*** ***

***Searching In Video Frames***

***محرك بحث وسائط متعددة***

***Teem member :***

***- Abdelrahman Ali Sayed - Mohamed Youssef Mostafa***

***01093739627 01273383721***

***- Ahmed Alaa Ahmed - Mohamed Mostafa Ahmed***

***01140316573 01113464707***

***- Ahmed Hossny Hamed - Ibrahem Ahmed Hamed***

***01100643170 01146702683***

***- Nermeen Abdelazeem Abdelrazek - Sohair Ahmed Ahmed***

***Supervisor:***

***Dr: Kaled Fathy***

***Table of Contents***

Part 1

1.0 Introduction …………………………………………………………………..…………………………

1.1 Problem statement …………………………………………………….…………………………….

1.2 Methodology and steps …………………………………………..………………………………..

Part 2

2.0 What is a video? ….…………………………………………….…………………………………….

2.1.0 What is the video frame? ..........................................................................................................

2.1.1 Video frame rate ………………………………………………………………………….

2.1.2 Most commonly used frame rates ..…………………………………………….

2.1.3 Why frame rate is important in video ….………………………………………

Part 3

3.0.0 Open computer vision library ……………….…………………………………………….

3.0.1 Program to extract frames using OpenCV ……………………………………

3.1 What’s AI ? ...................................................................................................

3.2.0 What is machine learning? ........................................................................

3.2.1 Machine Learning Approaches ……………………………………………………..

3.3 What is deep learning? ................................................................................

3.4.0 Artificial neural Networks Algorithm ..…………………………………………………..

3.4.1 Artificial Neural Networks ……………………………………………………………

3.4.2 Components of simple neuron .…………………………………………………….

3.4.3 Activation function types ……………………………………………………………..

3.4.4 Cost function .…………………..……….………………………………………………..

3.4.5 Optimization algorithms ..…..……….………………………………………………

3.5.0 What is the model? ..................................................................................

3.5.1 What is a Pre-trained Model? ..........................................................

3.5.2 Deep Learning Approaches ………………………………………………………….

3.5.3 What is yolo v3 model? ....................................................................

3.5.4 The network structure for YOLO V3 .…………………………………………….

3.5.5 How yolo v3 works? ….......................................................................

3.5.6 Why we use yolo v3 in this project?..................................................

Part 4

4.0 What is Classification? ..................................................................................

4.0.1 Image classification ..…………………………………………………………………….

4.2 Localization ………………………………………….…………………………………………………..

4.3 Instance segmentation ..…………….…………………………………………………………….

4.4.0 Object detection ………….………………………………………………………………………..

4.4.1 What is Object Detection?

4.4.2 Object Detection Challenges …..…………………………………………………….

Part 5

5.0 Tool identification …………………………………………………………………………………….

Part 6

5.0 References ……………………………………………..……………………………………………….

1. ***Introduction***

Video files are files that store motion pictures and sounds of real-life. Video is a form of animation that gathers many images appearing continuously in sequence. Usually one second of video has 24 (or maybe more) still images and 24 or more photos, the human eyes (with retinal image retention mechanism of 1/24 sec.) will feel the motion image as continuous, smooth and beautiful.

First, we want to detect all interesting objects in video. Then we want to identify their properties, and then we want to recognize objects which are in the video. That’s we want from video analysis [1].

After that, when the user searching for objects the application get all videos that contain this object and the time which this object appear and the end.

**Overview about searching of objects**

As we all know, video is a collection of discrete images that are constantly displayed to create motion effects. Thus, object detection in video files is also based on the idea of detecting objects in an image file. We need to perform the image-partitioning step to determine where the area of objects is, and where the area of the background is. Image segmentation is an important step in image processing, and this phase analyses the image into components of the same nature based on the boundaries or interconnected regions. An image is a detail, an object in panorama. An image area describes the surface properties of an image; this area is surrounded by a boundary and points with a relatively uniform grey level. Based on the physical properties of the image area, we define a number of partitioning techniques. The main partitioning methods include:

- Classification or threshold-based method.

- Structure-based method.

- Boundary-based method [2].

* 1. ***Problem Statement***

Detecting and tracking moving objects in video files has wide range of applications in real life. If a person just watches a video file, human eyes will not be able to give specific details about the time and the process of changing location of objects, especially those with a fast change process such as detection of human, cars on the road and others objects. In addition, the object detection in the video file at the moment will help a lot in real life, we will cover object detection in videos, including image classification , localization , object recognition and image search, various object detection techniques, motion estimation, object tracking in video.

* 1. ***Methodology and steps***

Object detection in video files should be based on the object's history of motion over discrete images over time. In function I(x, y, t), ‘I’ is an image, ‘x’, ‘y’ are the coordinates of the object's location in two-dimensional space, and ‘t’ is the time of the object at that coordinate. An approach based on the history of motion pictures is a view based on the method of pattern detection over time.

This is a simple but effective way of displaying motions, it is used in a number of studies related to motion recognition, motion analysis, and other related applications. An overview of real-time video detection, classification, and mark up of motion shows that this system can implement the following operations:

(1) distinguish between transient or stationary objects and static background objects in the scene.

(2) detect and distinguish objects, which move and disappear.

(3) classify objects that have been discovered in different groups such as people, groups, vehicles, and others;

(4) mark the objects and generate information such as the paths in the video image, video name, start time that object is appear and end time .

Video processing-based applications have common characteristics. Commonly used technologies for detecting motion objects are background subtraction, static methods, time difference, optical flow, and others.

Get videos then write the object name to search

Analysis the video and get the images by using open CV library

detect the objects in every frame

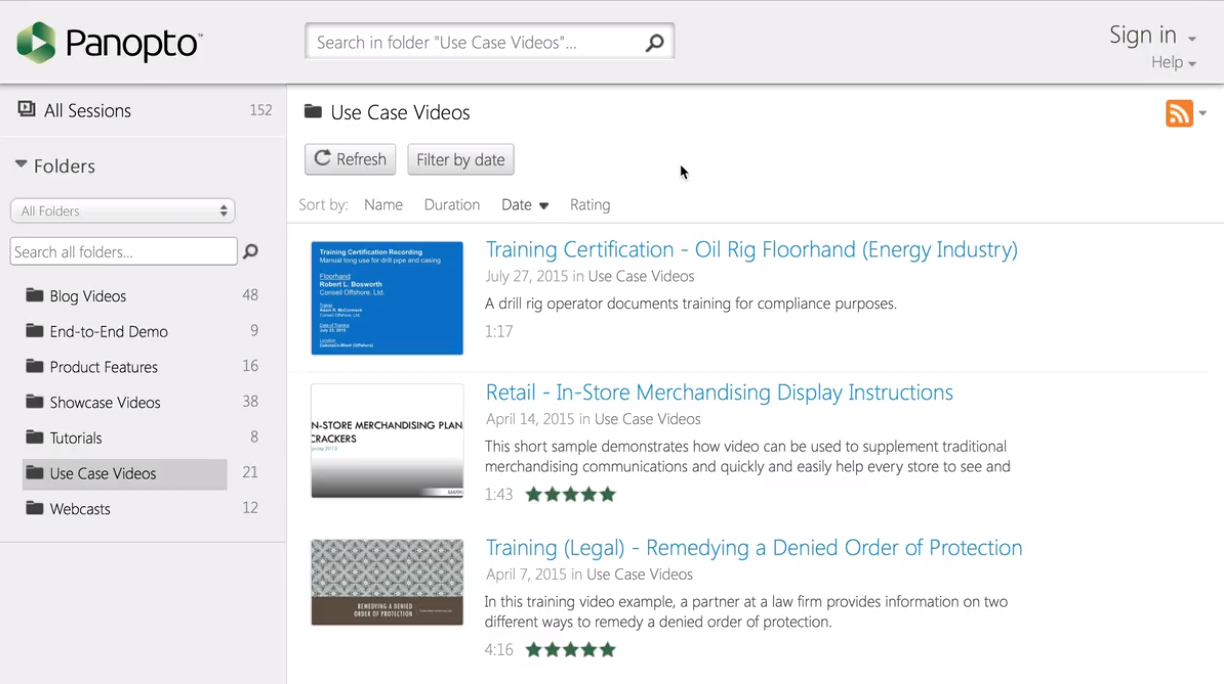
Recognize and tag the objects

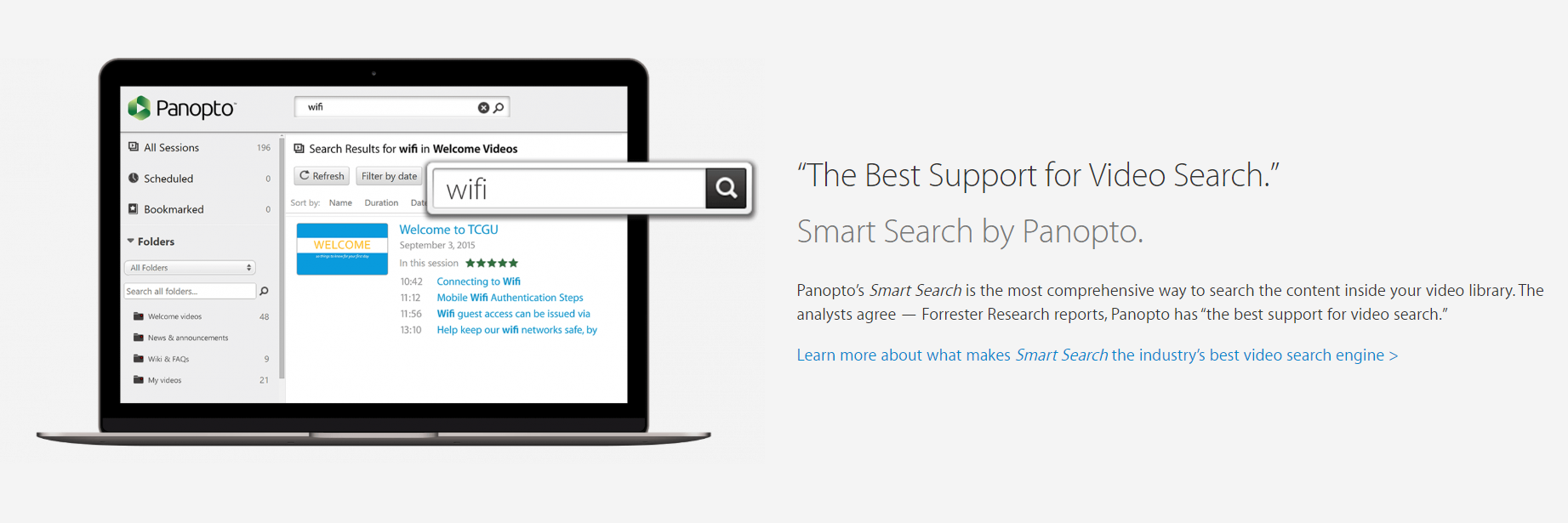
Showing the videos that contain this object and the time that is appear

***2.0 Provided solutions***

Searching in videos there are many companies have produced software to do the same task.

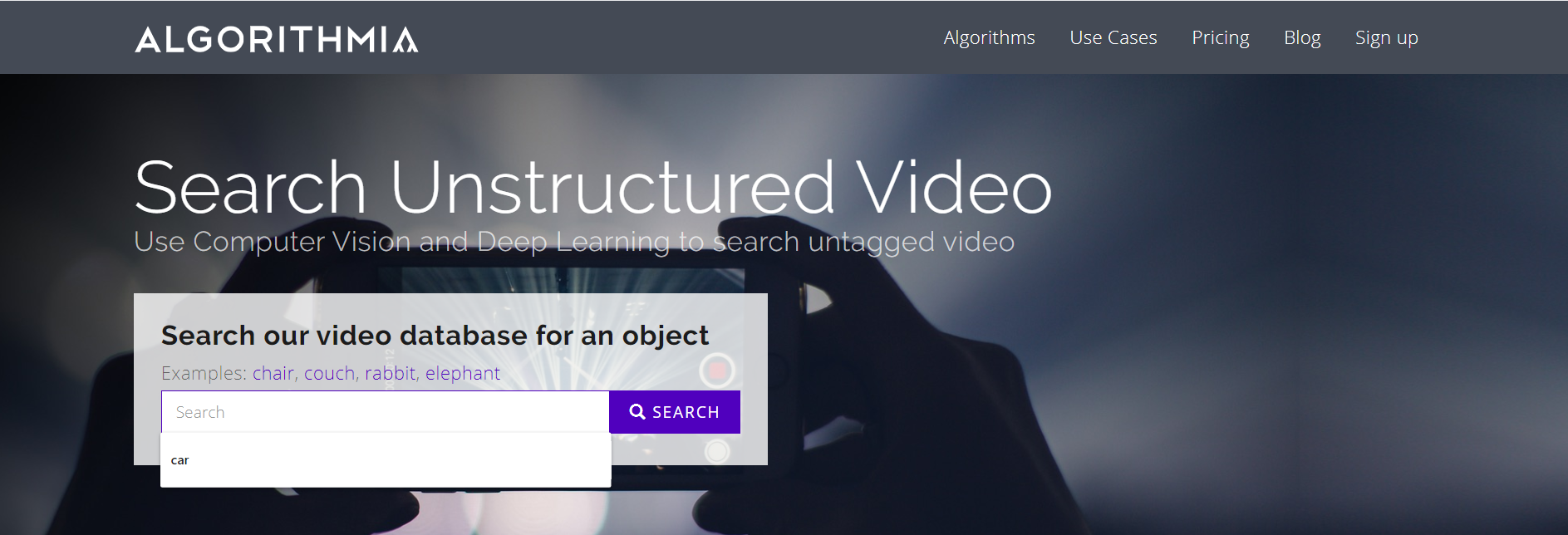
1. Panopto Application





*Smart Search* automatically indexes every word spoken or shown on-screen, so your viewers can find and fast-forward to the exact moment in videos where their search was mentioned. Whether the video was created with Panopto or not, every video uploaded to your video portal is searchable.[28]

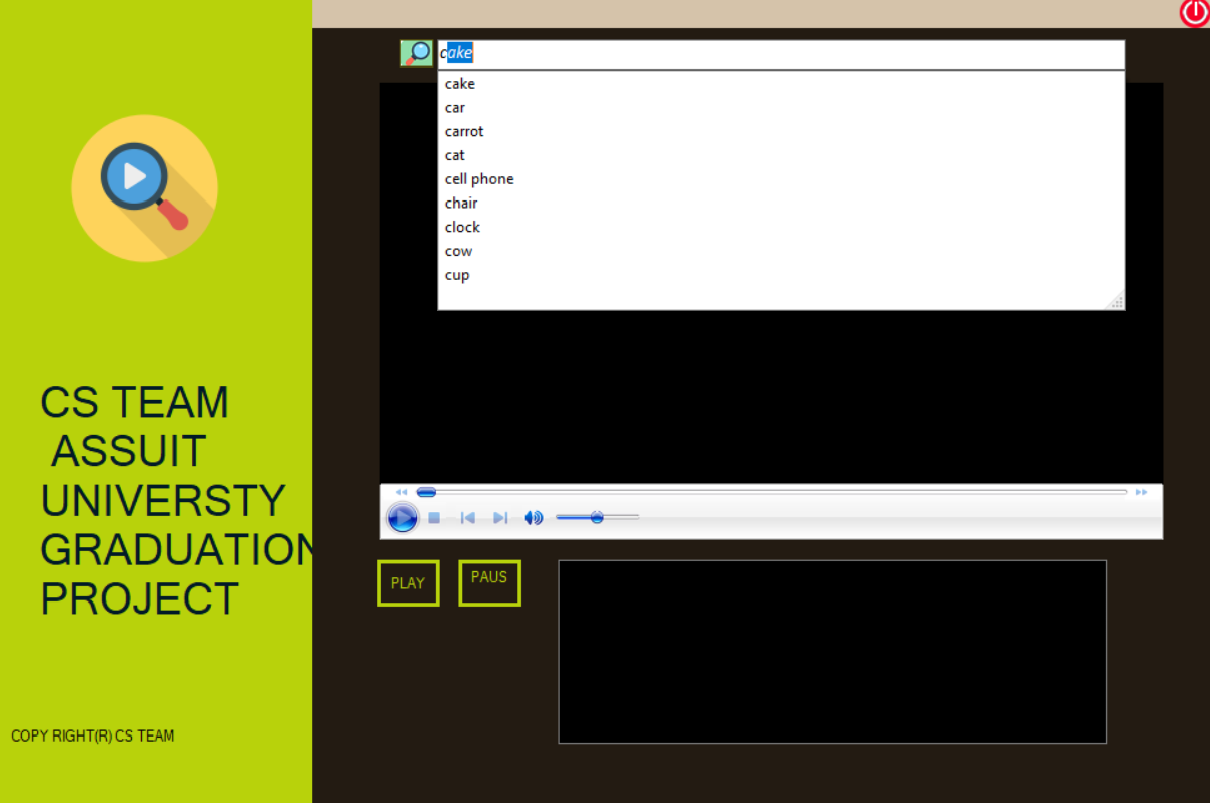
1. **Algorithmia website**

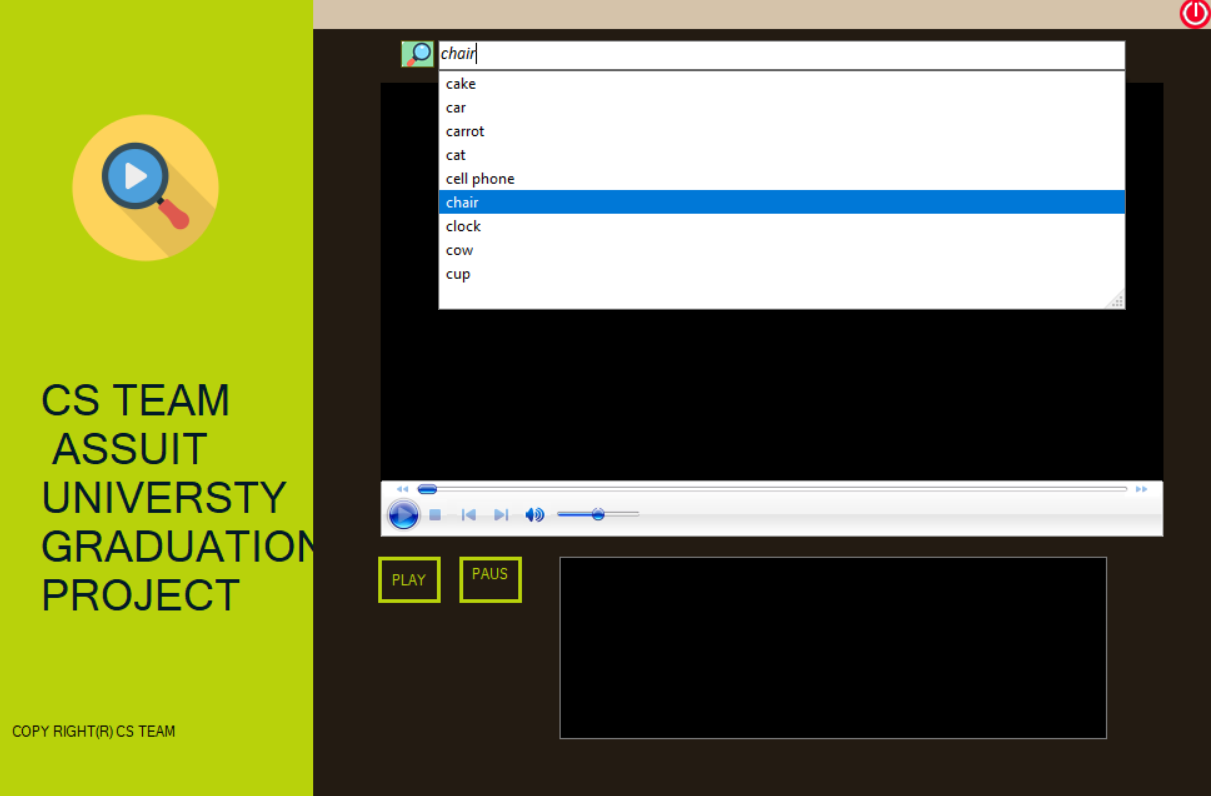


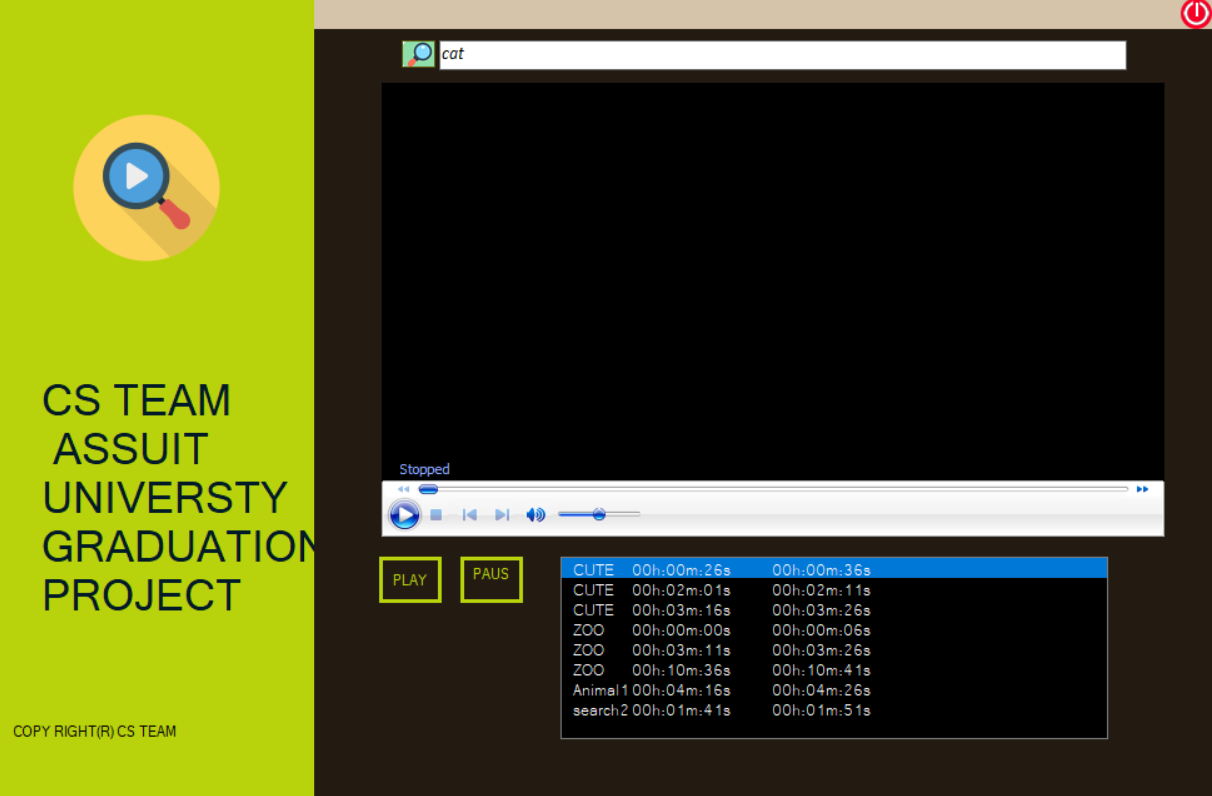
This web site give it object name and show all videos that have this object but we can’t upload any video to search on it .

This website use custom data set to search on it.

Our software is different such that use local data set , give object name and show all video parts that have this object and can give it video and do processing on it.







***2.0 What is a video?***

Video is just an older set of frames of the same resolution; usually frames are taken at regular time intervals. When constructing the video procession algorithm, we divide the video into two classes. Video stream is an ongoing video for online processing. In processing video stream, we don't know the future frames. Video sequence is a video of fixed lens. All frames are available at once, so we can process video sequence as a whole object. Video is much larger object than an image. Frame width of consumer video is usually the line range from 3-5 image per second to 30 or 50 frames per second. The resolution can be up to Full HD or 4K right now. So, the uncompressed data stream from Full HD video can reach 300 megabytes per second. [1]

***2.1.0 What is the video frame?***

In the computer world, a frame can be many different things. The different definitions of "frame" are listed below:

In video and animation, frames are individual pictures in a sequence of images. For example, a Flash movie you see on the Web may play 12 frames per second, creating the appearance of motion. Most video is shot at 24 or 30 frames per second, or FPS. FPS is often measured in 3D games as a way of checking how fast the graphics processor of a computer is, that’s we used in our project.[3]

In motion pictures, television, and in computer video displays, the frame rate is the number of frames or images that are projected or displayed per second. Frame rates are used in synchronizing [audio](https://whatis.techtarget.com/definition/audio) and pictures, whether film, television, or video. In motion pictures and television, the frame rates are standardized by the Society of Motion Picture and Television Editors (SMPTE). SMPTE Time Code frame rates of 24, 25 and 30 frames per second are common, each having uses in different portions of the industry. The professional frame rate for motion pictures is 24 frames per second and, for television, 30 frames per second (in the U.S.).[3]

In computer video streams, the frame rate describes playback rates for AVI and QuickTime movies. The video playback rate for an AVI or QuickTime movie directly relates to the perceived smoothness of its playback. The higher the number of frames playing per second, the smoother the video playback appears to the user. Lower rates result in a choppy playback. (As a reference point, film uses 24 frames per second to allow the viewer to perceive smooth playback.) Several factors affect the actual frame rate you get on your computer. For example, your PC [processor](https://whatis.techtarget.com/definition/processor) or graphics hardware may only be capable of playing 10-15 frames per second without acceleration.[3]

***2.1.1 Video frame rate***

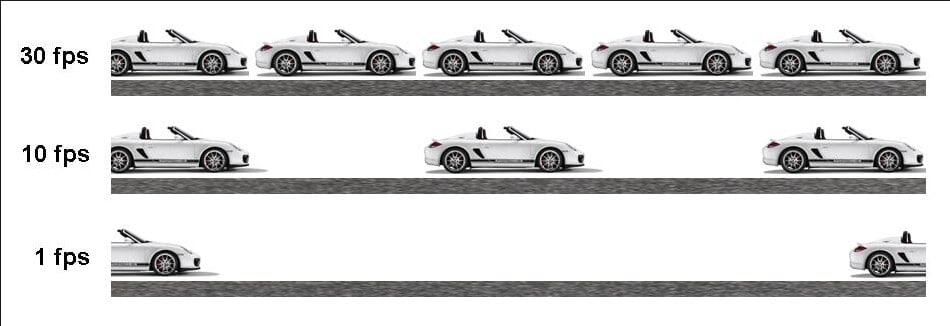
**A frame rate is simply the frequency at which independent still images appear on the screen. This means that you can choose the number of stills that are displayed in one second.** If you increase the number of frames per second the figures in the shot will appear to be moving slower, while if you decrease the frame rate the characters in the shot will move faster, like in the early Charlie Chaplin movies.

As a matter of fact, during the period when Chaplin made his first silent movies, all cameras were cranked which means that the camera operator had to roll the handle in order to start recording. Maintaining the permanent frame rate manually is not easy, which is why frame rates were much slower in the early years of cinema, as they varied **between 16 to 24 fps.**

The human eye can see as much as **10 or 12 images per second** and process them separately, while **16fps is already perceived as a movement**. However, in videos with lower frame rates movements appears jerky, while the characters move at unnatural speeds.

***2.1.2 Most commonly used frame rates***

**A large majority of DSLR and movie cameras offer only three different frame rates,** the standard **24fps**, **30fps** for those who want to produce videos in the style of TV shows, and **60fps** for sports and all other scenes that involve rapid motion.



Some Web sites use HTML frames, where the pages are broken up into various areas. Each area consists of an independent Web page. Frames allow the multiple Web pages to all show up in the same page.

Graphics and desktop publishing programs also use frames. In these programs, frames are rectangular areas meant for inserting graphics and text. They allow users to place objects wherever they want to on the page.

In developing motion pictures, television, and video, frame rate information is used as a reference for audio signals. The recorded signal includes information about location in time using a 24-hour clock, and individual frame numbers. This signal is used to synchronize multiple audio and video machines during the recording and editing process. Using a master synchronizing device, the operator can issue location commands from a central machine and have all slaved machine follow the master.[4]

***2.1.3 Why frame rate is important in videos?***

Frame Rate will impact the viewing experience, so it is important to choose a proper frame rate based on your needs. In most case, the frame rate of a movie is 24FPS, because 24FPS is similar to how we see the world. But if you want to see more details of the video, you should choose a higher frame rate, such as a [sporting video](https://filmora.wondershare.com/video-editing-tips/edit-gopro-4k-video.html), which will show a lot of details. Lower frame rate will drop some details and it is widely useful when [creating animated GIFs](https://filmora.wondershare.com/animated-gif/best-gif-maker-online.html). [5] But how to get frames from video?

To do that we use open computer vision library. what is open computer vision library?

***3.0.0 Open computer vision library***

OpenCV (Open Source Computer Vision Library) is an open source computer vision and machine learning software library. OpenCV was built to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in the commercial products. Being a BSD-licensed product, OpenCV makes it easy for businesses to utilize and modify the code.

The library has more than 2500 optimized algorithms, which includes a comprehensive set of both classic and state-of-the-art computer vision and machine learning algorithms. These algorithms can be used to detect and recognize faces, identify objects, classify human actions in videos, track camera movements, track moving objects, extract 3D models of objects, produce 3D point clouds from stereo cameras, stitch images together to produce a high resolution image of an entire scene, find similar images from an image database, remove red eyes from images taken using flash, follow eye movements, recognize scenery and establish markers to overlay it with augmented reality, etc. The library is used extensively in companies, research groups and by governmental bodies. [6]

It has C++, Python, Java and MATLAB interfaces and supports Windows, Linux, [Android](https://opencv.org/android/) and Mac OS. OpenCV leans mostly towards real-time vision applications and takes advantage of MMX and SSE instructions when available. A full-featured [CUDA](https://opencv.org/cuda/)and [OpenCL](https://opencv.org/opencl/) interfaces are being actively developed right now. There are over 500 algorithms and about 10 times as many functions that compose or support those algorithms.

OpenCV is written natively in C++ and has a templated interface that works seamlessly with STL containers. [6]

***3.0.1 Program to extract frames using OpenCV***

[OpenCV](https://www.geeksforgeeks.org/tag/opencv/) comes with many powerful video editing functions. In current scenario, techniques such as image scanning, face recognition can be accomplished using OpenCV.

OpenCV library can be used to perform multiple operations on videos. Let’s try to do something interesting using CV2. Take a video as input and break the video into frame by frame and save those frame. Now, number of operations can be performed on these frames. Like reversing the video file or crop the video etc. For playing video in reverse mode, we need only to store the frames in a list and iterate reverse in the list of frames. Use reverse method of the list for reversing the order of frames in the list[7]

**Function Used :**

***VideoCapture(File\_path) :****Read the video(.mp4 format)*

***read() :****Read data depending upon the type of object that calls*

***imwrite(filename, img[, params]) :****Saves an image to a specified file.*

**As we know computer vision is a branch of deep learning ,deep learning Is a subbranch of machine learning ,machine learning is a subbranch of artificial intelligence. But what is AI?**

***3.1 What’s AI ?***

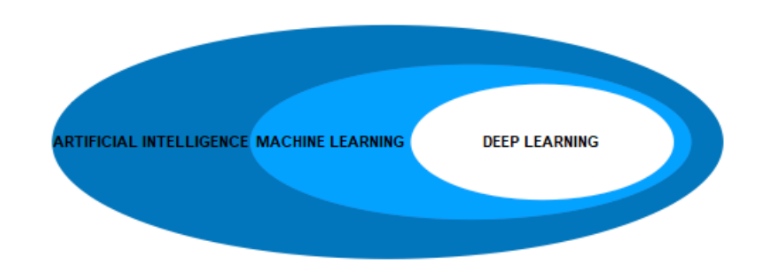
**Artificial intelligence** (AI) is the simulation of human intelligence processes by machines, especially computer systems. Specific applications of AI include [expert systems](https://searchenterpriseai.techtarget.com/definition/expert-system), natural language processing ([NLP](https://searchbusinessanalytics.techtarget.com/definition/natural-language-processing-NLP)), speech recognition and [machine vision](https://searchenterpriseai.techtarget.com/definition/machine-vision-computer-vision).

AI programming focuses on three cognitive skills: **learning, reasoning** and **self-correction**.

**Learning processes**. This aspect of AI programming focuses on acquiring data and creating rules for how to turn the data into actionable information. The rules, which are called [algorithms](https://whatis.techtarget.com/definition/algorithm), provide computing devices with step-by-step instructions for how to complete a specific task.

**Reasoning processes**. This aspect of AI programming focuses on choosing the right algorithm to reach a desired outcome.

**Self-correction processes**. This aspect of AI programming is designed to continually fine-tune algorithms and ensure they provide the most accurate results possible.[8]



***3.2.0 What is machine learning?***

**Machine Learning:** A branch of artificial intelligence where engineers and scientists manually select features within the data and train the model. Common machine learning algorithms include decision trees, support vector machines, neural networks, and ensemble methods.[9]

***3.2.1 Machine Learning Approaches***

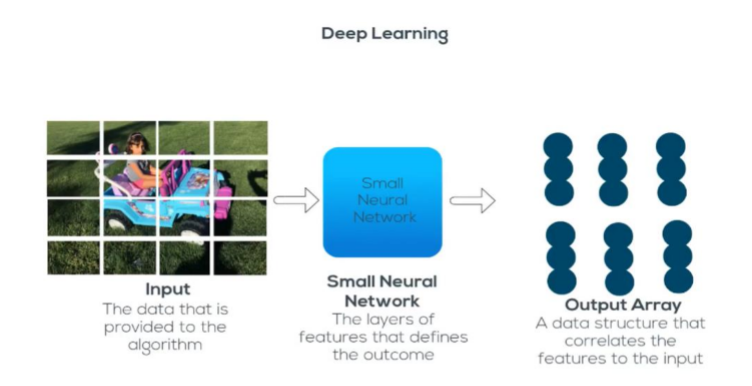
There are machine learning techniques that are used to detect an object within images, and these techniques are manually implemented algorithms, hence they are not learned systems.

Methods such as Histograms of Oriented Gradients(HOG)  introduced around 2005 used a combination HOG/SIFT(Scale Invariant Feature transformation) to identify interest points within images based on normalized local histograms of image gradients.

The method is based on the intuition that an object of interest has a distinct set of characteristics that can be identified by the local intensity of gradients and edge direction on a per-window basis. The result is to generate a HOG descriptor that is affine invariant and used in a detection chain that introduces an SVM(Support vector machine) to detect objects of interest-based on the HOG descriptors. This technique works relatively well for general detection scenarios, such as pedestrian detection.

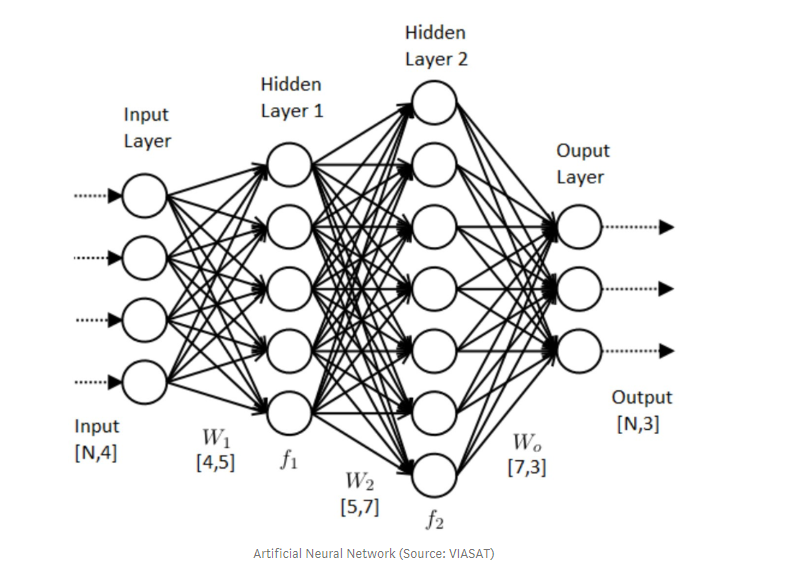
***3.3 What is deep learning?***

**Deep Learning:**  A branch of machine learning modeled loosely on the neural pathways of the human brain where the algorithm automatically learns what features are useful. Common deep learning algorithms include convolutional neural networks (CNNs), recurrent neural networks, and deep Q networks.[9]

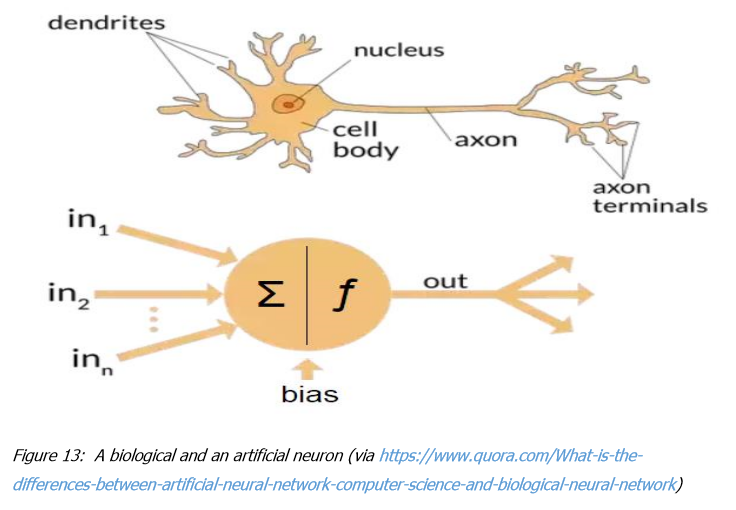


***3.4.0 Artificial neural Networks Algorithm***

**Artificial Neural Network** is a set of connected input/output units where each connection has a weight associated with it started by psychologists and neurobiologists to develop and test computational analogs of neurons. During the learning phase, the network learns by adjusting the weights so as to be able to predict the correct class label of the input tuples [10].



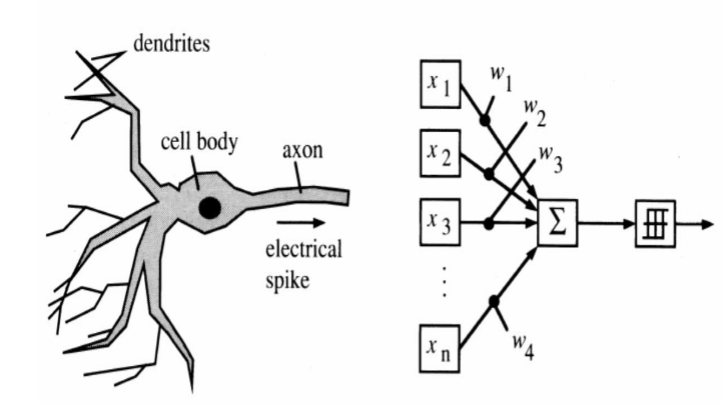
Artificial neurons and was inspired by the biological processes scientists were able to observe in the brain [11].



***3.4.1 Artificial Neural Networks***

Now that there is a general understanding of artificial neural networks, it is appropriate to explore them in greater detail. We have mentioned a word neuron. **But what is neuron?**

In biology Neurons (also called neurons or nerve cells) are the fundamental units of the brain and nervous system, the cells responsible for receiving sensory input from the external world, for sending motor commands to our muscles, and for transforming and relaying the electrical signals at every step in between. More than that, their interactions define who we are as people. Having said that, our roughly 100 billion neurons do interact closely with other cell types, broadly classified as glia (these may actually outnumber neurons, although it’s not really known) [12].This is the same concept at artificial NN.



***3.4.2 Components of simple neuron [12]***

**(1) Input Vector:** The input of technical neurons consists of many components, therefore it is a vector.

**(2) Scalar output:** The output of a neuron is a scalar.

**(3) Synapses change input:** In technical neural networks the inputs are preprocessed, too. They are multiplied by a number (the weight) – they are

weighted. The set of such weights represents the information storage of a neural network.

**(4) Accumulating the inputs:** In biology, the inputs are summarized to a pulse according to the chemical change – on the technical side this is often realized by the weighted sum.

**(5) Non-linear characteristic:** The input of our technical neurons is also not proportional to the output.

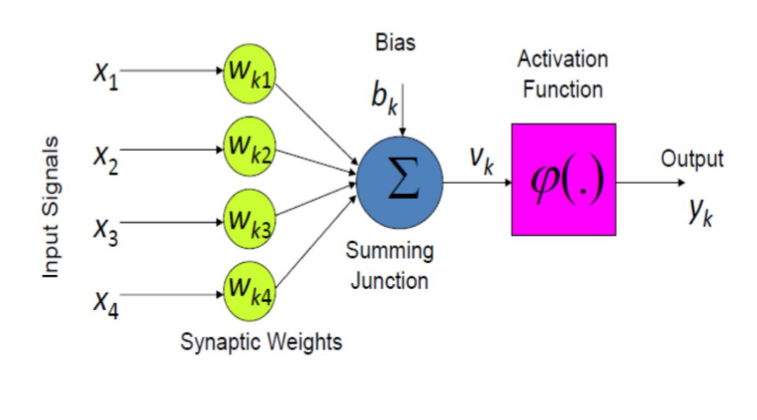
**(6) Adjustable weights:** The weights weighting the inputs are variable, similar to the chemical processes at the synaptic cleft. This adds a great dynamic to the network because a large part of the "knowledge" of a neural network is saved in the weights.

Input vector X with components 𝒙ᵢ these are multiplied by the appropriate weights 𝒘ᵢ and accumulated (called weighted sum) .

Nonlinear mapping f defines the scalar output y:

**y** = **f**( **)**

The model consists of a set of synapses each of which is characterized by a weight or strength of its own. An adder, an activation function and a bias [13].



Now we know about inputs, weights, and ***V****ᴋ* or **Z** in other books that equal but what about y that consider a function in **Z** or ***V****ᴋ* as you want to say. This function called activation function. If you have some mathematical background you may ask what is a formula of this function. This question will take us to this part.

***3.4.3 Activation function types***

Before talk about activation function types I like to tell you what is really activation function do. Activation functions are really important for an Artificial Neural Network to learn and make sense of something really complicated and Non-linear complex functional mappings between the inputs and response variable [14]. They introduce non-linear properties to our Network [14]. Their main purpose is to convert a input signal of a node in a A-NN to an output signal [14]. That output signal now is used as a input in the next layer in the stack [14].

Specifically in A-NN we do the sum of products of inputs(X) and their corresponding Weights (W) and apply an Activation function f(x) to it to get the output of that layer and feed it as an input to the next layer [14].

**Most popular types of Activation functions**

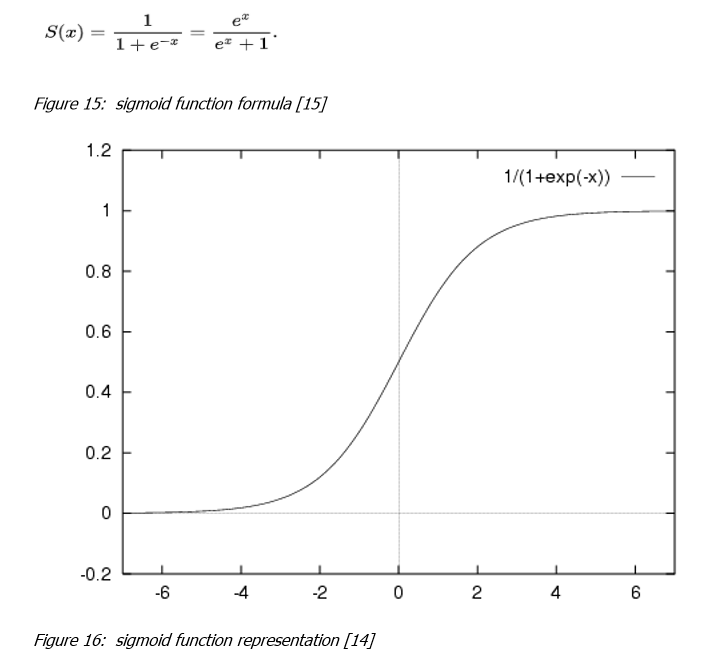
(1) Sigmoid or Logistic

(2) Tanh - Hyperbolic tangent

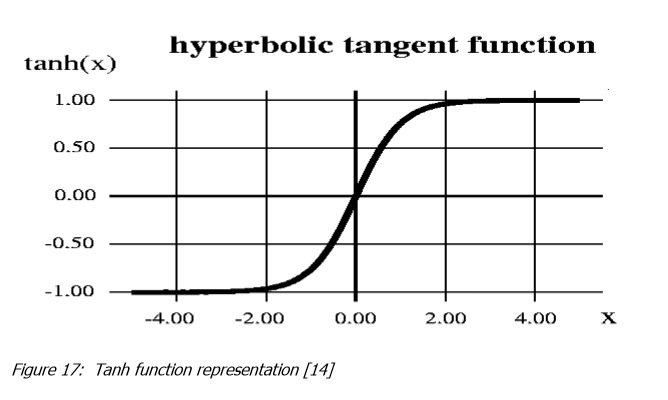
(3) ReLu -Rectified linear units

**Let's talk about each one separately**

**A sigmoid function** is a mathematical function having a characteristic "S"-shaped curve or sigmoid curve. Often, sigmoid function refers to the special case of the logistic function shown in the first figure and defined by the formula [15].



**Hyperbolic Tangent function- Tanh**: its mathematical formula is **f(x) = 1 —  exp(-2x) / 1 + exp(-2x).** Now its output is zero centered because its range in between -1 to 1 i.e -1 < output < 1. Hence optimization is easier in this method hence in practice it is always preferred over Sigmoid function. But still it suffers from Vanishing gradient problem [14].



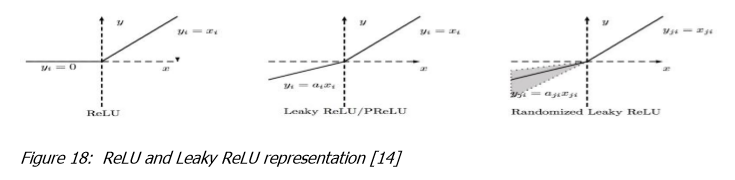
**ReLu- Rectified Linear units:** It has become very popular in the past couple of years. It was recently proved that it had 6 times improvement in convergence from Tanh function. It’s just R(x) = max(0, x) i.e if x < 0, R(x) = 0 and if x >= 0, R(x) = x. Hence as seeing the mathematical form of this function we can see that it is very simple and efficient. A lot of times in Machine learning and computer science we notice that most simple and consistent techniques and methods are only preferred and are best. Hence it avoids and rectifies vanishing gradient problem. Almost all deep learning Models use ReLu nowadays [14].

**But its limitation is that it should only be used within Hidden layers of a Neural Network Model.**

Hence for output layers we should use a **Softmax** function for a Classification problem to compute the probabilities for the classes, and for a regression problem it should simply use a **linear** function [14]. Without going into many details Softmax function used in NN when I have multi classification problem but we don't need here because our problem is binary classification problem.

Another problem with ReLu is that some gradients can be fragile during training and can die. It can cause a weight update which will makes it never activate on any data point again. Simply saying that ReLu could result in Dead Neurons [14].

To fix this problem another modification was introduced called **Leaky ReLu** to fix the problem of dying neurons. It introduces a small slope to keep the updates alive.



After we spoken about activation functions and their types we must ask a question why we can’t do NN without activating the input signal.

**Why we can’t do NN without activating the input signal?**

If we do not apply an Activation function then the output signal would simply be a simple linear function. A linear function is just a polynomial of **one degree.** Now, a linear equation is easy to solve but they are limited in their complexity and have less power to learn complex functional mappings from data.

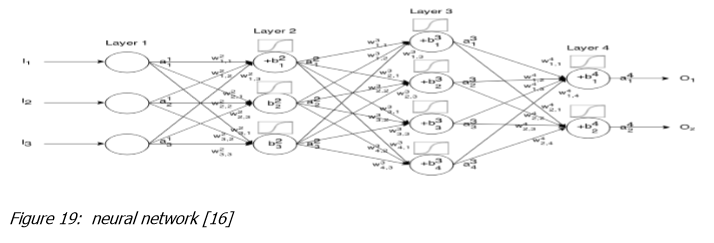
A Neural Network without Activation function would simply be **a linear regression Model**, which has limited power and does not performs good most of the times. We want our Neural Network to not just learn and compute a linear function but something more complicated than that.

Also without activation function our neural network would not be able to learn and model other complicated kinds of data such as images, videos, audio, speech etc. That is why we use Artificial Neural network techniques such as Deep learning to make sense of something complicated ,high dimensional, on-linear -big datasets, where the model has lots and lots of hidden layers in between and has a very complicated architecture which helps us to make sense and extract knowledge form such complicated big datasets [14]. After we talk about weights that consider as factors for inputs and get **Z** that consider a summation of weights multiple by inputs values what about if we have an error in classification result. To handle this error and prevent to occur again or reduce the chance of happening again we need what is called a cost function **but what is cost function?**

***3.4.4 Cost function***

**Cost function** is a measure of "how good" a neural network did with respect to its given training sample and the expected output. It also may depend on variables such as weights and biases. A cost function is a single value, not a vector, because it rates how good the neural network did as a whole [27].

Specifically, a cost function is of the form **C ( W, B, , )** where W is our neural network's weights, B is our neural network's biases, is the input of a single training sample, and is the desired output of that training sample. Note this function can also potentially be dependent on and for any neuron j in layer i, because those values are dependent on W, B, and .



We said that the input X provides the initial information that then propagates to the hidden units at each layer and finally produce the output. The architecture of the network entails determining its depth, width, and activation functions used on each layer. Depth is the number of hidden layers [15]. Width is the number of units (nodes) on each hidden layer since we don’t control neither input layer nor output layer dimensions. There are quite a few set of activation functions such Rectified Linear Unit, Sigmoid, Hyperbolic tangent, etc. Research has proven that deeper networks outperform networks with more hidden units. Therefore, it’s always better and won’t hurt to train a deeper network (with diminishing returns) [15**]. This called forward propagation**.

We mentioned that we need a cost function to reduce the chance of error from occur again this information not completely true. We need a cost function to get indication about efficient or performance of neural network. **The higher the cost function, the more the neural network suffers from some problems. Our goal to reduce the cost function value** for this we need what is called a **backpropagation** but **what is back-propagation?**

**Back-Propagation** Allows the information to go back from the cost backward through the network in order to compute the gradient. Therefore, loop over the nodes starting at the final node in reverse topological order to compute the derivative of the final node output with respect to each edge’s node tail. Doing so will help us know who is responsible for the most error and change the parameters in that direction [15].

At back-propagation step we need what is called **optimization algorithm** to optimize a cost function but **what is optimization algorithm**.

***3.4.5 Optimization algorithms***

**Optimization algorithm** is a procedure which is executed iteratively by comparing various solutions till an optimum or a satisfactory solution is found [16].

**Most popular types of Activation functions**

(1) Gradient Descent

(2) Gradient Descent with momentum

(3) RMSProp

(4) Adam

**As we mention we do this to make a model. But what is the model?**

***3.5.0 What is the model ?***

**A model** said in a very simplified form is nothing but a function that is used to take in certain input, perform certain operation to its best on the given input (learning and then predicting/classifying) and produce the suitable output.

Eg. Linear regression model has the linear function equation as its heart that maps the input to the output linearly.[17]But don’t make a model, we used a pre-trained model. what is it and why we use it , that which we talk about in next part.

***3.5.1 What is a Pre-trained Model?***

Simply put, a pre-trained model is a model created by some one else to solve a similar problem. Instead of building a model from scratch to solve a similar problem, you use the model trained on other problem as a starting point.

For example, if you want to build a self learning car. You can spend years to build a decent image recognition algorithm from scratch or you can take inception model (a pre-trained model) from Google which was built on ImageNet data to identify images in those pictures.

A pre-trained model may not be 100% accurate in your application, but it saves huge efforts required to re-invent the wheel. Let me show this to you with a recent example.[18]

**The Most common Models in object detection**

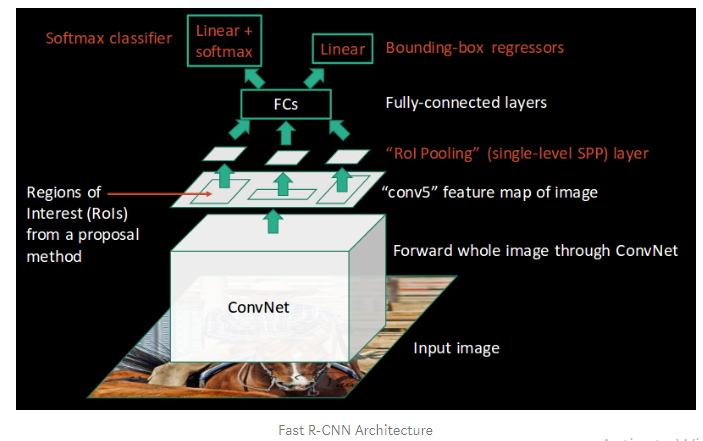
* **R-CNN**
* **Fast R-CNN**
* **Faster R-CNN**
* **Mask R-CNN**
* **SSD(single shot multibox defender)**
* **YOLO (You Only Look Once)**
* **Objects as Points**
* **Data Augmentation Strategies for Object Detection**

**3.5.2 Deep Learning Approaches**

The three classic Deep Learning Algorithms are Region Proposals(R-CNN), Single Shot Multibox Detector(SSD), You Only Look Once (YOLO), and Single Shot Refinement Neural Network for Object Detection (RefineDet).

**R-CNN**

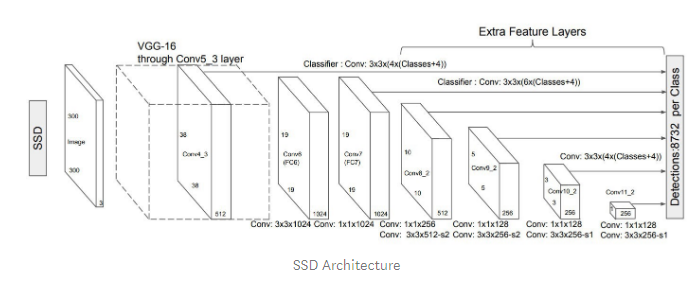
**The R-CNN** algorithm chooses a number of regions (two thousand originally) and proceeds to extract the features of each region separately using the convolutional neural network (CNN).



The algorithm then proceeds to search for a mixture of those regions in the test images. This algorithm is sometimes called a two-shot algorithm because it takes one shot to generate region proposals and another to detect the objects of each proposal. This algorithm has evolved into faster and simpler implementations that I will discuss in a future article.

**SSD**

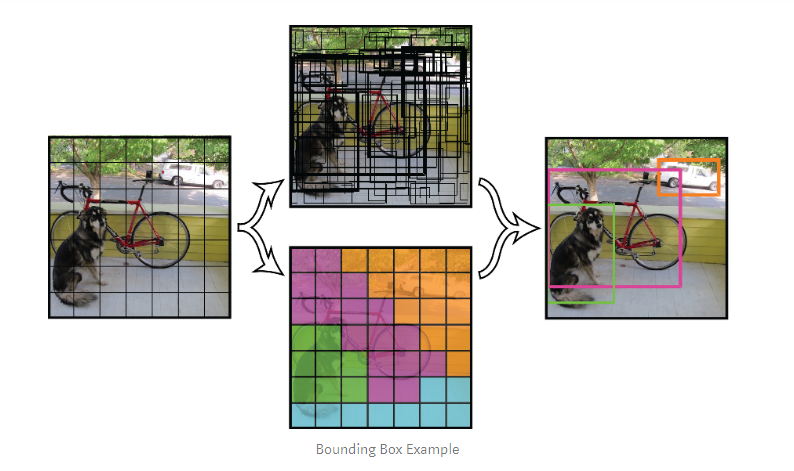
**The Single Shot Multibox Detector** (SSD) is an interesting algorithm that failed to get much traction in the community. The main idea behind this network is that CNN gradually shrinks the feature map size and increases the depth as it moves to the deeper layers.



Deep layers cover large receptive fields and construct more abstract representation while the shallow layers cover small receptive fields. Using this logic, we can simply say that shallow layers predict small objects and deep layers predict big objects. This simple logic means that we only take one single shot to detect multiple objects within the image, while in the R-CNN methods we take two shots. Obviously, this is hugely beneficial. I would argue that it was not embraced for the simple reason that hardware allowed for better-performing algorithms to be utilized.

**YOLO**

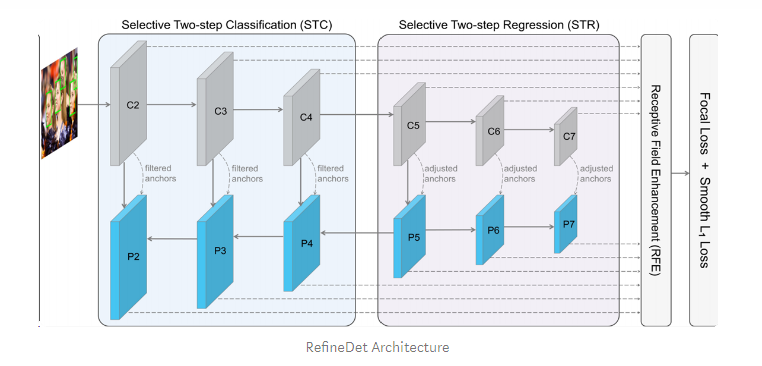
YOLO, in my opinion, was truly a revolutionizing algorithm because it took into account a whole image and thus utilized the full power of the CNN. In this algorithm, a single convolutional network predicts the bounding boxes and the class probabilities for these boxes.



The bounding boxes having the class probability above a threshold value is selected and used to locate the object within the image. The original algorithm proposed was Darknet, but since then various different algorithms have been proposed since it was utilized. In the future, I will give an implementation that utilized MobileNet.

**RefineDet**

The Single-Shot Refinement Neural Network for Object Detection (RefineDet) is a new novel approach that attempts to combine the R-CNN and SSD method.



The network consists of two inter-connected modules: the anchor refinement and object detection modules. The author made the code of this model public and I will be publishing a simple implementation of the code in the next few weeks.[19]

**The pre-Trained model that uses in this project is yolo v3 model.**

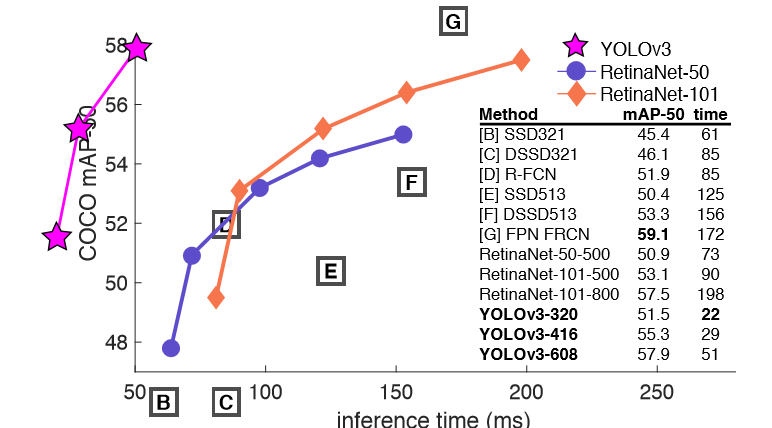
***3.5.3 What is yolo v3 model?***

YOLOv3 (You Only Look Once), is a model for object detection. The object detection task consists of determining the location on the image where certain objects are present, as well as classifying those objects. Previous methods for this, like R-CNN and its variations, used a pipeline to perform this task in multiple steps. This can be slow to run and also hard to optimize because each individual component must be trained separately. YOLOv3, does it all with a single neural network.

YOLO v3 makes prediction at three scales, which are precisely given by down-sampling the dimensions of the input image by 32, 16 and 8 respectively.[20]

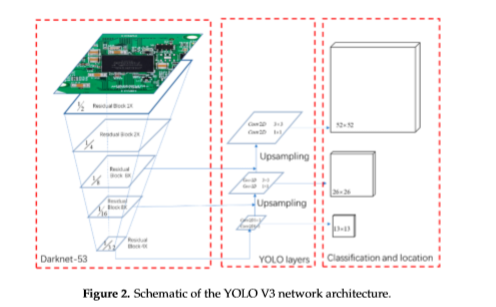
YOLO v3 uses a variant of Darknet, which originally has 53 layer network trained on Imagenet. For the task of detection, 53 more layers are stacked onto it, giving us a 106 layer fully convolutional underlying architecture for YOLO v3. This is the reason behind the slowness of YOLO v3 compared to YOLO v2.

YOLO v3 is the latest variant of a popular object detection algorithm YOLO – You Only Look Once. The published model recognizes 80 different objects in images and videos, but most importantly it is super fast and nearly as accurate as Single Shot MultiBox (SSD).[21]



experiments with COCO dataset

***3.5.4 The network structure for YOLO V3***



Darknet Architecture is pre-trained model for classifying 80 different classes. Our goal now is that we will use Darknet(YOLOv3) in OpenCV to classify objects using Python language.[22]

***3.5.5 How yolo v3 works ?***

Prior detection systems repurpose classifiers or localizers to perform detection. They apply the model to an image at multiple locations and scales. High scoring region of the image are considered detections.

We use totally different approach. We apply a single neural network to the full image. This network divides the image into regions and predicts bounding boxes and probabilities for each region. These bounding boxes are weighted by the predicted probabilities.

**This model has several advantages over classifier-based systems**.

It looks at the whole image at test time so its predictions are informed by global context in the image. It also makes predictions with a single network evaluation unlike systems like R-CNN which require thousands for a single image . this makes it extremely fast mor tha1000x faster than R-CNN and 100x faster than.[23]

***3.5.6 Why we use yolo v3 in this project ?***

YOLOv3 proved to be a very fast object detection system and performs well on real-life traffic situations. As with any other AI approach, test-and-adjust is the best way to reach optimal results.

**we use yolo v3 model for several reasons:**

-This model make prediction for large number of objects are 80 objects.

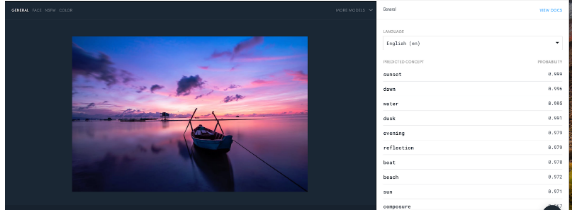
-This model is so fast because it is look to the image once and predict the objects on it.

-This model is pre-traind.

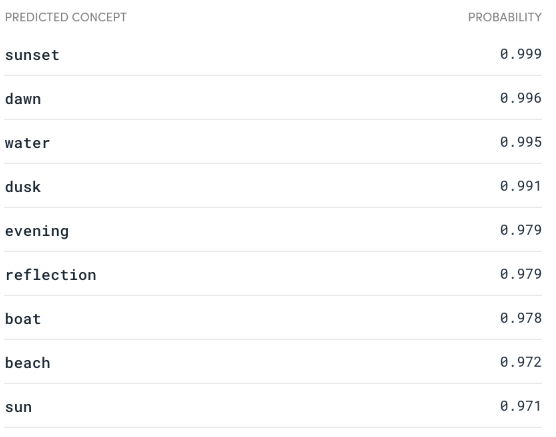
***4.0 What is Classification?***

Classification refers to a type of labeling where an image/video is assigned certain concepts, with the goal of answering the question, “What is in this image/video?”

An image can be classified into a number of categories. For example, the below screenshot, shows an image that has been uploaded to Clarifai’s General Model.



As we can see, the model has given us a list of predicted concepts. These represent how the model has classified the image with each concept representing a different “classification.”



***4.0.1 Image classification***

Probably the most well-known problem in computer vision. It consists of classifying an image into one of many different categories. One of the most popular datasets used in academia is ImageNet, composed of millions of classified images, (partially) utilized in the [ImageNet Large Scale Visual Recognition Challenge (ILSVRC)](http://www.image-net.org/challenges/LSVRC/) annual competition. In recent years classification models have surpassed human performance and it has been considered practically solved. While there are plenty of challenges to image classification, there are also plenty of [write-ups](https://medium.com/@ageitgey/machine-learning-is-fun-part-3-deep-learning-and-convolutional-neural-networks-f40359318721) on how it’s usually solved and which are the remaining challenges.



***4.1 Localization***

Similar to classification, localization finds the location of a single object inside the image.



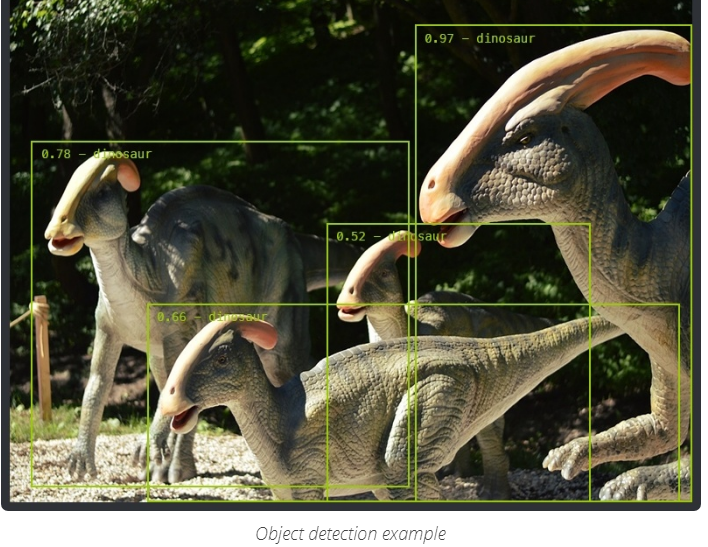
Localization can be used for lots of useful real-life problems. For example, smart cropping (knowing where to crop images based on where the object is located), or even regular object extraction for further processing using different techniques. It can be combined with classification for not only locating the object but categorizing it into one of many possible categories.

***4.2 Instance segmentation***

Going one step further from object detection we would want to not only find objects inside an image, but find a pixel by pixel mask of each of the detected objects. We refer to this problem as instance or object segmentation.

***4.3.0 Object detection***

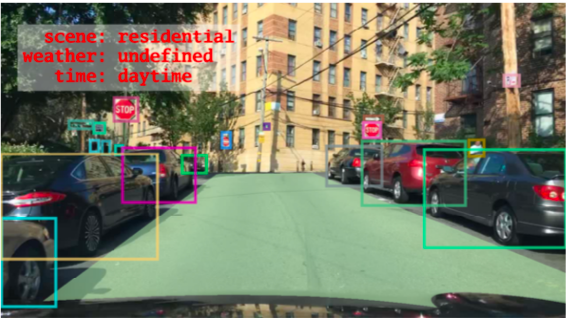
Iterating over the problem of localization plus classification we end up with the need for detecting and classifying multiple objects at the same time. Object detection is the problem of finding and classifying a variable number of objects on an image. The important difference is the “variable” part. In contrast with problems like classification, the output of object detection is variable in length, since the number of objects detected may change from image to image.[24]



***4.3.1 What is Object Detection?***

[**Object detection**](https://www.clarifai.com/blog/what-is-image-recognition-a-two-minute-rundown) is a computer vision technique that deals with distinguishing between objects in an image or video. While it is related to classification, it is more specific in what it identifies, applying classification to distinct objects in an image/video and using bounding boxes to tells us where each object is in an image/video. Face detection is one form of object detection.

This technique is useful if you need to identify particular objects in a scene, like the cars parked on a street, versus the whole image.[25]



***4.3.2 Object Detection Challenges***

A challenge within AI systems is to employ accurate object detection when in contact with fast-moving objects like vehicles. In July of 2018, researchers from the University of California created a 3-D printed neural network with the capacity to use light photons to rapidly process images and recognize certain patterns to enable object identification. The AI network had over 91 percent accuracy for object detection with the research team continuously working to improve performance.

Another challenge for AI is that studies are demonstrating fallacies within current systems. For instance, Auburn University released a November 2018 research paper that Inception, Google’s image-recognition system could be tricked by object rotations. As an example, a school bus was placed on its side by the research team with Inception incorrectly identifying it as a snowplow. The point of the study was to show Inception could be easily confused with out of the norm placement of objects.

Improvements in object detection are critical to the future of AI. If robots can understand their surroundings better, they are able to perform complex tasks.[26]

***5.0 Tools identification***

(1) Python programming language

Open source programming language for coding our models.

(2) Jupyter notebook platform

Platform to run python code on it.

(3) Spyder

Platform to run python code on it with more functionality.

(4) C# programming languages

programming language for desining GUI

(5) visual studio

Platform to run C# code on it

(6) MY sql

 an open-source relational database management system to manage output data

***6.0 References***

[1] <https://www.coursera.org/lecture/deep-learning-in-computer-vision/introduction-to-video-analysis-alApg>

[2] S. M. Bryan, Lecture 15: Segmentation (Edge Based, Hough Transform) (Brigham Young University, 2000)

[3] <https://techterms.com/definition/frame>

[4] <https://searchnetworking.techtarget.com/definition/frame-rate>

[5] <https://filmora.wondershare.com/video-editing-tips/video-frame-rate.html>

[6] <https://opencv.org/about/>

[7] <https://www.geeksforgeeks.org/python-program-extract-frames-using-opencv/>

[8] <https://searchenterpriseai.techtarget.com/definition/AI-Artificial-Intelligence>

[9] <https://www.embedded-computing.com/guest-blogs/when-to-use-machine-learning-or-deep-learning?https%3A%2F%2Fwww.embedded-computing.com%2Fprocessing=&gclid=Cj0KCQiApt_xBRDxARIsAAMUMu--8gvxhR8ilAOCMGhqavS8mpD5mowvyiQWyuDT_K3YA7oXnLjTe6caArlNEALw_wcB>

[10] https://towardsdatascience.com/machine-learning-classifiers-a5cc4e1b0623

[11] <https://towardsdatascience.com/the-differences-between-artificial-and-biological-neural-networks-a8b46db828b7>

[12] https://qbi.uq.edu.au/brain/brain-anatomy/what-neuron

[13] http://www.philadelphia.edu.jo/academics/qhamarsheh/uploads/Lecture%202%20%20 -%20Components%20of%20ANN%20and%20MATLAB%20representation.pdf

[14] https://towardsdatascience.com/activation-functions-and-its-types-which-is-bettera9a5310cc8f

[15] https://towardsdatascience.com/coding-neural-network-forward-propagation-andbackpropagtion-ccf8cf369f76

[16] https://mech.iitm.ac.in/nspch52.pdf

[17] <https://www.quora.com/What-exactly-is-a-model-in-machine-learning>

[18] <https://www.analyticsvidhya.com/blog/2017/06/transfer-learning-the-art-of-fine-tuning-a-pre-trained-model/>

[19] <https://medium.com/the-advantages-of-densenet/yolov3-pytorch-video-image-model488721214470>

[20] <https://towardsdatascience.com/pedestrian-tracking-in-real-time-using-yolov3-33439125efdf>

[21]<https://www.learnopencv.com/deep-learning-based-object-detection-using-yolov3-with-opencv-python-c/>

[22] <https://medium.com/analytics-vidhya/object-detection-with-opencv-python-using-yolov3-481f02c6aa35>

[23]<https://pjreddie.com/darknet/yolo/>

[24] <https://tryolabs.com/blog/2017/08/30/object-detection-an-overview-in-the-age-of-deep-learning/>

[25] <https://www.clarifai.com/blog/classification-vs-detection-vs-segmentation-models-the-differences-between-them-and-how-each-impact-your-results>

[26] <https://medium.com/datadriveninvestor/how-is-ai-enabling-object-identification-and-recognition-263e4166b02b>

[27] https://stats.stackexchange.com/questions/154879/a-list-of-cost-functions-usedin-neural-networks-alongside-applications

[28] <https://www.panopto.com/features/video-search/>