IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received October 6, 2021, accepted October 23, 2021, date of publication October 29, 2021, date of current version November 8, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3123952

Indoor/Outdoor Semantic Segmentation Using Deep Learning for Visually Impaired Wheelchair Users

ELHASSAN MOHAMED, KONSTANTINOS SIRLANTZIS, AND GARETH HOWELLS, (Senior Member, IEEE)

Intelligent Interactions Research Group, Kent Assistive RObotics Laboratory (KAROL), School of Engineering, University of Kent, Canterbury CT2 7NT, U.K.

Corresponding author: Elhassan Mohamed (enrm4@kent.ac.uk)

This work was supported by the Assistive Devices for empowering dis-Abled People through robotic Technologies (ADAPT) project. ADAPT was selected for funding by the INTERREG VA France (Channel) England Programme, which is co-financed by the European Regional Development Fund (ERDF). The European Regional Development Fund (ERDF) is one of the main financial instruments of the European Unions (EU) cohesion policy.

ABSTRACT Electrical Powered Wheelchair (EPW) users may find navigation through indoor and outdoor environments a significant challenge due to their disabilities. Moreover, they may suffer from near-sightedness or cognitive problems that limit their driving experience. Developing a system that can help EPW users to navigate safely by providing visual feedback and further assistance when needed can have a significant impact on the user's wellbeing. This paper presents computer vision systems based on deep learning, with an architecture based on residual blocks that can semantically segment high-resolution images. The systems are modified versions of DeepLab version 3 plus that can process high-resolution input images. Besides, they can simultaneously process images from indoor and outdoor environments, which is challenging due to the difference in data distribution and context. The proposed systems replace the base network with a smaller one and modify the encoder-decoder architecture. Nevertheless, they produce high-quality outputs with fast inference speed compared to the systems with deeper base networks. Two datasets are used to train the semantic segmentation systems: an indoor application-based dataset that has been collected and annotated manually and an outdoor dataset to cover both environments. The user can toggle between the two individual systems depending on the situation. Moreover, we proposed shared systems that automatically use a specific semantic segmentation system depending on the pixels' confidence scores. The annotated output scene is presented to the EPW user, which can aid with the user's independent navigation. State-of-the-art semantic segmentation techniques are discussed and compared. Results show the ability of the proposed systems to detect objects with sharp edges and high accuracy for indoor and outdoor environments. The developed systems are deployed on a GPU based board and then integrated on an EPW for practical usage and evaluation. The used indoor dataset is made publicly available online.

INDEX TERMS CNN architecture, disabled people, deep learning, object localization, object detection, pixels classification, semantic segmentation, visually impaired users.

I. INTRODUCTION

Driving an EPW can be challenging, especially for users who suffer from cognitive problems. In addition to their physical issues, they may suffer from near-sightedness or limited neck and head movement. These can affect their driving experience, especially in complex environments, as they cannot fully recognize their routes or the dimensions of their EPWs. Many accidents and injuries have been reported for users injuring themselves or falling from EPW as they could not distinguish between pavement edges and car routes or walls and doors [1]–[3]. Besides, some users cannot be prescribed an EPW because of their disability [4].

In the ADAPT (Assistive Devices for empowering dis-Abled People through robotic Technologies) project [5], developing assistive devices that can facilitate the driving experience of users with cognitive and physical problems is the primary objective. A computer vision system that can

The associate editor coordinating the review of this manuscript and approving it for publication was Felix Albu.

help users to distinguish between different components of a complex environment will significantly impact the user's experience, specifically if a visual feedback can be presented.

EPWs' users who do not accept fully autonomous or semi-autonomous navigation (shared control) or who want to be in full control of the EPW might benefit from such a system that can provide environmental cues to guide them. Autonomous systems can be frustrating to some users when they try to approach an object or a door, but the collision avoidance system prevents them from doing so. One of the main requirements for an EPW system is not to act as a caregiver but instead as an assistant, and the user can override the system control at any point [6]. Visually and cognitively impaired users can benefit from such a system that guides them while giving them full control over the EPW. However, the proposed system can be combined with autonomous ones or used as a standalone system, depending on the user's condition.

In this paper, two individual systems based on deep learning for pixel classification are presented. A manually collected dataset for indoor environments and an outdoor dataset (Cambridge-driving Labeled Video Database (CamVid) [7]) are used to train the two systems. The systems' architectures are based on DeepLab3 plus [8] (hereafter DLV3+ for simplicity) for semantic segmentation and ResNet-18 is used as the feature extraction backbone network [9]. ResNet-18 is an adequate choice as it has a smaller footprint and a lesser number of layers when compared to its elder sisters (ResNet50 and ResNet101).

We also introduce three novel shared systems that can semantically segment images from both indoor and outdoor environments simultaneously. The novelty of the proposed three shared systems is not only in the architecture but also in the elegance of reusing the learned information/weights by the individual systems without the need to retrain the shared systems. Most importantly, the shared systems can process two different environments with almost the same accuracy as the individual systems, which is challenging as the data (images) being processed comes from two different distributions (indoor and outdoor).

The paper's main contributions can be summarised as follows: a) development of computer vision scene understanding systems for disabled people, b) an extensive dataset for indoor objects of interest has been presented, c) a modified architecture based on residual blocks that can process high-resolution images has been employed, d) different systems' architectures that can simultaneously process both indoor and outdoor images have been proposed, and finally, e) the developed systems have been deployed on a GPU based board and then integrated on an EPW for practical usage and evaluation. Though, the proposed computer vision systems can be deployed on any robotic platform for navigation and scene understanding.

The paper is organized as follows: Section II section covers the related assistive technologies for EPWs and semantic segmentation literature. Section III section discusses the datasets, challenges, systems' architectures and training setup. Systems performances and outcomes are explored in the Section IV section. Section V section outlines the constraints and the future scope of the study. Lastly, Section VI and future work are highlighted.

II. RELATED WORK

Scene understanding approaches are widely used in the autonomous driving industry. Adopting these technologies to help EPW users to drive safely in indoor and outdoor environments is a novel research topic. Related work can be divided into assistive devices for EPW users and semantic segmentation for scene understanding. For the assistive devices subsection, we will focus on the need for such a system that can help users with visual impairments to use an EPW as some of these users are not prescribed a powered wheelchair due to their disabilities [4]. In the semantic segmentation subsection, the focus is on state-of-the-art systems for pixels classification.

A. ASSISTIVE DEVICES AND MOTIVATIONS

There are many motivations for disabled people to utilize EPWs. Apart from the main reason, which is mobility, other factors such as productivity, leisure and independence are involved [10]. That is why assistive devices should enable users to master their objectives independently and enhance their quality of life. At the same time, poor design and faulty assistive devices have a negative influence on the user's experience [10].

Clinicians report that they saw almost the same number of patients who cannot use a powered wheelchair as who can [4]. Patients find it extremely difficult to manoeuvre an EPW indoors, especially in small areas and while negotiating doorways. Clinicians also report that 40% of their powered wheelchair users find steering tasks difficult. At the same time, five to nine percent find them impossible. On the other hand, the percentage of those who cannot use a powered wheelchair due to visual impairment, cognitive disorder or motor skills is 85%. An automated navigation system is believed to half this percentage [4].

Navigation systems based on computer vision, such as driving a wheelchair using face tracking [11] or eye and iris movement [12], [13], offer semi-autonomous and fully-autonomous driving capabilities for EPWs' users. Moreover, technologies such as collision detection and avoidance can be used to assist the driver in negotiating obstacles [14]. Viswanathan *et al.* [15] introduce a 3D stereo-vision navigation-based system that can detect potential object collisions by stopping the movement towards that object, plan paths towards a specific goal using visual odometry, and prompt to assist the user in navigation based on the user's level of awareness. A comprehensive review of smart wheelchairs is presented in [16]. Although these systems provide great help, they can be unsatisfactory or faulty. For example, consider the case when a user wants to approach an object that has been detected by the system as an obstacle.

In this case, the autonomous system wants to avoid the object, while the user needs to reach that object. The only possible solution is to disable the system. In contrast, our proposed systems act as a guide for the user. They do not interfere in the navigation process. They are non-intrusive systems, which classify the environment into different classes to lead and smooth the user's navigation process.

One of the closely related systems to ours is presented in [17]. A wheelchair system to guide people with severe disabilities is used to track manually taught paths (reference paths stored on a memory) using optical encoders mounted on the wheelchairs' wheels and visual beacons (passive cues) placed throughout the wheelchair surrounding environment (on walls, stationary objects, etc.). Relying only on the optical encoders to estimate the wheelchair's position may introduce errors because of the inaccurate initial conditions, wheel slippage, etc. Environmental cues that are captured by the two cameras installed on the powered wheelchair are used to correct and update the wheelchair's position and orientation using Kalman filter algorithm. The system uses the difference between the reference path and the estimated position to drive the wheelchair automatically. However, the system does not override the control from the user to follow a path until the user request so.

The main disadvantages of such a system are as follows: it needs the deployment of visual cues in the wheelchair environment. It needs a manually taught reference path. Most importantly, the system needs a different setup for different environments. This means that if the environment changes, new reference paths are needed to teach the system. Although we do not use our systems for path tracking, our proposed systems detect visual cues automatically. There is not any need to add physical visual cues to the environment. The proposed systems can detect the distance to a specific object using the Intel®RealSense depth camera (video). Besides, our systems provide all the information to the user on a screen from which the user can take full control of the EPW (video).

In contrast to the fully autonomous EPWs systems that take the control away from the users, which are sometimes undesirable, our methods provide environmental cues to help and guide them. It keeps the users in full control. EPWs systems that provide collision avoidance support such as [18], [19] may not be suitable for drivers who are unable to determine their location and cannot navigate to a specific location. Our systems allow the users to understand their surroundings and provide them with the distance to an object when needed. Consequently, the proposed systems can be seen as in-between systems that can provide environmental cues (scene understanding). At the same time, the systems do not override the user's ability to fully control the EPW, which tackle both disadvantages of non-autonomous and fully autonomous systems. Though the proposed systems can be integrated with autonomous ones, and the users can decide the level of assistance.

B. SEMANTIC SEGMENTATION

Fully Convolutional Network (FCN) [20] represents the fundamental of many state-of-the-art deep learning techniques for semantic segmentation. Besides, it represents the base of full scene understanding using deep learning. Semantic segmentation techniques can be divided into two main categories: series architecture and encoder-decoder architectures. Though, the latter architectures stem from the series ones.

FCN is considered the first work to train a network end-to-end for pixel-wise prediction using supervised pre-trained networks. It adapts state-of-the-art classification networks such as AlexNet [21], VGG [22] and GoogleNet [23] to make use of the learned features by these networks on classification tasks and transfer them to semantic segmentation tasks through transfer learning [24] and architecture modifications. Architecture modifications include replacing all the fully connected layers with convolutional ones and in-network upsampling to the original input image size. FCN does not make use of pre/post-processing complications such as superpixels, region proposals or post-hoc refinement by random field or local classifiers [25], [26].

Although FCN architecture has achieved a high score on standard metrics (mean pixel Intersection over Union), the produced semantic segmentation output is unrefined. Spatial details are not accurate, and object boundaries are not well-defined. It does not comprise useful global context information, instance awareness is not presented, and performance is far from real-time execution. Also, it is not entirely suited for unstructured data such as 3D point cloud [27], [28].

The main challenge facing semantic segmentation is the tension between semantics and locations (global and local information). Many solutions have been proposed to integrate context knowledge, such as Conditional Random Fields (CRFs), dilated convolutions and multi-scale predictions. DeepLab [29], [30] makes use of CRFs to refine segmentation results and object boundaries as a separate post-processing stage. Dilated convolution, also known as atrous convolution, is used in DeepLab [8], [29]–[31] to boost output resolution. Also, multi-scale context aggregation [32] makes use of dilated convolution. Dilated convolutions support expanding receptive fields without trading-off the resolution. They allow efficient dense feature extraction on any arbitrary resolution. Besides, multi-scale sub-networks with different resolution output are proposed to refine the coarse prediction progressively [33].

Skip architecture is introduced in FCN [20] to overcome the global/local information dilemma. Skip design combines 'fuses' semantic information from deep, coarse layers with appearance 'context' information from shallow fine layers to produce detailed segmentation. By doing so, the model becomes capable of making local predictions in the sense of the global structure. Skip connections convert the series architecture of the FCN into a DAG one (Directed Acyclic Graph). Skip architecture is learned end-to-end to refine the semantics and spatial accuracy of the output [20].

On the other hand, there is the encoder-decoder network architecture. Many state-of-the-art semantic segmentation architectures follow this design such as U-Net [34], SegNet [35] and DLV3+ [8]. U-Net [34] is built upon FCN [20] with some modifications to yield precise segmentation with few training images. The main architecture modification is the addition of the decoder part (up-sampling), where a large number of feature channels allow the network to propagate context information to higher resolution layers.

U-Net is trained end-to-end and outperforms the sliding window based convolutional network [36], [37] in terms of accuracy and inference speed. The system has achieved high performance on biomedical image segmentation applications using a few annotated images thanks to the data augmentation and elastic deformation techniques. It is also promised to provide high-quality results on other segmentation applications.

Both DeconvNet [38] and SegNet [35] use VGG-16 [22] as their feature extraction (backbone) encoder part. Unlike DeconvNet, SegNet discards the fully connected layers of the VGG-16 architecture. The decoder part of the DeconvNet consists of deconvolution and un-pooling layers [38]. However, the SegNet decoder part recalls max-pooling indices from the corresponding encoder layer during the up-sampling process, unlike U-Net [34] which transfers the entire feature maps from the encoder to the decoder. This makes SegNet fast in both training and testing with a small model size and memory footprint.

DLV3+ [8] follows the encoder-decoder structure with DeepLabv3 (DLV3) [31] as the encoder attached to it a simple yet effective decoder module. DLV3 and DLV3+ avoid using CRF as it is a post-processing stage that obstructs the network models from end-to-end training, unlike their ancestor systems DeepLabV1 (DLV1) [29] and DeepLabV2 (DLV2) [30] which can be considered as two cascade modules systems (Deep Convolution Neural Network (DCNN) then CRFs). DLV3+ introduces atrous separable convolution, which is composed of a depthwise convolution (spatial convolution for each input channel) followed by a pointwise convolution (1×1 convolution to combine the output from depthwise convolution). This leads to a significant reduction in computation complexity. Atrous separable convolution is applied to both Atrous Spatial Pyramid Pooling (ASPP) and the decoder modules. ASPP is introduced in DLV2 inspired by the spatial pyramid pooling method [39] to capture objects and context at multiple scales.

The decoder part of DLV3+ is simpler than that of U-Net [34] and SegNet [35]. Encoder features are first bilinearly up-sampled by a factor of 4 and then concatenated with corresponding low-level features. A 1×1 convolution reduces the number of channels of the low-level features before concatenation. After concatenation, a few 3×3 convolutions are applied to refine the features, followed by another bilinear up-sampling by a factor of 4. This strategy is better than directly up-sampling the features by a factor of 16 as it reduces the required computations (the number of trainable parameters). Besides, it allows multi-scale features to propagate through multiple layers of the decoder part. Consequently, better information can be extracted from the images.

In this paper, DLV3+ (the encoder-decoder structure) is adapted with some modifications (detailed in the next section) and applied to a real-life application.

III. METHODOLOGY

A. DATASETS

Available standard datasets [40]–[44] contain general objects of indoor environment but lack objects related to the proposed application. Thus, collecting and annotating a task-specific dataset is a non-avoidable requirement. For example, the door handle class in the aforementioned datasets is generic. Whereas the proposed indoor dataset contains different kinds of door handles for better perception and human-system interaction. An Intel® RealSense depth camera is installed on the Roma Reno II EPW for data collection and inference (Fig. 1). Objects of interest are doors, floors, walls, fire extinguishers, key slots, switches, and different kinds of door handles such as moveable, pull, and push door handles.

These objects are not only important for EPWs users but also for any robotic platform. Any robotic platform which uses particular actuators to open a door would require information about the type of the door handle in order to engage a suitable strategy for opening the door. For example, pull door handles require different actuation than moveable door handles. The ADAPT project chooses these classes as they represent the main objects an EPW user may need to interact with or utilize on a daily life basis. Other classes of interest can be added later depending on the user's ability and the surrounding environment.

The proposed indoor dataset can be augmented using some classes from the ADE20K [40], [41], NYU depth [42], [43] and SceneNN [44] datasets which have objects instances of indoor environment. However, specific classes, such as door handle types, do not exist in these datasets. These classes, besides key slot and switch classes, are infrequent. Nevertheless, they are important for our application. To keep a balanced distribution of class pixels, abundant classes such as door, floor, and wall are not included from the standard datasets. However, more objects from standard datasets may be included to create a customized implementation of the system upon the user's need and the adequate distribution of important task-oriented labels. This may require systems retrain to tune their weights on the extended tasks.

While driving the EPW through the indoor environment, a one-minute video is recorded and annotated. Images extracted from the video are shuffled and split randomly into 70% for training (1084 images), 15% for validation (232 images), and 15% for testing (233 images). Examples of the collected data with ground truth annotations are shown in Fig. 2. Pixels that do not fit into any of the eight predefined classes were assigned to an extra class called 'Background Wall' class. At the same time, small areas between two

(a) Roma Reno II EPW (b) Intel® RealSense Camera

FIGURE 1. Camera installation. Camera is installed beneath the EPW's joy stick so that there is no interference with the users' legs which can obstruct vision.

different classes, such as door frames or cupboards, are kept unannotated (void pixels). These pixels cannot fit in one class, such as the 'Background Wall' class, as they belong to different categories of objects.

Unlike the well-known datasets [45], [46], which usually have one big object per image, the proposed dataset has many objects per image; some of them can be categorised as small objects such as door handles and switches. In addition, these small objects are not available in the aforementioned datasets. This needs novel approaches that can produce better accuracy and sharp edges. Using high resolution and large size images may help to tackle this problem as many pixels will be utilized and contribute to the object classification. This may need high computation than smaller and fewer resolution images. That is why we used the elegant architecture of residual blocks and a smaller but powerful base network such as ResNet-18 compared to ResNet-50, ResNet-101 [9] or Xception [47] that are used in the original implementation of DLV3+ [8]. Thus, the system footprint can be reduced, which may help to deploy the system on a GPU-based hardware for inference. Besides, a semantically segmented environment can be displayed to the users in real-time or near real-time without sacrificing the system's accuracy.

The only common thing between the proposed indoor dataset and the one presented in [48] is that both are recorded in the same environment. However, they are different in the following: the system setup and the devices used for recording are different because the dataset presented in [48] is recorded by a handheld standard camera. Whereas the proposed dataset is collected using an Intel®RealSense camera installed on a Roma Reno II EPW to gain the same perspective and orientation as an EPW user. Larger images with better resolution, consequently more pixels, are collected to overcome the problem of small objects presented in the same study [48] and to utilize more pixels to enhance the overall system accuracy for small objects. The proposed dataset is annotated on the pixel level (for semantic segmentation). In contrast, the previous version is annotated on the bounding boxes level (for object detection). Finally, unlike the previous version, the proposed dataset is made publicly available for other researchers using this link.

The proposed dataset images might look homogeneous as it has been collected from one trajectory. Many factors can affect the perspective of the captured dataset, such as camera installation, which is limited by the available space on the EPW. However, we were able to capture different angles, directions, and orientations of small and rare objects of interest under different illuminations. Fig. 3 shows the front, side, and partial moveable door handles captured during the data collection. Besides, data augmentation is employed during training, giving another dimension for the dataset and increasing the model's robustness and ability to generalize to other environments. Furthermore, the dataset will be extended along with the study and future work.

EPWs have limited positions where a vision camera can be integrated or placed. The size of the EPW constrains these positions. Also, placing a camera on an EPW should not be obscured by the driver's body, legs, or hands. We proposed two locations that can be used for this purpose. The first option is a camera installed below the joystick controller, as shown in Fig. 1b. The second choice is a camera installed on a stick/holder that can be extended above the driver's head. There might be other places depending on the EPW type and design. For each case, a video has been collected. Each of them is recorded in two different environments to capture different perspectives and trajectories. We annotated

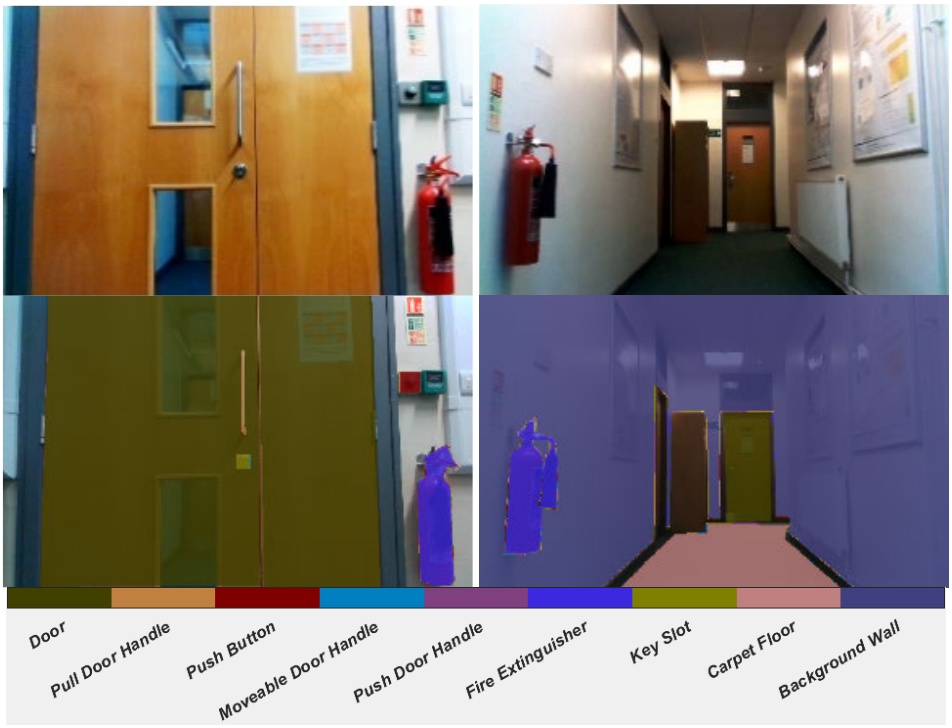


FIGURE 2. Indoor ground truth data. Examples from the collected indoor dataset with the first row represents the original images and the second one represents the annotated ones.

FIGURE 3. Moveable door handles. Although dataset objects might look similar, we were able to collect different angles and orientations of rare classes under different light conditions.

and used the first video in this study. The second one is under processing and will be added to the public dataset and used in our future work.

For the outdoor environment, we used the CamVid dataset [7] to train a second semantic segmentation system. CamVid dataset [7] has 701 images annotated on the pixel level for 32 classes. Images are captured outdoors from the perspective of a driving car. We categorised the 32 classes of the CamVid dataset into 11 classes for simplicity: Building, Pole, Road, Pavement, Tree. Sign/Symbol, Car, Pedestrian, Bicyclist, Sky, and Fence.

Similarly, the outdoor dataset is split randomly into 70% for training (491), 15% for validation (105 images) and 15% for testing (105 images). Examples of the CamVid [7] ground truth data are shown in Fig. 4.

B. CHALLENGES

Many of the objects of interest in the proposed indoor dataset can be classified as small size objects. Small size objects do not possess enough pixels to be utilized for feature extraction. Also, distinguishing between different door handles represents a great challenge because of their common colour and location in the dataset's images. Consequently, conventional object detection and semantic segmentation technique traditionally employed to detect objects occupying a

FIGURE 4. CamVid ground truth data. The first row represents the original images and the second one represents the annotated ones.

large portion of images cannot be used [48]. In particular, object boundaries and intersections between objects are very poorly detected or segmented using conventional deep learning methods [20].

A semantic segmentation system that can incorporate two different contexts (indoor and outdoor images) is another major challenge, not only because the images of the datasets are limited but also because images' types are different. A system that is trained to semantically segment indoor images can not perform well on outdoor scenarios and vice versa. This is because datasets' images have different distributions and modalities. We introduce shared systems that incorporate both scenarios. However, the achieved results are not as competent as the achieved results by the individual systems.

There will always be a trade-off between the system's speed and accuracy. As the proposed systems are meant to be deployed on an EPW for environment parsing, we propose using a relatively small backbone network (ResNet-18) that can achieve better Frame Per Second (FPS) compared to ResNet-50 and Xception without sacrificing accuracy thanks to the residual block architecture. Table 1 shows the number of layers and trainable parameters of the tested systems with different base networks.

TABLE 1. The number of layers and trainable parameters of the tested systems with different base networks.

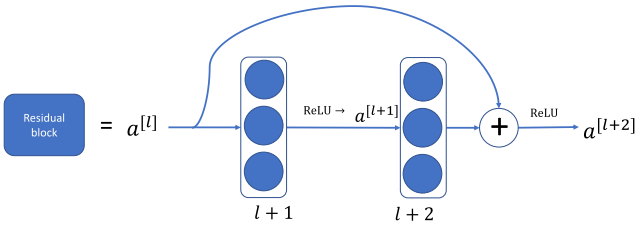
Model \ Metrics	Trainable parameters	Layers
$FCN-8s$	134.3 M	51
$FCN-16s$	134.3 M	47
$FCN-32s$	134.6 M	43
$SegNet\,(VGG-16)$	29.4 M	91
$SegNet\,(VGG-19)$	42.4 M	109
$U-Net$	30.9 M	58
$DLV3+(ResNet-18)$	20.6 M	100
$DLV3+(ResNet-50)$	44.1 M	206
$DLV3+(Xception)$	27.8 M	205
$Shared\ system\ 1$	30.0 M	133
$Shared\ system\ 2$	41.2 M	198
$Shared\ system\ 3$	41.2 M	200

M = Million.

C. SYSTEM ARCHITECTURE

The proposed systems are based on DLV3+ architecture for semantic segmentation [8] with some modifications. The architecture's base network uses residual blocks, which help the system to process high-resolution images ($960 \times 540 \times 3$ pixels) using a deep network (many layers) without losing information because of the vanishing gradients problem. In the original implementation of DLV3+, ResNet-50, ResNet-101 [9], and Xception [47] networks are used as the system's feature extraction network. In this paper, various feature extraction networks have been experimented as the backbone of the systems, besides those used in the original implementation. However, ResNet-18 is the choice due to its small size and fewer parameters compared to its elder sisters (ResNet-50 and ResNet-101). Also, it can produce better FPS and comparable accuracy, as shown in the Section IV section.

Very deep networks suffer from vanishing/exploding gradients [49], [50]. Residual blocks help to mitigate this problem by reusing the activations from previous layers until the adjacent layer learns its weights [9]. This allows the network



$$z^{[l+i]} = W^{[l+i]}a^{[l]} + b^{[l+i]}$$
$$a^{[l+i]} = \sigma(z^{[l+i]})$$

σ: Rectified Linear Unit (ReLU)
$a^{[l]}$: activation of layer l.
$a^{[l+1]}$: activation of layer $l+1$ after applying ReLU.
$a^{[l+2]}$: activation of the linear output of layer $l+2$ ($z^{[l+2]}$)added to the activation of layer l then applying ReLU.

$$\boldsymbol{a}^{[l+2]} = \boldsymbol{\sigma}(\boldsymbol{z}^{[l+2]} + \boldsymbol{a}^{[l]})$$

FIGURE 5. Residual block. The main building block for ResNet-18, ResNet-50, and ResNet-101.

to learn more low-level features without being worried about performance degradation as it goes deep. The architecture elegance is attributed to the short-cut connections that do not add either extra parameters or computational complexity [9]. A residual block structure can be seen in Fig. 5.

Unlike FCN [20], DLV3+ uses the encoder-decoder structure [8]. The encoder part uses the same design of DLV3 [31], which uses dilated convolution 'atrous' to increase the receptive field of the layers. Atrous convolution is used to control the resolution by enlarging the field of view to incorporate a large context without increasing the number of parameters or computation. At the same time, a simple but effective design is used as a decoder network. Combined, they represent the DLV3+ network. The encoder-decoder approach has proved its efficiency to refine object edges, resulting in better accuracy and Intersection over Union (IoU).

This study adopts DLV3+ design but using ResNet-18 as a backbone feature extraction network (Fig. 6). Besides, the input layer is modified to accept large size image inputs with $960 \times 540 \times 3$ pixels. The indoor, outdoor and shared proposed systems are used to semantically segment images of both indoor application-based and outdoor datasets.

Creating a system that can semantically segment indoor and outdoor environments simultaneously is challenging as data distribution and context differ. Also, the size of the datasets in both cases is limited. Consequently, it is challenging for any model to fit both scenarios. Thus, we introduce a novel approach by merging both the indoor and the outdoor systems after the training process of each system individually (Fig. 7). The intuition is to make use of the learned information and weights by both individual systems (indoor and outdoor) without retraining a new system on a new combined dataset. The proposed techniques can help to combine systems from different domains, save training time and resources, and achieve adequate results on different scenarios simultaneously.

Our first trial is depicted in Fig. 7a, which resulted in the proposed shared system 1. The system is constructed as follows: after training both systems (an indoor system on the indoor dataset and an outdoor system on the outdoor dataset), we extracted the feature extraction network (encoder) from one of the systems. Then, we connect this encoder to both decoders of the indoor and the outdoor systems. After that, we concatenate both outputs of both decoders. Lastly, the concatenated output is propagated through a softmax and pixel classification score layers that output the annotated image with the highest confidence score among all of the indoor and the outdoor classes.

The proposed shared system 1 is an end-to-end system that does not need any further post-processing for the output. However, the system performance is highly impacted by the encoder part. This means that if we use the encoder part of the indoor system, the overall shared system performance on the indoor dataset will be better than that on the outdoor dataset and vice versa. Consequently, this system is biased by its encoder part. This leads to the second and third trials which are depicted in Fig. 7b and Fig. 7c, respectively.

In the second trial (Fig. 7b), the encoders and the decoders of the trained indoor and outdoor semantic segmentation systems are included. After up-sampling to the original image size, we concatenate both images using the depth concatenation layer. Then, we add the softmax and the pixel classification score layers. Lastly, the displayed output is the segmented image with the highest pixels' confidence scores across all of the 20 classes (9 indoor and 11 outdoor classes).

Shared system 2 performs well on both the indoor and the outdoor datasets. It is an end-to-end system that does not need any post-processing. However, scoring the pixels with respect to the 20 indoor and outdoor classes of the shared system 2 is more challenging than scoring 9 or 11 classes of the individual indoor and outdoor systems, respectively. It is a highly competitive scoring process between the 20 classes where the uncertainty increases, especially between dominant classes such as 'Sky' and 'Background Wall' from the indoor and the outdoor datasets, respectively. Consequently, the system's performance is adequate but not as good as the individual systems.

The main intuition behind the shared system 3 (Fig. 7c) is to make use of the individual systems' high performances. We use both of the individual indoor and outdoor semantic segmentation systems to parse the same image. Then, we display the highest pixels' confidence scores annotated output to the user. Shared system 3 detailed process is as follows: the encoders of both the indoor and the outdoor systems are included. Similarly, the decoder parts of both systems are included. Using the proposed shared system 3, we obtain two outputs from the two parallel systems. We then apply one post-processing step to determine which output from the two individual systems should be displayed. The mean of each row of the output pixels confidence scores is calculated for both individual systems, resulting in two vectors of means with the same height as the input image. Then, the maximum values of each vector are compared. If the indoor system achieves a maximum value that is higher than that of the

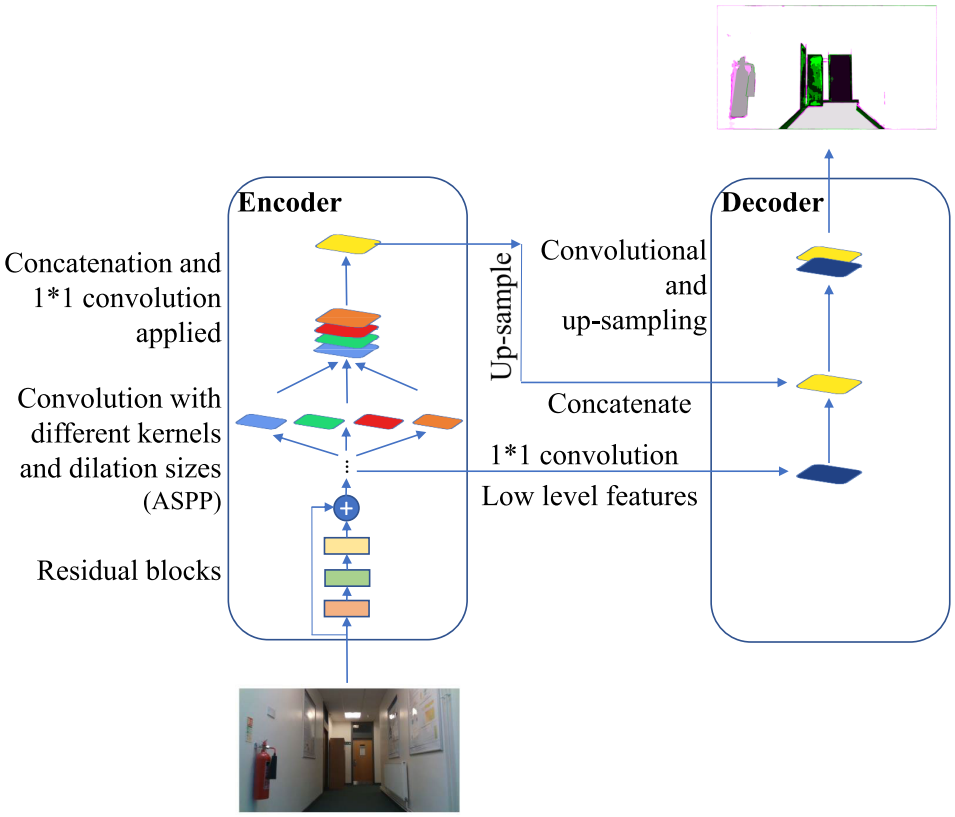


FIGURE 6. System architecture. The encoder part with ASPP and the decoder part with simple bilinear upsampling.

outdoor system, then the input image is assumed to be an indoor image and vice versa. Accordingly, we display the system's output that achieves the highest maximum value.

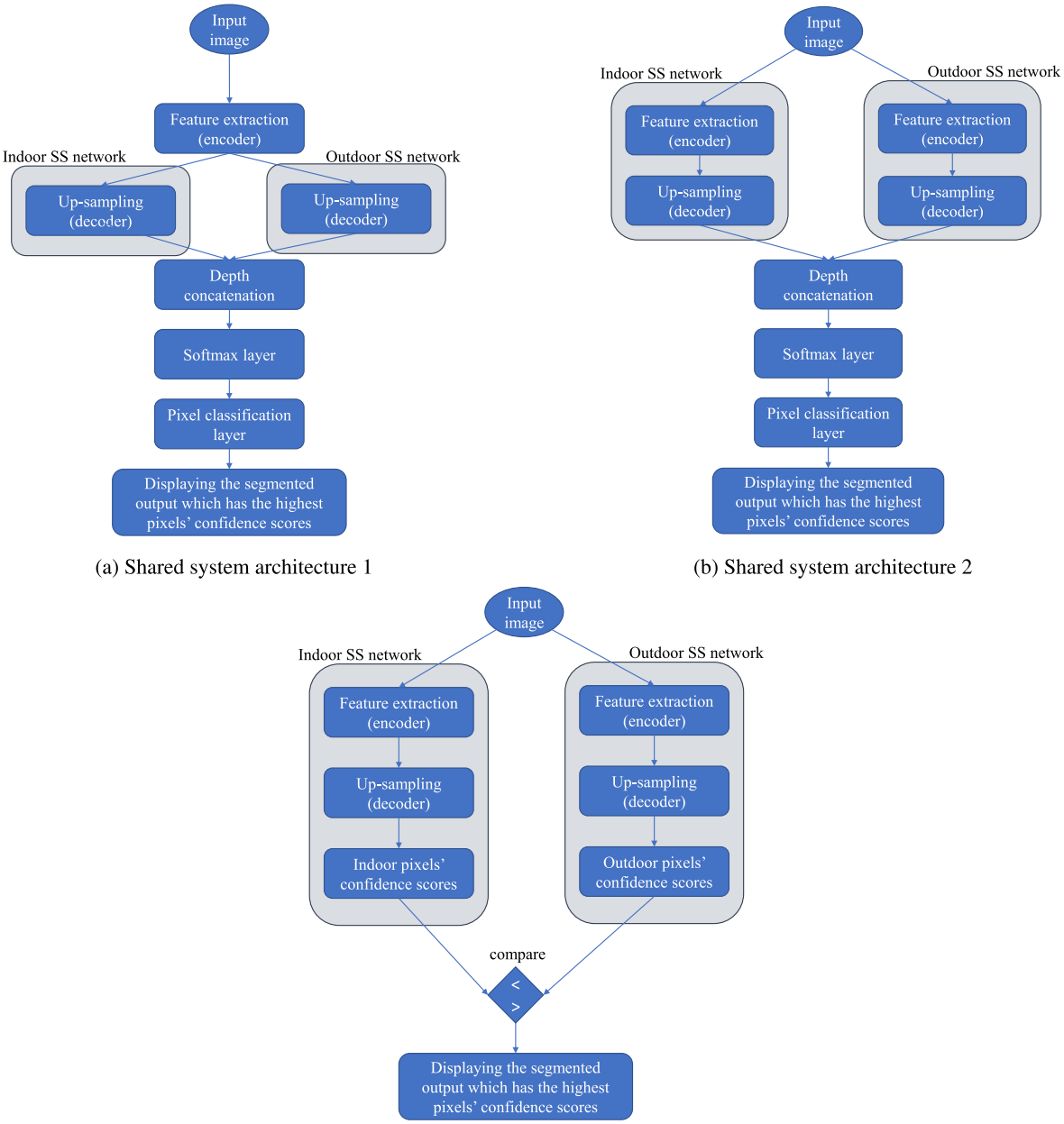
Different comparison techniques of the pixels' confidence scores are tried, for example, comparing pixel by pixel and displaying the system's output that has the highest number of pixels with the highest pixels' confidence scores. However, the 'max(mean(score)' approach has achieved the highest performance.

The proposed shared system 3 needs a post-processing step for the pixels comparison of the two individual systems. As the encoders and decoders of both systems are included, the system inference speed has slowed, which negatively impacts the system's real-time operation. On the bright side, the system can produce better results in both indoor and outdoor environments. One of the study's future work is to explore different system architectures that can enhance the system's inference speed while achieving high performance on two or more scenarios.

D. TRAINING

The indoor and the outdoor systems are trained end-to-end with the following parameters: Stochastic Gradient Descent with Momentum (SGDM) is used as the training optimiser with 0.9 momentum. The Learning rate starts at 0.001 and then drops by a factor of 0.3 every ten epochs. The aforementioned training parameters are chosen after several experiments with different parameters to achieve the best performance. To avoid overfitting, L2 regularisation is used. Training examples are shuffled every epoch to limit sequence memorising and avoid computing the gradients for the same batch of images. Image normalisation is employed to rescale all the pixels' values in the range of zero to one. Lastly, data augmentation with X and Y translations is employed to enhance model generalisation, which can increase the overall system accuracy. To avoid bias in favour of dominant classes, inverse frequency weighting is used to balance the classes weightings. This method increases class weights for under-represented classes. Additionally, different hyper-parameters and optimisation algorithms are tried to achieve the highest performance. Moreover, for reproducibility, systems are trained several times under the same configurations.

The introduced systems are trained on a personal computer with a NVIDIA GeForce RTX 2080. Training time varies as the training process can be stopped early when the loss of the validation dataset plateaus or when it reaches the maximum epochs of the training process (30 epochs). For the indoor dataset, the model's loss is validate every 200 iterations. However, the model's loss is validated every 50 iterations for the outdoor dataset. The difference in the two cases is attributed to the mini-batch size, the sizes of the datasets, and the model's size. The largest mini-batch size that can



(a) Shared system architecture 1

(b) Shared system architecture 2

(c) Shared system architecture 3

FIGURE 7. **Shared network architectures. Shared system 1 uses either the trained feature extraction network (encoder) of the indoor or the outdoor semantic segmentation systems. Shared system 2 uses both feature extraction networks of the indoor and the outdoor systems. Shared system 3 uses the indoor and the outdoor semantic segmentation system simultaneously with an added post-processing step to display the annotated output that has the highest pixels' confidence scores.**

accommodate the available memory is sought. The largest mini-batch sizes are 8 and 4 in the case of the outdoor and the indoor datasets, respectively. The mini-batch size is reduced if the available memory can not accommodate the model size with a large mini-batch size. Consequently, the number of iterations per epoch varies. Table 2 shows the training time of each model, the used mini-batch size, the stopping epoch and the trained model size.

Systems are trained end-to-end using high-resolution and large-size training images of 960 $\times$ 540 $\times$ 3 pixels from the indoor and the outdoor datasets, unlike the original implementation of DLV3+, which crops patches of 513 $\times$ 513 size from the PASCAL VOC dataset [45] images during training and testing. The proposed training approach enhances the system's ability to semantically segment small size objects alongside medium and larger size ones. Also, this boosts the

TABLE 2. Training details.

Model \ Metrics (Indoor/Outdoor)	Training Time (≈hours)	Mini-batch size	Stopping epoch	Model size (MB)
$FCN-8s$	3.5/0.75	2/2	14/6	477
$FCN-16s$	2/1.25	2/2	8/8	477
$FCN-32s$	2.25/2.25	2/2	9/14	478
$SegNet\,(VGG-16)$	5.5/8.5	2/2	19/23	104
$SegNet\,(VGG-19)$	9/8.75	2/2	26/21	142
$U-Net$	3/2.25	2/2	5/6	110
$DLV3+\,(ResNet-18)$	1.5/1.25	4/8	20/26	58.3
$DLV3+\,(ResNet-50)$	1/2	4/8	7/14	141
$DLV3+\,(Xception)$	9/3.5	4/8	17/18	83.4
$Shared\ system\ 1$	-	-	-	76.8
$Shared\ system\ 2$	-	-	-	116
$Shared\ system\ 3$	-	-	-	116.6

MB = Megabyte.

effectiveness of large rate atrous convolutions as its weight can be applied to actual pixels and not to zero paddings.

IV. RESULTS AND DISCUSSION

Average pixels intersection over union (mIoU) is the method employed to evaluate the system's performance. Table 3 shows the detailed results of state-of-the-art systems. The proposed DLV3+ with ResNet-18 based systems have achieved mIoU of 0.572/0.696 and mean BF scores of 0.673/0.772, for the indoor and the outdoor datasets, respectively. BF score measures the alignment of the predicted object boundaries with the true ones.

Both systems have achieved high global and mean accuracy (0.970/0.915 and 0.791/0.874 for the indoor and the outdoor datasets, respectively). Global accuracy is the ratio between correctly classified pixels, regardless of the class, to the total number of pixels. In comparison, mean accuracy represents the correctly classified pixels for each class averaged over all classes.

To ensure the reproducibility of our results, we trained both the indoor and the outdoor systems three times. Images are shuffled and randomly split to guarantee that different images are used for training and testing at each time. Table 4 shows the mean and the standard deviation of both systems' metrics. It can be seen that the proposed systems are robust and can reproduce the results under different conditions.

The detailed results for each class of the indoor dataset are shown in Table 5. It can be observed that objects with bigger sizes and larger numbers of pixels have achieved the highest IoU and BF scores, such as doors and background walls, while smaller objects have achieved the lowest IoU, such as pull and push door handles. This is understandable due to the few instances and pixels per object for small size objects in the proposed indoor dataset. Besides, it is challenging for any tested systems to align the predicted segments with the ground truth ones, reflected by the IoU metric, as these objects are tiny (for example, DLV3+ with ResNet-50 has achieved 0.102 IoU for the push door handle class. Detailed results for different models are shown in the appendix (Supplementary tables and figures)). However, small size objects have achieved satisfactory accuracy and BF score. An adequate BF score is vital to our application as it reflects the system's ability to define object boundaries effectively. This is very important for visually impaired users (Fig. 13).

The outdoor system has achieved similar results (Table 6) to the indoor one as small-sized objects such as pole has achieved the lowest IoU. Whereas medium and big size objects have achieved better IoU and BF scores.

The three-stream model (FCN-8s), which adds two skip connections at layers pool3 and pool4, has achieved better overall results compared to FCN-16s, which add one skip connection at pool4 layer, and the series version of FCN (FCN-32s). In contrast, the deeper version of SegNet (SegNet with VGG-19) is not as accurate as the smaller version (SegNet with VGG-16), similar to DLV3+ with ResNet-18 that can achieve better performance compared to its deeper version (DLV3+ with ResNet-50). It can be concluded that deeper versions of semantic segmentation models do not ensure better performance. U-Net performance is the lowest among the tested systems.

The achieved FPS for DLV3+ with ResNet-18 is better than that of DLV3+ with ResNet-50 and with Xception base networks (Table 7). The accuracy and speed can be enhanced further by increasing the number of small object instances in the proposed dataset and using a newer version of a GPU based board such as the Jetson AGX Xavier board. Also, the proposed shared systems have achieved adequate speed. The most accurate shared system (shared system 3) has achieved the lowest speed among the proposed ones. However, the lowest accurate shared system (shared system 1) has achieved the highest speed amongst the introduced shared systems. Interestingly, the proposed shared systems have achieved higher FPS than state-of-the-art systems such as FCN, SegNet and U-Net. Although the proposed shared systems have more layers, they have less trainable parameters and smaller footprints. Besides, they utilise residual blocks, which can explain their fast inference speed.

Similar observations can be extracted from the indoor confusion matrix shown in Fig. 8. It can be seen that the indoor model is slightly confused to distinguish between pixels of different door handles and key slot. The analogous silver colour and orientation of the door handles can represent a reason for that problem. This can be alleviated by increasing these object instances in the proposed dataset. For the outdoor confusion matrix, there is a slight confusion between the sign symbol and the pole classes, which can be attributed to the similarity of their structure.

Fig. 9 and Fig. 10 show some examples of the indoor and the outdoor systems in action where it can segment the scenery with good accuracy and sharp edges. Three rows of images are shown where the first row represents the ground truth data, the second one shows the model's prediction, and the third one demonstrates the difference between the prediction and the ground truth data. The intense green and magenta colours that are shown in the third row

TABLE 3. Results of running the trained individual models on the test set of the indoor and the outdoor datasets.

Model \ Metrics (Indoor/Outdoor)	Global Accuracy	Mean Accuracy	Mean IoU	Weighted IoU	Mean BF Score
$FCN-8s$	0.963/0.808	0.801/0.771	0.552/0.518	0.953/0.732	0.661/0.621
$FCN-16s$	0.961/0.845	0.785/0.836	0.549/0.600	0.952/0.783	0.652/0.684
$FCN-32s$	0.953/0.813	0.766/0.775	0.538/0.523	0.944/0.740	0.583/0.619
$SegNet\ (VGG-16)$	0.960/0.697	0.804/0.680	0.551/0.453	0.950/0.609	0.658/0.451
$SegNet\ (VGG-19)$	0.956/0.783	0.796/0.755	0.528/0.499	0.946/0.686	0.657/0.501
$U-Net$	0.807/0.535	0.505/0.359	0.314/0.207	0.717/0.418	0.358/0.323
$DLV3+\ (ResNet-18)$	0.970/0.915	0.791/0.874	0.572/0.696	0.963/0.860	0.673/0.772
$DLV3+\ (ResNet-50)$	0.965/0.934	0.788/0.906	0.562/0.748	0.957/0.889	0.622/0.825
$DLV3+\ (Xception)$	0.966/0.911	0.808/0.883	0.560/0.692	0.958/0.856	0.621/0.769

TABLE 4. Mean and standard deviation of three trained models on the indoor and the outdoor datasets.

Metrics \ Model	DLV3+ (ResNet-18) indoor	DLV3+ (ResNet-18) outdoor
Global Accuracy	0.970 ± 0.003	0.919 ± 0.004
Mean Accuracy	0.799 ± 0.007	0.888 ± 0.013
Mean IoU	0.570 ± 0.016	0.703 ± 0.008
Weighted IoU	0.963 ± 0.004	0.868 ± 0.007
Mean BF Score	0.680 ± 0.024	0.781 ± 0.010

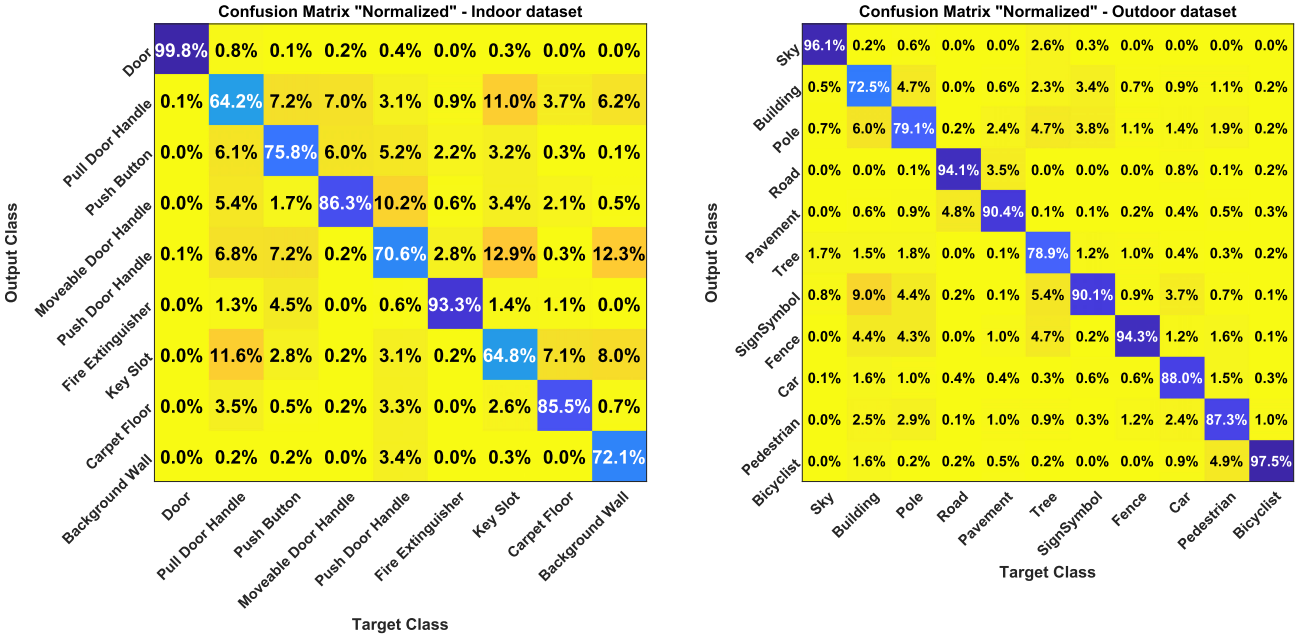


FIGURE 8. Confusion matrices for the indoor and the outdoor systems.

indicate these differences. These pixels are unannotated or misclassified. The green colour shows the unannotated pixels which do not belong to objects of interest. Whereas the magenta one shows the misclassification of some parts of an object.

It can be seen from Fig. 9 that the unannotated pixels in-between two annotated objects which do not belong to either object can represent a challenge to the proposed network. For instance, the indoor system struggles to classify door frame pixels as they do not belong to the door or the wall. Besides, they are not annotated in the proposed dataset. This represents a challenge during inference.

One solution is to annotate door frames as a separate class. Training a semantic segmentation system on a dataset that has some unannotated pixels increases the system's uncertainty. However, annotating every pixel, even if it does not belong to any class of interest, reduces the system's uncertainty because they act as a false positive for the objects of interest. This can be done by annotating all the pixels in an image. This will increase the overall system accuracy and enhance the

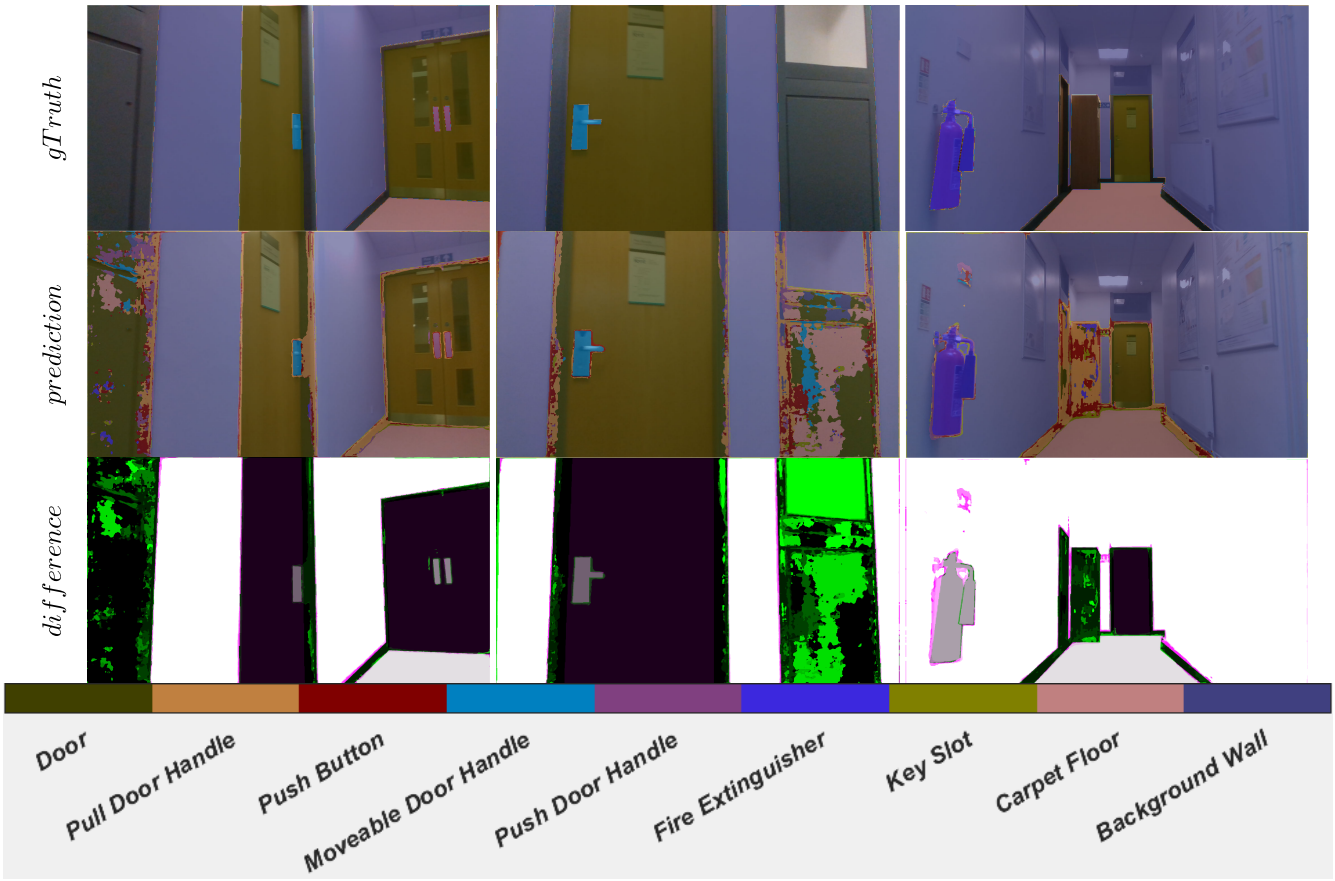


FIGURE 9. Results visualization using the proposed indoor system on the test set. The first row represents the ground truth data, the second row represents the system's output and the third row represents the difference between the ground truth and the prediction.

TABLE 5. Per-class metrics of the indoor system using DLV3+ with ResNet-18 on the test set.

Classes \ Metrics	Accuracy	IoU	Mean BF Score
Door	0.983	0.983	0.870
PullDoorHandle	0.593	0.150	0.593
PushButton	0.790	0.338	0.571
MoveableDoorHandle	0.786	0.665	0.543
PushDoorHandle	0.533	0.090	0.341
FireExtinguisher	0.909	0.889	0.650
KeySlot	0.654	0.186	0.488
CarpetFloor	0.901	0.889	0.751
BackgroundWall	0.967	0.962	0.778

TABLE 6. Per-class metrics of the outdoor system using DLV3+ with ResNet-18 on the test set.

Classes \ Metrics	Accuracy	IoU	Mean BF Score
Sky	0.958	0.937	0.932
Building	0.859	0.835	0.745
Pole	0.765	0.275	0.680
Road	0.952	0.939	0.934
Pavement	0.920	0.783	0.837
Tree	0.919	0.823	0.842
SignSymbol	0.722	0.432	0.592
Fence	0.810	0.624	0.709
Car	0.931	0.820	0.799
Pedestrian	0.873	0.483	0.640
Bicyclist	0.909	0.699	0.732

detection of the objects boundaries. However, the process of annotating every pixel is extremely labouring intense.

Fig. 12 shows the qualitative segmentation comparison between the proposed and state-of-the-art systems. FCN-32s is the series version of FCN with an up-sampling stride of 32 and no skip connections. It is demonstrated that DLV3+ can define object boundaries better than FCN. At the same time, FCN segmentations can be seen as patches with fuzzy boundaries. For example, it is challenging to distinguish the moveable door handle grip from the body in FCN-32s segmentation. Similarly, U-Net could not predict all the pixels correctly, especially small objects such as door handles. Although SegNet defines object boundaries well, the uncertainty pixels around the correctly predicted pixels are high.

On the other hand, the grip in the DLV3+ segmentation is well defined, which facilitates its manipulation using a robotic arm. Table 3 emphasizes the qualitative assessment. Compared to state-of-the-art systems, the proposed DLV3+ models have achieved better mIoU and mean BF scores (contour matching score).

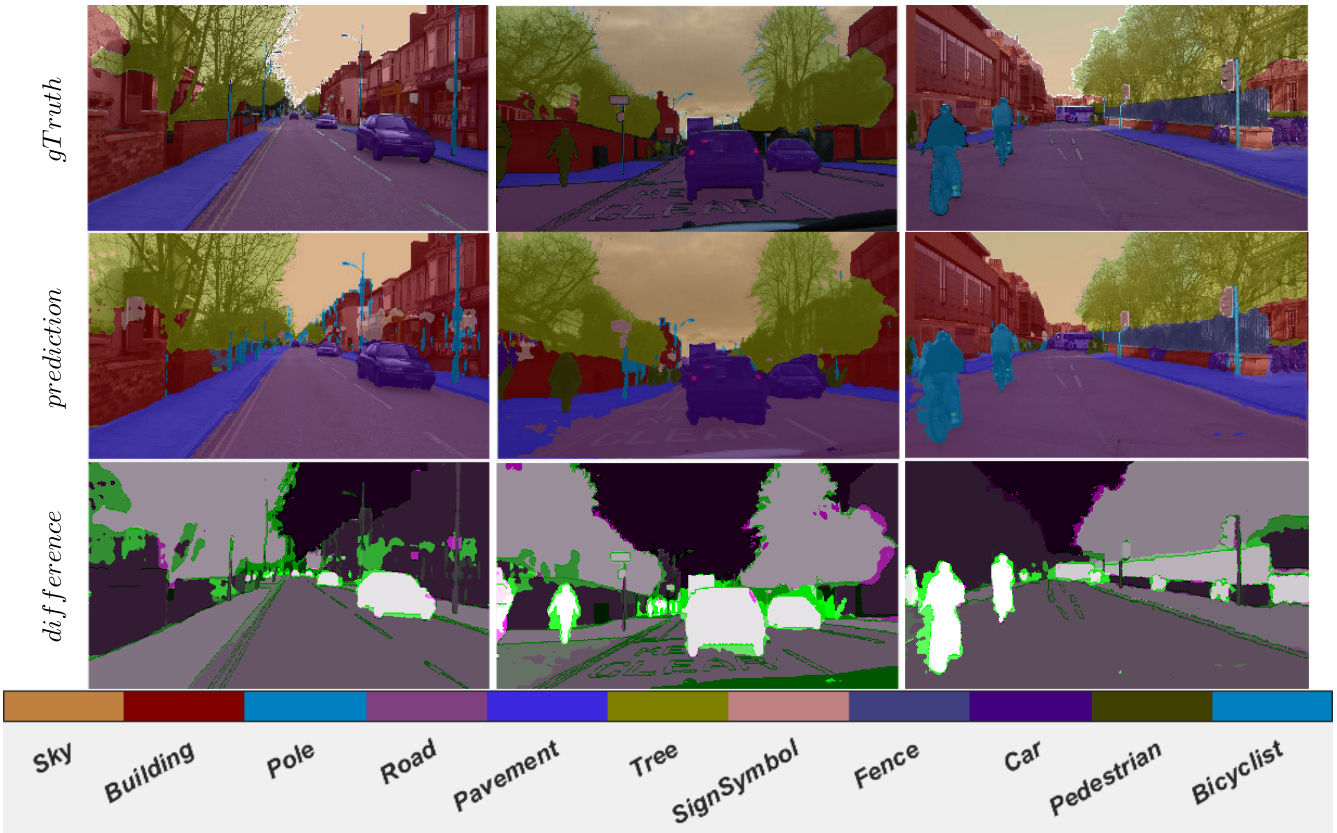


FIGURE 10. Results visualization using the proposed outdoor system on the test set. The first row represents the ground truth data, the second row represents the system's output and the third row represents the difference between the ground truth and the prediction.

TABLE 7. The average speed of the tested models in FPS when deployed on a Jeston TX2 GPU based board.

Model	Speed in FPS
$FCN-8s$	0.86
$FCN-16s$	0.86
$FCN-32s$	0.86
$SegNet\,(VGG-16)$	0.89
$SegNet\,(VGG-19)$	0.72
$U-Net$	0.75
$DLV3+(ResNet-18)$	2.65
$DLV3+(ResNet-50)$	1.57
$DLV3+(Xception)$	2.00
$Sharedsystem1$	1.49
$Sharedsystem2$	1.30
$Sharedsystem3$	1.16

Systems are tested on a never seen before prerecorded video of the indoor environment (from the same distribution of the indoor dataset used for training) and on the CamVid video. TensorRT has been used to optimize systems' inference. The performance of FCN, SegNet, and U-Net is far from real-time execution.

Shared systems 1, 2, and 3 have achieved adequate performance but are not as good as the individual ones (Table 8). For shared system 1 (Fig. 7a), when the encoder of the indoor semantic segmentation system is used, the system has achieved a mean accuracy of 0.676 and 0.456 on the indoor and outdoor datasets, respectively. Also, it has achieved mIoU of 0.591 and 0.300 on the indoor and the outdoor datasets, respectively. Whereas when the encoder of the outdoor semantic segmentation system is used, the system has achieved a mean accuracy of 0.185 and 0.852 on the indoor and the outdoor datasets, respectively. Also, it has achieved mIoU of 0.182 and 0.689 on the indoor and the outdoor datasets, respectively.

Results show that the used encoder has a direct impact on the overall system performance. The encoder of shared system 1, which has been trained on the indoor dataset, can produce better results on the indoor images compared to the outdoor ones and vice versa. This indicates the bias of shared system 1 to the used encoder.

Shared system 2 (Fig. 7b) has achieved mean accuracy and mIoU of 0.594 and 0.555 on the indoor dataset. Whereas it has achieved mean accuracy and mIoU of 0.830 and 0.657 on the outdoor dataset. The performance of the shared system is acceptable. However, the individual systems produce better results. Detailed results of the shared systems are shown in Table 8, where both shared systems 1 and 2 have achieved acceptable Mean BF scores.

As shared system 3 (Fig. 7c) propagates the images through both the individual indoor and outdoor semantic segmentation systems, the shared system's metrics are similar to the individual ones, which are the best-achieved metrics in terms of accuracy, IoU and BF score. However, as shared system 3 compares the pixels' scores of the individual systems (post-processing step), the displayed annotated image is dependant

TABLE 8. Shared systems 1 and 2 detailed metrics.

Model \ Metrics (Indoor/Outdoor)	Global Accuracy	Mean Accuracy	Mean IoU	Weighted IoU	Mean BF Score
Shared system 1 (indoor system's encoder)	0.920/0.582	0.676/0.456	0.591/0.300	0.915/0.506	0.601/0.360
Shared system 1 (outdoor system's encoder)	0.384/0.892	0.185/0.852	0.182/0.689	0.376/0.845	0.365/0.659
Shared system 2	0.725/0.838	0.594/0.830	0.555/0.657	0.725/0.790	0.540/0.608

TABLE 9. Classification capabilities of shared system 3 using different techniques for scores comparison.

Method	Dataset	Classification	
		Indoor	Outdoor
Comparing scores pixel by pixel	Indoor (233)	216	17
	Outdoor (105)	Zero	105
	In+Out (232+105)	206	131
Max(Mean(score))	Indoor (233)	233	Zero
	Outdoor (105)	Zero	105
	In+Out (232+105)	221	116

TABLE 10. Per-class metrics of the indoor system using FCN-8s on the test set.

Classes \ Metrics	Accuracy	IoU	Mean BF Score
Door	0.981	0.979	0.852
PullDoorHandle	0.582	0.159	0.623
PushButton	0.764	0.238	0.631
MoveableDoorHandle	0.780	0.616	0.492
PushDoorHandle	0.622	0.090	0.350
FireExtinguisher	0.897	0.853	0.481
KeySlot	0.722	0.205	0.677
CarpetFloor	0.917	0.883	0.741
BackgroundWall	0.945	0.941	0.647

TABLE 11. Per-class metrics of the indoor system using FCN-16s on the test set.

Classes \ Metrics	Accuracy	IoU	Mean BF Score
Door	0.980	0.979	0.843
PullDoorHandle	0.587	0.128	0.579
PushButton	0.757	0.269	0.540
MoveableDoorHandle	0.748	0.624	0.483
PushDoorHandle	0.598	0.078	0.306
FireExtinguisher	0.896	0.862	0.612
KeySlot	0.638	0.184	0.675
CarpetFloor	0.917	0.881	0.760
BackgroundWall	0.940	0.938	0.824

TABLE 12. Per-class metrics of the indoor system using FCN-32s on the test set.

Classes \ Metrics	Accuracy	IoU	Mean BF Score
Door	0.977	0.977	0.827
PullDoorHandle	0.491	0.123	0.391
PushButton	0.713	0.238	0.425
MoveableDoorHandle	0.795	0.505	0.469
PushDoorHandle	0.596	0.049	0.348
FireExtinguisher	0.903	0.851	0.804
KeySlot	0.506	0.279	0.497
CarpetFloor	0.918	0.893	0.753
BackgroundWall	0.916	0.910	0.800

on that comparison. Table 9 shows the ability of the system to classify the input images as indoor or outdoor ones depending on the pixels' confidence scores using different comparison techniques.

To test the ability of the system to correctly classify the input images as indoor or outdoor ones, we propagate the indoor and the outdoor test sets images through the system. Shared system 3 is able to classify all of the images correctly using Max(Mean(score)) comparison technique described in the system architecture subsection. To obtain more robust results, we shuffled the indoor and the outdoor datasets. Then, the mixed dataset is split randomly into 70% training set, 15% validation set, and 15% testing set. This results in a mix (In+Out) test set with 337 images (232 indoor images and 105 outdoor images). Shared system 3 miss-classified 11 images form the (In+Out) test set as outdoor ones using the Max(Mean(score)) comparison technique (Table 9).

The system's inference speed is dependant on many factors such as the number of trainable parameters, the system's footprint and whether any post-processing techniques are applied. Table 7 shows the speed of the proposed shared systems. Shared system 1 has fewer layers and footprint (Table 1) compared to Shared systems 2 and 3. Consequently, it has achieved the fastest inference speed among the proposed shared systems with 1.49 FPS. Shared system 3 is the slowest with 1.16 FPS. It has the largest footprint and a post-processing step. However, the proposed shared systems' inference speeds are higher than FCN, SegNet and U-Net systems.

Choosing the right system for the right application is a trade-off process between accuracy, inference speed, and the application domain. The deployment of the proposed indoor system can be seen in Fig. 11. The user is controlling the EPW while the information is being displayed on the screen.

FIGURE 11. System deployment. The proposed systems are deployed on an EPW with a display, a Nvidia Jetson TX2 board, and a depth camera.

TABLE 13. Per-class metrics of the indoor system using SegNet with VGG-16 on the test set.

Classes \ Metrics	Accuracy	IoU	Mean BF Score
Door	0.975	0.975	0.830
PullDoorHandle	0.559	0.153	0.642
PushButton	0.751	0.277	0.690
MoveableDoorHandle	0.774	0.614	0.510
PushDoorHandle	0.705	0.075	0.493
FireExtinguisher	0.907	0.839	0.397
KeySlot	0.635	0.190	0.662
CarpetFloor	0.906	0.888	0.659
BackgroundWall	0.947	0.938	0.650

TABLE 14. Per-class metrics of the indoor system using SegNet with VGG-19 on the test set.

Classes \ Metrics	Accuracy	IoU	Mean BF Score
Door	0.977	0.976	0.827
PullDoorHandle	0.542	0.149	0.639
PushButton	0.774	0.201	0.713
MoveableDoorHandle	0.787	0.570	0.576
PushDoorHandle	0.676	0.075	0.322
FireExtinguisher	0.907	0.802	0.368
KeySlot	0.602	0.172	0.696
CarpetFloor	0.894	0.873	0.659
BackgroundWall	0.932	0.927	0.668

TABLE 15. Per-class metrics of the indoor system using U-Net on the test set.

Classes \ Metrics	Accuracy	IoU	Mean BF Score
Door	0.787	0.758	0.603
PullDoorHandle	0.256	0.062	0.267
PushButton	0.092	0.017	0.244
MoveableDoorHandle	0.281	0.140	0.219
PushDoorHandle	0.328	0.043	0.380
FireExtinguisher	0.691	0.591	0.304
KeySlot	0.279	0.039	0.307
CarpetFloor	0.945	0.467	0.579
BackgroundWall	0.877	0.701	0.432

TABLE 16. Per-class metrics of the indoor system using DLV3+ with ResNet-50 on the test set.

Classes \ Metrics	Accuracy	IoU	Mean BF Score
Door	0.974	0.974	0.840
PullDoorHandle	0.555	0.102	0.437
PushButton	0.821	0.234	0.609
MoveableDoorHandle	0.783	0.661	0.460
PushDoorHandle	0.653	0.102	0.276
FireExtinguisher	0.855	0.846	0.582
KeySlot	0.562	0.278	0.557
CarpetFloor	0.915	0.892	0.765
BackgroundWall	0.971	0.965	0.768

TABLE 17. Per-class metrics of the indoor system using DLV3+ with Xception on the test set.

Classes \ Metrics	Accuracy	IoU	Mean BF Score
Door	0.979	0.979	0.853
PullDoorHandle	0.497	0.200	0.467
PushButton	0.759	0.261	0.630
MoveableDoorHandle	0.788	0.617	0.447
PushDoorHandle	0.621	0.0818	0.258
FireExtinguisher	0.932	0.883	0.612
KeySlot	0.843	0.172	0.534
CarpetFloor	0.890	0.886	0.749
BackgroundWall	0.961	0.958	0.752

V. LIMITATIONS OF THE STUDY

In this section, the limitations of the study and means of mitigation are discussed. Model choice is dependant on the application. The system's speed and accuracy are the main concerns of this application. More precisely, the ability of the system to clearly define objects boundaries. It is challenging to develop a model that can achieve significant accuracy with high inference speed. Tolerating high inference rates is acceptable as disabled users do not drive fast due to the speed limitation of the EPW. Consequently, the performance of the proposed system is adequate for the application.

One of the major problems facing semantic segmentation tasks is the ability of the systems to process data from two different distributions. The proposed shared systems offer solutions for this problem by merging the learned features of the two models (the indoor and the outdoor systems). However, solving the multi-model data processing issue has negatively impacted the system's speed and accuracy. Thus, the application should determine its needs and compromises to achieve the best model for a given application.

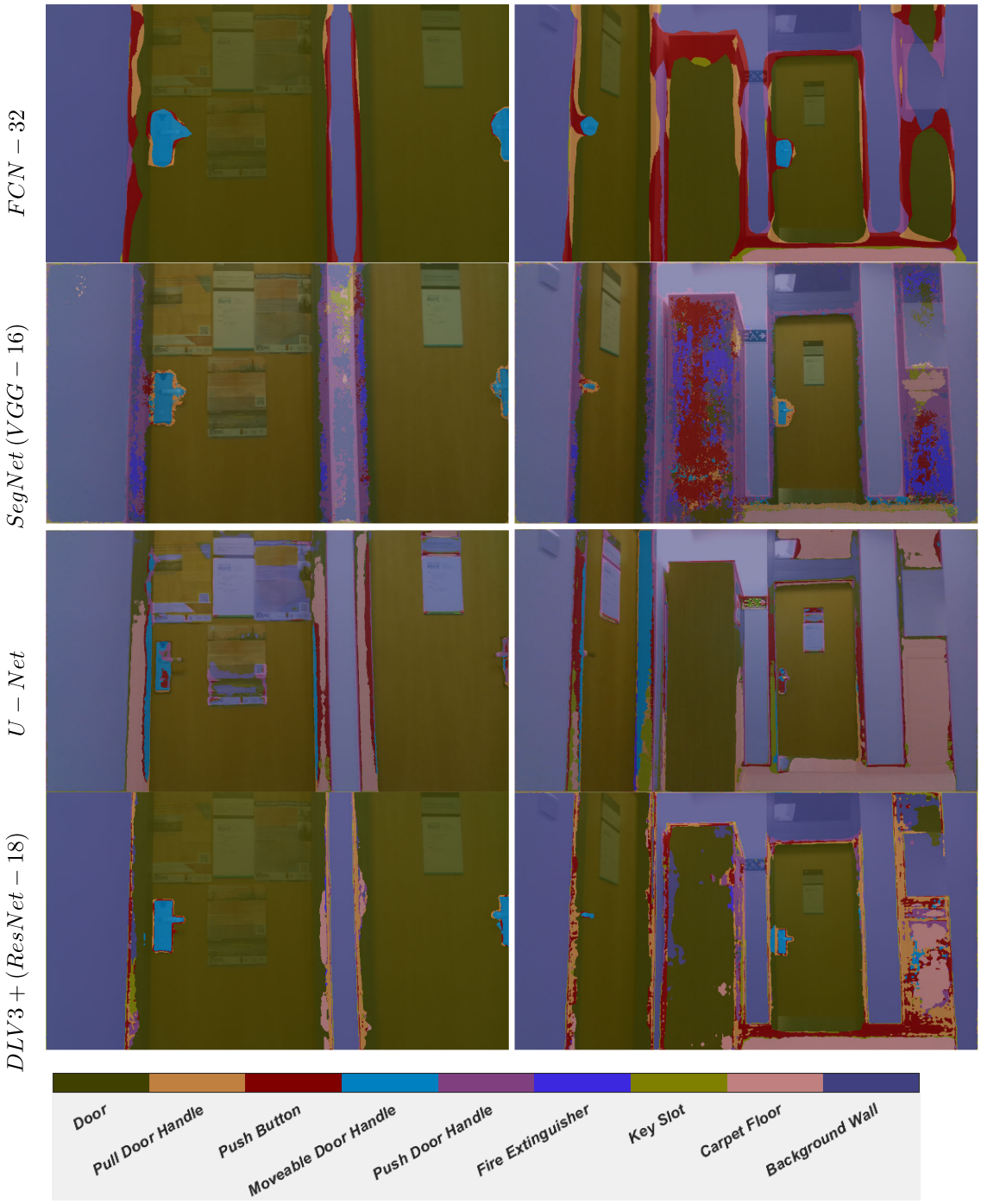


FIGURE 12. Qualitative comparison between the proposed indoor system based on DLV3+ and state-of-the-art systems.

Unannotated pixels of the ground truth data represent a challenge for the proposed semantic segmentation systems. As the systems need to assign each pixel in an image to one of the predefined classes, unannotated pixels, which belong to classes of non-interest, will be assigned to one of the predefined classes. Comparing predicted pixels with the unannotated ones of the ground truth data can result in inaccurate metrics. Usually, these predicted pixels have low confidence scores. We propose to assign the predicted pixels below a specific threshold to a 'Reject' class [51]. Consequently, they can not be included in the evaluation process, resulting in quantitatively and qualitatively accurate predictions.

The future work of our study will concentrate on expanding the proposed dataset, especially small size objects, which can have a positive impact on the overall system's accuracy. Besides, investigating different shared system architectures that can process multi-modal data at high inference speed.

(a) Short-sightedness (b) Semi-neglect

FIGURE 13. Visually impaired users. Illustrated by the clouded areas, short-sightedness users cannot see far object (a), while semi-neglect users cannot see half of the scene (b).

VI. CONCLUSION

In this paper, semantic segmentation systems for indoor and outdoor environments are presented. The proposed pixel classification systems have demonstrated high efficiency with adequate accuracy and BF scores. These systems are intended to help visually impaired EPWs' users to navigate safely and to interact with the environment. Results show the proposed systems' abilities to precisely localize and process images compared to state-of-the-art semantic segmentation techniques. The proposed indoor system has achieved better mean BF scores with 9% and 5% higher than FCN-32s and DLV3+ with ResNet-50, respectively. Whereas the outdoor system has achieved a 15% better mean BF score than the FCN-32s system. The indoor and the outdoor systems have also achieved a processing speed of 2.65 FPS compared to 1.57 FPS and 2 FPS that DLV3+ with ResNet-50 and Xception have achieved, respectively.

The proposed shared systems that can process indoor and outdoor images simultaneously have achieved adequate performance on both tasks. Though, the inference speed and the overall performance is lower than that of the individual systems. Trading-off accuracy and speed with multi-modal data processing is desirable in many applications. Besides, the introduced shared systems do not require any retraining, which is another advantage that makes them flexible and adaptable in many domains. Being able to segment images from two different data distributions simultaneously is challenging. Nevertheless, it is significantly important in many applications that we believe our shared systems can handle. The proposed systems are deployed on a GPU based board and integrated on an EPW for practical usage. Besides expanding the proposed indoor dataset, increasing the accuracy and speed of the systems are the project's future steps.

APPENDIX

SUPPLEMENTARY TABLES AND FIGURES

See Tables 10–17, and see Figs. 12 and 13.

REFERENCES

[1] A. Carlsson and J. Lundälv, "Acute injuries resulting from accidents in, volume ving powered mobility devices (PMDs)—Development and outcomes of PMD-related accidents in Sweden," *Traffic Injury Prevention*, vol. 20, no. 5, pp. 484–491, Jul. 2019, doi: 10.1080/15389588.2019.1606910.

[2] R. P. Gaal, N. Rebholtz, R. D. Hotchkiss, and P. F. Pfaelzer, "Wheelchair rider injuries: Causes and consequences for wheelchair design and selection," *J. Rehabil. Res. Develop.*, vol. 34, no. 1, pp. 58–71, 1997.

[3] W.-Y. Chen, Y. Jang, J.-D. Wang, W.-N. Huang, C.-C. Chang, H.-F. Mao, and Y.-H. Wang, "Wheelchair-related accidents: Relationship with wheelchair-using behavior in active community wheelchair users," *Arch. Phys. Med. Rehabil.*, vol. 92, no. 6, pp. 892–898, Jun. 2011.

[4] L. Fehr, W. E. Langbein, and S. B. Skaar, "Adequacy of power wheelchair control interfaces for perSons with severe disabilities: A clinical survey," *J. Rehabil. Res. Develop.*, vol. 37, no. 3, pp. 353–360, 2000.

[5] (2020). *Office Website ADAPT Project*. Accessed: Oct. 29, 2020. [Online]. Available: http://adapt-project.com/index-en.php

[6] P. Nelson, G. Verburg, D. Gibney, and L. Korba, "The smart wheelchair. A discussion of the promises and pitfalls," in *Proc. 13rd Annu. Conf.*, 1990, pp. 307–308.

[7] G. J. Brostow, J. Fauqueur, and R. Cipolla, "Semantic object classes in video: A high-definition ground truth database," *Pattern Recognit. Lett.*, vol. 30, no. 2, pp. 88–97, 2009.

[8] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 801–818.

[9] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 770–778.

[10] L. Jensen, "User perspectives on assistive technology: A qualitative analysis of 55 letters from citizens applying for assistive technology," *World Fed. Occupational Therapists Bull.*, vol. 69, no. 1, pp. 42–45, May 2014.

[11] Y. Matsumotot, T. Ino, and T. Ogsawara, "Development of intelligent wheelchair system with face and gaze based interface," in *Proc. 10th IEEE Int. Workshop Robot. Hum. Interact. Commun.*, Dec. 2001, pp. 262–267.

[12] G. C. Rascanu and R. Solea, "Electric wheelchair control for people with locomotor disabilities using eye movements," in *Proc. 15th Int. Conf. Syst. Theory, Control Comput.*, 2011, pp. 1–5.

[13] P. Arora, A. Sharma, A. S. Soni, and A. Garg, "Control of wheelchair dummy for differently abled patients via iris movement using image processing in MATLAB," in *Proc. Annu. IEEE India Conf. (INDICON)*, Dec. 2015, pp. 1–4.

[14] M. Henderson, S. Kelly, R. Horne, M. Gillham, M. Pepper, and J.-M. Capron, "Powered wheelchair platform for assistive technology development," in *Proc. 5th Int. Conf. Emerg. Secur. Technol.*, 2014, pp. 52–56.

[15] P. Viswanathan, J. Little, A. Mackworth, and A. Mihailidis, "Adaptive navigation assistance for visually-impaired wheelchair users," in *Proc. Workshop New Emerg. Technol. Assistive Robot.*, 2011, pp. 1–2.

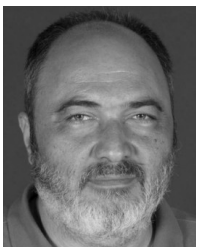
[16] J. Leaman and H. M. La, "A comprehensive review of smart wheelchairs: Past, present, and future," *IIEEE Trans. Human-Mach. Syst.*, vol. 47, no. 4, pp. 486–499, Aug. 2017.

[17] J.-D. Yoder, E. T. Baumgartner, and S. B. Skaar, "Initial results in the development of a guidance system for a powered wheelchair," *IEEE Trans. Rehabil. Eng.*, vol. 4, no. 3, pp. 143–151, Sep. 1996.

[18] P. Viswanathan, J. Boger, J. Hoey, and A. Mihailidis, "A comparison of stereovision and infrared as sensors for an anti-collision powered wheelchair for older adults with cognitive impairments," in *Proc. 2nd Int. Conf. Technol. Aging*, Toronto, ON, Canada, 2007, pp. 1–8.

[19] S. Chatzidimitriadis, P. Oprea, M. Gillham, and K. Sirlantzis, "Evaluation of 3D obstacle avoidance algorithm for smart powered wheelchairs," in *Proc. 7th Int. Conf. Emerg. Secur. Technol. (EST)*, Sep. 2017, pp. 157–162.
[20] E. Shelhamer, J. Long, and T. Darrell, "Fully con, vol.utional networks for semantic segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 640–651, Apr. 2017.
[21] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.
[22] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.
[23] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.
[24] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell, "DeCAF: A deep convolutional activation feature for generic visual recognition," in *Proc. Int. Conf. Mach. Learn.*, 2014, pp. 647–655.
[25] C. Farabet, C. Couprie, L. Najman, and Y. LeCun, "Learning hierarchical features for scene labeling," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1915–1929, Aug. 2013.
[26] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik, "Simultaneous detection and segmentation," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 297–312.
[27] Y. Guo, Y. Liu, T. Georgiou, and M. S. Lew, "A review of semantic segmentation using deep neural networks," *Int. J. Multimedia Inf. Retr.*, vol. 7, no. 2, pp. 87–93, Jun. 2018.
[28] A. Garcia-Garcia, S. Orts-Escolano, S. Oprea, V. Villena-Martinez, and J. Garcia-Rodriguez, "A review on deep learning techniques applied to semantic segmentation," 2017, *arXiv:1704.06857*.
[29] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Semantic image segmentation with deep convolutional nets and fully connected CRFs," 2014, *arXiv:1412.7062*.
[30] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, Apr. 2017.
[31] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," 2017, *arXiv:1706.05587*.
[32] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," 2015, *arXiv:1511.07122*.
[33] D. Eigen and R. Fergus, "Predicting depth, surface normals and semantic labels with a common multi-scale con, vol.utional architecture," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 2650–2658.
[34] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.* Cham, Switzerland: Springer, 2015, pp. 234–241.
[35] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, Dec. 2017.
[36] D. Ciresan, A. Giusti, L. M. Gambardella, and J. Schmidhuber, "Deep neural networks segment neuronal membranes in electron microscopy images," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 2843–2851.
[37] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. pattern Recognit.*, Jun. 2014, pp. 580–587.
[38] H. Noh, S. Hong, and B. Han, "Learning decon, vol.ution network for semantic segmentation," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1520–1528.
[39] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1904–1916, Sep. 2015.
[40] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, and A. Torralba, "Scene parsing through ADE20K dataset," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 633–641.
[41] B. Zhou, H. Zhao, X. Puig, T. Xiao, S. Fidler, A. Barriuso, and A. Torralba, "Semantic understanding of scenes through the ADE20K dataset," 2016, *arXiv:1608.05442*.
[42] N. Silberman and R. Fergus, "Indoor scene segmentation using a structured light sensor," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops (ICCV)*, Nov. 2011, pp. 601–608.
[43] P. K. Nathan Silberman, D. Hoiem, and R. Fergus, "Indoor segmentation and support inference from rgbd images," in *Proc. ECCV*, 2012, pp. 746–760.
[44] B.-S. Hua, Q.-H. Pham, D. T. Nguyen, M.-K. Tran, L.-F. Yu, and S.-K. Yeung, "SceneNN: A scene meshes dataset with aNNotations," in *Proc. 4th Int. Conf. 3D Vis. (3DV)*, Oct. 2016, pp. 92–101.
[45] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal visual object classes (VOC) cHallenge," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, Sep. 2009.
[46] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 740–755.
[47] F. Chollet, "Xception: Deep learning with depthwise separable convolutional," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1251–1258.
[48] E. Mohamed, K. Sirlantzis, and G. Howells, "Application of transfer learning for object detection on manually collected data," in *Proc. SAI Intell. Syst. Conf.* Cham, Switzerland: Springer, 2019, pp. 919–931.
[49] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE Trans. Neural Netw.*, vol. 5, no. 2, pp. 157–166, Mar. 1994.
[50] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," *J. Mach. Learn. Res.*, vol. 9, pp. 249–256, May 2010.
[51] E. Mohamed, K. Sirlantzis, and G. Howells, "Incorporation of rejection Criterion–A novel technique for evaluating semantic segmentation systems," in *Proc. 14th Int. Conf. Hum. Syst. Interact. (HSI)*, Jul. 2021, pp. 1–7.

ELHASSAN MOHAMED received the B.Sc. degree in electronics and communications from Mansoura University, Mansoura, Egypt, in 2011, and the M.Sc. degree (Hons.) in embedded systems and instrumentations form the University of Kent, Canterbury, U.K., in 2016, where he is currently pursuing the Ph.D. degree with the School of Engineering and Digital Arts. He is also a part of the ADAPT Team that is working on developing smart assistive devices for disabled people. His research interests include computer vision, embedded systems, artificial intelligence, and robotics.



KONSTANTINOS SIRLANTZIS is currently an Associate Professor of Intelligent Systems, the Head of the Intelligent Interaction Research Group, and the Academic Lead of the Kent Assistive RObotics Laboratory (KAROL), School of Engineering and Digital Arts, University of Kent. He has published more than 120 peer-reviewed papers and organized international conferences (EST 2019) and thematic sessions (AAATE 2019) on topics of robotic assistive systems. His main research interests include pattern recognition, artificial intelligence, robotics, computer vision, and their application to assistive technology (AT) systems and their security. He successfully gained over 3M in research awards from public and private funders in the U.K. and internationally.

GARETH HOWELLS (Senior Member, IEEE) is currently a Professor of secure electronic systems with the University of Kent, U.K., and the Founder, Director, and Chief Technology Officer of Metrarc Ltd., a university spin-out company. He has been involved in research relating to pattern recognition and image processing for over 30 years and has published over 200 articles in the technical literature, co-editing two books, and contributing to several other edited publications. His core research interests include applying soft computing and pattern recognition techniques to the domains of device authentication, biometrics, secure communications, and identity management.