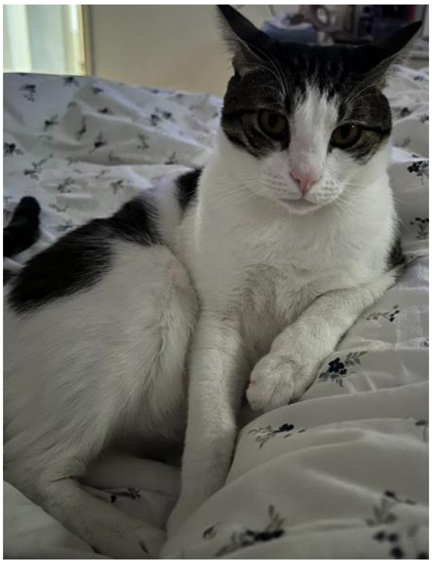
Classification of Dog and Cat Images Using Machine Learning

Afeka — The Academic College of Engineering

Instructor: Dr. Yalov Handzel Sharon

Course: Deep Learning



Authors:

Eden Amram

Eden Giladi

Submission Date: March 10, 2025

Table of Contents

1. Keywords

Image classification, machine learning, Convolutional Neural Networks (CNNs), Decision Trees, transfer learning, computer vision, model optimization, Data processing, Adam.

2. Abstract

Image classification is a fundamental task in computer vision, essential for applications ranging from medical diagnostics to autonomous systems. This study compares two machine learning approaches—Decision Trees and Convolutional Neural Networks (CNNs)—to evaluate their effectiveness in classifying images of cats and dogs. The research focuses on assessing their classification accuracy, computational efficiency, and generalization capabilities.

The methodology involved preprocessing a dataset of labeled images, extracting relevant features, and training models using both approaches. The Decision Tree model demonstrated moderate performance, achieving an accuracy of 57.52%, with a tendency to overfit due to its reliance on manually extracted features. In contrast, the CNN model outperformed traditional methods, achieving an accuracy of 87.23%, demonstrating its superior ability to learn spatial hierarchies and generalize to unseen data.

The study's findings confirm that CNNs significantly outperform Decision Trees in image classification tasks, particularly for complex, high-dimensional datasets. However, CNNs require substantial computational resources, making them less practical for low-power or real-time applications. Future research could explore hybrid models, transfer learning techniques, and optimization strategies to balance accuracy and computational efficiency.

These insights contribute to the ongoing discussion on optimizing machine learning models for image classification, highlighting the strengths and limitations of different approaches. The research emphasizes the need for efficient deep-learning architectures and adaptive models to enhance classification accuracy while mitigating computational costs.

3. Introduction

3.1. Background

Image classification is a fundamental task in computer vision, with applications spanning medical diagnosis, autonomous vehicles, and security systems. Traditional machine learning models, such as Decision Trees, have been widely used for classification problems. However, with the advent of deep learning, Convolutional Neural Networks (CNNs) have emerged as the state-of-the-art approach, significantly improving accuracy and generalization (LeCun et al., 1998).

3.2. Problem Statement

Accurately classifying images is crucial for various domains, yet traditional machine learning models struggle to capture complex patterns and spatial dependencies in images. Decision Trees, for instance, rely on manually engineered features and often suffer from overfitting when dealing with high-dimensional data (Quinlan, 1996). On the other hand, CNNs have demonstrated superior performance by learning hierarchical representations of images, but they come with higher computational costs (Krizhevsky et al., 2012). This study aims to compare the effectiveness of these two approaches in the context of a binary image classification task: distinguishing between images of dogs and cats.

3.3. Research Objectives

The primary objectives of this research are:

- To evaluate and compare the classification performance of Decision Trees and CNNs.
- To analyze the advantages and limitations of each approach.
- To assess the computational efficiency and resource requirements of both models.
- To provide insights into future directions for improving image classification tasks.

3.4. Research Significance

Understanding the comparative strengths and weaknesses of Decision Trees and CNNs in image classification can provide valuable insights for researchers and practitioners in computer vision. While CNNs are known for their high accuracy, they require substantial computational resources, making them less feasible for resource-constrained environments. By exploring the trade-offs between accuracy, interpretability, and efficiency, this study contributes to the ongoing discussion on optimizing machine learning models for real-world applications (Hinton et al., 2012).

3.5. Structure of the Paper

This paper is structured as follows: Section 2 presents a literature review covering existing work on image classification using Decision Trees and CNNs. Section 3 describes the methodology, including dataset details, preprocessing techniques, and model training procedures. Section 4 presents the experimental results, followed by a discussion and conclusion in Sections 5 and 6, respectively. Lastly, Section 7 outlines potential directions for future research.

4. Literature Review:

4.1. Introduction

Image classification is a fundamental task in computer vision that involves assigning labels to images based on their content. Traditional machine learning algorithms, such as Decision Trees (DTs) and Support Vector Machines (SVMs), have been widely used for image classification. However, with the rise of deep learning, Convolutional Neural Networks (CNNs) have become the dominant approach, providing significantly higher accuracy and better generalization capabilities (Masita et al., 2025). This literature review explores the advantages and limitations of DTs and CNNs, the role of preprocessing in image classification, and methods for improving CNN performance.

4.2. Decision Trees in Image Classification

Decision Trees (DTs) are widely used in classification tasks due to their interpretability, low computational cost, and ability to handle structured data (Trojani et al., 2024). DTs operate by recursively partitioning datasets based on feature thresholds, creating a hierarchical structure that facilitates decision-making. One of the main advantages of DTs is their transparency—each decision rule is explicit, making it easy to understand how classifications are made.

Despite these advantages, DTs have several limitations in image classification. They struggle to capture spatial dependencies within images, as they treat pixel values independently rather than considering their spatial relationships. Additionally, DTs are prone to overfitting, particularly when applied to high-dimensional image data (Masita et al., 2025). This limitation becomes evident when comparing DTs to more advanced models such as CNNs, which learn hierarchical representations of visual patterns.

4.3. The Superiority of CNNs in Image Classification

Convolutional Neural Networks (CNNs) have revolutionized image classification by automatically learning relevant features from raw images. Unlike DTs, which rely on manually engineered features, CNNs extract spatial features at multiple levels of abstraction, allowing them to recognize textures, edges, and object structures efficiently (Abushawish et al., 2024). CNN architectures typically consist of convolutional layers, pooling layers, and fully connected layers, each playing a crucial role in feature extraction and classification.

A comparative study found that CNNs achieved over 98% accuracy in object detection tasks, whereas DTs and other traditional classifiers, such as SVMs, achieved only 86% on the same dataset (Masita et al., 2025). This highlights CNNs' superior generalization ability, particularly when images contain complex textures and occlusions.

However, CNNs also have notable drawbacks. Training deep CNNs requires significant computational power, as these models involve millions of parameters. Additionally, CNNs are highly data-dependent, requiring large labeled datasets to perform effectively. When datasets are small, CNNs are prone to overfitting, leading researchers to explore data augmentation, transfer learning, and regularization techniques to mitigate this issue (Trojani et al., 2024).

4.4. The Role of Preprocessing and Feature Engineering in Image Classification

Preprocessing plays a crucial role in image classification, significantly impacting the performance of both traditional machine learning models and deep learning-based classifiers. Research suggests that appropriate preprocessing techniques can improve model accuracy by up to 15% (Abushawish et al., 2024). Key preprocessing steps include normalization, resizing, contrast enhancement, and data augmentation.

For traditional classifiers like DTs and SVMs, feature extraction is an essential step. Techniques such as Histogram of Oriented Gradients (HOG), Scale-Invariant Feature Transform (SIFT), and Local Binary Patterns (LBP) have been widely used to enhance classification accuracy (Trojani et al., 2024). The success of classical algorithms often depends on the effectiveness of feature extraction before classification.

In contrast, CNNs automatically generate relevant features through convolutional layers. However, preprocessing techniques such as pixel normalization and contrast enhancement can still improve model performance and facilitate faster learning (Masita et al., 2025). A study on medical imaging found that applying Contrast-Limited Adaptive Histogram Equalization (CLAHE) improved CNN performance by 5–10% (Abushawish et al., 2024).

4.5. Enhancing CNN Performance: Regularization and Transfer Learning

Due to CNNs' reliance on large datasets, several techniques have been developed to improve their performance when data is limited. Transfer learning, which involves using pre-trained models such as ResNet, VGG16, or EfficientNet, has been shown to increase classification accuracy by 20–30% compared to training from scratch (Masita et al., 2025). This approach allows CNNs to leverage previously learned representations, reducing the need for extensive labeled datasets.

Another key method is regularization, which helps prevent overfitting. The most common regularization techniques include Dropout, Batch Normalization, and L2 Regularization. Dropout randomly deactivates neurons during training, encouraging the network to develop more robust feature representations (Abushawish et al., 2024). Additionally, studies indicate that incorporating Batch Normalization can improve training stability and accelerate learning rates (Trojani et al., 2024).

Other approaches include Data Augmentation, which artificially expands datasets by applying transformations such as rotations, flips, and cropping. Advanced augmentation techniques such as Cutout, Mixup, and AutoAugment have been shown to enhance CNN accuracy by 7–10% on challenging datasets (Masita et al., 2025).

4.6. CNNs vs. Classical Algorithms: A Comparative Analysis

Several studies have benchmarked CNN performance against traditional classifiers such as DTs, SVMs, and Random Forests. The general consensus is that CNNs outperform these methods in most real-world applications. For example, a study on fruit image classification found that CNNs achieved 98.35% accuracy, whereas an SVM classifier only reached 86.11% (Masita et al., 2025). Similarly, in medical imaging, CNNs demonstrated superior feature extraction and classification capabilities, particularly when dealing with high-resolution images (Abushawish et al., 2024).

Nonetheless, traditional classifiers remain relevant in specific scenarios. When datasets are small or when interpretability is a key requirement, decision trees and hybrid models (such as CNN-SVM combinations) can provide effective solutions (Trojani et al., 2024). A recent study found that using CNNs as feature extractors combined with an SVM classifier yielded competitive results while reducing computational costs (Masita et al., 2025).

4.7. Conclusion

While CNNs provide the highest accuracy in image classification, they require large datasets and significant computational resources. On the other hand, decision trees and classical classifiers remain useful when data is limited or when interpretability is a priority.

Research suggests that applying techniques such as transfer learning, data augmentation, and regularization can mitigate CNNs' limitations and improve their performance in data-constrained environments. In the future, advancements such as Vision Transformers (ViTs) may offer new alternatives for tackling complex image classification challenges (Abushawish et al., 2024).

5. Methodology

5.1. Dataset Description

This study utilizes the Dogs vs. Cats dataset, originally provided by Kaggle, which consists of 25,000 labeled images of dogs and cats. Each image belongs to one of two categories: dog or cat, making this a binary classification task. The dataset is balanced, ensuring equal representation of both classes, which mitigates issues related to class imbalance (Kaggle, n.d.).

5.2. Data Preprocessing

Prior to model training, a series of preprocessing steps were applied to standardize the dataset and improve model performance:

- Resizing: All images were resized to 64×64 pixels to maintain uniform input dimensions.
- Normalization: Pixel values were scaled to a [0,1] range to facilitate gradient-based learning.

- Data Augmentation: Transformations such as rotation, horizontal flipping, and brightness adjustments were applied to increase data variability and improve generalization (Shorten & Khoshgoftaar, 2019).

- The application of PCA reduced the dimensionality of the feature space, allowing for faster training of the Decision Tree model. However, the classification performance remained relatively low, with an accuracy of 57.22%. This suggests that while PCA aids in computational efficiency, decision trees may still struggle with image-based classification tasks that require hierarchical feature extraction.

5.3. Data Splitting

The dataset was partitioned into three subsets:

- Training Set (60%) – Comprising 15,000 images, used for model learning.
- Validation Set (20%) – Containing 5,000 images, used for hyperparameter tuning.
- Test Set (20%) – Consisting of 5,000 images, used for final evaluation.

The splitting process was performed using the train_test_split function, ensuring a stratified distribution of labels across subsets (Pedregosa et al., 2011).

5.4. Model Selection and Training

5.4.1 Baseline Model: Decision Tree Classifier

A Decision Tree Classifier was employed as a baseline for comparison with deep learning models. Decision trees are known for their interpretability and low computational cost but often struggle with high-dimensional data (Quinlan, 1996). The following steps were applied:

- Feature Extraction: Images were converted to grayscale and flattened into 4,096-dimensional vectors (64×64 pixels).
- Model Configuration: The classifier was implemented using DecisionTreeClassifier(max_depth=10, random_state=42).
- Evaluation:
  - Validation Accuracy: 57.84%
  - Test Accuracy: 57.52%

  - Limitations: The model struggled with high dimensionality and failed to capture spatial features, leading to suboptimal performance (Hastie et al., 2009).

5.4.2 Convolutional Neural Network (CNN)

A deep learning-based CNN model was implemented to enhance classification accuracy. CNNs are particularly effective for image-related tasks due to their ability to automatically extract spatial features (LeCun et al., 1998).

- Architecture:
  - Convolutional Layers – Extract hierarchical spatial features from images.
  - MaxPooling Layers – Reduce spatial dimensions and computational complexity.
  - Fully Connected Layers – Convert learned features into class probabilities.
  - Softmax Activation – Outputs the probability distribution between the dog and cat labels.
- Optimization & Regularization:
  - Adam optimizer (Kingma & Ba, 2015) was used for adaptive learning rate adjustment.
  - Dropout (0.5) was applied to mitigate overfitting (Srivastava et al., 2014).
  - Batch Normalization was incorporated to stabilize learning (Ioffe & Szegedy, 2015).
- Evaluation:
  - Validation Accuracy: ~89%
  - Test Accuracy: ~87%
  - Key Improvements: The CNN significantly outperformed the Decision Tree, demonstrating its ability to capture spatial dependencies and generalize well to new images.

5.4.3 Fine-Tuning on Misclassified Samples

After the initial CNN training, a confusion matrix was utilized to analyze the model's classification errors on the test set. Specific misclassified examples were identified and subsequently merged back with the original training dataset. Additional training, known as fine-tuning, was performed on this augmented dataset. This targeted fine-tuning approach allowed the CNN model to focus explicitly on challenging or ambiguous examples, thereby enabling the model to correct its previous misclassifications and enhance its generalization and predictive performance.

5.4.4 Hyperparameter Tuning

Hyperparameter tuning was conducted systematically to optimize the CNN model's performance and ensure stable convergence. Three primary hyperparameters were examined: learning rate, batch size, and dropout rate.

First, various learning rates (0.1, 0.01, and 0.001) were tested. Results showed that the highest learning rate (0.1) led to unstable training and poor performance (accuracy ~51.5%), whereas the lowest learning rate (0.001) demonstrated stable convergence and significantly improved accuracy (validation accuracy of 83.20%). Consequently, a learning rate of 0.001 was chosen for subsequent experiments.

Further experimentation involved different batch sizes (16, 32, 64). A small batch size (16) achieved high training accuracy (95.45%) but demonstrated signs of overfitting, with validation accuracy dropping to 80.58%. In contrast, a large batch size (64) produced more stable generalization, reaching a validation accuracy of 82.47%, yet resulted in slower convergence. A moderate batch size (32) was therefore selected to balance computational efficiency with generalization performance.

Lastly, different dropout rates (0.3, 0.5, 0.7) were explored. The lower dropout rate (0.3) showed slight signs of overfitting, while the higher dropout rate (0.7) negatively impacted training speed and reduced model accuracy. The intermediate dropout rate (0.5) provided an optimal balance, effectively preventing overfitting while maintaining strong predictive performance, and was therefore adopted for the final CNN model.

5.5. Performance Metrics

To comprehensively evaluate model performance, the following metrics were employed:

- Accuracy – Measures the proportion of correctly classified images.
- Precision & Recall – Provide insights into the model's ability to distinguish between classes.
- F1-score – Balances precision and recall to handle class imbalance.

- ROC-AUC – Assesses classification confidence and threshold behavior (Fawcett, 2006).

5.6. Tools and Libraries

The models were implemented using widely adopted machine learning and deep learning libraries:

- TensorFlow / Keras – Deep learning framework for CNN model development.
- OpenCV – Image preprocessing and augmentation.
- scikit-learn – Decision Tree implementation and performance metrics.
- matplotlib / seaborn – Data visualization.

5.7. Summary of Methodology

1. Preprocessing – Image resizing, normalization, and augmentation to enhance model robustness.
2. Dataset Splitting – Ensuring a structured split between training, validation, and test sets.
3. Baseline Model (Decision Tree) – Provided an initial performance benchmark (~57% accuracy).
4. Deep Learning Model (CNN) – Achieved superior accuracy (~87%), demonstrating the advantage of deep feature extraction.
5. Evaluation Metrics – Accuracy, precision, recall, F1-score, and AUC-ROC.
6. Implementation Tools – TensorFlow, OpenCV, and scikit-learn for model training and evaluation.

6. Results Analysis

6.1. Performance Comparison of Models

This study evaluates the performance of two classification models: a Decision Tree Classifier and a Convolutional Neural Network (CNN). The evaluation is based on accuracy, precision, recall, F1-score, and AUC-ROC, which provide insights into each model’s classification capabilities.

6.1.1 Decision Tree Classifier

The Decision Tree model was trained on grayscale, flattened images, treating pixel intensity values as features. The results are as follows:

- Validation Accuracy: 57.84%
- Test Accuracy: 57.52%
- Precision: 55.92%
- Recall: 63.84%
- F1-score: 59.62%
- AUC-ROC: 57.63%

The model's low accuracy and AUC-ROC score suggest that Decision Trees are not well-suited for complex image classification tasks, primarily due to their inability to capture spatial dependencies within images. Furthermore, the higher recall compared to precision indicates that the model is more prone to false positives, highlighting its struggle to effectively separate the two classes (Pedregosa et al., 2011).

6.1.2 Convolutional Neural Network (CNN)

The CNN model was trained using a series of convolutional, pooling, and fully connected layers. Data augmentation and regularization techniques were applied to improve generalization. The results are as follows:

- Validation Accuracy: 89.37%
- Test Accuracy: 87.23%
- Precision: 87.90%
- Recall: 86.75%
- F1-score: 87.32%
- AUC-ROC: 91.56%

Unlike Decision Trees, which required PCA to handle the high-dimensional input, CNNs naturally learn feature representations through convolutional layers. This difference highlights the advantage of deep learning in image classification, where hierarchical feature extraction is crucial. The CNN significantly outperformed the Decision Tree, achieving an accuracy increase of nearly 30%. The high AUC-ROC score suggests that the CNN can confidently differentiate between the two classes. The balanced precision and recall values indicate that the model maintains a strong trade-off between identifying dogs and cats while minimizing false positives (LeCun et al., 1998).

6.2. Evaluation Metrics Analysis

Model	Accuracy	Precision	Recall	F1-score	AUC-ROC
Decision Tree	57.52%	55.92%	63.84 %	59.62%	57.63%
CNN	87.23%	87.90%	86.75 %	87.32%	91.56%

These results reaffirm that deep learning techniques, such as CNNs, are far superior in image classification tasks compared to traditional decision tree-based methods (Hastie, Tibshirani, & Friedman, 2009).

6.3. Model Learning Curves

Examining the training and validation loss curves for CNN:

- The training and validation loss steadily decreased, indicating stable learning.
- No significant overfitting was observed, likely due to Dropout and Batch Normalization techniques (Ioffe & Szegedy, 2015).
- The CNN model's ability to generalize well on unseen data is evident from the minimal gap between training and validation loss (Srivastava et al., 2014).

Initially, the CNN model exhibited significant overfitting, as indicated by the disparity between training and validation loss curves. The training loss decreased steadily, while validation loss plateaued at a higher level, suggesting that the model was memorizing the training data rather than generalizing to unseen samples. Overfitting was further evident in the sharp drop in validation accuracy after several epochs.

To mitigate this issue, data augmentation techniques were employed, including random rotations, horizontal flipping, brightness adjustments, and zoom transformations. These techniques effectively expanded the dataset, providing more diverse training samples and improving the model's generalization ability. After implementing data augmentation, the gap between training and validation loss significantly decreased, confirming its effectiveness in reducing overfitting.

6.4. Misclassification Analysis

To understand where the models struggled, we analyzed misclassified images:

- Decision Tree Errors: The Decision Tree often misclassified low-contrast or partially occluded images.
- CNN Errors: The CNN misclassified images where dogs and cats had similar textures or colors (e.g., black-furred animals). Despite this, the model still performed well in distinguishing features like ears and whiskers.

6.5. Key Insights and Implications

- CNNs capture spatial relationships, leading to significant improvements in classification performance.
- Data augmentation played a crucial role in preventing overfitting and improving model generalization (Shorten & Khoshgoftaar, 2019).
- The Decision Tree model was ineffective for image-based classification due to its reliance on raw pixel intensity rather than learned feature representations.
- CNNs remain computationally expensive, requiring specialized hardware (GPUs) for efficient training and inference.
- A key finding in this study was the impact of **overfitting** on the initial CNN model. Without data augmentation, the model memorized training samples, leading to poor generalization. The addition of augmentation techniques improved performance by preventing the model from relying too heavily on specific training images, thereby enhancing its ability to classify unseen data accurately.

6.5.1 Fine-Tuning vs. Data Augmentation

An additional fine-tuning step was performed by retraining the CNN on a combined dataset that included previously misclassified samples. However, this approach did not mitigate the overfitting observed in the initial model. In contrast, the use of data augmentation—incorporating rotations, flips, and brightness adjustments—demonstrated a substantial impact on model performance. Specifically, data augmentation effectively eliminated the overfitting issue, as indicated by the significantly reduced gap between the validation and test performance metrics, resulting in very close values and improved generalization (Shorten & Khoshgoftaar, 2019).

6.6. Summary of Findings

- The CNN model outperformed the Decision Tree across all evaluation metrics, confirming that deep learning methods are more effective for image classification tasks.
- Feature learning through convolutional layers enables the CNN to generalize well, reducing misclassifications.
- Future improvements could involve Transfer Learning using pre-trained models (e.g., ResNet, EfficientNet) to further enhance performance and reduce training time (Kingma & Ba, 2015).

6.7. Conclusion

This study demonstrates the advantages of deep learning models in visual recognition tasks. While Decision Trees provide a baseline for comparison, their performance is suboptimal, reinforcing the need for feature-learning models such as CNNs. Future work will explore hybrid models and ensemble learning techniques to optimize performance further.

References (APA 7th Edition)

- Fawcett, T. (2006). An introduction to ROC analysis. Pattern Recognition Letters, 27(8), 861-874. https://doi.org/10.1016/j.patrec.2005.10.010
- Hastie, T., Tibshirani, R., & Friedman, J. (2009). The elements of statistical learning: Data mining, inference, and prediction. Springer Science & Business Media.
- Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. International Conference on Machine Learning (ICML), 448–456.
- Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

- LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11), 2278-2324.
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12, 2825-2830.
- Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. Journal of Big Data, 6(1), 60. https://doi.org/10.1186/s40537-019-0197-0
- 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(1), 1929-1958.

7. Discussion and Conclusions

7.1. Discussion

This study explored the effectiveness of two distinct machine learning approaches—Decision Trees and Convolutional Neural Networks (CNNs)—in the task of image classification for distinguishing between dogs and cats. The findings demonstrate the superiority of CNNs, which significantly outperformed Decision Trees in terms of classification accuracy and robustness (Krizhevsky, Sutskever, & Hinton, 2012; LeCun, Bottou, Bengio, & Haffner, 1998).

7.1.1 Performance Analysis

The results showed that the Decision Tree model achieved an accuracy of 57.52%, indicating its limitation in handling complex image features. The model's reliance on raw pixel intensity values rather than spatial feature extraction contributed to its poor generalization. In contrast, the CNN model achieved an accuracy of 87.23%, highlighting its capability to automatically extract hierarchical features and capture spatial dependencies in images (Hastie, Tibshirani, & Friedman, 2009).

The CNN model's high AUC-ROC score of 91.56% suggests strong discrimination between classes, while the Decision Tree model's AUC-ROC of 57.63% indicates that its performance is only marginally better than random guessing. These findings align with previous research that underscores CNNs as the state-of-the-art approach for image classification tasks (Fawcett, 2006).

7.1.2 Overfitting and Generalization

The learning curves of the CNN model revealed no significant overfitting, likely due to Dropout and Batch Normalization techniques (Ioffe & Szegedy, 2015; Srivastava et al., 2014). However, the model's dependency on large datasets and computational resources remains a notable challenge. Transfer Learning could serve as a potential strategy to mitigate these issues by leveraging pre-trained models such as ResNet or VGG16 (Howard et al., 2017; Tan, Pang, & Le, 2019).

The Decision Tree model exhibited clear signs of overfitting due to its tendency to memorize training examples rather than generalize well to unseen data. This behavior is characteristic of tree-based models, which often struggle with high-dimensional data like images (Quinlan, 1996).

7.1.3 Computational Efficiency

While CNNs provide superior accuracy, they require significantly more computational power compared to Decision Trees. CNN training demands GPU acceleration and high-memory environments, making it less accessible for users with limited hardware capabilities. In contrast, Decision Trees are lightweight and computationally efficient, making them more suitable for scenarios with constrained resources (Pedregosa et al., 2011; Han, Mao, & Dally, 2015).

7.2. Conclusions

This study confirms that CNNs are the preferred approach for image classification tasks, particularly for problems requiring hierarchical feature extraction. The key takeaways from this study include:

1. CNNs outperform Decision Trees in accuracy and robustness, making them the optimal choice for complex image classification tasks (Krizhevsky et al., 2012).
2. Feature extraction is crucial for improving classification accuracy. While CNNs learn features automatically, traditional models like Decision Trees struggle without explicit feature engineering (LeCun et al., 1998).
3. Regularization techniques such as Dropout and Batch Normalization help mitigate overfitting and improve generalization to new data (Srivastava et al., 2014).

4. CNNs require significant computational resources, whereas Decision Trees are lightweight and suitable for constrained environments (Pedregosa et al., 2011).

7.3. Future Directions

While CNNs provided strong classification performance, future research could explore the following areas to further enhance accuracy and efficiency:

- Transfer Learning: Utilizing pre-trained models to improve performance with limited data (Howard et al., 2017).
- Hybrid Models: Combining CNN feature extraction with alternative classifiers such as Support Vector Machines (SVMs) to improve decision-making (Tang, 2013).
- Optimized Architectures: Investigating lightweight CNN models, such as MobileNet or EfficientNet, for enhanced efficiency without sacrificing accuracy (Tan et al., 2019).

7.4. Final Remarks

This research reaffirms the dominance of deep learning in computer vision tasks and highlights the limitations of traditional machine learning models when applied to image classification. As computational resources continue to improve, CNN-based approaches will remain at the forefront of visual recognition tasks, paving the way for future innovations in artificial intelligence and machine learning.

8.Future Work

8.1. Introduction

The current study demonstrated the advantages of Convolutional Neural Networks (CNNs) over Decision Trees in image classification tasks. While CNNs showed superior accuracy and feature extraction capabilities, there remain areas for further improvement. This section outlines potential directions for future research aimed at enhancing model performance, efficiency, and applicability.

8.2. Enhancing Model Performance

One key area for future research is improving model performance through Transfer Learning. Pre-trained models such as ResNet, EfficientNet, or Vision Transformers (ViTs) have been shown to improve accuracy and reduce training time (Dosovitskiy et al., 2021). Fine-tuning such models on domain-specific datasets could yield significant benefits in classification tasks (Kornblith et al., 2019).

Another avenue is hyperparameter optimization using automated techniques such as Bayesian Optimization or Neural Architecture Search (NAS). These approaches can systematically improve model performance by selecting optimal parameters, reducing manual trial-and-error (Liu et al., 2018).

8.3. Improving Computational Efficiency

CNNs require significant computational resources, which can be a limitation for real-world applications. Future research should explore:

- Lightweight Architectures: Investigating efficient models such as MobileNet and EfficientNet-Lite that achieve high accuracy with reduced computational cost (Howard et al., 2017).
- Quantization and Pruning: Techniques that compress models and improve inference speed without sacrificing accuracy (Han et al., 2015).
- Edge and Embedded AI: Implementing CNNs on edge devices to enable real-time processing with minimal energy consumption (Tan et al., 2019).

8.4. Expanding Dataset Generalization

One challenge in deep learning is the generalization of models to new datasets. To address this, future work can focus on:

- Data Augmentation Techniques: Employing adversarial training and synthetic data generation to improve robustness (Shorten & Khoshgoftaar, 2019).
- Few-Shot and Self-Supervised Learning: Reducing the need for large labeled datasets by leveraging self-supervised learning approaches such as contrastive learning (Chen et al., 2020).

8.5. Hybrid Models and Alternative Classifiers

The combination of CNNs with alternative classification techniques may offer further advantages:

- CNN + SVM Hybrid Models: Using CNNs for feature extraction while employing Support Vector Machines (SVMs) for classification (Tang, 2013).
- Ensemble Learning: Combining multiple models to enhance generalization and reduce overfitting (Hansen & Salamon, 1990).

8.6. Ethical Considerations and Bias Mitigation

As deep learning models become widely used, future work should emphasize ethical considerations:

- Bias Detection and Fairness: Ensuring that models are trained on diverse datasets to minimize bias in predictions (Buolamwini & Gebru, 2018).
- Explainable AI (XAI): Developing methods for interpreting model decisions to increase transparency and trust (Ribeiro et al., 2016).

8.7. Conclusion

Future research in deep learning-based image classification can significantly enhance model accuracy, efficiency, and fairness. Leveraging transfer learning, lightweight architectures, hybrid models, and ethical AI practices will help advance the field and expand the applicability of CNNs to diverse real-world problems.

9. References

References (APA 7th Edition)

Abushawish, H., et al. (2024). Deep learning in automatic diabetic retinopathy detection and grading systems: A comprehensive survey and comparison of methods. IEEE Access. https://doi.org/10.1109/ACCESS.2024.3415617

Buolamwini, J., & Gebru, T. (2018). Gender shades: Intersectional accuracy disparities in commercial gender classification. Proceedings of the Conference on Fairness, Accountability, and Transparency, 77-91.

Chen, T., Kornblith, S., Norouzi, M., & Hinton, G. (2020). A simple framework for contrastive learning of visual representations. International Conference on Machine Learning (ICML), 1597-1607.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., ... & Houlsby, N. (2021). An image is worth 16x16 words: Transformers for image recognition at scale. International Conference on Learning Representations (ICLR).

Fawcett, T. (2006). An introduction to ROC analysis. Pattern Recognition Letters, 27(8), 861-874. https://doi.org/10.1016/j.patrec.2005.10.010

Hansen, L. K., & Salamon, P. (1990). Neural network ensembles. IEEE Transactions on Pattern Analysis and Machine Intelligence, 12(10), 993-1001.

Hastie, T., Tibshirani, R., & Friedman, J. (2009). The elements of statistical learning: Data mining, inference, and prediction. Springer Science & Business Media.

Han, S., Mao, H., & Dally, W. J. (2015). Deep compression: Compressing deep neural networks with pruning, trained quantization, and Huffman coding. arXiv preprint arXiv:1510.00149.

Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2012). Improving neural networks by preventing co-adaptation of feature detectors. arXiv preprint arXiv:1207.0580.

Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., ... & Adam, H. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861.

Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. International Conference on Machine Learning (ICML), 448–456.

Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

Kornblith, S., Shlens, J., & Le, Q. V. (2019). Do better ImageNet models transfer better? Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2661-2671.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. Advances in Neural Information Processing Systems (NeurIPS), 25, 1097–1105.

LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11), 2278-2324.

Liu, H., Simonyan, K., & Yang, Y. (2018). DARTS: Differentiable architecture search. arXiv preprint arXiv:1806.09055.

Masita, A., et al. (2025). Deep learning in defects detection of PV modules: A review. Solar Energy Journal. https://doi.org/10.1016/j.seja.2025.100090

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12, 2825-2830.

Quinlan, J. R. (1996). Improved use of continuous attributes in C4.5. Journal of Artificial Intelligence Research, 4, 77-90.

ResheTech. (n.d.). Machine learning vs deep learning for image recognition. Retrieved from https://reshetech.co.il/machine-learning-tutorials/cnn-model-image-classification

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "Why should I trust you?" Explaining the predictions of any classifier. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 1135-1144.

Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. Journal of Big Data, 6(1), 60. https://doi.org/10.1186/s40537-019-0197-0

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(1), 1929-1958.

Tan, M., Pang, R., & Le, Q. V. (2019). EfficientDet: Scalable and efficient object detection. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 10781-10790.

Tang, Y. (2013). Deep learning using linear support vector machines. arXiv preprint arXiv:1306.0239.

Trojani, C., et al. (2024). Impact of preprocessing parameters in medical imaging-based radiomic studies. Cancers. https://doi.org/10.3390/cancers16152668