] and PIE [32]. However, JAAD consists of short videos showing a pedestrian labeled as crossing (C) or non-crossing (NC). Thus, C/NC labels are implicit in the videos, there is no need to find start-end sequences of C/NC pedestrians, which is the key task for a C/NC self-labeling procedure. Thus, this paper focuses on PIE since it presents video sequences more *in the wild*. For self-labeling with S2R-UDA-CP, we use PedSynth as our synthetic dataset for two reasons. Firstly, it is the only publicly available synthetic dataset with C/NC labels at this time. Secondly, as shown in [36], it combines effectively with real-world datasets containing C/NC labels. Tables I and II summarize the main features of PIE and PedSynth datasets. We use the standard Train/Val./Test split of the PIE dataset. We could use the full PedSynth for

TABLE I
RELEVANT FEATURES OF PIE AND PEDSYNTH

Feature	PIE	PedSynth
Videos with C/NC labels	55	947
Length per video (s)	~ 600	~ 20
Frames per second (fps)	30	30
Frame resolution (pix)	1920×1080	1600×600
Frames with C/NC labels	$\sim 293K$	$\sim 398K$
Pedestrian skeleton	no	yes
Pedestrian tracks	yes	yes

TABLE II

LET'S A *sample* BE A PARTICULAR PEDESTRIAN COMPLETING A C/NC SEQUENCE. THUS, DIFFERENT SAMPLES CAN OVERLAP IN THE SAME FRAME. LET N_F^s BE THE NUMBER OF LABELED FRAMES OF THE C/NC SEQUENCE OF SAMPLE s . LET N_S BE THE NUMBER OF SAMPLES IN A PARTICULAR SUBSET OF VIDEOS. FOR EACH DATASET AND SPLIT SUBSET, THIS TABLE REPORTS

$$\#l = \sum_{s=1}^{N_S} \sum_{f=1}^{N_F^s} gt(s, f, l), \text{ WHERE } l \in \{C, NC\}$$

IS THE LABEL, AND $gt(s, f, l) = 1$ IF l MATCHES THE C/NC LABEL ASSOCIATED TO THE PEDESTRIAN SAMPLE s AT FRAME f , AND $gt(s, f, l) = 0$ OTHERWISE

	PIE			PedSynth		
	Train	Valid.	Test	Train	Valid.	Test
#C	116.2K	18.1K	130.9K	155.1K	50.4K	50.5K
#NC	110.7K	22.1K	76.8K	82.3K	29.9K	29.5K
TOTAL	226.9K	40.2K	207.7K	237.4K	80.3K	80.0K

training purposes, however, to compare new results to our previous work, we keep the Train/Val./Test split proposed in [36]. Specifically, $\sim 80\%$ of PedSynth is allocated for training, $\sim 10\%$ for validation, and another $\sim 10\%$ for testing (see Table II). This means, that in this paper we use the $\sim 80\%$ of PedSynth as the synthetic source domain.

To evaluate self-labeling quantitatively, we use the metrics typically applied in C/NC prediction literature, namely accuracy, precision, recall, and F1-score. These metrics are standard for classification tasks. In this case, F1-score is the best metric, since it compensates for the bias of PIE validations and testing sets toward NC frames. In fact, as mentioned in Section III, S2R-UDA-CP internally uses F1-score for hyper-parameter tuning and convergence assessment.

As noted earlier, for self-labeling with S2R-UDA-CP we use PedGNN and ST-CrossingPose, both of which are significantly different in terms of their deep architecture designs. On the other hand, both models take 2D skeleton fitting as input to generate pose features for each tracked pedestrian. In particular, a skeleton consists of 17 joints obtained via AlphaPose [14] and additional joint interpolations to obtain Neck and CHip joints (not provided by AlphaPose). The original ST-CrossingPose proposal utilizes skeletons provided by OpenPose [8]. However, AlphaPose is known to provide more accurate skeletons. Therefore, we have adapted ST-CrossingPose to work with the same skeletons as PedGNN. To predict a C/NC label, we use N_F consecutive skeletons of a tracked pedestrian as input. A temporal sliding window is

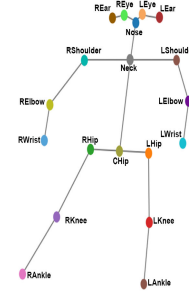


Fig. 3. Pedestrian skeleton (19 joints) as expected by PedGNN.

applied to process each track, incorporating the current frame and the preceding $N_F - 1$ frames, with a step size of one frame. We emphasize that synthetic and real-world videos undergo the same processing. Thus, we use AlphaPose to extract pedestrian skeletons from CARLA-based PedSynth, as it provides high accuracy and helps avoid domain gaps that might arise from using skeletons derived from the CARLA's ground truth [36]. Additionally, it is important to remember that the individual pedestrian motion patterns in CARLA were created, in general terms, through a two-step process: (1) capturing visual markers placed on the clothing of people performing movements of interest (walking, running, standing); (2) manually adjusting each pattern to fit each synthetic person model, considering factors like physical build, gender, and simulated age.

To evaluate the effectiveness of the pseudo-labels generated by S2R-UDA-CP for training third-party models, we conduct experiments using PedGraph+ [6], a literature reference on onboard C/NC prediction. This evaluation is performed twice, once with PedGNN and once with ST-CrossingPose as the models operating within S2R-UDA-CP. In other words, PedGraph+ is not used to operate S2R-UDA-CP.

B. Training Protocol

For training PedGNN and ST-CrossingPose, we use AdamW with binary cross-entropy loss and default parameters except for the learning rate, l_r . Training is conducted for M_E epochs with a $D\%$ dropout. The mini-batch size N_B is adjusted to fit a 24GB GPU memory. As per Algorithm 1, $l_r, M_E, D\%, N_B \in \mathcal{H}^M$. Therefore, we followed the approach outlined in Section III/Hyper-parameter tuning to determine the values for the hyper-parameters in $\mathcal{H}^M$ and $\mathcal{H}^{SI}$. This resulted in $l_r = 0.001$, $M_E = 100$, $D\% = 50$, and $N_B = 500$. Although these are standard hyper-parameters associated with the training of deep models, we also note that $N_F \in \mathcal{H}^M$. Consequently, we tested $N_F \in \{4, 8, 16, 32, 64\}$ and found that $N_F = 32$ (~ 1 second of past data) yielded the best prediction results. Once $\mathcal{H}^M$ was set, we focused on $\mathcal{H}^{SI}$, where $\beta_{SR}, N_W, \tau_v \in \mathcal{H}^{SI}$. To ensure a balanced representation of synthetic and real-world domains and allow the model to focus equally on each, we set $\beta_{SR} = 50$ for mixing samples at the mini-batch level. Imbalanced contributions, revealed that the model tended to overfit towards the dominant domain. Overfitting towards the synthetic domain may prevent convergence of the self-labeling process, while overfitting towards the real-world domain may lead to classical error drifting.

TABLE III

SELF-LABELING RESULTS ON PIE VALIDATION SET, USING PEDGNN. $\mathcal{M}_t$ AND p_t (F1-SCORE) ARE THE INTERMEDIATE TRAINED MODEL AND ITS PERFORMANCES AT CYCLE t IN ALGORITHM 1. THUS, $\mathcal{M}_0$ CORRESPONDS TO A TRAINING BASED ONLY IN SYNTHETIC LABELED DATA (PEDSYNTH). FOR $t > 0$, THE MODELS ARE TRAINED WITH SYNTHETIC DATA AND SELF-LABELED REAL-WORLD DATA (PIE TRAINING SET). SINCE THE F1-SCORE IS LOWER FOR $\mathcal{M}_4$ THAN FOR $\mathcal{M}_3$, ALGORITHM 1 STOPS WITH $\mathcal{V}_3^{RL}$ AS THE OUTPUT PIE TRAINING SET WITH PSEUDO-LABELS

Model	Accuracy	Precision	Recall	F1-Score (p_t)
$\mathcal{M}_0$	56.37	59.89	60.26	60.07
$\mathcal{M}_1$	58.38	60.98	68.47	64.55
$\mathcal{M}_2$	60.49	61.61	83.44	70.88
$\mathcal{M}_3$	62.29	62.23	87.55	72.75
$\mathcal{M}_4$	53.24	53.17	99.09	69.21

TABLE IV

ANALOGOUS TO TABLE III BUT USING ST-CROSSINGPOSE. IN THIS CASE, ALGORITHM 1 ALSO OUTPUTS $\mathcal{V}_3^{RL}$

Model	Accuracy	Precision	Recall	F1-Score (p_t)
$\mathcal{M}_0$	56.79	51.60	74.18	60.86
$\mathcal{M}_1$	58.03	68.32	55.09	60.99
$\mathcal{M}_2$	60.33	72.13	60.54	65.83
$\mathcal{M}_3$	62.91	76.18	64.49	69.85
$\mathcal{M}_4$	59.51	89.84	50.39	64.57

Note that while running S2R-UDA-CP, real-world samples are provided with pseudo-labels (in [36] the labels are provided by humans). We found that $N_W = 10$ frames (~ 333 ms) provides temporal stability in predictions, as shorter windows failed to effectively smooth spurious labels, while longer windows risked over-smoothing genuine transitions. Finally, we found $\tau_v = 1.0$ to be a suitable value for the minimum F1-score increment to avoid prematurely stopping the self-labeling loop.

C. S2R-UDA-CP Experiments

In these experiments, PIE training (ignoring the C/NC labels) and validation sets play the role of $\mathcal{V}^{RL}$ and $\mathcal{V}^{SL}$ in Algorithm 1, respectively. PedSynth plays the role of $\mathcal{V}^{SL}$. $\mathcal{H}^M$ and $\mathcal{H}^{SL}$ are set according to the training protocol in Section IV-B, and we set $T_{max} = \infty$ to allow the algorithm to stop by convergence.

Accordingly, Table III presents the performance of four intermediate C/NC prediction models on the PIE validation set when using PedGNN as C/NC prediction architecture (role of the input $\mathcal{M}$ in Algorithm 1). Table IV is the analogous when using ST-CrossingPose architecture. These intermediate models are produced by the S2R-UDA-CP procedure during its iterative process towards convergence (in Algorithm 1 these models are referred to as $\mathcal{M}_t$, and their F1-score on the validation set as p_t). $\mathcal{M}_0$ corresponds to training PedGNN/ST-CrossingPose solely with PedSynth. $\mathcal{M}_1$ additionally uses the pseudo-labels generated by $\mathcal{M}_0$ in the PIE training set, $\mathcal{M}_2$ utilizes PedSynth and the PIE training set with pseudo-labels generated by $\mathcal{M}_1$, and so forth.

TABLE V

C/NC PREDICTION RESULTS ON PIE TESTING SET. $\mathcal{M}_{3,PedGNN}$ AND $\mathcal{M}_{3,ST-CrossingPose}$ REFER TO $\mathcal{M}_3$ IN TABLES III AND IV, RESPECTIVELY. AS REFERENCE, WE INCLUDE RESULTS FROM [36], WHICH ARE BASED ON PEDGNN. SPECIFICALLY, $\mathcal{M}_S$ IS TRAINED ONLY USING PEDSYNTH (IT CORRESPONDS TO $\mathcal{M}_0$ IN ALGORITHM 1), $\mathcal{M}_P$ IS TRAINED ONLY USING PIE TRAINING SET WITH HUMAN-PROVIDED C/NC LABELS, AND $\mathcal{M}_{P+S}$ COMBINES BOTH TRAINING SETS ENSURING EQUAL DATASET SAMPLING PER MINI-BATCH

Model	Accuracy	Precision	Recall	F1-Score
$\mathcal{M}_S$	62.74	69.37	79.31	74.01
$\mathcal{M}_P$	68.11	66.98	68.36	67.81
$\mathcal{M}_{P+S}$	70.52	74.80	82.73	79.12
$\mathcal{M}_{3,PedGNN}$	70.83	75.35	90.25	82.11
$\mathcal{M}_{3,ST-CrossingPose}$	69.79	83.76	76.55	79.99

TABLE VI

USING THE SAME CRITERION THAN IN TABLE II, WE REPORT THE PIE SAMPLES USED IN THE PAPER PRESENTING PEDGRAPH+ [6]

Crossing (#C)			Not Crossing (#NC)		
Train	Valid.	Test	Train	Valid.	Test
5,959	1,369	5,346	17,481	5,217	13,621

We observe in Table III and Table IV that the respective F1-score values are lower for $\mathcal{M}_4$ than for $\mathcal{M}_3$. Consequently, in both cases, Algorithm 1 terminates with $\mathcal{V}_3^{RL}$ as the output PIE training set with pseudo-labels, being the respective $\mathcal{M}_3$ models the generators of these pseudo-labels.

D. On the Output $\mathcal{M}$: Onboard C/NC Prediction Experiments

In [36], we assessed the utility of PedSynth by training PedGNN under two scenarios: one using PedSynth as the sole training data (it corresponds to $\mathcal{M}_0$ in Algorithm 1), and another combining PedSynth and the PIE training set with human-provided C/NC labels, ensuring equal dataset sampling per mini-batch. Evaluations of the two resulting PedGNN versions were conducted on the PIE testing set. Since this set is not involved in the S2R-UDA-CP procedure, evaluating its output $\mathcal{M}$ on it can offer valuable insights into the effectiveness of the pseudo-labels compared to human-provided labels. Therefore, we have done it for PedGNN and ST-CrossingPose. We remark that the temporal smoothing on C/NC labels performed during self-labeling (N_W window) is not applied since $\mathcal{M}$ is used here to simulate onboard C/NC prediction (so there is no access to the future). The results are presented in Table V. As previously noted in [36], PedSynth and the human-labeled PIE training set complement each other, achieving higher accuracy and F1-score than training on a single domain. However, the most interesting outcome is that using PedSynth combined with the self-labeled PIE training set produces a performance similar to (with $\mathcal{M}_{3,ST-CrossingPose}$) that obtained using PedSynth combined with human-provided labels ($\mathcal{M}_{P+S}$), or even better (with $\mathcal{M}_{3,PedGNN}$). This confirms that S2R-UDA-CP is an effective self-labeling procedure that can work with different architectures for C/NC prediction.



Fig. 4. Discrepancies between human-provided C/NC labels and pseudo-labels in the PIE training set. FID represents the frame ID, which increments as the sequence progresses. GT denotes ground truth, referring to the human-provided C/NC label. Prediction refers to the pseudo-label. Predictions are highlighted in red if they do not match the GT and in green if they do. The same color scheme applies to the bounding box framing the pedestrian. For enhanced visualization, the pedestrian of interest is zoomed in and displayed as a circular patch. Please refer to the main text for further discussion on this figure.

TABLE VII

RESULTS OF PEDGRAPH+ [6] IN PIE TESTING SET USING DIFFERENT C/NC LABELS FOR THE SAME PIE TRAINING VIDEO SEGMENTS. MANUAL REFERS TO HUMAN LABELING, $\mathcal{V}_3^{RL}$ REFER TO PSEUDO-LABELS PROVIDED BY THE S2R-UDA-CP PROCEDURE WHEN USING PEDGNN AND ST-CROSSINGPOSE DEEP ARCHITECTURES

PIE training set labels	Accuracy	Precision	Recall	F1-Score
Manual	62.48	72.94	54.90	62.65
$\mathcal{V}_3^{RL}$ / PedGNN	69.35	75.29	71.00	72.02
$\mathcal{V}_3^{RL}$ / ST-CrossingPose	67.70	74.86	74.50	68.73

E. On the Output $\mathcal{V}^{RL}$: Time-to-Event Onboard Experiments

We further evaluate the effectiveness of the pseudo-labels produced by S2R-UDA-CP, $\mathcal{V}^{RL}$, by training a C/NC prediction model not involved in this procedure. We employ PedGraph+ [6], a literature referent whose code is publicly available. The training and testing setting in [6] differs from ours. We use the full PIE videos, while in [6] video segments just before an NC-C transition are taken as C examples, and some segments without crossing events are taken as NC examples. This results in significantly lower samples for training, validation, and testing (Table VI) than those that we have used so far. However, for the experiments in this section, this is right. The PIE training data used by PedGraph+ is available so we have been able to set it to work with their skeleton keypoint information, focusing on C/NC classes, and either using the human-provided C/NC labels (Table VI) or replacing those with our C/NC pseudo-labels. In the latter case, we train PedGraph+ with $\mathcal{V}^{RL}$, resulting in two scenarios. Firstly, when $\mathcal{V}^{RL} = \mathcal{V}_3^{RL}$ from Table III (pseudo-labels produced by using PedGNN), and secondly, when $\mathcal{V}^{RL} = \mathcal{V}_3^{RL}$ from Table IV (pseudo-labels produced by using ST-CrossingPose). As done in the S3R-UDA-CP procedure, the human-labeled PIE validation set (Table VI) is used to select the best checkpoint while training PedGraph+, both when using human-provided labels and with our pseudo-labels.

Table VII presents comparative results on the PIE training set used by PedGraph+ (Table VI). It is worth noting that, although the metrics shown in Table VII are the same as those used in previous tables (III, IV, V), there is a significant difference. In the previous tables, the C/NC classification results are based on all frames in the validation/testing set. However, the benchmarking protocol for PedGraph+ [6] proposes a time-to-event (TTE) classification task, so the metrics refer to this task. It involves detecting a C event with an anticipation (TTE) of

1-2 seconds. We observe that the results based on pseudo-labels are substantially better than those based on human labeling, with the PedGNN-based results being the most effective.

F. Qualitative Results

At this stage, it is also important to perform a qualitative analysis of the results. We begin by examining the C/NC labeling of the PIE training set. Specifically, we highlight some illustrative differences between human-provided labels and pseudo-labels. For simplicity, we focus on the pseudo-labels generated by using PedGNN for S2R-UDA-CP, *i.e.*, the $\mathcal{V}_3^{RL}$ associated to $\mathcal{M}_{3, PedGNN}$ (Table III).

Figure 4 depicts an adult accompanying two children, sampling a sequence of ~ 1.2 seconds. We think that the adult starts to cross the road (FID=1475), but since the human-driven ego-vehicle does not stop, the adult stops and waits for it to pass. In this case, the human labeler classifies the entire sequence as Non-Crossing (NC) because that is what ultimately occurs regarding the lane in what the ego-vehicle is moving. However, we argue that labeling this sequence as Crossing (C) is also valid, as an autonomous vehicle might slow down or even stop for safety reasons.

Figure 5 covers a sequence lasting ~ 16 seconds. A pedestrian walks along a sidewalk parallel to the ego-vehicle's lane. The pedestrian starts crossing a zebra crossing at FID=2886 and finishes after FID=3116. As a result, the human labeler has marked the entire sequence as a Crossing case. However, our self-labeling process only identifies it as such from FID=2900 onward. We contend that the human labeling from FID=2626 to FID=2886 is incorrect, as there is no evidence that the pedestrian intends to cross in front of the ego-vehicle during this approximately 9-second period. From FID=2900 onward, the ego-vehicle moves towards the pedestrian while they are crossing the zebra crossing. We suggest that labeling this interval as Crossing could introduce causal confusion during the training of C/NC prediction models.

Figure 6 spans a sequence of ~ 15 seconds. A person is walking his bike, in fact waiting for a green traffic light to cross a zebra crossing. Presumably, by analyzing the sequence until the ego-vehicle passes through the zebra crossing (after FID=1920), the human labeler has correctly identified a Non-Crossing situation. However, this case was challenging for our self-labeling procedure due to the small size of the person at the beginning of the sequence and the bike that can cause flickering in the skeleton fitting. At FID=1920, when the person is



Fig. 5. Discrepancies between human-provided C/N labels and pseudo-labels in the PIE training set. The naming and color conventions follow those in Figure 4. Please refer to the main text for further discussion on this figure.

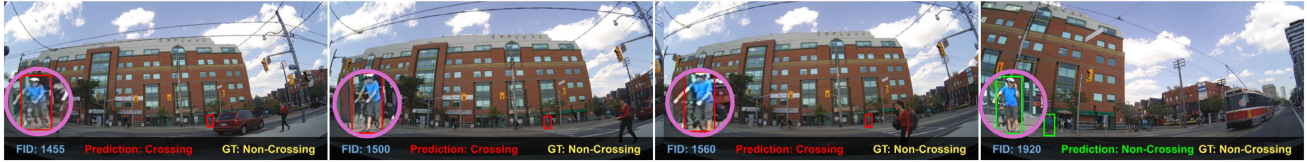


Fig. 6. Discrepancies between human-provided C/N labels and pseudo-labels in the PIE training set. The naming and color conventions follow those in Figure 4. Please refer to the main text for further discussion on this figure.

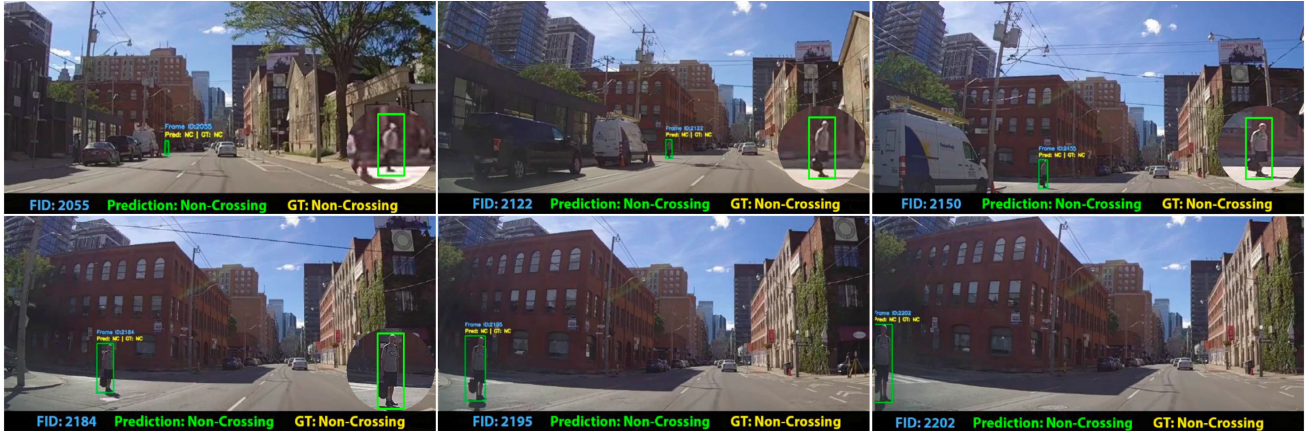


Fig. 7. Non-Crossing (NC) example from the PIE testing set. The naming and color conventions follow those in Figure 4. Prediction matches GT.

visible at a higher resolution, the self-labeling procedure also sets a Non-Crossing label. At this point, we could imitate the human labeler and propagate the new label backward in the sequence. However, this is not trivial to do automatically since we can inject causal confusion as the human labeler does in previous example. Thus, for addressing these cases, we think it is better to focus on improving the C/NC prediction model itself.

Figures 7 to 10 show qualitative results of the model $\mathcal{M}_{3, PedGNN}$ in the PIE testing set (Table V). Figures 7 and 8 show examples in which the prediction of the model matches the GT for NC and C scenarios, respectively. In both cases, classified pedestrians are seen within a wide range of distances. On the other hand, Figures 9 and 10 show more difficult cases where the model prediction does not match the GT in all frames, for NC and C scenarios, respectively. The discrepancies are concentrated in the initial frames of the C/NC events. These discrepancies arise from the online predictions of $\mathcal{M}_{3, PedGNN}$ during inference, in contrast to

the human labels of GT derived from the offline video analysis. In other words, $\mathcal{M}_{3, PedGNN}$ predicts without neither using future frames nor temporal label smoothing, whereas human labelers can review videos bidirectionally to assign C/NC GT labels. Accordingly, we think it is unclear if the model is performing wrong since the C/NC decision of these pedestrians may vary according to even uncapturable factors. For example, in Figure 9, the pedestrian appeared to be moving with the intention to cross the road (so the prediction of $\mathcal{M}_{3, PedGNN}$ during the initial ~ 0.9 seconds) but it stopped due to the traffic light (so the GT label event at the sequence starting). In this case, what is the right vehicle action before the pedestrian stops? ignore the pedestrian and keep the speed or even accelerate? slowing down to break on time if the pedestrian is mentally distracted and crosses in red? We think it is not an easy decision. Similarly, Figure 10 depicts a difficult case. Between FID=16940 and FID=17026 the pedestrian is mainly stopped and smoothly starts to move. During that ~ 2.8 seconds, $\mathcal{M}_{3, PedGNN}$ con-



Fig. 8. Crossing (C) example from the PIE testing set. The naming and color conventions follow those in Figure 4. Prediction matches GT.



Fig. 9. Non-Crossing (NC) example from the PIE testing set. The naming and color conventions follow those in Figure 4. Prediction and GT do not match at the beginning of the sequence. Please refer to the main text for further discussion on this figure.

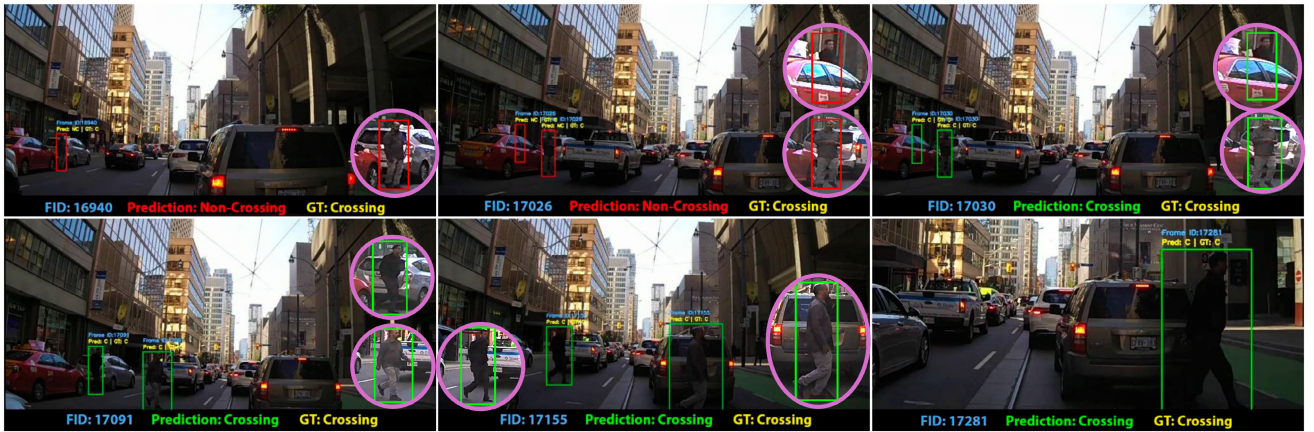


Fig. 10. Crossing (C) example from the PIE testing set. The naming and color conventions follow those in Figure 4. Prediction and GT do not match at the beginning of the sequence. Please refer to the main text for further discussion on this figure.

siders that the pedestrian is not crossing. At the beginning of the sequence, the vehicles start to stop due to a red traffic light, progressively all the vehicles stop and the pedestrian starts crossing the road (FID=17030). Accordingly, the GT and the $\mathcal{M}_{3, PedGNN}$ prediction match, indicating a crossing case since then. A slightly further away pedestrian proceeded similarly, creating the same discrepancies between the GT and the $\mathcal{M}_{3, PedGNN}$ predictions. Note how humans labeled both cases as crossing just because they propagated the label

backward in the video once the C/NC situation was clear. Onboard in real-time, we cannot make decisions based on observations from the future, we must decide what is the optimum behavior according to what has been observed so far and prior knowledge of pedestrian behaviors in the driving area. In this case, the pedestrians are crossing the road through a non-allowed path just because all the vehicles are stopped, thus, the driver of the ego-vehicle already decided to stop; what would the pedestrians do if not? would they stay without

crossing (as $\mathcal{M}_{3, PedGNN}$ was predicting) or would they try a riskier crossing (because the ego-vehicle is not fully stopped)? It is difficult to know, therefore, to determine the right labeling in this case is not trivial.

Overall, cases like those shown in Figures 9 and 10 show that final decisions on C/NC must consider not only the motion pattern of pedestrians but also global traffic information in the scene (e.g., presence and state of traffic lights, zebra crossings, priors on per-area pedestrian habits, etc.).

However, from the point of view of the goal in this paper, we have shown how our S2R-UDA-CP proposal could save the C/NC labeling of 226.9K samples (Table II) without losing C/NC prediction performance, in fact improving over manual labeling as seen in Tables V and VII.

V. CONCLUSION

In our previous work [36], we demonstrated that our synthetic dataset, PedSynth, complemented real-world pedestrian tracks manually labeled for C/NC prediction. In this paper, we challenge this costly and subjective process by introducing a self-labeling procedure, termed S2R-UDA-CP. It utilizes a synthetic dataset (e.g., PedSynth) and a deep architecture for C/NC prediction (e.g., PedGNN, ST-CrossingPose) to generate C/NC pseudo-labels for video streams with pedestrian tracks. By employing S2R-UDA-CP to self-label the well-known PIE training set, we have demonstrated that we can avoid manually labeling 226.9K samples for C/NC without compromising the resulting model's performance. Moreover, the qualitative analysis has shown that discrepancies between predictions and ground truth labels do not necessarily mean that predictions are wrong. Underlying those discrepancies is the onboard real-time behavior expected for any C/NC decision system in contrast to the offline nature of the human labeling process (that allows playing videos forward and backward). Moreover, sometimes, human labeling can introduce causal confusion and follow inconsistent labeling criteria. We plan to release the S2R-UDA-CP code, the trained models, and the resulting pseudo-labels. For future research, it would be core to investigate pedestrian C/NC prediction under adverse weather conditions. However, merely augmenting datasets such as PIE with simulated adverse weather, like rain, is not enough. Adverse weather affects human behavior and attire (e.g. the use of umbrellas), making it essential to develop new synthetic and real-world datasets with C/NC labels to drive progress in pedestrian C/NC prediction.

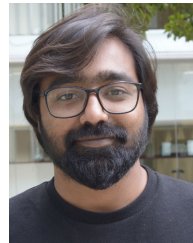
ACKNOWLEDGMENT

This work only expresses the opinion of the author and neither the European Union nor ACCIÓ are liable for the use made of the information provided. The authors acknowledge the support of the Generalitat de Catalunya CERCA Program and its ACCIO agency to CVC's general activities. Open Access Funding has been provided by Universitat Autònoma de Barcelona (UAB).

REFERENCES

- [1] L. Achaji, J. Moreau, T. Fouqueray, F. Aioun, and F. Charpillat, "Is attention to bounding boxes all you need for pedestrian action prediction?" in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2022, pp. 895–902.
- [2] J. Bai, X. Fang, J. Fang, J. Xue, and C. Yuan, "Deep virtual-to-real distillation for pedestrian crossing prediction," in *Proc. IEEE 25th Int. Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2022, pp. 1586–1592.
- [3] S. A. Bouhsain, S. Saadatnejad, and A. Alahi, "Pedestrian intention prediction: A multi-task perspective," 2020, *arXiv:2010.10270*.
- [4] A. Brunetti, D. Buongiorno, G. F. Trotta, and V. Bevilacqua, "Computer vision and deep learning techniques for pedestrian detection and tracking: A survey," *Neurocomputing*, vol. 300, pp. 17–33, Jul. 2018.
- [5] Y. Cabon, N. Murray, and M. Humenberger, "Virtual KITTI 2," 2020, *arXiv:2001.10773*.
- [6] P. R. G. Cadena, Y. Qian, C. Wang, and M. Yang, "Pedestrian graph+: A fast pedestrian crossing prediction model based on graph convolutional networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 11, pp. 21050–21061, Nov. 2022.
- [7] P. R. G. Cadena, M. Yang, Y. Qian, and C. Wang, "Pedestrian graph: Pedestrian crossing prediction based on 2D pose estimation and graph convolutional networks," in *Proc. IEEE Intell. Transp. Syst. Conf. (ITSC)*, Oct. 2019, pp. 2000–2005.
- [8] Z. Cao, G. Hidalgo, T. Simon, S. Wei, and Y. Sheikh, "OpenPose: Realtime multi-person 2D pose estimation using part affinity fields," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 1, pp. 172–186, Jan. 2021.
- [9] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2D pose estimation using part affinity fields," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1302–1310.
- [10] E. Cheung, T. K. Wong, A. Bera, and D. Manocha, "MixedPeds: Pedestrian detection in unannotated videos using synthetically generated human-agents for training," in *Proc. AAAI Conf. Artif. Intell.*, Apr. 2018, doi: [10.1609/aaai.v32i1.12325](https://ojs.aaai.org/index.php/AAAI/article/view/12325). [Online]. Available: <https://ojs.aaai.org/index.php/AAAI/article/view/12325>
- [11] G. Csurka, "A comprehensive survey on domain adaptation for visual applications," in *Domain Adaptation in Computer Vision Applications* (Advances in Computer Vision and Pattern Recognition). Cham, Switzerland: Springer, 2017, ch. 1.
- [12] A. Dosovitskiy, G. Ros, F. Codevilla, A. M. López, and V. Koltun, "CARLA: An open urban driving simulator," in *Proc. Conf. Robot Learn. (CoRL)*, 2017, pp. 1–16.
- [13] E. Bochinski, V. Eiselein, and T. Sikora, "Training a convolutional neural network for multi-class object detection using solely virtual world data," in *Proc. 13th IEEE Int. Conf. Adv. Video Signal Based Surveill. (AVSS)*, Aug. 2016, pp. 278–285.
- [14] H.-S. Fang et al., "AlphaPose: Whole-body regional multi-person pose estimation and tracking in real-time," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 6, pp. 7157–7173, Jun. 2023.
- [15] H.-S. Fang, S. Xie, Y.-W. Tai, and C. Lu, "RMPE: Regional multi-person pose estimation," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2353–2362.
- [16] Z. Fang and A. M. López, "Intention recognition of pedestrians and cyclists by 2D pose estimation," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 11, pp. 4773–4783, Nov. 2020.
- [17] J. Gesnouin, S. Pechberti, B. Stanculescu, and F. Moutarde, "TrouSPI-net: Spatio-temporal attention on parallel atrous convolutions and U-GRUs for skeletal pedestrian crossing prediction," in *Proc. 16th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG)*, Dec. 2021, pp. 1–7.
- [18] J. L. Gómez et al., "All for one, and one for all: UrbanSyn dataset, the third musketeer of synthetic driving scenes," 2023, *arXiv:2312.12176*.
- [19] A. Gurram, A. F. Tuna, F. Shen, O. Urfalioglu, and A. M. Lopez, "Monocular depth estimation through virtual-world supervision and real-world SFM self-supervision," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 8, pp. 12738–12751, Aug. 2021.
- [20] J. L. Gómez, G. Villalonga, and A. M. López, "Co-training for unsupervised domain adaptation of semantic segmentation models," *Sensors*, vol. 23, no. 2, p. 621, Jan. 2023.
- [21] J.-S. Ham, D. H. Kim, N. Jung, and J. Moon, "CIPF: Crossing intention prediction network based on feature fusion modules for improving pedestrian safety," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2023, pp. 3666–3675.
- [22] H. Hattori, V. N. Boddeti, K. Kitani, and T. Kanade, "Learning scene-specific pedestrian detectors without real data," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3819–3827.
- [23] L. Hoyer, D. Dai, H. Wang, and L. Van Gool, "MIC: Masked image consistency for context-enhanced domain adaptation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 11721–11732.

- [24] C. G. Keller and D. M. Gavrila, "Will the pedestrian cross? A study on pedestrian path prediction," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 2, pp. 494–506, Apr. 2014.
- [25] W. Kim et al., "PedX: Benchmark dataset for metric 3-D pose estimation of pedestrians in complex urban intersections," *IEEE Robot. Autom. Lett.*, vol. 4, no. 2, pp. 1940–1947, Apr. 2019.
- [26] I. Kotseruba, A. Rasouli, and J. K. Tsotsos, "Benchmark for evaluating pedestrian action prediction," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2021, pp. 1257–1267.
- [27] C. Li and G. H. Lee, "From synthetic to real: Unsupervised domain adaptation for animal pose estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 1482–1491.
- [28] X. Liu et al., "Importance-aware semantic segmentation in self-driving with discrete Wasserstein training," in *Proc. AAAI Conf. Artif. Intell.*, vol. 34, Apr. 2020, pp. 11629–11636.
- [29] Google Self-Driving Car Testing Report on Disengagements of Autonomous Mode, Google Auto LLC, Dec. 2015. [Online]. Available: <https://www.driverlesstransportation.com/portal/wcm/connect/dff67186-70dd-4042-bc8c-d7b2a9904665/GoogleDisengagementReport2014-15.pdf>
- [30] J. Marín, D. Vázquez, D. Gerónimo, and A. M. López, "Learning appearance in virtual scenarios for pedestrian detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 137–144.
- [31] P. Oza, V. A. Sindagi, V. V. Sharmini, and V. M. Patel, "Unsupervised domain adaptation of object detectors: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 6, pp. 4018–4040, Jun. 2024.
- [32] A. Rasouli, I. Kotseruba, T. Kunic, and J. Tsotsos, "PIE: A large-scale dataset and models for pedestrian intention estimation and trajectory prediction," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6261–6270.
- [33] A. Rasouli, I. Kotseruba, and J. K. Tsotsos, "Are they going to cross? A benchmark dataset and baseline for pedestrian crosswalk behavior," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops (ICCVW)*, Oct. 2017, pp. 206–213.
- [34] A. Rasouli, M. Rohani, and J. Luo, "Bifold and semantic reasoning for pedestrian behavior prediction," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 15580–15590.
- [35] H. Razali, T. Mordan, and A. Alahi, "Pedestrian intention prediction: A convolutional bottom-up multi-task approach," *Transp. Res. C, Emerg. Technol.*, vol. 130, Sep. 2021, Art. no. 103259.
- [36] M. N. Riaz, M. Wielgosz, A. G. Romera, and A. M. López, "Synthetic data generation framework, dataset, and efficient deep model for pedestrian intention prediction," in *Proc. IEEE 26th Int. Conf. Intell. Transp. Syst. (ITSC)*, Sep. 2023, pp. 2742–2749.
- [37] S. R. Richter, V. Vineet, S. Roth, and V. Koltun, "Playing for data: Ground truth from computer games," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Jan. 2016, pp. 102–118.
- [38] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. M. Lopez, "The SYNTHIA dataset: A large collection of synthetic images for semantic segmentation of urban scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 3234–3243.
- [39] F. Schneemann and P. Heinemann, "Context-based detection of pedestrian crossing intention for autonomous driving in urban environments," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2016, pp. 2243–2248.
- [40] N. Schneider and D. M. Gavrila, "Pedestrian path prediction with recursive Bayesian filters: A comparative study," in *Proc. German Conf. Pattern Recognit. (GCPR)*, Jan. 2013, pp. 174–183.
- [41] M. Schwonberg et al., "Survey on unsupervised domain adaptation for semantic segmentation for visual perception in automated driving," *IEEE Access*, vol. 11, pp. 54296–54336, 2023.
- [42] S. Shah, D. Dey, C. Lovett, and A. Kapoor, "AirSim: High-fidelity visual and physical simulation for autonomous vehicles," in *Proc. Field Service Robot. (FSR)*, Nov. 2017, pp. 621–635.
- [43] T. Stauner et al., "SynPeDS: A synthetic dataset for pedestrian detection in urban traffic scenes," in *Proc. Comput. Sci. Cars Symp.*, Dec. 2022, pp. 1–10.
- [44] I. Triguero, S. García, and F. Herrera, "Self-labeled techniques for semi-supervised learning: Taxonomy, software and empirical study," *Knowl. Inf. Syst.*, vol. 42, no. 2, pp. 245–284, Feb. 2015.
- [45] J. E. van Engelen and H. H. Hoos, "A survey on semi-supervised learning," *Mach. Learn.*, vol. 109, no. 2, pp. 373–440, Feb. 2020.
- [46] G. Villalonga and A. M. L. Peña, "Co-training for on-board deep object detection," *IEEE Access*, vol. 8, pp. 194441–194456, 2020.
- [47] F. Wang, J. Bai, and J. Fang, "Pedestrian crossing prediction based on invariant feature extraction of cross-spectral images," in *Proc. Int. Conf. Auto. Unmanned Syst. (ICAUS)*, Jan. 2023, pp. 2186–2196.
- [48] M. Wang and W. Deng, "Deep visual domain adaptation: A survey," *Neurocomputing*, vol. 312, pp. 135–153, Oct. 2018.
- [49] G. Wilson and D. J. Cook, "A survey of unsupervised deep domain adaptation," *ACM Trans. Intell. Syst. Technol.*, vol. 11, no. 5, pp. 1–46, Oct. 2020.
- [50] M. Wrenninge and J. Unger, "Synscapes: A photorealistic synthetic dataset for street scene parsing," 2018, *arXiv:1810.08705*.
- [51] D. Yang, H. Zhang, E. Yurtsever, K. A. Redmill, and Ü. Özgüner, "Predicting pedestrian crossing intention with feature fusion and spatio-temporal attention," *IEEE Trans. Intell. Vehicles*, vol. 7, no. 2, pp. 221–230, Jun. 2022.
- [52] Y. Yao, E. Atkins, M. Johnson Roberson, R. Vasudevan, and X. Du, "Coupling intent and action for pedestrian crossing behavior prediction," 2021, *arXiv:2105.04133*.
- [53] X. Zhang, P. Angeloudis, and Y. Demiris, "ST CrossingPose: A spatial-temporal graph convolutional network for skeleton-based pedestrian crossing intention prediction," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 11, pp. 20773–20782, Nov. 2022.
- [54] Z. Zhang, R. Tian, and Z. Ding, "TrEP: Transformer-based evidential prediction for pedestrian intention with uncertainty," in *Proc. AAAI Conf. Artif. Intell.*, vol. 37, Jun. 2023, pp. 3534–3542.
- [55] S. Zhao et al., "A review of single-source deep unsupervised visual domain adaptation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 2, pp. 473–493, Feb. 2022.
- [56] Y. Zhou, G. Tan, R. Zhong, Y. Li, and C. Gou, "PIT: Progressive interaction transformer for pedestrian crossing intention prediction," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 12, pp. 14213–14225, Dec. 2023.



Muhammad Naveed Riaz received the bachelor's degree from COMSATS University Islamabad (CUI), Pakistan, and the master's degree from Shanghai Jiao Tong University (SJTU), China, where his research focus was facial expression recognition. He is currently a Ph.D. Scholar with the Computer Vision Center (CVC), Universitat Autònoma de Barcelona (UAB), where he works in the field of autonomous driving for pedestrian safety.



Maciej Wielgosz received the Habilitation degree in information and communication technology in 2021 for his series of scientific articles titled "Design of Low-Latency Architectures for Machine Learning Algorithms." He has been a Machine Learning Researcher at Norwegian Institute of Bioeconomy Research (NIBIO), focusing on ML solutions for point cloud data, 3-D models, and imagery in forestry. Before joining NIBIO, he was involved in the ARCANE Project. This allowed the development of adversarial cases for autonomous vehicles (ARCANE) at the Computer Vision Center (CVC), Universitat Autònoma de Barcelona.



Antonio M. López (Member, IEEE) currently leads the Autonomous Driving Laboratory, Computer Vision Center (CVC), Universitat Autònoma de Barcelona (UAB). He is currently a Full Professor at the Computer Science Department, UAB. He is one of the creators of the SYNTHIA and UrbanSyn datasets and the CARLA simulator. He collaborates with industry partners to conduct cutting-edge research in autonomous driving. His research focuses on the intersection of computer vision, simulation, machine learning, and autonomous driving. He has been awarded the Catalan ICREA Academia Program for his research contributions.