Efficient Deep Learning Models for Real-time Waste Material Recognition

Alberto Cavallo
Polytechnic of Turin
Turin, Italy
s312743@studenti.polito.it

Giuseppe Cavaleri
Polytechnic of Turin
Turin, Italy
s314734@studenti.polito.it

Stefano Chiartano
Polytechnic of Turin
Turin, Italy
s313651@studenti.polito.it

Abstract

This paper investigates the efficacy of deep learning-based computer vision modules for material detection and categorization in waste management. Three state-of-the-art models, ENet, BiSeNet, and ICNet, are compared based on mean Intersection-Over-Union (mIoU), with a focus on the image semantic segmentation task. Two distinct modalities, namely binary and instance segmentation, are explored to enable image classification using two different approaches. The experiments utilize the ReSort-IT dataset, enabling accurate classification of objects into Paper, Aluminium, Bottle, and Nylon classes. The study includes data augmentation techniques to enhance model training and implements MobileNet to reduce the size of ICNet while maintaining high performance. Indeed, the research aims to optimize these algorithms for resource-constrained devices, thereby reducing the overall cost associated with model size and the number of parameters. Methods pertaining to pruning are also examined to obtain faster processing. The results demonstrate the potential of these models in achieving real-time and accurate waste material recognition.
(Project git repository)

1. Introduction

The current waste crisis and the urgent need to adopt sustainable measures to address it require the implementation of innovative solutions in waste management. In recent times, automated systems have become increasingly prevalent in all disposal centers, contributing to improved efficiency and effectiveness.

In order to address this global challenge, scientific research is increasingly focusing on the application of advanced technologies, such as the use of neural networks, to enhance waste management [4].

These technologies offer promising solutions for optimizing waste collection, sorting, and recycling processes, leading to increased efficiency, reduced environmental impact, and improved resource recovery.

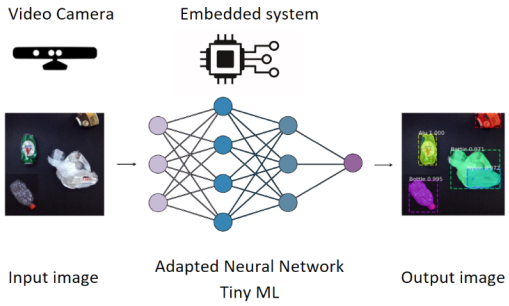

Figure 1. Waste sorting system pipeline

By leveraging innovative approaches, waste management systems can achieve intelligent decision-making capabilities, efficient resource allocation, and predictive modeling to effectively address the complexity of waste management in an increasingly dynamic environment.

Moreover, the application of artificial intelligence enables the identification and classification of various waste materials, leading to improved sorting efficiency. This facilitates effective recycling and proper disposal, ensuring optimal utilization of resources while mitigating adverse impacts associated with waste mismanagement.

This research paper presents a comprehensive analysis of computer vision modules based on deep learning technologies that are used in the semantic segmentation problem. The primary objective of this study is to compare and evaluate these modules in the context of vision-based material detection and categorization for the purpose of assigning specific classes to the objects. One of the major challenges addressed in this work is the issue of Tiny ML, where algorithms are tailored to operate effectively on devices with limited resources. In order to overcome this challenge, various techniques are explored to reduce the number of parameters and the overall model size, while maintaining optimal performance. Such algorithms (Figure 1) typically employ video cameras in embedded systems, enabling real-

time capture and analysis of visual data, that allows continuous monitoring and non-intrusive data acquisition.

The image segmentation problem involves dividing an image into multiple regions or segments based on their visual properties. It can be done in two different ways: binary and instance segmentation. Binary segmentation, as shown in Figure 2 is used to detect the presence of an object in the image, in fact, this process classifies each pixel into one of two possible categories: foreground or background. Instance segmentation (Figure 3), takes the task further by identifying and differentiating individual objects in order to assign a specific class. The goal, in both cases, is to create a pixel-wise mask to compare with the ground truth to evaluate the performance of the model.

The dataset utilized in the conducted experiments is referred to as ReSort-IT. It comprises a collection of labeled recyclable images. The system employed demonstrates the capability to assign one of four classes to an object, namely Paper, Aluminium, Bottle, and Nylon. The training set is composed of 5,500 images of different garbage types, while the validation set comprises 1,460 images. Each pixel within an image can be assigned a numerical label ranging from 0 to 4 (background, paper, bottle, aluminum, nylon), where each number corresponds to a specific class.

Firstly, we evaluated the three models using the mean Intersection-Over-Union (mIoU) metric, then, we conducted supplementary analyses and tests to evaluate the influence of data augmentation on these metrics. Moreover, we made efforts to modify the model by utilizing MobileNet as a lighter backbone, aiming to decrease its weight while preserving high performance. The implementation of this approach resulted in reductions of 79% and 85% in weight for the BiSeNet and ICNet models, respectively. Besides, we explored pruning methods to decrease the latency of the model.

Figure 2. Binary segmentation

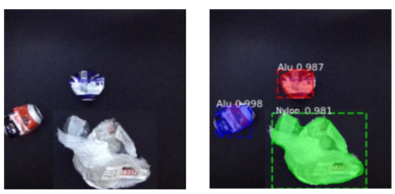

Figure 3. Instance segmentation

2. Related Works

In the domain of waste sorting, garbage collectors play a fundamental role in efficiently collecting and disposing of waste materials. Various studies have explored the optimization of garbage collection routes, resource allocation, and scheduling techniques to improve overall efficiency. On the other hand, real-time semantic segmentation networks have gained significant attention for their ability to accurately analyze and classify waste materials in real-time. This section provides an overview of existing research efforts, highlighting advancements, challenges, and potential synergies between garbage collectors and real-time semantic segmentation networks in enhancing waste management systems.

2.1. Garbage Collectors

The majority of waste sorting and garbage collection systems comprise a visual detection module along with a robotic component responsible for collecting and organizing the waste.

For instance, a garbage recycling robot [3] implements this setup by utilizing the GarbageNet framework in its initial module to determine the bounding box and object type using a pixel-level mask. The second module pertains to the robotic system, which enables to grasp the objects, calculate the trajectory, and deposit them in the appropriate bin.

Another application is derived from the Zen Robotics Recycler [6], which leverages machine learning techniques to facilitate object and material recognition, as well as object manipulation. This technology exhibits the potential to substitute manual sorters during the concluding phases of waste sorting.

2.2. Real-time Semantic Segmentation Networks

Real-time semantic segmentation networks refer to a class of deep learning models designed to perform pixel-level semantic segmentation in real-time. These networks utilize convolutional neural network architectures and advanced algorithms to analyze and classify each pixel of an input image into predefined semantic categories, such as object classes or regions of interest. The key advantage of real-time semantic segmentation networks is their ability to provide accurate and detailed segmentation results with minimal time delay, enabling applications that require immediate and responsive processing.

Principally, three main architectures are analyzed: ENet [7], BiSeNet [11], ICNet [12]. All of these structures are state-of-the-art approaches in semantic segmentation that address the challenges of accuracy, efficiency, and real-time processing. Each model offers unique architectural designs and techniques to achieve the desired segmentation performance while respecting specific requirements in terms of computational resources and application scenarios.

- **ENet**, short for "Efficient Neural Network", was created specifically for tasks requiring low latency operation. The architecture is based on an encoder-decoder structure that respectively cover the downsampling and the upsampling operation. Overall, ENet offers an efficient and lightweight solution for real-time semantic segmentation tasks by striking a balance between computational complexity and accuracy.

- **BiSeNet** incorporates the use of two paths to elaborate the input, the Spatial path and the Context path, these two branches are fused together using a bilateral fusion module to produce the final segmentation result. The high-resolution branch preserves fine-grained details by employing a U-shaped architecture, while the low-resolution branch utilizes a lightweight spatial path to capture global context information.

- **ICNet** adopts a cascaded structure that processes images at multiple resolutions in parallel, allowing for efficient and fast inference. It consists of three interconnected branches having different scales, that after the processing are fused to produce the output. This approach allows to capture information at different levels.

3. Methodology

In this section, we present the methodology employed in our study to evaluate the effectiveness of deep learning-based computer vision modules for material detection and categorization in waste management.

3.1. Model Selection

To begin, we selected three state-of-the-art computer vision models, namely ENet, BiSeNet, and ICNet, for our investigation. These models were chosen based on their established performance in semantic segmentation tasks and their suitability for real-time and resource-constrained applications.

To enable the validation for instance segmentation with the cited models, we computed the mean IoU (mIoU) metric for each category of litter: the IoU was computed by measuring the overlap between the predicted and ground truth masks, which involved determining the number of overlapping pixels and dividing it by the total number of pixels in the union of the two masks. This resulted in a value between 0 and 1, where higher values indicated better segmentation accuracy for the respective class.

To obtain the mean IoU (mIoU), we accumulated the computed IoU values for each class and divided the sum by the total number of validation samples. This step allowed us to assess the overall segmentation performance across all classes. The resulting values represented the mIoU for each class, indicating the average overlap between the predicted and ground truth masks.

3.2. Loss Functions

In the context of semantic segmentation, different loss functions are employed to measure the discrepancy between the predicted segmentation masks and the ground truth labels.

For binary segmentation, we utilized the Binary Cross Entropy with Logits (BCEWithLogitsLoss) function. This loss function combines a sigmoid activation function and binary cross-entropy loss, enabling efficient training and handling of class imbalance. The BCEWithLogitsLoss encourages the model to produce high probabilities for pixels belonging to the foreground class and low probabilities for the background class.

On the other hand, for instance segmentation, where the objective is to segment individual instances of the 5 different classes, we employed the Cross-Entropy Loss. This loss function is commonly used in multi-class segmentation tasks and measures the dissimilarity between the predicted class probabilities and the ground truth labels. The Cross-Entropy Loss encourages the model to assign high probabilities to the correct class and lower probabilities to other classes.

The effectiveness of the chosen loss functions in training the models was assessed through monitoring the training loss over epochs. As shown in Figure 4, the decreasing trend in the training loss demonstrates that the models are learning and adapting their parameters to minimize the discrepancy between their predictions and the ground truth labels. This indicates that the selected loss functions effectively guide the training process and contribute to the models' ability to accurately segment waste materials.

3.3. Model Optimization

To address the constraints posed by resource-constrained devices, we focused on optimizing the models, BiSeNet and ICNet, which were initially selected for their effectiveness in semantic segmentation tasks. However, we found that their large sizes made them unsuitable for our TinyML experiment. Thus, our objective was to reduce the model size while preserving satisfactory performance.

To achieve model optimization, we replaced the original backbone, based on ResNet [1], with a lighter alternative: MobileNetV2 [10]. MobileNetV2 is a well-established lightweight convolutional neural network architecture renowned for its efficiency and applicability to mobile and embedded devices. It achieves this through techniques such as depthwise separable convolutions, which reduce the number of parameters and computations while maintaining performance.

Our choice of MobileNetV2 was motivated by several

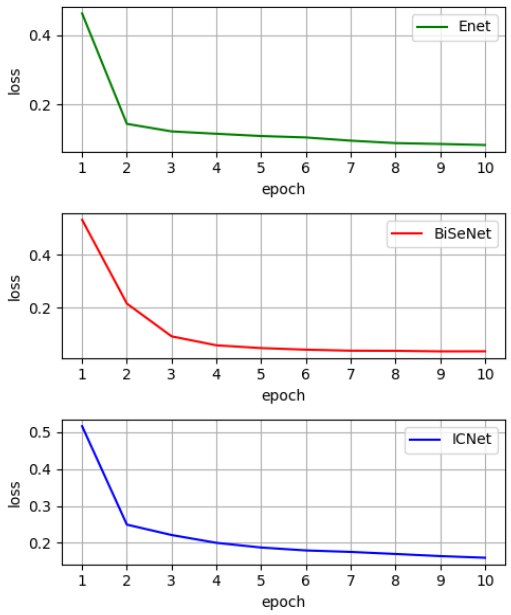

Figure 4. Average loss of the three models comparison

factors. Firstly, it strikes a balance between model size and accuracy, making it suitable for our resource-constrained scenario. Secondly, it has been extensively evaluated and utilized in various computer vision tasks, including semantic segmentation, demonstrating its effectiveness and reliability [2].

ResNet employs residual connections to address the vanishing gradient problem, enabling training of deep networks and facilitating accurate feature representation. In contrast, MobileNetV2 introduces Inverted Residual Blocks, which prioritize computational efficiency. These blocks leverage depth-wise separable convolutions to decrease the number of parameters and computational complexity, resulting in a lightweight model ideal for resource-constrained applications, the difference between the two architectures can be visualized in Figure 5.

(a) Residual block (b) Inverted residual block

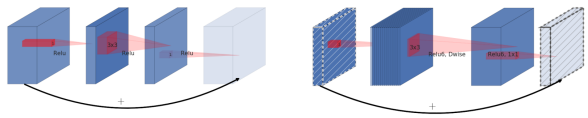

Figure 5. Comparison of Residual Block (a) and Inverted Residual Block (b) architectures. The Residual Block (a) utilizes residual connections for deep network training, while the Inverted Residual Block (b) prioritizes computational efficiency with depth-wise separable convolutions.

To assess the impact of replacing ResNet with MobileNetV2, we conducted a comparative analysis focusing on model size and segmentation accuracy. The integration of MobileNetV2 as the backbone led to a significant reduction in model size without compromising overall performance extensively. Although a slight decrease in accuracy was observed compared to the original ResNet-based models, this trade-off was considered acceptable given the substantial reduction in model complexity.

As part of our model optimization process, we incorporated a pruning technique to enhance the efficiency of our deep learning models. Pruning is a method used to reduce the complexity and computational requirements of neural networks by selectively removing less important connections or weights [5].

To implement pruning, we utilized L1 unstructured pruning on both the convolutional and linear layers of the network. By applying a pruning amount of 0.4, we removed 40% of the weights in these layers, effectively eliminating redundant or less significant parameters. Figure 6 showcases the effect of weight pruning on a neural network.

The incorporation of pruning serves the purpose of reducing the model size and computational demands, making it more suitable for deployment on resource-constrained devices. By selectively removing unimportant parameters, we achieved a more compact and efficient model architecture.

Through the combined approach of utilizing the MobileNetV2 backbone and integrating pruning, we successfully achieved model optimization without significant compromise in overall performance.

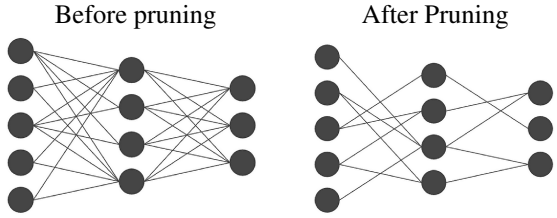

Figure 6. Illustration of the pruning technique applied to a neural network

3.4. Data Augmentation

To enhance the generalization capabilities of our models and mitigate overfitting, we employed various data augmentation techniques during the training phase.

Specifically, we applied flipping, rotation, scaling, and translation randomly over both the x and y axes to augment our training dataset. These techniques help to increase the diversity and variability of the training samples, enabling the models to learn more robust and invariant features.

We conducted experiments to assess the impact of data augmentation on the performance of our models. The re-

sults demonstrated that data augmentation improved models' performance on the ReSortIT dataset. The augmented training data provided a richer training set, allowing the models to learn more effectively and achieve higher accuracy in waste material detection and categorization.

Figure 7 illustrates the application of the cited data augmentation techniques.

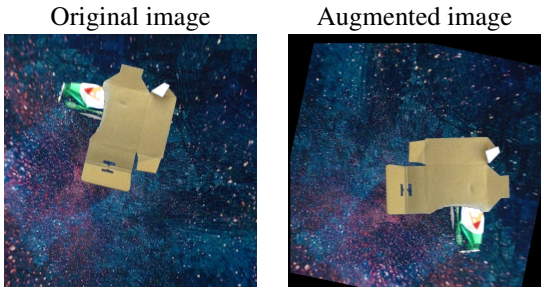

Figure 7. Image Augmentation. Original images (left) and augmented versions (right) showcase variations in scale, rotation, and translation

3.5. Hyperparameter tuning

Hyperparameter tuning was a crucial aspect for our model development, aiming to find the optimal values for the learning rate and the batch size that maximize model performance (values explored are shown in Table 1)

Learning Rate	Batch Size
5×10^{-2}	4
5×10^{-3}	**8**
$\mathbf{5 \times 10^{-4}}$	16
5×10^{-5}	32
5×10^{-6}	64

Table 1. Hyperparameter Tuning

The learning rate determines the step size at which the model's weights are adjusted during training. A learning rate that is too high can result in overshooting the optimal solution or even divergence, while a learning rate that is too low can lead to slow convergence or getting trapped in suboptimal solutions. In this case, a learning rate of 5×10^{-4} might have struck a balance between convergence speed and stability, allowing the model to efficiently reach an optimal solution without overshooting or getting stuck.

Concerning the batch size, it refers to the number of training examples processed in each iteration of the model's training. Larger batch sizes provide more stable gradient estimates due to the increased number of images used, but they also demand more memory and computational resources. Smaller batch sizes introduce more noise into the gradient estimates but may help avoid getting stuck in sharp minima. The choice of a batch size of 8 indicates a sweet spot in this scenario, balancing stability and computational efficiency. Figure 8 shows the trend of the mIou for increasing values of learning rate and batch size.

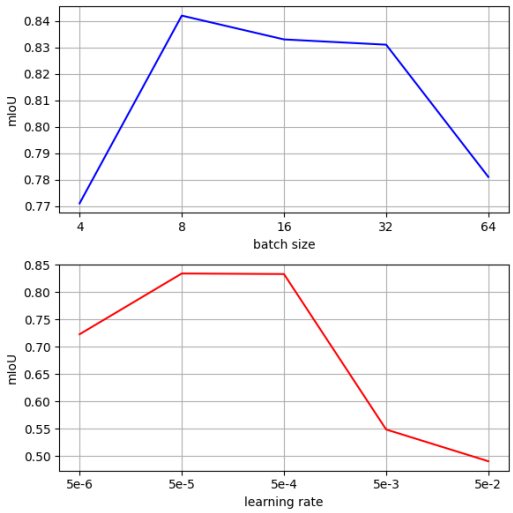

Figure 8. Hyperparameter tuning results for ENet Binary Segmentation

4. Experimental Results

In this paper, we conducted experiments to evaluate the performance of the three network architectures, in the tasks of binary and instance segmentation. The evaluation metric used to measure the performance was the mean Intersection over Union (mIoU). The results of our experiments are summarized in Table 2, which shows that ENet outperformed the other two architectures in terms of mIoU.

To further improve the accuracy of the segmentation results, we introduced a data augmentation technique, which is described in detail in the methodology paragraph. The goal of this technique was to enhance the quality and relevance of the training data by selectively augmenting the input images. The results of incorporating data augmentation into our experiments are presented in the Table 3, which provides the updated mIoU scores.

Additionally, we explored the integration of MobileNet as the backbone architecture for our segmentation networks. The use of MobileNet as a backbone offers the advantage of reducing the overall network size while still maintaining reasonable accuracy for ICNet where the mIoU is around 0.8, on the other hand the score of BiSeNet decrease around 0.72. The impact of this integration on the network size reduction are documented in Table 4.

The effect of pruning on the latency of our segmentation networks is presented in Table 5. We observed that the la-

	Binary segmentation				Instance segmentation				
Network	**mIoU**	**Params (M)**	**Size (MB)**	**GFLOPs**	**Aluminium**	**Paper**	**Bottle**	**Nylon**	**Avg**
ENet	**0.834**	0,36	1,39	8,30	0.799	0.831	0.915	0.787	**0.833**
BiSeNet	0.820	12.4	47.30	105,58	0.792	0.776	0.828	0.815	0.802
ICNet	0.804	22,2	84.72	19.78	0.749	0.737	0.822	0.717	0.756

Table 2. Models performance comparison

Segmentation	**Models**	**mIoU**
Binary	Enet	0.861
	BiSeNet	0.856
	ICNet	0.859
Instance	Enet	0.854
	BiSeNet	0.831
	ICNet	0.804

Table 3. Final results with data augmentation

Backbone	**Dimensions**	**BiSeNet**	**ICNet**
ResNet	Params (M)	12.4	22.2
	Size (MB)	47.30	84.72
MobileNet	Params (M)	2.6	3.3
	Size (MB)	9.91	12.7
Reduction		**79%**	**85%**

Table 4. Model size reduction with MobileNet implementation

Model	**Step**	**ENet**	**BiSeNet**	**ICNet**
Not Pruned	training	164 s	162 s	183 s
	validation	39 s	49 s	53 s
Pruned	training	112 s	110 s	112 s
	validation	28 s	24 s	25 s
Reduction		**31%**	**32%**	**39%**

Table 5. Latency for the different pruned models in binary segmentation.

tency was reduced on average by 34%, preserving the accuracy of the models. By selectively removing non-essential parameters, the pruned architecture achieved enhanced efficiency. This highlights the effectiveness of pruning in optimizing and deploying resource-friendly models.

Overall, our study presents a comprehensive evaluation of ENet, BiSeNet, and ICNet in binary and instance segmentation tasks. By incorporating data segmentation techniques and exploring the integration of MobileNet, we strive to improve the efficiency of these network architectures. The obtained results demonstrate the potential of these advancements in achieving more precise and compact segmentation models for various computer vision applications.

5. Conclusions

The overarching objective of this research was to optimize the aforementioned algorithms for resource-constrained devices, effectively minimizing the associated costs related to model size and the number of parameters. The results showcased the potential of these models in achieving real-time and accurate waste material recognition.

A line of work for future research could be the exploration of Transfer learning [9, 14] which offer promising avenues for improving the models performance when labeled data is limited or new waste material classes need to be added. By leveraging pre-trained models and adapting them to the waste management domain, the models can quickly adapt to new scenarios and exhibit improved generalization capabilities.

Another idea could be to explore alternative loss functions to address the class imbalance inherent in waste material detection and categorization tasks. The imbalanced distribution of waste materials in real-world scenarios can lead to biased model predictions, with certain classes being underrepresented and consequently poorly detected. By adopting specialized loss functions such as focal loss [8] or weighted cross-entropy [13], which assign higher weights to minority classes, the model can learn to effectively capture and classify rare waste material classes. This would further enhance the accuracy and robustness of the deep learning models in waste management applications.

Furthermore, future research directions should focus on the integration of the developed deep learning models with robotic systems in waste management. The deployment of autonomous robots capable of identifying and sorting different types of waste materials can revolutionize waste management processes, making them more efficient and automated.

In conclusion, this study underscores the effectiveness of deep learning-based computer vision modules for material detection and categorization in waste management. The comparative analysis of ENet, BiSeNet, and ICNet, along with the exploration of different segmentation modalities, provides valuable insights into their capabilities. The experiments conducted on the ReSort-IT dataset demonstrate accurate classification of waste materials, while the reduction of the net size, specifically the implementation of techniques like the implementation of MobileNet to reduce the size of the ICNet and BiSeNet, holds significant advantages for small robots. The future integration of these models with robotic systems and the exploration of transfer learning techniques hold promising potential for advancing waste management practices.

References

[1] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015. 3

[2] Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications, 2017. 4

[3] Kirsty Ellis Denis Hadjivelichkov Danail Stoyanov Arash Ajoudani Jingyi Liu, Pietro Balatti and Dimitrios Kanoulas. Garbage collection and sorting with a mobile manipulator using deep learning and whole-body. Tech. Rep., 2021. 2

[4] Maria Koskinopoulou, Fredy Raptopoulos, George Papadopoulos, Nikitas Mavrakis, and Michail Maniadakis. Robotic waste sorting technology: Toward a vision-based categorization system for the industrial robotic separation of recyclable waste, 2021. 1

[5] Lucas Liebenwein, Cenk Baykal, Brandon Carter, David Gifford, and Daniela Rus. Lost in pruning: The effects of pruning neural networks beyond test accuracy, 2021. 4

[6] D. T. J. Lukka, D. T. Tossavainen, D. J. V. Kujala, and D. T. Raiko. Zenrobotics recycler – robotic sorting using machine learning. Tech. Rep., 2014. 2

[7] Adam Paszke, Abhishek Chaurasia, Sangpil Kim, and Eugenio Culurciello. Enet: A deep neural network architecture for real-time semantic segmentation, 2016. 2

[8] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A. Efros. Context encoders: Feature learning by inpainting, 2016. 6

[9] Ricardo Ribani and Mauricio Marengoni. A survey of transfer learning for convolutional neural networks, 2019. 6

[10] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks, 2019. 3

[11] Changqian Yu, Jingbo Wang, Chao Peng, Changxin Gao, Gang Yu, and Nong Sang. Bisenet: Bilateral segmentation network for real-time semantic segmentation, 2018. 2

[12] Hengshuang Zhao, Xiaojuan Qi, Xiaoyong Shen, Jianping Shi, and Jiaya Jia. Icnet for real-time semantic segmentation on high-resolution images, 2018. 2

[13] Z. Zhou, H. Huang, and B. Fang. Application of weighted cross-entropy loss function in intrusion detection, 2021. 6

[14] Fuzhen Zhuang, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, and Qing He. A comprehensive survey on transfer learning, 2020. 6