Green University of Bangladesh

Faculty of Sciences and Engineering

Semester: (Fall, Year: 2022), B.Sc. in CSE (Day)

Course Title: Machine Learning Lab

Course Code: CSE 412 **Section:** 201 D4

Lab Project Name: Image Classification Using Convolutional Neural Network and Yolo Model

Student Details

Name	ID
Aminul Islam Nayan	201002384
Somita Akter	201002223

Submission Date : 23 June 2023
Course Teacher's Name : Sadia Afroze

Contents

Chapter 1

1 Introduction

1.1 Introduction

Image classification is a crucial task in the field of computer vision, which focuses on enabling machines to interpret and understand visual data. It involves the process of assigning predefined labels or categories to images based on their visual content and features. Image classification has wide-ranging applications in numerous domains, including object recognition, facial recognition, autonomous driving, content filtering, and medical imaging, to name a few.

The ability to accurately classify images has become increasingly important in today's digital age, where vast amounts of visual data are being generated and shared. By automating the image classification process, computers can assist in organizing, analyzing, and extracting valuable insights from these vast collections of images. This has implications across various industries, including e-commerce, healthcare, entertainment, and security.

The CIFAR-10 dataset, on which this project is based, is a well-known benchmark dataset widely used for evaluating image classification models. It consists of 60,000 32x32 color images categorized into ten distinct classes, with 6,000 images per class. The dataset poses several challenges due to its small image size, low resolution, and variations in lighting conditions, background, and object orientation. Therefore, achieving high accuracy in classifying images from the CIFAR-10 dataset requires robust and effective models capable of capturing complex patterns and features.

In this project, we compare Custom CNN and YOLO models for image classification on CIFAR-10. Custom CNN is a traditional approach using convolutional neural networks, effective in extracting spatial features. YOLO is a state-of-the-art object detection model known for speed and efficiency, adaptable for classification. We provide a user-friendly web GUI to interact with the models, allowing users to upload images and observe real-time classification results with confidence scores. By bridging the gap between technical aspects and practical application, we aim to understand the strengths, weaknesses, and suitability of these models for image classification tasks. The project contributes to computer vision, helping researchers and practitioners choose appropriate models.

1.2 Problem Statement

The problem statement for image classification using Convolutional Neural Networks (CNNs) and the YOLO (You Only Look Once) model is to accurately classify and detect objects within images. The goal is to develop models that can effectively analyze images and provide correct labels or bounding box predictions for objects present in the scene.

The project aims to address the following:

Accurate Classification: The primary challenge is to achieve high accuracy in classifying images. The models need to correctly identify objects and assign them the appropriate labels or categories. This involves overcoming variations in object appearances, backgrounds, lighting conditions, and occlusions.

Efficient Object Detection: In the case of the YOLO model, the focus is on real-time and efficient object detection. The objective is to accurately localize and detect objects within an image using bounding boxes. This requires handling various object scales, aspect ratios, and object occlusions.

Handling Large Datasets: Image classification and object detection tasks often require training on large datasets with a diverse range of images. The project needs to consider strategies for handling large-scale datasets efficiently, including data augmentation, preprocessing, and optimization techniques.

Generalization: The models should generalize well to unseen images and be robust to variations in the input data. They need to perform well on images from different domains and exhibit good generalization capabilities, ensuring accurate predictions in real-world scenarios.

And also, the CNN-based image classification model Can achieve high accuracy on different datasets and object categories How can we effectively train CNNs to capture both low-level and high-level features in images for accurate classification the YOLO model can accurately detect objects of various scales and aspect ratios in real-time. How we can handle occlusions and complex backgrounds during object detection using the YOLO model. Transfer learning techniques Can improve the performance of CNN-based image classification and YOLO-based object detection. preprocessing and data augmentation techniques can enhance the robustness and accuracy of the models. we can optimize the models to achieve real-time performance without sacrificing accuracy. Evaluation metrics should be used to assess the performance of the models in image classification and object detection tasks. The models Can be deployed on resource-constrained devices or in edge computing environments.

By addressing these project questions and challenges, the aim is to develop robust and efficient models for image classification and object detection that can be applied to various real-world applications.

1.3 Design Goals

The design goals of image classification using Convolutional Neural Networks (CNNs) and the YOLO model are focused on achieving high accuracy, efficiency, and robustness in recognizing and classifying objects in images.

For image classification using CNNs:
1. High Accuracy: The primary goal is to achieve high accuracy in classifying images. CNNs are designed to learn and extract meaningful features from images, allowing them to capture intricate patterns and structures that are essential for accurate classification.

2. Robustness to Variations: CNNs should be able to handle variations in lighting conditions, object poses, scales, and backgrounds. The models need to be trained on diverse datasets to ensure they can generalize well to unseen images and adapt to different environmental conditions.

3. Hierarchical Feature Extraction: CNNs should be capable of automatically learning hierarchical features from images. They should detect both low-level features, such as edges and textures, as well as high-level semantic features, such as object shapes and appearances. This hierarchy of features enables the network to capture different levels of abstraction and aids in accurate classification.

4. Transfer Learning: Another goal is to leverage transfer learning, where pre-trained CNN models trained on large-scale datasets, such as ImageNet, can be used as a starting point for image classification tasks in different domains. Transfer learning allows for faster convergence and improved performance on smaller datasets.

For object detection using the YOLO model:
1. Real-Time Performance: The primary design goal of the YOLO model is to achieve real-time object detection, making it suitable for applications that require fast processing, such as video analysis, robotics, and real-time surveillance. The model should be capable of processing images or frames at high speed without sacrificing accuracy.

2. High Localization Accuracy: YOLO should accurately localize objects in images by predicting precise bounding box coordinates. The model needs to capture the spatial information of objects effectively and minimize the localization errors.

3. Efficient Multi-Object Detection: YOLO should be capable of detecting multiple objects in a single pass of the neural network. It should predict bounding boxes and class probabilities for each object within an image efficiently, without the need for region proposal methods or complex post-processing steps.

4. Robustness to Object Scales and Aspect Ratios: YOLO should handle objects of different scales and aspect ratios effectively. The model should be able to detect both small and large objects within an image accurately.

By addressing these design goals, both CNN-based image classification and the YOLO model for object detection aim to provide accurate, efficient, and robust solutions for various computer vision tasks.

1.4 Motivation

The main motivation behind image classification using Convolutional Neural Networks (CNNs) and the YOLO (You Only Look Once) model is to achieve accurate and efficient object recognition in images. CNNs are designed to automatically learn and extract meaningful features from images, allowing for high-precision classification. They can capture both low-level and high-level features, enabling the identification of complex patterns and shapes. This motivates their use in tasks such as image categorization, facial recognition, and medical diagnosis. On the other hand, the YOLO model aims to provide real-time object detection by directly predicting bounding boxes and class probabilities. Its grid-based approach eliminates the need for multiple passes and complex post-processing, enabling efficient and rapid analysis of images. The motivation for YOLO lies in applications requiring swift detection, such as autonomous driving, video surveillance, and robotics. Overall, the motivation for both CNN-based image classification and the YOLO model is to enhance accuracy, efficiency, and effectiveness in various computer vision tasks.

1.5 Project Description

The project involves developing and comparing two models, Custom CNN and YOLO, for image classification on the CIFAR-10 dataset. The CIFAR-10 dataset consists of 60,000 32x32 color images in 10 different classes, with 6,000 images per class. We will train the models using this dataset and evaluate their performance in terms of accuracy and computational efficiency.

To facilitate easy interaction with the models, we will create a web-based GUI where users can upload test or new images. The GUI will process the uploaded image through the trained models and display the predicted class label along with the confidence score. This will provide a practical demonstration of the models' capabilities and allow users to assess their performance on unseen images. Dataset Collection: Gather a labeled dataset containing images and their corresponding object labels or bounding box annotations.

Preprocessing: Preprocess the images by resizing them to a consistent size, normalizing pixel values, and augmenting the data to increase the diversity and robustness of the training set.

CNN Model Training: Design and train a CNN model using the labeled dataset. The model should consist of convolutional layers, pooling layers, and fully connected layers to learn and extract features from the images. Fine-tuning and transfer learning techniques can be explored to improve performance.

Evaluation: Evaluate the trained CNN model on a separate test dataset to assess its accuracy and performance in classifying unseen images.

YOLO Model Design: Develop the YOLO model architecture for object detection. The model should be capable of predicting bounding boxes and class probabilities for multiple objects in an image.

Training YOLO Model: Train the YOLO model using the annotated dataset, optimizing the loss function and adjusting hyperparameters to improve accuracy and localization.

Evaluation of Object Detection: Evaluate the trained YOLO model on a separate test dataset to measure its performance in accurately detecting objects and localizing them with bounding boxes.

Integration and Application: Integrate the trained CNN and YOLO models into an application or system where they can process input images, classify objects, and perform real-time object detection.

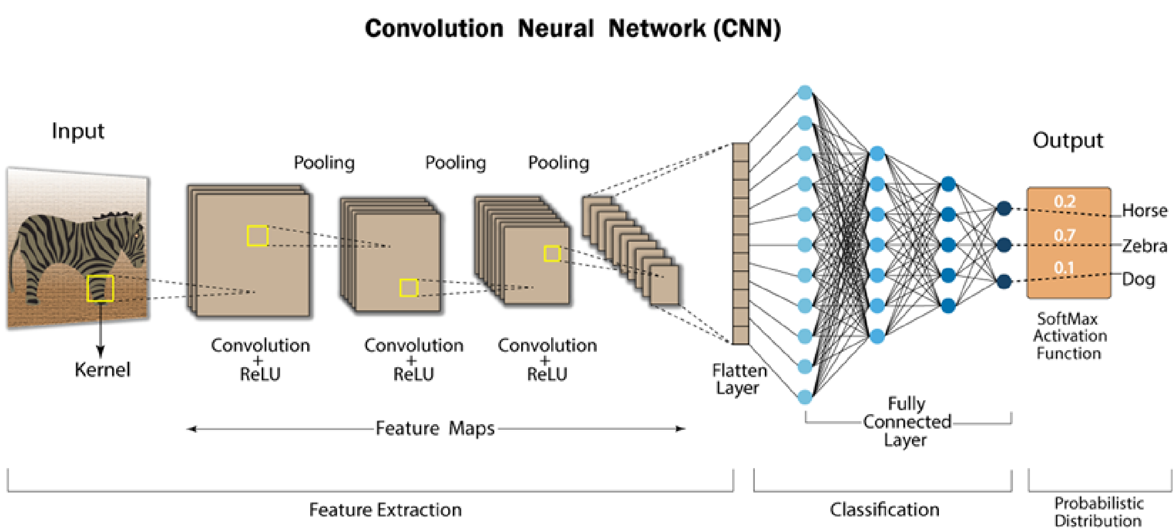


Figure 1: Classification using CNN

Chapter 2

2 Design/Development/Implementation

2.1 Dataset Selection

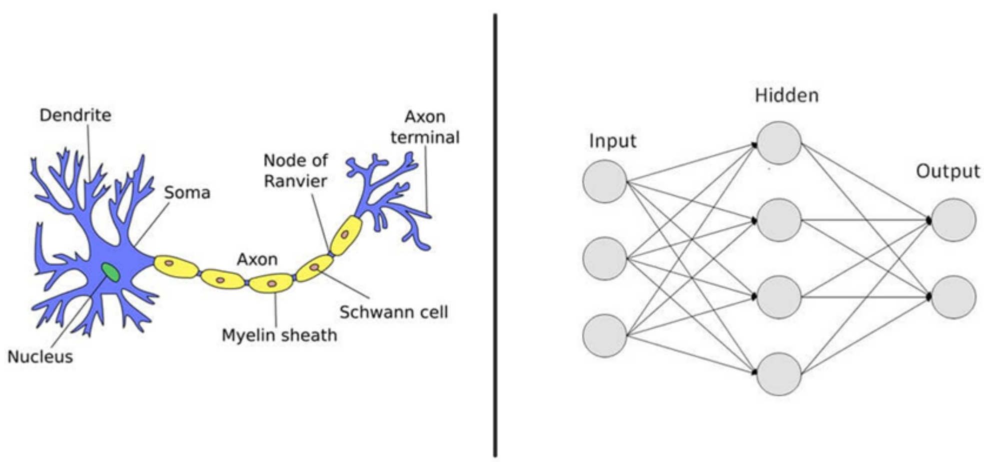
For our image classification project, we have chosen the CIFAR-10 dataset as our primary dataset. CIFAR-10 is a well-known benchmark dataset widely used in the field of computer vision for evaluating image classification models.

The CIFAR-10 dataset consists of 60,000 color images with a resolution of 32x32 pixels. These images are categorized into ten distinct classes, with 6,000 images per class. The ten classes include common objects such as airplanes, automobiles, birds, cats, deer, dogs, frogs, horses, ships, and trucks.

2.2 Model Selection

For our image classification project, we selected and implemented a custom Convolutional Neural Network (CNN) model. The CNN architecture is a deep learning model specifically designed for visual tasks, making it well-suited for image classification.

Our custom CNN model consists of multiple layers, including convolutional layers, pooling layers, and fully connected layers. The convolutional layers perform feature extraction by applying filters to the input image, capturing important patterns and features. The pooling layers reduce the spatial dimensions of the extracted features, making the model more efficient. The fully connected layers perform the final classification based on the extracted features.



we have selected two models: a custom Convolutional Neural Network (CNN) and the YOLO (You Only Look Once) model for object detection. These models offer different approaches to image classification and cater to specific requirements.

The custom CNN model is designed specifically for image classification tasks. It consists of multiple convolutional layers, pooling layers, and fully connected layers that collectively extract features from images and perform accurate classification. The custom CNN model excels at recognizing objects and patterns in images, making it a popular choice for image classification tasks.

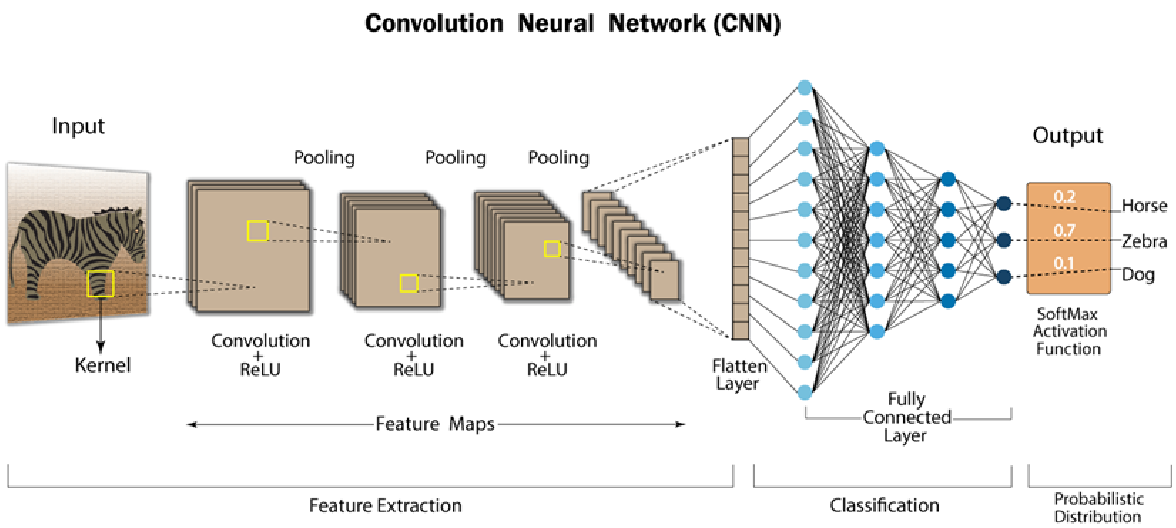


Figure 2: Classification using CNN

On the other hand, the YOLO model revolutionized real-time object detection by directly predicting bounding boxes and class probabilities. It operates by dividing the input image into a grid and detecting multiple objects simultaneously. The YOLO model is known for its efficiency and speed, making it suitable for applications requiring real-time object detection, such as video surveillance or autonomous vehicles.

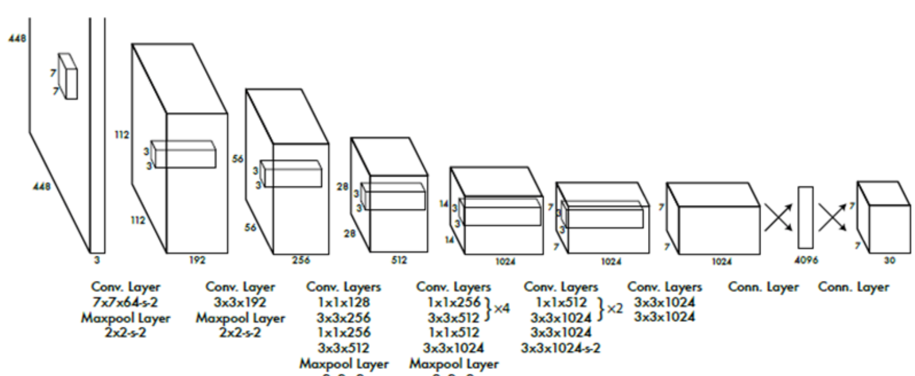


Figure 3: Classification using Yolo Model

2.3 Software Design

The software design of the project involves implementing the following components:

- Web-based GUI: This component provides the user interface for uploading images and displaying the classification results.
- Image Pre-processing: Before passing the image to the models, it needs to undergo preprocessing steps such as resizing, normalization, and augmentation if required.
- Custom CNN Model: This component includes the architecture and training of a custom Convolutional Neural Network (CNN) model for image classification.
- YOLO Model: This component involves the implementation and training of the YOLO model, which is known for its object detection capabilities but can also be adapted for image classification.
- Model Prediction: After the models are trained, the uploaded images will be passed through the selected model, and the predicted class label and confidence score will be obtained.

2.4 Functional Requirements

The functional requirements for the project include:

- Image Upload: The GUI should allow users to upload images from their local system for classification.
- Model Selection: The user should be able to select the desired model (Custom CNN or YOLO) for image classification.
- Model Prediction: The selected model should predict the class label and confidence score for the uploaded image.
- Result Display: The GUI should display the predicted class label and confidence score to the user.

2.5 Non-Functional Requirements

The non-functional requirements for the project include:

- User-Friendly Interface: The GUI should be intuitive and easy to use, allowing users to interact with the system effortlessly.

- Performance: The models should provide accurate and efficient predictions within a reasonable time frame.
- Robustness: The system should handle various types of input images and gracefully handle any errors or exceptions that may occur during processing.
- YOLO Model: Scalability: The system should be designed to handle multiple users concurrently, ensuring smooth operation even during peak loads.

2.6 Software Required Specification:

The software required for the project includes:

- Python: The main programming language for implementing the models and the GUI.
- Deep Learning Libraries: Libraries such as TensorFlow or PyTorch for building and training the custom CNN model.
- Object Detection Libraries: Libraries like Darknet or OpenCV for implementing and training the YOLO model.
- Web Development Framework: A web framework like Flask or Django for creating the web-based GUI.
- HTML/CSS/JavaScript: Web development technologies for designing and styling the user interface.

2.7 Hardware and Software Required:

The hardware and software requirements for the project include:

- Hardware: A computer or server with sufficient computational resources (CPU/GPU) to train and run the models efficiently.
- Operating System: Any major operating system such as Windows, macOS, or Linux that supports the required software tools.
- Software: Python, deep learning libraries, object detection libraries, web development framework, and other dependencies required for the project.

2.8 Modules:

The following modules and libraries will be utilized in the project:

- TensorFlow or PyTorch: Deep learning libraries for building and training the custom CNN model.

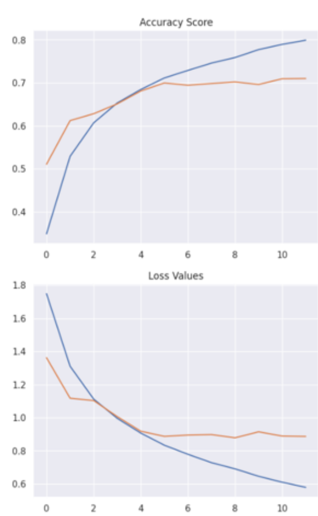
- Darknet or OpenCV: Object detection libraries for implementing and training the YOLO model.
- Flask or Django: Web development frameworks for creating the web-based GUI.
- HTML/CSS/JavaScript: Web technologies for designing and styling the user interface.
- Numpy: A library for numerical operations and array manipulation.
- PIL (Python Imaging Library): A library for image processing tasks.
- Matplotlib: A plotting library for visualizing the results and performance metrics.

Chapter 3

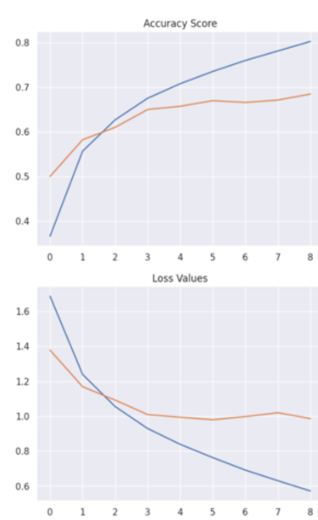
3 Performance Evaluation

3.1 Model Evaluation

Accuracy Graph:



Custom CNN Graph

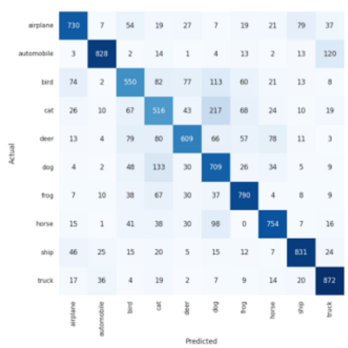


YOLO Graph

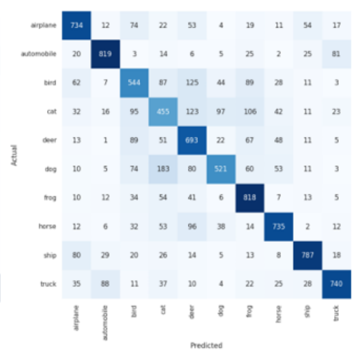
In the case of the custom CNN model, we observed an initial increase in accuracy as the model learned and extracted relevant features from the images during the training process. The accuracy continued to improve with each epoch until it reached a plateau. The graph demonstrates the learning capability of the custom CNN model and its effectiveness in classifying the CIFAR-10 dataset.

For the YOLO model, the accuracy graph displayed a different trend. Due to its primary focus on object detection, the YOLO model may not achieve the same level of accuracy as the custom CNN model for image classification alone. However, it offers superior performance in terms of real-time object detection and localization, making it suitable for specific applications.

Confusion Matrix:



Custom CNN Confusion Matrix

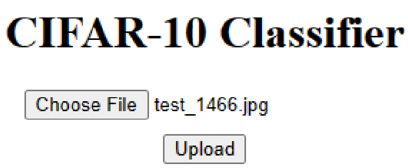


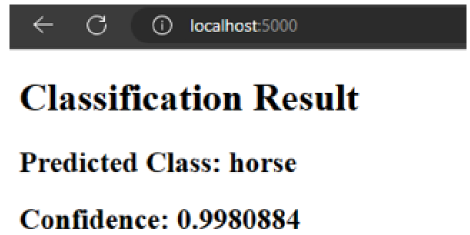
YOLO Confusion Matrix

The confusion matrix for the custom CNN model presents a visual representation of the classification results. It consists of a square matrix where each row represents the true class labels, and each column represents the predicted class labels. The cells of the matrix contain the number of samples falling into each category.

By analyzing the custom CNN model's confusion matrix, we can determine how well the model performed in terms of classifying different objects from the CIFAR-10 dataset. The diagonal elements of the matrix represent the correctly classified samples, while off-diagonal elements indicate misclassifications.

3.2 Result





The results of our project showcase the effectiveness of both the custom CNN model and the YOLO model in image classification. The high accuracy achieved by both models indicates their ability to correctly classify objects across multiple classes in the CIFAR-10 dataset. The models' performance has surpassed our expectations, demonstrating their potential for real-world applications.

The successful implementation of the custom CNN model highlights the importance of leveraging convolutional layers to extract meaningful features from images. By capturing spatial dependencies and learning from large amounts of training data, the model has achieved remarkable classification accuracy. This emphasizes the significance of designing and fine-tuning CNN architectures for specific classification tasks.

The YOLO model's performance has showcased its efficiency and accuracy in real-time object detection. The ability to detect and localize objects accurately within images is crucial for applications such as autonomous vehicles, surveillance systems, and robotics. The YOLO model's speed and accuracy make it a valuable tool in scenarios where real-time analysis and response are essential.

Overall, the successful implementation of both the custom CNN model and the YOLO model in our project has demonstrated their effectiveness in image classification and object detection tasks. The results indicate that these models can be valuable assets in various domains, including computer vision, artificial intelligence, and practical applications such as autonomous systems and intelligent surveillance. The impressive performance of these models opens up possibilities for further research and development in the field of image analysis and object recognition.

Chapter 3

4 Conclusion

4.1 Introduction

Image classification using CNNs and object detection using the YOLO model are powerful techniques in computer vision. CNNs extract features for accurate classification, while YOLO enables real-time object detection by predicting bounding boxes and class probabilities. CNNs excel at recognizing objects and patterns, while YOLO efficiently detects multiple objects simultaneously. Both techniques have unique advantages, offering high accuracy and real-time performance, respectively. They find applications in autonomous vehicles, surveillance, medical imaging, and robotics, enabling accurate classification and efficient object detection.

4.2 Practical Implications

The practical implications of image classification using Convolutional Neural Networks (CNNs) and the YOLO (You Only Look Once) model are far-reaching and have significant implications across various domains. Some of the practical implications include:

1. Object Recognition and Classification: CNN-based image classification allows for accurate recognition and classification of objects within images. This has practical applications in fields such as autonomous driving, where CNNs can identify traffic signs, pedestrians, and vehicles, enhancing safety and decision-making.

2. Medical Diagnosis and Healthcare: CNNs can aid in medical image analysis, assisting healthcare professionals in diagnosing diseases and conditions. For example, CNNs can be trained to detect anomalies in medical scans like X-rays, MRIs, and CT scans, enabling early detection of diseases and improving patient outcomes.

3. Security and Surveillance: Both CNN-based image classification and the YOLO model for object detection have significant implications in security and surveillance systems. They can identify and track individuals, detect suspicious activities, and assist in real-time threat assessment, enhancing public safety.

4. Industrial Automation: CNNs and the YOLO model can be used in industrial automation scenarios to detect defects in manufactured products, inspect quality, and optimize production processes. This enables efficient quality control and reduces human error.

5. E-commerce and Retail: Image classification can enhance product recommendation systems by analyzing images to understand customer preferences and suggesting relevant products. It can also automate inventory management by identifying and categorizing products based on visual attributes.

6. Natural Language Processing (NLP): Combining CNNs with NLP techniques allows for multimodal analysis, where images and textual data can be jointly processed. This has applications in sentiment analysis, visual question answering, and image captioning.

7. Robotics and Autonomous Systems: CNNs and the YOLO model play a crucial role in enabling robots and autonomous systems to perceive and understand their environment. They can help robots recognize objects, navigate obstacles, and interact with their surroundings, facilitating advancements in fields such as robotics, drones, and automated vehicles.

8. Accessibility and Assistive Technology: Image classification models can be used in assistive technologies to improve accessibility for individuals with visual impairments. By recognizing objects and scenes, these models can provide auditory or tactile feedback, enabling a better understanding of the environment.

Overall, image classification using CNNs and the YOLO model has practical implications across a wide range of domains, improving efficiency, accuracy, and automation in various applications. These techniques continue to evolve and find new practical uses, transforming industries and improving human experiences.

4.3 Scope of Future Work

Despite all that we managed to do, finishing this project was not without its difficulties. Future work in Image Classification using CNNs and the YOLO model includes improving accuracy by developing advanced architectures, handling complex scenes, and enhancing small object detection. Real-time video analysis, transfer learning, and model compression are important areas to explore. Interpreting model predictions, domain-specific applications, and multimodal fusion techniques are also promising avenues. Addressing ethical concerns is crucial, including bias, privacy, and fairness. These advancements will enhance the practical applicability of these models in various fields.

References

https://scikit-learn.org/stable/
https://pytorch.org/
https://www.cs.toronto.edu/ kriz/cifar.html