KSÜ Tarım ve Doğa Derg 27 (Ek Sayı 1), 243-253, 2024
KSU J. Agric Nat 27 (Suppl 1), 243-253, 2024
https://doi.org/10.18016/ksutarimdoga.vi.1486797

Comparative Analysis of CNN Algorithms for Mushroom Classification with Proposed Lightweight CNN Model

Ahmet NAMLI[1]✉, Didem ÖLÇER[2]
[1,2] Department of Computer Engineering, Faculty of Engineering, Ankara, Türkiye
[1]https://orcid.org/0000-0002-4649-3299, [2]https://orcid.org/0000-0001-7736-1021
✉: ahmetnamlics@gmail.com

ABSTRACT

The classification of mushroom species presents significant ecologic and health-related challenges; advancement in classification techniques is required to gain reliable identifications. This study aims to explain a methodology that was devised and evaluated in the development of a novel, lightweight Convolutional Neural Network (CNN) designed specifically for the task of mushroom classification. The paper provides a custom CNN model that is computationally cost-effective and capable of high-precision classification, fit for real-time usage. Hence, the proposed model was evaluated on this dataset of curated mushroom images with traditional classifiers and state-of-the-art CNN architectures, such as EfficientNet-B7, ResNet50, InceptionV3, and MobileNetV2. The custom model is depth-wise separations engineered in such a way that while they reduce the computational load, they don't compromise the effectiveness of the model. The custom model achieved a test score of 0.68, which is moderate compared to more established models such as EfficientNet-B7 or ResNet50. This approach helps the model function effectively even on platforms having low computational resources. A comprehensive evaluation reveals that a custom CNN has reasonable accuracy in the identification of different mushroom species vis-à-vis existing models, but also significantly lightens the classification process.

Agricultural Economics

Research Article

Article History
Received : 20.05.2024
Accepted : 14.08.2024

Keywords
Mushroom Classification
Convolutional Neural Network
Deep Learning

CNN Algoritmalarının Mantar Sınıflandırmada Önerilen Hafif CNN Modeli ile Karşılaştırmalı Analizi

ÖZET

Mantar türlerinin sınıflandırılması, ekolojik ve sağlıkla ilgili önemli zorluklar ortaya koymaktadır; güvenilir tanımlamalar elde etmek için sınıflandırma tekniklerinde ilerleme kaydedilmesi gerekmektedir. Bu çalışma, mantar sınıflandırma görevi için özel olarak tasarlanmış yeni, hafif bir Evrişimsel Sinir Ağının (CNN) geliştirilmesi üzere tasarlanan ve değerlendirilen bir metodolojiyi açıklamayı amaçlamaktadır. Makale, hesaplama açısından uygun maliyetli ve yüksek hassasiyetli sınıflandırma yapabilen, gerçek zamanlı kullanıma uygun özel bir CNN modeli sunmaktadır. Bu nedenle, önerilen model, geleneksel sınıflandırıcılar ve EfficientNet-B7, ResNet50, InceptionV3 ve MobileNetV2 gibi son teknoloji CNN mimarileri ile mantar görüntülerinden oluşan bu veri kümesi üzerinde değerlendirilmiştir. Özel modelin, hesaplama karmaşıklığını azaltırken modelin etkinliğinden ve yeteneğinden ödün vermeyecek şekilde tasarlanmasına özen gösterilmiştir. Özel model, EfficientNet-B7 veya ResNet50 gibi daha yerleşik modellerle karşılaştırıldığında orta düzeyde bir değer olan 0,68'lik bir test puanı elde etti. Bu yaklaşım, modelin düşük hesaplama kaynaklarına sahip platformlarda bile etkili bir şekilde çalışmasına yardımcı olur. Kapsamlı bir değerlendirme, tasarlanan CNN'in yalnızca mevcut modellere kıyasla farklı mantar türlerinin tanımlanmasında makul bir doğruluğa sahip olduğunu değil, aynı zamanda sınıflandırma sürecini de önemli ölçüde hafiflettiğini ortaya koymaktadır.

Tarım Ekonomisi

Araştırma Makalesi

Makale Tarihçesi
Geliş Tarihi : 20.05.2024
Kabul Tarihi : 14.08.2024

Anahtar Kelimeler
Mantar Sınıflandırma
Evrişimsel Sinir Ağları
Derin Öğrenme

Atıf Şekli: Namlı A, Ölçer D 2024. CNN Algoritmalarının Mantar Sınıflandırmada Önerilen Hafif CNN Modeli ile Karşılaştırmalı Analizi. KSÜ Tarım ve Doğa Derg 27 (Ek Sayı 1), 243-253. https://doi.org/10.18016/ksutarimdoga.vi.1486797

To Cite : Namlı A, Ölçer D 2024. Comparative Analysis of CNN Algorithms for Mushroom Classification with Proposed Lightweight CNN Model. KSU J. Agric Nat 27 (Suppl 1), 243-253. https://doi.org/10.18016/ksutarimdoga.vi.1486797

INTRODUCTION

A mushroom is the edible reproductive structure of various types of fungi that are attached to basidiomycetes and grow on the surface of the ground. Fungi are a diverse and integral part of ecological systems, closely linked to living beings (Hibbett et al., 2007; Liu et al., 2016; Wang, 2022; Zahan et al., 2021; Zhang et al., 2021). The presence of diverse fungi is essential for comprehending these organisms' complex nature and importance in our ecosystems. Although there have been recent advancements in the identification of fungal species, only a small fraction, specifically 5%, of the estimated 3.8 million fungal species have been identified (Hawksworth & Lücking, 2017). Approximately 14,000 (Zahan et al., 2021) - 19,000 (Hawksworth & Lücking, 2017) mushroom species have been identified, while experts concur that there are still several undiscovered species. The delicate balance between edible and poisonous mushrooms emphasizes the significance of meticulous species identification. While certain edible mushrooms may not possess a pleasant taste, there exist compelling reasons to consume them. These reasons include the fact that edible mushrooms possess notable advantages, such as their ability to eradicate cancer cells, combat infections, and enhance the human immune system. In addition, mushrooms are exceptionally nourishing, serving as a valuable protein source, containing few calories and unsaturated fats, and offering a substantial supply of vitamins and iron (Ria et al., 2021). Edible mushrooms are favored for their delectable flavor and abundant nutritional content. Mushrooms are a common dietary source that is rich in protein and low in fat (Dan, 2020; Yan et al., 2023; Yu et al., 2020). Therefore, mushrooms provide a more nutritious substitute for conventional sources of protein. The therapeutic potential of mushrooms highlights their usefulness beyond their nutritional composition. Moreover, due to a lack of awareness regarding toxic mushrooms, a significant number of individuals perish as a result of consuming them. This highlights the necessity for enhanced public knowledge and education regarding mushroom species. Presumably, mushrooms with flawless cellular structure, vibrant colors, and no interaction with birds and insects are toxic (Zahan et al., 2021). When discussing the hazards of eating mushrooms, it is crucial to emphasize the problems linked to poisonous species. The distinctive morphological characteristics of poisonous mushrooms include vibrant and colorful scales on the cap and a ring-shaped structure beneath the top. Novice gatherers are prone to committing errors. Toxic mushrooms have a detrimental impact on the neurological system, perhaps resulting in fatalities when ingested in large amounts (Ketwongsa et al., 2022; Zahan et al., 2021). Therefore, precise identification is essential for ensuring safety. Performing biochemical tests and interpreting morphological traits might be challenging for non-specialists in everyday life. These issues require the creation of identifying systems that are easier for users to understand and utilize. Consequently, numerous researchers have dedicated their efforts to developing various models and methodologies (Tutuncu et al., 2022). These endeavors are a reaction to the increasing demand for easily accessible resources for identifying mushrooms.

The majority of the existing articles on the categorization of wild mushrooms employ machine learning (ML) techniques. These strategies offer a more easily understood and attainable method for identifying mushrooms. The literature employs many ML techniques, such as decision trees, simple Bayesian, AdaBoost, and support vector machine (SVM) methods (Tutuncu et al., 2022). Each of these strategies possesses unique advantages and constraints. Nevertheless, these techniques depend on manually labeled features, and the algorithm acquires knowledge about the labeled wild mushroom data rather than the gathered visual data for classification (Peng et al., 2023). The reliance on manual labeling is a constraint that could be overcome by employing more sophisticated methods, such as deep learning DL. The wide-ranging applications exemplify the extensive influence of DL. DL techniques offer notable benefits in the field of pattern recognition because they can extract the most useful information from complex and multidimensional data. For this reason, it is especially pertinent when it comes to identifying mushrooms. Possessing this skill is essential for precisely discerning the species of mushrooms.

Several studies have utilized artificial intelligence to analyze the features of mushrooms and create models that aid consumers in identifying various mushroom species and preventing mushroom poisoning. Tarawneh et al. (Tarawneh et al., 2023) developed an innovative model that integrates Decision Trees, Naive Bayes, and SVM to distinguish between edible and poisonous mushrooms, achieving an impressive accuracy of 94%. This integrated approach leverages the strengths of individual ML algorithms to synthesize a more reliable decision framework, utilizing the UCI Mushroom Dataset to validate its efficacy. Similarly, Tutuncu et al. (2022) employed a combination of Decision Tree, Naïve Bayes, SVM, and AdaBoost algorithms, with AdaBoost achieving %100 classification accuracy. This study underscores the potential of ensemble methods in enhancing the predictive capabilities of ML models, particularly in applications where public health is at stake. The study of Ria et al. (2021) further illustrates the role of neural networks in mushroom classification. By reviewing various supervised learning approaches (Ma et al., 2020), including Random Forest and K-Nearest Neighbour alongside artificial neural network (ANN), the study highlights ANN's superior ability to manage complex data patterns, thereby enhancing classification processes.

In an innovative shift towards deep learning, Peng et al. (2023) proposed a multidimensional feature fusion attention network that combines the strengths of CNNs and Vision Transformers. This approach, validated on two comprehensive datasets, showcases superior accuracy and model robustness, setting a new benchmark in the field. Complementing these studies, Zhao et al. (2021) explored the use of an ensemble of CNNs through a bagging algorithm to improve classification accuracy. This research highlights a paradigm shift in biological classifications, emphasizing the transition towards DL models that are capable of handling complex image data and achieving high levels of accuracy. Wang (2022) utilized a Vision Transformer model for mushroom classification, achieving a remarkable 95.97% accuracy. This study not only marks a significant advancement in the application of transformer-based models but also demonstrates their effectiveness in reducing intraclass variability and enhancing feature discrimination. Additionally, Zahan et al. (2021) demonstrated the efficacy of CNNs in recognizing complex patterns inherent in biological entities, including fungi. They employed InceptionV3, VGG16, and ResNet50 models to classify mushrooms, showcasing InceptionV3's superior performance on a contrast-enhanced dataset, thus highlighting the importance of preprocessing techniques and architectural choices. Long et al. (2023) developed an advanced ML model using the MobileViT architecture, optimized for efficiency and accuracy in mobile applications. This research addresses the challenge of large variance within mushroom classes by implementing an innovative separable self-attention mechanism, resulting in improved computational efficiency and classification accuracy.

These collective efforts in applying ML and DL models to mushroom classification not only advance our understanding of fungal biodiversity but also significantly contribute to public health by improving the accuracy and reliability of identifying edible and poisonous mushrooms. The ongoing development of these technologies promises further enhancements in food safety and ecological research, underpinning the critical role of interdisciplinary approaches in modern scientific inquiries.

The widespread adoption of ML techniques has led to the development of numerous applications across various domains. ML algorithms can independently acquire real-world knowledge by emulating human learning processes, making them useful in various fields (Portugal et al., 2018). In agriculture, ML approaches have been used for different areas (Boyacı et al., 2023; Karadaş&Bulut, 2024). ML approaches have been primarily used to categorize mushrooms (Kamilaris & Prenafeta-Boldú, 2018), but many methods require extensive training and testing, leading to limited precision. Previous studies have explored ML strategies like Decision Trees, Naïve Bayes, AdaBoost, and SVMs, but these often rely on manually labeled features, which limits their generalizability to new datasets (Zhao et al., 2021). Additionally, complex models in CNNs require high processing power and hardware requirements, and there is a lack of research comparing the efficacy of commonly used models in mushroom classification. This study aims to achieve similar results using a more lightweight model to address these challenges and improve the effectiveness of ML in mushroom classification.

The study presents a significant advancement in the use of CNNs for classifying mushrooms, focusing on the application of artificial intelligence (AI) in addressing biological classification issues. It creates a specialized CNN model to identify 11 specific mushroom species, enhancing the accuracy of species identification. The methodology used in the study provides a comprehensive framework for future research in related domains, offering a repeatable model for researchers interested in using ML techniques for mushroom classification. One of the contributions of the study is the comparison study between the custom CNN model and well-established models like EfficientNet-B7, ResNet50, InceptionV3, and MobileNetV2 enhances discussions on the efficacy of different neural network architectures in dealing with highly specialized classification tasks. The smaller size of the custom model results in a decreased computational load, which can significantly speed up the training and inference processes, especially on less powerful hardware. The study also addresses the problem of data imbalance when training ML models, emphasizing the importance of constructing a robust dataset and proposing techniques to improve data representation. The paper offers a basis for creating lightweight systems that can assist mycologists, ecologists, and the public in rapidly and precisely identifying mushroom species, promoting biodiversity monitoring, and public safety.

MATERIAL and METHOD

The dataset (Zhecheng, 2023) used in this study consists of 7,767 photos, with each image representing one of 11 different mushroom species. The species are classified into the following categories: Lactarius, Russula, Boletus, Cortinarius, Amanita, Inocybe, Exidia, Entoloma, Agaricus, Hygrocybe, and Suillus. The dataset was thoroughly examined to identify authentic photos, excluding seven unsuitable files. 74 exact duplicates and eight nearly identical pairs were identified. Quality control included removing blurry, dark, black-and-white, bright, and low-informational images. The dataset was filtered to 7,671 photos, a valuable resource for training machine learning models. Table 1 shows the class distributions of the images.

Table 1. Number of samples from each mushroom species

Çizelge 1. Her mantar türünden örnek sayısı

Mushroom Species	Number of Samples
Lactarius	1498
Russula	1141
Boletus	1069
Cortinarius	834
Amanita	748
Inocybe	611
Exidia	432
Entoloma	363
Agaricus	351
Hygrocybe	314
Suillus	310

Dataset Splits

One often-used approach to assess the performance of these models is to divide the dataset into several subsets that are used for training, validation, and testing. The test set is not used throughout the model training process but is instead kept aside for a final evaluation. The approach was to assign 10% of the dataset for the testing phase which corresponds to 768 images. The remaining 90% of the data is utilized for training and validating the model. However, employing a random data split into training and validation sets might result in notable issues, particularly when dealing with imbalanced datasets characterized by the underrepresentation of certain classes. To tackle this problem, the technique of Stratified K-Fold Cross-Validation is utilized.

Dataset Pre-processing

In that study, a methodological approach was adopted where image pre-processing steps were integral, utilizing specific functions associated with several well-established CNN models provided by the TensorFlow Keras applications module (Chollet, 2015; Abadi et al., 2015). These models included ResNet50, InceptionV3, EfficientNetB7, and MobileNetV2, each requiring tailored pre-processing to conform input images to the conditions that optimally match each model's training environment. Additionally, the study introduced a novel CNN architecture for which, rather than developing a new pre-processing function, the 'preprocessing_input' function which is meant to adequate your image to the format the model requires from 'tensorflow.keras.applications.resnet50' (Chollet, 2015; Abadi et al., 2015) was adopted. This choice not only facilitated consistency in handling input data but also strategically leveraged established pre-processing norms to ensure the robustness and reliability of the new model under test conditions.

Methods

In the realm of mushroom classification using CNNs, several well-known models have demonstrated substantial efficacy, including ResNet50 (He et al., 2016), InceptionV3 (Szegedy et al., 2016), EfficientNetB7 (Tan & Le, 2019), and MobileNetV2 (Sandler et al., 2018). These models, utilizing DL architectures, will be tested one by one in this study for their ability to handle the inherent complexity of identifying and classifying various mushroom species from images, and will be compared with the model subject to the study.

In the process of fine-tuning various deep learning architectures such as ResNet50, InceptionV3, EfficientNetB7, and MobileNetV2 for a specific classification task with 11 output classes, several strategic modifications are implemented to adapt these models more effectively to the task. Initially, a 2D Global Average Pooling layer is introduced after the final convolutional layer of each model. This layer aggregates the features into a single 2D map per channel, effectively reducing the spatial dimensions and the complexity of the model while retaining essential spatial information.

Following this dimensionality reduction, two fully connected Dense layers with 128 and 256 neurons are added respectively. Each of these layers employs a Rectified Linear Unit (ReLU) activation function to introduce non-linearity, enhancing the model's ability to capture and learn complex patterns from the data. To further improve the training dynamics and stabilize the learning process, batch normalization layers follow each dense layer. These layers normalize the inputs for each layer, centering the mean output close to zero and the standard deviation close to one, which helps in mitigating issues related to input variation sensitivity.

To combat the risk of overfitting, a dropout layer follows each batch normalization step. During training, these dropout layers randomly nullify a fraction of the input units, ensuring that the model does not overly depend on any specific neuron and thereby promoting a more generalized learning outcome. The culmination of this fine-tuning process is a final dense layer with 11 units, corresponding to the number of classification classes, which utilizes a softmax activation function to yield the probability distribution across the classes.

These enhancements collectively tailor the original architectures of ResNet50, InceptionV3, EfficientNetB7, and MobileNetV2, optimizing them for improved performance on the dataset in question by effectively balancing complexity, computational efficiency, and generalization capability.

Regarding the model constructed which is the focus of the study, in constructing the CNN described, the architecture is meticulously organized into sequential layers, each designed to handle specific transformations of the input data for effective feature learning and classification. The model adopts 10 convolutional layers arranged in increasing complexity with filters ranging from 16 to 256. Each convolutional layer utilizes a (3, 3) kernel size and employs 'same' padding to ensure that the spatial dimensions of the output feature maps remain unchanged, thus preserving edge information across the network's depth.

The activation function selected for this model is the Exponential Linear Unit (ELU), which presents several advantages over the commonly used ReLU. According to research by Clevert et al. (2016), ELU can help reduce the vanishing gradient problem common in deep neural networks by maintaining mean activations closer to zero, which facilitates a faster and more effective learning process. The ELU function is particularly effective in DL architectures as it allows for negative outputs, contributing to a more robust learning mechanism by adding slight perturbations to the activation map, thereby enhancing generalization.

Weight initialization in this model is performed using the “He uniform” method, which is particularly suited to networks employing ELU activation functions. He et al. (2015) demonstrated that this initializer could significantly impact the network's ability to learn efficiently by maintaining the variance of activations throughout the layers. By initializing weights from a uniform distribution within a range derived from the number of input units, the “He uniform” initializer ensures that the gradient magnitudes are neither too small (causing vanishing gradients) nor too large (leading to exploding gradients), thus facilitating stable and rapid convergence during training.

Batch normalization is applied consistently after each convolutional operation. This technique, as expounded by Ioffe and Szegedy (2015), normalizes the outputs of the previous layers by recalibrating the mean and variance. Such normalization stabilizes the learning process and allows for higher learning rates, reducing the model's training time significantly. Additionally, it provides a form of regularization, albeit indirectly, by smoothing the optimization landscape.

Dropout is strategically incorporated at various stages within the network with rates determined through hyper-parameter tuning, which helps in identifying optimal values that prevent overfitting while allowing the network to retain a significant capacity for learning patterns in the data. Srivastava et al. (2014) first introduced this technique, demonstrating its effectiveness in reducing overfitting by randomly omitting subsets of features at each training stage, thereby compelling the network to learn more robust features. The varying dropout rates from 0.1 to 0.7 reflect a targeted approach, with higher rates likely used in layers that are more prone to overfitting due to their complexity and capacity.

The inclusion of pooling layers, specifically max pooling, serves to reduce the dimensionality of the feature maps, thus decreasing the computational load and the number of parameters in the network. This reduction not only speeds up the training process but also minimizes overfitting by abstracting the highest-valued features from the preceding feature maps.

Towards the end of the model, global average pooling is utilized to convert each feature map to a single value, effectively summarizing the spatial information, which is critical for maintaining the most relevant features for classification tasks. Following this, dense layers with ELU activation and additional dropout are employed to finalize the feature processing and lead to a classification decision made by a softmax-activated output layer. This layer distributes the probability across the various classes, facilitating a multi-class classification.

All CNN configurations are compiled with the Adam optimizer, known for its efficiency in handling sparse gradients and adapting the learning rate during training, and it employs categorical cross-entropy as a loss function, ideal for multi-class problems where each class is mutually exclusive.

In sum, as indicated in Figure 1, the detailed architectural choices and parameter settings in mentioned CNN are aligned with current best practices in DL for image classification, emphasizing stability, efficiency, and robustness in learning. The integration of advanced techniques like ELU, He initialization, and batch normalization alongside strategic dropout application underpins the model's capability to perform effectively in complex visual recognition tasks.

When creating the model, care was taken to make it lightweight. The custom model created for this study consists of 1,450,523 parameters, occupying approximately 5.53 MB of memory. The parameter counts and memory requirements of other models, such as EfficientNet-B7 (64,462,882 parameters, 245.91 MB), ResNet50 (23,887,371 parameters, 91.12 MB), InceptionV3 (22,102,443 parameters, 84.31 MB), and MobileNetV2 (2,459,339 parameters, 9.38 MB), are significantly higher than this. The smaller size of the custom model results in a decreased computational load, which can significantly speed up the training and inference processes, especially on less powerful hardware.

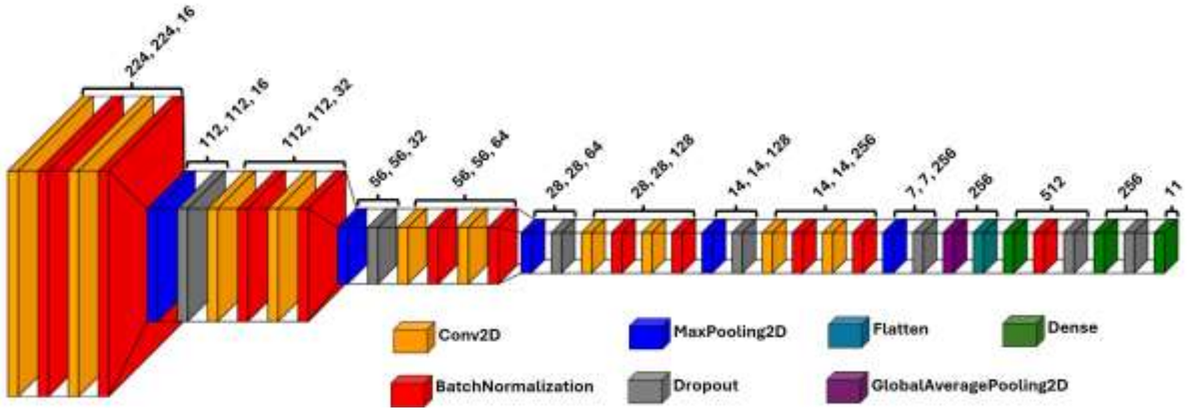
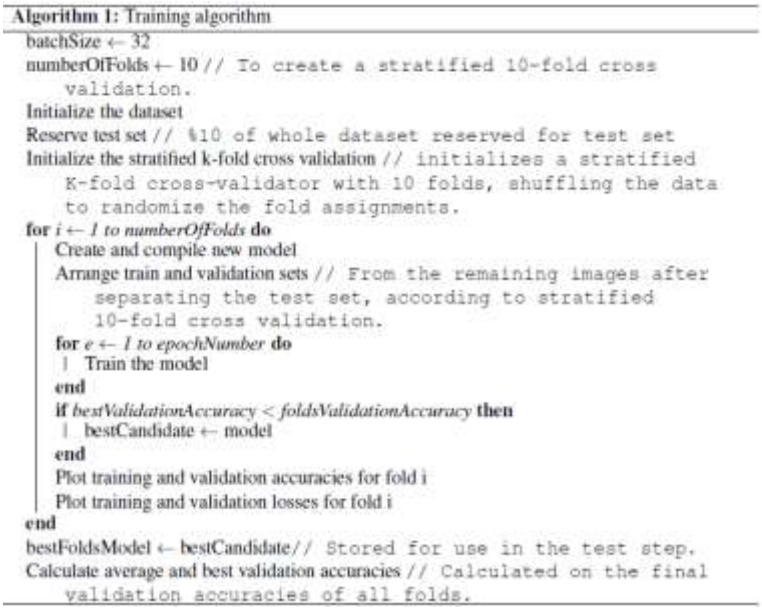


Figure 1. Custom CNN model architecture
Şekil 1. Tasarlanan CNN model mimarisi



```
Algorithm 1: Training algorithm
  batchSize ← 32
  numberOfFolds ← 10 // To create a stratified 10-fold cross
      validation.
  Initialize the dataset
  Reserve test set // %10 of whole dataset reserved for test set
  Initialize the stratified k-fold cross validation // initializes a stratified
      K-fold cross-validator with 10 folds, shuffling the data
      to randomize the fold assignments.
  for i ← 1 to numberOfFolds do
      Create and compile new model
      Arrange train and validation sets // From the remaining images after
          separating the test set, according to stratified
          10-fold cross validation.
      for e ← 1 to epochNumber do
          Train the model
      end
      if bestValidationAccuracy < foldsValidationAccuracy then
          bestCandidate ← model
      end
      Plot training and validation accuracies for fold i
      Plot training and validation losses for fold i
  end
  bestFoldsModel ← bestCandidate// Stored for use in the test step.
  Calculate average and best validation accuracies // Calculated on the final
      validation accuracies of all folds.
```

Figure 2. Pseudocode of training process
Şekil 2. Eğitim sürecinin sözde kodu

In this study, 10-fold cross-validation was employed, considering the specific attributes of the dataset. The batch size was set to 32 for training iterations, balancing computational efficiency and maintaining a sufficiently stochastic gradient estimation. The training was conducted over 75 epochs to allow the network sufficient iterations to adequately learn and adapt to the dataset without overfitting.

During each fold of cross-validation, models were monitored for validation accuracy improvements, and the best-performing weights were saved. These methodological choices ensure the training process is both efficient and robust, leading to a CNN well-tuned for generalization beyond the training dataset. The same methodology, as described in Figure 2, was used for all CNN models used, only the models themselves were changed. By keeping the other variables constant, the aim was to observe the performance of the models under identical situations. After successfully training and identifying the best-performing model across various validation folds, the research focus shifts towards evaluating the model's practical efficacy on unseen data and ensuring the robustness of its predictive capabilities. The model is systematically evaluated against the test data to ascertain its performance metrics, notably accuracy and loss. During the test phase, similar to the training phase, all the models were treated uniformly and exposed to identical testing procedures.

RESULTS and DISCUSSION

This chapter presents all the outcomes of the process of model training, validation, and testing. The results of the setup are thoroughly examined, and detailed comparisons and illustrations are provided.

Figure 3 displays the training and validation metrics of the most successful fold of the model, which is retained for the testing phase. Upon analysis, it is evident that the training and validation accuracies exhibit a consistent upward trend. However, despite the inability to reach the same level of continuity, it is evident that the training and validation losses are decreasing. However, there is a gradual widening of the gap between them. With this information, the validation accuracies of each fold of the model and the average validation accuracy are shown in Table 2 below.

Figure 3. Training and validation metrics graph

Şekil 3. Eğitim ve doğrulama metrikleri grafiği

Table 2. Validation accuracies of the model in folds

Çizelge 2. Modelin katlamalardaki validasyon doğrulukları

Fold Number:	1	2	3	4	5	6	7	8	9	10	Average
Validation Accuracy:	0.67	0.70	0.69	0.62	0.68	0.65	0.67	0.67	0.68	0.70	0.67

As a result of the cross-validation, the model trained on the most successful fold was passed to the test step in which the images (%10 of the dataset) that were never seen in the training phase were used. The tests yielded a prediction accuracy rate of 0.68.

Table 3 displays the precision, recall, and F1-score values for each mushroom species, together with the corresponding number of samples representing these species in the test dataset. Precision, measuring the accuracy of positive predictions, shows substantial variability across classes. For instance, Hygrocybe achieves perfect precision, indicating precise predictions, but has low recall. In contrast, Agaricus and Suillus show lower precision, highlighting issues with false positives. Recall evaluates the model's ability to detect all relevant examples in a class. Boletus excels in recall, capturing most instances, whereas Hygrocybe performs poorly, missing many true instances. The F1 score combines precision and recall, offering a unified measure of performance. Boletus has a high F1 score, indicating balanced precision and recall, while Suillus has a notably low score, suggesting areas for improvement. According to Table 3 and Table 4, the model exhibits varied proficiency across classes, with strong results in some and underperformance in others. These findings could inform targeted improvements or training adjustments to boost accuracy and recall across all categories, ensuring robust performance irrespective of class distribution.

The confusion matrix which is given in Figure 4 offers a comprehensive evaluation of a model's performance for each class by classifying predictions into true positives, true negatives, false positives, and false negatives, and this is an important key to understanding the precision, recall, and F1-score metrics mentioned earlier. This comprehensive data is essential for discerning the model's distinct advantages and disadvantages, demonstrating its effectiveness in other domains, and revealing any inclinations to inaccurately classify one category as another.

Stratified k-fold cross-validation ensures that there is minimal variation in the validation accuracies, but results showed that some classes did not achieve the desired level of success in fulfilling the model's purpose due to the distribution of the dataset. When the dataset is analyzed, as indicated in Table 1, it is seen that there is a linear proportionality between recall scores and image distributions. In the rigorous assessment of CNNs applied to an 11-class mushroom classification task, various established and custom models were evaluated using stratified 10-fold cross-validation to gauge their performance during the training phase and subsequently tested with independent test data to verify their generalization capabilities.

During the training phase, as indicated in Table 5, the EfficientNet-B7 model demonstrated the highest efficacy, in best fold validation accuracy and average accuracy across folds. This superior performance suggests that EfficientNet-B7's methodical approach to scaling network dimensions systematically is highly effective for this task. In contrast, the InceptionV3 model, despite its sophisticated architecture that allows for complex feature extraction through varied convolutional filter sizes, lagged in performance, in best fold accuracy and an average of 63.78%. Test scores of each model are shown in Table 5. EfficientNet-B7 continued to outperform the other models in test scores. ResNet50, known for its deep residual learning framework that facilitates the training of deeper networks by addressing the vanishing gradient problem, also performed commendably in both the validation and testing phases. This indicates its effectiveness in capturing and generalizing the underlying patterns of the dataset.

Table 3. Custom model’s test scores by species
Çizelge 3. Tasarlanan modelin türlere göre test sonuçları

Mushroom species	Precision	Recall	F1-Score	Sample
Agaricus	0.49	0.49	0.489	39
Amanita	0.84	0.61	0.71	77
Boletus	0.77	0.92	0.84	104
Cortinarius	0.55	0.68	0.61	81
Entoloma	0.43	0.69	0.53	32
Exidia	0.87	0.75	0.81	36
Hygrocybe	1.0	0.40	0.57	30
Inocybe	0.66	0.61	0.63	67
Lactarius	0.67	0.64	0.65	146
Russula	0.74	0.76	0.75	132
Suillus	0.47	0.33	0.39	24

Table 4. Metrics of custom CNN model
Çizelge 4. Tasarlanan CNN modelinin ölçümleri

Metrics	Precision	Recall	F1-Score	Sample
Macro average	0.68	0.62	0.63	768
Weighted average	0.70	0.68	0.68	768

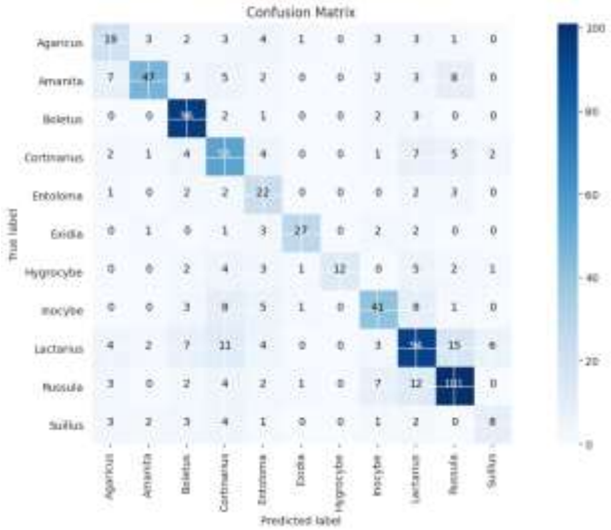


Figure 4. Confusion matrix of custom CNN model
Şekil 4. Tasarlanan CNN modelinin karmaşıklık matrisi

Conversely, the Custom Model and InceptionV3 showed moderate to low performance in both phases. These outcomes may signal the need for further model refinement or a re-evaluation of their network architectures and training parameters to better suit the classification task at hand. MobileNetV2, designed for efficiency in mobile environments, achieved reasonable success. Its performance indicates a balanced trade-off between computational efficiency and predictive accuracy, making it a viable option for applications where deployment constraints are a factor.

For macro averages, which are given in Table 6, which treat all classes equally, the EfficientNet-B7 model demonstrates superior performance in F1-score, precision, and recall. This suggests that EfficientNet-B7 is exceptionally consistent across all classes, balancing accuracy and coverage effectively. ResNet50 follows with a solid F1-score, indicating reliable performance, though slightly less consistent across classes compared to EfficientNet-B7.

Table 5. Comparison of models by validation accuracies and test scores
Çizelge 5. Modellerin validasyon doğrulukları ve test puanlarına göre karşılaştırılması

Model	Best Fold’s Validation Accuracy	Average Validation Accuracy	Test Score
Custom model	0.70	0.67	0.68
ResNet50	0.82	0.79	0.75
EfficientNet-B7	0.82	0.80	0.81
InceptionV3	0.68	0.64	0.64
MobileNetV2	0.75	0.72	0.72

Table 6. Comparison of models by macro average

Çizelge 6. Modellerin makro ortalamaya göre karşılaştırılması

Model	Precision	Recall	F1-Score
Custom model	0.68	0.63	0.64
ResNet50	0.78	0.74	0.75
EfficientNet-B7	0.81	0.80	0.80
InceptionV3	0.65	0.60	0.61
MobileNetV2	0.70	0.72	0.70

The Custom Model, InceptionV3, and MobileNetV2 show lower effectiveness, in macro average F1-scores. These scores indicate challenges in either precision or recall, which could be due to various factors such as model architecture limitations, insufficient training, or the intrinsic difficulty of some classes that could not be effectively learned by these models.

When considering the weighted averages, as indicated in Table 7, which account for the prevalence of each class, a similar pattern emerges. EfficientNet-B7 maintains its lead in F1-score, highlighting its robustness and ability to generalize well across the varied sizes of the classes. It is followed by ResNet50 and MobileNetV2, which also show competent performance with weighted F1 scores. This indicates that while these models perform well on more populous classes, there might be room for improvement in handling less represented classes.

Table 7. Comparison of models by weighted average

Çizelge 7. Modellerin ağırlıklı ortalamaya göre karşılaştırılması

Model	Precision	Recall	F1-Score
Custom model	0.70	0.68	0.68
ResNet50	0.77	0.75	0.75
EfficientNet-B7	0.82	0.82	0.81
InceptionV3	0.66	0.65	0.64
MobileNetV2	0.73	0.72	0.72

The Custom Model and InceptionV3 exhibit less optimal results with weighted F1 scores. These outcomes suggest that these models, while reasonably effective for some classes, struggle with achieving high accuracy and coverage across all classes, particularly those that are less frequent in the dataset. The data suggests that while some models like EfficientNet-B7 and ResNet50 are capable of delivering robust and balanced performances across diverse class distributions, others such as the Custom Model and InceptionV3 may benefit from further tuning and training to enhance their precision and recall capabilities, ensuring more consistent performance across all classes. In addition, it was observed that the custom model showed more accurate values than InceptionV3, although it showed lower accuracies than the other 3 models. Considering the number of model variables and their complexity, it can be concluded that the results are promising, although not completely satisfactory.

Finally, EfficientNet-B7's performance across various fungal classes, which is observed to be the most successful in the light of the results mentioned, as evidenced by the detailed metrics for precision, recall, and F1-score, showcases a complex landscape of effectiveness that varies significantly from one class to another. These results provide a more granular insight into the model's capacity to accurately identify and classify instances within a multi-class framework.

CONCLUSION

The study explores the application of CNNs for mushroom classification, focusing on developing a customized model capable of accurately categorizing 11 different mushroom classes. The custom model demonstrates a feasible approach to mushroom classification, especially suitable for scenarios where computational efficiency is a priority given its relatively simpler architecture compared to deeper, more computationally expensive models. The model's performance, particularly in the testing phase, reveals significant insights for both academic and practical applications in mycological classification. The custom model achieved a test score of 0.68, which is moderate compared to more established models like EfficientNet-B7 or ResNet50, known for their complex architecture and ability to achieve higher accuracy in image-based tasks. However, the model faced difficulties with the Suillus class, resulting in a lower F1-score of 0.39. The uneven distribution of training data contributed to the variation in findings, suggesting that the model struggles with under-represented categories. The results indicate a direct relationship between class representation in the training data and the model's performance. Classes with fewer examples tended to yield poorer recall and precision, suggesting that the model struggles with under-represented categories. The testing phase also highlighted a consistent average validation accuracy of about 0.67, pointing to the model's stability but highlighting its limited capability to transcend its training when faced with new, unseen data. Advanced models like EfficientNet-B7, ResNet50, and MobileNetV2 have the potential to significantly improve performance, incorporating advanced methods like compound scaling and residual learning. However, the model's applicability is limited by its moderate accuracy and the need for enhanced training strategies to improve its generalization capabilities across all mushroom classes. Despite all this considering the number of model variables and their complexity, it can be concluded that the results are promising, although not completely satisfactory. The model was found to achieve better results than the much more complex InceptionV3. This is promising for future studies. The results highlight the importance of a well-prepared dataset in training a successful ML model. At the same time, dataset augmentation strategies to increase the robustness of the model are planned for future work. In conclusion, the process of refining and optimizing CNNs for mushroom classification is still in progress, offering a strong foundation for further research. This study provides a good foundation and a strong starting point for further research that aims to expand the capabilities of these advanced ML models in mushroom classification. The ultimate goal is to develop a model that not only achieves high accuracy across all mushroom classes but also serves as a reliable tool for mycologists and enthusiasts in the field, marrying the technical prowess of CNNs with the intricate beauty of mushroom species.

Contribution of the Authors

1- Ahmet Namlı executed the experiment with the help of Assist. Prof. Didem Ölçer.

2- Ahmet Namlı wrote the article and critically reviewed it by Assist. Prof. Didem Ölçer.

Conflict of Interest

The authors declare that there is no conflict of interest

REFERENCES

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., ... & Zheng, X. (2015). TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. Retrieved from https://www.tensorflow.org

Boyacı, S., Ertugrul, O., Ertuğrul, G. Ö., & Gökalp, D. D. (2023). Kırşehir İlinde Seralarda Kullanılan Sulama Sularının Kalite Parametrelerinin Belirlenmesi. *Kahramanmaraş Sütçü İmam Üniversitesi Tarım ve Doğa Dergisi, 26*(5), 1178-1185.

Chollet, F., (2015). Keras: Deep learning for humans. GitHub. Inc.

Clevert, D. A., Unterthiner, T., & Hochreiter, S. (2016). Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs). 4th International Conference on Learning Representations, ICLR 2016 - Conference Track Proceedings. Retrieved from https://arxiv.org/abs/1511.07289v5

Dan, K. (2020). The Nutritional Value and Application Progress of Edible Fungi. *Modern Food, 15*, 53–55.

Hawksworth, D. L., & Lücking, R. (2017). Fungal Diversity Revisited: 2.2 to 3.8 Million Species. *Microbiol Spectr, 5*(4), 10-1128. doi: 10.1128/MICROBIOLSPEC.FUNK-0052-2016.

He, K., Zhang, X., Ren, S., & Sun, J. (2015). Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification. CoRR, abs/1502.01852. Retrieved from http://arxiv.org/abs/1502.01852

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. In 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (pp. 770–778). doi: 10.1109/CVPR.2016.90.

Hibbett, D. S., Binder, M., Bischoff, J. F., Blackwell, M., Cannon, P. F., Eriksson, O. E., ... & Zhang, N. (2007). A higher-level phylogenetic classification of the Fungi. *Mycol Res, 111*(5), 509–547. doi: 10.1016/J.MYCRES.2007.03.004.

Ioffe, S., & Szegedy, C. (2015). Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. 32nd International Conference on Machine Learning, ICML 2015, 1, 448–456. Retrieved from https://arxiv.org/abs/1502.03167v3

Kamilaris, A., & Prenafeta-Boldú, F. X. (2018). Deep learning in agriculture: A survey. *Comput Electron Agric, 147*, 70–90. doi: 10.1016/J.COMPAG.2018.02.016.

Karadaş, K., & Bulut, O. D. (2024). Comparison of Predictive Performance of Data Mining Algorithms in Predicting Tomato Yield with the A Case Study in Igdir. *Kahramanmaraş Sütçü İmam Üniversitesi Tarım ve Doğa Dergisi, 27*(2), 443-452.

Ketwongsa, W., Boonlue, S., & Kokaew, U. (2022). A New Deep Learning Model for The Classification of Poisonous and Edible Mushrooms Based on Improved AlexNet Convolutional Neural Network. *Applied Sciences (Switzerland), 12*(7), 3409. doi: 10.3390/app12073409.

Liu, N. G., Ariyawansa, H. A., Hyde, K. D., Maharachchikumbura, S. S. N., Zhao, R. L., Phillips, A. J. L., ... & Jumpathong, J. (2016). Perspectives into the value of genera, families and orders in classification. *mycosphere, 7*, 1649–1668. doi: 10.5943/mycosphere/7/11/3.

Long, C., Yu, P., Li, H., & Li, H. (2023). Wild mushroom classification based on improved MobileViT_v2. In Proceedings of 2023 IEEE 3rd International Conference on Information Technology, Big Data and Artificial Intelligence, ICIBA 2023 (pp. 12–18). IEEE. doi: 10.1109/ICIBA56860.2023.10165212.

Ma, L., Gao, R., Han, H., Chen, C., Yan, Z., Zhao, J., ... & Xie, L. (2020). Efficient identification of Bachu mushroom by flourier transform infrared (FT-IR) spectroscopy coupled with PLS-GS-SVM. *Optik (Stuttg), 224*, 165712. doi: 10.1016/J.IJLEO.2020.165712.

Peng, Y., Xu, Y., Shi, J., & Jiang, S. (2023). Wild Mushroom Classification Based on Improved MobileViT Deep Learning. *Applied Sciences*

(Switzerland), 13(8), 4680. doi: 10.3390/app13084680.

Portugal, I., Alencar, P., & Cowan, D. (2018). The use of machine learning algorithms in recommender systems: A systematic review. *Expert Syst Appl, 97*, 205–227. doi: 10.1016/J.ESWA.2017.12.020.

Ria, N. J., Badhon, S. M. S. I., Khushbu, S. A., Akter, S., & Hossain, S. A. (2021). State of art Research in Edible and Poisonous Mushroom Recognition. In 2021 12th International Conference on Computing Communication and Networking Technologies, ICCCNT 2021. Institute of Electrical and Electronics Engineers Inc. doi: 10.1109/ICCCNT51525.2021.9579987.

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L. C. (2018). MobileNetV2: Inverted Residuals and Linear Bottlenecks. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 4510–4520. doi: 10.1109/CVPR.2018.00474.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research, 15*(56), 1929–1958. Retrieved from http://jmlr.org/papers/v15/srivastava14a.html

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the Inception Architecture for Computer Vision. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2818–2826. doi: 10.1109/CVPR.2016.308.

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

Tarawneh, O., Tarawneh, M., Sharrab, Y., & Husni, M. (2023). Mushroom classification using machine-learning techniques. AIP Conf Proc, 2979(1), e030003. doi: 10.1063/5.0174721.

Tutuncu, K., Cinar, I., Kursun, R., & Koklu, M. (2022). Edible and Poisonous Mushrooms Classification by Machine Learning Algorithms. In 2022 11th Mediterranean Conference on Embedded Computing, MECO 2022. Institute of Electrical and Electronics Engineers Inc. doi: 10.1109/MECO55406.2022.9797212.

Wang, B. (2022). Automatic Mushroom Species Classification Model for Foodborne Disease Prevention Based on Vision Transformer. *Journal of Food Quality, 2022*(1), 1173102. doi: 10.1155/2022/1173102.

Yan, Z., Liu, H., Li, J., & Wang, Y. (2023). Application of Identification and Evaluation Techniques for Edible Mushrooms: A Review. *Critical Reviews in Analytical Chemistry, 53*(3), 634–654. doi: 10.1080/10408347.2021.1969886.

Yu, Q., Guo, M., Zhang, B., Wu, H., Zhang, Y., & Zhang, L. (2020). Analysis of Nutritional Composition in 23 Kinds of Edible Fungi. *Journal of Food Quality, 2020*, 8821315. doi: 10.1155/2020/8821315.

Zahan, N., Hasan, M. Z., Malek, M. A., & Reya, S. S. (2021). A Deep Learning-Based Approach for Edible, Inedible and Poisonous Mushroom Classification. In 2021 International Conference on Information and Communication Technology for Sustainable Development, ICICT4SD 2021 - Proceedings (pp. 440–444). Institute of Electrical and Electronics Engineers Inc. doi: 10.1109/ICICT4SD50815.2021.9396845.

Zhang, Y., Mo, M., Yang, L., Mi, F., Cao, Y., Liu, C., ... & Xu, J. (2021). Exploring the Species Diversity of Edible Mushrooms in Yunnan, Southwestern China, by DNA Barcoding. *J Fungi (Basel), 7*(4), 310. doi: 10.3390/JOF7040310.

Zhao, H., Ge, F., Yu, P., & Li, H. (2021). Identification of Wild Mushroom Based on Ensemble Learning. In 2021 IEEE 4th International Conference on Big Data and Artificial Intelligence, BDAI 2021 (pp. 43–47). IEEE. doi: 10.1109/BDAI52447.2021. 9515225.

Zhecheng, L. (2023). Mushroom Classification Dataset. Retrieved from https://www.kaggle.com/datasets/lizhecheng/mushroom-classification, Accessed: Apr. 30, 2024.