Turkish Journal of Science & Technology
20(1), 269-279, 2025

Research Paper
https://doi.org/10.55525/tjst.1603517

# The Effect of Virtual Reality Images on Artificial Intelligence Classification of Real Environment Images

**Nur ÖZBEK[1*], Berna GÜRLER ARI[2], Sevgi ÖZTORUN[3], Ömer Faruk ALÇİN[4]**

[1] Software Engineering, Faculty of Technology, Elâzığ, Turkiye
[2]Computer Engineering, National Defense University, Ankara, Turkiye
[3]English Language Teaching, Turkiyem Secondary School, Malatya, Turkiye
[4]Software Engineering, Faculty of Engineering, İnönü University, Malatya, Turkiye

[*1]nurozbek97@gmail.com, [2]berna.gurlerari@msu.edu.tr, [3]oztorun84@gmail.com, [4]omer.alcin@inonu.edu.tr

**(Geliş/Received: 18/12/2024; Kabul/Accepted: 25/02/2025)**

**Abstract:** Since virtual reality (VR) is a new and current field of study, it is intensively studied by researchers. Health, education, engineering, culture and tourism, architecture, military fields and many other fields of study have taken steps to support their studies with VR technology and the related subject has become the focus of many researchers. In this thesis, VR technology was used to improve the classification performance of real media images. The proposed approach consists of the classification of real environment images with transfer learning. The transfer learning mentioned in the thesis study can be defined as training an untrained deep architecture with VR images and then retraining the network with real images (fine-tuning). For this purpose, VR scenes are designed in the UNITY environment. A dataset consisting of 15 environments, called V-Env15, was prepared from the designed VR scenes. The proposed approach was tested with the Scene-15 dataset, which is frequently used in environment classification studies. In the thesis, serial and parallel network and GoogLeNet and Inception-ResNet-V2 deep learning architectures were used. In the experimental studies, 0.56% higher accuracy performance increase was achieved in the serial architecture and 4.68% higher accuracy performance increase in parallel architecture. A 4.79% accuracy performance increase was achieved between GoogLeNet and the Serial network, and a 0.44% decrease was achieved between the Parallel network. A 4.47% increase was achieved between Inception-ResNet-V2 and the Serial network, and a 4.57% decrease was achieved between the Parallel network.

**Key words:** Virtual Reality, Machine Learning, Image Processing, Environment Classification.

# Gerçek Ortam Görüntülerinin Yapay Zekâ ile Sınıflandırılmasında Sanal Gerçeklik Görüntülerinin Etkisi

**Öz:** Sanal gerçeklik (SG) yeni ve güncel bir çalışma alanı olduğundan araştırmacılar tarafından yoğun şekilde çalışılmaktadır. Sağlık, eğitim, mühendislik, kültür ve turizm, mimari, askeri alanlar ve daha birçok çalışma alanı SG teknolojisi ile çalışmalarını destekleyici adımlar atmış ve ilgili konu birçok araştırmacının odağı haline gelmiştir. Bu çalışmada gerçek ortam görüntülerinin sınıflandırma başarımını artırmak için SG teknolojisinden yararlanılmıştır. Önerilen yaklaşım transfer öğrenme ile gerçek ortam görüntülerinin sınıflandırılması işleminden oluşmaktadır. İlgili çalışmada bahsedilen transfer öğrenme, eğitilmemiş bir derin mimarinin SG görüntüleri ile eğitilmesi ardından ağın gerçek görüntülerle yeniden eğitimi (fine-tuning) olarak tanımlanır. UNITY ortamında tasarlanan SG sahnelerinden V-Env15 olarak isimlendirilen ve 15 ortamdan oluşan bir veri seti hazırlanmıştır. Ortam sınıflama çalışmalarında sıklıkla kullanılan Scene-15 veri seti ile önerilen yaklaşım test edilmiştir. Çalışmada tasarlanan Seri ve Paralel ağ ile GoogLeNet ve Inception-ResNet-V2 derin öğrenme mimarileri kullanılmıştır. Deneysel çalışmalarda tasarladığımız seri mimaride %0,56 ve paralel mimaride ise %4,68 daha yüksek doğruluk performans artışı elde edilmiştir. GoogLeNet ile Seri ağ arasında performasn doğruluğu açısından %4,79 artış, Paralel ağ arasında %0,44 azalma elde edilmiştir. Inception-ResNet-V2 ile Seri ağ arasında %4,47 artış, Paralel ağ arasında %4,57 azalma elde edilmiştir.

**Anahtar kelimeler:** Sanal Gerçeklik, Makine Öğrenmesi, Görüntü İşleme, Ortam Sınıflandırma.

## 1. Introductıon

Recently, the concept of "Virtual Reality (VR)" has been widely researched. VR concept consists of software and hardware components. In the hardware component, the VR device to be used varies according to the purpose of the

* Corresponding author: nurozbek97@gmail.com ORCID Number of authors: [1]0009-0000-2958-3172, [2]0000-0003-1000-2619,[3]0009-0002-0274-8648, [4]0000-0002-2917-3736

research and the existing infrastructure. Thanks to the sensors whose development has accelerated with the advancement of technology, a large number and variety of data such as audio or image are obtained. Due to the variety of objects in the scenes, there is a deficiency in making sense of scenes with complex backgrounds. As a result of the studies in the field of image processing, scene classification has emerged with the semantic classification of images. Semantic inferences made on the image differ from person to person. Therefore, scene classification is seen as an important problem in image processing. Virtual Reality (VR) technology is widely used in many fields due to the flexibility it provides in the data collection process and its advantage of offering a controlled environment. However, datasets collected from the real world are often costly, time-consuming, and subject to variability due to various environmental factors. This makes it particularly challenging to create large-scale datasets required for training deep learning models. There is a noticeable difference between the low-level features extracted from images and the high-level semantic inferences made by users from images. For this reason, studies on image and video datasets, the number of which increases over time, play an important role in the correct classification of images in terms of semantic information extraction [1,2]. In object detection studies, one can start from images that carry semantic information in scene classification [3]. For example, when identifying the tree object, it is necessary to first examine the forest scene images. Because the number and type of trees in each forest scene should be taught to the classification algorithm. In parallel with the increase in the performance of computer graphics cards in recent years, Deep Neural Networks (DNN) are used to process these data in a shorter time. In the ImageNet competition held in 2012, Krizhevsky et al.'s AlexNet Deep Convolutional Artificial Neural Networks (AlexNet Deep Convolutional Artificial Neural Networks (DPNN) classification performance result increased the use of deep learning methods in the field of artificial intelligence [4]. Deep learning has gained popularity in the field of computer vision and it has been observed that deep learning is used in most of the scientific publications. The topics of these publications are segmentation, object detection, classification and scene classification [5]. Segmentation is the separation of different features in an image into meaningful regions; object detection is the determination of the location and characteristics of each object in an image. Classification is the process of separating each object in an image according to its visual characteristics, while scene classification is the process of analyzing and making sense of the objects in the image as a whole. The main difference between classification and scene classification is that in classification, the objects in an image are analyzed separately, while in scene classification, the image is considered as a whole and the meaning is made in its entirety [6].

In a study by Szummer and Picard [7], they investigated how to infer high-resolution images from the classification of low-resolution images based on the problem of determining whether the environment is indoor or outdoor. In the study, 19 of the 1343 images in the dataset created by Kodak were removed from the dataset due to their ambiguity. Three types of attributes were used in the study: color, frequency and texture. While 75-86% success was achieved in studies using the same dataset, 90.30% success was achieved with the method used in the related study.

In their study [8], Fei-Fei and Perona proposed a new approach for classifying categories of specific areas. They used a dataset with 13 classes: city (308), highway (260), street (292), building (356), residential (241), beach (360), forest (328), mountain (374), bedroom (174), countryside (410), kitchen (151), living room (289) and office (216). Compared to other studies, the researchers emphasized the fact that their work learned the intermediate themes in scenes without any supervision or human intervention. In the Bayesian hierarchical model designed for this purpose, probabilistic frames were first used to learn texture models using code words in the algorithm. Each of the images has an average size of 250×300 pixels and only black and white images were used in the training and testing process. In the complexity matrix used to obtain the performance of the model, the models belonging to the scene category and the basic reality categories of the scenes were compared. The applied method achieved 64% success rate. The classes where the method gave the most errors were bed and living room, kitchen and office. The reason for the method's errors is that these classes have walls and sharp vertical and horizontal edges.

In their study [9], Vailaya et al. aimed to classify holiday image scenes hierarchically. In order to extract features from low-level images based on global features, the scenes were first classified as indoor and outdoor using binary Bayes classifiers. Then, outdoor images were classified as city and landscape, and the landscape was hierarchically classified as sunset, forest, mountain and other natural environments. From 6931 scene images, the classification accuracy was 90.50% for indoor/outdoor, 95.30% for city/landscape, 96.60% for sunset/forest and mountain, and 96% for forest/mountain problems.

In their study [10], Bird et al. aimed to classify the beginning and end scenes of scene images from both virtual and real-world environments with deep learning models. They trained images belonging to six classes on a finely tuned VGG16 on the datasets they created. Transfer Learning (TL) networks trained on virtual data were then compared with real-world data. As a result of the study, it was observed that all transfer learning networks had a higher initial pre-training compared to the others, even showing an average increase of 38.33% in all compared hyperparameter sets. There are 6 class labels in this study. Due to the small number of class labels and the presence of images (doors, walls,

roads, etc.) in the dataset that may be common in many environments, the training error amount is high, which is seen as a gap in the literature.

Herranz et al. [11] emphasized the importance of accurate scene recognition, but also emphasized that this cannot be done only with scene recognition, but also with object recognition. How to effectively combine scene-centered and object-centered knowledge has been the main topic of this research. The scene and object classification on the 397-category SUN397 dataset was performed with CNN architectures. By choosing ImageNet-CNN and Places-CNN combinations, 70.17% success was achieved.

Thanks to the success of studies in the field of pattern recognition, the use of ESA-based architectures has become quite common [20, 21].

The aim of this study is to improve the effect of virtual images prepared using VR technology on the classification performance of real environment images. A dataset containing virtual environment images was created to be used in training the network while classifying real environment images.

## 2. Material and Method

In this section, information about the deep learning methods used in the study, the datasets, and the environment in which the dataset was prepared are given.

### 2.1. Convolutional Neural Network (CNN)

In computer vision, feature extraction and classification has been a very common field of study. CNN is a deep feed-forward artificial neural network that performs well in analyzing images. CNN is often used in image processing [12]. CNN architectures consist of input, convolution, pooling, smoothing, fully connected and classification layers. Each of these layers in the architecture performs different tasks and are connected to each other sequentially. While features are extracted in the input, convolution and pooling layers, classification processing is performed in the smoothing, fully connected and classification layers. In CNN architectures, data is divided into parts and a filter is applied to each part. Depending on the size of the filter applied, the image shrinks. When a 16×16 filter is applied to a 16×16 area of the image, the result of this process is reduced to a single pixel. The resulting 1×1 sized output is used for identification [13].

### 2.2. GoogLeNet

GoogLeNet is a deep learning model with a complex structure consisting of 22 layers and Inception modules. It won the 2014 ImageNet competition with an error rate of 5.70%. The images in the input layer are 224×224×3 [14]. The GoogLeNet architecture does not stack convolution and pooling layers consecutively. This optimizes memory and power usage while minimizing the risk of model memorization. This is because stacking layers and using a large number of filters incurs computational and memory costs. In the GoogLeNet architecture, parallel interconnected modules are used to reduce this cost [15]. The working principle of the GoogLeNet architecture is given in Figure 1.

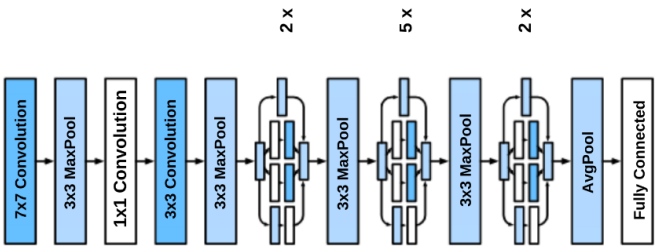


**Figure 1.** GoogLeNet Architecture.

### 2.2. Inception-ResNet-V2

The Inception-ResNet-V2 architecture is used in different fields such as image recognition, natural language processing and computer vision. The Inception-ResNet-V2 architecture consists of two main components, Inception and ResNet blocks. Inception blocks combine filters of different sizes to extract more features from the input image. ResNet blocks enable the learning of deeper networks compared to other networks. In this way, it helps in the extraction of

complex features. ResNet blocks use jump connections that add data from the previous layer to the next layer. Figure 2 shows the Inception-ResNetV2 architecture [16].

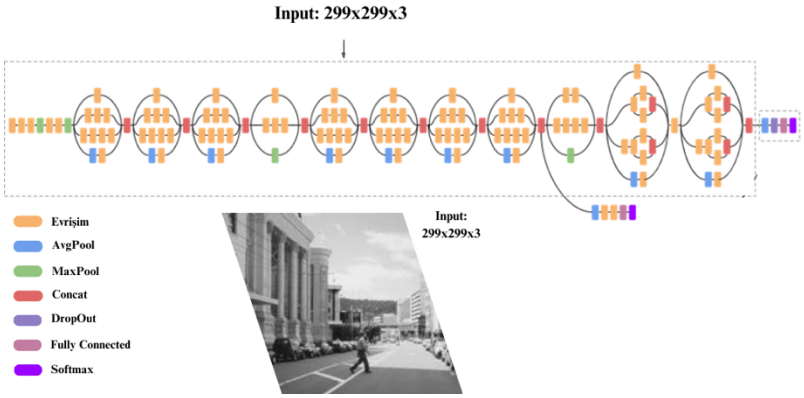


**Figure 2.** Inception-ResNet-V2 Architecture.

### 2.3. Serial Network

Serial networks are a very popular type of algorithm in image classification problems due to their simple structure and high accuracy. In this study, an 11-layer Serial network architecture is designed and adapted to image classification. Images of size 80×80×3 were used as the input data of the images in the dataset. The classification, probability and fully connected layers in the last 3 layers of the designed Serial network were removed. After the network trained with V-Env15 prepared within the scope of the study, instead of these 3 layers, the last 3 layers of the trained network, which are fully connected, probability and classification layers, were added. The purpose of this process is fine tuning. Figure 3 shows the Serial network architecture.

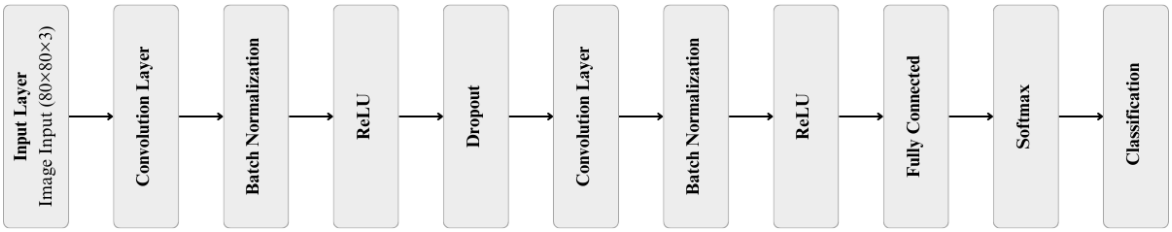


**Figure 3.** Designed Serial Network Architecture.

### 2.4. Parallel Network

Parallel network architecture can perform multiple tasks simultaneously. In image classification studies, Parallel networks are used to analyze images in the dataset faster and classify them more accurately. In parallel network approaches, images are divided into parts and each part is analyzed separately on the processor. This approach allows for faster analysis. Because each segmented part is processed in parallel [16]. In addition, Parallel networks provide more feature extraction by taking into account many different aspects of the image and accurate classification is performed. In the parallel network architecture, the last 3 layers of classification, probability and fully connected layers are removed. After the network trained with V-Env15, these 3 layers were replaced with fully connected, probability and classification layers respectively in parallel. This was done for fine tuning. Figure 4 shows the parallel network architecture.

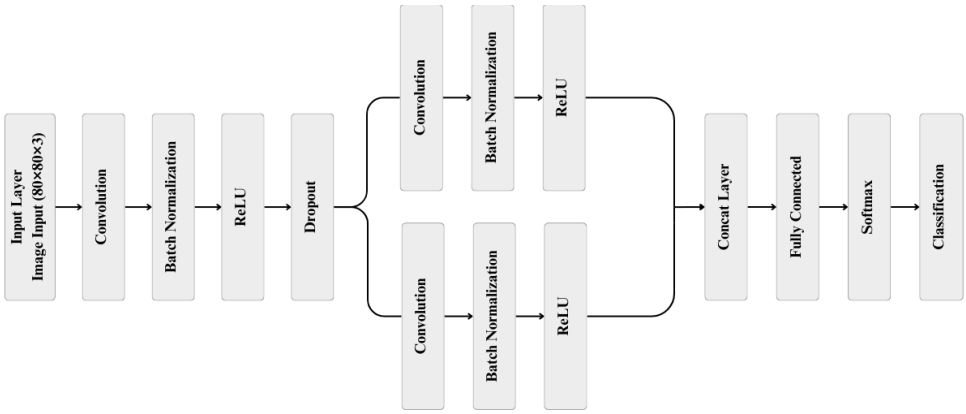


**Figure 4.** Designed Parallel Network Architecture.

### 2.5. Dataset

In this study, a transfer learning based approach for image classification is proposed. The dataset consists of real-world and virtual images created in a computer environment using Unity editor. VR environment images are used for training purposes and real environment images are used for testing purposes. In the proposed approach, images obtained from environments created in Unity editor using free assets are used for virtual images and Scene-15 dataset is used for real images.

#### 2.5.1. Scene-15

The images in the Scene-15 dataset, which contains the real images used in the testing process, were collected through Corel and Google images. The first 8 classes in the Scene-15 dataset were added by Oliva and Torralba [17], the next 5 classes by Fei-Fei and Perona [8] and the last 2 classes by Lazebnik et al [18]. There are 15 classes in the dataset: apartment, bedroom, factory, forest, house, kitchen, land, living room, mountain, office, road, sea, market, market, street and skyscraper. The number of images in the classes of the dataset is given in Table 1.

**Table 1.** Scene-15 dataset image distribution.

| | Class Label | Number of Data |
|---|---|---|
| **Outdoor** | Apartment | 308 |
| | Factory | 311 |
| | Forest | 328 |
| | House | 241 |
| | Land | 410 |
| | Mountain | 374 |
| | Path | 260 |
| | Sea | 360 |
| | Street | 292 |
| | Skyscraper | 356 |
| **Indoor** | Bedroom | 216 |
| | Kitchen | 210 |
| | Sitting room | 289 |
| | Office | 215 |
| | Market | 315 |
| **Total** | | **4485** |

### 2.5.2. V-Env15

The Scene-15 dataset, which contains indoor and outdoor scene images, has been the subject of research in many scientific studies for image classification [8,17,18]. A dataset with the same class labels as the Scene-15 dataset, where virtual images can be used for training, has not been found in the literature. Within the scope of this study, it was aimed to use real and virtual images in the same study by creating labels parallel to these environment labels and there was a need to create a dataset for virtual images. In order to fill the gap in the literature and make a contribution, a new dataset called V-Env15 was created within the scope of the study. The virtual images needed were created with the following steps: In the 3D project created through the Unity editor, the Meta XR Interaction SDK library was included in the project and environments where Oculus Quest 2 can run were prepared. Afterwards, free and publicly available objects from the Unity AssetStore were added to the project. Environments were designed for the class labels created with reference to the Scene-15 dataset. 3D objects, each taken from separate packages, were selected and added to create the relevant scenes in the V-Env15 dataset. The scenes created in Unity were tested on Oculus Quest 2 and video recordings were taken. The video recording was recorded at 1º angles and the images on the screen at each angle change were taken frame-by-frame through the MATLAB program. This process was repeated for each class in the V-Env15 dataset and images were created. As a result of the operations, a total of 40033 images were obtained. In the created dataset, images that are not suitable for use in the research and that may be common to each class label (such as walls, road lines) were excluded. As a result of the excluded images, each class consists of 1000 images. The V-Env15 dataset created for virtual images contains a total of 15,000 images. Table 2 shows the distribution of the media labels of the dataset. Sample images of the V-Env15 dataset are given in Figure 5.

**Table 2.** V-Env15 dataset image distribution.

| | Class Label | Number of Data |
|---|---|---|
| **Outdoor** | Apartment | 1000 |
| | Factory | 1000 |
| | Forest | 1000 |
| | House | 1000 |
| | Land | 1000 |
| | Mountain | 1000 |
| | Path | 1000 |
| | Sea | 1000 |
| | Street | 1000 |
| | Skyscraper | 1000 |
| **Indoor** | Bedroom | 1000 |
| | Kitchen | 1000 |
| | Sitting room | 1000 |
| | Office | 1000 |
| | Market | 1000 |
| **Total** | | **15000** |

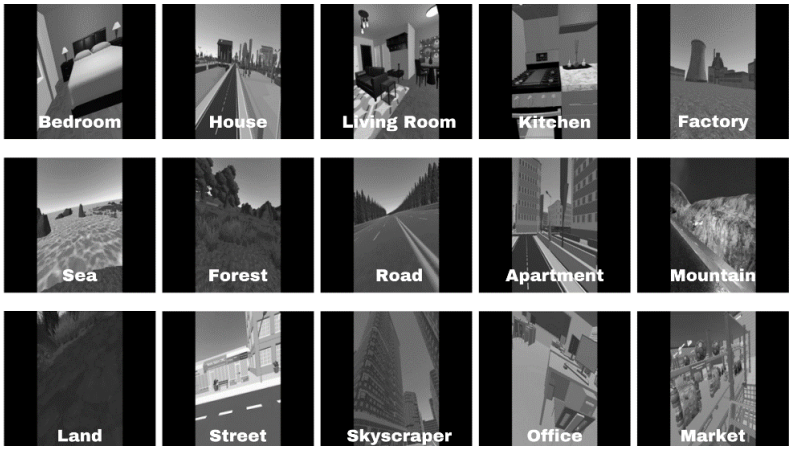


**Figure 5.** Sample images of the V-Env15 dataset.

### 2.6. Classification Performance Metrics

Some measurement criteria, such as the complexity matrix, were used to determine the performance of the methods after the classification process. Figure 6 shows a representation of the complexity matrix.

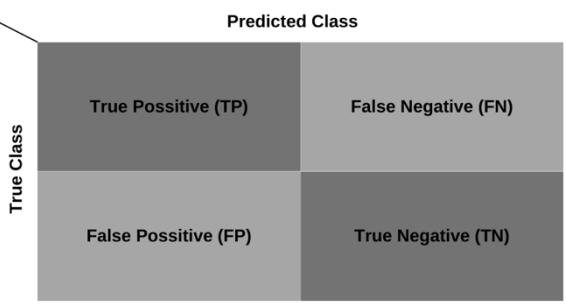


**Figure 6.** Confusion Matrix.

True Positive (TP): Correctly predicting the predicted class value as class A when the actual class value should be positive A

False Negative (FN): Incorrectly predicting the predicted class value as class A when the actual class value should be negative B

False Positive (FP): The predicted class value is incorrectly predicted as class B when the actual class value should be positive A

True Negative (TN): Correctly predicting the predicted class value as class B when the actual class value should be negative B

When the values in the complexity matrix are analyzed; TP indicates the correct classification value, TN indicates the correct classification of the values in the other class. FP refers to the cases where the values that should actually be in another class are in the relevant class, and FN refers to the cases where the values that should be in the relevant class are in another class [19]. In this study, accuracy, sensitivity, specificity, precision, F1 Score, Mathew Correlation Coefficient (MCC) and Kappa were used. Accuracy is the measurement criterion that shows how much of the predictions of the model are correct. Sensitivity is the measurement criterion that shows how accurately the model classifies positive samples as positive. Specificity is the criterion that shows how much of the negative samples the model classifies as negative. Precision measures how many of the samples predicted as positive are actually positive. The F1 Score is derived from the sensitivity and precision criteria. Mathew Correlation Coefficient (MCC) Accuracy is measured by sensitivity and specificity. Kappa value measures the relationship between different classifier methods. The closer the Kappa value is to 1, the higher the correlation result. The equations for these measurement criteria are given below in Equations 1-6:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{1}$$

$$Sensivity = \frac{TP}{TP+FN} \tag{2}$$

$$Specificity = \frac{TN}{TN+FP} \tag{3}$$

$$F1\text{-}Score = 2 \times \frac{\text{Sensitivity} \times \text{Precision}}{\text{Sensitivity} + \text{Precision}} \tag{4}$$

$$\text{MCC} = \frac{TP \times TN - FP \times FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} \tag{5}$$

$$Kappa = 2 \times \frac{TP \times TN - FP \times FN}{(TP \times FN + TP \times FP + 2 \times TP \times TN + FN \times TN + FP \times FP + FP \times TP)} \tag{6}$$

## 3. Experimental Design

The aim of this study is to classify the images in Scene-15 and V-Env15 datasets as belonging to the indoor or outdoor environment. In this study, two different datasets, Scene-15, which contains real images, and V-Env15, which is prepared by utilizing VR technology within the scope of the study used for transfer learning, were used. There are 4485 images in Scene-15 dataset and 15,000 images in V-Env15 dataset. Of the 15 categories in the datasets, 5 classes consist of indoor environment and 10 classes consist of outdoor environment. All images used were first converted to gray level in MATLAB environment. For the implementation of the method in the study, a computer with Intel (R), @2.30 GHz processor, 6 GB Graphics card and 32 GB RAM was used. In the experimental studies, the hyperparameters of the deep network architectures were obtained empirically. The results section is given under 3 headings for better explanation.

### 3.1. Serial Network

In the architecture created as a serial network, the Scene-15 dataset was first classified as belonging to the indoor/outdoor environment. In the experimental studies, the dataset was divided into three parts: 70% training, 10% validation and 20% testing. In the experiments, the maximum number of repetitions (Epoch) was fixed as 5 and the optimization method was determined empirically as SGDM (Stochastic Gradient Descent with Momentum). The training time of the serial network was completed in 70 seconds. In Table 3, the Scene-15 row shows the results of applying the Scene-15 dataset to the serial network, the V-Env15 row shows the results of applying the prepared V-Env15 dataset to the serial network, and the TL row shows the results of transfer learning for Scene-15 of the serial network trained with the V-Env15 dataset. This notation is also used in the following subsections.

**Table 3.** Binary classification results of the architecture created with Serial Network (%).

| | Accuracy | Sensivity | Specificity | Precision | F1 | MCC | Kappa |
|---|---|---|---|---|---|---|---|
| Scene-15 | 95.09 | 89.58 | 97.34 | 93.17 | 91.34 | 87.95 | 87.92 |
| V-Env15 | 93.20 | 88.72 | 95.54 | 91.20 | 89.94 | 84.82 | 84.81 |
| TL | 95.65 | 89.47 | 98.26 | 95.58 | 92.43 | 89.48 | 89.38 |
| $\Delta F$ | 0.56 | -0.11 | 0.92 | 2.41 | 1.09 | 1.53 | 1.46 |

$\Delta \boldsymbol{F}$ shows the difference in the classification performance of the Scene-15 dataset as a result of TL of the Serial network trained with V-Env15. $\Delta \boldsymbol{F}$ will be used to represent the difference in all experimental studies conducted during the study. Table 3 shows that Accuracy improved by 0.56%, Specificity improved by 0.92%, Precision improved by 2.41%, F1 improved by 1.09%, MCC improved by 1.53% and Kappa improved by 1.46%, while Sensitivity improved by 0.11%.

### 3.2. Parallel Network

In the architecture created as a parallel network, the Scene-15 dataset was first classified as belonging to the indoor/outdoor environment. In the experimental studies, the dataset was divided into three parts: 70% training, 10% validation and 20% testing. In the experiments, the maximum number of repetitions (Epoch) was fixed as 5 and the optimization method was determined empirically as SGDM. The training time of the parallel network was completed in 80 seconds.

**Table 4.** Binary classification results of the architecture created with Parallel Network (%).

| | Accuracy | Sensivity | Specificity | Precision | F1 | MCC | Kappa |
|---|---|---|---|---|---|---|---|
| Scene-15 | 77.93 | 61.54 | 83.28 | 56.62 | 57.87 | 43.13 | 43.00 |
| V-Env15 | 87.13 | 74.00 | 93.70 | 85.45 | 79.31 | 70.43 | 70.05 |
| TL | 82.61 | 68.53 | 88.08 | 69.08 | 68.80 | 56.75 | 56.74 |
| $\Delta \boldsymbol{F_1}$ | 4.68 | 6.99 | 4.80 | 12.46 | 10.93 | 13.62 | 13.74 |

The performance performance resulting from TL realized with the parallel network is given in Table 4. ΔF1 shows the difference in the classification performance of the Scene-15 dataset as a result of TL with V-Env15. When Table 4 is examined, an improvement of 4.68% in Accuracy, 6.99% in Sensitivity, 4.80% in Specificity, 12.46% in Precision, 10.93% in F1, 13.62% in MCC and 13.74% in Kappa.

### 3.3. Commonly Used Networks

The classification process was repeated with GoogLeNet with 224×224×3 dimensions and Inception with 299×299×3 dimensions. In the experimental studies, the dataset was divided into three parts as 70% training, 10% validation and 20% test. In the experiments, the maximum number of repetitions (Epoch) was fixed as 5 and the optimization method man (ADAptive Moment) was obtained empirically. Table 5 shows the performance performance and increase rates of the TL resulted from the GoogLeNet architecture and Serial and Parallel network architectures. In Table 5, $\Delta F_2$ represents the difference between Serial network and GoogLeNet and $\Delta F_3$ represents the difference between Parallel network and GoogLeNet.

**Table 5.** Binary classification between GoogLeNet architecture (%).

| | Serial Network 80×80×3 | Parallel Network 80×80×3 | Googlenet 224×224×3 | $\Delta F_2$ | $\Delta F_3$ |
|---|---|---|---|---|---|
| TL | 95.65 | 82.61 | 83.05 | +4.79 | -0.44 |

Table 6 shows the performance results of Inception-ResNet-V2 with the Serial and Parallel network architectures and the increase rates. In Table 6, $\Delta F_4$ represents the difference between Serial network and Inception-ResNet-V2, $\Delta F_5$ represents the difference between Parallel network and Inception-ResNet-V2.

**Table 6.** Binary classification difference between Inception-Resnet-V2 architecture (%).

| | Serial Network 80×80×3 | Parallel Network 80×80×3 | Inception-ResNet-V2 299×299×3 | $\Delta F_4$ | $\Delta F_5$ |
|---|---|---|---|---|---|
| TL | 95.65 | 82.61 | 87.18 | +8.47 | -4.57 |

## 4. Results and Recommendations

In this study, the performance of VR technology in classifying real environment images is investigated. For this purpose, a scene with 15 classes was designed in the Unity editor by taking the Scene-15 dataset as an example. A VR dataset called V-Env15 was prepared by moving 360º at 1º angles from the designed environments. From a network trained using the V-Env-15 dataset, we focused on increasing the classification performance of the environment images in the dataset containing real images called Scene-15 by performing TL. Within the scope of the study, two CNN architectures with serial and parallel structures were designed. First, the Scene-15 dataset was applied to the Serial network and the highest classification accuracy of 95.09% was obtained. Then, after training the Serial network with the V-Env15 dataset, TL was applied and an accuracy of 95.65% was achieved in the environment classification of the Scene-15 dataset. Thus, by applying TL, 0.56% higher accuracy was achieved in the Serial network. Scene-15 dataset was applied to the Parallel network architecture and 77.93% accuracy was achieved in the experiment. Then, by applying TL to the Parallel network architecture trained with the V-Env15 dataset, 82.61% accuracy was achieved in the environment classification of the Scene-15 dataset. TL on the parallel network resulted in an accuracy improvement of 4.68%. Studies were continued with GoogLeNet and Inception-ResNet-V2 architectures, which are widely used in TL applications in image classification. The Scene-15 dataset achieved 83.05% classification accuracy by performing TL with GoogLeNet. Inception-ResNet-V2 network classified the Scene-15 dataset with 87.18% accuracy with TL. In the experiments, binary classification as indoor/outdoor environment was performed on the images in the datasets. In binary classification studies, the highest performance was obtained in the experimental study with the Serial network. Compared to other classification studies, the classification study with Parallel network showed lower performance compared to other architectures.

Experiments were conducted for the case with 15 classes. Classification accuracies for this case are presented in Table 7. When Table 7 is examined, the optimization method rmsprop and the learning rate were changed to 1e-3 for the parameters used in the internal/external classification process in the trained Serial network. SGDM optimization and 1e-4 learning rate in the network trained with parallel network; rmsprop optimization and 1e-3 learning rate in the

network trained with Inception-ResNet-V2 architecture; man optimization and 1e-3 learning rate in the network trained with GoogLeNet architecture.

**Table 7.** 15 class environment type accuracy results of Scene-15 dataset in different architectures (%).

| | Serial Network | Parallel Network | Inception-ResNet-V2 | GoogLeNet |
|---|---|---|---|---|
| TL | 83.17 | 60.00 | 75.81 | 82.50 |

In experimental studies with a 15-class dataset, the classification with the Serial network was 83.17% higher than the other architectures. The lowest classification performance was obtained from the Parallel network with 60% accuracy.

Another study in the literature on environment classification with VR images is given in Table 8. In the study of 6 classes, there are Living room, Staircase, Forest, Land, Computer lab and Bathroom classes. When Table 8 is analyzed, it is seen that the success rate in the study conducted by Bird et al. is 88.27%. However, this performance was obtained in the classification study conducted on only 6 classes. In the study conducted on the 15-class V-Env15 dataset designed within the scope of the study, 83.17% success was achieved.

**Table 8.** Study on classification with virtual images.

| **Ref. Name** | **Accuracy (%)** | **$\Delta F_1$ (%)** | **Class Number** | **Class Labels** |
|---|---|---|---|---|
| Paper [10] | 88.27 | 38.33 | 6 | Living Room, Staircase, Forest, Terrain, Computer Laboratory, Bathroom |
| **Proposed Method** | **83.17** | **5.36** | **15** | ***Apartment, Land, Street, Mountain, Sea, House, Factory, Skyscraper, Market, Kitchen, Office, Forest, Living Room, Bedroom, Road*** |

When the studies conducted within the scope of the scientific study and the literature are evaluated, the following points are suggested to be taken into consideration in order to increase the classification success in the future studies;

1. In the studies to be conducted with virtual reality images, it is evaluated to create a dataset that does not contain images that may be common in every environment (wall, road, door, chair, etc.) in the data to be created other than the datasets in the literature.
2. In addition to the class labels in the V-Env15 dataset created for use in this study and to contribute to the literature, it is recommended to add real-life environments to the dataset.
3. Serial network, Parallel network, GoogLeNet and Inception architectures were used on Scene-15 and V-Env15 datasets. It is thought that using more up-to-date and complex CNN architectures on the same datasets will contribute to the improvement of the results.
4. It is thought that it would be useful to perform different classification studies on the V-Env15 dataset with 15 class labels (apartment, bedroom, factory, forest, house, kitchen, land, living room, mountain, office, road, sea, market, street and skyscraper) and 40033 images.

Using Scene-15 and V-Env15 datasets, it is suggested to conduct studies on whether malicious computer generated images are real or virtual.

## Author Contribution Statement

In the study, Author 1 contributed to the idea, data set creation and methodological analysis; Authors 2 and 4 contributed to the design and analysis; Author 3 contributed to the literature review, spelling check; Authors 1, 2 and 4 contributed to the evaluation of the results, provision of the materials used and examination of the results.

## References


[1] Çavuş Ö. Semantic Scene Classification for Content-Based Image Retrieval. Doktora Tezi, Bilkent Üniversitesi, Türkiye, 2008.

[2] Karabulut A. Bayes Tabanlı Sahne Sınıflandırması. Yüksek Lisans Tezi, Hacettepe Üniversitesi, Türkiye, 2006.

[3] Torralba A, Murphy KP, Freeman WT, Rubin MA. Context-based vision system for place and object recognition. Proceedings of the IEEE International Conference on Computer Vision; Ekim 2003; IEEE Computer Society. s. 273-273.

[4] Meldrum D, Glennon A, Herdman S, Murray D, McConn-Walsh R. Virtual reality rehabilitation of balance: assessment of the usability of the Nintendo Wii® Fit Plus. Disabil Rehabil Assist Technol 2012; 7(3): 205-210.

[5] Song H, Chen F, Peng Q, Zhang J, Gu P. Improvement of user experience using virtual reality in open-architecture product design. Proc Inst Mech Eng B J Eng Manuf 2018; 232(13): 2264-2275.

[6] Boas YAGV. Overview of virtual reality technologies. Proceedings of the Interactive Multimedia Conference; Ağustos 2013.

[7] Szummer M, Picard RW. Indoor-outdoor image classification. Proceedings of the IEEE International Workshop on Content-based Access of Image and Video Databases; 1998.

[8] Fei-Fei L, Perona P. A Bayesian hierarchical model for learning natural scene categories. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; 2005; Cilt 2. s. 524-531.

[9] Vailaya A, Figueiredo MAT, Jain AK, Zhang HJ. Image classification for content-based indexing. IEEE Trans Image Process 2001; 10(1): 117–130.

[10] Bird JJ, Faria DR, Ekárt A, Ayrosa PP. From simulation to reality: CNN transfer learning for scene classification. Proceedings of the IEEE 10th International Conference on Intelligent Systems; Ağustos 2020. s. 619-625.

[11] Herranz L, Jiang S, Li X. Scene recognition with CNNs: objects, scales and dataset bias. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; 2016. s. 571–579.

[12] Karadağ B, Arı A, Karadağ M. Derin öğrenme modellerinin sinirsel stil aktarımı performanslarının karşılaştırılması. Politeknik Dergisi 2021; 24(4): 1611-1622.

[13] Arı B. Kayısı Yapraklarının Evrişimsel Sinir Ağları Kullanılarak Sınıflandırılması. Yüksek Lisans Tezi, Fen Bilimleri Enstitüsü, 2022.

[14] Arı A. Derin Öğrenme Tabanlı Beyin MR Görüntülerinden Beyin Tümörlerinin Tespit Edilmesi ve Sınıflandırılması. Doktora Tezi, İnönü Üniversitesi, Malatya, 2019.

[15] Özküçük M, Alçin ÖF, GENÇOĞLU M. EMG sinyalleri kullanılarak GoogLeNet ve çok seviyeli DPD ile el tutma hareketlerinin sınıflandırılması. Fırat Üniversitesi Mühendislik Bilimleri Dergisi 2022; 34(1): 33-43.

[16] Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z. Rethinking the inception architecture for computer vision. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; 2016. s. 2818-2826.

[17] Oliva A, Torralba A. Modeling the shape of the scene: A holistic representation of the spatial envelope. Int J Comput Vis 2001.

[18] Lazebnik S, Schmid C, Ponce J. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; 2006.

[19] Arı B. Medikal Veri Setleri için Yeni Bir Aşırı Öğrenme Makinesi Otomatik Kodlayıcı Tasarımı. Doktora Tezi, Fırat Üniversitesi, Elazığ, 2022.

[20] Donuk K, Arı A, Hanbay D. A CNN based real-time eye tracker for web mining applications. Multimed Tools Appl 2022; 81(27): 39103-39120.

[21] Turkoglu M, Aslan M, Arı A, Alçin ZM, Hanbay D. A multi-division convolutional neural network-based plant identification system. PeerJ Comput Sci 2021; 7: e572.