ARTIFICIAL INTELLIGENCE ALGORITHMS IN FACE RECOGNITION AND OBJECT DETECTION

Makhmudova Shakhzoda Yorkinovna
Tashkent University of information technologies named after Muhammad al Khwarizmi

Sharopova Barno Abdunabiyevna
Institute of counter-irrigation and agro technologies of the national research university

ABSTRACT

Facial recognition is a well-established and popular field in Computer Vision, especially with advancements in deep learning and data sets. Deep facial recognition has made significant progress and is widely applied in real-world scenarios. A complete facial recognition system involves three main components: facial recognition, orientation, and representation. This system detects faces, aligns them to a standard view, and extracts features for recognition using deep convolutional neural networks. This article provides a detailed overview of the latest advancements in these areas, showing how deep learning has greatly enhanced their abilities. Object detection in machine vision is a challenging area that requires significant improvements. While image classification accuracy is nearing 2.25%, surpassing human performance, object detection algorithms are still in the early stages. Current algorithms achieve only 40.8 MAPS on modern objects, so careful dataset selection is crucial for optimal results.

Introduction

Computer vision encompasses image recognition, teaching computers to recognize and respond to people. We frequently encounter facial recognition technology unknowingly. Image processing involves machine learning to analyze digital images, allowing AI systems to perform image recognition and classify objects efficiently. Unmanned vehicles use computer vision to identify pedestrians, signs, and other vehicles. The concept of "face recognition" is easily understood. It involves using computer vision algorithms to identify and validate the identity of a face in an image or video. The process can be broken down into three main steps: recognition locates a face in the image, analysis maps out facial features and converts them into a unique set of numbers, and identification determines the person's identity or categorizes them into a group based on characteristics like gender or age.

Pretrained Convolutional Neural Networks (CNN) are crucial in AI image recognition technology. To enhance performance for specific applications, appropriate training data must be collected, annotated, and used to retrain and fine-tune these models. Accuracy is the primary criterion for assessing image recognition tools, with speed and adaptability typically evaluated later.

Common CNN-based pretrained models for image recognition work include:

•A two-stage pretrained model that uses a CNN to produce candidate object regions, which are then passed through a separate CNN to classify images and refine bounding boxes. Accuracy is the key benefit, but it can take a long time to retrain.

•A one-stage model that uses a CNN to predict class labels and bounding boxes of objects in an image. The main advantage is fast inference time (or quick delivery) and low memory usage. On the downside, it is less accurate compared to Faster R-CNN.

•A one-stage model that uses a single CNN to predict bounding boxes and class labels of objects in an image. It has a good balance between accuracy and performance speed.

Algorithms for Face Recognition

There are multiple face recognition algorithms, such as ones using traditional, manually crafted features and those utilizing deep learning. Some of the most commonly used algorithms are: Eigenfaces uses PCA to extract features from face images, an early face recognition algorithm still used today. Fisherfaces is an extension that considers class labels, more robust to lighting and expression variations. DeepFace uses a deep CNN for feature extraction, achieving human-level performance. FaceNet uses triplet loss to map face images to a high-dimensional space, excelling in benchmark tests. Evaluation datasets are crucial for comparing algorithm performance, including LFW, AgeDB, CFP-FP.

In these days the best algorithms for detection and recognition is artificial intelligence algorithms based on convolutional neural network (CNN).

A convolutional neural network (CNN) is a specialized type of neural network in deep learning designed for processing data through multiple array layers. CNN is ideal for tasks such as image recognition and is commonly employed in facial recognition software. The magic of CNN lies in its convolutional layers, which are essential for identifying image features through filters. An analogy is helpful in grasping the workings of these layers in image recognition.

Imagine if to spot someone approaching from a distance. Initially, eyes will scan the outline of the figure to distinguish it from surrounding objects. As the person gets closer, and start focusing on distinguishing features like gender, weight, and clothing. Eventually, they pay attention to finer details like facial features and accessories. Similarly, in a CNN, different layers with filters work to detect specific features, starting with general ones and progressing to more intricate details. The filter values in each layer are determined through training on a specific dataset, allowing for accurate predictions on new images based on the learned features.

Figure 1 shows a typical CNN network. The first few convolutional layers (conv1 to conv4) detect the various features (from abstract to specific) in an image (such as edges, lines, etc.). The final few layers (the fully connected layers and the final softmax/logistic layer) are used to classify the result (such as that the image contains faces belong to person A, B, or C).

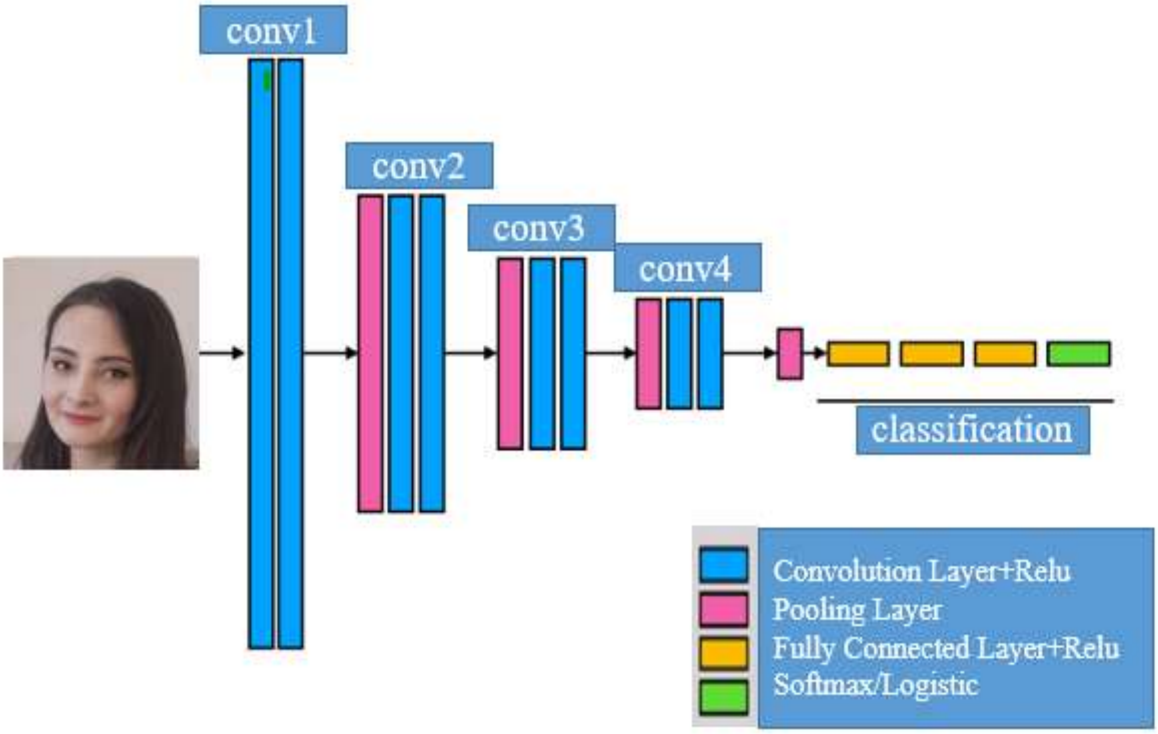

Figure 1: A typical Convolutional Neural Network (CNN) architecture

The structure of Convolutional Neural Networks comprises various layers: Convolution, Activation (ReLU) Layer, Pooling, Flattening layer, and Fully connected layer. We will explore each of these layers thoroughly. Below is a basic diagram showing each of these layers.

The primary function in the Convolutional Layer is Convolution, a crucial mathematical operation with numerous applications, particularly in signal processing and Image data where it is highly effective. The Convolution formula is as follows:

$$(x * w)(t) = \int_{-\infty}^{\infty} x(a) \cdot w(t-a)da$$

Convolution in mathematics combines two functions to create a third function showing how one function affects the other. In image processing, we convolve image data to produce a new output reflecting the two functions. The convolution operation is denoted by an asterisk (*). It is expressed in CNN terminology with *x* as input and *w* as weights, filters or kernels used to convolve the image data. Parameter *t* represents the spatial location in the input image.

Nevertheless, this is not the precise way in which we illustrate the Convolution operation in CNN. It is important to recognize that certain complex terminology is not directly utilized in CNNs – it is simply presented here to provide an understanding of the operation. Additionally, this equation is designed for continuous inputs. While continuous inputs are typically used in signal processing, image processing involves a discrete array of pixels, so the$\int_{-\infty}^{\infty}$. imply changes to the $\sum_{a=-\infty}^{\infty}$. meaning that the combined integration of two function changes to the summation of the corresponding elements.

$$(x * w)(t) = \sum_{-\infty}^{\infty} x(a) \cdot w(t-a)da$$

Nonetheless, this notation may seem complex, simplify it to the true equation for Convolution in Convolutional Neural Networks (CNN).

$$(I * K)(i,j) = \sum_{i=1}^{m} \sum_{j=1}^{n} \left(I_{(i+m-1)(j+n-1)} K_{ij}\right) + b$$

The letter I represents our input, also known as a 2D image, while K stands for Kernel, which is also 2D. The indices i and j are used to denote a single pixel in the input image, kernel, and resulting output. m, n are used to represent the dimensions (height, width) of the filter or kernel. The $I_{(i+m-1)(j+n-1)}$denotes the pixel value of the input image at position$(i+m-1), (j+n-1)$, where the filter is applied to each input pixel as i and j change. The outer summation $\sum_{i=1}^{m}$, slides the kernel over the input image, iterating over i from 0 to m-1 vertically. The inner summation $\sum_{j=1}^{n}$, represents element-wise multiplication, iterating over j from 0 to n-1 horizontally. A bias term bi is then added to enhance network flexibility.

REFERENCES

1. Joseph Redmon, Ali Farhadi(2017). YOLO9000: Better, Faster, Stronger. 2017 CVPR

2. Konark Modi, Lakshmipathi Devaraj Advancements in Biometric Technology with Artificial Intelligence December 2022 6p.

3. Deep Learning (Adaptive Computation and Machine Learning series) Hardcover – 18 November 2016

4. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems, Third Edition (Full Colour Print) Paperback – 10 October 2022

5. Rahul D Chaudhari, Ashok A Pawar, Rakesh S Deore / International Journal of Engineering Research & Technology (IJERT)/The Historical Development Of Biometric Authentication Techniques: A Recent Overview 3921-3928 p.