Practical Work 3: Convolutional Neural Networks and Computer Vision

Louis Fippo Fitime
Claude Tinku
Kerolle Sonfack

Department of Computer Engineering, ENSPY
University of Yaoundé I

October 8, 2025

Abstract

This third Practical Work focuses on Convolutional Neural Networks (CNNs) and their applications in Computer Vision. The objective is to understand convolutional architectures, residual learning, and advanced vision tasks such as image classification, segmentation, object detection, and neural style transfer using Keras.

1 Part 1: CNN Fundamentals

1.1 Theoretical Concepts

Convolution Operation

In a convolution operation, a **filter (kernel)** slides over the input image to extract local features such as edges, textures, or patterns. The kernel size determines the receptive field, while the **stride** controls how many pixels the filter moves at each step.

The main objective of a convolutional layer is to learn spatially local and hierarchical features while preserving the spatial structure of the image.

Pooling Operations

Pooling layers reduce the spatial dimensions of feature maps and improve robustness to small spatial variations.

- **Max Pooling** selects the maximum value in each region, highlighting the most salient features.
- **Average Pooling** computes the average value, resulting in smoother representations.

Pooling reduces computational cost and helps prevent overfitting.

From Image to Classification

After several convolutional and pooling layers, the extracted spatial feature maps are flattened into one-dimensional vectors. These vectors are then fed into fully connected (Dense) layers, which perform high-level reasoning and output class probabilities.

Residual Networks (ResNets)

Residual connections address the **vanishing gradient problem** in very deep networks. By adding skip connections, the network learns residual mappings instead of direct mappings, allowing gradients to flow more easily through deep architectures and enabling the training of much deeper networks.

2 Part 2: Basic CNN Implementation

2.1 Exercise 1: Classic CNN Architecture

The proposed CNN architecture consists of successive convolutional and pooling layers for feature extraction, followed by fully connected layers for classification.

Evaluation

After training for multiple epochs, the model is evaluated on the test dataset. The test accuracy provides an estimate of the model's generalization performance on unseen images.

2.2 Exercise 2: Introduction to Residual Blocks

Residual blocks introduce a shortcut connection that adds the input of a block to its output.

Advantage of Skip Connections

Adding the input x to the output of the convolutional path allows the network to learn identity mappings more easily. This improves gradient propagation, reduces training degradation, and enables deeper networks to achieve higher performance.

3 Part 3: Advanced Applications

3.1 Exercise 3: Recognition, Segmentation, and Detection

Image Segmentation

Unlike classification, which outputs a single class label, an image segmentation model produces a **pixel-wise classification map**. Each pixel is assigned a class label.

In U-Net architectures, **upsampling layers** recover spatial resolution by combining high-level semantic information with fine-grained spatial details from earlier layers.

Object Detection

Object detection extends image classification by predicting both the object class and its location using **bounding boxes**. A CNN can output coordinates (x, y, w, h) along with class probabilities, enabling precise object localization.

3.2 Exercise 4: Neural Style Transfer

Neural style transfer uses a pre-trained CNN (such as VGG16) to separate content and style representations of images.

Content and Style Losses

- **Content loss** ensures that the generated image preserves the high-level structure and semantics of the content image.
- **Style loss** measures the similarity of texture and patterns between the generated image and the style image, typically using Gram matrices.

The optimization process minimizes a weighted combination of content and style losses to produce a visually coherent stylized image.

4 Conclusion

This practical work introduced convolutional neural networks and their applications in computer vision. Through classification, residual learning, segmentation, detection, and style transfer, students gained a comprehensive understanding of modern CNN-based architectures and their real-world applications.