REPUBLIC OF TURKEY

ALTINBAŞ UNIVERSITY

Institute of Graduate Studies

Electrical And Computer Engineering

AMERICAN SIGN LANGUAGE RECOGNITION USING YOLOV4 METHOD

Ali ALSHAHEEN

Master's Thesis

Supervisor

Asst. Prof. Dr. Mesut CEVIK

Istanbul, 2022

AMERICAN SIGN LANGUAGE RECOGNITION USING YOLOV4 METHOD

Ali ALSHAHEEN

Electrical and Computer Engineering

Master's Thesis

ALTINBAŞ UNIVERSITY

2022

The thesis titled AMERICAN SIGN LANGUAGE RECOGNITION USING YOLOV4 METHOD prepared by ALI ALSHAHEEN and submitted on 18/5/2022 has been **accepted unanimously** for the degree of Master of Science in Electrical and Computer Engineering.

Asst. Prof. Dr. Mesut Cevik

the Supervisor

Thesis Defence Committee Members:

Asst. Prof. Dr. Mesut Cevik	Faculty of Engineering and Architecture Altınbaş University	_____
Asst. Prof. Dr. Abdullahi Abdu Ibrahim	Faculty of Engineering and Architecture, Altınbaş University	_____
Asst. Prof. Dr. Baran Tander	Faculty of Engineering, Aydin University	_____

I hereby declare that this thesis meets all format and submission requirements of a Master's thesis

Submission date of the thesis to Institute of Graduate Studies: ___/___/___

I hereby declare that all information/data presented in this graduation project has been obtained in full accordance with academic rules and ethical conduct. I also declare all unoriginal materials and conclusions have been cited in the text and all references mentioned in the Reference List have been cited in the text, and vice versa as required by the abovementioned rules and conduct.

Ali Mahmood ALSHAHEEN

Signature

ABSTRACT

AMERICAN SIGN LANGUAGE RECOGNITION USING YOLOV4

Alshaheen, Ali

M.Sc., Electrical and Computer Engineering, Altınbaş University,

Supervisor: Asst. Prof. Dr. Mesut Cevik

Date: 18/5/2022

Pages: 57

Sign language is a means of communication mainly between deaf and mute people to communicate their thoughts and feelings among themselves or between normal people. It can also be indicated that sign language has specific vocabulary and associated grammar and dictionaries. There are different types of sign language that differ geographically or according to the context of the language such as American Sign Language, British Sign Language, Japanese etc., in this research, we will focus on American Sign Language. Sign language contains certain words with simple gestures of their own that can be easily interpreted through these gestures such as mom, dad, hello, I love you, etc. However, there are words that do not have specific gestures that can be invoked easily, so a technique called fingerspelling is used to spell the word to be pronounced through letters because each letter of any language in the year has its own gesture or sign that differs from the gestures of other letters. Usually, this technique is used to spell the name. Previously, there was very little research on sign language before the introduction of deep learning algorithms or machine learning algorithms. The most used way to interpret and translate sign language through a computer is to build deep learning algorithms to discover objects that are able to process images and extract important features from images and then use convolutional neural networks to learn these features and train models on them. The great advances in deep learning and machine learning have led to the construction of algorithms for detecting objects and objects in images and videos,

which are associated with their work with neural networks, where they can challenge, classify and sort objects in images and videos into several categories such as people, cars, animals, traffic lights and Sign language gestures etc. You Look Only Once (YOLO) is a model that can be trained on a specific dataset in which objects are detected through images, videos, or in real-time. In this research, we will build a system capable of detecting ASL translation from gestures and signs to words, letters and numbers based on a data set created by the author, which consists of 8000 images divided into 40 classes, each class represents either a letter or a number or a word, and 200 auras for each class were photographed with excellent accuracy under different lighting conditions and from different dimensions. which allows the model to be able to differentiate the signs regardless of the intensity of the lighting or the clarity of the image. And after training the model on the dataset many times, in the experiment using image data we got very good results in terms of MAP = 98.01% as accuracy and current average loss=1.3 and recall=0.96 and F1=0.96 as a final result, and for video results, it has the same accuracy and 28.9 frames per second (fps).

Keywords: American Sign Language, YOLO, CNN, Computer Vision, Machine Learning,

TABLE OF CONTENTS

Pages

LIST OF TABLES

Pages

LIST OF FIGURES

Pages

ABBREVIATIONS

AI	:	Artificial Intelligence
ASL	:	American Sign Language
CNN	:	Convolutional Neural Networks
KNN	:	k-Nearest Neighbours
RCNN	:	Region and Convolutional Neural Networks
ML	:	Machine Learning
VJD	:	Viola Jones Detector
HOG	:	Histogram of Oriented Gradients
DPM	:	Deformable Part-based Model
SPPNet	:	Spatial Pyramid Pooling Networks
YOLO	:	You Only Look Once
SSD	:	Single Shot Multi-Box Detector
FPN	:	Feature Pyramid Networks

1. INTRODUCTION

Sign language is a non-verbal language of communication used by people with a lack of speech and hearing to communicate with the world. Sign language has existed since the presence of the deaf in the world, sign language began in Spain in the17th century (Madrid in 1620 AD), (Joan Pablo Bonnett) published an article in Spanish entitled (Abstract letters and art to teach the mute to speak), and this was considered the first means of dealing with phonology, and treatment of speech difficulties. It has also become a means of verbal education for deaf people with the movements of the hands, which represent the shapes of the letters of the alphabet; To facilitate communication with others. And through the ABCs of Punnett; Deaf children in the school (Charles Michel Delbey) borrowed those letters, and adapted them to what is now known as the Guide to the French Alphabet for the Deaf. Unified Sign Language has been used in the education of the deaf in Italy and Spain since the 17th century, and in France since the 18th century, Old French Sign Language was used in deaf communities in Paris long before the advent of (Abbe Charles Deleby), who began teaching the deaf, and with That is, he learned the language from the deaf people who are there, then he introduced and adopted the French sign language that he learned and modified in his school, so that it appeared similar to the natural sign language, which is used in the origin area of deaf community, and always with oral language additions to show sides of the oral language. In 1755 A.D., Abe established a deaf public school people in Paris. His lectures were based on his supervision of deaf humans gesturing in Paris, under French grammars, until became a French sign language. Laurent Clerc, a graduate and former teacher in Paris, went with Thomas Hopkins Guidant to the United States to establish an American school for the deaf in Hardford., Connecticut, the Eighteenth School for the Deaf was founded. In 1817, Clerk and Galaudent also established an American educational centre for deaf and mute people, which is called the American School of the Deaf now. Then, in 1864, a deaf college was founded in Washington, D.C. And it was actually approved and empowered by President (Abraham Lincoln) who called it: (National College of the Deaf and Dumb), and its name was changed in 1894 to (Galudent College), and then (Galudent University) in the year 1986 AD. [75] [76] [77] Since many people don’t understand sign language because it is simply not easy for normal people and is not required, they will not be able to communicate with deaf and dumb people. With the great development in computer technologies and artificial intelligence, computer vision technology has received great interest and development, as it is used in building detection systems, classifying

objects, and even in security systems. In previous research, the k-Nearest Neighbours (KNN) classifier was used to detect and classify sign language gestures. [78] In another research, convolutional neural networks were used once to detect and identify sign language signals, where Google Net was used in the structure of convolutional neural networks, which was trained on the ILSVRS data set in 2012.This training obtained a good accuracy rate of 72%, this lack of accuracy is caused by the great similarity in the signs of some letters such as g, h, m, n, s, t. [79] R-CNN It is a method that combines the proposals of region and convolutional neural networks where you localize the sores in a convolutional neural network and then train a high capacity model with little annotated detection data. This algorithm achieves good accuracy in object detection by using the deep network "ConvNet" to classify object proposals. Also, it's able to detect thousands of objects without the need for approximate techniques. One of its drawbacks is the weakness in detecting small things, and also that it predicts only a sign and not a hash square. [80] [81] [82] [83] Fast R-CNN It's an algorithm written in Python programming language and is a training algorithm for object detection, developed to solve R-CNN problems regarding to accuracy and speed. One of its advantages is that it has a high accuracy than R-CNN in objects detecting and it also trains in one stage and the training can update all layers of the network. But one of its drawbacks is that it's slow when detecting objects, and this is due to the slow establishment of the selective search area. [84] [85] [86] [87] Faster R-CNN It's an object detection method similar to R-CNN that uses a Region Proposal Network (RPN), which is effectively shares the convolutional features of the image with the detection network. Faster R-CNN is much faster than Fast R-CNN and R-CNN because it's used RPN (Region Proposal Network) for generating anchor boxes (region proposals). And It can be used in real-time object detection. There is one drawback in this method, which is after the RPN is trained, all the anchors are extracted in the mini-batch, because of the great similarity in the features of all the objects in the same image, which causes slow detection and the network may take a long time to reach a point of similarity between the objects. [89] [90] [91] YOLO is a newly recognition methods in deep learning, which has been developed on the basis of CNN, it's more effective and accurate in objects detection [92] [93] [94]. Deep learning models have achieved great success and tremendous developments in the field of object detection, including YOLO, which is a highly efficient model capable of differentiating and detecting objects through images, videos, or in real-time. It's an algorithm that depends in its work on the principle of regression, which means that instead of detecting and specifying the clear and interesting part, it analyses and predicts the categories and boxes that surround the entire image, and that is done

through one run of the algorithm. YOLO used in researches that require the detection of certain objects, or in determining the time, detection of traffic lights, pedestrians, cars plates and parking spaces, people, animals, etc. [95] [96] [97].

2. LITERATURE REVIEW

2.1 AMERICAN SIGN LANGUAGE

American Sign Language used primarily by deaf and dumb people in America. This language originated in the early twentieth century and was influenced by many sign languages such as French and others. This language has evolved until it has a specific vocabulary and grammar and a unique structure. American Sign Language provides gestures and signs for every alphabet of the English language, however, it is not limited to letters or numbers only, but contains many gestures for complete words and sentences. [1]

(Figure 2.1) shows gestures for the abbreviation of the ASL phrase consisting of the letters A, S and L.

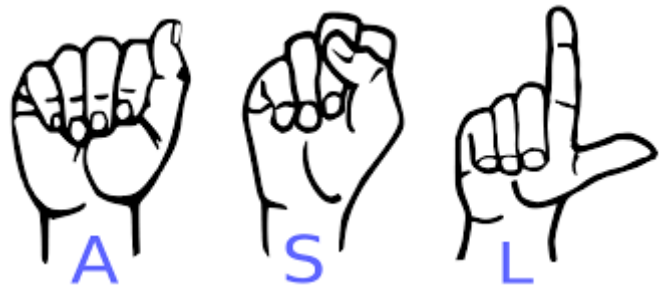

Figure 2.1: Representation of A, S and L in American sign language

The ASL signal consists of five clear aspects, which are the shape of the hand, the position of the hand, facial expressions, orientation and hand movement. Each of these factors has a direct impact on the meaning of the signal. Based on these factors, the signs are classified as fixed signs and other movable signs. The fixed sign does not include any movement of the hand when represented as in Figure 1. As for gestures and animated signs, movement of the hand during the representation of the word is defined as in the word Open in (Figure 2.2) [1]

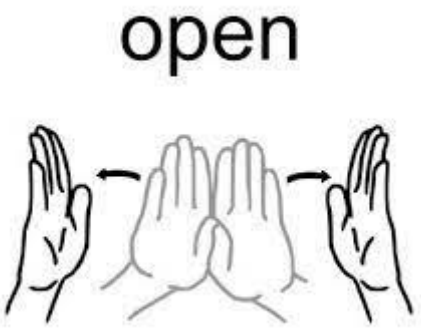

Figure 2.2: Gesture with moving hand

2.2 OBJECT DETECTION

It's a computer technique that relies mainly on computer vision and image processing to identify and classify objects according to specific categories such as people, cars, animals, hand movements, etc. through images, videos, or even in real time This technique is one of the most popular and diverse fields of computer vision research. It has become the cornerstone of many applications in our daily life, such as object tracking, self-driving, face detection, and video surveillance within security systems. Pictures and videos can contain multiple objects in one place or multiple places. The work of this technique is not limited to discovering objects only, but it can provide information related to the detected object, its location and the coordinates of the object. The information may also include a bounding box that specifies the location of the detected object and also the probability of the object being discovered. Object detection and image classification are sometimes confused, so in image classification, a specific image is categorized into pre-scheduled categories. where the image represents the process input, and the output represents the name of the element in the image. For example, if the input is a picture containing a cat, the output will be the word "cat" and if the input is a picture containing two cats, the output will also be "cat". From this we can conclude that the process of classifying images, despite its importance, is limited by the results it gives us. In contrast, in the object detection process, each of the detected elements is identified within a frame, and the location of that element in the given scene is additionally determined. From this we conclude that object identification is a more complex process than

classifying images and gives us more information about the image, so we can benefit from it more. [2][3][4][5]

In recent years, the old traditional computer vision has been gradually replaced by deep learning models, and algorithms have become the basic structure in the work of object detection technology. Since the object detection technology deals with images directly, we must know how the image is understood by the computer. How to analyse the information that a computer can understand from the image is the main problem in computer vision? Deep learning models have become a focus of research due to the great capabilities they offer in detecting objects. There are three levels of understanding the image:

2.2.1 Classification

It is a task of deep learning to identify and classify objects in the image or video, where it refers to the categories of objects and classifies them as present or not in the image, and then puts each object in the image in its own group. [6]

2.2.2 Object Detection

Detection is useful in identifying objects in the image or video, as the detected objects are surrounded by a bounding box with the name of the detected object. Among the applications of object detection is face detection and analysis, as well as estimating the age or knowing the gender of a person through some characteristics in the face. There are also many applications in the field of real-time object detection, such as traffic management, vehicle detection systems, etc. The most common methods in computer vision are classification and discovery to identify objects and determine their location and what those objects are. [7]

2.2.3 Segmentation

Each image consists of useful and non-useful information based on the user's interest. The image is fragmented into regions, each of which has its own shape and boundaries, giving these regions the possibility of being useful or not. In classification and discovery, these areas may not occupy the entire image, nor be detected or categorized, but the goal of segmentation is to highlight and evaluate the elements in the image. Segmentation provides details and information for each pixel of the object in the image, which makes it different from other objects. [8]

The (figure 2.3) below shows the difference between alignment, Object detection, and segmentation

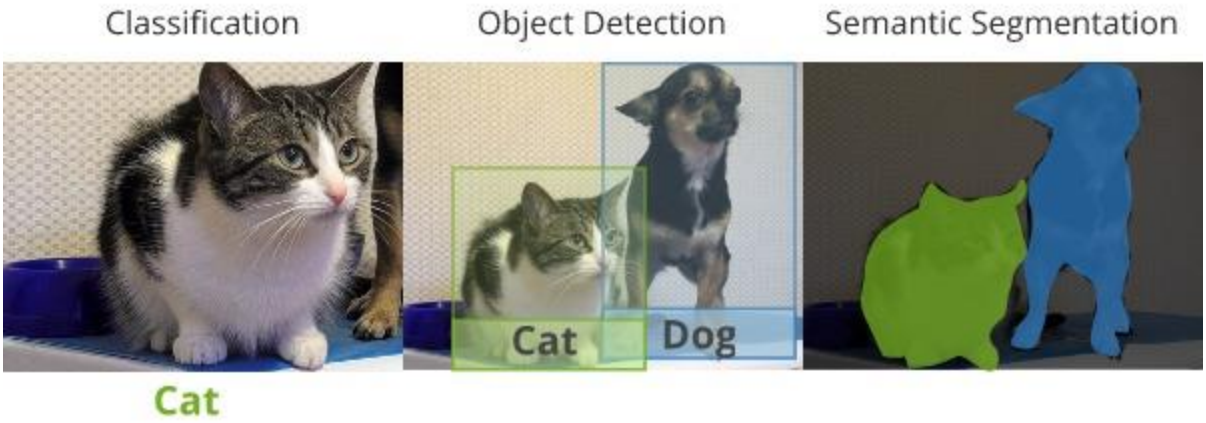

Figure 2.3: The difference between alignment, object detection, and segmentation [8]

2.3 CONVOLUTIONAL NEURAL NETWORKS (CNN)

Artificial intelligence is rapidly expanding to bridge the gap between human and computer capacities. Researchers and hobbyists alike are working on a variety of fronts to accomplish amazing results. Computer vision is just one of several related topics. [9]

This field's goal is to enable machines to see and understand the world in the same way that humans do, and to use that knowledge for a variety of tasks such as image and video recognition, image analysis and classification, media recreation, recommendation systems, natural language processing, and so on. Convolutional Neural Network (ConvNet/CNN) is a deep learning method that can take an input image and assign importance (learnable weights and biases) to distinct sides in the image, allowing it to differentiate one from the other. In comparison to other rating algorithms, ConvNet requires far less pre-processing. Filters are created manually in primal approaches, but ConvNets can learn the attributes of these filters with enough training. [9] [10] ConvNet's architecture is inspired by the organization of the Visual Cortex and is similar to the connection network of neurons in the human brain. Individual neurons can only respond to stimuli

in a small portion of the visual field called the receptive field. A group of these fields can be stacked on top of each other to cover the full visible region. [9] [10]

CNN was developed and utilized for the first time in the 1980s. At the time, the best CNN could do was recognize scribbled digits. It was primarily used in the postal sector to read postal codes, personal codes, and other information. The most important thing to understand about any deep learning model is that it requires a lot of data to train as well as a lot of computational power. This was a major disadvantage of CNNs at the time, therefore they were limited to the postal industry and never made it into machine learning. [9] [10]

A convolutional neural network (CNN/ConvNet) is a type of deep neural network used to evaluate visual pictures in deep learning. When we think about neural networks, we usually think of matrix multiples, but this isn't the case with ConvNet. It employs a method known as convolution. In mathematics, a convolution is now an arithmetic operation on two functions that yields a third function that explains how one's shape is changed by the other.

The building design of a convolutional neural network includes many building layers like convolutional layers, pooling layers, and fully connected layers. [9] [10]

Many convolutional layers are repeated in a common architectural design, followed by a pooling layer with one or more layers of the fully connected layer.

Forward propagation is the process of converting input data into output data through various layers.

Many building layers, such as convolutional layers, pooling layers, and fully connected layers, make up the architecture of a convolutional neural network. [9] [10]

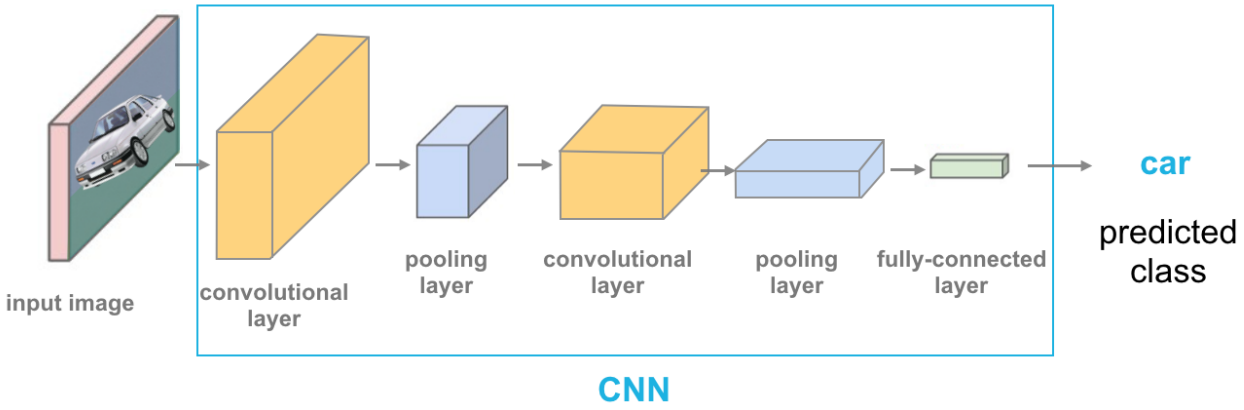

Figure 2.4: CNN architecture [10]

2.3.1 Convolution Layer

The convolution layer is a fundamental part of the construction of convolutional neural networks, and its job is to extract attributes from images (weights), which are usually obtained by a series of linear and non-linear operations.

Convolution is a specialized sort of operation for extracting features from weights in which a limited collection of numbers called a kernel is applied over the input, which is grouped from numbers called tensors.

The output value at the position corresponding to the output tensor, which is called the feature map, is determined in terms of the elements between each element of the kernel and the input tensor at each position of the tensor. [9] [10] [11]

2.3.2 Pooling Layer

It's performs a conventional down sampling procedure with the purpose of reducing the internal dimensions of feature maps to ensure translation stability for tiny feature transformations and deformations, as well as reducing the amount of features that will be learnable later. [9] [10] [11]

2.3.3 Fully Connected Layer

The final convolution or aggregation layer's output feature maps are normalized by transforming them to an array of one-dimensional numbers and connecting them to one or more fully connected layers, also known as dense layers, with learnable weights connecting the inputs to the outputs. [9] [10] [11]

Following the extraction of features from the convolution and reduction layers, these features are translated to the network's final output by a subset of fully connected layers.

The number of output nodes in the final fully connected layer is usually equal to the number of classes. [9] [10] [11]

2.4 DEEP LEARNING

Deep learning is a subtype of machine learning that consists of three or more layers of a neural network. This neural network tries to emulate the human brain's functioning and activity by allowing it to learn and train from massive amounts of data. An approximation prediction can be made with a single layer neural network. Deep learning offers a wide range of applications in artificial intelligence that help to increase automation and execute analytical tasks without the need for human intervention. In addition to highly serious techniques like self-driving cars, deep learning approach provides daily applications and services such as (digital assistants in phones, voice control devices, and detecting credit card fraud and other applications). [12] [13]

2.4.1 How deep Learning Works

Deep learning neural networks, also known as artificial neural networks, use a combination of data inputs, weights, and biases to replicate the human brain. These parts collaborate to appropriately identify and describe the data's objects. Deep neural networks are made up of numerous layers of interconnected nodes, each of which relies on the previous layer to increase prediction, detection, or classification accuracy. Forward propagation refers to the advancement of computational activities through a network.

The visual layers of a deep neural network are the input and output layers. The deep learning model takes in data for processing in the input layer, and the final detection, prediction, or classification is done in the output layer. Backpropagation is a method of training a model that involves using algorithms to calculate the detection or prediction error rate and then adjusting weights and biases by traveling backwards through network layers. The neural network can do error correction using forward and reverse propagation until the method becomes increasingly accurate. [14] [15]

2.4.2 Deep Learning Applications

Applications of deep learning in the real world are part of our daily life, most of the time it is integrated into products and services that work on deep learning and neural networks, sometimes users are not aware of the complex processing processes that happen in these products they use. [16] [17]

Some examples of deep learning applications:

2.4.2.1 Law enforcement

Deep learning algorithms can examine, learn, and rehearse transaction data in order to detect and identify possible fraudulent or criminal behaviour.

By revealing guides in audio recordings, bull, videos, and documents, speech recognition technology, computer vision technology, and artificial intelligence applications that rely on deep learning can improve the capacity and activity of investigative analysis, allowing law enforcement to analyse vast amounts of data in an efficient manner. Faster and more precise [18] [19]

2.4.2.2 Financial services

Predictive analytics is used by some financial and banking institutions to drive algorithmic stock trading, assess business risk for loan approvals, detect fraud, manage credit portfolios, and other duties. [20] [21]

2.4.2.3 Customer service

Many businesses are incorporating deep learning into their customer service operations.

Auto responders are a type of advanced artificial intelligence and deep learning that may be found in a variety of applications and mobile phones. These auto-responders, such as Apple's Siri, Amazon Alexa, or Google Assistant, speak to customers using natural language and even visual recognition, determining if there is a Multiple answer to a group of questions by learning from the responses he receives, then analysing the question and giving the appropriate answer and directing it to the user. [22] [23]

2.4.2.4 Healthcare

Object detection technology can enable medical imaging specialists and radiologists study and evaluate multiple images or rays accurately and in less time, which has profited from the use of artificial intelligence and deep learning in the health and medical fields. [24] [25]

2.5 DEEP LEARNING HAEDWARE REQUIREMENTS

Deep learning necessitates vast computing operations, for which high GPUs are perfect since they can handle large quantities of arithmetic in multiple cores while also saving a lot of memory. [26] [27]

2.6 MACHINE LEARNING

Machine learning is a section of artificial intelligence and computer science that focuses on simulating the way the human brain learns and steadily improving its accuracy using data and algorithms. Data science necessitates the use of machine learning. In data mining projects, models are trained by algorithms using statistical approaches to produce classifications or predictions and detect objects. Machine learning, on the other hand, can fit new data independently, and through

iterations, applications may learn from previous computations and operations to improve output accuracy and minimize processing time. Machine learning's rapid advancement has resulted in use cases, expectations, and the importance of machine learning in our daily lives.

This is because of the growing use of machine learning, which allows for the analysis of massive amounts of data. Machine learning has also altered the way data is retrieved and evaluated, automating generic algorithms and therefore displacing older technologies with newer ones that keep up with this rapid advancement. [28] [29] [30]

2.6.1 How Does Machine Learning Work

One of the most fascinating areas of artificial intelligence is machine learning. With the device-specific input, completes the task of learning from data. The training data is entered into the appropriate algorithm to begin the machine learning process. To build the work of the final machine learning algorithm, the training data is known or unknown. The algorithm is affected by the type of data entered. The iron data is fed into the algorithm to see if it's performing properly. Both the prediction and the results are cross-checked. If the forecast does not match the results, the algorithm is retrained until the results are satisfactory. As a result, the machine learning algorithm may learn on its own, delivering exceptional results and responses while gradually improving accuracy. [31] [32] [33]

2.6.2 Machine Learning Types

There are two types of learning: supervised and unsupervised. Each has a distinct activity and goal that leads to results and uses various types of data.

Surveillance learning accounts for over 70% of machine learning, whereas unsupervised learning accounts for 10% to 20% of all learning in any discipline, and reinforcement learning accounts for the rest. [34] [35]

2.6.2.1 Supervised learning

In supervised learning, we use known or categorized data for training. Because the data is classified and known, training and learning are supervised, which means they are geared toward effective implementation. The data is entered into a machine learning algorithm, and the model is then

trained. After the model has been trained on known data, unknown data can be included into the model, and a new response can be returned. [36] [37]

2.6.2.2 Unsupervised learning

The training data in unsupervised learning is unknown and unidentified, which implies no one has ever seen it before. The input cannot be passed to the algorithm without the known data side. [38] [39]

The data is supplied into the machine learning algorithm, which is then used to train and learn the model. The trained model looks for a pattern and responds accordingly. [38] [39]

2.6.2.3 Reinforcement learning

The algorithm in this sort of learning learns data through trial and error and then chooses the procedure that delivers the best results. Because the agent is the learner, the environment is everything the agent interacts with, and procedures are the agent's duties, reinforcement learning has three components: the agent, the environment, and the procedures. When an agent chooses activities that raise predicted outcomes over time, this is referred to as enhanced learning. [40] [41]

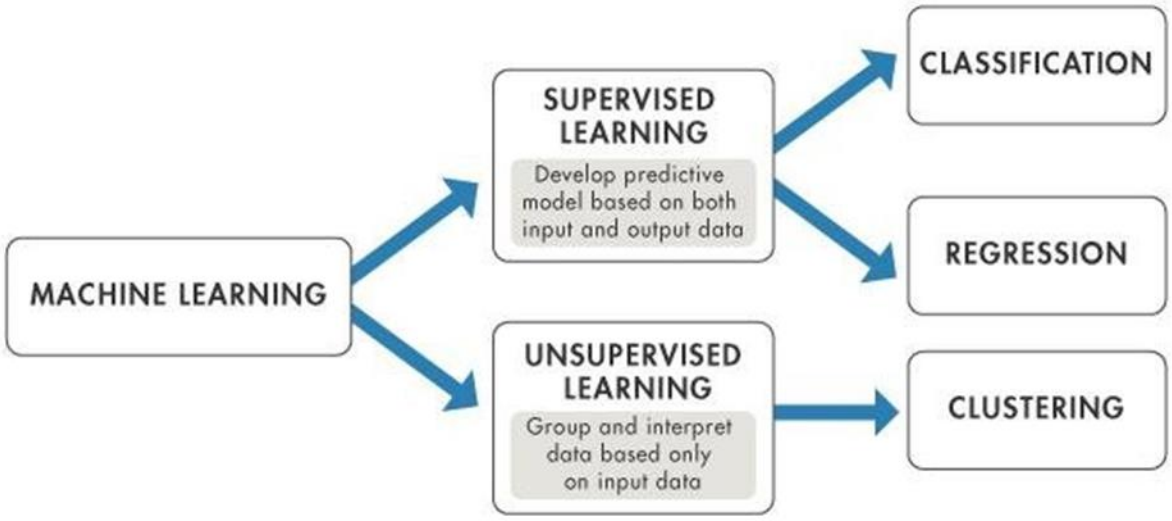

Figure 2.5: Machine learning types [34]

2.7 DEEP LEARNING VS. MACHINE LEARNING

Deep learning differs from machine learning in the kind of data it works with and the methods it uses to train and work. To make predictions, machine learning algorithms use bound data, which means that features from the model's input data are identified and tabled. This doesn't mean that it doesn't use unbound data; it just means that if it does, it goes through a series of pre-processors to combine and arrange the entered data. Deep learning eliminates some of the data pre-processing steps that machine learning conducts, as deep perception algorithms can ingest unstructured and structured material, including as text and images, and process it without difficulties, extracting features from them, all without the need for human interaction. If we have a collection of photographs of several automobile models and wish to categorize them based on their appearance, deep learning algorithms can find the most critical qualities that separate one car from another. This feature hierarchy is developed by humans in machine learning. Machine learning and deep learning can learn in a variety of methods, which are typically divided into three categories: supervised learning, unsupervised learning, and reinforcement learning. Supervised learning categorizes or predicts using pre-labelled data sets; this sort of learning involves human participation to appropriately label the input data. Unsupervised learning, on the other hand, does not require a set of classified data because it finds patterns in the data and groups them according to distinct features. Reinforcement learning is a technique for teaching a model to become more

proficient at performing any task in a feedback-dependent environment in order to achieve better results. [42] [43]

2.8 OBJECT DETECTION METHODS

The technique of object discovery generally went into two historical periods, the traditional detection period before 2014 and the time of detection based on deep learning after 2014 [12] as shown in the (figure 2.6) below

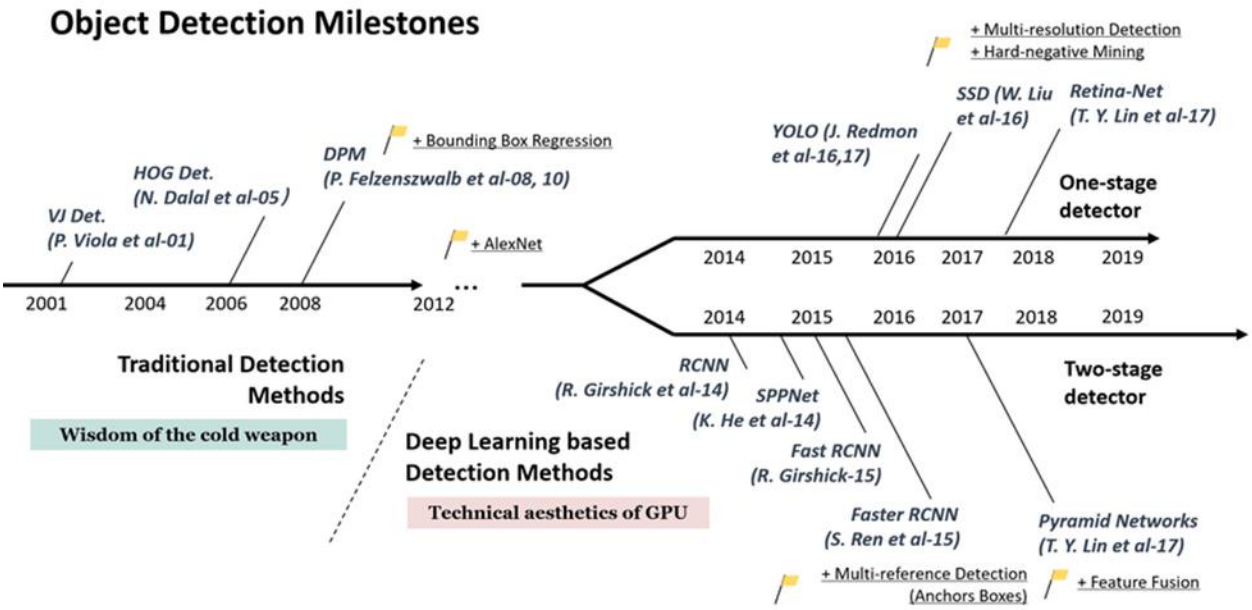

Figure 2.6: Object detection lifetime [44]

2.8.1 Traditional Detection Methods

Most object detection algorithms were built on features made by hand because of the weak image representation at the time, which was 20 years ago. [44]

2.8.1.1 Viola jones detector

Eighteen years ago, scientists P. Viola and M. Jones made a remarkable progress when they invented an algorithm to detect human faces in real time for the first time without restrictions. This algorithm was first run on a Pentium 3 700 MHz processor.

And this algorithm was hundreds of times faster than any other algorithm.

This algorithm was mentioned in the names of the two scientists in memory of their important contribution to the development of this algorithm.

The VJ detector employs a straightforward detection method known as sliding windows, in which it examines all potential positions in the image to determine whether or not there is a human face, then repeats this procedure for each window. The processors and calculations that were based on

it were beyond the capability of the computer at the time, despite the fact that it appeared to be a very basic and uncomplicated operation. [45] [46]

This significantly increased the detection speed by integrating three important techniques:

Integral image: Haar waves are utilized in the VJ detector as a distinct impersonation of the picture, and it is a computational way to speed up the box filter or convolution process. [47] [48]

Feature selection: Rather than employing a collection of manually selected Haar-based filters, the authors employed the Adaboost algorithm to select a small range of characteristics that are most beneficial for pain identification from vast sets of randomized features. [49]

Detection cascades: This technique works on a sequence of detection in order to reduce computational expenses as it performs fewer computations on back windows but more on facial targets or facial boundaries. [49]

2.8.1.2 HOG detector

It's a detector that describes the feature of an object that has been detected. It's used to detect objects in computer vision and image processing. [50]

It is also a representation of an image or an excised part of an image so that it simplifies the image by extracting only useful information from it and getting rid of useless information. [51] [52]

This detector calculates the frequency of the gradient direction in the detected part of the image. [53]

This detector focuses its work on the structure or shape of the object.

Although HOG can detect a wide range of objects, it was developed to address the problem of pedestrian detection.

The HOG detector rescales the input image numerous times while keeping the detection window size unchanged to detect objects of various sizes.

For many years, this technique has served as the foundation for many object-detection devices as well as a wide range of computer vision applications. [54] [55] [56]

2.8.1.3 Deformable part-based model (DPM)

P. Felzenszwalb suggested DPM as an extension of the HOG detector in 2008, and R. Girshick made numerous enhancements to it.

This discovery is founded on the "divide and conquer" idea, in which training may be thought of as simply learning a good approach to study an item, and the conclusion as a collection of discoveries on various areas of the object.

A root filter and a number of partial filters make up a typical DPM detector. In the detector, an unsupervised learning method has been devised where all fragment filter configurations can be detected automatically as latent variables rather than manually specifying them as previously worked.

Despite the fact that today's object detection devices considerably outperform DPM in terms of detection speed and accuracy, many of them are still impacted by the nature of this detector's work. [55] [58] [59] [60]

2.8.2 CNN Based Two-Stage Detectors

the world witnessed the rebirth of convolutional neural networks in 2012.

The challenge was whether the deep convolutional neural network might be utilized to discover items because it has such strong and high capabilities for representing images. In the year 2014, the scientist R. Girshick took the initiative to break the impasse by offering object detection using regions with characteristics (RCNN).

then, the technology for discovering objects has begun to develop at an unmatched speed.

In the era of deep learning, object detection technology used in 2 types: "two-stage detection" and "single-stage detection". [60] [61] [55] [62]

2.8.2.1 R-CNN

R-CNN It is a method that combines the proposals of the region and convolutional neural networks where you localize the sores in a convolutional neural network and then train a high capacity model with little annotated detection data. In other words, this algorithm begins to extract a set of suggestions for objects through selective search. After that, each width of a fixed-size image is rescaled and input into an ImageNet Feature Extraction dataset-trained CNN model. After that, each width of a fixed-size image is rescaled and input into an ImageNet Feature Extraction dataset-trained CNN model. RCNN achieves a significant increase in performance using the VOC07 dataset, with a significant improvement of the average mAP accuracy 33.7% to 58.5%. Although this algorithm has made significant progress, it has obvious flaws, such as redundant feature calculations on a large number of overlapping proposals (more than 2000 boxes in a single image), which results in a significant reduction in detection speed, as one image takes 14 seconds to process with GPU usage. Later that same year (SPPNet), this problem was resolved. [62] [63] [64] [65]

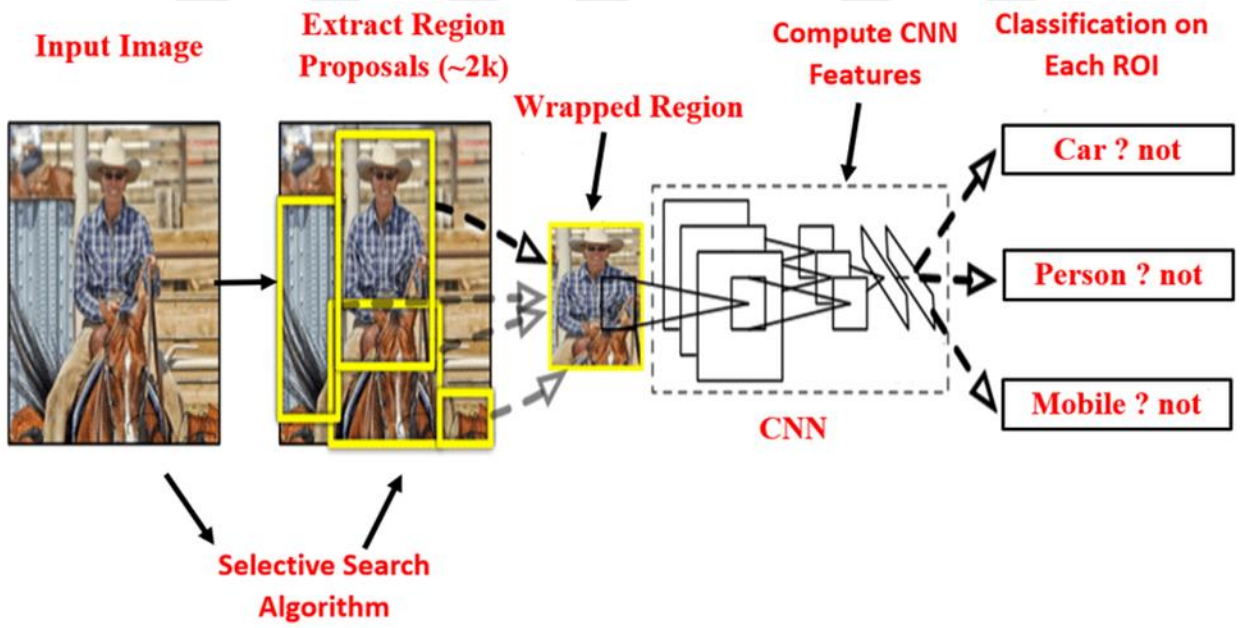

Figure 2.7: RCNN architecture [65]

2.8.2.2 SPPNet

In 2014, the scientist K. He presented the Spatial Pyramid Pooling Networks (SPPNet), which adds to the introduction of the SPP spatial pyramid pooling layer, which allows CNN to construct a fixed-length representation independent of image size and region of interest without scaling it. [65]

Where, when using SPPNet for object detection, feature maps can be computed from the whole image only once and then fixed-length representations of regions can be generated to train detectors on, avoiding repetitive convolutional feature computation.

SPPNet is 20 times faster than RCNN without any loss of detection accuracy. [62]

Although it has significantly increased detection speed, it still has flaws, such as the fact that training is still multi-stage and SPPNet only modifies its fully linked layers while disregarding all preceding layers.

Fast RCNN was presented later that year, and the problem was solved. [65]

2.8.2.3 Fast R-CNN

In 2015, R. Girshick improved the RCNN algorithm with the SPPNet dataset and called it fast RCNN. [66]

Fast R-CNN It's an algorithm written in Python programming language and is a training algorithm for object detection, developed to solve R-CNN problems regarding to accuracy and speed. One of its advantages is that it has a high accuracy than R-CNN in objects detecting and it also trains in one stage and the training can update all layers of the network. Where the accuracy rate of mAP was raised from 58.5% to 70.0%, and even the detection speed increased by 200 times that of the detection speed in RCNN. [65] [61]

However, despite successfully combining the benefits of R-CNN and SPPNet, Fast-detection RCNN's speed is still limited by detecting suggestions and predictions, which is one of the algorithm's flaws. [65]

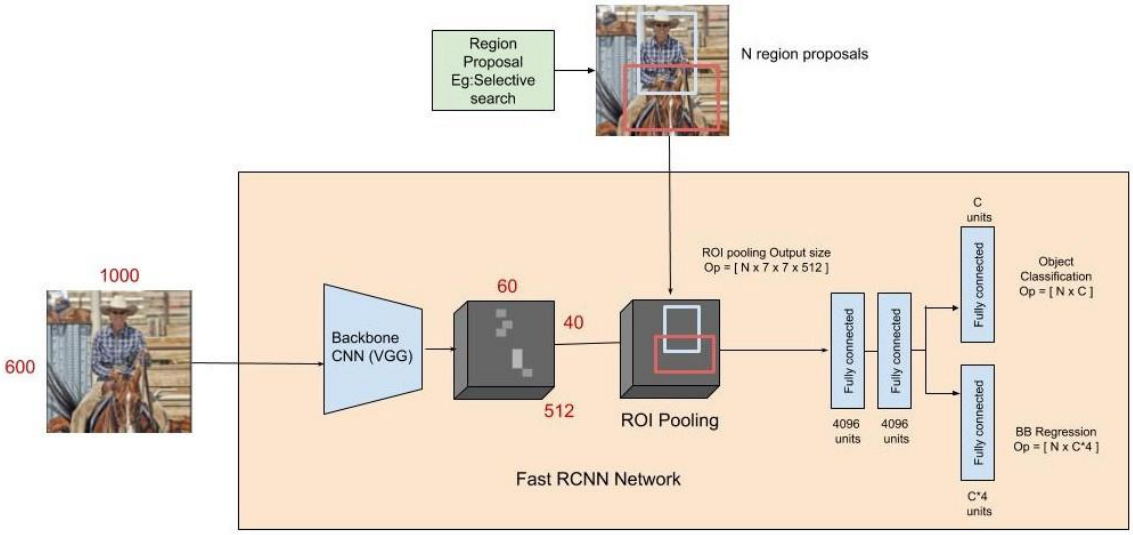

Figure 2.8: Fast R-CNN architecture [65]

2.8.2.4 Faster R-CNN

In 2015, S. Ren suggested a quicker detector dubbed faster RCNN, which includes a novel addition from faster RCNN: the inclusion of a Region Suggestion Network (RPN) that allows essentially cost-free region recommendations.

From R-CNN to RCNN, which is quicker Motion detection, feature extraction, bounding box regression, and other discrete object detection system blocks are gradually merged into a single and comprehensive learning framework.

Due to the great similarity in the features of all the objects in the same image, which causes slow detection and the appearance of the network may take a long time to reach a similarity point between the objects. [67] [68] [69] [70]

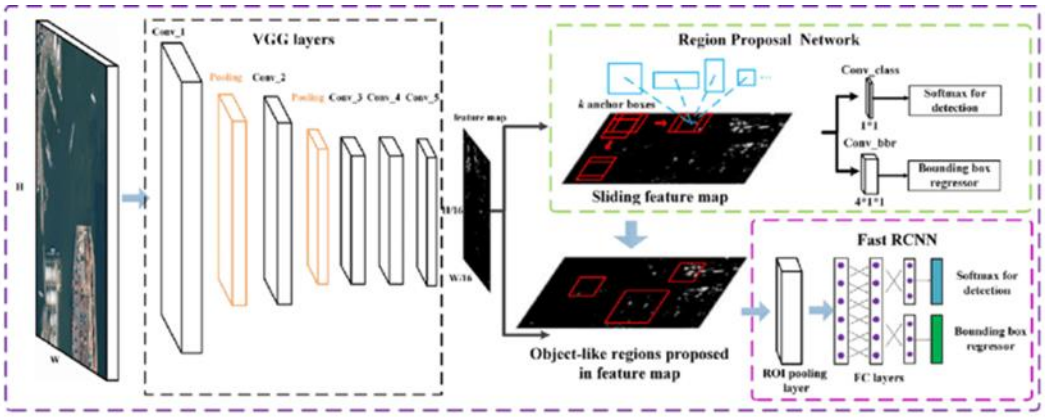

Figure 2.9: Faster R-CNN architecture. [70]

2.8.2.5 Feature pyramid networks

T.-Y. Lin, a scientist, developed Feature Pyramid Networks (FPN) based on Faster RCNN in 2017.

Most deep learning-based object detection methods previously only detected objects in the top layers of the network, despite the fact that characteristics in the deep layers of a convolutional neural network are valuable for object-class recognition but not for object localization.

A top-down design with FPN side links was created to tackle this challenge, allowing high-level HMs to be built at all levels.

Since convolutional neural networks form a distinct hierarchy through forward propagation, FPNs show tremendous advances beyond the faster RCNN platform.

FPN has now become an essential part of many modern detectors. [71]

2.8.3 CNN Based One-Stage Detectors

The single-stage detectors predict the bounding boxes on the images without the step of suggesting the area, the opposite of what the two-stage detectors do.

This technique can be used for real-time detection because it takes little time.

Detectors that use this technique give priority to inference speed, which is very fast, but it is not good at identifying objects that have irregular shapes and even small objects.

What stands out about One-stage detectors is the speed at which it detects in multiple stages and is structurally simple. [72]

2.8.3.1 You only look once (YOLO)

R. Joseph suggested YOLO as the first single-stage object detector in the deep learning era in 2015.

Yolo is characterized by speed as it works at a speed of 155 frames per second with an accuracy of VOC07 mAP = 52.7%, while the improved version of it works at 45 frames per second with an accuracy of VOC07 mAP = 63.4% and VOC12 mAP = 57.9%.

Yolo suggests using a global neural network that makes bounded-squares predictions and class probabilities all at once. [72] [73] [74] [75]

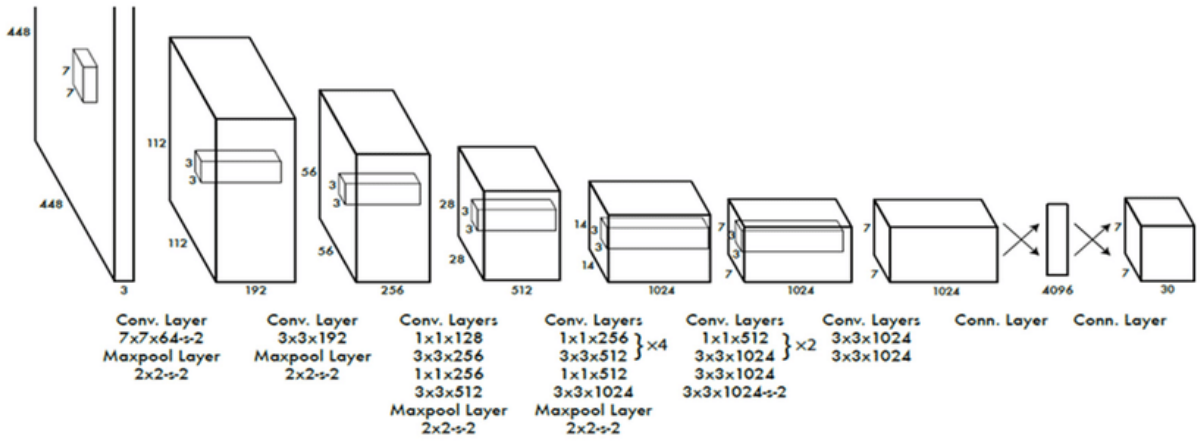

Figure 2.10: YOLO architecture [72]

You only look once (YOLO) timeline

YOLO V2

YOLOv2 is proposed to fix major YOLO problems which are the problem of small object detection in groups and the problem of translation accuracy.

YOLOv2 increases the average network average accuracy by introducing batch normalization, where the batch standard increases the accuracy value by 2%.

The most effective addition to the YOLO algorithm that YOLOv2 added was the addition of anchor boxes.

Yolo predicts one object per cell of the grid and this leads to problems that make the built model simple and also creates problems when a cell contains more than one object, as Yolo can assign one class to the cell.

For YOLOv2 it gets rid of this limitation by allowing multiple bounding boxes to be predicted from a single cell, this is achieved by having the network predict 5 bounding boxes per cell. [72] [73] [74] [75]

YOLO9000

YOLO9000 is proposed as an algorithm to discover more categories of data set (COCO).

The COCO-Trained Object Detection (COCO) dataset contains 80 classes compared to classification networks such as ImageNet, which contains 22,000 classes.

In order to detect many classes, YOLO9000 uses the labels from both ImageNet and COCO, which results in a powerful and effective combination of classification and detection functions to perform the detection only.

YOLO9000 provides a lower average accuracy than YOLOv2, but it is capable of detecting more than 9000 classes, making it a powerful algorithm. [72] [73] [74] [75]

YOLOv3

YOLOv3 was proposed by the scientist to improve YOLO with modern convolutional neural networks by Joseph Redmon in 2018.

DarkNet-53 is the backbone of a more complicated model in YOLOv3 than DarkNet-19 was in YOLOv2.

YOLOv3 allows prediction at three different levels with feature mapping at layers 82, 94 and 106 because DarkNet-53 is a 106-layer neural network complete with rest blocks and upsampling networks.

It compensates for the shortcomings of YOLOv2 by detecting features in three different stages, especially in the detection of small objects.

The exact features that are extracted are preserved with the structure that allows the multiply layer output to be sequenced with features from the previous layer making detection of smaller objects easier and this was the reason behind the proposal of YOLOv3.

YOLOv3 only predicts 3 bounding boxes per cell while YOLOv2 predicts 5, but it makes three predictions of different scales. [72] [73] [74] [75]

YOLOv4

YOLOv4 was proposed in 2020 by Bochkovskiy as an addition and improvement to YOLOv3, where this algorithm achieved the latest results with an average accuracy of 43.5% working at 65 frames per second using Tesla v100 GPU.

These results were achieved after incorporating a number of changes in the architecture and methodology of training and learning in YOLOv3.

Some authors have released an updated version of YOLOv4 called YOLOv4 tiny, which provides faster object detection and higher FPS. [72] [73] [74] [75]

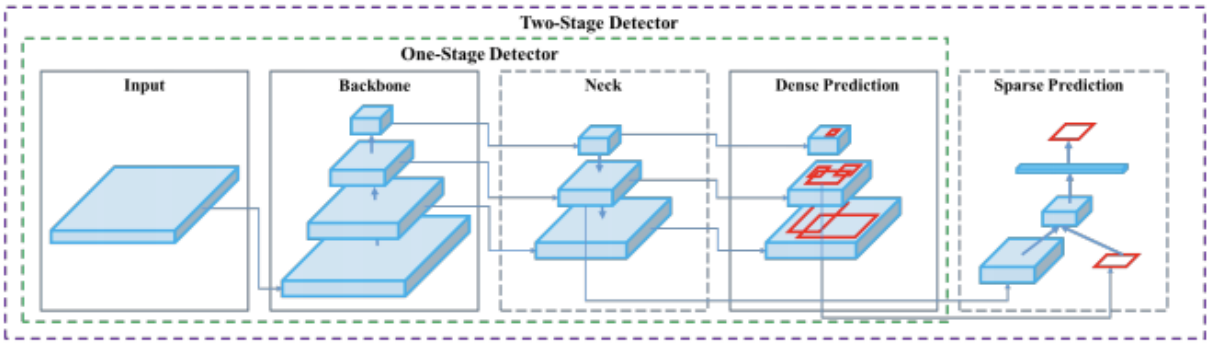

Figure 2.11: YOLOv4 architecture [74]

2.8.3.1 Single shot multi-box detector (SSD)

This algorithm, proposed by scientist W. Liu in 2015, was the deep learning era's second one-stage detector.

The introduction of multi-reference and multi-accuracy detection approaches, which considerably improve detection accuracy, particularly for small objects, was a significant contribution to this algorithm. The SSD recognizes items at five levels and five different layers of the network, whereas the other algorithms only detect objects in the upper layer of its layers. [73]

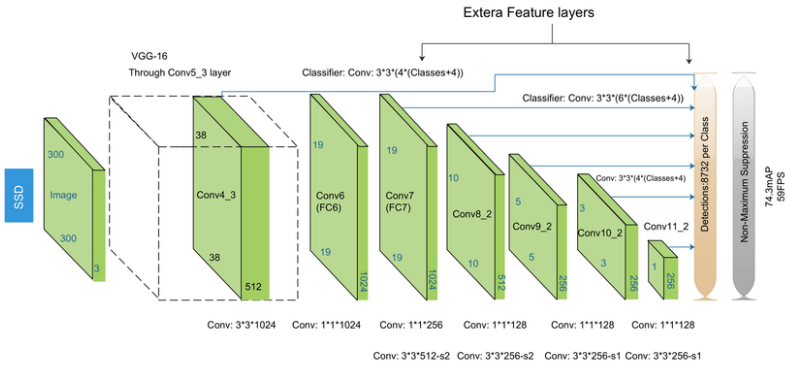

Figure 2.12: Single shot multi-box detector (SSD) architecture [73]

3. DESIGN

3.1 DATASET

3.1.1 Image Collecting

Initially more than one data set was used, downloaded from the Internet. The model has been trained on each data set, but no one of these exercises hasn't given good results as there were many problems in training because the data set was poor in terms of light, accuracy and clarity of images.

Also in the first exercises, the computer specifications were not enough to train correctly. Because of these problems have been obtained weak results accuracy and detection time.

The work environment has also been changed and a new data set was built by the author, where the data set was collected in the form of a set of 8000 images.

This dataset included images of 40 Class, all classes representing a letter or number or a word from the language of American Sign Language (ASL).

All Class contains 200 images as shown in the (Table 3.1) below.

Table 3.1: Dataset specification

Specification	Value
Resolution	1920*1080
Extension	.JPG
Number of images	8000
Number of class	40
Number of images per class	200
Image size	900-1000 Kb

The data set was taken in different conditions and also different dimensions as shown in the (Table 3.2) and Figures below

Table 3.2: Dataset conditions

Conditions	Description
Condition1	Brightness, taken from 20-30 cm away
Condition2	Brightness, taken from 50-60cm away
Condition3	Low Brightness, taken from 20-30 cm away
Condition4	Low Brightness, taken from 50-60cm away

Figure 3.1: Condition1

Figure 3.2: Condition2

Figure 3.3: Condition3

Figure 3.4: Condition4

3.1.2 Data Labelling

Data label is a first step in data determination such as images, videos, texts and so on.

Where one or more labels that give a definition of the object to be detected.

When building a computer vision system, you need to name images or objects in the image by creating a box surrounded by a fully surrounding the object that is called Bounding BOX. As shown in the (Figure 3.5) below

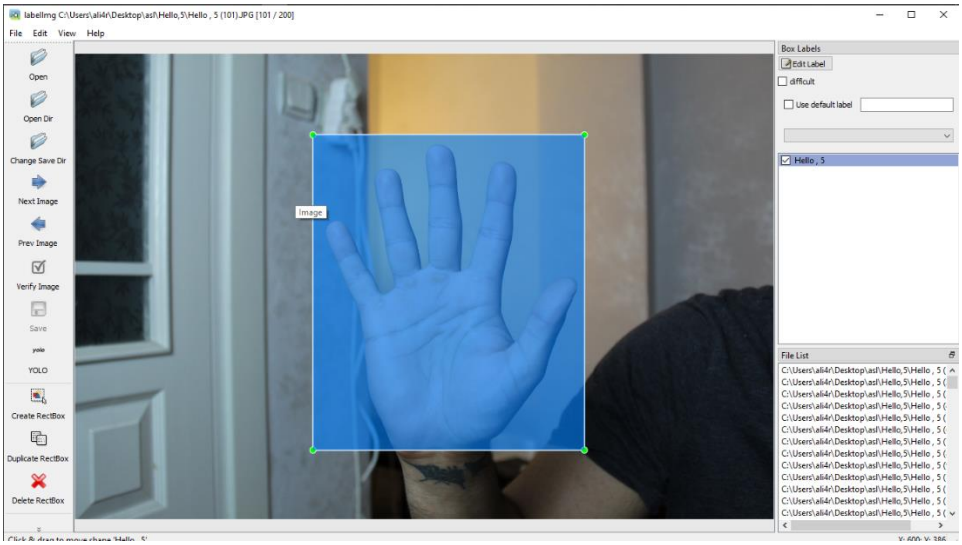

Figure 3.5: Image labelling using labelimg

This data can then be used to train the form to discover or classify objects or discover sites in the image.

In this search, a program named “Labelimg” is an open source program to classify images using a graph, which was released in 2015 and is written in Python.

Comments or labels can be saved with xml files from ImageNet

The name is given for each class, by drawing a box, surrounds the object to be renamed, and this object is then attributed to the five values, which are numbers that specify the location and size of the object in the image. as shown in the form below

It is then saved in the XML format as we mentioned before, this process is made for each image in one-class and also for all classes,

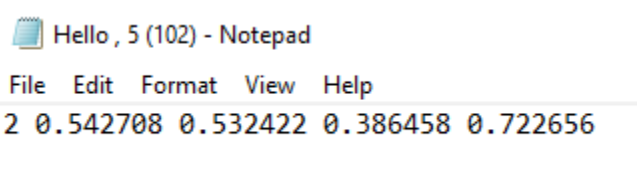

Figure 3.6: Object values

3.1.3 Model Training and Results

3.1.3.1 Platform

The operating systems or environment used to train the model differ according to the users' use. This environment is built by the user in order to be prepared to use the previously prepared data set to train the model on.

In this tutorial, the author used Windows 10 as the operating system, with a Ryzen 7 5800H processor and an NVIDIA RTX3070 GPU as the graphics card. Version 3.7 was used.

From the Python programming language within the Darknet framework.

3.1.3.2 Training

The data set is divided into two parts with a ratio of 80 to 20, where the largest percentage is for training and the other for test.

After the data set was created, the model was trained on it, but no good results were obtained due to the high resolution of the images, where the resolution was 3840 x 2160.

Then the resolution was reduced to 1920 x 1080 and the model was trained on the new data set several times.

And the results began to improve in the first training, the model was trained for 3000 iterations, and the results began to improve as shown in (Figure 3.7) below.

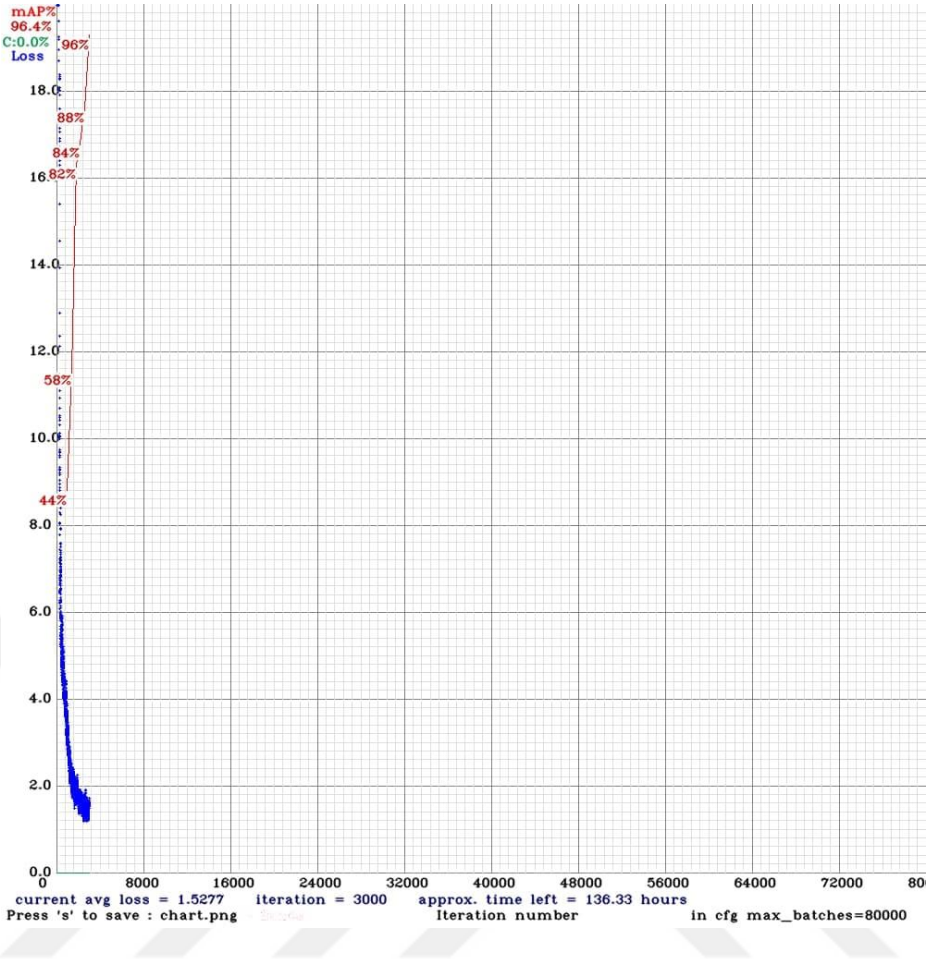

Figure 3.7: First training results

As we can see in the (Figure 3.7) that the accuracy started to rise, so at iteration 1000 the accuracy was 44%, and rise to 96% at iteration 3000 and the rate of loss started decreasing at iteration 1000 also down to iteration 3000 where it was 1.52 and in this iteration stopped decreasing and the first training ended here.

After that, we re-trained for 4000 iterations, and we got an increase in accuracy and a lower average loss, as shown in (Figure 3.8) below.

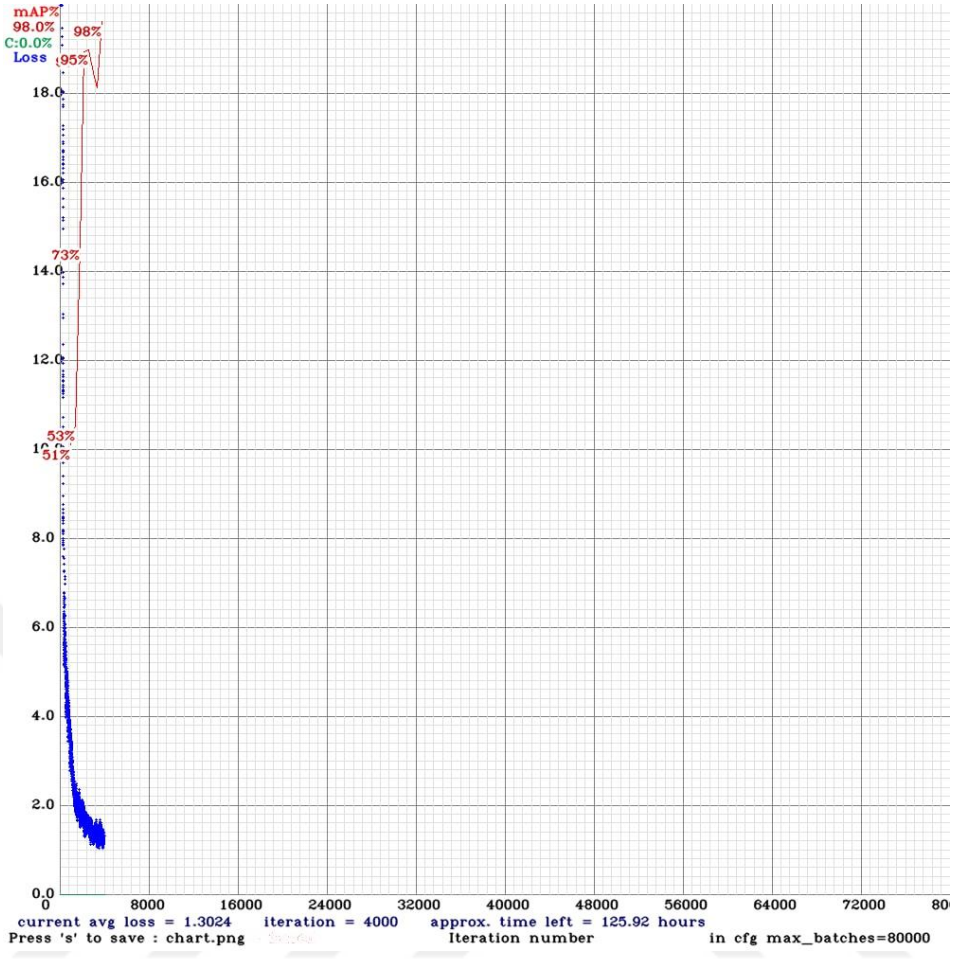

Figure 3.8: Second training results

And as we can see in the (Figure 3.8) that the accuracy started to rise, so at iteration 1000 the accuracy was 51%, and rise up to 98% at iteration 4000 and the rate of loss started decreasing at iteration 1000 also down to iteration 4000 where it was 1.3 and in this iteration stopped decreasing and the second training ended here.

The increase in iterations doesn't mean getting better results. The number of iterations has been increased, but we did not get good results and Training has been stopped when the current average loss doesn't decrease anymore.

4. FINAL RESULTS

The latest results for each class of the data set can be summarized according to the accuracy and the percentage of true and false in precision in the (Table 4.1) below.

Table 4.1: Classes final results

Class name	Average Precision (ap)	True positive (tp)	False positive(fp)
1	100.00%	40	0
2,V	99.10%	39	2
3	100.00%	40	0
4	83.88%	18	4
5,Hello	99.62%	39	3
6,W	97.78%	39	5
7	100.00%	40	0
8	100.00%	40	0
9	100.00%	40	0
Zero ,O	100.00%	40	0
A	99.88%	40	1
B	100.00%	40	0
C	100.00%	40	0
D	100.00%	40	0
E	100.00%	39	1
F	100.00%	40	4
G	100.00%	40	0
H	100.00%	40	0
I	100.00%	40	0
J	100.00%	40	1
K	100.00%	40	0
L	100.00%	40	0
M	86.87%	35	7
N	100.00%	40	0
P	100.00%	40	1
Q	96.00%	39	1
R	100.00%	40	0
S	100.00%	40	0
T	99.40%	40	6
U	100.00%	40	5
X	92.86%	20	1
Y	100.00%	40	4
Z	100.00%	40	0
Yes	100.00%	40	0
No	100.00%	40	2
Please	76.03%	25	8

Table 4.1: Classes final results "Table continued"

Thanks	100.00%	40	0
Ok	100.00%	40	0
Sorry	89.10%	40	11
I Love You	100.00%	40	0

Examples of images testing results:

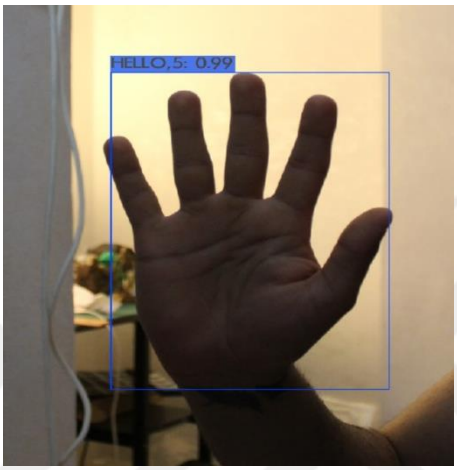

Figure 4.1: Hello

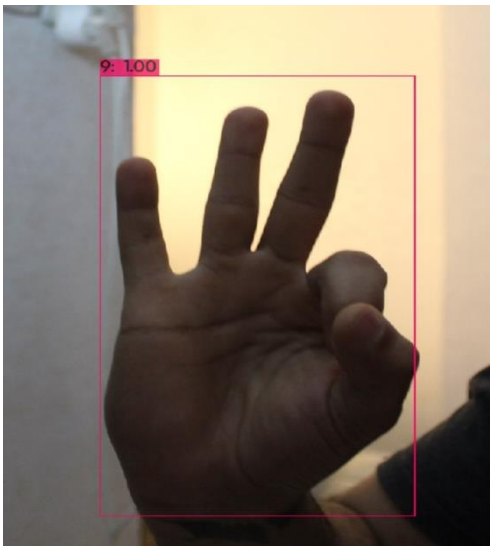

Figure 4.2: 9

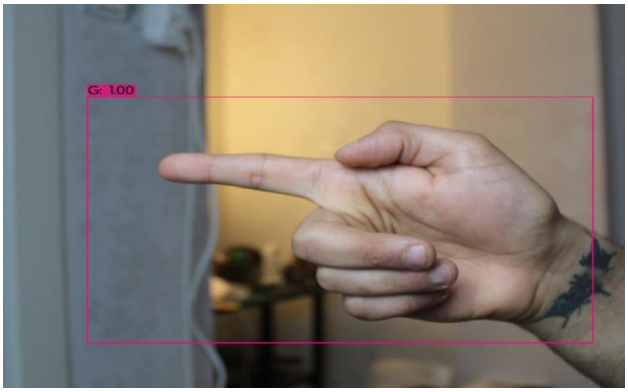

Figure 4.3: G

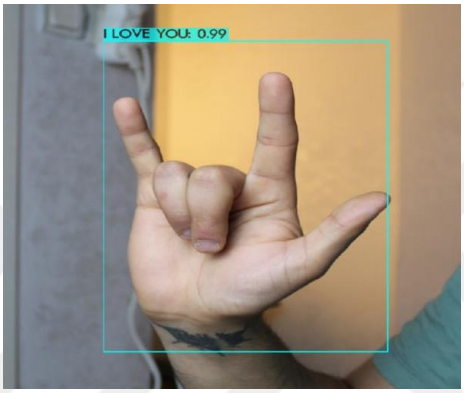

Figure 4.4: I love you

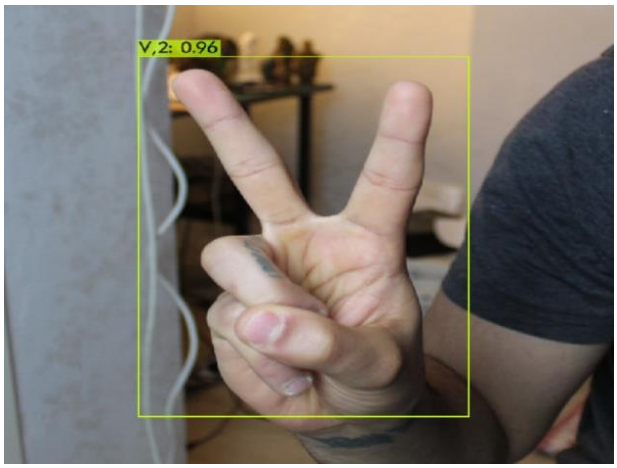

Figure 4.5: V or 2

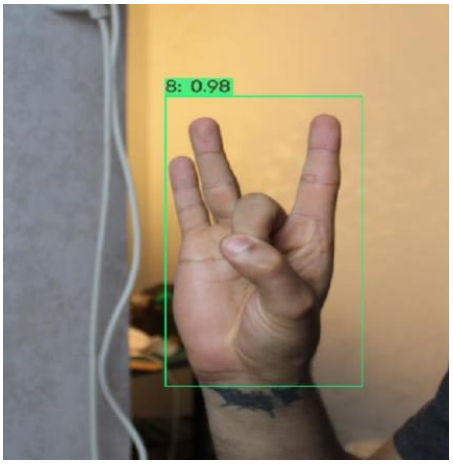

Figure 4.6: 8

And we can summarize the results in both trainings in the (Table 4.2) below

Table 4.2: Final results

Results	**First Training**	**Second Training**
Precision	0.88	0.96
Recall	0.94	0.96
F1-Score	0.91	0.96
TP	1510	1533
FP	215	67
FN	90	67
Average IoU	69.86%	76.67%
mAp	96.44%	98.01%
Detection Time	41 seconds	42 seconds

And for video testing results in first training we got 28.3 frame per second (fps) and in second training we got 28.9 frame per second (fps).

5. CONCLUSION

In this research, a recognition system for The American Sign Language (ASL) using YOLOv4 method has been done. Also we introduced a new dataset of American Sign Language. The experiment on images got 98% MAP as an accuracy which is very good and the current average loss is 1.53 and for video testing results we got 28.9 fps. In the future, we will work to increase the dataset by adding new words and signs. We will also improve the accuracy of the images in the data set to get better results.

REFERENCES

[1] C. P. Vogler, American sign language recognition: reducing the complexity of the task with phoneme-based modeling and parallel hidden markov models. University of Pennsylvania, 2003.

[2] Y. He and L. Guan, “Resource Optimization for Distributed Video Communications,” in Multimedia Image and Video Processing, Citeseer, 2017, pp. 549–570.

[3] J. Wu, A. Osuntogun, T. Choudhury, M. Philipose, and J. M. Rehg, “A scalable approach to activity recognition based on object use,” in 2007 IEEE 11th international conference on computer vision, 2007, pp. 1–8.

[4] M. Nanda, “You Only Gesture Once (YouGo): American Sign Language Translation using YOLOv3.” Purdue University Graduate School, 2020.

[5] S. Dasiopoulou, V. Mezaris, I. Kompatsiaris, V.-K. Papastathis, and M. G. Strintzis, “Knowledge-assisted semantic video object detection,” IEEE Trans. Circuits Syst. Video Technol., vol. 15, no. 10, pp. 1210–1224, 2005.

[6] C. Szegedy, A. Toshev, and D. Erhan, “Deep neural networks for object detection,” Adv. Neural Inf. Process. Syst., vol. 26, 2013.

[7] X. Wang, X. Bai, W. Liu, and L. J. Latecki, “Feature context for image classification and object detection,” in CVPR 2011, 2011, pp. 961–968.

[8] R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2014, pp. 580–587.

[9] T. Kattenborn, J. Leitloff, F. Schiefer, and S. Hinz, “Review on Convolutional Neural Networks (CNN) in vegetation remote sensing,” ISPRS J. Photogramm. Remote Sens., vol. 173, pp. 24–49, 2021.

[10] K. Ovtcharov, O. Ruwase, J.-Y. Kim, J. Fowers, K. Strauss, and E. S. Chung, "Accelerating deep convolutional neural networks using specialized hardware," Microsoft Res. Whitepaper, vol. 2, no. 11, pp. 1–4, 2015.

[11] J. Wu, "Introduction to convolutional neural networks," Natl. Key Lab Nov. Softw. Technol. Nanjing Univ. China, vol. 5, no. 23, p. 495, 2017.

[12] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol. 521, no. 7553, pp. 436–444, 2015.

[13] D. Learning, "Deep learning," High-Dimensional Fuzzy Clust., 2020.

[14] P. P. Brahma, D. Wu, and Y. She, "Why deep learning works: A manifold disentanglement perspective," IEEE Trans. neural networks Learn. Syst., vol. 27, no. 10, pp. 1997–2008, 2015.

[15] S. Minaee, A. Abdolrashidi, H. Su, M. Bennamoun, and D. Zhang, "Biometrics recognition using deep learning: A survey," arXiv Prepr. arXiv1912.00271, 2019.

[16] P. P. Shinde and S. Shah, "A review of machine learning and deep learning applications," in 2018 Fourth international conference on computing communication control and automation (ICCUBEA), 2018, pp. 1–6.

[17] A. Omidi, A. Heydarian, A. Mohammadshahi, B. A. Beirami, and F. Haddadi, "An Embedded Deep Learning-based Package for Traffic Law Enforcement," in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 262–271.

[18] A. Omidi, A. Heydarian, A. Mohammadshahi, B. A. Beirami, and F. Haddadi, "An Embedded Deep Learning-based Package for Traffic Law Enforcement," in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 262–271.

[19] J. Dalins, Y. Tyshetskiy, C. Wilson, M. J. Carman, and D. Boudry, "Laying foundations for effective machine learning in law enforcement. Majura–A labelling schema for child exploitation materials," Digit. Investig., vol. 26, pp. 40–54, 2018.

[20] G. Gensler and L. Bailey, “Deep learning and financial stability,” Available SSRN 3723132, 2020.

[21] A. Wong, A. Hryniowski, and X. Y. Wang, “Insights into fairness through trust: Multi-scale trust quantification for financial deep learning,” arXiv Prepr. arXiv2011.01961, 2020.

[22] M. Nuruzzaman and O. K. Hussain, “A survey on chatbot implementation in customer service industry through deep neural networks,” in 2018 IEEE 15th International Conference on e-Business Engineering (ICEBE), 2018, pp. 54–61.

[23] Y. Wu, W. Mao, and J. Feng, “AI for Online Customer Service: Intent Recognition and Slot Filling Based on Deep Learning Technology,” Mob. Networks Appl., pp. 1–13, 2021.

[24] A. Esteva et al., “A guide to deep learning in healthcare,” Nat. Med., vol. 25, no. 1, pp. 24–29, 2019.

[25] O. Faust, Y. Hagiwara, T. J. Hong, O. S. Lih, and U. R. Acharya, “Deep learning for healthcare applications based on physiological signals: A review,” Comput. Methods Programs Biomed., vol. 161, pp. 1–13, 2018.

[26] Y. LeCun, “1.1 deep learning hardware: Past, present, and future,” in 2019 IEEE International Solid-State Circuits Conference-(ISSCC), 2019, pp. 12–19.

[27] K. S. Zaman, M. B. I. Reaz, S. H. M. Ali, A. A. A. Bakar, and M. E. H. Chowdhury, “Custom hardware architectures for deep learning on portable devices: a review,” IEEE Trans. Neural Networks Learn. Syst., 2021.

[28] E. Alpaydin, Machine learning. MIT Press, 2021.

[29] Q. Bi, K. E. Goodman, J. Kaminsky, and J. Lessler, “What is machine learning? A primer for the epidemiologist,” Am. J. Epidemiol., vol. 188, no. 12, pp. 2222–2239, 2019.

[30] S. Sra, S. Nowozin, and S. J. Wright, Optimization for machine learning. Mit Press, 2012.

[31] T. M. Mitchell, “Does machine learning really work?,” AI Mag., vol. 18, no. 3, p. 11, 1997.

[32] J. Burrell, “How the machine ‘thinks’: Understanding opacity in machine learning algorithms,” Big Data Soc., vol. 3, no. 1, p. 2053951715622512, 2016.

[33] M. Gopal, Applied machine learning. McGraw-Hill Education, 2019.

[34] T. O. Ayodele, “Types of machine learning algorithms,” New Adv. Mach. Learn., vol. 3, pp. 19–48, 2010.

[35] K. M. Ghori, R. A. Abbasi, M. Awais, M. Imran, A. Ullah, and L. Szathmary, “Performance analysis of different types of machine learning classifiers for non-technical loss detection,” IEEE Access, vol. 8, pp. 16033–16048, 2019.

[36] V. Nasteski, “An overview of the supervised machine learning methods,” Horizons. b, vol. 4, pp. 51–62, 2017.

[37] F. Fabris, J. P. de Magalhães, and A. A. Freitas, “A review of supervised machine learning applied to ageing research,” Biogerontology, vol. 18, no. 2, pp. 171–188, 2017.

[38] M. Usama et al., “Unsupervised machine learning for networking: Techniques, applications and research challenges,” IEEE access, vol. 7, pp. 65579–65615, 2019.

[39] M. Khanum, T. Mahboob, W. Imtiaz, H. A. Ghafoor, and R. Sehar, “A survey on unsupervised machine learning algorithms for automation, classification and maintenance,” Int. J. Comput. Appl., vol. 119, no. 13, 2015.

[40] L. P. Kaelbling, M. L. Littman, and A. W. Moore, “Reinforcement learning: A survey,” J. Artif. Intell. Res., vol. 4, pp. 237–285, 1996.

[41] R. S. Sutton and A. G. Barto, “Introduction to reinforcement learning,” 1998.

[42] C. Ning and F. You, “Optimization under uncertainty in the era of big data and deep learning: When machine learning meets mathematical programming,” Comput. Chem. Eng., vol. 125, pp. 434–448, 2019.

[43] N. J. Majaj and D. G. Pelli, “Deep learning—Using machine learning to study biological vision,” J. Vis., vol. 18, no. 13, p. 2, 2018.

[44] C. Janiesch, P. Zschech, and K. Heinrich, “Machine learning and deep learning,” Electron. Mark., vol. 31, no. 3, pp. 685–695, 2021.

[45] P. Viola and M. Jones, “Rapid object detection using a boosted cascade of simple features,” in Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001, 2001, vol. 1, pp. I–I.

[46] P. Viola and M. J. Jones, “Robust real-time face detection,” Int. J. Comput. Vis., vol. 57, no. 2, pp. 137–154, 2004.

[47] F. Jalled and I. Voronkov, “Object detection using image processing,” arXiv Prepr. arXiv1611.07791, 2016.

[48] A. Mohan, C. Papageorgiou, and T. Poggio, “Example-based object detection in images by components,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 23, no. 4, pp. 349–361, 2001.

[49] Y. Freund, R. Schapire, and N. Abe, “A short introduction to boosting,” Journal-Japanese Soc. Artif. Intell., vol. 14, no. 771–780, p. 1612, 1999.

[50] Z. Zou, Z. Shi, Y. Guo, and J. Ye, “Object detection in 20 years: A survey,” arXiv Prepr. arXiv1905.05055, 2019.

[51] D. G. Lowe, “Object recognition from local scale-invariant features,” in Proceedings of the seventh IEEE international conference on computer vision, 1999, vol. 2, pp. 1150–1157.

[52] D. G. Lowe, “Distinctive image features from scale-invariant keypoints,” Int. J. Comput. Vis., vol. 60, no. 2, pp. 91–110, 2004.

[53] S. Belongie, J. Malik, and J. Puzicha, “Shape matching and object recognition using shape contexts,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 24, no. 4, pp. 509–522, 2002.

[54] P. Felzenszwalb, D. McAllester, and D. Ramanan, “A discriminatively trained, multiscale, deformable part model,” in 2008 IEEE conference on computer vision and pattern recognition, 2008, pp. 1–8.

[55] P. F. Felzenszwalb, R. B. Girshick, and D. McAllester, “Cascade object detection with deformable part models,” in 2010 IEEE Computer society conference on computer vision and pattern recognition, 2010, pp. 2241–2248.

[56] T. Malisiewicz, A. Gupta, and A. A. Efros, “Ensemble of exemplar-svms for object detection and beyond,” in 2011 International conference on computer vision, 2011, pp. 89–96.

[57] P. F. Felzenszwalb, R. B. Girshick, and D. McAllester, “Cascade object detection with deformable part models,” in 2010 IEEE Computer society conference on computer vision and pattern recognition, 2010, pp. 2241–2248.

[58] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, “Object detection with discriminatively trained part-based models,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 32, no. 9, pp. 1627–1645, 2009.

[59] R. Girshick, P. Felzenszwalb, and D. McAllester, “Object detection with grammar models,” Adv. Neural Inf. Process. Syst., vol. 24, 2011.

[60] R. B. Girshick, From rigid templates to grammars: Object detection with structured models. The University of Chicago, 2012.

[61] R. J. T. JitendraMalik, “Rich feature hierarchies for accurate object detection and semantic segmentation.”

[62] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” Adv. Neural Inf. Process. Syst., vol. 25, 2012.

[63] K. E. A. Van de Sande, J. R. R. Uijlings, T. Gevers, and A. W. M. Smeulders, “Segmentation as selective search for object recognition,” in 2011 international conference on computer vision, 2011, pp. 1879–1886.

[64] R. B. Girshick, P. F. Felzenszwalb, and D. McAllester, “Discriminatively trained deformable part models, release 5,” 2012.

[65] K. He, X. Zhang, S. Ren, and J. Sun, “Spatial pyramid pooling in deep convolutional networks for visual recognition,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 37, no. 9, pp.

1904– [1] R. Girshick, “Fast r-cnn,” in Proceedings of the IEEE international conference on computer vision, 2015, pp. 1440–1448.1916, 2015.

[66] R. Girshick, “Fast r-cnn,” in Proceedings of the IEEE international conference on computer vision, 2015, pp. 1440–1448.

[67] S. Ren, K. He, R. Girshick, and J. Sun, “Faster r-cnn: Towards real-time object detection with region proposal networks,” Adv. Neural Inf. Process. Syst., vol. 28, 2015.

[68] M. D. Zeiler and R. Fergus, “Visualizing and understanding convolutional networks,” in European conference on computer vision, 2014, pp. 818–833.

[69] J. Dai, Y. Li, K. He, and J. Sun, “R-fcn: Object detection via region-based fully convolutional networks,” Adv. Neural Inf. Process. Syst., vol. 29, 2016.

[70] Z. Li, C. Peng, G. Yu, X. Zhang, Y. Deng, and J. Sun, “Light-head r-cnn: In defense of two-stage object detector,” arXiv Prepr. arXiv1711.07264, 2017.

[71] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, “Feature pyramid networks for object detection,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 2117–2125.

[72] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 779–788.

[73] W. Liu et al., “Ssd: Single shot multibox detector,” in European conference on computer vision, 2016, pp. 21–37.

[74] J. Redmon and A. Farhadi, “YOLO9000: better, faster, stronger,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 7263–7271.

[75] J. Redmon and A. Farhadi, “Yolov3: An incremental improvement,” arXiv Prepr. arXiv1804.02767, 2018.

[76] Z. Zafrulla, H. Brashear, P. Yin, P. Presti, T. Starner, and H. Hamilton, “American sign language phrase verification in an educational game for deaf children,” in 2010 20th International Conference on Pattern Recognition, 2010, pp. 3846–3849.

[77] C. Oz and M. C. Leu, “American sign language word recognition with a sensory glove using artificial neural networks,” Eng. Appl. Artif. Intell., vol. 24, no. 7, pp. 1204–1213, 2011.

[78] S. Lang, M. Block, and R. Rojas, “Sign language recognition using kinect,” in International Conference on Artificial Intelligence and Soft Computing, 2012, pp. 394–402.

[79] D. Aryanie and Y. Heryadi, “American sign language-based finger-spelling recognition using k-Nearest Neighbors classifier,” in 2015 3rd International Conference on Information and Communication Technology (ICoICT), 2015, pp. 533–536.

[80] R. A. Kadhim and M. Khamees, “A real-time american sign language recognition system using convolutional neural network for real datasets,” Tem J., vol. 9, no. 3, p. 937, 2020.

[81] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 779–788.

[82] J. Du, “Understanding of object detection based on CNN family and YOLO,” in Journal of Physics: Conference Series, 2018, vol. 1004, no. 1, p. 12029.

[83] S. Daniels, N. Suciati, and C. Fathichah, “Indonesian Sign Language Recognition using YOLO Method,” in IOP Conference Series: Materials Science and Engineering, 2021, vol. 1077, no. 1, p. 12029.

[84] M. J. Shafiee, B. Chywl, F. Li, and A. Wong, “Fast YOLO: A fast you only look once system for real-time embedded object detection in video,” arXiv Prepr. arXiv1709.05943, 2017.

[85] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, “Yolov4: Optimal speed and accuracy of object detection,” arXiv Prepr. arXiv2004.10934, 2020.

[86] J. Yu and W. Zhang, “Face mask wearing detection algorithm based on improved YOLO-v4,” Sensors, vol. 21, no. 9, p. 3263, 2021.

[87] R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2014, pp. 580–587.

[88] R. Girshick, “Fast r-cnn,” in Proceedings of the IEEE international conference on computer vision, 2015, pp. 1440–1448.

[89] S. Ren, K. He, R. Girshick, and J. Sun, “Faster r-cnn: Towards real-time object detection with region proposal networks,” Adv. Neural Inf. Process. Syst., vol. 28, 2015.

[90] Y. Chen, W. Li, C. Sakaridis, D. Dai, and L. Van Gool, “Domain adaptive faster r-cnn for object detection in the wild,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 3339–3348.

[91] Z. He and L. Zhang, “Multi-adversarial faster-rcnn for unrestricted object detection,” in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 6668–6677.

[92] X. Wang, A. Shrivastava, and A. Gupta, “A-fast-rcnn: Hard positive generation via adversary for object detection,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 2606–2615.

[93] X. Wang, H. Ma, and X. Chen, “Salient object detection via fast R-CNN and low-level cues,” in 2016 IEEE International Conference on Image Processing (ICIP), 2016, pp. 1042–1046.

[94] J. Li, X. Liang, S. Shen, T. Xu, J. Feng, and S. Yan, “Scale-aware fast R-CNN for pedestrian detection,” IEEE Trans. Multimed., vol. 20, no. 4, pp. 985–996, 2017.

[95] C. Chen, M.-Y. Liu, O. Tuzel, and J. Xiao, “R-CNN for small object detection,” in Asian conference on computer vision, 2016, pp. 214–230.

[96] Z. Cai and N. Vasconcelos, “Cascade r-cnn: Delving into high quality object detection,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 6154–6162.

[97] X. Xie, G. Cheng, J. Wang, X. Yao, and J. Han, “Oriented r-cnn for object detection,” in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 3520–3529.