Düzce University
Journal of Science & Technology

Research Article

Performance Evaluation Of Image Recognition Algorithms On Marine Vessels And Optimum Parameter Selection

Cansu CANBOLAT[a], Yasemin ATILGAN ŞENGÜL[b]
Ahmet Yekta KAYMAN[c]

[a] *Information Security Technology Program, Vocational School, Istanbul Bilgi University, Istanbul, TURKEY*
[b] *Department of Industrial Engineering, Faculty of Engineering, Dogus University, Istanbul, TURKEY*
[c] *Department of Industrial Engineering, Faculty of Engineering, Dogus University, Istanbul, TURKEY*
* *Sorumlu yazarın e-posta adresi: cansu.canbolat@bilgi.edu.tr*
DOI: 10.29130/dubited.1543061

ABSTRACT

Today, advancements in sensor technology, image processing models, and deep neural network methods have driven significant progress in the field of autonomous driving. This dynamic area has attracted substantial attention, with numerous studies being conducted across both academia and the private sector; hence studies specifically focused on the safe driving of driverless vehicles are still very limited. The basis of the studies conducted was created for land vehicles, and the data sets created for the operation of artificial intelligence models were prepared in this context. In this study, the algorithms used for autonomous driving were tested using the original data set created from objects on the sea to optimize the navigation of sea vehicles on the sea. Image processing methods have recently gained great importance in terms of recognizing vehicles on the sea and providing autonomous driving. In this study, a high-resolution and wide-ranging original data set consisting of 44965 objects sea objects was developed to identify objects on the sea. With this dataset, image processing technology was utilized to conduct analyses and optimizations for the recognition and classification of objects on the sea. Efforts were made to identify the most effective model among various alternatives. The study aims to detect and classify objects on the sea surface from long distances (over 1000 meters), ensuring the safe operation of sea vehicles and providing decision support. To enable the successful real-time identification of the created dataset, it was categorized into six distinct classes.As a result of the classification process, data labeling was performed according to 6 classes: Cargo_Ship, Tanker_Ship, RoRo/Ferry/Passenger, Boats, Tug_Boats, Speciality_Vessels. The created data set was tested with the most common real-time recognition models, SSD, Faster R-CNN, EfficientDet algorithms under the TensorFlow library. Results were obtained according to 6 different output parameter values, AP-50, AP-75, Av. Recall, F1-50, F1-75 and L/TL, on the models. According to the obtained results, SSD Mobilnet v1 was determined as the most successful algorithm.

Keywords: Marine, Vehicle detection, Image processing, Deep neural network, Autonomous driving

Received: 03/09/2024, Revised: 05/01/2025, Accepted: 14/01/2025

Deniz Taşitlari Üzerinde Görüntü Tanima Algoritmalarinin Performans Değerlendirmesi Ve Optimum Parametre Seçimi

ÖZET

Günümüzde gelişen sensör teknolojisi, görüntü işleme modelleri ve derin sinir ağları yöntemleri ile otonom sürüş alanında da önemli gelişmeler yaşanmakta ve hem özel sektörde hem de akademide bu yönde çeşitli çalışmalar gerçekleştirilmektedir. Sürücüsüz araçların güvenli sürüşüne yönelik bu çalışmalar henüz çok kısıtlıdır. Yapılan çalışmaların temeli kara taşıtları için oluşturulmuş, yapay zekâ modellerinin çalıştırılması için oluşturulan veri setleri bu bağlamda hazırlanmıştır. Bu çalışmada otonom sürüş için kullanılan algoritmalar deniz taşıtlarının deniz üzerinde seyrederken optimize edilmesi için deniz üzerindeki nesnelerden oluşturulan orijinal veri seti kullanılarak test edilmiştir. Görüntü işleme metotları, deniz üzerindeki taşıtların tanınması ve otonom sürüş sağlanması açısından son zamanlarda büyük önem kazanmıştır. Bu çalışmada, deniz üzerindeki nesneleri tanımlamak için, deniz üzerindeki nesnelerden oluşan 44965 adetlik yüksek çözünürlüklü ve geniş kapsamlı orijinal veri seti oluşturulmuştur. Bu veri seti ile deniz üzerindeki nesnelerin tanınma ve sınıflandırılmasına yönelik görüntü işleme teknolojisi ile analiz ve optimizasyonlar yapılarak, modeller arasında en iyi model belirlenmeye çalışılmıştır. Deniz yüzeyindeki nesneleri, uzak mesafeden (1000m+) tespit edilip sınıflandırılması, deniz araçları için güvenli kullanım oluşturulması ve karar desteği sağlanması hedeflenmektedir. Oluşturulan veri setinin gerçek zamanlı ortamda başarılı şekilde tanımlanabilmesi için veri seti 6 adet sınıfa ayrılmıştır. Sınıflandırma işlemi sonucunda oluşturulan; Cargo_Ship, Tanker_Ship, RoRo/Ferry/Passenger, Boats, Tug_Boats, Speciality_Vessels olmak üzere 6 adet sınıfa göre veri etiketleme işlemi yapılmıştır. Oluşturulan veri seti, en yaygın gerçek zamanlı tanıma modelleri olan, TensorFlow kütüphanesi altındaki SSD, Faster R-CNN, EfficientDet algoritmaları ile test edilmiştir. Modeller üzerinde de AP-50, AP-75, Av. Recall, F1-50, F1-75 ve L/TL olmak üzere 6 farklı çıktı parametresi değerine göre sonuçlar elde edilmiştir. Elde edilen sonuçlara göre, SSD Mobilnet v1 en başarılı algoritma olarak tespit edilmiştir.

Anahtar Kelimeler: *Deniz, aracı tespiti, Görüntü işleme, Derin sinir ağı, Otonom sürüş*

I. INTRODUCTION

Today, various research and developments are being carried out for the safe driving of vehicles, but these studies are mostly focused on land vehicles. Studies are also being carried out for the autonomous driving of sea vehicles, but it has been observed that studies have not yet been conducted with a sufficient data set. Interest in autonomous vehicles is increasing worldwide. The need for development in this technology is clearly seen [1]. In the field of autonomous vehicles, there are also studies on automatic collision avoidance and autonomous travel and route planning.

It is seen that the studies in these areas are increasing and a need for a larger data set arises. In the literature, there are studies showing that navigation applications in marine systems using only image data can be used to navigate without hitting obstacles while other systems such as GPS and radar are disabled [2]. Similarly, an autonomous vehicle that can be used effectively for high-speed vehicles and complex interactions with the environment with a dynamic motion planning approach has also been developed [3]. In autonomous vehicles, it has become possible to detect environmental objects and move without hitting them. The perception of environmental objects by autonomous vehicles plays a critical role, especially in dynamic and variable environmental conditions [4].

Object detection and recognition is one of the most important research topics in autonomous vehicle technologies. The main reason for this is that in autonomous driving, a control movement first detects the object and then identifies that object [1].

Object detection and recognition is a critical technology that allows autonomous vehicles to accurately identify their environment. Developments in object detection and recognition play an important role in improving the safety and performance of vehicles[5]. Recently, object recognition applications for real-life vehicles have developed considerably. In autonomous driving systems, in order to achieve a high accuracy rate of object recognition, accurate labeling of the data and a highly diverse dataset are required [6]. A good dataset is one that contains all real-time conditions to the maximum extent and has a high number of object diversity.

Deep learning systems are based on very comprehensive calculations that mimic the functions of the human brain. In 1943, McCulloch and Pitts developed a model that imitates the thought process. This model is based on mathematics and algorithms called neural logic [7]. Deep learning systems have come to the fore in many areas such as voice recognition, image recognition, and natural language processing. Deep learning algorithms have provided significant performance increases, especially in areas such as speech recognition, computer vision, and natural language processing, by using large data sets and powerful computational methods [8]. Studies on deep learning systems include research on the ability to analyze by processing visual data [9]. The labeling of images, object identification, and object classification systems used by companies such as Facebook and Google have been realized with Deep Learning models[10]. Thanks to the application of deep learning technologies, autonomous vehicles have become safer and more efficient [11].

Within the scope of the study, deep learning and image processing based methods focusing on object detection in the literature were examined. The most popular of these methods are Yolo-V3, Yolo-V4 under the Darknet library and SSD, Faster R-CNN, EfficientDet models under the TensorFlow library were examined.

Yolo-V3 and Yolo-V4 under the Darknet library and SSD, Faster R-CNN and EfficientDet models under the TensorFlow library have achieved significant success in object detection and recognition systems today. Yolo-V3 is a widely used model for real-time object detection, especially by offering high accuracy and speed. Yolo-V3 is quite successful in terms of speed, as it can perform both classification and localization in a single network [12]. Yolo-V4, on the other hand, has achieved better results in larger data sets and high-resolution images. The efficiency of Yolo-V4 has become much faster as a result of the optimized architecture of the model and many improvements[13].

SSD (Single Shot MultiBox Detector) is a model designed for more efficient object detection. SSD offers the ability to detect objects at different scales and types, which makes the SSD model quite attractive for applications that require fast and accurate detection. SSD can quickly classify and locate each object using multiple detection boxes on the image [14].

Faster R-CNN is a model that accelerates object detection using region proposal networks (RPN). This model works much more efficiently than previous object detection models by greatly reducing the time required to determine a region. Due to this working system, Faster R-CNN exhibits a very high success rate in terms of both speed and accuracy [15]. EfficientDet stands out with its smaller model sizes and more efficient computational requirements. This model is especially suitable for mobile devices and systems with object detection under limited hardware conditions. It increases its usability on mobile devices and real-time applications [16].

Each of these models offers different advantages and application areas in object detection tasks, and each is used according to different needs and usage scenarios. For example, Yolo-V3 and Yolo-V4 stand out in real-time applications that require fast and efficient detection, while EfficientDet provides high efficiency on more limited hardware, and Faster R-CNN is suitable for situations where precision is critical. These differences allow each model to use its own special advantages in applications in object detection and recognition systems.

In this study, it is aimed to detect objects faster and more accurately. Since it is aimed to see the closest performance to real-time detection, SSD Mobilnet v1 algorithm was selected.

As a result of this study, it is anticipated that high accuracy will be achieved in the detection and classification of objects on the sea, and the knowledge and output obtained from the study will form the basis for decision support on the safe use of marine vehicles.

The other parts of the study are organized as follows; in Section II, the materials and methods used are explained, in Section III, the calculations and discussions are explained. In the last section, Section IV, the study is concluded.

II. MATERIALS AND METHODS

A. AUTONOMOUS VEHICLES

The technological progress of autonomous vehicles has accelerated, especially with the integration of artificial intelligence and deep learning methods, which has enabled the development of safer and more efficient driving systems [11]. In recent years, it has been observed that sensor technologies and data processing algorithms developed for autonomous vehicles have provided significant improvements in object perception and decision-making processes [1]. Autonomous vehicles are mechanisms that interpret the data they collect from their environment in order to move in physical environments, thanks to their motion systems, decision mechanisms, sensor systems and algorithms to perceive the environment [17]. The first experiments on autonomous driving started in the 1920s and the first prototypes emerged in the 1950s. In 1997, the Tsukuba Mechanical Engineering Laboratory in Japan developed the first truly autonomous vehicle [18].

B. OBJECT DETECTION AND OBJECT RECOGNITION

Object detection and object recognition, which are the most important elements of image processing applications, are important topics that have been studied for years [19]. Various algorithms and methods have been developed for object detection and recognition. There are popular libraries used in object detection and recognition. The most common libraries include YOLO, Single Shot Multibox Detactor (SSD), Region Based Convolutional Networks (R-CNN), Fast R-CNN, Faster R-CNN and Mask R-CNN [20]. The success rate of object detection and classification is high accuracy detection and classification. One of the most important parameters for a good detection and classification model to be applied in real time is to create an efficient data set. The effect of the data set on the success rate of the model is quite large. In deep learning-based models, the model performance increases in direct proportion to the size and diversity of the data set [21]. The first condition for a good model is to pass through a good data set, and the success of the created model depends on creating a model suitable for the data set and training it.

Studies on object detection and classification argue that object detection in the marine environment that we aim to reach can approach the desired success [22].

C. DATA SET PREPARATION AND CLASSIFICATION

The data set used in the study consists of high-resolution photographs taken in the Bosphorus and different seas, created together with Massive Yacht Technologies. For the models to work more efficiently during the training phase, photographs of the same marine vehicles were taken from many different angles and the data set is consisting from 44965 pictures of marine vehicles originally created by Massive Yacht Technologies company for this purpose. Since the data set size is considered to be enough for the algorithms to run efficiently no more photographs were taken and added to the data set. The dataset is planned to be further expanded for future studies.. Classifying marine vessels plays a crucial role in ensuring maritime safety and managing traffic, with significant applications in both civilian and military sectors[23]. However, this task is more complex than other target recognition problems due to the substantial variations in viewing angles, lighting conditions, and scale, in addition to the chaotic nature of the image background [24].

Convention on the International Regulations for Preventing Collisions at Sea (COLREGs) defines the rules and gives clear indication about passing, approaching, giving way and overtaking other boats and according to rule 18 of The lower most vessel on the list is the give way vessel, and must stay out of the way of vessels that are higher on the list. 1.Overtaken vessel (top priority), 2.Vessels not under command, 3.Vessels restricted in their ability to maneuver, 4.Vessels constrained by draft, 5.Fishing vessels engaged in fishing, with gear deployed, 6.Sailing vessels, 7. Power driven vessels [25].

Initially, 20 classes were identified in our study. These classes are given in Table 1. In order to achieve a higher success rate in a real-time recognition system, the more important point than the number of classes is the correct categorization of the classes. As the vehicle moves over the sea, it is aimed to recognize objects faster and more accurately. For this reason, the number of groups was reduced from 20 to 6 according to characteristics such as the right of way (passage priorities) and the speed limit, so that the autonomous device can drive safely without disrupting maritime traffic. The final form of the newly determined classes after the class merging process and the number of photographs in each category is given in Table 2.

Table 1. Initial Classes of Objects

Group	Category	Group	Category
1	Cargo Ship	11	Boat
2	Tanker Ship	12	Research Ship
3	RoRo	13	Rescue Ship
4	Cruise Ship	14	Buoy
5	Ferry	15	Sailboat
6	Passenger Boat	16	Yacht
7	Supply Ship	17	Jet Ski
8	Dredger Ship	18	Kayak
9	Tugboats	19	Military Ships
10	Fishing Ship	20	Coast Guard

Table 2. Number of Tagged Photos per each class

Group	Category	Quantity	Ratio
1	Cargo Ship	7082	15,75%
2	Tanker Ship	8292	18,44%
3	RoRo/Ferry/Passanger	8966	19,94%
4	Boats	7174	15,95%
5	Tug_Boats	7680	17,08%
6	Speciality_Vessels	5771	12,83%
Total		44965	

The LabelImg program was used to label all photos added to the data set according to their classes. A common resolution ratio was determined with the LabelImg Program and all images were labeled. Sample images of the data set are shown in Figure 1.

Figure 1. Sample images with different characteristics in the dataset

To start the training process, 14257 photos were selected from 44965 photos in the dataset and labeled in 6 categories. Since some photos in the dataset contain more than one marine vessel, a photo is labeled for different numbers of categories and because of this the total number of photos in the dataset and the total number of labels differ.

Table 3. Number of Tagged Photos per each category

Group	Category	Quantity	Ratio
1	Cargo Ship	2338	16,40%
2	Tanker Ship	2529	17,74%
3	RoRo/Ferry/Passanger	2568	18,01%
4	Boats	2353	16,50%
5	Tug_Boats	2532	17,76%
6	Speciality_Vessels	1937	13,59%
	Total	14257	

D. TRAINING MODELS AND PARAMETERS

In the study, the classification of objects with SSD, Faster R-CNN, EfficientDet models under the TensorFlow library was tested on the created data set. In the study, 14257 images were used for training and 44965 images were used for performance measurements. In this paper, similar to the dataset division of the VisDrone2019 Challenge, we divided the entire dataset into training, validation, and testing sets, each containing 6471 samples, 548 samples, and 1610 samples and the sample images for training and testing are both set to the size of 640*640 gibi… Step number and batch size were given as input parameters to the models. As a result of the training, 6 output parameter results were obtained, namely AP-50, AP-75, Av. Recall, F1-50, F1-75 and L/TL. The success rates will be compared according to the obtained output parameters.

There are 4 possibilities for the prediction result in the trained models:

True Positive (TP): It is one of the possibilities that our model predicted correctly. An instance for which both predicted and actual values are positive. There is an object in the photo, the model detected an object..

True Negative (TN): It is one of the possibilities that our model predicted correctly. An instance for which both predicted and actual values are negative. There is no object in the photo, the model did not detect an object..

False Positive (FP): It is one of the possibilities that our model predicted incorrectly. An instance for which predicted value is positive but actual value is negative. There is no object in the photo, the model detected an object.
False Negative (FN): It is one of the possibilities that our model predicted incorrectly. An instance for which predicted value is negative but actual value is positive. There is an object in the photo, the model did not detect an object.
The parameters used to compare the performance rates of the models are given in Table 4.

Table 4. Parameters Compared in the Training Process

Parameters	
Response 1	AP-50
Response 2	AP-75
Response 3	Av. Recall
Response 4	F1-50
Response 5	F1-75
Response 6	L/TL

Average Precision (AP): AP50 and AP75 mean AP at 50% and 75% IoU threshold respectively.
Precision (P): Measures how accurate model predictions are.

$$Precision = \frac{TP}{(TP+FP)} \qquad \textit{Equation (1)}$$

Recall (R): Measures how well model finds all positives.

$$Recall = \frac{TP}{(TP+FN)} \qquad \textit{Equation (2)}$$

Harmonic Mean: F1 Score score is used when we need to find a balance between precision and recall. It is especially preferred for unequally distributed classes.

$$Harmonic\ Mean = \frac{2*P*R}{(P+R)} \qquad \textit{Equation (3)}$$

III. CALCULATIONS AND DISCUSSION

An experimental design model was established to optimize the values of AP50, AP75, Av.Recall, F1/50, F1/75 and L/TL for the determined (Efficientdet d0, Efficientdet d1, Efficientdet d2, Efficientdet d3, Efficientdet d4, F RCNN Inception, F RCNN Resnet 152, F RCNN Resnet101 v1, F RCNN Resnet50 v1, SSD Mobilnet v1, SSD Mobilnet v2, SSD Resnet 101, SSD Resnet 152, SSD Resnet50 algorithms with Batch Size 2, 4 and Run numbers (Run) 90, 130, 250 thousand. In the model, there are 2 input parameters, namely batch size and step number, and 6 output parameters, namely AP-50, AP-75, Av. Recall, F1-50, F1-75 and L/TL. is available. The experimental design created with the Design Expert program created the combinations that should be tried for the Batch size 2, 4 and Run 90, 130, 250 parameters for 14 algorithms. With the 53 combinations created by the program and that should be tried, instead of performing 14*2*3 = 84 full factorial number of experiments, 63% of the required experiments were performed and 37% time was saved. Considering that each experiment lasts about 1 day, this saved about 37 days. The results of the 53 experiments performed on input parameters and 6 output parameters are given in Table 5. It is desired that the AP-50, AP-75, Av. Recall, F1-50, F1-75 values from the outputs are close to 1 and the L/TL value is close to 0.

Table 5. AP50, AP75, Av. Recall, F1/50, F1/75, L/TL Values of Algorithms

	Factor 1	Factor 2	Factor 3	Response 1	Response 2	Response 3	Response 4	Response 5	Response 6
Run	**A:Run**	**B:BS**	**C:Algoritma**	**AP50**	**AP75**	**Av.Recall**	**F1/50**	**F1/75**	**L/TL**
1	90	4	Efficientdet d0	0,665	0,476	0,628	0,646	0,542	0,704
2	90	2	F RCNN Inception	0,881	0,811	0,825	0,852	0,818	0,112
3	250	4	F RCNN Resnet50 v1	0,853	0,778	0,792	0,821	0,785	0,222
4	90	4	SSD Resnet 101	0,711	0,646	0,777	0,743	0,705	0,437
5	250	4	SSD Mobilnet v1	0,866	0,8	0,825	0,845	0,832	0,097
6	90	4	F RCNN Resnet 152	0,549	0,427	0,554	0,551	0,482	0,356
7	130	4	F RCNN Inception	0,9	0,832	0,84	0,87	0,836	0,102
8	250	2	Efficientdet d1	0,735	0,597	0,697	0,715	0,643	0,379
9	90	4	SSD Resnet 152	0,726	0,642	0,734	0,73	0,685	0,456
10	130	4	Efficientdet d4	0,823	0,73	0,754	0,787	0,742	0,348
11	130	2	Efficientdet d0	0,744	0,619	0,681	0,711	0,649	0,397
12	250	4	Efficientdet d0	0,722	0,66	0,724	0,747	0,69	0,394
13	250	2	F RCNN Inception	0,602	0,543	0,601	0,601	0,571	0,47
14	90	4	F RCNN Resnet50 v1	0,848	0,773	0,779	0,812	0,776	0,2
15	130	4	Efficientdet d4	0,823	0,73	0,754	0,787	0,742	0,348
16	130	2	F RCNN Resnet 152	0,481	0,391	0,507	0,494	0,441	0,422
17	130	4	Efficientdet d1	0,758	0,573	0,658	0,704	0,613	0,397
18	250	4	F RCNN Resnet 152	0,653	0,562	0,608	0,63	0,584	0,336
19	250	4	F RCNN Inception	0,425	0,305	0,487	0,454	0,375	0,593
20	250	4	SSD Resnet50	0,806	0,737	0,804	0,805	0,769	0,353
21	130	2	F RCNN Resnet101 v1	0,477	0,372	0,496	0,486	0,425	0,443
22	130	4	SSD Resnet 101	0,788	0,718	0,775	0,781	0,745	0,379
23	250	4	SSD Resnet 152	0,754	0,666	0,727	0,74	0,695	0,453
24	130	2	Efficientdet d3	0,781	0,673	0,717	0,748	0,694	0,369
25	250	4	Efficientdet d2	0,792	0,691	0,752	0,771	0,72	0,385
26	130	4	Efficientdet d1	0,758	0,573	0,658	0,704	0,613	0,397
27	90	4	Efficientdet d3	0,778	0,672	0,731	0,754	0,7	0,383
28	90	2	SSD Resnet 101	0,516	0,438	0,701	0,594	0,539	0,574
29	90	2	SSD Mobilnet v2	0,704	0,631	0,743	0,722	0,682	0,399
30	250	2	SSD Resnet 101	0,781	0,708	0,771	0,776	0,738	0,389
31	250	4	F RCNN Resnet101 v1	0,856	0,784	0,8	0,827	0,791	0,178
32	250	4	SSD Mobilnet v2	0,835	0,767	0,8	0,817	0,783	0,302
33	90	2	Efficientdet d4	0,749	0,629	0,708	0,728	0,666	0,382
34	250	4	Efficientdet d3	0,797	0,702	0,759	0,778	0,729	0,289
35	90	4	Efficientdet d2	0,674	0,56	0,689	0,681	0,618	0,548
36	250	2	SSD Mobilnet v2	0,759	0,681	0,765	0,762	0,721	0,357
37	130	2	Efficientdet d0	0,744	0,619	0,681	0,711	0,649	0,397
38	130	2	SSD Resnet 101	0,538	0,462	0,72	0,616	0,563	0,553
39	130	2	SSD Resnet 152	0,666	0,58	0,731	0,697	0,647	0,49
40	90	2	SSD Resnet50	0,435	0,357	0,684	0,532	0,469	0,656

41	130	2	SSD Mobilnet v1	0,777	0,712	0,779	0,778	0,744	0,478
42	130	4	SSD Mobilnet v2	0,814	0,75	0,789	0,801	0,769	0,325
43	250	2	Efficientdet d4	0,706	0,606	0,705	0,705	0,651	0,426
44	90	4	SSD Mobilnet v1	0,821	0,753	0,797	0,809	0,774	0,538
45	130	2	Efficientdet d3	0,781	0,673	0,717	0,748	0,694	0,369
46	90	2	Efficientdet d1	0,74	0,579	0,662	0,699	0,618	0,383
47	250	4	SSD Resnet 101	0,826	0,761	0,796	0,811	0,778	0,346
48	130	2	F RCNN Resnet101 v1	0,477	0,372	0,496	0,486	0,425	0,443
49	130	2	Efficientdet d2	0,594	0,463	0,647	0,619	0,54	0,517
50	90	4	SSD Resnet50	0,793	0,729	0,79	0,791	0,758	0,379
51	130	2	F RCNN Resnet50 v1	0,612	0,488	0,597	0,604	0,537	0,36
52	90	4	F RCNN Resnet101 v1	0,584	0,49	0,567	0,575	0,526	0,389
53	250	2	SSD Resnet50	0,806	0,737	0,804	0,805	0,769	0,353

When the performances of the algorithms need to be evaluated collectively in terms of multiple outputs, using direct output values is not appropriate for comparison, so the success ranks of the algorithms for each output were determined and according to this, the success rank values of each combination for 6 outputs according to the outputs for 53 trials are given in Table 6. According to the findings, the F RCNN Inception algorithm demonstrated the best performance when configured with a batch size of 4 and executed over 130,000 iterations. The optimal results were achieved across five key parameters: AP50, AP75, Average Recall, F1/50, and F1/75.. The performance of the same algorithm in batch size 2 and 90 thousand runs was generally in second place. But in contrast to this F RCNN Inception algorithm performs as the worst algorithm with 250 thousand runs. This inconsistency has brought the performance of this algorithm into doubt.

Another algorithm with the highest performance is the SSD Mobilnet v1 batch size 4 and 250 thousand runs, which came first for one output (L/TL), second for two outputs (Av. Recall, F1/75) and third for three outputs (AP50, AP75, F1/50).

Table 6. Success Rankings of Algorithms According to Parameters

Run	A:Run	B:BS	C:Algoritma	Rank 1	Rank 2	Rank 3	Rank 4	Rank 5	Rank 6
				AP50	**AP75**	**Av. Recall**	**F1/50**	**F1/75**	**L/TL**
1	90	4	Efficientdet d0	40	44	44	40	43	53
2	90	2	F RCNN Inception	2	2	2	2	3	3
3	250	4	F RCNN Resnet50 v1	5	5	10	5	5	6
4	90	4	SSD Resnet 101	35	27	15	26	22	38
5	250	4	SSD Mobilnet v1	3	3	2	3	2	1
6	90	4	F RCNN Resnet 152	46	48	49	48	48	16
7	130	4	F RCNN Inception	1	1	1	1	1	2
8	250	2	Efficientdet d1	32	34	35	31	35	23
9	90	4	SSD Resnet 152	33	28	24	28	28	42
10	130	4	Efficientdet d4	9	13	20	14	16	13
11	130	2	Efficientdet d0	29	31	38	32	32	34
12	250	4	Efficientdet d0	34	26	28	25	27	30
13	250	2	F RCNN Inception	43	41	46	45	41	43
14	90	4	F RCNN Resnet50 v1	6	6	13	7	8	5

15	130	4	Efficientdet d4	9	13	20	14	16	13
16	130	2	F RCNN Resnet 152	49	49	50	50	50	36
17	130	4	Efficientdet d1	25	37	41	35	38	34
18	250	4	F RCNN Resnet 152	41	39	45	41	40	10
19	250	4	F RCNN Inception	53	53	53	53	53	51
20	250	4	SSD Resnet50	13	11	4	10	10	15
21	130	2	F RCNN Resnet101 v1	50	50	51	51	51	40
22	130	4	SSD Resnet 101	18	16	16	16	14	23
23	250	4	SSD Resnet 152	27	25	27	27	24	41
24	130	2	Efficientdet d3	19	22	30	23	25	20
25	250	4	Efficientdet d2	17	20	22	20	21	27
26	130	4	Efficientdet d1	25	37	41	35	38	34
27	90	4	Efficientdet d3	22	24	25	22	23	26
28	90	2	SSD Resnet 101	48	47	34	46	45	50
29	90	2	SSD Mobilnet v2	37	29	23	30	29	35
30	250	2	SSD Resnet 101	19	18	17	19	18	29
31	250	4	F RCNN Resnet101 v1	4	4	6	4	4	4
32	250	4	SSD Mobilnet v2	7	7	6	6	6	8
33	90	2	Efficientdet d4	28	30	32	29	30	24
34	250	4	Efficientdet d3	15	19	19	17	19	7
35	90	4	Efficientdet d2	38	40	36	39	36	48
36	250	2	SSD Mobilnet v2	24	21	18	21	20	17
37	130	2	Efficientdet d0	29	31	38	32	32	34
38	130	2	SSD Resnet 101	47	46	29	43	42	49
39	130	2	SSD Resnet 152	39	35	25	38	34	45
40	90	2	SSD Resnet50	52	52	37	49	49	52
41	130	2	SSD Mobilnet v1	23	17	13	17	15	44
42	130	4	SSD Mobilnet v2	12	10	12	12	10	9
43	250	2	Efficientdet d4	36	33	33	34	31	37
44	90	4	SSD Mobilnet v1	11	9	8	9	9	47
45	130	2	Efficientdet d3	19	22	30	23	25	20
46	90	2	Efficientdet d1	31	36	40	37	36	26
47	250	4	SSD Resnet 101	8	8	9	8	7	11
48	130	2	F RCNN Resnet101 v1	50	50	51	51	51	40
49	130	2	Efficientdet d2	44	45	43	42	44	46
50	90	4	SSD Resnet50	16	15	11	13	13	23
51	130	2	F RCNN Resnet50 v1	42	43	47	44	46	18
52	90	4	F RCNN Resnet101 v1	45	42	48	47	47	29
53	250	2	SSD Resnet50	13	11	4	10	10	15

In Figure 2 the minimum and maximum rank values obtained by each combination in terms of 6 outputs are shown, and the narrower these ranges are, the more stable the performance of the algorithm in terms of outputs can be said to be. For example, the rank values obtained in the Batch size 4 and 130 thousand run parameters of the F RCNN Inception algorithm were always in the 1st and 2nd places for 6 outputs, and the range value was 1, which can be interpreted as the algorithm being generally successful.

Similarly, while F RCNN Resnet 152 was generally in the 39th - 45th places in the Batch size 4 and 250 thousand runs, it was in the 10th place for the L/TL output, and it does not seem to be consistent in terms of performance.

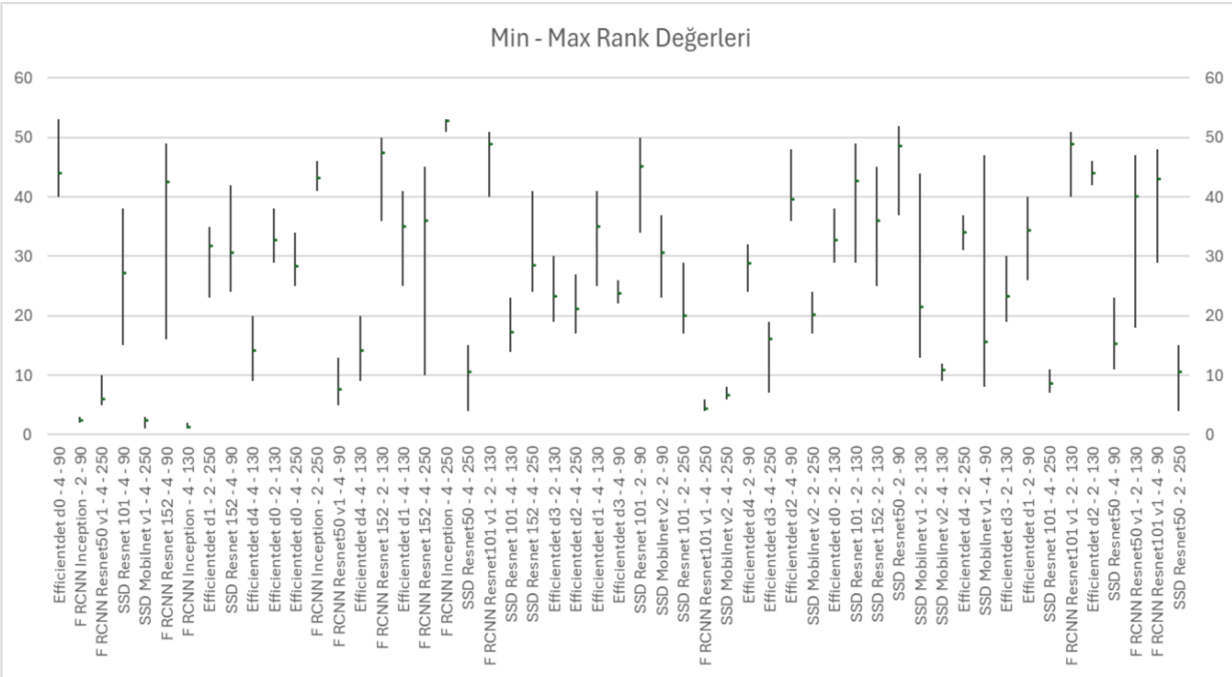

Figure 2. Range of Rank Values of Algorithms According to Parameters

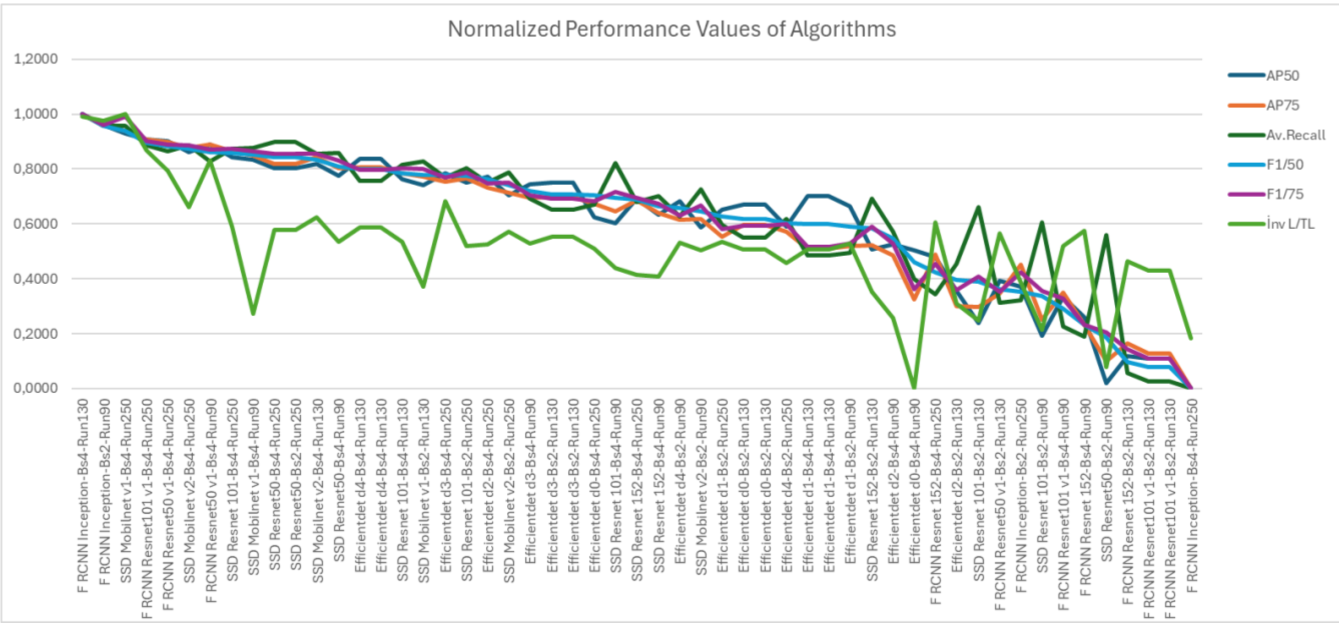

Figure 3. Normalized AP50, AP75, Av. Recall, F1/50, F1/75, inverse L/TL Values of Algorithms

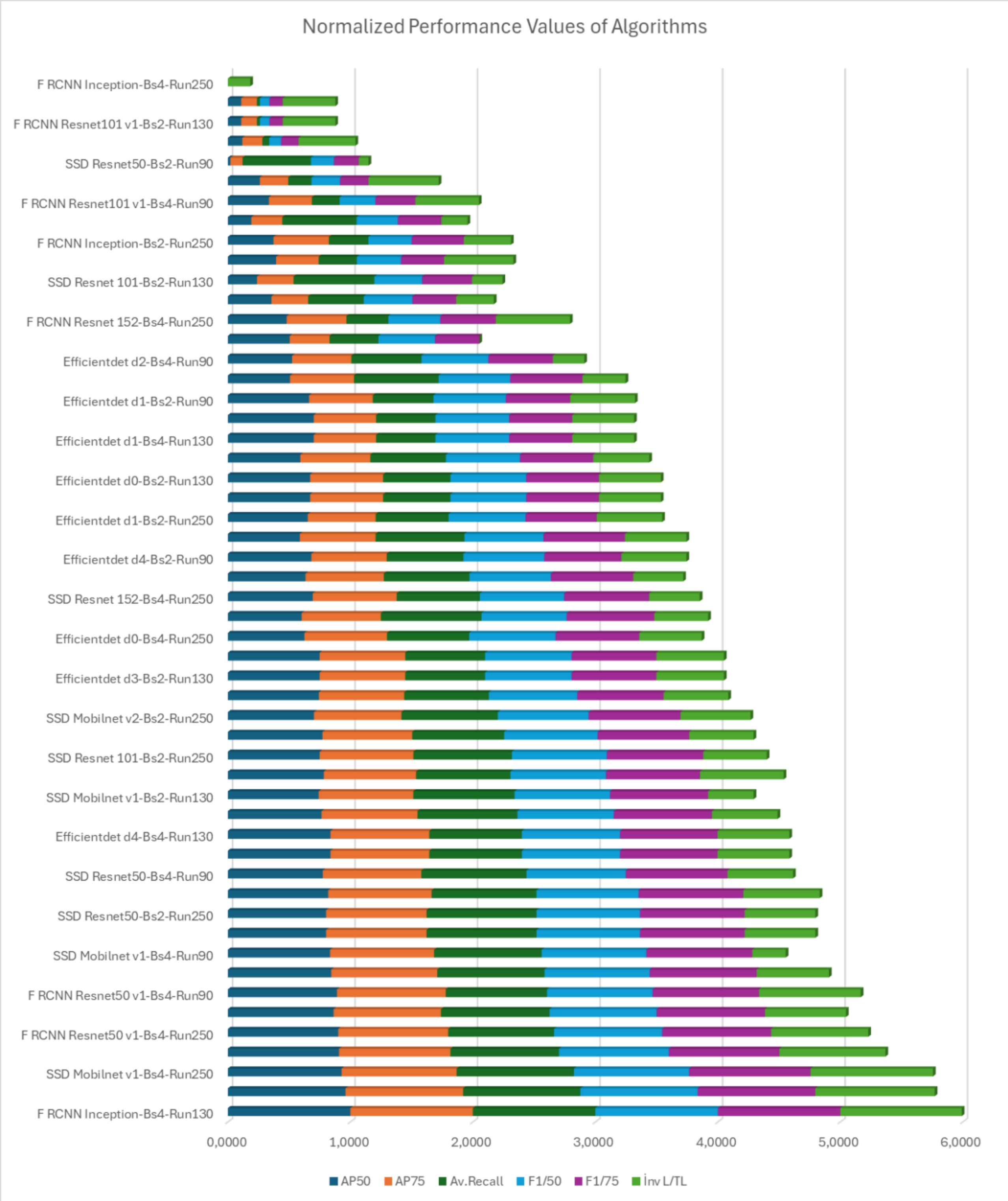

Figure 4. Cumulative Normalized AP50, AP75, Av. Recall, F1/50, F1/75, inversre L/TL Values of Algorithms (higher is better)

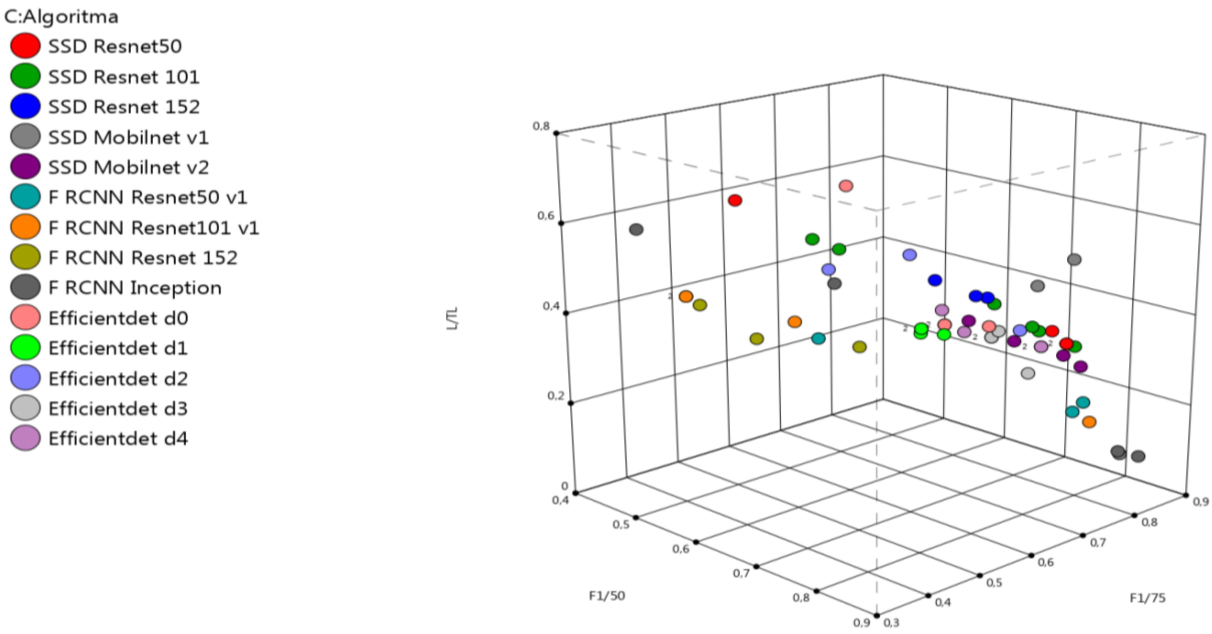

Figure 5. F1/50, F1/75 and L/TL Values of Algorithms

When the proposed Response 4 F1/50 and Response 5 F1/75 are evaluated together to compare the algorithms on the current problem, the order of SSD Mobilnet v1, F RCNN Resnet101 v1, F RCNN Resnet50 v1 does not change.

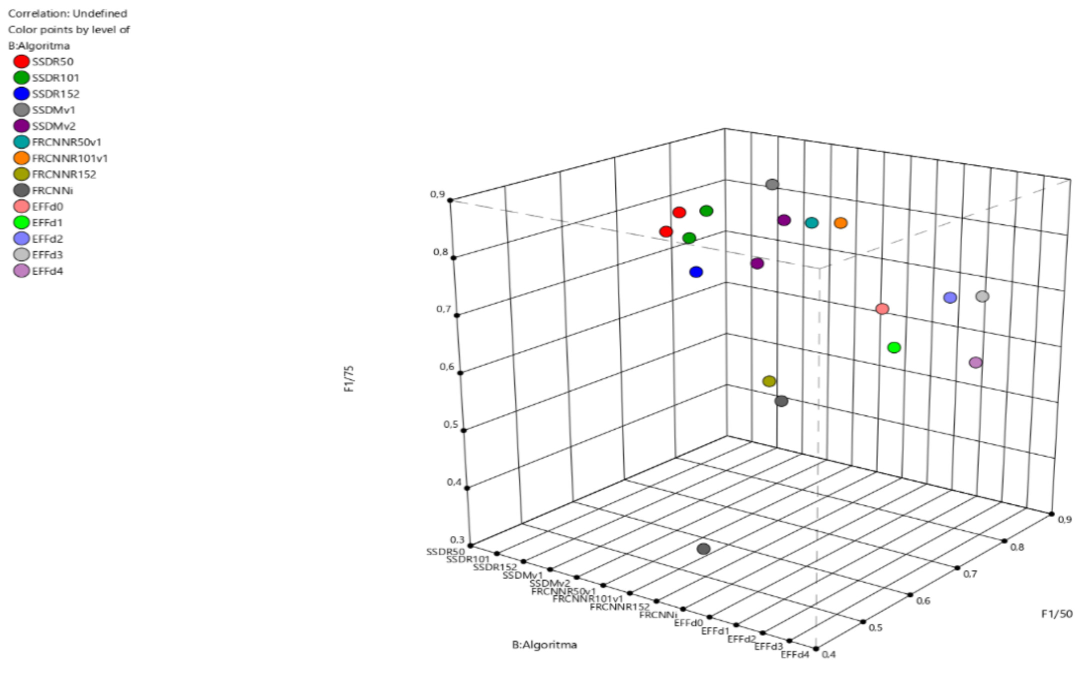

Figure 6. F1/50 and F1/75 Values of Algorithms

The overall performance of various algorithm classes is outlined in Table 6, with SSD MobilNet v1 identified as the top-performing algorithm for this dataset.

Algorithms	Average AP50	Average AP75	Av.Recall	Average F1/50	Average F1/75	Average L/TL
Efficientdet d0	0,7188	0,5935	0,6785	0,7038	0,6325	0,4730
Efficientdet d1	0,7478	0,5805	0,6688	0,7055	0,6218	0,3890
Efficientdet d2	0,6867	0,5713	0,6960	0,6903	0,6260	0,4833
Efficientdet d3	0,7843	0,6800	0,7310	0,7570	0,7043	0,3525
Efficientdet d4	0,7753	0,6738	0,7303	0,7518	0,7003	0,3760
F RCNN Inception	0,7020	0,6228	0,6883	0,6943	0,6500	0,3193
F RCNN Resnet 152	0,5610	0,4600	0,5563	0,5583	0,5023	0,3713
F RCNN Resnet101 v1	0,5985	0,5045	0,5898	0,5935	0,5418	0,3633
F RCNN Resnet50 v1	0,7710	0,6797	0,7227	0,7457	0,6993	**0,2607**
SSD Mobilnet v1	**0,8213**	**0,7550**	**0,8003**	**0,8107**	**0,7833**	0,3710
SSD Mobilnet v2	0,7780	0,7073	0,7743	0,7755	0,7388	0,3458
SSD Resnet 101	0,6933	0,6222	0,7567	0,7202	0,6780	0,4463
SSD Resnet 152	0,7153	0,6293	0,7307	0,7223	0,6757	0,4663
SSD Resnet50	0,7100	0,6400	0,7705	0,7333	0,6913	0,4353
Average	**0,7186**	**0,6232**	**0,7091**	**0,7125**	**0,6613**	**0,3915**

Table 7. Average AP50, AP75, Av. Recall, F1/50, F1/75, L/TL Values of Algorithm Classes

The use of the ranking method according to the outputs to compare the algorithms on the current problem is suitable for obtaining a dominant option from the results of the experiments made for the current problem, but the limitation of this method is that it depends on the evaluation of the researcher only on the results obtained with the current parameter combinations and under conditions where all outputs are evaluated as equally important. However, the optimization module of the Design expert program was used to obtain the optimum result with fewer experiments without trying all possible factor combinations, which is the purpose of using the experimental design.

When the optimization module is run and the analysis is made by selecting AP50, AP75, Av.Recall, F1/50, F1/75 for the maximum, L/TL for the minimum and F1/50 and F1/75 for the highest (5) importance, AP50, AP75, Av.Recall and L/TL for the lowest importance (1), the program estimates that the most ideal solution will occur when the parameters of the SSD Mobilnet V1 algorithm are BS=2 and Run=250 thousand. This situation supports the choice of SSD Mobilnet V1, which also gave good results in the ranking findings.

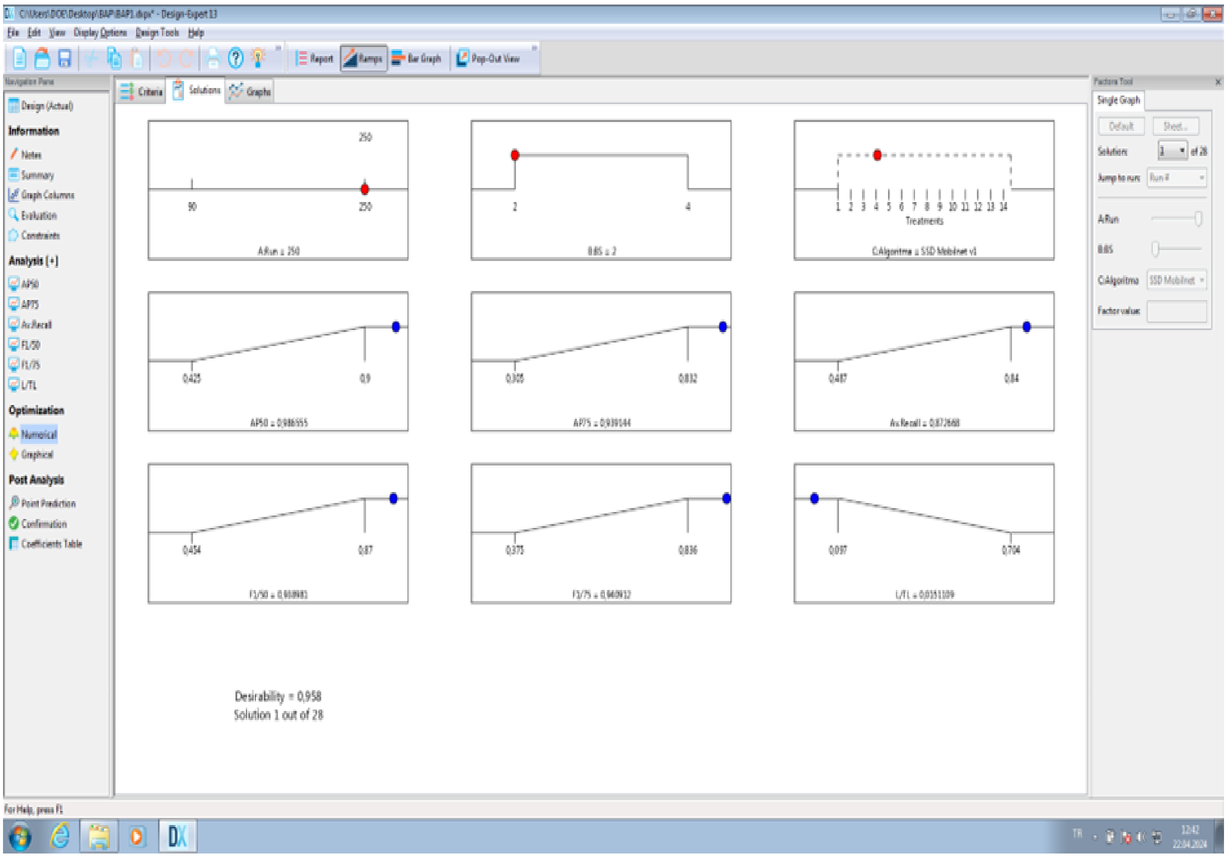

Figure 7. Design Expert Program Optimization Result

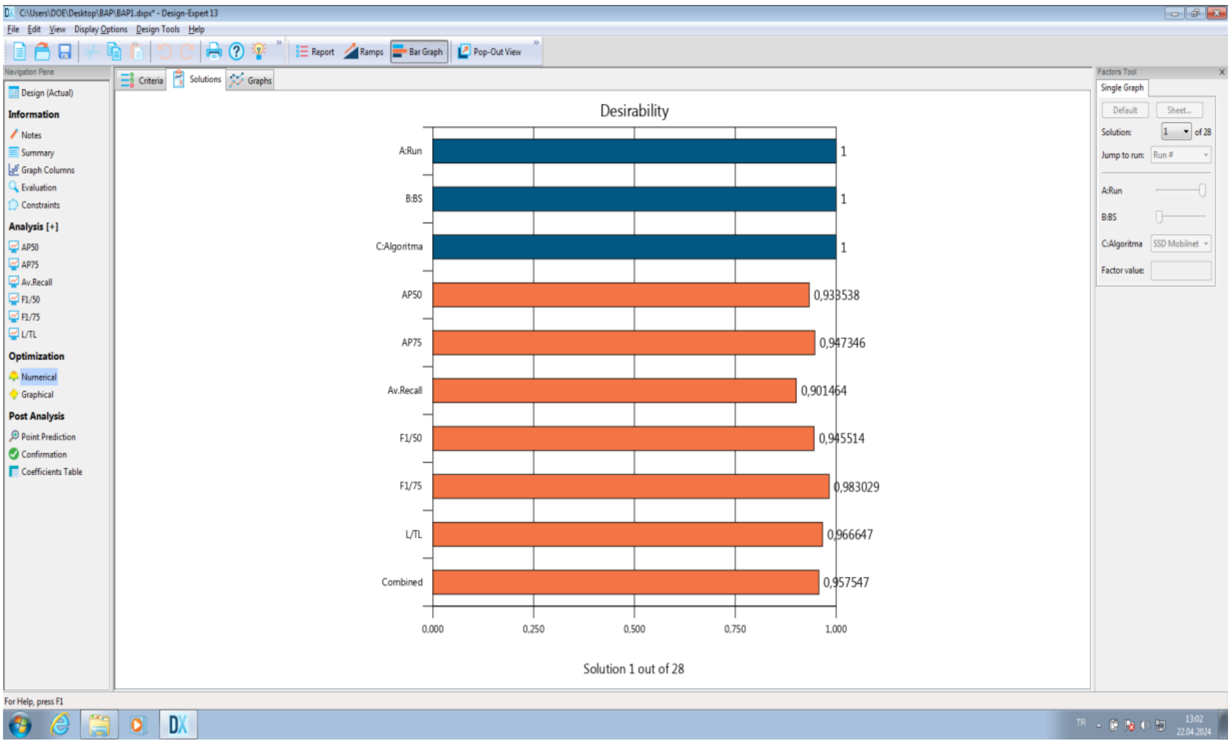

Figure 8. Desirability Levels of Optimization Results for SSD Mobilnet V1, BS=2, Run=250 Thousand in Terms of Output

IV. CONCLUSION

Object detection plays an important role in all autonomous land, sea and air vehicles. Image processing models are the cornerstone of providing autonomous control in vehicles. To test image processing models more successfully, a comprehensive data set plays an important role. There is no comprehensive data set for marine vehicles in the literature. For this reason, our study will contribute to literature. Our study was tested on the most common real-time recognition models, SSD, Faster R-CNN, EfficientDet algorithms under the TensorFlow library. As a result of the tested models, the most successful algorithm in 250 thousand steps was SSD Mobilnet v1, followed by F RCNN Resnet101 v1 and F RCNN Resnet50 v1 algorithms, respectively. Although our study was tested on traditional image processing models, the contribution of our study to the literature is that a comprehensive data set about marine vehicles has been added to the literature. In our future studies, we plan to increase the data set and optimize the SSD Mobilnet v1 to perform better on this dataset so that it can be used for autonomous device to drive safely with real-time data without disrupting maritime traffic.

ACKNOWLEDGEMENTS: This study was supported by Doğuş University Scientific Research Projects. (Project No: 2019-20-D2-B03)

V. REFERENCES

[1] M. S. Bingöl, Ç. Kaymak, and A. Uçar, "Derin öğrenme kullanarak otonom araçların insan sürüşünden öğrenmesi," *Fırat Üniversitesi Mühendislik Bilimleri Dergisi*, vol. 31, no. 1, pp. 177-185, 2019.

[2] E. Frazzoli, M. A. Dahleh, and E. Feron, "Real-time motion planning for agile autonomous vehicles," *Journal of Guidance, Control, and Dynamics*, vol. 25, no. 1, pp. 116-129, 2002.

[3] P. Zhao, J. Chen, T. Mei, and H. Liang, "Dynamic motion planning for autonomous vehicle in unknown environments," in *2011 IEEE Intelligent Vehicles Symposium (IV)*, 2011, pp. 284-289.

[4] Z. Tan and M. Karaköse, "Dinamik ortamlarda derin takviyeli öğrenme tabanlı otonom yol planlama yaklaşımları için karşılaştırmalı analiz," *Adıyaman Üniversitesi Mühendislik Bilimleri Dergisi*, vol. 9, no. 16, pp. 248-262, 2022.

[5] R. Girshick, "Fast R-CNN," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 1440-1448.

[6] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," *arXiv preprint* arXiv:1804.02767, 2018.

[7] W. S. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity," *The Bulletin of Mathematical Biophysics*, vol. 5, no. 4, pp. 115-133, 1943.

[8] I. Goodfellow, *Deep learning*, MIT Press, 2016.

[9] N. Aalami, “Derin öğrenme yöntemlerini kullanarak görüntülerin analizi,” *Eskişehir Türk Dünyası Uygulama ve Araştırma Merkezi Bilişim Dergisi*, vol. 1, no. 1, pp. 17-20, 2020.

[10] R. Kiros, R. Salakhutdinov, and R. S. Zemel, “Unifying visual-semantic embeddings with multimodal neural language models,” *arXiv preprint* arXiv:1411.2539, 2014.

[11] M. Bojarski, “End to end learning for self-driving cars,” *arXiv preprint* arXiv:1604.07316, 2016.

[12] S. Çelik and A. Altınörs, “Environmental waste detection from UAV images with YOLOv3 deep learning algorithm,” *unpublished*.

[13] J. Redmon and A. Farhadi, “YOLOv3: An incremental improvement,” *arXiv preprint* arXiv:1804.02767, 2018.

[14] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Y. Fu, and A. C. Berg, “SSD: Single shot multibox detector,” in *European Conference on Computer Vision (ECCV)*, 2016, pp. 21-37.

[15] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards real-time object detection with region proposal networks,” in *Advances in Neural Information Processing Systems (NIPS)*, 2015, pp. 91-99.

[16] M. Tan, R. Pang, and Q. V. Le, “EfficientDet: Scalable and efficient object detection,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 10781-10790.

[17] M. J. Jung, H. Myung, S. G. Hong, D. R. Park, H. K. Lee, and S. Bang, “Structured light 2D range finder for simultaneous localization and map-building (SLAM) in home environments,” in *Micro-Nanomechatronics and Human Science, 2004 and The Fourth Symposium Micro-Nanomechatronics for Information-Based Society*, 2004, pp. 371-376.

[18] M. N. Demir and Y. Altun, “Otonom araçla genetik algoritma kullanılarak haritalama ve lokasyon,” *Düzce Üniversitesi Bilim ve Teknoloji Dergisi*, vol. 8, no. 1, pp. 654-666.

[19] D. A. Ş. Resul, B. Polat, and G. Tuna, “Derin öğrenme ile resim ve videolarda nesnelerin tanınması ve takibi,” *Fırat Üniversitesi Mühendislik Bilimleri Dergisi*, vol. 31, no. 2, pp. 571-581, 2019.

[20] E. Özbaysar and E. Borandağ, “Vehicle plate tracking system,” in *2018 26th Signal Processing and Communications Applications Conference (SIU)*, 2018, pp. 1-4.

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

[22] D. K. Prasad, H. Dong, D. Rajan, and C. Quek, "Are object detection assessment criteria ready for maritime computer vision?," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 12, pp. 5295-5304, 2019.

[23] G. Margarit and A. Tabasco, "Ship classification in single-pol SAR images based on fuzzy logic," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 49, pp. 3129–3138, 2011.

[24] M. Leclerc, R. Tharmarasa, M. C. Florea, A. C. Boury-Brisset, T. Kirubarajan, and N. Duclos-Hindié, "Ship classification using deep learning techniques for maritime target tracking," in *Proceedings of the 21st International Conference on Information Fusion*, Cambridge, UK, Jul. 10–13, 2018, pp. 737–744.

[25] International Maritime Organization, *Convention on the International Regulations for Preventing Collisions at Sea, 1972 (Consolidated edition)*, International Maritime Organization, 2018.