REPUBLIC OF TURKEY
ERCIYES UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
DEPARTMENT OF COMPUTER ENGINEERING

TRAFFIC SIGNS CLASSIFICATION WITH TRANSFER LEARNING OF CNN BASED MODELS AND PERFORMANCE COMPRASION

Prepared by
Mohammed Ghazi Khassaf ALSHAMI

Supervised by
Asst. Prof. Fehim KÖYLÜ

M.Sc. Thesis

April 2023
KAYSERI

REPUBLIC OF TURKEY
ERCIYES UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
DEPARTMENT OF COMPUTER ENGINEERING

TRAFFIC SIGNS CLASSIFICATION WITH TRANSFER LEARNING OF CNN BASED MODELS AND PERFORMANCE COMPRASION

(M.Sc. Thesis)

Prepared by
Mohammed Ghazi Khassaf ALSHAMI

Supervised by
Asst. Prof. Fehim KÖYLÜ

April 2023
KAYSERI

COMPLIANCE WITH SCIENTIFIC ETHICS

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Owner of Thesis

Mohammed Ghazi Khassaf ALSHAMI

COMPLIANCE WITH GUIDELINES

This master's thesis writes up on the topic **“Traffic Signs Classification With Transfer Learning of CNN Based Models And Performance Comprasion”** has been prepared following Erciyes University Graduate Education and Teaching Institute Thesis Preparation and Writing Guide.

Owner of Thesis	Supervisor
Mohammed Ghazi Khassaf ALSHAMI	Asst. Prof. Fehim KÖYLÜ

Head of Department

Prof. Dr. Prof. Dr. Veysel ASLANTAŞ

ACKNOWLEDGMENTS

During my Master journey, my valued professors, respected university staff, and dear colleagues have helped me to achieve my goals.

First, I would like to warmly thank my supervisor **Asst. Prof. Fehim KÖYLÜ**, for his valued guidance and his extremely useful ideas and endless professional support through this project.

Thank you very much for your valuable time and patience.

I also send my warm thanks and gratitude towards my colleagues and friends, **Mr. Anmaar Hamed, Mr. Sabah Faisal, MSc. Ahmed Noori ALGBURI, MSc. Ibrahem Alabdullah, Dr. Ibrahem HADI, Tarek Miqdam, Mr. Ali Sattar and MSc. Almustafa Mahdi Saleh ALFAWADI** for supporting me during my stay period in Kayseri and being more than a brother to me, also my **friends in IRAQ** for continuous supporting me.

Finally, most importantly, I would also express my sincere gratitude to my beloved **Parents**, my **brothers,** my **Sisters,** and **my Cousins** for continuous moral support and understanding during the last three years and a half in which most of my time and energy was devoted to the completion of this project.

Mohammed Ghazi Khassaf ALSHAMI

Kayseri, APRIL 2023

TRAFFIC SIGNS CLASSIFICATION WITH TRANSFER LEARNING OF CNN BASED MODELS AND PERFORMANCE COMPRASION

Mohammed Ghazi Khassaf ALSHAMI

Erciyes University, Graduate School of Natural and Applied Sciences M.Sc. Thesis, April 2023
Supervisor: Asst. Prof. Fehim KÖYLÜ

ABSTRACT

Traffic sign recognition systems are essential to popular real-world applications such as autonomous driving and driver assistance. Several learning systems based on ResNet, VGG, and DenseNet classification models, have been published recently to improve traffic sign recognition accuracy. However, it is debatable as to which of these systems is better than the others. It must be thoroughly evaluated with the same dataset. First, they must have similar hyperparameter values and evaluation mechanisms to test the system under similar conditions. Also, training should be common under the same number of intervals. The language used for coding and the technology used to call the training system must also be identical. Under these conditions, different classification models were compared under similar conditions using the GTSRB dataset and the Turkish traffic sign dataset. Thus the inference of this analogy would be valid. This study uses both datasets for traffic sign recognition by transfer learning for ResNet50, DenseNet121, VGG16, VGG19, MobileNet, Xception, and EfficientNetB0. Training loss, validation loss, training accuracy, validation accuracy, test accuracy, test recall, F1 test measurement, and test accuracy are used to compare models. The results are compared and presented for use in traffic sign recognition.

Keywords: Traffic sign classification; deep learning; convolutional neural network; transfer learning

CNN AĞLARI KULLANILARAK TRAFIK İŞARETLERININ TESPITI VE PERFORMANS KARŞILAŞTIRMASI

Mohammed Ghazi Khassaf ALSHAMI

Erciyes Üniversitesi, Fen Bilimleri Enstitüsü
Yüksek Lisans Tezi, Nisan 2023
Danışman: Dr. Öğr. Üyesi Fehim KÖYLÜ

ÖZET

Trafik işaretlerini tespit etmeye yönelik sistemler, otonom sürüş ve sürücü yardımı gibi yaygın gerçek dünya uygulamalarının önemli bir parçasıdır. Diğerlerinin yanı sıra ResNet, VGG ve DenseNet dahil olmak üzere birçok öğrenme sistemi, trafik işareti algılamanın doğruluğunu artırmak için son yıllarda kullandırılmıştır. Ancak bunlardan hangisinin diğerlerinden daha iyi performans gösterebileceği net değildir. Modellerin benzer koşullarda ve hiperparametre değerleri altında araştırılmaları gerekir. Sistemin aynı veri kümesi için verdiği performans değerleri karşılaştırılabilir. Ek olarak, aynı iterasyon sayısı ile eğitilmesi gibi standartlar belirlenmelidir. Her ikisi de aynı olması gereken kodlama için kullanılan dil ve eğitim sistemini çağırmak için kullanılan teknik, ayrıca dikkate alınması gereken hususlardır. Sonuç olarak, bu koşullar altında, farklı sınıflandırma modellerinin karşılaştırmasının altında yapıldığı söylenebilir. Benzer durumlarda, bu analojinin sonucu doğru olacaktır. Bu çalışmada, ResNet50, DenseNet'in transfer öğrenmesi ile trafik işaretlerinin tespiti için Türk trafik işareti ve Alman trafik işareti veriseti (GTSRB) kullanılmıştır. DenseNet121, VG16, VGG19, MobileNet, Xception, EfficientNet-B0 derin öğrenme sınıflandırma modelleri transfer öğrenme yöntemiyle veri kümesi üzerinde aynı koşullarda uygulanmıştır. Elde edilen sonuçlar karşılaştırılıp trafik işareti tespitinde kullanımı değerlendirilmiştir. Modeller için eğitim kaybı, doğrulama kaybı, eğitim doğruluğu, validasyon doğruluğu, test kesinliği, test hatırlama, test F1 ölçümü ve test doğruluğu metrikleri verilmiştir.

Anahtar Kelimeler: Trafik işareti sınıflandırma; derin öğrenme; evrişimli sinir ağı; transfer öğrenme

CONTENTS

TRAFFIC SIGNS CLASSIFICATION WITH TRANSFER LEARNING OF CNN BASED MODELS AND PERFORMANCE COMPRASION

CONTENT OF FIGURE

CONTENT OF TABLE

INTRODUCTION

Companies such as Google, Volkswagen, Toyota, and Recent advances in the automotive sector demonstrate the growing importance of computer technology across the board, with a focus on improved functionality, speed, and safety. Global automobile and IT General Motors have all made major investments in the development and deployment of autonomous cars.

Safety is more important than quality, minimal delay, and comfort - characteristics people want to see in self-driving cars. In this direction, the issue of active safety comes into play, complementing the passive safety of autonomous vehicles. Passive safety focuses primarily on the overall configuration of the vehicles. On the other hand, active safety is the safest approach to emergencies that may occur to prevent accidents while the vehicle is in motion. The Society of Automotive Engineers has divided safe autonomous driving into categories by considering numerous vehicle signs. The levels of autonomous driving are rated between 0 and 5. Vehicles that require 100% manual operation are classified as level 0, while vehicles that require no human attention and drive fully autonomously are classified as level 5 vehicles.

Driverless vehicles are expected to provide many benefits to people in the future. Reducing traffic congestion, fast and safe driving, eliminating accidents caused by human error, public transportation, commercial and logistical purposes, and economic benefits are exemplary expectations. Infrastructure projects should be implemented to enable autonomous driving unhindered and increase safety, speed, and comfort, and more investments should be made in this direction.

Stop, road, and traffic signs are the latest traffic control devices at intersections. The main goal in developing these control systems was to control traffic and improve safety at intersections. These control systems' efficiency and safety at junctions have recently been investigated. We need solutions to the problems we see, especially because we know that

the number of cars worldwide will at least double by 2050. One of the options that promises progress in this regard is the use of autonomous cars [1].

One device supporting autonomous vehicle driving is Cooperative Adaptive Cruise Control (CACC). CACC is an improvement of adaptive cruise control, which uses long-range sensors to measure the distance and approach speed of vehicles to the oncoming vehicle. AHS needs complex processing methods to filter the noise and extra waves from the sensor input data. The signal filtering's delays make it harder for AHS to follow other cars up front precisely. CACC gets around this restriction by accepting extra data delivered through a mobile data connection. CACC may learn about vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) links to make smarter and quicker judgments. It may also closely follow the car in second place simultaneously. After analyzing the information, CACC can choose whether to quicken, slow down, or maintain its speed. The fact that the current generation of CACC is portable is not their fault [2]. Autonomous systems are used in areas such as traffic monitoring, disaster management, agriculture, mineral exploration, navigation target tracking, photogrammetric acquisition, and surveying [3]. In particular, the importance of autonomous systems in applications such as video-based target tracking and location determination has contributed to the advancement of autonomous vehicle navigation systems in geospatial applications. Autonomous vehicles are expected to provide safe driving away from human-caused accidents because they drive free from human error. The systems that ensure the vehicle's movement are designed accordingly, minimizing fuel consumption due to driver differences, and preventing pollution.

The demand for self-driving vehicles is increasing daily, in parallel with technology, due to the requirements and needs of the battlefield and pre-battle superiority, border security, security of facilities and buildings, space exploration, scientific applications, and various other requirements and needs. Investments for this purpose increase exponentially yearly, and the successes achieved bring important benefits and advantages.

Most traffic accidents are caused by human errors [4], and by using computers, sensors, and other systems, an autonomous vehicle can analyze traffic faster and provide a safer ride. Autonomous vehicles adhere more strictly to the rules of the road, do not exceed the speed limit, and do not get distracted, thus ensuring safer driving. In the future,

autonomous vehicles can converse with one another, request directions, and notify each other when they change lanes. This way, people can work while driving, and traffic chaos will end [5].

CHAPTER 1

GENERAL INFORMATION

1.1. Self-Driving Car

Today, solving numerous problems, especially various traffic jams, accidents caused by human error, driver comfort, and environmental pollution, presents scientists with the task of developing and applying modern technologies and traditional approaches. In this regard, the development of self-determining devices occupies a very important place. The development of automated driving promises tremendous results, especially in solving or reducing human-based traffic problems. Moreover, it is predicted to have a beneficial future in terms of social and economic aspects such as reducing pollution, saving natural resources, and low-cost transportation, with autonomous vehicle manufacturers focusing on the ability to operate on electricity.

1.2. Features of self-driving vehicles

Autonomous driving is meant that assistance systems accompany the driver while driving. On the other hand, the driver does not intervene with 100% autonomous in any way. For the features that enable automatic driving, we can write two related subsets: The first is the ability to perceive the environment or events, and the other is to react to perceived situations.

Many important automotive industry innovations have been introduced to the world of science and industry in developing autonomous vehicles' environment and event detection capabilities. Nowadays, autonomous vehicles use artificial intelligence technology and additional sensors and devices to sense the environment and perform partially automatic driving as robot driver. Figure 1.1 shows how autonomous vehicles detect important environmental objects or events while driving.

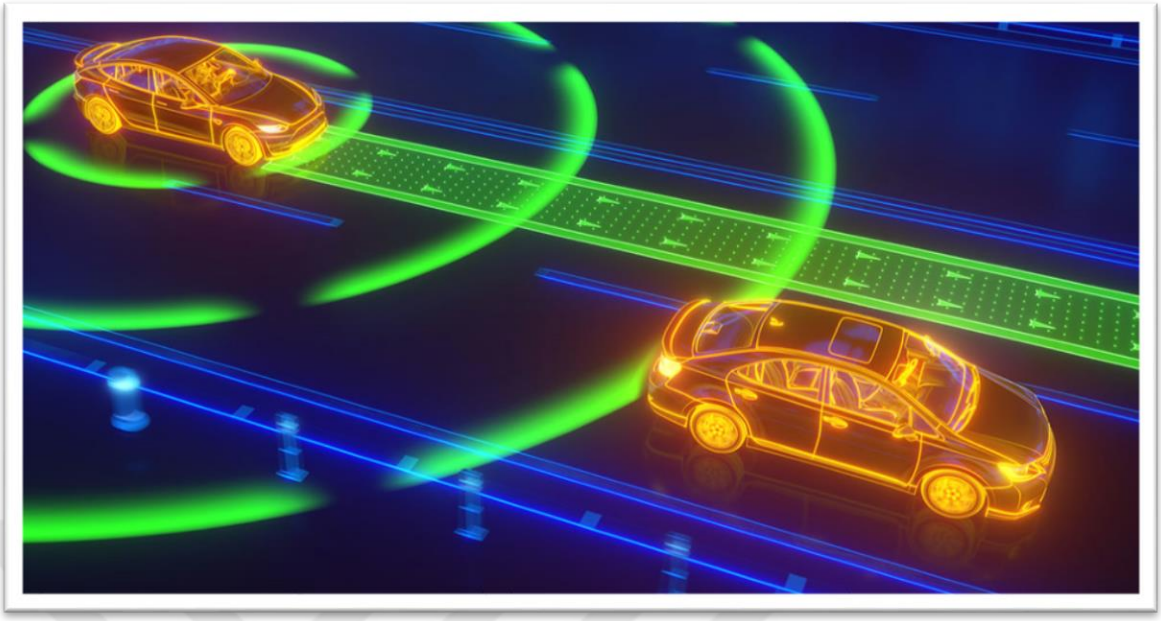


Figure 1.1. sensing technologies of vehicles.

1.3. Autonomous Driving Levels of Vehicles

In 2013, the Automotive Engineers Association took an approach that classifies the autonomous driving characteristics of vehicles. OMD evaluated the automation levels of the vehicles in the 0-5 range by prioritizing the technical definitions of the vehicles [6]. According to these evaluations, "Level 0" vehicles do not have any automation feature. The rides take place full-time under the control of the driver. Many modern cars have vehicle speed control, electronic stability, emergency braking, and driver assistance. Such vehicles only have one driver's assistant and belong to the "Level 1" type of automation evaluation.

Today, partially autonomous vehicles with lane-following features during public use have undergone an important development process. Vehicles defined as "Level 2" have two or more automation systems and several driver assistance features simultaneously. Since they do not react to unexpected events, second-level vehicles require human monitoring and intervention of road conditions and risky events during the journey.

"Level 3" vehicles perform conditional automation by detecting the driving environment, static and dynamic objects, and traffic events in the environment through various sensors. These vehicles require less human tracking at the time of driving, with the ability to make wireless connections to other movement participants in the traffic and infrastructure

elements on the roads and send and receive data with the devices they communicate with. These tools are perceived as conditional automation. That is, the vehicle can be driven independently under certain conditions. For example, weather or road types can belong to these conditions.

Vehicles that can drive automatically in terms of dynamic driving, steering, and speed change without waiting for a human response in emergencies and dangerous situations and requiring minimum human intervention in the defined conditions are defined as "Level 4" in the automation classification [7]. Vehicles with the second name "High Automation" perform a significant independent movement.

"Full Automation" vehicles reach the target address without the need for human control while driving. These tools belong to the definition of "Level 5" for public use in the future [8]. Automation levels classified according to their degree of autonomy are shown in Figure 1.2.

However, the development ideas for advanced autonomous cars mostly focus on a scenario and task-oriented techniques to accomplish specific functional development under predetermined scenarios. At the same time, the research directions adopted by various universities have been specifically highlighted. This idea may be utilized to implement certain functions inside predetermined scenarios and has a special control function for the design of "Level 1" to "Level 3" driving assistance systems. Task-oriented, "Level 3" and "Level 4" autonomous driving can, under some circumstances, allow shared driving between people and robots. The scenario-oriented approach might be problematic for "Level 5" vehicles since there are many conceivable scenarios, and the vehicles must adapt to each one. There are too many restrictions because of more mission-focused concerns, which hinder the growth of Autonomous cars at "Level 5," which are capable of unrestricted and free driving [9]. For instance, adherence to traffic regulations may not be strongly and sufficiently understood. Due to distinct road user behavior patterns and unique aspects of the driving environment, the implementation challenges of current scenario- and task-oriented development approaches have greatly increased [10].

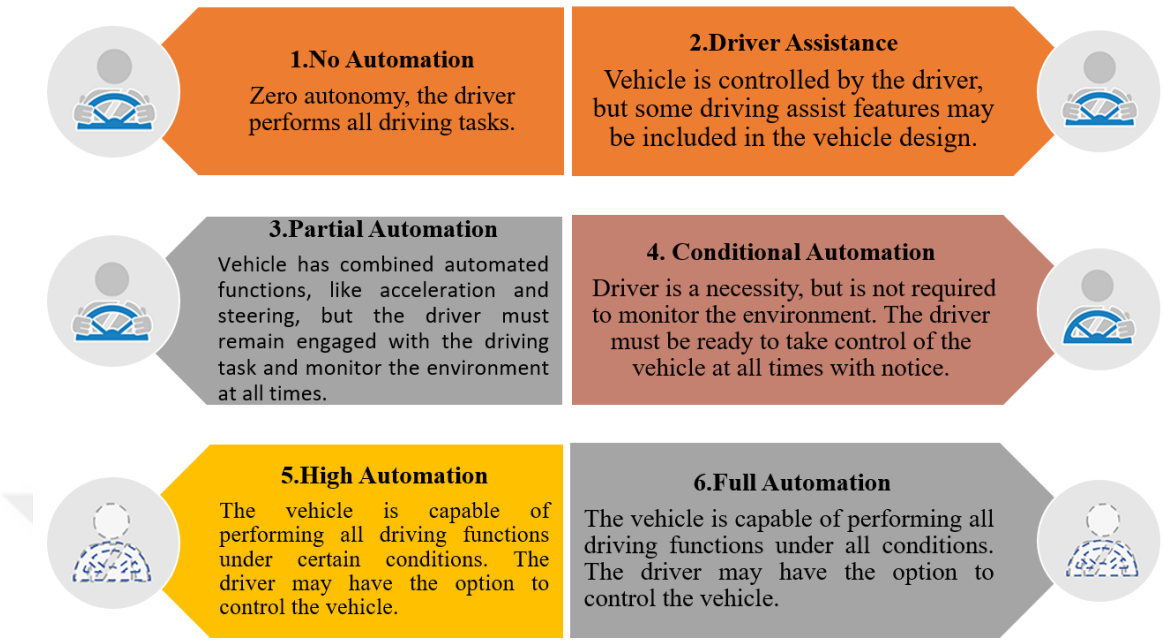


Figure 1.2. Autonomous driving levels of vehicles.

It is assumed that "Level 5" autonomous vehicles will operate at a level above that of human drivers. It is believed that "Level 5" autonomous vehicles will eventually surpass the limitations of human driving, as they often learn from and adapt to human driving styles [11]. Several study paradigms examine the physical operation of sophisticated self-driving systems and reevaluate the nature of driving from a systematic, cohesive, and impartial standpoint. Considering the flaws of present procedures, an effort is made to abandon traditional research methodologies using a novel style of reasoning and deduction. Before studying the self-learning paradigm and previous knowledge research, the framework of coordination and balance based on the brain, cerebellum, and organ principles is first examined. The groundbreaking research's objectives are to achieve self-learning, self-adaptation, and self-transcendence features for autonomous cars and to present fresh research ideas for the design of "Level 5" autonomous vehicles [12].

1.4. Current Situation of Autonomous Vehicles

According to market research conducted by Ipsos on behalf of Audi, 98% of automotive industry participants in China are interested in the production of autonomous vehicles. Only 72% of participants in the US and 77% of participants in Germany are autonomous vehicle entrepreneurs.

Today's autonomous vehicles on the market and offered for public use are at "Level 2" in the automation classification. Autopilot from Tesla, Super Cruise from GM, and ProPilot from Nissan are examples of "Level 2" autonomous driving systems examples. "Level 2" vehicles operate with some limitations, although they provide steering, acceleration, and braking management. Therefore, you need a driver who is always ready to take control. Such vehicles perform automatic driving with route planning based on environment maps, traffic data, and lane tracking based on image processing. However, inadequate infrastructures and regulatory restrictions currently limit the use of the full potential of these vehicles. For example, the color stripe cannot be detected with sufficient resolution due to infrastructure-related problems. However, the issue of penalizing the driver or the vehicle manufacturer's company in cases such as traffic accidents and non-compliance results in controversial issues [13]

"Level 3" vehicles have been available in the industry since 2017, and these vehicles are characterized by their ability to establish connections with other road users, traffic signals, and traffic elements. However, they are not widely used on highways because they are legally in a "grey area." Only a few "Levels 3" capable cars, like the new Honda Legend, are available in limited numbers in certain regions. Audi has disabled its "Level 3" system on the new A8 because of the current limitations in road infrastructure and legal regulations.

Google Waymo's fleet is among the "Level 4" vehicles undergoing test drives in several US states. Although the "Level 4" autonomous vehicles have a high level of autonomy, they require a driver who is always prepared for unexpected situations, considering the principle of worst-case worst decisions. Due to the lack of infrastructure faced by previous generations of vehicles, these vehicles are also not used in public transport. In "Level 5" vehicles, there is no way for the driver to take control and intervene in what is happening. An example of a vehicle level that promises fully automated driving is Audi's “AI: CON” concept.

1.5. Advantages of Self Driving Vehicles

The rise of human and social well-being progresses with the application of modern technologies. The development of human-independent, automatic systems affects people's lives differently. From the perspective and actual outcomes, there are different

considerations of the advantages and disadvantages of such escalation. These approaches, which prioritize safety, time and cost savings, and pollution, are important for evaluating the potential of autonomous vehicles, one of the latest achievements in the technological world.

1.5.1. Challenges in Urban Traffic

Road traffic incidents frequently result in fatalities and serious injuries. According to the World Health Organization reports, there were 1.35 million traffic deaths worldwide in 2016 [14]. According to the National Highway Traffic Safety Administration, there were 51,872 cars and 33,645 fatal collisions in the United States in 2018 [15]. The United Kingdom Department of Transportation statistics indicated 160,597 injuries of all severities in police-reported road traffic crashes in 2018 [16]. In 2017 the Chinese Ministry of Public Security reported 8.419 million traffic accidents. There were 203,049 fatal and serious injury accidents, in which 63,729 people died, and 209,654 were seriously injured. In 8.216 million traffic accidents, cars alone were damaged or slightly injured, and 1.465 million people suffered minor injuries [17]. For 2021, a statistical projection of traffic fatalities shows an increase of about 10.5 percent compared to reported fatalities in 2020. Some categories showed large shifts in fatalities and fatality rates in each month compared to the corresponding month in 2020. For instance, the share of fatalities on urban roads went from 57 percent in March 2020 to 62 percent in March 2021, a 5-percentage point increase. Correspondingly, the total fatalities (fatality counts) on urban roads increased from 21,940 in 2020 to 25,411 in 2021, a 16-percent increase. In summary, the traffic fatalities (fatality counts) relatively large increases in 2021 compared to 2020. In addition, Indiana College conducted a study that showed that human factors were the main cause or culprit in 93% of the accidents studied. In addition, the US National Road Transport Safety Administration reports show that accidents caused by human error account for 90% on average [18]. The fact that traffic accidents depend on human mistakes generally. Autonomous vehicles will operate more carefully, analyze the environment and events more quickly, and give the necessary response is high. One of the benefits expected from autonomous vehicles in the future is that the number of traffic accidents will greatly decrease. In terms of human safety, fewer accidents, fewer fatalities, and in economic terms, reducing accident costs is one of the benefits that

autonomous vehicle technologies will provide to society. Based on this information, fully driverless vehicles will result in 90% fewer traffic accidents.

Three key elements may be used to characterize how autonomous vehicles affect traffic congestion. Reduction in traffic delays with a significant reduction in accident rates, increase in vehicle efficiency, and change in total vehicle miles traveled.

An expected reduction in vehicle accidents will lead to reduced delays and, thus, higher transportation system reliability. The changes in total vehicle mileage due to the advent of autonomous vehicles are still unclear, but some researchers believe that mileage will increase. They speculate this will be due to increased mileage from self-fueling and self-parking combined, increased use of autonomous vehicles by non-drivers, increased number of trips (both empty and frequent), and distance from the vehicle. The National Highway Traffic Safety Administration concluded that at 10%, the recovery rate will remain stable.

The proliferation of autonomous vehicles may also be a way to reduce the burden of traffic congestion. A reservation-based approach is suggested by Dresdner and Stone [19] to lessen traffic congestion, particularly at crossings where cars are linked. Their findings demonstrate that the reservation-based system for communication between autonomous vehicles can surpass traffic signals by a factor of two to three. It can thus easily manage situations with significantly more crowded traffic [20]. Dresner and Stone demonstrate that traffic delays drop to the levels shown in their prior experiments as the number of autonomous cars on the road rises [21].

The result shows that autonomous car technology will soon benefit from easing congestion, if it does not spur increased demand that may put further strain on a network that is already crowded [22].

1.5.2. Traffic Signs in Self Driving Cars

In recent decades, many efforts are presented for autonomous cars. Self-driving vehicles are more dependable [23]. but it is still challenging to drive autonomously in an actual metropolitan setting [24]. For an autonomous vehicle to perform appropriate activities at an intersection in an urban context, active localization and detection of regulating traffic

signs are essential. However, strong activity traffic sign detection is difficult because there would be a frightening jumble of objects in an image whose colors match those of the activity, and it is difficult to extract enough highlights because the shape of the sign is so simple [25]. Some traffic light and signs taken from Turkey is given in Figure 1.3. Even more unfortunate is that there are many types of signs, some horizontal, some vertical, some circular, and some with screws.

Figure 1.3. Traffic lights and signs of several sorts.

Human vision is the best method for determining the status of traffic signals. Cameras to detect traffic signals are present in almost all autonomous test vehicles for urban environments. In recent years, numerous traffic sign detection methods have been presented and have successfully used the algorithms to detect shapes and coordinate the layout to identify traffic signals. However, these tactics are based on fixed cameras. These techniques would be less effective if coupled with independent cars when the car camera is traveling with the car. In [26]. measurement data in HSV color space was used to determine color boundaries. After that operation, images were partitioned based on these boundaries, and finally, a machine learning algorithm was applied for classification. In [27]. they use a Gaussian mask to look at the activity centers of the traffic lights and a proposed existence weight contour to identify the potential traffic light opportunities. In any case, only the circular traffic lights can be distinguished and recognized with these two techniques.

Furthermore, the endeavors mentioned above are utilized to categorize and identify traffic lights based on onboard images without explicit priority maps. For an a priori outline, they mapped more than 4,000 objects using predefined methods to spontaneously map the positioning of traffic signals in three dimensions [28]. An inactive camera-based traffic light localization and detection pipeline is utilized on Google driverless cars and successfully completed several tests in a real urban environment. Great accuracy was attained in an image with the information on the traffic signs zone in both algorithms. The fundamental disadvantage of the approaches is that they call for the previous identification of traffic sign positions and vehicle localization.

1.6. Transfer Learning

Given the rarity of large data sets, very few people can train a complete model from scratch. But many scientists like pre-training a model on a huge dataset, like ImageNet, and then using its characteristics. It is claimed that transferring knowledge improves learning and reduces the effort required to retrieve information [29].

"What to transfer", "How to transfer", and "When to transfer" are all issues that are well expressed. Regarding "How to transfer", "What to transfer" refers to the specific component that needs to be moved between domains. "When to transfer" simply refers to when to apply what has been learned. The above statement gives the context of transfer learning in this study. Transfer learning is acquiring knowledge from one task and applying it to another in deep learning models. In transfer learning, the coherent relations (weights) between the initial and the target tasks are transferred to the new application area. The weak link between the initial and target task leads to negative transfer. Therefore, in learning the target task, it is crucial to recognize and reject incorrect information. In this case, performance is at least no worse due to transfer learning than without it. This case illustrates the need to carefully consider transfer learning and avoid negative transfer in my study [30]. Transfer learning is the enhancement of learning for a new task based on applying information from a related task that has already been learned. Transfer learning examples include computer vision, which produces better object categorization outcomes. Fortunately, some techniques aim to avoid overfitting the neural network, such as dropout and data augmentation [31].

1.7. Literature Review

Algorithms for uncontrolled intersections are primarily intended to provide passing cars with an effective means of negotiation and cooperation to gather information needed to enter conflict zones at uncontrolled intersections. The area of an intersection where two separate transit vehicles pass at the same time is referred to as a conflict zone. There are two main categories into which It is possible to categorize algorithms for uncontrolled intersections [32].

One class of techniques uses centralized management. As they approach the junction, vehicles interact with a central controller. The response from the central controller contains the instructions each vehicle needs to cross the intersection safely. A simple core system based on first-in-first-out (FWC) priority was presented by Dresdner and Stone [32]. The authors of this work used a controller and instrument-based multi-agent time reservation system. A car requests a small window of time to pass through the intersection as it approaches. When the controller receives a request from the driver's agent, it simulates a vehicle crossing an intersection and then makes decisions based on exit paths that minimize accidents.

The safe passage of cars at uncontrolled intersections as a planning problem has been the subject of several research projects. Colombo and Vecchio addressed this problem by finding the largest controllable invariant set and using an algorithm approximating the answer [33]. The collision avoidance controller was designed using this approximation. Colombo and Vecchio's method have limitations as it requires error-free snapshots and data.

By removing the above limitations, Bruni and Colombo improved the architecture and developed a controller to manage the inaccurate state information and unclear input [34]. The two controllers mentioned above predicted all cars would have driver assistance technology (controlled cars). A rigorous idle scheduling approach was put out by Ahn and Colombo [35] to enable controller design when there is more than one uncontrolled vehicle.

In the other method category, distributed, i.e., decentralized, control was used. Vehicles approach each other immediately when they approach an intersection. (V2V) Then, the vehicle that has the right-of-way at each intersection is selected. A set of policies

developed by Guanghua and Lumia is used to prioritize traffic passing through an intersection. Adherence to these guidelines has been shown to help resolve disputes and prevent accidents. According to the guidelines, each approaching driver consults the other cars for information before determining whether to "yield" to them. A decentralized algorithm using a navigation function was proposed by Makarem and Gillet [36]. Their algorithms selected tools based on various parameters to maximize security and latency.

A thorough intersection utilization strategy is developed when the leading vehicles in each approach lane communicate. The system reduces the overall latency of the intersection by prioritizing traffic in the heavier lanes. Van Middlesworth and Dresner presented another decentralized method, but it requires coordination between parties to adjust stop signs and program transit vehicles at local intersections.

A central control simulator was proposed in the related study by Zohdy and Rakha [37]. The paths of the vehicles were continuously changed by the optimization algorithm of the control simulation to reduce the time they spent in the conflict zones of the intersection. The application of game theory in this study resulted in a comparison to traditional junction control techniques like the use of traffic signals or stop signs, and proposed method considerably reduce in vehicle delay [38].

1.7.1. ResNet Model

The ability of the layers inside deep networks to extract the attribute map of the data is the essential characteristic that sets them apart from conventional artificial neural network models. These constructions enable the assessment of attribute maps produced by applying data to filters of various numbers and sizes. When the image is questioned for testing, it is easy to tell which image is there because of this [39].

Conventional neural networks began to perform admirably following the architecture that arose with the victory of "AlexNet" due to the ImageNet competition in fields like object detection, picture segmentation, classification, face recognition, human location estimation, and tracking. Along with these adjustments, it was observed that there are crucial principles to consider while obtaining best practices for prospective future network architectures [40].

Recent years have seen many developments in image categorization thanks to deep convolutional neural networks. They were inspired by the neuroscience research of Hubel

and Wiesel on receptive fields and basic cells. Deep convolutional neural networks (CNNs) have a layered structure, with convolutional filters making up each layer. By disclosing the parameters, it is possible to quickly understand how to combine these filters with the input picture to produce feature vectors for the following layer. The initial layers of convolutional neural networks represent low-level local data such as edges and color contrasts. In contrast, the deeper layers aim to capture more complex forms and are more specialized. Deepening the model and making these convolutional filters more diverse and particular would increase CNN's classification performance.

Deep networks are harder to train for two reasons, even though they often produce better classification results: First, gradients disappear or explode: Depending on the neuron's activation function, a neuron may disappear during the training phase and not recover. Initialization strategies that aim to start the optimization process with an active group of neurons can solve this problem.

The second optimization is more challenging because training the network when the model adds more parameters is more difficult. Since adding more layers leads to even more training mistakes, it is only sometimes an overfitting problem [39].

Deep CNNs are, therefore, more difficult to train, even if they provide better categorization performance. ResNet networks are a practical approach to address these issues (ResNets). ResNets differ from conventional convolutional layers primarily because they have parallel shortcut links. Unlike convolutional layers, these links are constantly active, and gradients can easily propagate backward, speeding up training. The difference between ResNets and standard ConvNets is simple. The goal is to obtain a direct channel for gradients that runs through the first layers of the net. Learning is accelerated by eliminating the problem of fading gradients and dead neurons. By solving the vanishing gradient problem, it took first place in the 2015 IMAGENET competition. While error rates in computer vision operations range from 5% to 10%, this network achieved an error rate of 3.57% [39].

This architecture was proposed by the developers of ResNet in 2015 using the VGG19 model as inspiration. This architecture is one of the most complex developed for the ImageNet dataset. In a CNN, numerous layers are typically interconnected and trained to perform various tasks. At the end of its layers, the network learns different levels of

features. The convolutional layers of this model usually include 33 filters. The layers have the same number of filters for the same size as the initial attribute map in ResNet. The number of filters is doubled when the feature map size is reduced by half to maintain the time complexity for each layer. Subsampling is immediately performed in two phases using convolutional layers. The global average commonality layer and a SoftMax-enabled final ResNet layer are fully connected layers (FC).

Residual learning can be defined as the extraction of input features learned by this layer. The ResNet learning process is connected to each of the 33 filter pairs via links. It performs the layer's input by connecting (k+x) directly to the layer. The reason for skipping layers is to reuse the activations from the previous layer to eliminate the vanishing gradient problem until we know the weights of the current layer. When training the network, the weights are adjusted to strengthen the layer next to the current layer and mute the previous layer. It has been shown that training this network is simpler than training simple deep convolutional neural networks. It also solves the problem of reducing accuracy. ResNet is characterized by a series of parallel short-path connections between each of its linked basic blocks, which can be added to its output sequentially. Figure 1.4 shows a core component of ResNet. The shortcut connection is a discriminant matrix when input and output sizes are the same for a basic block. Otherwise, the size can be changed using average pooling (to reduce size) and zero padding (to increase size). He demonstrated how adding a parameterized layer after addition can reduce the benefits of ResNets by removing the fast path that gradients previously had for propagation Figure 1. 4. However, under these circumstances, adding an unparameterized layer such as ReLU or Dropout after the addition module has no significant advantage or disadvantage or disadvantage.

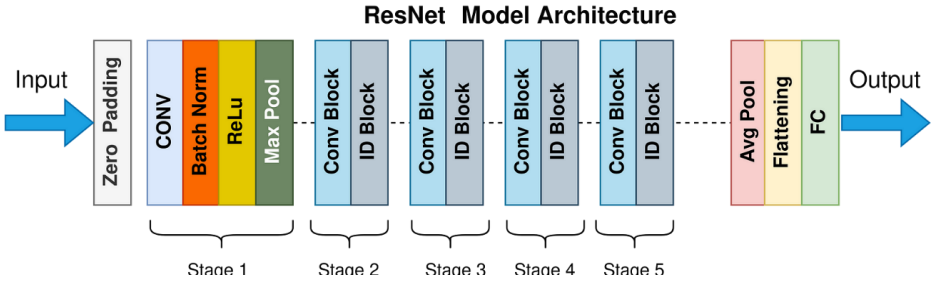


Figure 1.4. ResNet architecture [41].

1.7.2. MobileNet

Google's MobileNet architecture is designed to deliver positive results for image processing research in embedded and mobile devices. The hyperparameters of this architecture, made possible by the thorough deconstruction of convolutional layers, allow users to choose between model success and speed. Researchers can achieve the desired result by customizing the network using parameters while solving their challenges. Despite their basic similarity, the two different MobileNet designs differ significantly in the "rest" structures they have in reverse.

A lightweight deep CNN using the Howard et al. [40] depth-wise separable convolution-based MobileNet architecture makes a model too tiny and speeds up processing. The MobileNet architecture is depicted in the schematic in Figure 1a. MobileNet may solve various identification issues, including object detection, facial characteristics, fine-grain classification, and landmark recognition.

The following was the architecture for MobileNet. First, we used the pre-trained MobileNet model. We eliminate the last three layers from the original network, including the average pooling, completely linked, and SoftMax layers. Second, related to our classes, three additional layers—the SoftMax, batch normalization (BN), and global average pooling (GAP) layers—are applied. We suggested MobileNet architecture's primary goal is to assist the network in training more quickly and accurately. The dropout strategy is then suggested to stop overfitting.

Additionally, the batch normalization layer facilitates quicker network training. Between the batch normalization and dropout layers, the rectified linear unit (ReLU), a kind of activation function, is calculated. Instead of using the standard pooling, we applied the GAP layers, demonstrating how the model's parameters are being reduced, which has an immediate influence on the model's size. Finally, we trained the proposed network on the ETH Food-101 dataset using MobileNet, that had been tweaked.

The MobileNet architecture is computed based on depthwise separable convolutions (DS). The concept of decomposition of convolution called factorization is considered to factorize a standard convolution into a depthwise convolution.

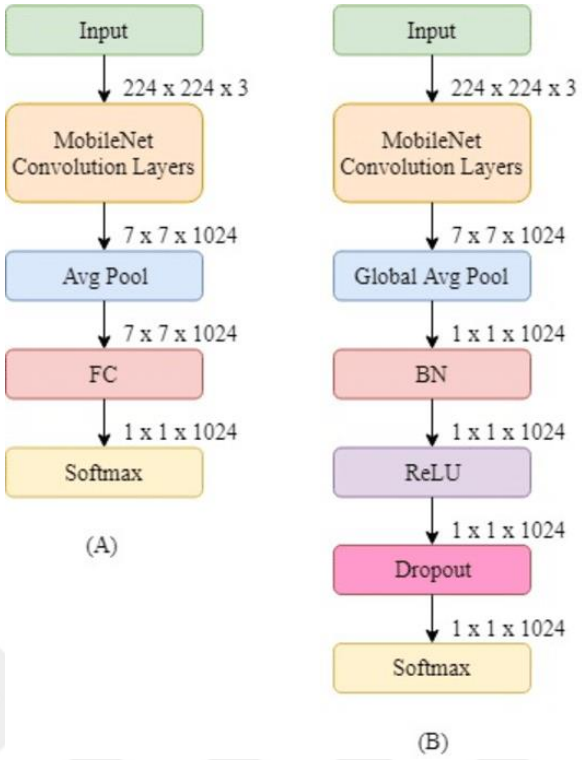


Figure 1.5. The architectures of the MobileNet. (A) the original MobileNet and, (B) the proposed MobileNet architectures [42].

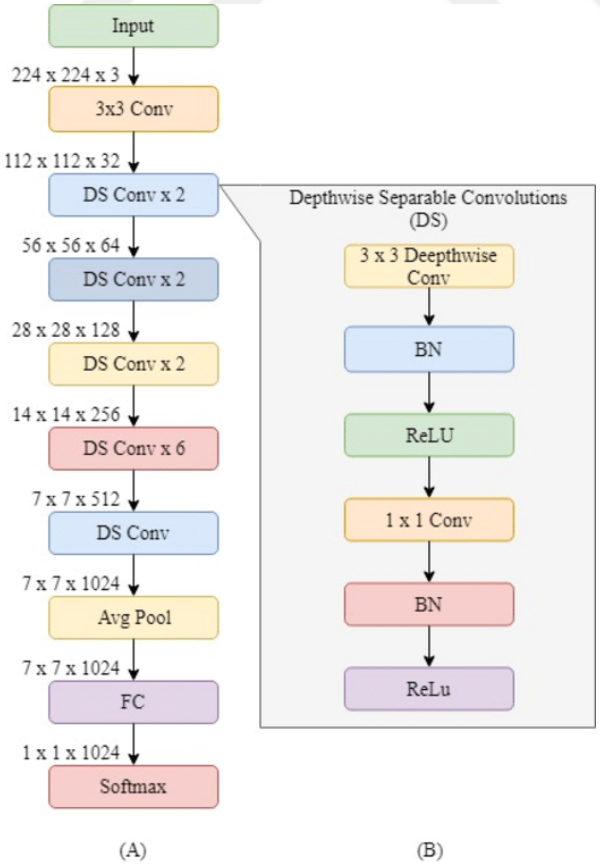


Figure 1.6. Illustration of the MobileNet architecture. (A) The overall MobileNet architecture and (B) an in-depth explanation of the DS layer [42].

Following that, a 1x1 convolution, known as a pointwise convolution, is used to compute all depth-wise convolution layers before combining them as the outputs for the following layer. The MobileNet, which consists of convolutional, depth-wise separable convolutions (DS), average pooling, fully connected (FC), and SoftMax layers, is depicted in detail in Figure 1.5.

An extensive description of the depthwise convolution (DC), batch normalization (BN), and rectified linear unit (ReLU) components of the DS layer are shown in Figure 1.6.

1.7.3. VGG Model

VGGNet was developed by Simonyan and Zisserman, two members of the Visual Geometry Group at Oxford College. It achieved a remarkable result at the TOP -5 level of the 2014 ILSVRC competition, which included more than 14 million data and 1000 classes. The model, which has a very similar structure to ALEXNet, reduces the filter structure of ALEXNet from 11x11 and 5x5 dimensions to 3x3 dimensions. For VGGNet, there are models with 16 and 19 layers. The VGGNet16 model consists of 13 convolutional layers and three fully linked layers, while the VGGNet19 model consists of 16 convolutional layers and three fully linked layers [43].

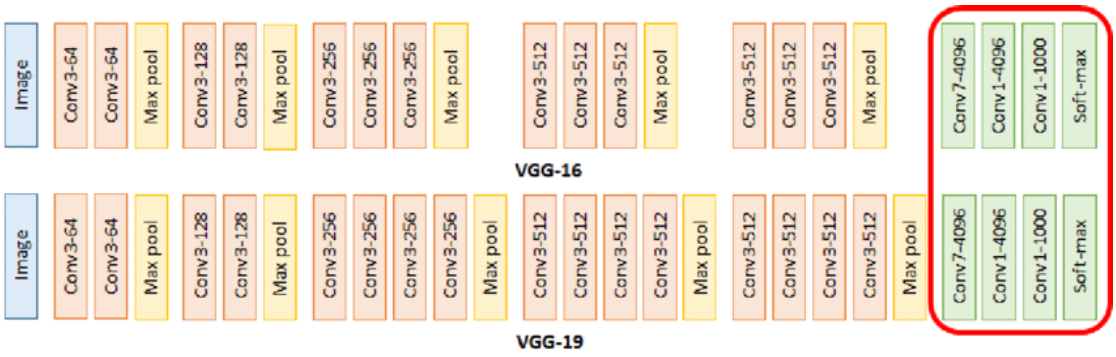


Figure 1.7. Images of VGGNet Models. (a) VGG16 and (b) VGG19 [44]

Typically, "VGGNet" refers to a deep convolutional network for object recognition introduced and trained by Oxford's renowned Visual Geometry Group (VGG), which has shown excellent results on the ImageNet dataset. Convolutional layers piled on one another at increasing depths were used to introduce the VGG network. Max pooling reduces the volume size. A softmax classifier is then followed by two fully connected layers with 4,096 nodes each. To limit the number of parameters in this deep network,

VGG utilizes relatively small convolution filters (3 3), with a convolutional step equal to 1 pixel [45]. VGGNet comes in two models: VGG16 and VGG19.

VGG16 features 16 convolutional layers with relatively tiny receptive fields (3x3), five max-pooling layers of size 2x2 for performing spatial pooling, three fully connected layers, and the soft-max layer as the final layer, a total requiring 144 million parameters [46]. All buried layers receive ReLU activation. In the fully linked layers of the model, dropout regularization is also used. More than a million photos from the ImageNet collection were used to train VGG16. A keyboard, mouse, and pencil are just a few of the 1000 item categories into which the network can identify photos [46]. The VGG16 model uses the 224*224*3 (RGB picture) input image size. The number of completely connected nodes and depth causes the VGG to be larger than 533 MB for VGG16. Sixteen deep neural network layers make up the VGG16 Neural Network. Less weight is given to the VGG16 network. More than a million photos from the ImageNet collection were used to train CNN VGG19.

The application of the VGG has 1000 categories for images such as photos, such as keyboard, mouse, and pencil.

The VGG19 model uses the 224*224*3 (RGB picture) input image size. Due to the quantity of completely linked nodes and deep, the VGG for VGG19 is larger than 574 MB. 19 layers of deep neural networks make up VGG19. More weight is present in the VGG19 network (138Mweights and 15.5M MACs [54]). VGG16 and VGG19 architectural schematic developed using the ImageNet database is shown in Figure 1.7.

1.7.4. Xception Model

On the other hand, Xception networks have a structure with small improvements to the Inception models. Applying the reconvolution process to the 1x1 convolutional output channel is its contribution. Moreover, this architecture is an inception model with a deep parsable convolution block instead of the inception block. When the ReLu activation function is not used, the results are better. The parameters of the inception model are generally comparable [47].

François Chollet proposed the Xception model [48]. Depthwise separable convolutions are the theoretical foundation of this paradigm. Depthwise Separable Convolutions are a variation to regular convolutions that are intended to be faster to compute. Two primary processes make up Depth Wise Separable Convolutions: (1) Depth wise Convolution, and (2) Pointwise Convolution.

a) Depth wise convolution

The standard convolution is carried out by employing dxdxC kernel, where d is the kernel size if the supplied input size is AxBxC, where an image AxB is the width and length of the image, and C is the depth of the image, commonly 3 for RGB pictures. The kernel size for depth wise convolution is dd1. In other words, only one channel is used for each convolution computation rather than all the channels. The size of the resultant convolution is KxKxC.

b) Pointwise convolution

With a kernel size of 11N across the KxKxC volume, pointwise convolution acts the same way as conventional convolution. This approach enables a volume with the previously mentioned form KxKxN.

In the Xception model, the sequence is flipped. The first step is pointwise convolution, while the second is depth wise convolution. Figure 1.8 depicts the Xception model's architecture. The three major building components of the Xception architecture are shown in this picture. Data first passes through the input flow, which comes first. Following that, the data pass via the middle flow, which is iterated eight times, before eventually passing through the exit flow [49].

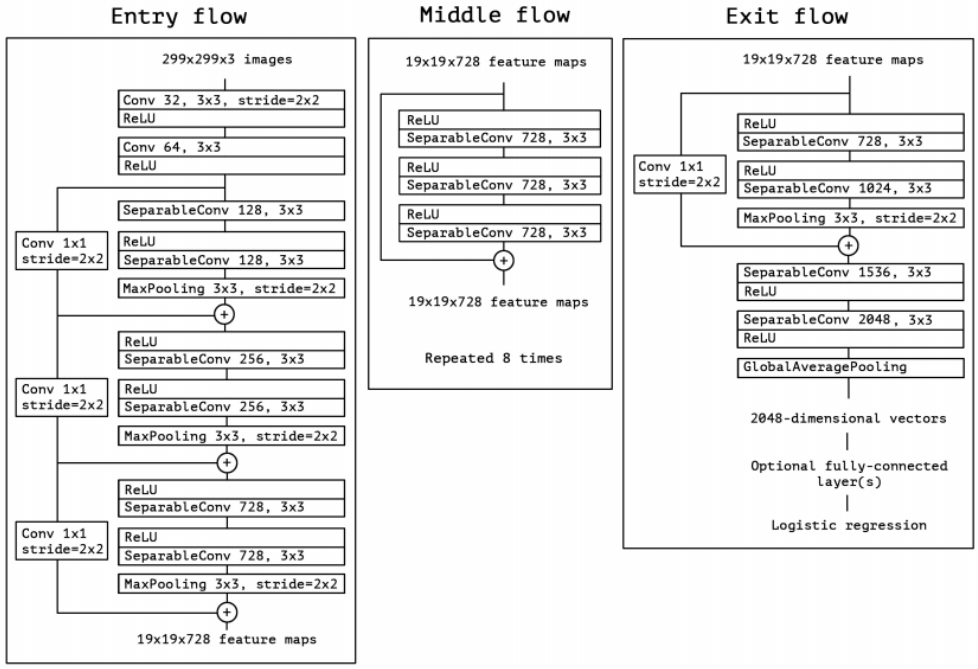


Figure 1.8. The Xception architecture [50].

1.7.5. DenseNet Model

Huang et al. developed DenseNet with the hypothesis that short connections between the first layers and subsequent layers in CNN models could be deeper, more precise, and more effective. In the traditional CNN paradigm, DenseNet has layers that are all fully connected. Unlike a typical CNN with N layers, which has N connections, the DenseNet model has N(N+1)/2 connections per layer because each layer is connected to the layer above it. The feature maps of all previous layers are used as input to each layer, and the feature maps of each layer serve as input to all subsequent layers. In this way, DenseNet and ResNet attempt to solve the vanishing gradient problem, improve feature propagation, promote feature reuse, and dramatically reduce network complexity and the number of parameters. DenseNet has variants with 121, 169, 201, and 264 layers.

Densely linked networks DenseNets; [19] comprise several dense blocks with numerous layers inside each one. Each layer generates k features, where k is the network's growth rate. The unique characteristic of DenseNets is that each layer's input concatenates all the feature maps produced by all previous layers inside a single dense block. As seen in the left portion of Figure 1.9, each layer applies a series of subsequent changes. The first transformation (BN-ReLU, blue) comprises rectified linear units and batch normalization. Utilizing 11 filters, the first convolutional layer in the series decreases the number of

channels to lower computing costs. Another BN-ReLU follows the output. Transformation is then reduced to the final k output features through a 3×3 convolution.

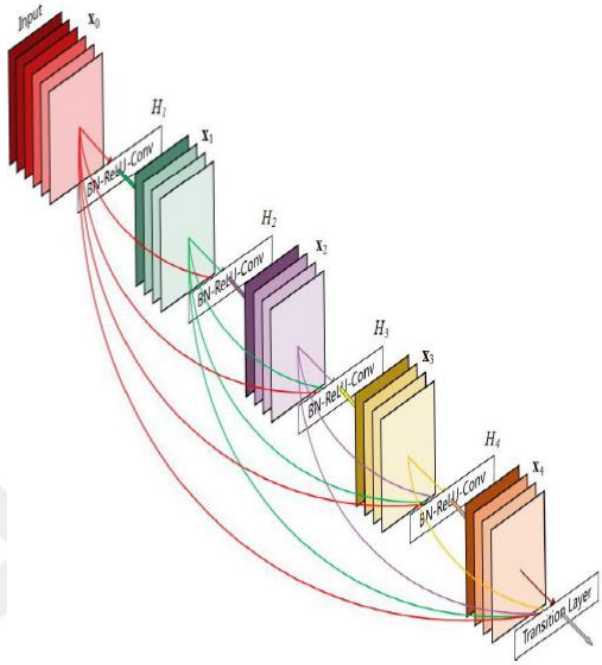


Figure 1.9. DenseNet Architecture [51].

1.7.6. EfficientNet-B0

EfficientNet-B0 is the entry-level model of the EfficientNet family. The EfficientNet-B0 was designed by the neural network, not by engineers. These researchers developed this model using a multi-objective neural architecture search method that maximizes accuracy and floating-point operations.

The scientists introduced a comprehensive family of EfficientNet-B1 through EfficientNet-B7, starting with B0 as a baseline model. These networks achieved cutting-edge accuracy on ImageNet while being extremely efficient compared to its rivals. The EfficientNetB0 fundamental model is given in Figure 1.10. We suggest using the EfficientNetB0 baseline model, which accepts an input picture with a size of 224x224x3, as our starting point. The model then extracts features from each layer from numerous convolutional (Conv) layers with a 3x3 receptive field and the mobile inverted bottleneck Conv (MBConv). We naturally chose to use the EfficientNet-B0 because of its well-balanced depth, width, and resolution, which can result in a scalable, accurate, and simple model to install. EfficientNet-B0 scales each dimension using a fixed set of scaling coefficients compared to other DCNNs. This method outperformed previous cutting-edge

algorithms developed using the ImageNet dataset. EfficientNet showed remarkable results even with transfer learning, demonstrating its usefulness beyond the typical ImageNet. The model had scales of 0 to 7 when it was first released, indicating improved parameter size and even precision. Users and developers may now access and deliver enhanced ubiquitous computing enhanced with DL capabilities across several platforms for various applications thanks to the current EfficientNet [52].

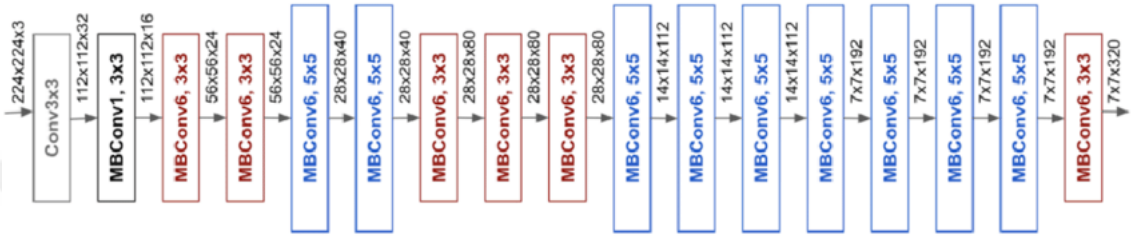


Figure 1.10. EfficientNet-B0 baseline model architecture [52]

CHAPTER 2

MATERIAL AND METHODS

The method presented in this report uses deep neural networks to analyze one of the most important parameters in self-driving (autonomous) vehicles, namely the detection of traffic signals and the corresponding state classification. Instead of relying on the traditional schemes where image processing methods need to be applied only for background reduction and region extraction of traffic signals, this approach uses deep networks such as ResNet50, MobileNet, VGG19, etc., to process the images in their raw form so that no additional image processing or region convergence techniques are required. The models were trained using the Google Colab platform..

2.1. Material

The two-dataset involved in the study are as follows:

2.1.1. Turkish Traffic Sign Dataset

The dataset consists of 11023 images in total. Thirty-four different traffic signs are included in the dataset. The Turkish traffic sign dataset is used as a training set in the dissertation study. Fehim KOYLU collected this dataset. System modeling starts with the collection of data in the form of images in different scenarios. The traffic signs were captured in their respective states against different backgrounds. In addition, the dataset included data at different times of the day, including morning, afternoon, and night, to allow the extraction of different conditions that can help evaluate the robustness of deep learning models. In addition, the zoom levels of the images were kept different, as some images were taken near traffic lights, while others contained more background images. Below are some sample images of signs from the dataset. As can be seen, the dataset is classified and thus falls within the domain of the supervised learning mechanism. We

selected 34 traffic signs classified by importance and necessity in the road, and each traffic sign was captured as a real image. The details of the dataset are given in Table 2.1, such as id, number of samples, the meaning of the traffic sign, and shape.

2.1.2. German Traffic Sign Recognition Benchmark (GTSRB)

The GTSRB dataset has 44 classes, and the collected image size is 32x32x3. Both the images have been passed through normalization to map pixel values between 0 and 1. The Turkish traffic sign dataset images have been resampled to 32x32x3 to retain uniformity between datasets. The details of GTSRB are given in Table 2.2.

The current study involved transfer learning models (originally trained on an extensive database of image nets). The models have been developed to serve as classifiers converging the desired number of classes. Two datasets have been investigated by passing through the training, validation, and testing phases. Labels in each dataset have been transformed using one-hot encoders, which fit the Keras sequential models well. The parameters given in Figure 2. 1. are used in evaluations to adopt a general architecture for various transfer learning models.

Table 2.1. Description of the Turkish traffic sign dataset.

Class Id	Number of images	Traffic Sign Label	Image
1	134	School crosswalk	
2	1317	Crosswalk	
3	291	Pedestrian crossing	

4	118	School crossing	
5	444	Maximum speed limit (30 km/h)	
7	267	Maximum speed limit (50 km/h)	
8	40	Maximum speed limit (20 km/h)	
9	49	Curve to left	
10	76	Curve to right	
11	331	No right turn	
12	458	No left turn	
13	975	Traffic light (green)	

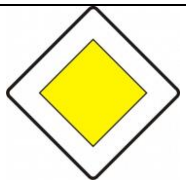
14	567	Traffic light (red)	
15	187	Traffic light (yellow)	
17	13	Proceed straight	
18	116	Compulsory turn right	
19	27	Compulsory turn left	
20	64	Compulsory turn left	
21	583	Compulsory turn right	
22	1038	Compulsory ahead or turn left	
23	369	Compulsory ahead or turn right	

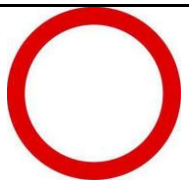
24	10	Double curve, first to right	
25	9	Double curve, first to left	
26	229	No U-turn	
27	460	No standing or parking	
28	390	No parking	
29	65	Parking	
30	60	Buses stop	
33	499	No entry mopped	
34	849	Stop	

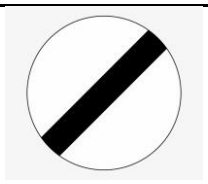
36	280	Bump road	
38	83	Be careful	
39	333	Yield	
40	292	Side road	

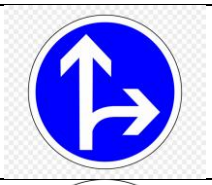
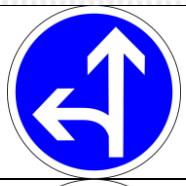
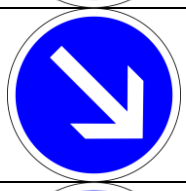
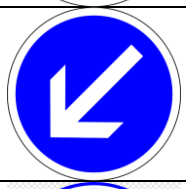
Table 2.2. The details of GTSRB dataset.

Class Id	Number of images	Traffic Sign Label	Image
0.	151	Maximum speed limit (20 km/h)	
1.	1501	Maximum speed limit (30 km/h)	
2.	1501	Maximum speed limit (50 km/h)	
3.	961	Maximum speed limit (60 km/h)	

4.	1321	Maximum speed limit (70 km/h)	
5.	1261	Maximum speed limit (80 km/h)	
6.	301	Maximum end speed limit (80 km/h)	
7.	961	Maximum speed limit (100 km/h)	
8.	961	Maximum speed limit (120 km/h)	
9.	991	No passing	
10.	1351	No passing for vehicles over 3.5 meteic tons	
11.	901	Right of way	
12.	1411	Priority road	
13.	1441	yield	

14.	541	stop	
15.	421	No vehicles	
16.	301	vehicles over 3.5 meteic tons prohibited	
17.	751	No entry	
18.	811	General caution	
19.	151	Dengerous curve to the left	
20.	241	Dengerous curve to the right	
21.	241	Double curve	
22.	271	Bumpy road	

23.	361	Slippery road	
24.	181	Road narrows on the right	
25.	1021	Road work	
26.	421	Traffic signals	
27.	181	pedestrians	
28.	361	Children crossing	
29.	181	Bicycles crossing	
30.	301	Beware of ice/snow	
31.	541	Wild animals crossing	
32.	181	End of all speed and passing limts	

33.	481	Turn right ahead	
34.	301	Turn left ahead	
35.	811	Ahead only	
36.	271	Go straight or right	
37.	151	Go straight or left	
38.	1381	Keep right	
39.	211	Keep left	
40.	241	Roundabout mandatory	
41.	181	End of no passing	
42.	181	End of no passing by vehicles over 3.5 meteic tons	

2.2. Material models

The classification architectures have been developed using the following as base models:

1) ResNet50
2) MobileNet
3) VGG16
4) VGG19
5) Xception Net
6) EfficientNetB0
7) DenseNet121

The architecture adopted is presented in the figure below:

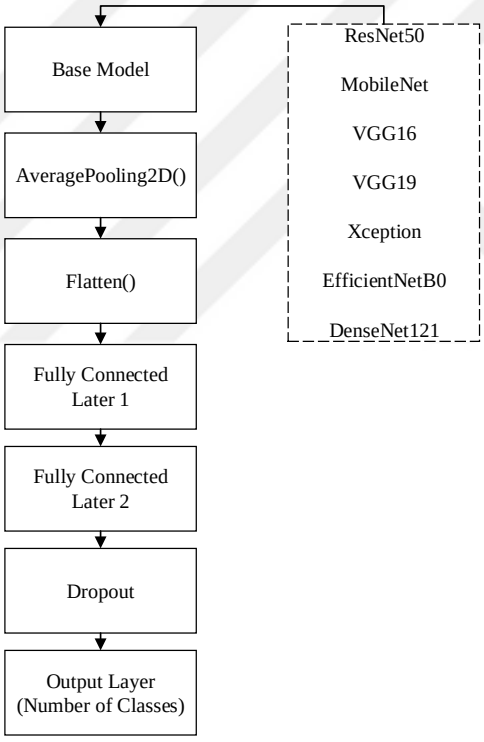


Figure 2.1. Architectural Framework for the Transfer Learning Models

2.3. Training Parameters

While the general architecture remains consistent, training parameters are configured based on the iterative approach where epochs on run time have been evaluated and determined to be deemed suitable. The training parameters have been provided in the table below:

Table 2.3. Training Parameters for the Models

MODEL / PARAMETERS	RESNET50	MOBILENET	VGG16	VGG19	XCEPTION	EFFICIENTNET B0	DENSENET121
LEARNING RATE	1e-4	1e-4	1e-5	1e-5	1e-5	1e-4	1e-4
OPTIMIZER	Adam	Adam	SGD	SGD	SGD	SGD	SGD
LOSS FUNCTION	MSE	MSE	MSE	MSE	MSE	MSE	MSE
EPOCHS	50	50	100	100	100	100	100
BATCH SIZE	32	32	32	32	32	32	32

2.4. Performance Parameters

The performance parameters for each of the above models are based on the following:

- Accuracy
- Precision
- F1-Score
- Recall

These models offer a complete picture of how well the model has been trained and found to fit the testing dataset. Additionally, a confusion matrix has been generated against the outcome of each model.

2.4.1. Models and Methods

As the recognition model was repeatedly trained by different CNN learners, the accuracy at the end of the training, the amount of processing required, and the training time were monitored. The dataset used was used many times. The learning models used in the study, whose results are compared, are ResNet50, DenseNet121; VGG16, VGG19; MobileNet, Xception, and EfficientNetB0.

The system design was divided into two main phases. We used several deep-learning models in the first phase and tested their accuracy. These models included DenseNet121, VGG16, ResNet50, MobileNet, VGG19, Xception, and EfficientNetB0. In the second

phase, we will compare these models after applying them to the GTSRB Figure 2.2.

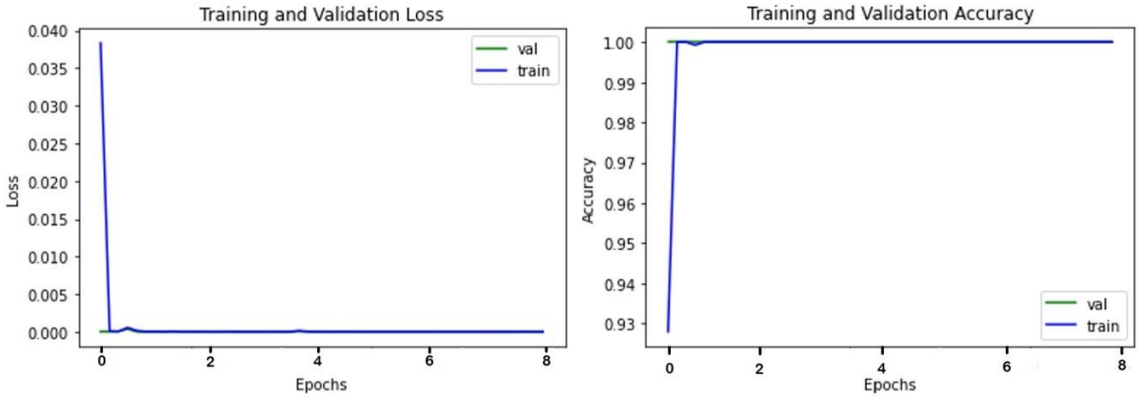


Figure 2.2. Train and validation accuracy of the model with 8 epochs and loss values of the model.

2.4.2. Performance Measures

The performance measures, such as accuracy, precision, recall, and F1 score, were used to evaluate the performance of each model. These measures are based on True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN). The mathematical equations for each of these models are listed below [29]:

$$Accuracy = \frac{TN + TP}{TN + FN + TP + FP}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F1 = 2\frac{(precision)(recall)}{precision + recall}$$

All of these metrics are further merged and provided in the form of a confusion matrix.

CHAPTER 3

RESULTS AND DISCUSSION

In this chapter, we have presented the results for the results obtained across each of these models.

3.1. Model Selection

The parameters already established in the previous chapter were used to select the model. The experiment metrics are accuracy, precision, recall, and F1 score. Depending on the data set, the parameters may vary in their results, so the parameter of interest is determined based on the specific applications. In this case, all four parameters have been found to provide the same value (due to the conditions of the dataset), and therefore the selection reference is given the same weighting. The following table shows the corresponding values of all parameters collected.

The results against both datasets have been presented below:

3.2. Training and Validation Performance

The training and validation performance of each of the models against the two datasets have been presented below:

3.2.1. ResNet50

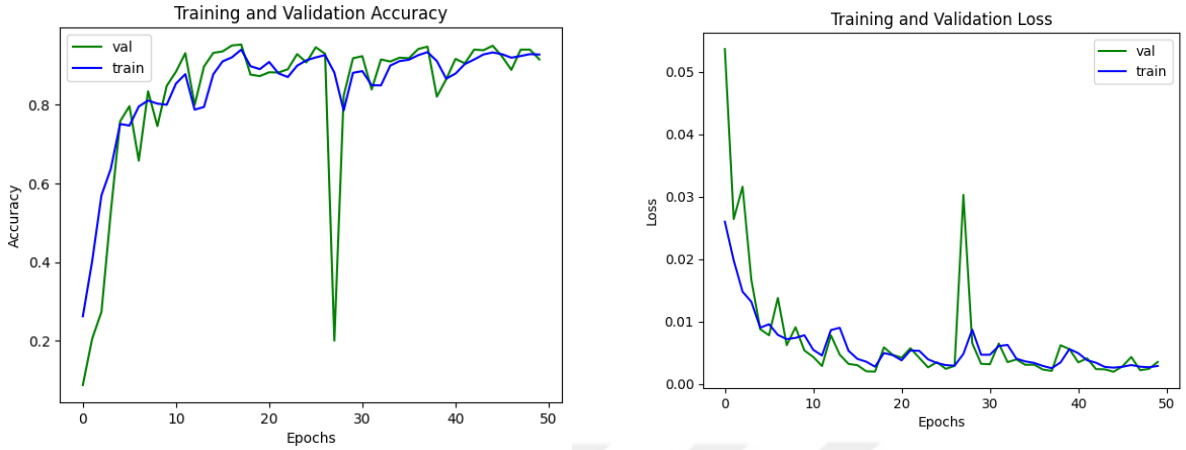


Figure 3.1. Training accuracy and loss for ResNet50 model on Turkish dataset.

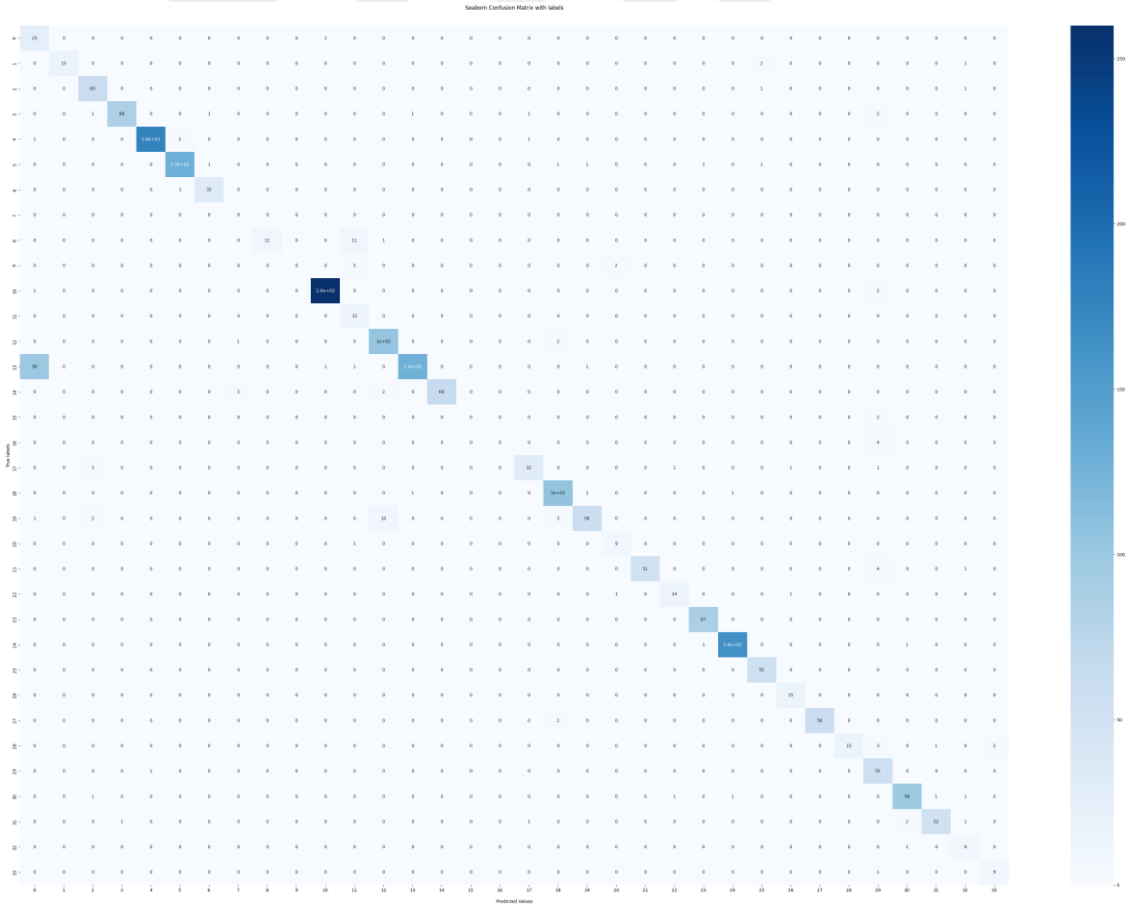


Figure 3.2. Confusion matrix of ResNet50 model for Turkish dataset.

Table 3.1. Metric values of ResNet50 - Self-Data.

Epoch	Train Loss	Train Accuracy	Valid Loss	Valid Accuracy
1	0.0260	0.2622	0.0537	0.0877
2	0.0164	0.5135	0.0280	0.1398
3	0.0144	0.5752	0.0292	0.4059
4	0.0132	0.6059	0.0113	0.6674
5	0.0108	0.6819	0.0104	0.7007
10	0.0070	0.8386	0.0082	0.8186
20	0.0046	0.8899	0.0047	0.8723
50	0.0029	0.9270	0.00091712	0.0035

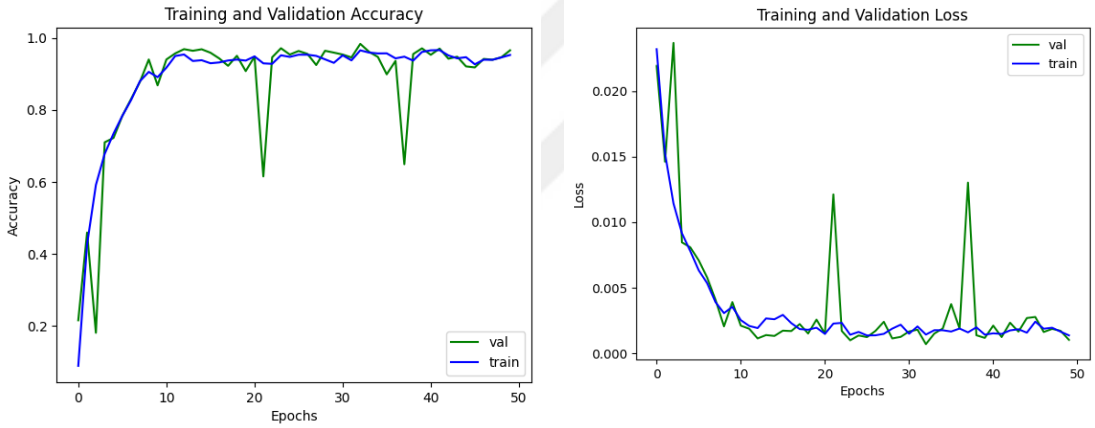


Figure 3.3. Training Accuracy and Loss Figure for ResNet50-GTSRB Dataset

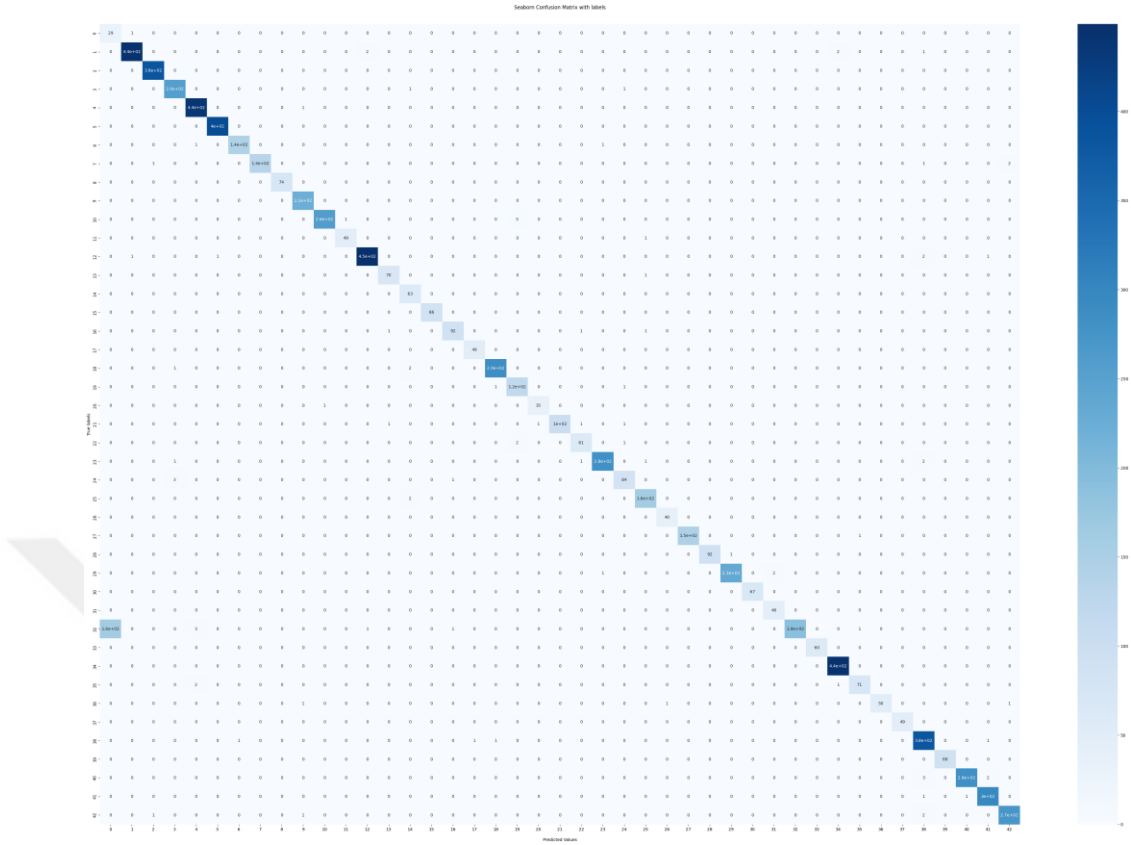


Figure 3.4. confusion matrix for ResNet50- GTSRB Dataset .

Table 3.2. Values of ResNet50 – GTSRB Dataset.

Epoch	Train Loss	Train Accuracy	Valid Loss	Valid Accuracy
1	0.0232	0.0893	0.0219	0.2158
2	0.0156	0.3968	0.0171	0.4310
3	0.0096	0.6640	0.0080	0.7169
4	0.0064	0.7777	0.0055	0.8173
5	0.0061	0.7901	0.0057	0.8091
10	0.0051	0.8378	0.0045	0.8664
20	0.0019	0.9367	0.0026	0.9076
50	0.0014	0.9524	0.0010	0.9653

3.2.2. MobileNet

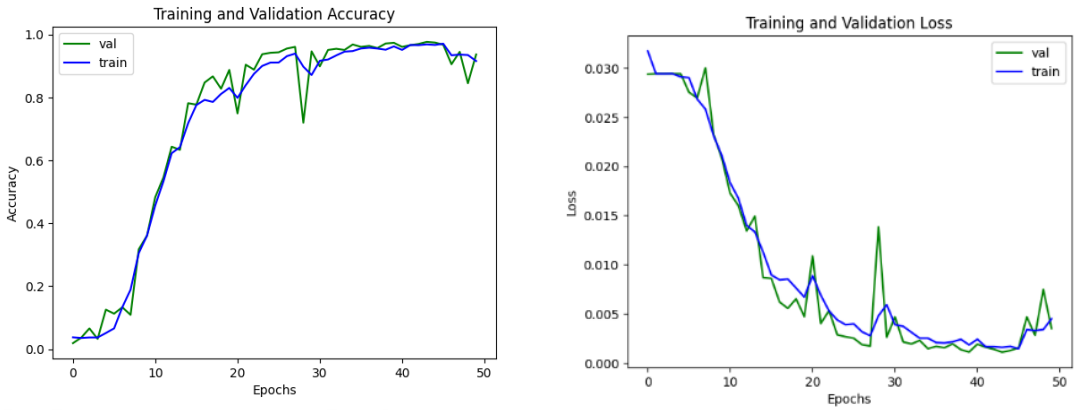


Figure 3.5. Training Accuracy and Loss Figure for MobileNet-Self-Data.

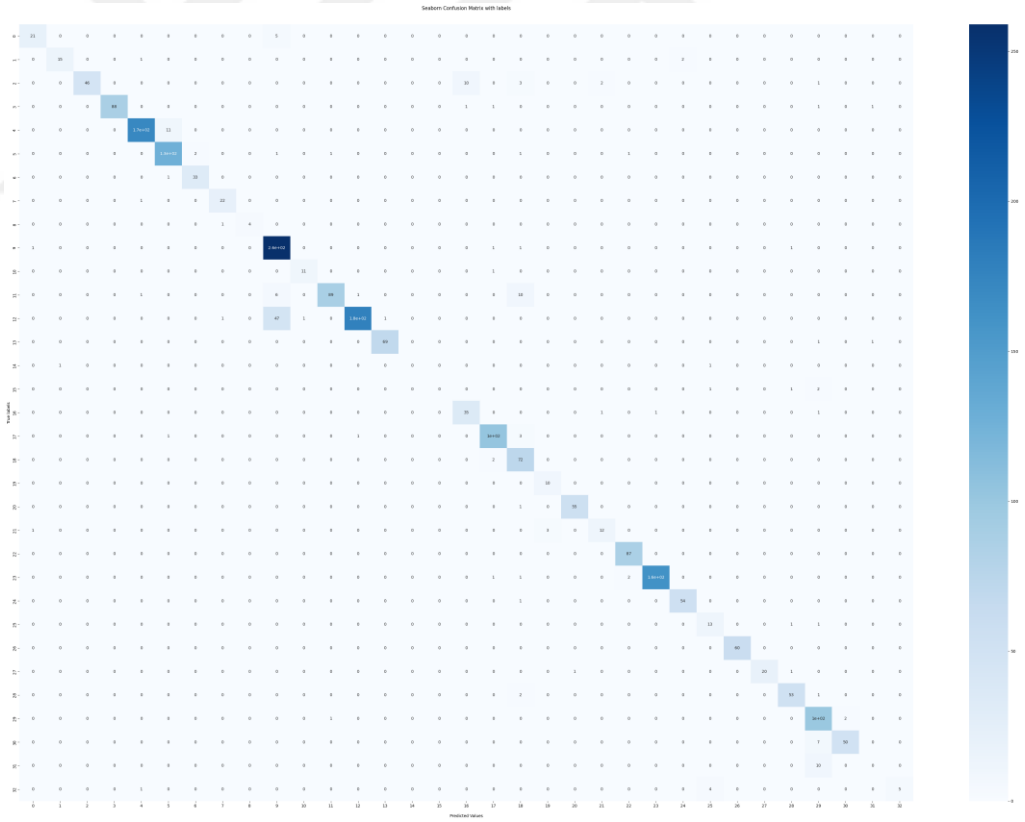


Figure 3.6. confusion matrix for MobileNet- Self-Data.

Table 3.3. Values of MobileNet for Self-Data.

Epoch	Train Loss	Train Accuracy	Valid Loss	Valid Accuracy
1	0.0317	0.0374	0.0294	0.0189
2	0.0258	0.1959	0.0269	0.1232
3	0.0212	0.3956	0.0190	0.5132
4	0.0145	0.6084	0.0163	0.5941
5	0.0126	0.6740	0.0104	0.7173
10	0.0078	0.7871	0.0074	0.7846
20	0.0067	0.8302	0.0047	0.8874
50	0.0045	0.9157	0.0035	0.9365

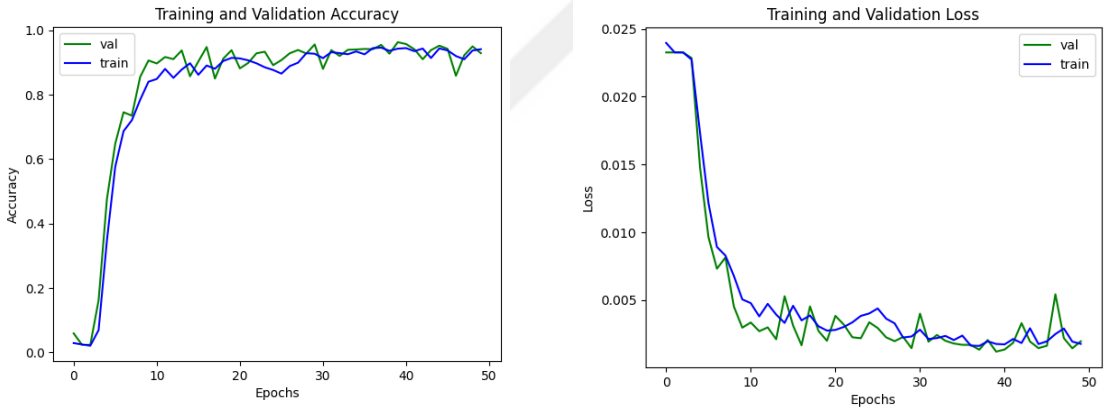


Figure 3.7. Training Accuracy and Loss Figure for MobileNet- GTSRB Dataset.

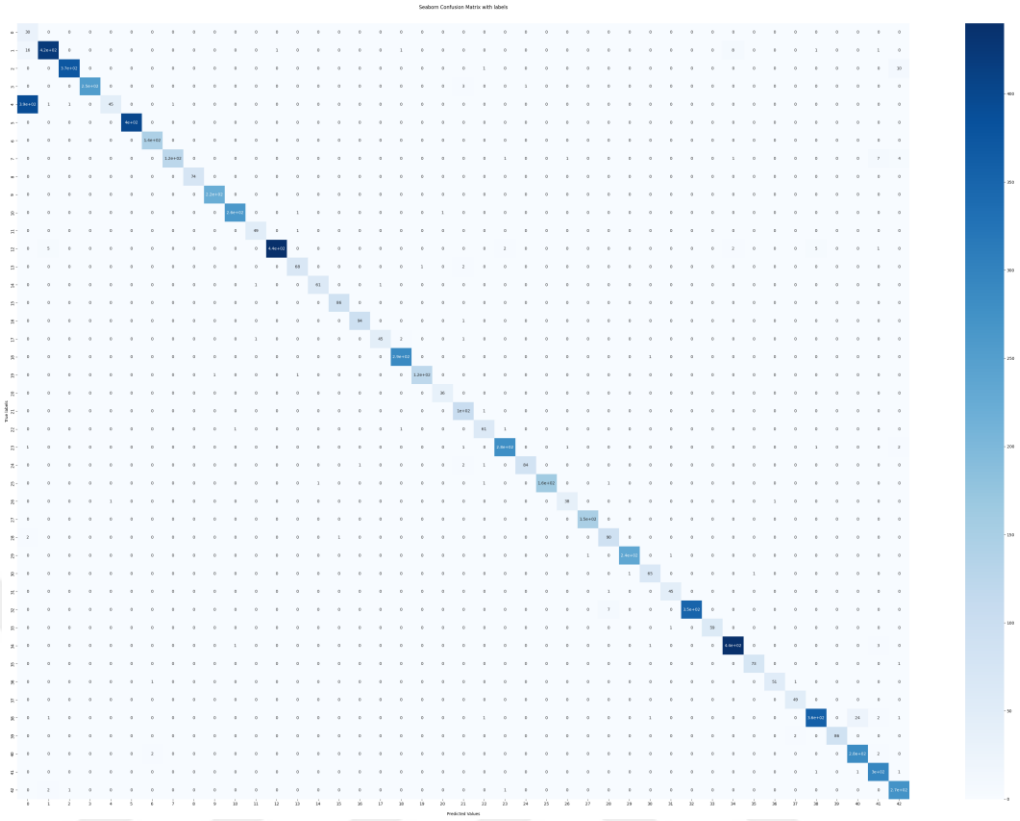


Figure 3.8. confusion matrix for MobileNet- GTSRB.

Table 3.4. Values of MobileNet for GTSRB Dataset.

Epoch	Train Loss	Train Accuracy	Valid Loss	Valid Accuracy
1	0.0240	0.0289	0.0233	0.0590
2	0.0233	0.0248	0.0233	0.0383
3	0.0233	0.0253	0.0233	0.0210
4	0.0233	0.0241	0.0233	0.0551
5	0.0233	0.0261	0.0233	0.0226
10	0.0233	0.0198	0.0233	0.0255
20	0.0027	0.9141	0.0020	0.9378
50	0.0018	0.9409	0.0020	0.9286

3.2.3. VGG16

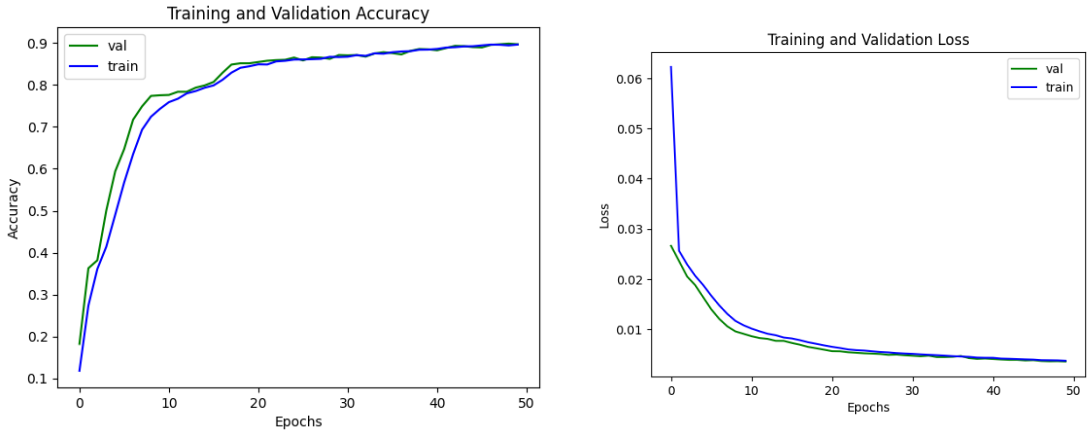


Figure 3.9. Training Accuracy and Loss Figure for VGG16-Self-Data.

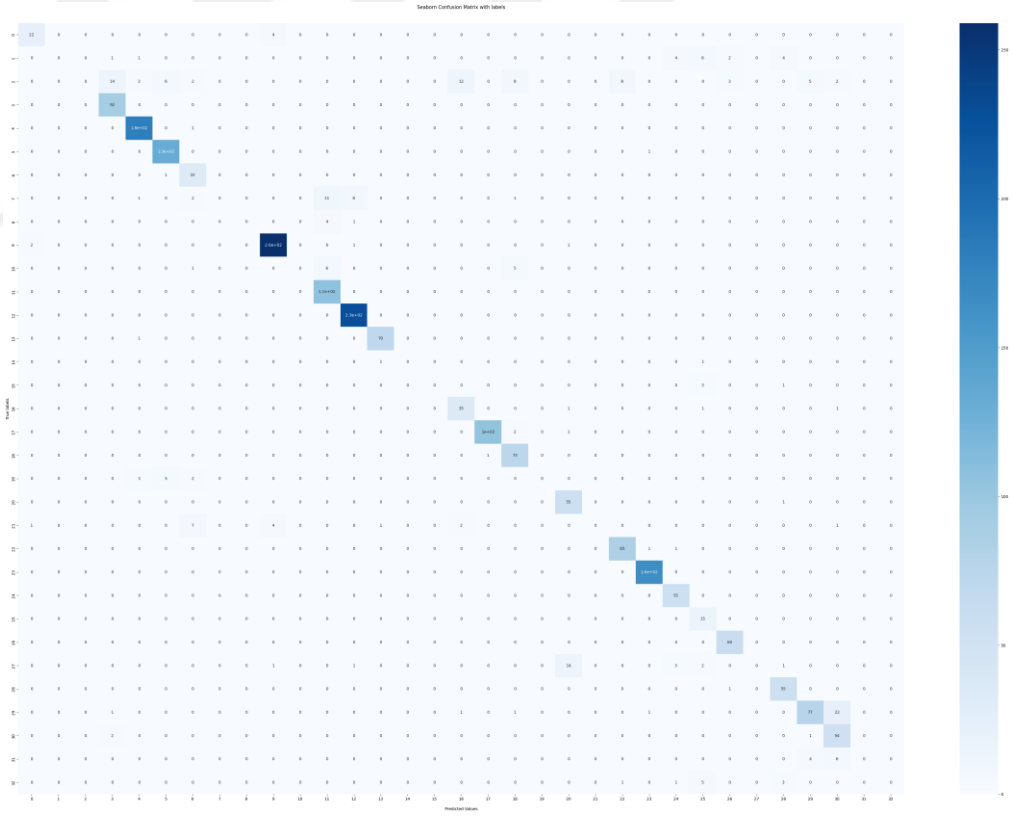


Figure 3.10. Confusion matrix for Self-Data

Table 3.5. Values of VGG16 for Self-Data.

Epoch	Train Loss	Train Accuracy	Valid Loss	Valid Accuracy
1	0.0623	0.1181	0.0266	0.1822
2	0.0255	0.2757	0.0226	0.3628
3	0.0225	0.3706	0.0203	0.3893
4	0.0206	0.4258	0.0187	0.4649
5	0.0190	0.4686	0.0173	0.5246
10	0.0125	0.6841	0.0108	0.7098
20	0.0071	0.8411	0.0062	0.8519
50	0.0037	0.8965	0.0036	0.8972

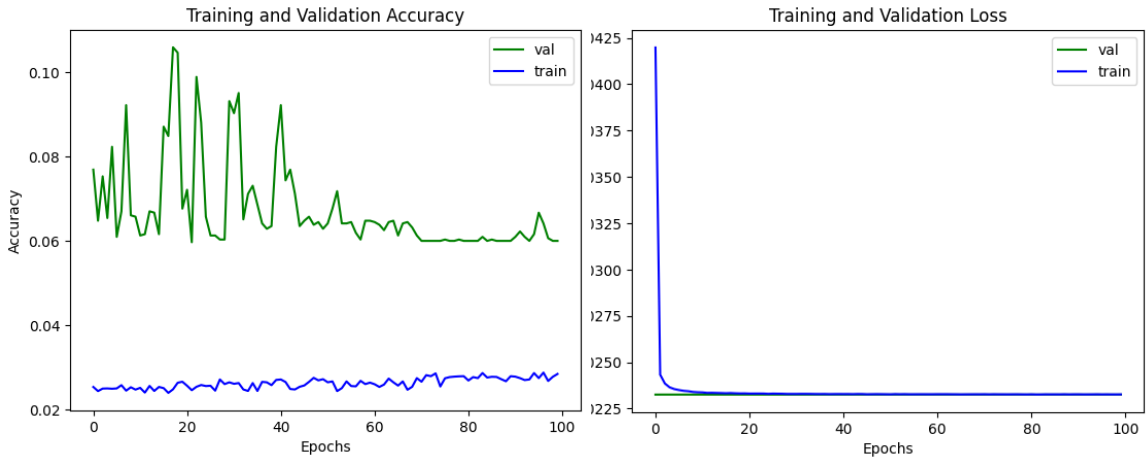


Figure 3.11. Training Accuracy and Loss Figure for VGG16 GTSRB Dataset.

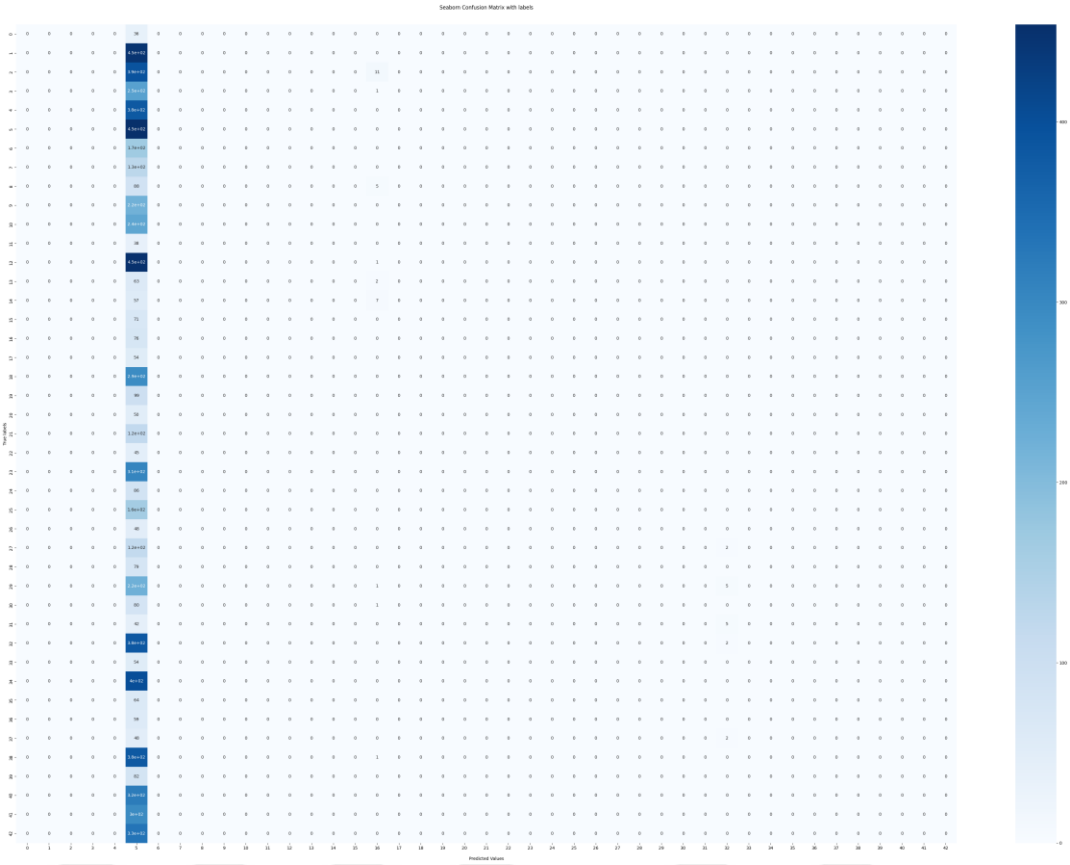


Figure 3.12. Confusion matrix for GTSRB.

Table 3.6. Values of VGG16 for GTSRB Dataset.

Epoch	Train Loss	Train Accuracy	Valid Loss	Valid Accuracy
1	0.0420	0.0253	0.0233	0.0768
2	0.0242	0.0282	0.0233	0.0319
3	0.0238	0.0308	0.0233	0.0242
4	0.0236	0.0269	0.0233	0.0338
5	0.0235	0.0282	0.0233	0.0360
10	0.0234	0.0298	0.0233	0.0166
20	0.0233	0.0265	0.0233	0.0676
50	0.0233	0.0271	0.0233	0.0628
100	0.0233	0.0284	0.0233	0.0599

3.2.4. VGG19

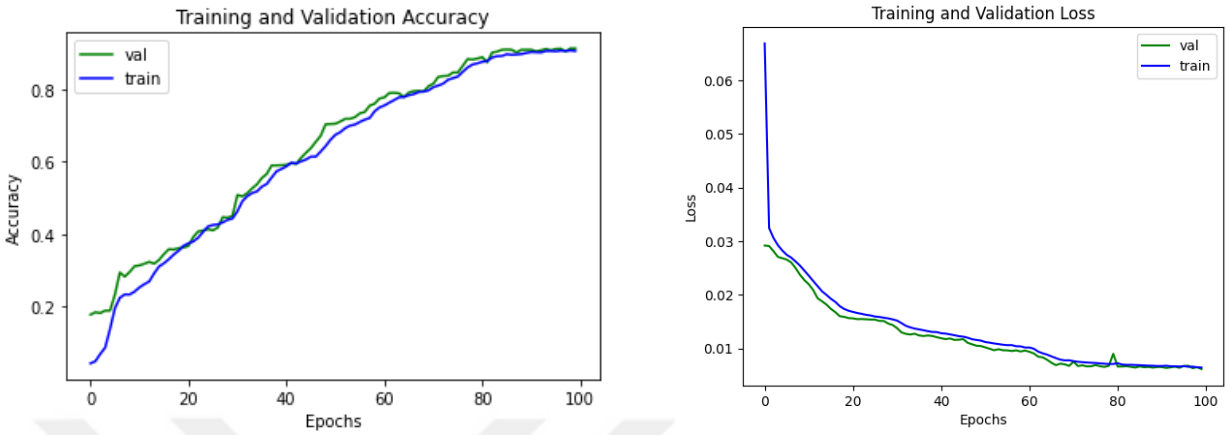


Figure 3.13. Training Accuracy and Loss Figure for VGG19-Self-Data.

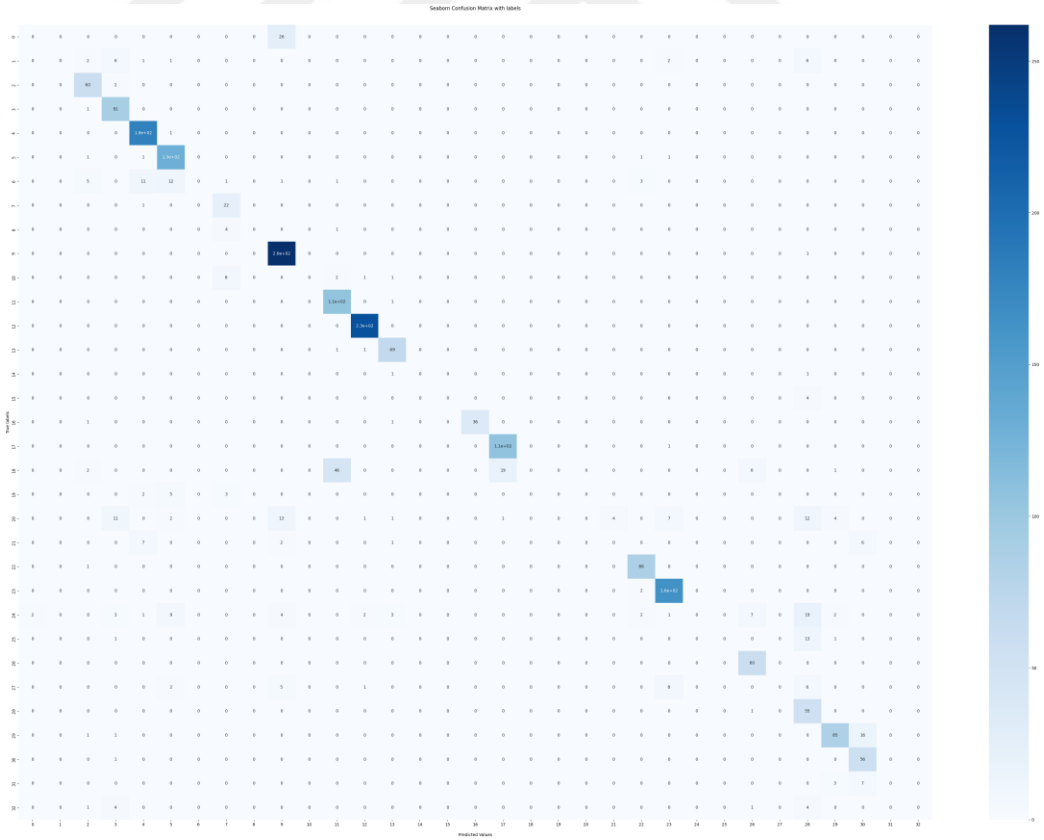


Figure 3.14. Confusion matrix for Self-Data.

Table 3.7. Values of VGG19 for Self-Data.

Epoch	Train Loss	Train Accuracy	Valid Loss	Valid Accuracy
1	0.0669	0.0471	0.0292	0.2162
2	0.0325	0.0499	0.0293	0.1852
3	0.0309	0.0700	0.0292	0.1829
4	0.0301	0.0875	0.0287	0.1897
5	0.0291	0.1389	0.0259	0.1897
10	0.0248	0.2420	0.0243	0.3122
20	0.0213	0.3684	0.0201	0.3636
50	0.0114	0.6855	0.0104	0.6984
100	0.0064	0.8211	0.0061	0.8171

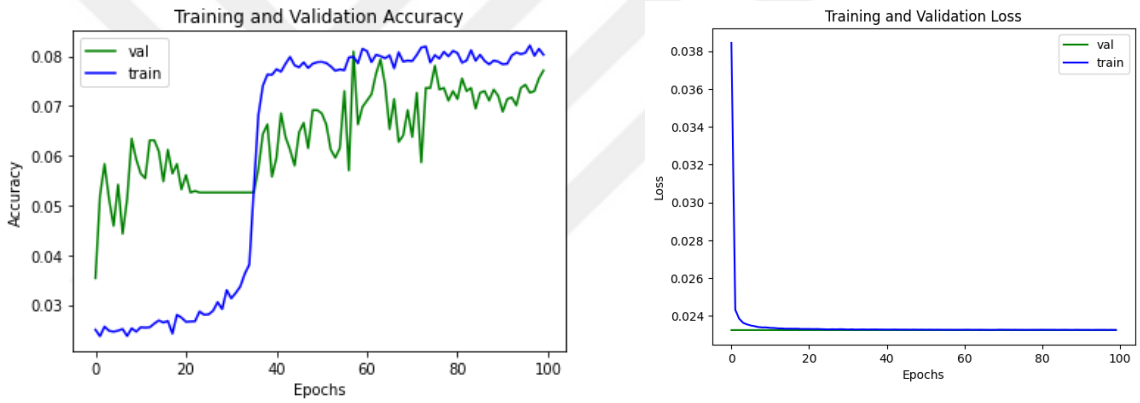


Figure 3.15. Training Accuracy and Loss Figure for VGG19-GTSRB Dataset.

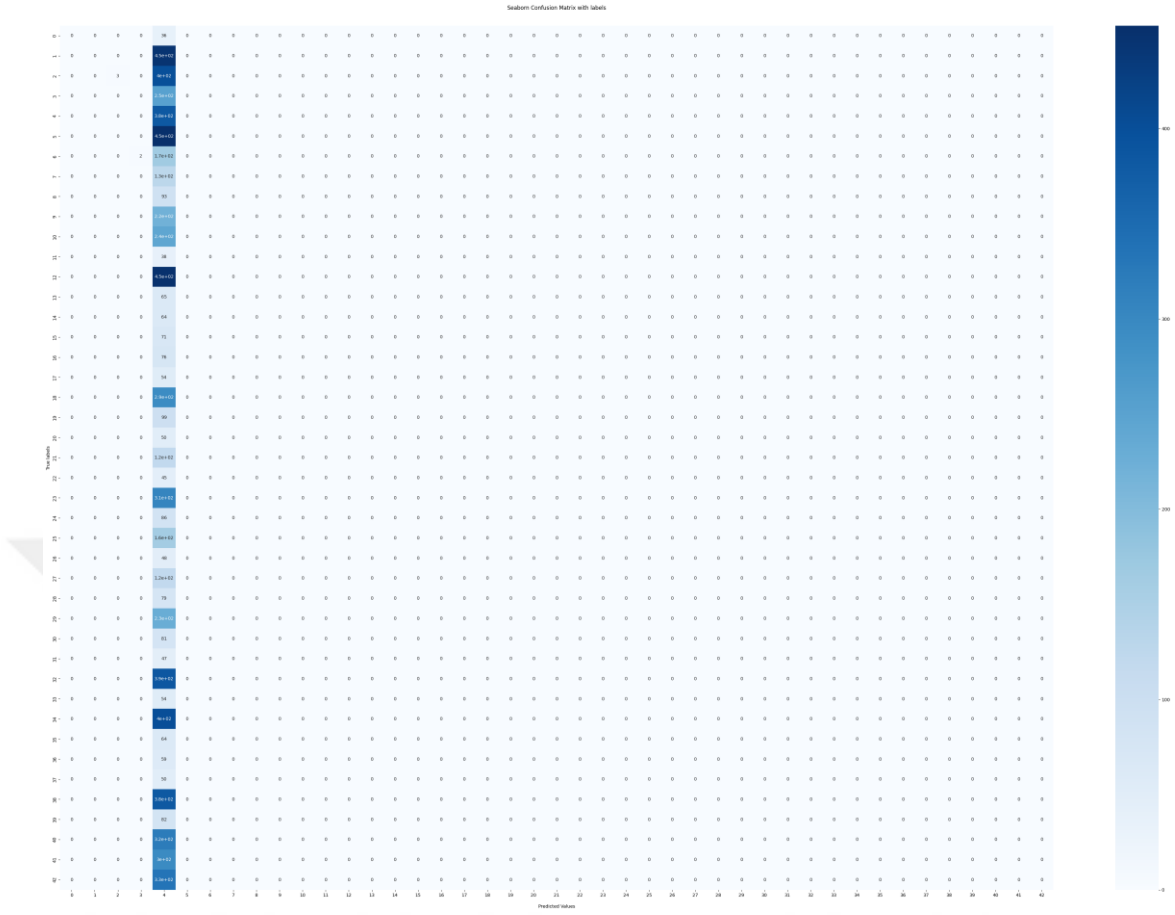


Figure 3.16. Confusion matrix for GTSRB Dataset.

Table 3.8 Values of VGG19 for GTSRB Dataset.

Epoch	Train Loss	Train Accuracy	Valid Loss	Valid Accuracy
1	0.0384	0.0276	0.0233	0.0462
2	0.0244	0.0246	0.0233	0.0437
3	0.0239	0.0260	0.0233	0.0577
4	0.0237	0.0266	0.0233	0.0631
5	0.0236	0.0258	0.0233	0.0778
10	0.0234	0.0268	0.0233	0.0638
20	0.0233	0.0282	0.0233	0.0704
50	0.0233	0.0267	0.0233	0.0571
100	0.0233	0.0273	0.0233	0.0577

3.2.5. EfficientNet-B0

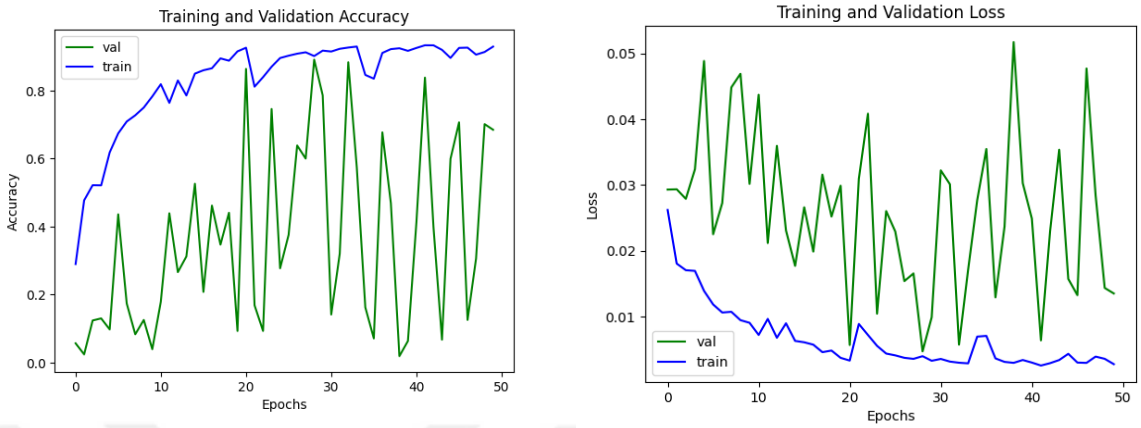


Figure 3.17. Training Accuracy and Loss Figure for EfficientNetB0- Self-Data.

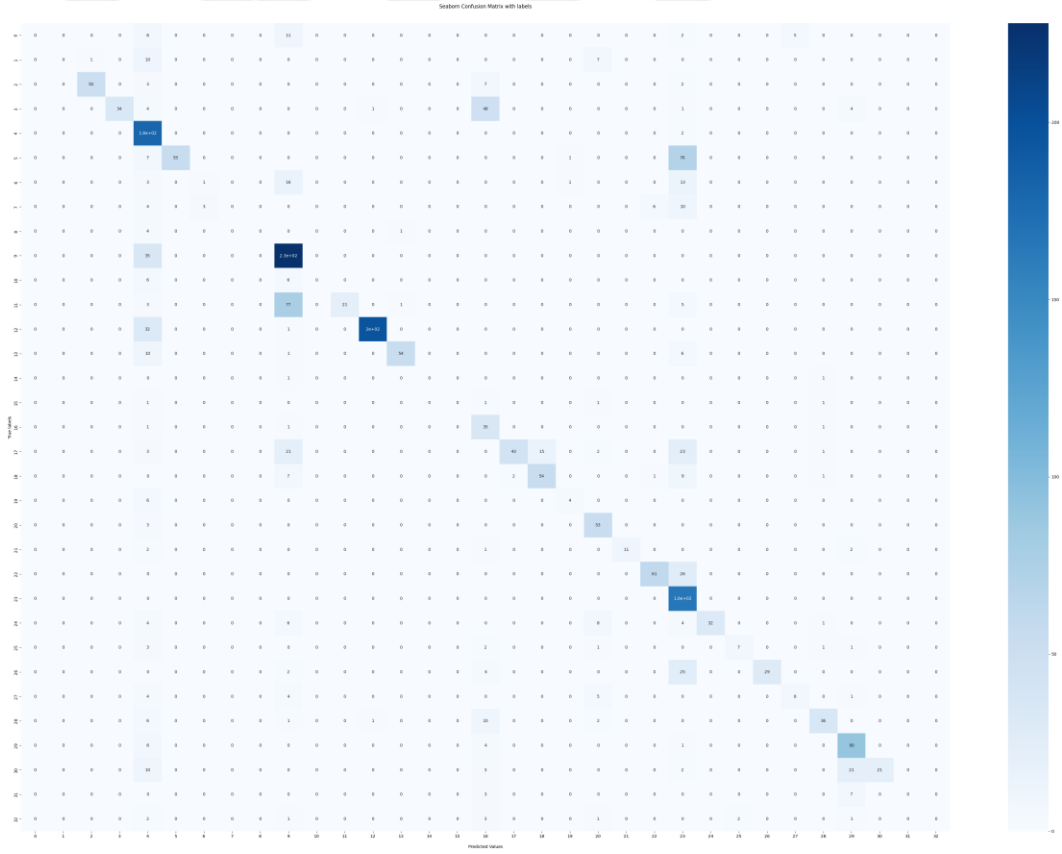


Figure 3.18. Confusion matrix for EfficientNet-B0 - Self-Data.

Table 3.9. Values of EfficientNet-B0 for Self-Data.

Epoch	Train Loss	Train Accuracy	Valid Loss	Valid Accuracy
1	0.0262	0.2898	0.0293	0.0567
2	0.0178	0.4854	0.0289	0.0658
3	0.0154	0.5534	0.0294	0.0877
4	0.0152	0.5597	0.0293	0.1141
5	0.0139	0.6009	0.0263	0.2124
10	0.0117	0.7005	0.0285	0.1156
20	0.0037	0.9154	0.0299	0.0930
50	0.0027	0.9296	0.0135	0.6848

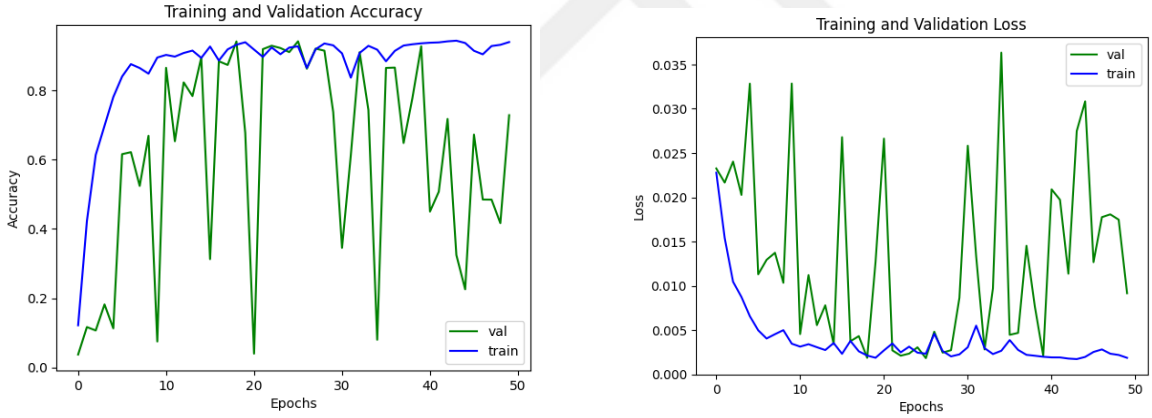


Figure 3.19. Training Accuracy and Loss Figure for EfficientNet-B0-GTSRB.

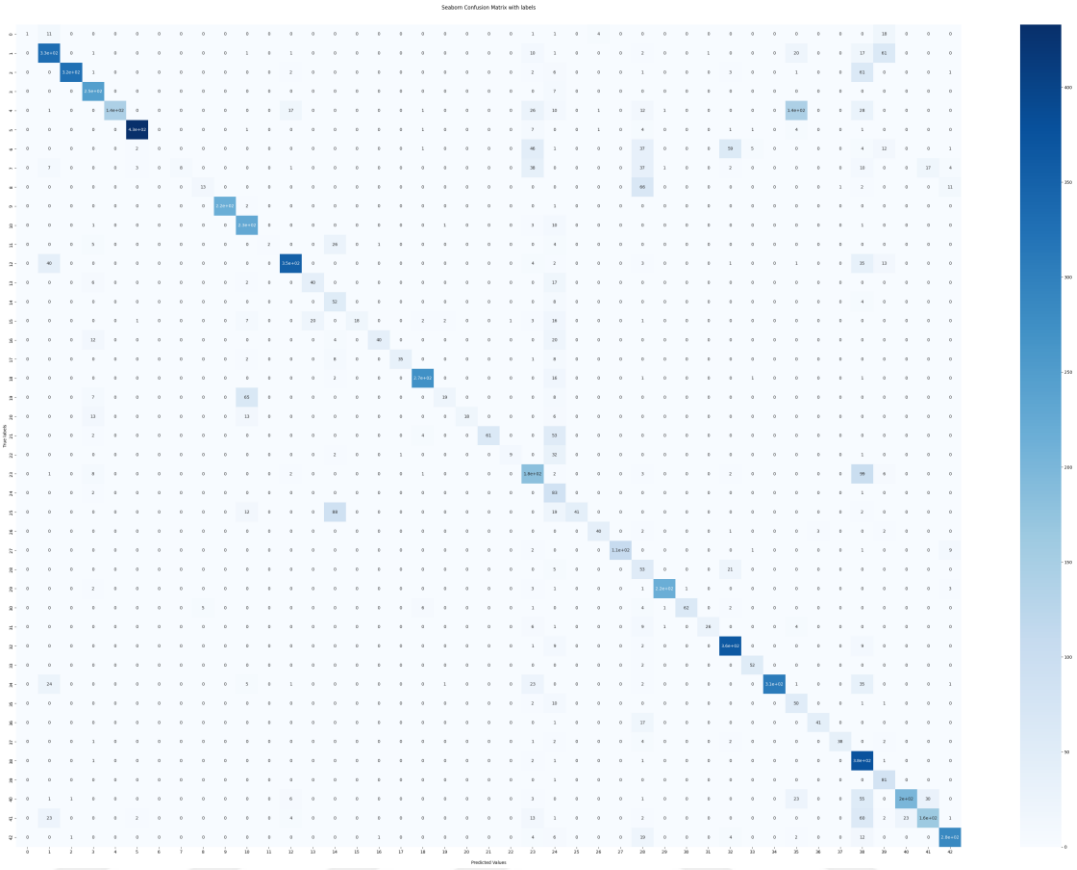


Figure 3.20. Confusion matrix for EfficientNet-B0 for GTSRB Dataset.

Table 3.10. Values of EfficientNet-B0 for GTSRB Dataset.

Epoch	Train Loss	Train Accuracy	Valid Loss	Valid Accuracy
1	0.0228	0.1217	0.0233	0.0373
2	0.0155	0.4220	0.0232	0.0124
3	0.0112	0.5673	0.0203	0.1801
4	0.0104	0.5944	0.0227	0.0673
5	0.0105	0.6078	0.0162	0.3835
10	0.0081	0.7225	0.0231	0.0128
20	0.0019	0.9390	0.0127	0.6771
50	0.0019	0.9395	0.0092	0.7284

3.2.6. DenseNet121

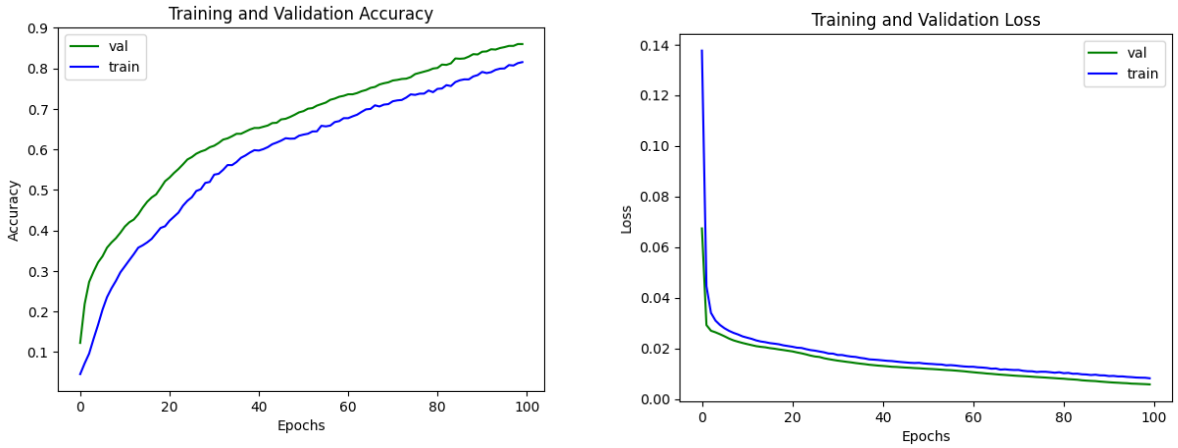


Figure 3.21. Training Accuracy and Loss Figure for DenseNet- Self-Data.

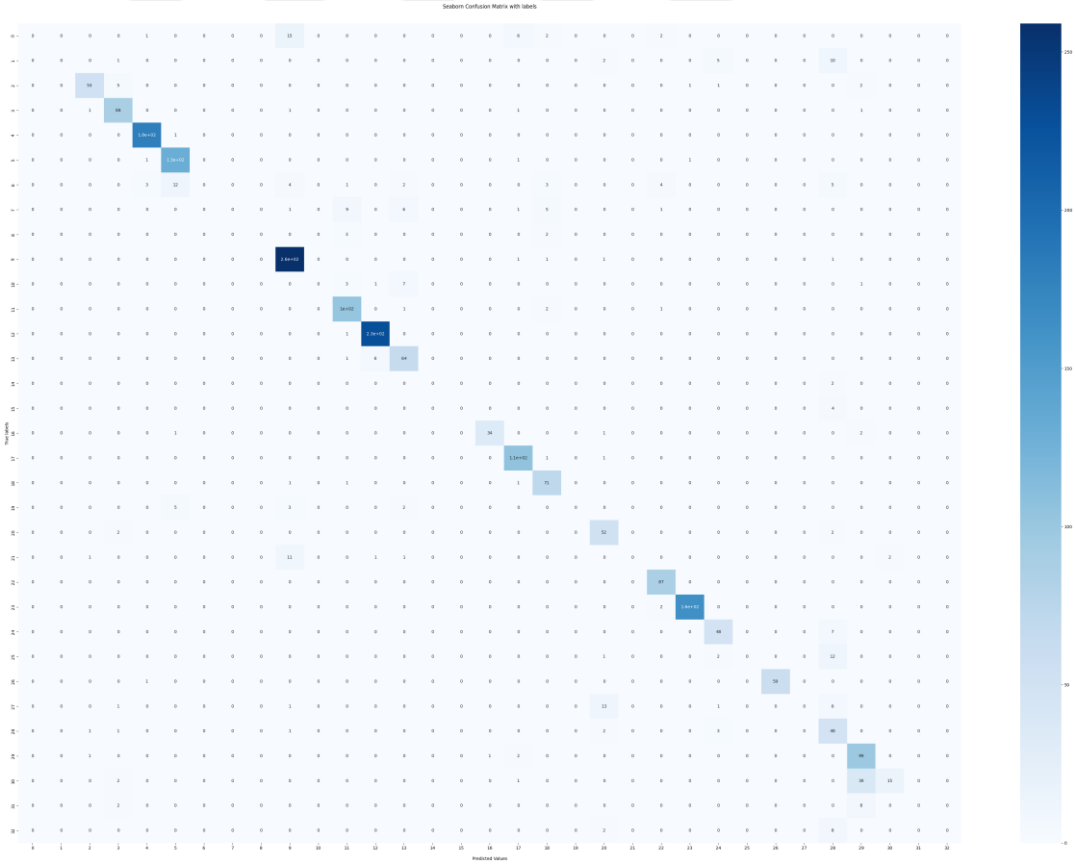


*Figure 3.22.*Confusion matrix for DenseNet121 for Self-Data.

Table 3.11. Values of DenseNet121 for Self-Data.

Epoch	Train Loss	Train Accuracy	Valid Loss	Valid Accuracy
1	0.1376	0.0454	0.0674	0.1224
2	0.0442	0.0516	0.0298	0.1308
3	0.0343	0.0714	0.0279	0.2275
4	0.0314	0.0915	0.0273	0.2986
5	0.0300	0.1213	0.0268	0.3190
10	0.0261	0.2590	0.0234	0.3643
20	0.0215	0.4065	0.0191	0.5004
50	0.0140	0.6339	0.0120	0.6916
100	0.0082	0.8156	0.0058	0.8602

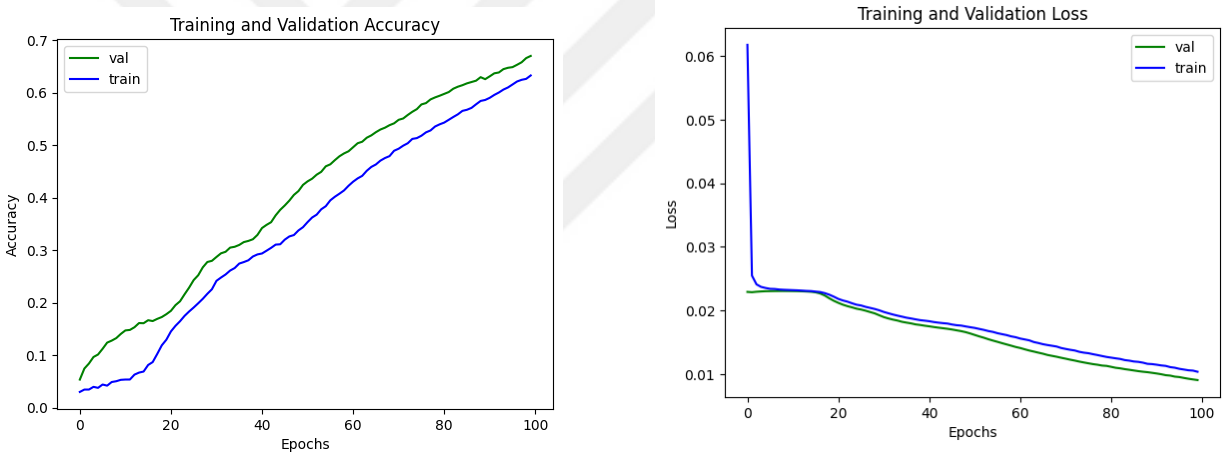


Figure 3.23. Training Accuracy and Loss Figure for DenseNet121for GTSRB Dataset.

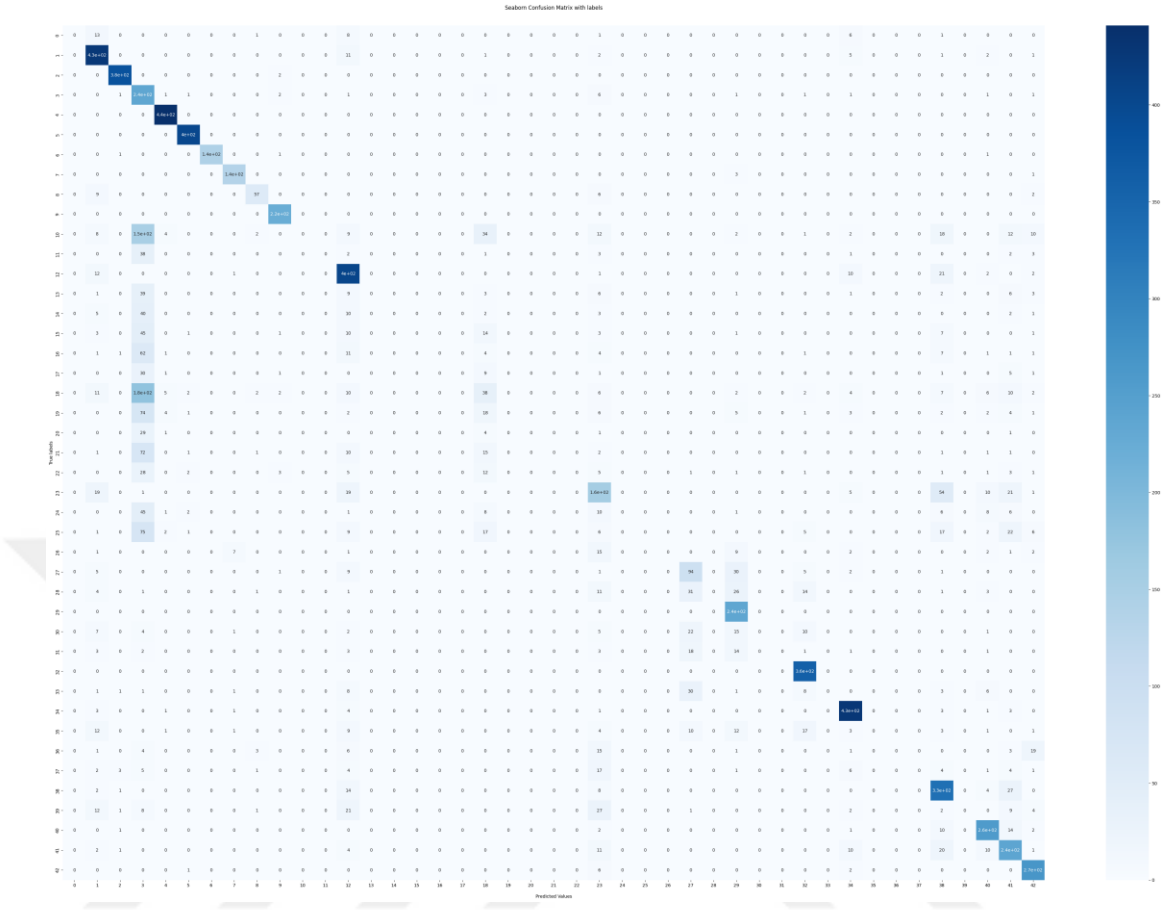


Figure 3.24. Confusion matrix for DenseNet121 for GTSRB Dataset.

Table 3.12. Values of DenseNet121 for GTSR Dataset.

Epoch	Train Loss	Train Accuracy	Valid Loss	Valid Accuracy
1	0.0618	0.0297	0.0229	0.0532
2	0.0255	0.0332	0.0229	0.0622
3	0.0241	0.0371	0.0229	0.0778
4	0.0237	0.0409	0.0230	0.0880
5	0.0235	0.0412	0.0230	0.0998
10	0.0232	0.0575	0.0230	0.1371
20	0.0216	0.1549	0.0211	0.2056
50	0.0174	0.3437	0.0164	0.4243
100	0.0104	0.6328	0.0091	0.6701

3.2.7. Xception

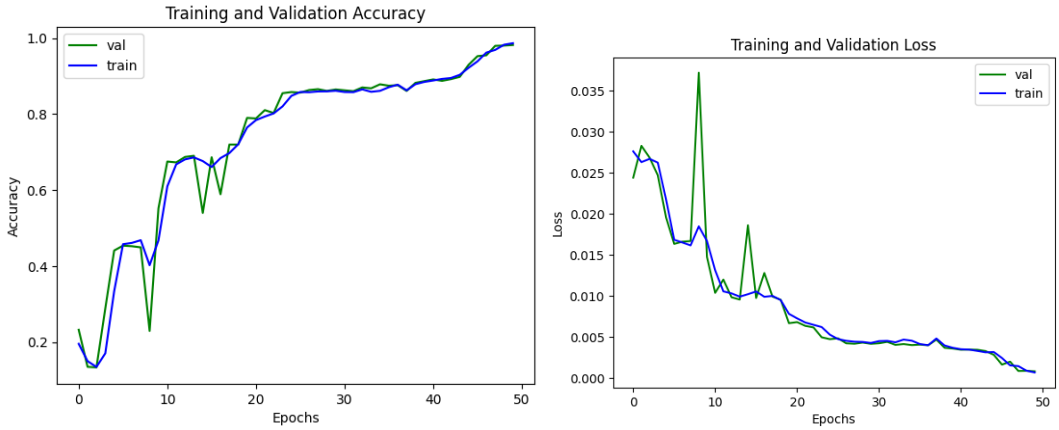


Figure 3.25. Training Accuracy and Loss Figure for Xception for Self-Data.

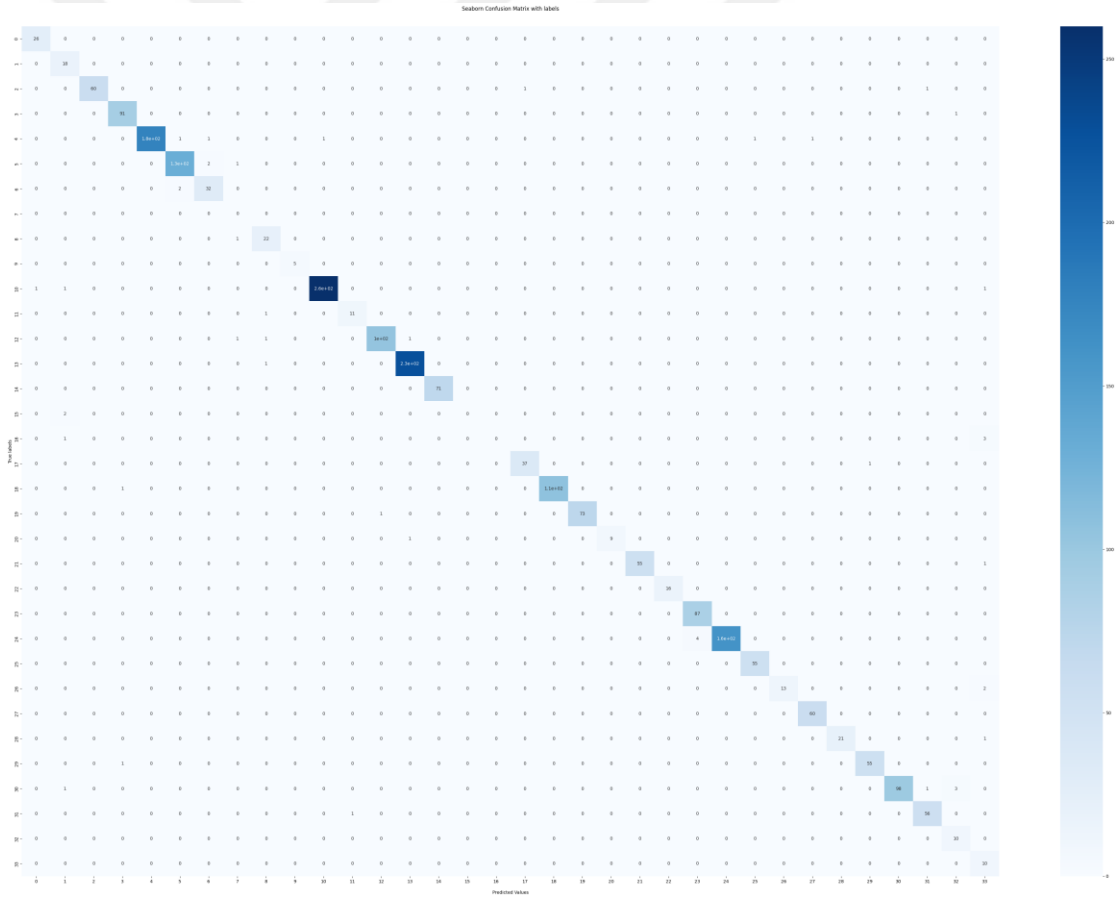


Figure 3.26. confusion matrix for Xception for Self-Data.

Table 3.13. Values of Xception for Self-Data.

Epoch	Train Loss	Train Accuracy	Valid Loss	Valid Accuracy
1	0.0276	0.1951	0.0244	0.2320
2	0.0263	0.1494	0.0283	0.1345
3	0.0267	0.1342	0.0269	0.1330
4	0.0262	0.1700	0.0247	0.2880
5	0.0218	0.3334	0.0196	0.4407
10	0.0167	0.4674	0.0148	0.5518
20	0.0078	0.7646	0.0067	0.7899
50	0.00065447	0.9865	0.00079618	0.9819

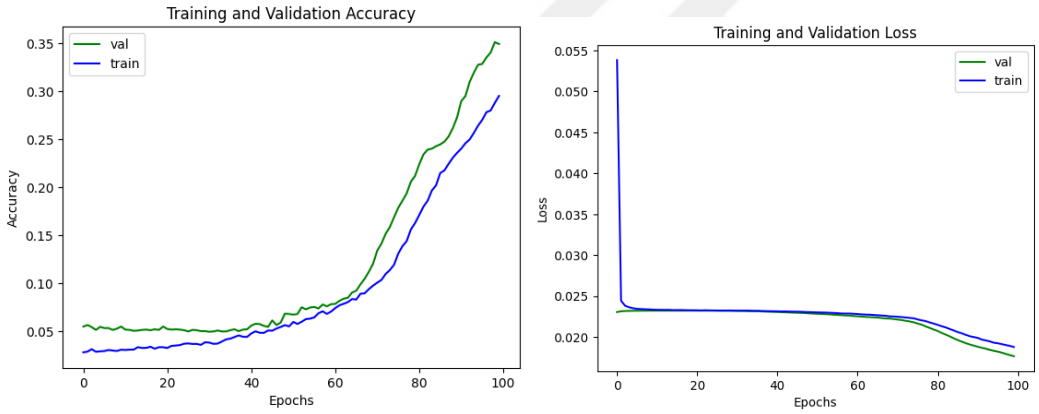


Figure 3.27. Training Accuracy and Loss Figure for Xception for GTARB Dataset.

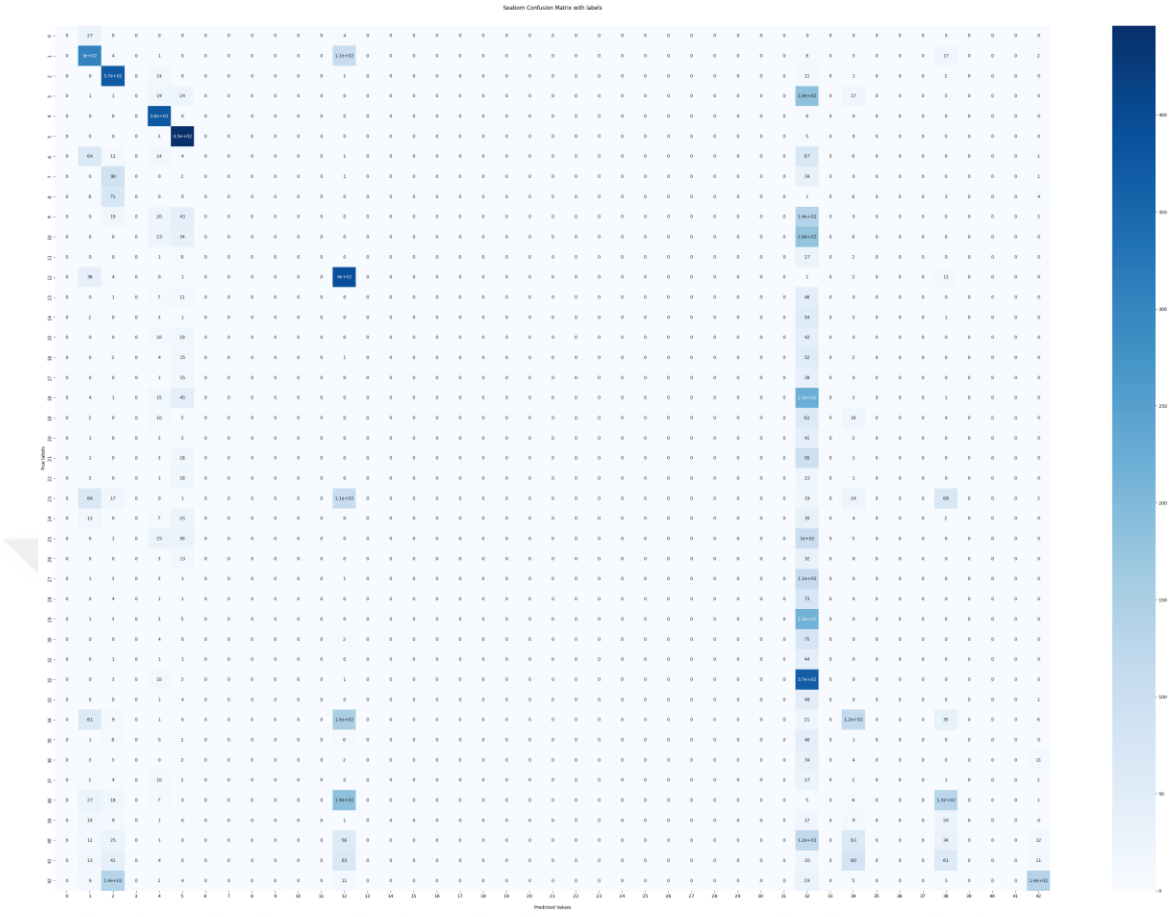


Figure 3.28. Confusion matrix for Xception for GTSRB Dataset.

Table 3.14. Values of Xception for GTSRB Dataset.

Epoch	Train Loss	Train Accuracy	Valid Loss	Valid Accuracy
1	0.0538	0.0280	0.0231	0.0548
2	0.0247	0.0278	0.0233	0.0657
3	0.0240	0.0292	0.0232	0.0759
4	0.0237	0.0290	0.0232	0.0778
5	0.0236	0.0283	0.0232	0.0784
10	0.0234	0.0318	0.0232	0.0771
20	0.0233	0.0333	0.0232	0.0768
50	0.0230	0.0550	0.0229	0.0679
100	0.0188	0.2950	0.0177	0.3491

3.3. Performance Metrics

Table 3.15. Comparison of self-dataset with the GTSRB dataset.

Model/ Performance	ResNet50		MobileNet		VGG16		VGG19		Xception		EfficientNet-B0		DenseNet121	
	D1	GTSRB	D1	GTSRB	D1	GTSRB	D1	GTSRB	D1	GTSRB	D1	GTSRB	D1	GTSRB
Precision	0.781	0.966	0.836	0.954	0.538	0.004	0.413	0.024	0.868	0.090	0.573	0.797	0.489	0.337
Recall	0.797	0.975	0.820	0.956	0.615	0.023	0.501	0.023	0.889	0.150	0.450	0.634	0.525	0.402
Accuracy	0.902	0.968	0.918	0.927	0.888	0.058	0.815	0.049	0.979	0.338	0.666	0.719	0.856	0.671
F1 Score	0.770	0.962	0.822	0.937	0.569	0.003	0.449	0.003	0.875	0.104	0.462	0.624	0.497	0.360

After testing all models under the same specific conditions and training them on 11,023 images distributed over 34 types of traffic lights for self-data, and in the same conditions, we tested all models using GTSRB dataset and trained them on more than 50,000 images. The Table 3.15 shows you the most prominent results.

.

CHAPTER 4

CONCLUSION

Based on training two datasets, it was concluded that most models generally fit both categories of datasets well. In some cases, performance against one dataset can outperform another. However, the differences are very significant.

ResNet50, MobileNet, and Xception were found to perform better for self-data. As for GTSRB dataset, ResNet50, MobileNet, and EfficientNet-B0 got the best results. The reason can be attributed to the type of features being extracted and the resulting labels. Thus, these three models can be used safely to achieve a higher degree of performance.

Traffic sign recognition is a challenging task that has occupied many computer vision communities for years. Road hazards mainly occur when drivers do not pay attention to traffic lights. In this study, a new automated detection and recognition of traffic signs using deep learning methods is proposed. Various deep learning models such as ResNet50, DenseNet121, VGG16, VGG19, MobileNet, Xception, and EfficientNet-B0, are used for transfer learning.

The trained models work effectively in recognizing traffic signs in day and night images. In our future work, we intend to develop new object recognition models for traffic lights. It can be combined with the autonomous driving algorithm to track multiple traffic lights in different video sequences. The major contribution of this research lies in its practicality. Given the continuous evolution and adoption of autonomous driving technology, efficient and accurate traffic sign recognition is paramount. The insights from this study could be instrumental in the development of more sophisticated autonomous driving systems. Furthermore, the application of the proposed methods extends beyond autonomous vehicles and can be utilized in other areas requiring image recognition.

This research serves as a substantial foundation for further advancements in traffic sign recognition systems. It underscores the importance of careful model selection, meticulous comparison conditions, and the vast potential of deep learning models in improving the safety and efficiency of autonomous driving systems. The road ahead is promising, with numerous avenues for further exploration and innovation.

Future Work

As for future research directions, the potential for advancement is vast. Firstly, the development of new object recognition models specifically for traffic lights would be a beneficial extension of this work. By integrating these models with autonomous driving algorithms, it could be possible to track multiple traffic lights in different video sequences, potentially enhancing the safety and efficiency of autonomous vehicles.

Additionally, future work could also focus on refining and improving existing models. Given the variability in performance among different models on different datasets, there is room for optimization to achieve greater consistency and overall performance. For example, this could be achieved by further exploring hyperparameters or feature extraction techniques.

Moreover, while the current research utilized two specific datasets, future research could expand to include a broader range of datasets. This would allow for a more comprehensive understanding of how different models perform under varied conditions. For instance, the inclusion of datasets with signs in different languages or from different regions would significantly contribute to the robustness of the models.

Finally, future research could also look at ways of integrating the traffic sign recognition system with other aspects of autonomous driving, such as path planning, obstacle detection, and collision avoidance. This holistic approach could lead to the development of an all-encompassing autonomous driving system, effectively pushing the boundaries of current technology.

REFERENCES

[1] Cheon, S., 2003. An overview of automated highway systems (AHS) and the social and institutional challenges they face, pp.4-16.

[2] Elhenawy, M., Elbery, A.A., Hassan, A.A. and Rakha, H.A., 2015, September. An intersection game-theory-based traffic control algorithm in a connected vehicle environment. *In 2015 IEEE 18th international conference on intelligent transportation systems* (pp. 343-347). IEEE.

[3] Lu, G., Li, L., Wang, Y., Zhang, R., Bao, Z. and Chen, H., 2014. October. A rule based control algorithm of connected vehicles in uncontrolled intersection. *In 17th international IEEE conference on intelligent transportation systems (itsc)* (pp. 115-120). IEEE.

[4] Al Hakim, E., 2018. 3d yolo: End-to-end 3d object detection using point clouds, pp.1-58.

[5] Al-Sultan, S. and Moath, M., Al-Doori, Ali H. Al-Bayatti and Hussien Zedan 2005. A comprehensive survey on vehicular Ad Hoc network. **Journal of Network and Computer Applications, 37**.

[6] Boesch, P.M., Ciari, F. and Axhausen, K.W., 2016. Autonomous vehicle fleet sizes required to serve different levels of demand. **Transportation Research Record, 2542**(1), pp.111-119.

[7] Wadud, Z., MacKenzie, D. and Leiby, P., 2016. Help or hindrance? The travel, energy and carbon impacts of highly automated vehicles. **Transportation Research Part A: Policy and Practice, 86**, pp.1-18.

[8] Pendleton, S.D., Andersen, H., Du, X., Shen, X., Meghjani, M., Eng, Y.H., Rus, D. and Ang Jr, M.H., 2017. Perception, planning, control, and coordination for autonomous vehicles. **Machines, 5**(1), p.6.

[9] Ulbrich, S., Reschka, A., Rieken, J., Ernst, S., Bagschik, G., Dierkes, F., Nolte, M. and Maurer, M., 2017. Towards a functional system architecture for automated vehicles. **arXiv preprint arXiv:1703**.08557.

[10] Wang, J., Huang, H., Li, K. and Li, J., 2021. Towards the unified principles for level 5 autonomous vehicles. **Engineering, 7**(9), pp.1313-1325.

[11] Ulbrich, S., Reschka, A., Rieken, J., Ernst, S., Bagschik, G., Dierkes, F., Nolte, M. and Maurer, M., 2017. Towards a functional system architecture for automated vehicles. **arXiv preprint arXiv:1703**.08557.

[12] Fridman, L., 2018. Human-centered autonomous vehicle systems: Principles of effective shared autonomy. **arXiv preprint arXiv:1810**.01835.

[13] Pillath, S., 2016. Automated vehicles in the EU.

[14] World Health Organization, 2018. Global status report on road safety 2018: Summary (No. WHO/NMH/NVI/18.20). World Health Organization.

[15] Tan, H., Zhao, F., Hao, H. and Liu, Z., 2020. Cost analysis of road traffic crashes in China. **International journal of injury control and safety promotion, 27**(3), pp.385-391.

[16] DfT, U.K., 2011. Reported road casualties in Great Britain: 2019 annual report.

[17] Tan, H., Zhao, F. and Liu, Z., 2021. Impact of adaptive cruise control (ACC) system on fatality and injury reduction in China. **Traffic injury prevention, 22**(4), pp.307-312.

[18] National Highway Traffic Safety Administration, 2020. Early Estimates of Motor Vehicle Traffic Fatalities and Fatality Rate by Sub-Categories Through June 2020.

[19] Dresner, K. and Stone, P., 2004, July. Multiagent traffic management: A reservation-based intersection control mechanism. In Autonomous Agents and Multiagent Systems**, International Joint Conference on** (Vol. 3, pp. 530-537). IEEE Computer Society.

[20] Dresner, K.M. and Stone, P., 2007, January. Sharing the road: Autonomous vehicles meet human drivers. In Ijcai (Vol. 7, pp. 1263-1268).

[21] Fajardo, D., Au, T.C., Waller, S.T., Stone, P. and Yang, D., 2011. Automated intersection control: Performance of future innovation versus current traffic signal control. **Transportation Research Record, 2259**(1), pp.223-232.

[22] Bagloee, S.A., Tavana, M., Asadi, M. and Oliver, T., 2016. Autonomous vehicles: challenges, opportunities, and future implications for transportation policies. **J Mod Transp 24**: 284–303.

[23] Luettel, T., Himmelsbach, M. and Wuensche, H.J., 2012. Autonomous ground vehicles—Concepts and a path to the future**. Proceedings of the IEEE, 100**(Special Centennial Issue), pp.1831-1839.

[24] Levinson, J., Askeland, J., Becker, J., Dolson, J., Held, D., Kammel, S., Kolter, J.Z., Langer, D., Pink, O., Pratt, V. and Sokolsky, M., 2011, June. Towards fully autonomous driving: Systems and algorithms. *In 2011 IEEE intelligent vehicles symposium (IV)* (pp. 163-168). IEEE.

[25] Buch, N., Velastin, S.A. and Orwell, J., 2011. A review of computer vision techniques for the analysis of urban traffic. IEEE Transactions on intelligent transportation systems, 12(3), pp.920-939.

[26] De Charette, R. and Nashashibi, F., 2009, June. Real time visual traffic lights recognition based on spot light detection and adaptive traffic lights templates. *In 2009 IEEE Intelligent Vehicles Symposium* (pp. 358-363). IEEE.

[27] Tae-Hyun, H., In-Hak, J. and Seong-Ik, C., 2006. Detection of traffic lights for vision-based car navigation system. In Advances in Image and Video Technology: First Pacific Rim Symposium, PSIVT 2006, *Hsinchu, Taiwan, December 10-13, 2006. Proceedings 1* (pp. 682-691). Springer Berlin Heidelberg.

[28] Levinson, J., Askeland, J., Dolson, J. and Thrun, S., 2011, May. Traffic light mapping, localization, and state detection for autonomous vehicles. *In 2011 IEEE International Conference on Robotics and Automation* (pp. 5784-5791). IEEE.

[29] Pan, S.J. and Yang, Q., 2010. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering, 22*(10), pp.1345-1359.

[30] Torrey, L. and Shavlik, J., 2010. Transfer learning. In Handbook of research on machine learning applications and trends: algorithms, methods, and techniques (pp. 242-264). IGI global.

[31] Elhenawy, M., Elbery, A.A., Hassan, A.A. and Rakha, H.A., 2015, September. An intersection game-theory-based traffic control algorithm in a connected vehicle environment. *In 2015 IEEE 18th international conference on intelligent transportation systems* (pp. 343-347). IEEE.

[32] Dresner, K. and Stone, P., 2006, July. Traffic intersections of the future. In AAAI (Vol. 6, pp. 1593-1596).

[33] Colombo, A. and Del Vecchio, D., 2012, April. Efficient algorithms for collision avoidance at intersections. *In Proceedings of the 15th ACM international conference on Hybrid Systems: Computation and Control* (pp. 145-154).

[34] Bruni, L., Colombo, A. and Del Vecchio, D., 2013, December. Robust multi-agent collision avoidance through scheduling. *In 52nd ieee conference on decision and control* (pp. 3944-3950). IEEE.

[35] Kanet, J.J. and Sridharan, V., 2000. Scheduling with inserted idle time: problem taxonomy and literature review. **Operations Research, 48**(1), pp.99-110.

[36] Makarem, L. and Gillet, D., 2012, October. Fluent coordination of autonomous vehicles at intersections. *In 2012 IEEE international conference on systems, man, and cybernetics* (SMC) (pp. 2557-2562). IEEE.

[37] Zohdy, I.H. and Rakha, H.A., 2016. Intersection management via vehicle connectivity: The intersection cooperative adaptive cruise control system concept. **Journal of Intelligent Transportation Systems, 20**(1), pp.17-32.

[38] Elhenawy, M., Elbery, A.A., Hassan, A.A. and Rakha, H.A., 2015, September. An intersection game-theory-based traffic control algorithm in a connected vehicle environment. *In 2015 IEEE 18th international conference on intelligent transportation systems* (pp. 343-347). IEEE.

[39] He, K., Zhang, X., Ren, S. and Sun, J., 2016. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778).

[40] Phiphiphatphaisit, S. and Surinta, O., 2020, March. Food image classification with improved MobileNet architecture and data augmentation. *In Proceedings of the 3rd International Conference on Information Science and Systems* (pp. 51-56).

[41] Mukherjee, S., 2022. The Annotated ResNet-50. Towards Data Science.

[42] Phiphiphatphaisit, S. and Surinta, O., 2020, March. Food image classification with improved MobileNet architecture and data augmentation. *In Proceedings of the 3rd International Conference on Information Science and Systems* (pp. 51-56).

[43] Alom, M.Z., Taha, T.M., Yakopcic, C., Westberg, S., Sidike, P., Nasrin, M.S., Van Esesn, B.C., Awwal, A.A.S. and Asari, V.K., 2018. The history began from alexnet: A comprehensive survey on deep learning approaches. **arXiv preprint arXiv:1803**.01164.

[44] Simonyan, K. and Zisserman, A., 2014. Very deep convolutional networks for large-scale image recognition. **arXiv preprint arXiv:1409**.1556.

[45] Karen, S., 2014. Very deep convolutional networks for large-scale image recognition. **arXiv preprint arXiv: 1409**.1556.

[46] Simonyan, K. and Zisserman, A., 2014. Very deep convolutional networks for large-scale image recognition. **arXiv preprint arXiv:1409**.1556.

[47] Chollet, F., 2017. Xception: Deep learning with depthwise separable convolutions. *In Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1251-1258).

[48] Carreira, J., Madeira, H. and Silva, J.G., 1998. Xception: A technique for the experimental evaluation of dependability in modern computers. *IEEE Transactions on Software Engineering*, *24*(2), pp.125-136.

[49] Endah, S.N. and Shiddiq, I.N., 2020, November. Xception architecture transfer learning for garbage classification. *In 2020 4th International Conference on Informatics and Computational Sciences* (ICICoS) (pp. 1-4). IEEE.

[50] Phiphiphatphaisit, S. and Surinta, O., 2020, March. Food image classification with improved MobileNet architecture and data augmentation. *In Proceedings of the 3rd International Conference on Information Science and Systems* (pp. 51-56).

[51] Huang, G., Liu, Z., Van Der Maaten, L. and Weinberger, K.Q., 2017. Densely connected convolutional networks. *In Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4700-4708).

[52] Tan, M. and Le, Q., 2019, May. Efficientnet: Rethinking model scaling for convolutional neural networks. *In International conference on machine learning* (pp. 6105-6114). PMLR.

CURRICULUM VITAE

PERSONAL INFORMATION

Name Surname: Mohammed Ghazi Khassaf ALSHAMI

Nationality: IRAQ

EDUCATION

Degree	Institution	Date of graduation
Bachelor	Computer Engineering from Madenat Alelem University	2018
High School	Alshuruq-Baghdad	2013

Education work experience

Year	Institution	Task
2017-2019	Zain Telecom Company	Designer

Languages

Arabic, English, Turkish

SAMPLE [30] Allen C. Tsai, Peter W. Gibbens, and R. Hugh Stone, 2006. Terminal Phase Vision-Based Target Recognition and 3D Pose Estimation for a Tail-Sitter, Vertical Takeoff and Landing Unmanned Air Vehicle, First Pacific Rim Symposium (PSIVT), December 10-13, 2006, Hsinchu, Taiwan.