Toward Parking Spot Occupancy Recognition: A Self-Supervised Approach

Luan Marko Kujavski*, Rayson Laroca†,*, Paulo Lisboa de Almeida*
*Federal University of Paraná, Curitiba, Brazil
†Pontifical Catholic University of Paraná, Curitiba, Brazil
*{luan.marko, paulorla}@ufpr.br †rayson@ppgia.pucpr.br

***Abstract*—As urban areas expand, automatic monitoring of parking lots becomes essential for efficient and sustainable cities. This work proposes a self-supervised approach for parking spot occupancy recognition that requires no labeled samples from the target parking lot. Building upon a self-supervised transfer learning fine-tuning protocol, the proposed training strategy consists of two self-supervised stages: first on unlabeled generic data and then on unlabeled target-specific data, followed by supervised fine-tuning using only generic parking lot labels. We adopt SimCLR with a ResNet-50 encoder and evaluate the method under a leave-one-out cross-environment protocol on three public datasets: PKLot, CNRPark-EXT, and PLds. We also introduce a two-stage deployment strategy in which a *Strong General Model* is initially deployed, followed by a *Specialized Model* that incorporates unlabeled images collected during the first N days of deployment in a self-supervised manner. Experimental results show that the *Strong General Model* alone outperforms supervised and self-supervised baselines, achieving an average accuracy of 97.2%, which further improves to 97.8% with the proposed two-stage strategy. These results demonstrate that self-supervised learning enables a scalable and label-efficient solution for real-world parking occupancy monitoring. Our trained models and source code are publicly available at *https://github.com/LoanMaikon/Parking-Spot-Occupancy-Recognition*.**

***Index Terms*—Self-supervised Learning, Parking Spot Occupancy Recognition, Smart Cities.**

I. Introduction

As urban areas continue to expand in both size and population density, the time drivers spend searching for available parking spots contributes substantially to traffic congestion, increased fuel consumption, and higher carbon emissions [1]. In this context, vision-based automated parking spot occupancy classification (which determines whether a parking space is empty or occupied) emerges as a key enabling technology for more efficient and sustainable urban environments. By enabling real-time monitoring of parking facilities, such systems support faster, data-driven decision-making for both drivers and city management platforms.

Despite recent advances [2]–[8], the high human cost associated with labeling data from the target environment remains a major challenge [5]. In the absence of such labeled data, most models suffer a significant drop in accuracy when deployed in unseen environments [5], [6]. In this work, we propose a self-supervised learning approach, coupled with a dedicated training pipeline, to create robust general and specialized models that do not rely on labeled data from the target environment. To the best of our knowledge, self-supervised learning remains an underexplored family of methods in the context of vision-based parking spot occupancy recognition.

A typical self-supervised evaluation under a transfer learning via fine-tuning protocol consists of a pretraining stage on generic unlabeled data, followed by supervised fine-tuning on task-specific data [9]. We refer to this standard approach as the *Self-supervised Baseline*. Inspired by [10], we extend this protocol by introducing an additional self-supervised fine-tuning stage using domain-specific data (i.e., parking lot images). As illustrated in Figure 1, the proposed training scheme follows a three-stage pipeline: (1) self-supervised pretraining on generic data (ImageNet [11]); (2) self-supervised fine-tuning on parking lot data; and (3) supervised fine-tuning using labeled samples from generic parking lot data.

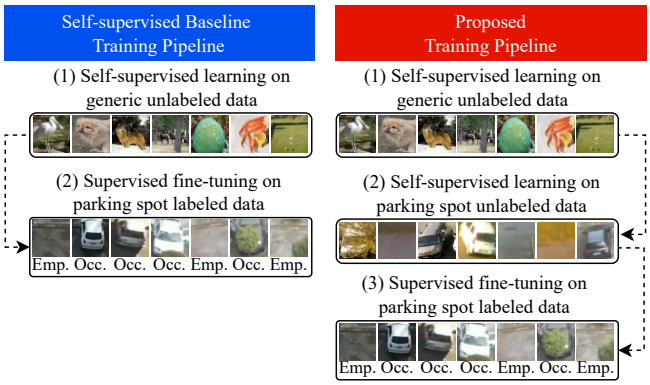


Fig. 1. Overview of the proposed training pipeline (right), which comprises two self-supervised stages followed by supervised fine-tuning. This pipeline extends the standard self-supervised transfer learning protocol (left).

The pipeline shown on the right in Figure 1 yields a *Strong General Model* for generic parking spot classification. Inspired by [8], we further propose and evaluate a two-stage deployment scheme, illustrated in Figure 3. In the first stage, the *Strong General Model* is deployed in the first N days. After the N-th day, a *Specialized Model* is trained using the same pipeline depicted in Figure 1, incorporating unlabeled samples collected from the target parking lot during the first N days together with the generic parking lot data in stage (2).

We evaluate our approach on three public parking spot oc-

This work has been supported by the Brazilian National Council for Scientific and Technological Development (CNPq) – Grant 405511/2022-1.

Accepted at SMC 2026. The final published version will be available in IEEE Xplore.

cupancy recognition datasets: PKLot [2], CNRPark-EXT [3], and PLds [4]. These datasets cover a broad range of real-world conditions, including variations in lighting, weather, and viewpoints [5]. The main findings of this work are summarized in Figure 2. The proposed *Two-stage Deployment* scheme achieves the highest accuracy, followed by the *Strong General Model*. We compare our approach against two baselines: the *Supervised Baseline* and the *Self-supervised Baseline*, which follow standard transfer learning via fine-tuning protocols [9]. These baselines consist of a pretraining stage on generic data, performed either in a supervised or self-supervised manner, followed by supervised fine-tuning on task-specific data.

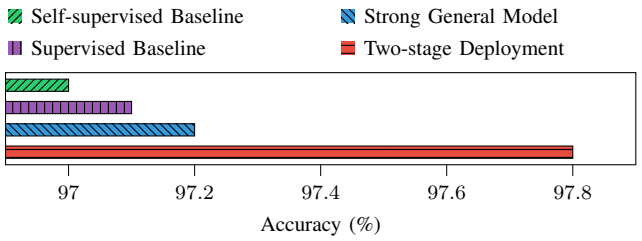


Fig. 2. Summary of the main findings of this work. The reported results correspond to the average accuracy across all experiments.

Our results show that the proposed approach achieves performance comparable to methods that rely on labeled samples, while eliminating the need for labeled data from the target deployment environment, which remains a major bottleneck in parking spot occupancy research [5]. Accordingly, the main contributions of this work are:

- We propose a self-supervised learning-based training pipeline to obtain a strong parking spot occupancy classifier without relying on labeled data from the target.
- We show that unlabeled data collected from the target parking lot can be effectively exploited to train a specialized model tailored to the deployment scenario.

The remainder of this paper is organized as follows. Section II reviews the related literature. The proposed approach is described in Section III. The experimental protocol and results are presented and discussed in Sections IV and V, respectively. Finally, conclusions are drawn in Section VI.

II. Related Work

This Section reviews prior work on self-supervised learning and parking spot occupancy classification, emphasizing the main paradigms and how they motivate the proposed approach.

A. Self-Supervised Learning

Self-supervised learning has recently emerged as a powerful paradigm for representation learning without labeled data. Instead of relying on annotations, these methods employ pretext tasks that enable models to learn invariant features capturing semantic information from large amounts of unlabeled data. Once trained, the learned representations can be transferred to downstream tasks, typically by adding a task-specific head on top of the learned features [9], [12]–[15].

Determining when self-supervised learning was first introduced is challenging. Nevertheless, early work such as [16] demonstrated that algorithms could learn to classify data without explicit external labels by exploiting relationships across multiple modalities. Subsequent approaches focused on learning visual representations by associating different regions of an image, for instance by predicting relative patch positions [17], identifying the correct permutation of shuffled patches [18], or performing inpainting-based prediction tasks [19].

More recent approaches, as discussed in [20], explore strategies such as canonical correlation analysis, in which the model jointly learns feature representations and assigns samples to prototype clusters [13]; self-distillation, where a network learns from its own predictions [14]; and masked image modeling, which trains the model to reconstruct missing or masked regions of the input image [15]. Another line of work, explored in this paper, relies on deep metric learning, where models are trained to pull similar samples closer in the embedding space while pushing dissimilar ones apart [9], [12]. Inspired by [10], we incorporate a deep metric self-supervised approach into the parking spot occupancy classification problem, enabling the exploitation of unlabeled data from the target parking lot and the learning of richer, domain-specific features.

B. Parking Spot Occupancy Classification

Related work on parking spot occupancy classification can be broadly divided into three phases. In the first phase, researchers focused on the creation of large-scale datasets for training and evaluation, including PKLot [2], CNRPark-EXT [3], and PLds [4].

The second phase is characterized by the development of feature extraction and deep learning based methods capable of achieving high accuracy rates, often above 99%, provided that a large number of labeled samples from the target parking lot are available. Representative works include [2], [3], with a comprehensive overview presented in [5]. Despite their strong performance, these approaches are costly, as they require extensive data annotation whenever the system is deployed.

The third and current phase focuses on reducing or eliminating the need for labeled samples from the target environment and on developing lightweight models suitable for deployment on power-restricted platforms, such as edge devices [5]. Representative works include [2], [3], [6]–[8]. The methods proposed in [2] and [3] introduced models based on Support Vector Machines (SVMs) and deep learning, respectively. Although computationally efficient, these models require large amounts of labeled data from the target for training.

The authors in [6] reduced the labeling effort by showing that, with approximately 1,000 labeled samples from the target environment, a model can be fine-tuned to achieve accuracies close to 97%. This gap was reduced in [8], where similar performance was achieved without using labeled data from the target environment. In that work, a teacher-student framework was adopted, in which a teacher model generates pseudo-labels during the first N days of deployment, and a lightweight model is subsequently trained using these pseudo-labels.

In this work, we draw inspiration from [8] and [10] to combine self-supervised learning with the collection of unlabeled data from the target environment. This strategy enables the fine-tuning of a custom model that can be deployed directly on the target device, such as a smart camera, without requiring labeled data from the target environment. Unlike the method proposed in [8], our approach does not rely on a teacher model.

III. Proposed Approach

The proposed approach leverages a self-supervised learning strategy [9], [10] to build two models: a *Strong General Model*, designed to classify instances from diverse parking lots, and a *Specialized Model*, tailored to achieve high performance on a specific target parking lot.

As illustrated in Figure 3, the *Strong General Model*, trained using generic parking lot data, is deployed during the first N days of operation. After the N-th day, a *Specialized Model* is trained by augmenting the initial generic dataset with unlabeled parking spot images collected from the target environment during this initial period. Both models follow the training pipeline depicted in Figure 1 (right) and require no labeled data from the target parking lot, eliminating the need for manual annotation during real-world deployment.

By relying on a pre-trained encoder and a self-supervised learning paradigm, the computational cost of training is substantially reduced, since generic visual representations are learned during pretraining. Both the *Strong General Model* and the *Specialized Model* are initialized from a encoder pre-trained using the SimCLR framework [9]. SimCLR learns visual representations by maximizing agreement between representations of different augmented views of the same image while contrasting them with representations from other images in the batch. For the i-th image in the batch and its corresponding j-th augmented view, the NT-Xent loss is defined as

$$\ell_{i,j} = -\log \frac{\exp(\text{sim}(\mathbf{z}_i, \mathbf{z}_j)/\tau)}{\sum_{k=1}^{2N} \mathbf{1}_{[k \neq i]} \exp(\text{sim}(\mathbf{z}_i, \mathbf{z}_k)/\tau)}, \qquad (1)$$

where $\text{sim}(\cdot, \cdot)$ denotes the cosine similarity between two vectors, τ is a temperature scalar, $\mathbf{z}$ represents the output of the projection head, N is the batch size, and k indexes the k-th view in the batch. As in [9], the encoder is initially pre-trained on the ImageNet dataset [11].

In the second stage, the encoder is further fine-tuned through an additional self-supervised training stage using domain-specific datasets, allowing it to capture task-relevant characteristics. A similar strategy was adopted in [10] for the medical domain. At this point, the training procedures for the *Strong General Model* and the *Specialized Model* diverge. The former is fine-tuned using unlabeled data from generic parking lots. The latter is fine-tuned using the same unlabeled generic data combined with unlabeled data collected from the target parking lot during the first N days of deployment, enabling the incorporation of target-domain information into the self-supervised stage.

Finally, in the third stage, a linear classifier is placed on top of the encoder, and the entire model is fine-tuned in a supervised manner using labeled data from generic parking lots. This procedure is applied to both the *Strong General Model* and the *Specialized Model*. The intuition behind the proposed training pipeline is straightforward. Generic unlabeled images are first used to learn robust and transferable representations. These representations are then refined for the parking lot domain using unlabeled parking spot images. Finally, the resulting domain-adapted features are guided toward the target classes, empty or occupied, through supervised fine-tuning on generic parking lot data.

IV. Experimental Setup

This section describes the datasets, evaluation protocol, and training configuration used in our experiments.

A. Datasets

Following prior work, we use ImageNet [11] for pretraining on general-purpose visual data. For domain-specific data, we employ three parking spot occupancy datasets: PKLot [2], CNRPark-EXT [3], and PLds [4]. These datasets include images captured under different parking lots, cameras, viewing angles, and weather conditions, as summarized in Table I[1].

TABLE I
Summary of the parking-space classification datasets.

Dataset	Annotations	Days	Parking Lots	Angles	Weather Conditions
PKLot [2]	1,199,857	100	2	3	3
CNRPark-EXT [3]	165,513	23	1	9	3
PLds [4]	104,728	115	1	3	5

The PKLot dataset contains images collected at the Universidade Federal do Paraná and the Pontifícia Universidade Católica do Paraná, both in Brazil. It comprises approximately 1.2 million annotated cropped parking space images, with an average resolution of 57×59 pixels. PKLot includes data from three cameras, UFPR04, UFPR05, and PUCPR, and covers three weather conditions: sunny, rainy, and cloudy.

The CNRPark-EXT dataset consists of approximately 165,000 annotated samples collected at the National Research Council in Italy. Each cropped image has an average resolution of 96×91 pixels. The dataset includes cameras 1 to 9 and the same weather conditions as PKLot.

The PLds dataset contains approximately 104,000 labeled samples with an average resolution of 303×108 pixels, collected at the Pittsburgh International Airport. It includes five weather conditions, sunny, rainy, cloudy, snowy, and clear night, as well as three cameras[2]: isshk, qridr, and vxusd/vmlix. Example images from each dataset are shown in Figure 4.

To avoid biases, we follow the evaluation protocol suggested in [5], which is also adopted in [3] and [8]. Specifically, we employ a leave-one-out procedure to assess cross-environment

[1]New annotations were added and previous annotations were standardized to allow a fair comparison of results.

[2]As suggested in [5], we merged vxusd and vmlix.

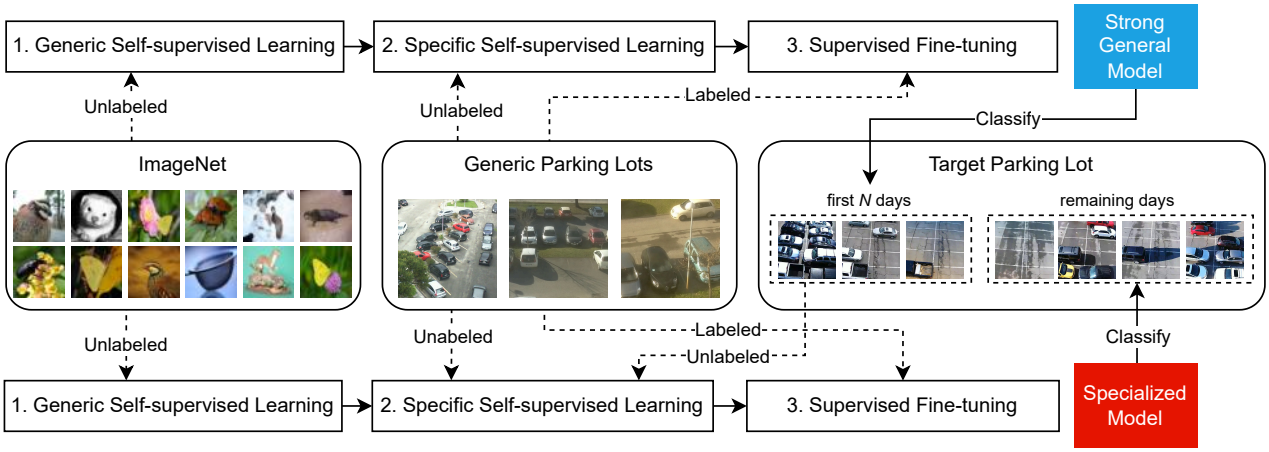
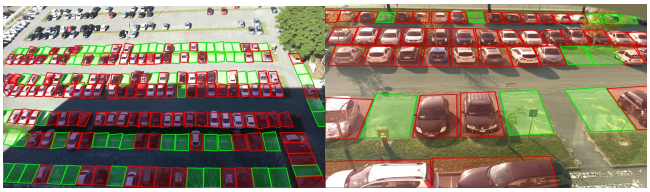


Fig. 3. Proposed two-stage deployment scheme. During the first N days, classification is performed using the *Strong General Model*. Afterward, classification is carried out by the *Specialized Model*, trained by incorporating unlabeled data collected during the first N days into the initial generic parking lot data.



(a) PKLot [2] (b) CNRPark-EXT [3]

(c) PLds [4]

Fig. 4. Examples from the parking lot datasets explored in this work.

generalization. Models are trained on two datasets and evaluated on the remaining one. The evaluation splits are: training on PKLot and CNRPark-EXT and testing on PLds; training on PKLot and PLds and testing on CNRPark-EXT; and training on CNRPark-EXT and PLds and testing on PKLot.

B. Base Model and Hyperparameter Settings

As in [9], we employ the ResNet-50 architecture [21] as the backbone of the SimCLR framework. Both the proposed *Strong General Model* and *Specialized Model* follow a training pipeline composed of two self-supervised stages followed by supervised fine-tuning, as illustrated in Figure 3. All input images are resized to 224×224 and normalized using the ImageNet mean and standard deviation.

We initialize the encoder using publicly available SimCLR weights[3], corresponding to the first self-supervised stage. In the second stage, self-supervised training is performed on task-specific data. Following [10], we employ the LARS optimizer with a learning rate of 0.3, a batch size of 512, a weight decay of 10^{-6}, NT-Xent temperature of $\tau = 0.5$, and project the representation to a 128-dimensional latent space in the non-linear projection head. The total number of training steps is set to 80,000, without the use of learning rate scheduling or warm-up.

The number of steps was determined by temporarily splitting each training set into training and validation subsets, where the smaller dataset was used for validation (e.g., when using PKLot + PLds as the training set, PLds served as the validation set). The network was initially trained for 200,000 steps, and the optimal number of steps was selected by monitoring the validation accuracy using a classifier head added on top of the encoder, with the entire model fine-tuned following the protocol described in [9]. Once the optimal number of steps was identified, the model was retrained on the complete dataset (i.e., training + validation). We adopt the same data augmentation strategy as in [9]: random resized cropping with a uniform scale from 0.08 to 1.0 and an aspect ratio ranging from 3/4 to 4/3, random horizontal flipping ($p = 0.5$), color jittering ($p = 0.8$), random grayscale conversion ($p = 0.2$), and Gaussian blur ($p = 0.5$).

In the third stage, supervised fine-tuning is conducted on task-specific data. Following [10], we use the SGD optimizer with Nesterov momentum of 0.9, trained for 30,000 steps with a batch size of 256. For each training and validation split defined in the second stage, we perform a grid search over learning rates $\in [10^{-3.5}, 10^{-3}, 10^{-2.5}, 10^{-2}, 10^{-1.5}, 10^{-1}, 10^{-0.5}]$ and weight decay values $\in [10^{-5}, 10^{-4}, 10^{-3}, 0]$. No data augmentation is applied during supervised fine-tuning, and both the *Strong General Model* and *Specialized Model* use identical hyperparameter configurations.

We compare our approach against two baselines: a *Supervised Baseline* and a *Self-supervised Baseline*. In both cases, standard ResNet-50 models are trained following conventional transfer learning via fine-tuning protocols, as described in [9]. Each baseline includes an initial pretraining phase on ImageNet, performed either in a supervised[4] or self-supervised

[3]https://github.com/google-research/simclr.

[4]Supervised ResNet-50 weights are available at https://docs.pytorch.org/vision/main/models/generated/torchvision.models.resnet50.

manner. This stage is followed by supervised fine-tuning on task-specific data using the same strategy adopted in the proposed method. For fairness, learning rate and weight decay values are selected through the same grid search procedure.

Finally, following [8], we define $N = 7$ as the number of days before switching from the *Strong General Model* to the *Specialized Model* in all experiments.

V. Results and Discussion

The results reported in this section correspond to the average of five runs. All experiments were conducted on a server equipped with two Intel Xeon Gold 6430 processors, 512 GB of DDR5 DRAM, and an NVIDIA RTX 6000 Ada Generation GPU with 48 GB of VRAM. To simulate a real-world deployment on an edge device, we also conducted experiments on a Raspberry Pi 5 with 8 GB of DRAM.

A. Main Results

Table II presents the results for each leave-one-out split. As the dataset subsets are relatively well balanced, results are reported in terms of accuracy. Both the *Self-supervised Baseline* and *Supervised Baseline* achieved strong performance, with overall average accuracy rates of 97.0% and 97.1%, respectively. These results exceed the average accuracy of 91.8% reported in [5] for cross-parking-lot evaluation scenarios and remain consistently high across all leave-one-out splits, which we primarily attribute to the network depth and the large amount of generic parking lot data available during training.

TABLE II
Accuracy (%) of baselines and proposed methods across multiple parking lot datasets.

Test Subset	Self-supervised Baseline	Supervised Baseline	Strong General Model (ours)	Two-stage Deployment (ours)
		PKLot		
UFPR04	98.1 ± 0.7	97.5 ± 0.8	98.6 ± 0.1	98.9 ± 0.4
UFPR05	96.1 ± 0.7	96.6 ± 0.4	97.3 ± 0.2	97.9 ± 0.1
PUCPR	97.0 ± 0.7	97.3 ± 0.2	97.0 ± 0.1	97.9 ± 0.1
Average	97.0	97.2	97.3	98.0
		CNRPark-EXT		
camera1	96.5 ± 0.1	95.3 ± 0.6	94.7 ± 0.8	95.1 ± 0.5
camera2	99.5 ± 0.1	99.2 ± 0.2	98.9 ± 0.4	99.2 ± 0.3
camera3	98.5 ± 0.1	97.4 ± 0.4	97.9 ± 0.9	98.0 ± 0.5
camera4	97.7 ± 0.1	98.0 ± 0.1	97.7 ± 0.4	97.8 ± 0.2
camera5	95.1 ± 0.2	96.3 ± 0.2	97.1 ± 0.5	97.2 ± 0.4
camera6	96.9 ± 0.1	96.2 ± 0.3	96.3 ± 0.2	96.5 ± 0.2
camera7	94.6 ± 0.2	94.2 ± 0.2	95.2 ± 0.1	95.5 ± 0.5
camera8	98.4 ± 0.1	98.0 ± 0.2	97.8 ± 0.3	97.8 ± 0.5
camera9	94.1 ± 0.1	94.8 ± 0.4	96.2 ± 0.3	96.3 ± 0.4
Average	96.3	96.5	96.7	96.8
		PLds		
isshk	95.8 ± 0.8	95.8 ± 0.9	94.4 ± 1.1	94.4 ± 1.4
qridr	99.2 ± 0.1	98.9 ± 0.5	99.2 ± 0.2	99.1 ± 0.2
vxusd/vmlix	98.0 ± 0.3	98.9 ± 0.2	98.4 ± 0.6	98.1 ± 0.7
Average	97.2	97.6	96.8	96.7
Global Avg.	97.0	97.1	97.2	97.8

The *Strong General Model* column reports classification results obtained using only the trained *Strong General Model*, without switching to the Specialized Model after the N-th day. This approach slightly outperformed the baselines, achieving an overall average accuracy of 97.2%. However, it is observed that the *Strong General Model* struggled on the PLds dataset, yielding results inferior to those of the baselines. This behavior suggests that the generic features extracted by the encoder may not be sufficiently discriminative for this dataset. PLds poses additional challenges due to low-angle image captures (see Figure 4), which often result in occlusions.

The proposed *Two-stage Deployment* scheme, in which the *Strong General Model* is deployed during the first N days and the *Specialized Model* thereafter, achieved the best overall performance, with a global average accuracy of 97.8%. This strategy improves performance across two datasets compared to the *Strong General Model* alone. Nevertheless, on the PLds dataset, the *Two-stage Deployment* underperformed the *Supervised Baseline* by 0.7 percentage points.

Overall, the results obtained by both the *Strong General Model* and the *Two-stage Deployment* indicate that self-supervised learning can achieve strong performance for parking spot occupancy classification. Even without access to unlabeled samples from the target domain, the *Strong General Model* outperformed the baselines on average, and the obtained results are competitive with those reported in previous works. For instance, [8], which also adopts a two-stage deployment strategy, reports average accuracy values of 95.3% for the teacher model and 97.0% for the student model. Our proposed approach, on the other hand, achieved an average accuracy of 97.3% in the first N days with the *Strong General Model*, and 97.9% in the subsequent days using the *Specialized Model*. Similarly, Hochuli et al. [6] reported an accuracy of 90.1% when evaluating on the PKLot dataset.

Finally, we present the results regarding the training and inference times. Training a *Strong General Model* or a *Specialized Model* takes approximately 25 hours on a single GPU. When deployed on a Raspberry Pi, classifying a single parking spot takes an average of 0.2282 seconds, of which 0.2140 seconds correspond to inference and the remaining time to pre-processing steps such as image cropping. Thus, images from a parking lot with 100 visible parking spaces can be processed in under 23 seconds on an edge device. Although this runtime is approximately 20 times slower than that reported in [8], it remains acceptable for periodically refreshing parking-space occupancy status. Nevertheless, we note that processing time may become prohibitive when using more severely power-constrained edge devices.

B. Using Labeled Samples from the Target Parking Lot

Using a strategy similar to that of [7], here we evaluate the label efficiency of the proposed approach when labeled samples from the target parking lot are available. Figure 5 illustrates the relationship between classification accuracy and the number of labeled samples. The labeled samples are taken randomly from the first N days of deployment, and evaluation is performed considering only the remaining days. As can be seen, the *Specialized Model* consistently outperforms all

competing methods across the entire range of labeled samples. As a comparison, the approach proposed in [7] achieves an accuracy of 97%[5] when 1,000 samples from the target are given, while our proposed approach reaches an accuracy of 98.8%.

It is also worth noting that the *Self-supervised Baseline* exhibited a slower rate of improvement and a stronger dependence on labeled data, as its performance with 8,192 samples remains noticeably lower than that of all other models, including the supervised one.

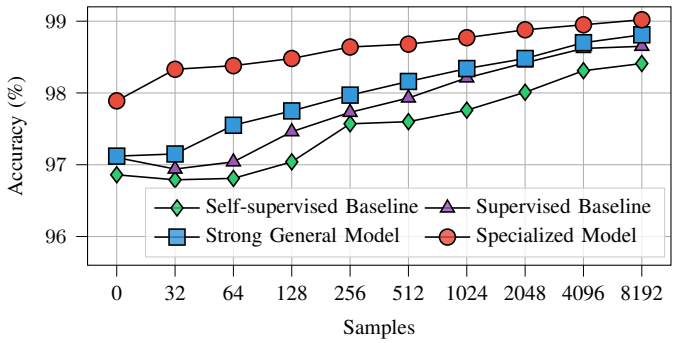


Fig. 5. Accuracy of the models when labeled samples from the target parking lot are given for training. The labeled samples come from the first N days, and evaluation is done in the remaining days.

VI. Conclusions

As discussed in [5], recent advances in image-based parking lot management systems should prioritize methods that achieve high accuracy without relying on labeled samples from the target environment, while remaining lightweight. In this work, we introduce a self-supervised approach for parking spot occupancy recognition and evaluate it in a cross-dataset setting using SimCLR and three widely used datasets. The proposed method is based on a training pipeline composed of two self-supervised phases, one using generic data and another using domain-specific data, followed by supervised fine-tuning with generic labeled parking lot images.

The proposed approach results in a *Strong General Model* that achieves an average accuracy of 97.2%. When sufficient training resources are available (25 GPU hours), a *Specialized Model* can be trained after the N-th day of deployment (the 7th day in our experiments) to replace the *Strong General Model*, yielding an average accuracy of 97.8%. This *Two-stage Deployment* scheme requires no manual annotation, making the approach both practical and scalable.

Regarding inference cost, classifying a single parking space image takes 0.2282 seconds on a Raspberry Pi 5. This result indicates that the proposed approach is suitable for edge computing when relatively capable edge devices are available. Nevertheless, inference latency may still be prohibitive for more resource-constrained platforms.

Although computational demands during training and deployment remain a challenge, the overall results demonstrate that self-supervised learning is an effective strategy for achieving both cross-environment generalization and domain specialization. Future work will explore alternative self-supervised paradigms, different model architectures, and trade-offs between accuracy and efficiency for deployment on edge devices.

[5]This comparison must be considered with caution, since in [7] only the PKLot dataset was used, and the model has considerably fewer parameters.

References

[1] V. Paidi, H. Fleyeh, J. Håkansson, and R. G. Nyberg, "Smart parking sensors, technologies and applications for open parking lots: a review," *IET Intelligent Transport Systems*, vol. 12, no. 8, pp. 735–741, 2018.

[2] P. R. de Almeida, L. S. Oliveira, A. S. Britto, E. J. Silva, and A. L. Koerich, "Pklot – a robust dataset for parking lot classification," *Expert Systems with Applications*, vol. 42, no. 11, pp. 4937–4949, 2015.

[3] G. Amato, F. Carrara, F. Falchi, C. Gennaro, C. Meghini, and C. Vairo, "Deep learning for decentralized parking lot occupancy detection," *Expert Systems with Applications*, vol. 72, pp. 327–334, 2017.

[4] R. Martín Nieto, Á. García-Martín, A. G. Hauptmann, and J. M. Martínez, "Automatic vacant parking places management system using multicamera vehicle detection," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 3, pp. 1069–1080, 2019.

[5] P. R. L. De Almeida, J. H. Alves, R. S. Parpinelli, and J. P. Barddal, "A systematic review on computer vision-based parking lot management applied on public datasets," *ESWA*, vol. 198, p. 116731, 2022.

[6] A. G. Hochuli, J. Barddal, G. Palhano, L. Mendes, and P. L. de Almeida, "Deep single models vs. ensembles: Insights for a fast deployment of parking monitoring systems," in *ICMLA*, 2023, pp. 1379–1384.

[7] A. G. Hochuli, A. S. Britto, P. R. de Almeida, W. B. Alves, and F. M. Cagni, "Evaluation of different annotation strategies for deployment of parking spaces classification systems," in *IJCNN*. IEEE, 2022, pp. 1–8.

[8] P. L. Alves, A. Hochuli, L. E. de Oliveira, and P. L. de Almeida, "Optimizing parking space classification: Distilling ensembles into lightweight classifiers," in *ICMLA*, 2024, pp. 1016–1020.

[9] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *International Conference on Machine Learning (ICML)*, 2020, pp. 1597–1607.

[10] S. Azizi *et al.*, "Big self-supervised models advance medical image classification," in *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pp. 3478–3488.

[11] O. Russakovsky *et al.*, "ImageNet Large Scale Visual Recognition Challenge," *IJCV*, vol. 115, no. 3, pp. 211–252, 2015.

[12] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 9729–9738.

[13] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, and A. Joulin, "Unsupervised learning of visual features by contrasting cluster assignments," *Advances in Neural Information Processing Systems*, vol. 33, pp. 9912–9924, 2020.

[14] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin, "Emerging properties in self-supervised vision transformers," in *IEEE/CVF ICCV*, 2021, pp. 9650–9660.

[15] M. Assran, Q. Duval, I. Misra, P. Bojanowski, P. Vincent, M. Rabbat, Y. LeCun, and N. Ballas, "Self-supervised learning from images with a joint-embedding predictive architecture," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 15 619–15 629.

[16] V. de Sa, "Learning classification with unlabeled data," in *Advances in Neural Information Processing Systems*, vol. 6, 1993.

[17] C. Doersch, A. Gupta, and A. A. Efros, "Unsupervised visual representation learning by context prediction," in *IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 1422–1430.

[18] M. Noroozi and P. Favaro, "Unsupervised learning of visual representations by solving jigsaw puzzles," in *ECCV*, 2016, pp. 69–84.

[19] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros, "Context encoders: Feature learning by inpainting," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2536–2544.

[20] R. Balestriero, M. Ibrahim, V. Sobal, A. Morcos, S. Shekhar, T. Goldstein, F. Bordes, A. Bardes, G. Mialon, Y. Tian *et al.*, "A cookbook of self-supervised learning," *arXiv preprint arXiv:2304.12210*, 2023.

[21] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.