

Imbalance-Aware MobileNet for Automated Classification of Radio Technologies in Spectrograms

Eliezer S. Flores, Paulo R. B. da Silva, Rodrigo Florindo, Rubem V. Pacelli, Filipe J. de Souza, Eric M. Delgado, Fabio S. Lobão, Marcelo L. Nunes, Dimas I. Alves, Felix Antreich, and Renato Machado

Abstract—This paper investigates the automatic classification of radio access technologies from spectrograms using the MobileNetV3-Small architecture. To address the large variability in image size, a spectrogram partitioning strategy is adopted to preserve relevant time-frequency patterns while enabling compatibility with the classifier input. In addition, the impact of class imbalance is analyzed using a weighted loss formulation. The results indicate that the proposed approach achieves effective classification performance at a low computational cost, making it suitable for resource-constrained scenarios and improving recognition of underrepresented classes.

Keywords—Radio access technology classification, spectrum monitoring, MobileNetV3, class imbalance.

I. INTRODUCTION

The increasing complexity of wireless communication systems and the growing demand for efficient spectrum utilization have intensified the need for automated signal analysis techniques [1]. In this context, identifying radio technologies from observed signals is important for applications such as spectrum monitoring and regulatory enforcement [2].

A common strategy for addressing this problem is to represent the received signal in the time-frequency domain, so that relevant transmission patterns can be analyzed visually or automatically. In particular, spectrograms provide a joint time-frequency description that is well suited to the characterization of radio emissions [3]. This representation also enables casting the problem as an image-based classification problem.

More recently, deep learning methods [4] have enabled the development of data-driven solutions that learn discriminative features directly from signal representations, reducing dependence on manually designed descriptors. In the context of wireless signal analysis, Convolutional Neural

Networks (CNNs) have shown promising performance when applied to spectrogram-based inputs [5].

Despite these advances, practical deployment still requires models that combine satisfactory classification performance with low computational cost. This requirement is particularly relevant in embedded or resource-constrained scenarios associated with large-scale spectrum monitoring. In this regard, lightweight architectures such as MobileNet have attracted considerable attention due to their favorable balance between efficiency and accuracy [6]. Among them, MobileNetV3 introduces architectural refinements that improve this trade-off, making it a suitable candidate for edge-oriented applications [7].

Motivated by these considerations, this paper addresses the problem of radio technology classification from spectrogram images. The task is formulated as a multiclass image classification problem, where each input corresponds to a time-frequency representation of a signal, and the objective is to predict the underlying radio technology.

More specifically, this study investigates the use of MobileNetV3-Small, combined with a segmentation strategy, for wide spectrograms to mitigate distortions caused by aggressive resizing. In addition, the effect of class imbalance is analyzed by comparing training procedures with and without sample weighting.

The remainder of this paper is organized as follows. Section II presents the MobileNet architecture adopted in this work. Section III describes the dataset and methodological aspects. Section IV reports and discusses the experimental results. Finally, Section V concludes the paper.

II. MOBILENETV3 ARCHITECTURE

Convolutional Neural Networks (CNNs) have achieved remarkable success in image classification tasks; however, their deployment in real-world applications is often limited by computational and memory constraints [4]. To address these challenges, lightweight architectures such as MobileNet were proposed, aiming to provide efficient models suitable for embedded and mobile environments [6].

MobileNetV3 represents an evolution of this family, incorporating architectural refinements obtained through neural architecture search and hardware-aware optimization [7]. These improvements enable a more favorable trade-off between accuracy and computational efficiency when compared to previous versions of the model.

Eliezer S. Flores and Filipe J. de Souza are with the Federal University of Pampa (UNIPAMPA), Alegrete-RS, Brazil.

E-mails: {eliezerflores, filipejose.aluno}@unipampa.edu.br

Eliezer S. Flores, Paulo R. B. da Silva, Rodrigo Florindo, Rubem V. Pacelli, Dimas I. Alves, Felix Antreich, and Renato Machado are with the Aeronautics Institute of Technology (ITA), São José dos Campos-SP, Brazil. E-mails: {eliezer, pbranco, florindo, rubemrvm, dimasirion, fean, rmachado}@ita.br

Eric M. Delgado, Fabio S. Lobão, and Marcelo L. Nunes are with the National Telecommunications Agency (ANATEL), Brazil. E-mails: {eric, lobao, mlnunes}@anatel.gov.br

This work was supported in part by the National Institute of Science and Technology for Signals (INCT Signals), with funding from the National Council for Scientific and Technological Development (CNPq), under Grant 406517/2022-3 and 305064/2025-8 (PQ-C), the National Telecommunications Agency (ANATEL), and the São Paulo Research Foundation (FAPESP), under Grant 20/09838-0 (BIOS – Brazilian Institute of Data Science).

The architecture is primarily based on depthwise separable convolutions, which decompose a standard convolution into depthwise and pointwise operations, significantly reducing the computational cost while preserving representational capacity. In addition, MobileNetV3 employs inverted residual bottleneck blocks with linear bottlenecks [8], which improve gradient propagation and facilitate efficient feature reuse. The model also incorporates squeeze-and-excitation mechanisms [9] in selected blocks, allowing channel-wise feature recalibration, as well as efficient nonlinear activation functions, such as ReLU and Hard-Swish.

Table I presents the layer configuration reported for MobileNetV3-Small in [7]. As can be seen, the network is organized as a sequence of convolutional and bottleneck blocks, with progressively transformed feature maps. This design allows the extraction of increasingly abstract representations while keeping the computational burden compatible with mobile and embedded applications.

TABLE I: Configuration of the MobileNetV3-Small architecture, based on Table 2 of [7]. Here, SE denotes squeeze-and-excitation, NL the nonlinear activation function, HS Hard-Swish, ReLU the rectified linear unit, NBN the absence of batch normalization, and K the number of output classes.

Input	Operator	Exp size	#Out	SE	NL	s
$224^2 \times 3$	conv2d, 3×3	–	16	–	HS	2
$112^2 \times 16$	bneck, 3×3	16	16	✓	ReLU	2
$56^2 \times 16$	bneck, 3×3	72	24	–	ReLU	2
$28^2 \times 24$	bneck, 3×3	88	24	–	ReLU	1
$28^2 \times 24$	bneck, 5×5	96	40	✓	HS	2
$14^2 \times 40$	bneck, 5×5	240	40	✓	HS	1
$14^2 \times 40$	bneck, 5×5	240	40	✓	HS	1
$14^2 \times 40$	bneck, 5×5	120	48	✓	HS	1
$14^2 \times 48$	bneck, 5×5	144	48	✓	HS	1
$14^2 \times 48$	bneck, 5×5	288	96	✓	HS	2
$7^2 \times 96$	bneck, 5×5	576	96	✓	HS	1
$7^2 \times 96$	bneck, 5×5	576	96	✓	HS	1
$7^2 \times 96$	conv2d, 1×1	–	576	–	HS	1
$7^2 \times 576$	pool, 7×7	–	–	–	–	1
$1^2 \times 576$	conv2d 1×1 , NBN	–	1024	–	HS	1
$1^2 \times 1024$	conv2d 1×1 , NBN	–	K	–	–	1

It is worth noting that MobileNetV3 is available in different sizes. Large configurations generally provide higher representational capacity, whereas small configurations are designed to reduce computational complexity while maintaining competitive performance. In this work, the MobileNetV3-Small variant is adopted to enable efficient deployment in resource-constrained scenarios and to facilitate real-time inference.

III. MATERIALS AND METHODS

This section presents the dataset and the proposed method used in this work. It first introduces the ElectroSense PSD Spectrum Dataset [10] and then details the proposed method.

A. Dataset

The dataset used in this work is the publicly available ElectroSense PSD Spectrum Dataset. According to its official

repository, this dataset contains real-world spectrum measurements collected through RTL-SDR sensors, with full-spectrum scans spanning from 24 MHz to 1.7 GHz. Subsequently, specific licensed bands were filtered and labeled to support wireless technology classification tasks. More broadly, these data are part of the ElectroSense initiative, an open and large-scale crowdsensing platform for spectrum monitoring [11].

The dataset comprises 229 spectrogram images from six radio access technology classes: DAB (32), DVB-T (33), FM (41), GSM (45), LTE (40), and TETRA (38). Figure 1 presents representative spectrogram segments from these classes, highlighting the diversity of time-frequency patterns across the different technologies.

A key characteristic of this dataset is the significant variability in image dimensions, particularly along the horizontal axis, which reflects the temporal axis of the observed signals. An exploratory analysis shows that the height of the spectrograms remains relatively stable, with a mean value of approximately 424.96 pixels and a standard deviation of 54.61. In contrast, the width varies substantially, ranging from 74 to 168,013 pixels, with a mean of 3,336.44 and a standard deviation of 11,600.16.

This large variation makes direct resizing to a fixed resolution unsuitable, as it may introduce severe distortions in the time-frequency structure of the signals. Therefore, a dedicated preprocessing strategy is required to preserve relevant patterns while enabling compatibility with convolutional neural networks.

B. Proposed Method

An overview of the proposed method is illustrated in Figure 2. The pipeline consists of two main stages: (i) partitioning and resizing the input spectrograms, and (ii) classifying the resulting segments using MobileNetV3-Small with a weighted loss formulation. These stages are detailed next.

Spectrogram images in the ElectroSense dataset exhibit relatively consistent dimensions along the vertical axis (frequency), while showing high variability in the number of columns, reflecting the temporal extent of the signals. To handle this variability, each input spectrogram is divided into multiple horizontal segments, each containing 215 columns, consistent with the 2-MHz acquisition granularity originally adopted in [10]. This partitioning resulted in a total of 3408 segments, comprising 1076 DAB, 824 DVB-T, 494 FM, 182 GSM, 631 LTE, and 201 TETRA segments. These segments are subsequently resized to 224×224 pixels, matching the input size required by the classifier. Each segment is then individually rescaled to the full $[0, 255]$ range. This procedure reduces distortions arising from direct resizing while preserving relevant local time-frequency structures.

The classification stage is based on the MobileNetV3-Small architecture, whose convolutional backbone is initialized with weights pre-trained on the ImageNet dataset [12] and kept fixed during training. This strategy allows the model to leverage previously learned visual representations while reducing the number of trainable parameters, which is a commonly adopted practice, especially when the target dataset is relatively limited in size [13].

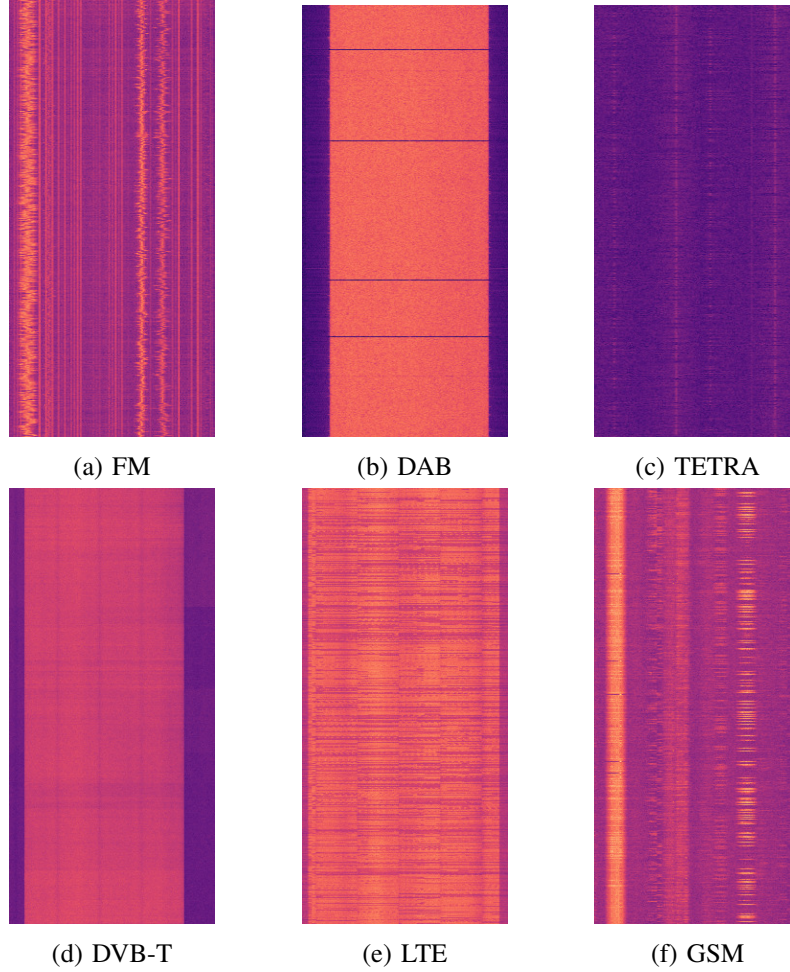


Fig. 1: Examples of spectrogram segments from the ElectroSense dataset.

To adapt the pre-trained network to the target classification task, the original ImageNet-oriented fully connected classification layers of MobileNetV3-Small are removed and replaced with a task-specific classification head. More specifically, the output feature maps of the convolutional backbone are first aggregated by global average pooling, followed by batch normalization and dropout regularization with rate 0.2. Next, a fully connected layer with 300 neurons and ReLU activation is employed, followed by an additional dropout layer with rate 0.3. Finally, a dense output layer with softmax activation and 6 neurons is used to match the number of radio technology classes considered in this work.

The model was trained for up to 100 epochs using the AdamW optimizer, with an initial learning rate of 3×10^{-4} , weight decay of 10^{-4} , and a batch size of 32. Each training segment is randomly perturbed in contrast and brightness by up to 30%. To improve training stability and reduce overfitting, the learning rate was multiplied by 0.5 when the validation loss did not improve for 10 consecutive epochs, down to a minimum value of 10^{-6} . Early stopping was also adopted by monitoring the validation loss, with a patience of 25 epochs and restoration of the best model weights. These hyperparameters were defined empirically based on the observed training and validation behavior.

To address class imbalance during training, a weighted cross-entropy loss is adopted. Let N be the total number of segments and K the total number of classes. For a training segment indexed by $i \in \{1, \dots, N\}$, let $y_i \in \{0, \dots, K-1\}$ and $\mathbf{p}_i = (p_{i,0}, \dots, p_{i,K-1})$ denote the ground-truth class label and predicted class-probability vector, respectively. The standard multiclass cross-entropy loss is defined as

$$\ell(y_i, \mathbf{p}_i) = -\log(p_{i,y_i}), \quad (1)$$

where p_{i,y_i} is the predicted probability assigned to the correct class. The average loss is given by

$$\mathcal{L} = \frac{1}{N} \sum_{i=1}^N \ell(y_i, \mathbf{p}_i). \quad (2)$$

To compensate for class imbalance, the proposed method employs the weighted loss

$$\mathcal{L} = \frac{1}{N} \sum_{i=1}^N w_{y_i} v_{r_i} \ell(y_i, \mathbf{p}_i), \quad (3)$$

where w_{y_i} is the class weight and v_{r_i} the spectrogram weight of the i -th segment, defined as

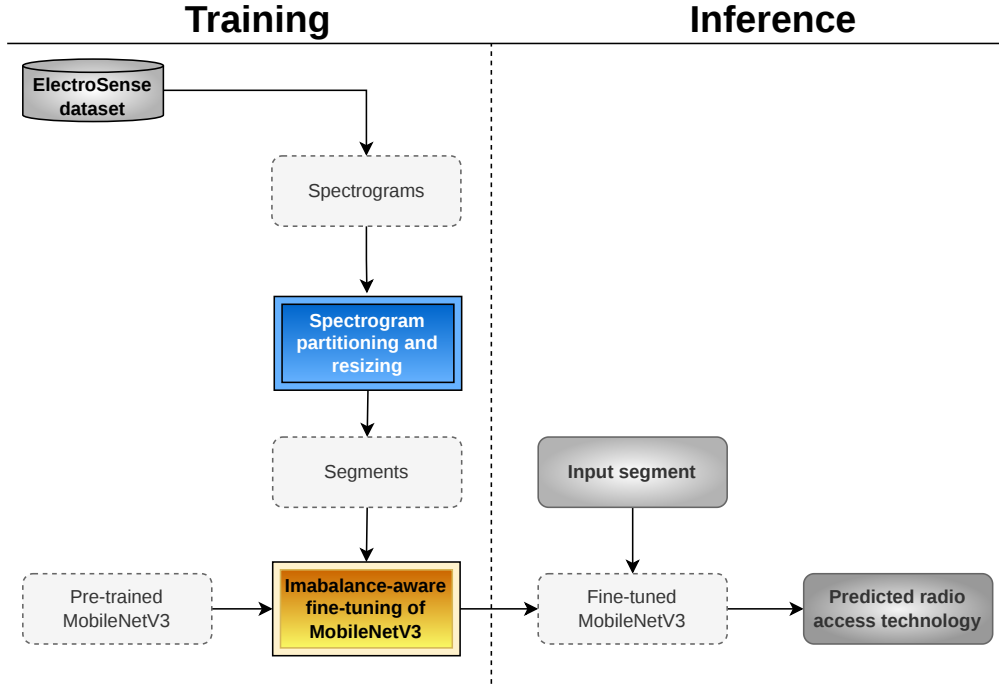


Fig. 2: Overview of the proposed pipeline.

$$w_c = \frac{N}{Kn_c}, \quad v_r = \frac{n_c}{m_cn_r}, \quad (4)$$

with n_c and m_c denoting the number of training segments and spectrograms of class c , and n_r the number of segments of a spectrogram r of class c (at most 60 are retained from each spectrogram). Underrepresented classes, and underrepresented spectrograms within each class, thus receive larger weights during optimization.

IV. EXPERIMENTAL RESULTS

In this section, the experiments and corresponding results are presented and discussed. All experiments were conducted on a workstation equipped with an AMD Ryzen 9 5900X processor, an NVIDIA GeForce RTX 4090 GPU with 24 GB of VRAM, 64 GB of DDR4 RAM, and a 4 TB M.2 SSD. For reproducibility purposes, all source code developed in this work is publicly available at <https://github.com/eli-izersflores/imbalance-aware-rat-spectrogram-classification>.

For the experimental evaluation carried out in this work, the original spectrograms were partitioned into training, validation, and test subsets. First, a stratified split based on the class labels was applied to assign 80% of the original spectrograms (183) to the training-validation subset and the remaining 20% (46) to the final test subset. Then, 20% of the spectrograms of each class were randomly assigned to the validation subset (36), leaving 146 for training; one spectrogram was too narrow to produce any segment. Since the splitting procedure is performed at the level of the original spectrograms, all

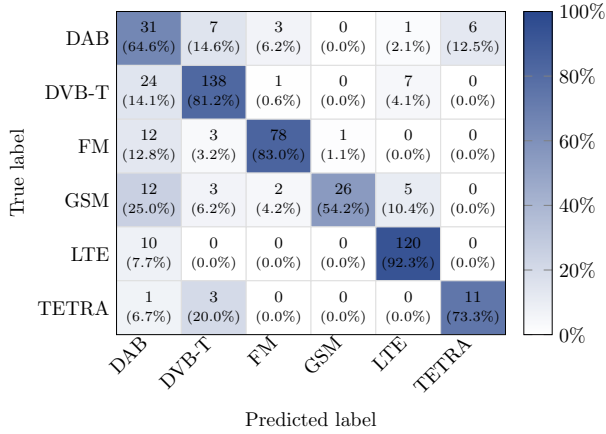
segments derived from the same spectrogram remain within the same subset, reducing the risk of data leakage.

The validation subset was used to monitor generalization during training, and the test subset was reserved exclusively for the final evaluation. The analysis compares two training configurations: a standard training procedure and a variant incorporating sample weighting to mitigate class imbalance. The results reported for the test subset are assessed using per-class accuracy, obtained from the normalized confusion matrices, as well as the mean per-class accuracy across the considered radio technologies.

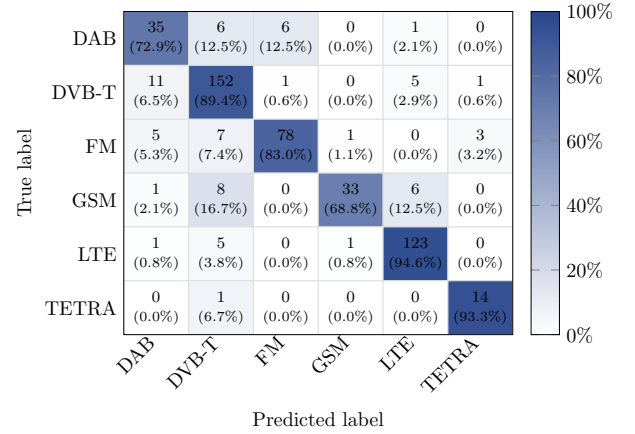
Figure 3a and Figure 3b present the class-normalized confusion matrices obtained without and with sample weighting, respectively. These matrices provide a detailed view of the classification behavior by showing the distribution of predictions within each true class, thereby highlighting how errors are distributed across the considered radio technologies.

A quantitative comparison between the two configurations is summarized in Table II. Except for FM, whose performance remained the same, the weighted strategy improves the results for all the other technologies, most notably TETRA and GSM, the two classes with the fewest training segments. As a result, the mean per-class accuracy increases from 74.8% to 83.7%. This indicates that class weighting promotes more balanced behavior across radio technologies, benefiting underrepresented classes, which is desirable in spectrum-monitoring applications, where reliable recognition should not be limited to the most frequent or dominant technologies.

From a qualitative perspective, the model performs well when spectrogram patterns are well defined and exhibit high signal-to-noise ratios. In such cases, characteristic structures



(a) Without sample weighting



(b) With sample weighting

Fig. 3: Confusion matrices obtained without and with sample weighting.

TABLE II: Per-class accuracy (%) comparison between standard and weighted training strategies.

Class	Without Weighting	With Weighting
DAB	64.6	72.9
DVB-T	81.2	89.4
FM	83.0	83.0
GSM	54.2	68.8
LTE	92.3	94.6
TETRA	73.3	93.3
Mean	74.8	83.7

of each radio technology are captured, enabling reliable classification. More challenging scenarios arise when spectrograms exhibit ambiguous patterns, overlapping spectral components, or low signal-to-noise ratios. These conditions increase confusion between classes with similar time-frequency signatures.

V. CONCLUSION

This paper presented a promising deep learning-based approach for classifying radio access technologies using spectrogram. By first applying an appropriate preprocessing strategy for handling variable-sized spectrograms and then leveraging the lightweight MobileNetV3 architecture as a feature extractor, the proposed method achieves effective classification performance.

The experimental results demonstrate that the model can accurately identify distinct radio technologies, particularly in scenarios with well-defined time-frequency patterns. Furthermore, the analysis highlights the importance of addressing class imbalance: using sample weighting improves macro-averaged performance and promotes a more balanced classification across classes.

While standard training tends to favor dominant classes, the weighted approach improves recognition of underrepresented classes without penalizing the dominant ones, which is particularly relevant in practical spectrum-monitoring applications. In deployment, a full spectrogram can be labeled by majority voting over its segments.

Future work may explore more advanced data augmentation strategies, domain adaptation techniques, and the integration of

temporal context to further improve classification performance in challenging scenarios.

REFERENCES

- [1] T. J. O'Shea and J. Hoydis, "An introduction to deep learning for the physical layer," *IEEE Transactions on Cognitive Communications and Networking*, vol. 3, no. 4, pp. 563–575, 2017.
- [2] E. Fonseca, J. F. Santos, F. Paisana, and L. A. DaSilva, "Radio access technology characterisation through object detection," *Computer Communications*, vol. 168, pp. 12–19, 2021.
- [3] O. A. Dobre, A. Abdi, Y. Bar-Ness, and W. Su, "Survey of automatic modulation classification techniques: Classical approaches and new trends," *IET Communications*, vol. 1, no. 2, pp. 137–156, 2007.
- [4] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [5] T. J. O'Shea, J. Corgan, and T. C. Clancy, "Convolutional radio modulation recognition networks," in *Proceedings of the International Conference on Engineering Applications of Neural Networks (EANN)*, 2016, pp. 213–226.
- [6] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.
- [7] A. Howard, M. Sandler, G. Chu, L.-C. Chen, B. Chen, M. Tan, W. Wang, Y. Zhu, R. Pang, V. Vasudevan, Q. V. Le, and H. Adam, "Searching for MobileNetV3," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 1314–1324.
- [8] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4510–4520.
- [9] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7132–7141.
- [10] A. Scalingi, D. Giustiniano, R. Calvo-Palomino, N. Apostolakis, and G. Bovet, "A framework for wireless technology classification using crowdsensing platforms," in *Proceedings of the IEEE Conference on Computer Communications (INFOCOM)*, 2023, pp. 1–10.
- [11] S. Rajendran, B. Van den Bergh, R. Calvo-Palomino, H. Cordobes, D. Giustiniano, M. Fuchs, S. Pollin, and V. Lenders, "Electrosense: Open and big spectrum data," *IEEE Communications Magazine*, vol. 56, no. 1, pp. 210–217, 2018.
- [12] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 248–255.
- [13] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" in *Advances in Neural Information Processing Systems*, vol. 27, 2014, pp. 3320–3328.