

UNIT 3: Making Machines See

Title: Making Machines See	Approach: Team Discussion, Web search, Hands-on activities
Summary: Computer vision has become a cornerstone technology in today's digital era, enabling machines to "see" and interpret visual data much like humans. This lesson delves into the fascinating world of computer vision, exploring its fundamental principles, key processes, real-world applications, and future potential.	
Learning Objectives: <ol style="list-style-type: none">1. Understand the fundamentals of computer vision and its role in processing and analysing digital images and videos.2. Explore the various stages involved in the computer vision process.3. Gain insight into the applications of computer vision across different industries.4. Identify the challenges and ethical considerations associated with computer vision technology, including privacy concerns, data security, and misinformation.5. Recognize the future potential of computer vision technology and its impact on society.	
Key Concepts: <ol style="list-style-type: none">1. Introduction to Computer Vision2. Working of Computer Vision3. Applications of Computer Vision4. Challenges of Computer Vision5. The Future of Computer Vision	
Learning Outcomes: Students will be able to - <ol style="list-style-type: none">1. Explain the concept of computer vision and its significance in analysing visual data.2. Demonstrate an understanding of the key stages involved in computer vision process and their respective roles in interpreting images and videos.3. Identify real-world applications of computer vision technology in various industries and understand how it enhances efficiency and productivity.4. Evaluate the ethical implications and challenges associated with computer vision, including privacy concerns and the spread of misinformation.5. Envision the future possibilities of computer vision technology.	
Prerequisites: Basic understanding of digital imaging concepts, and knowledge of machine learning.	

With the rapid expansion of social media platforms such as Facebook, Instagram, and Twitter, smartphones have emerged as pivotal tools, thanks to their integrated cameras facilitating effortless sharing of photos and videos. While the Internet predominantly consists of text-based content, indexing and searching images present a distinct challenge. Indexing and searching images involve organizing image data for quick retrieval based on specific features like colour, texture, shape, or metadata. During indexing, key attributes are extracted and stored in a searchable format. Searching uses this index to match query parameters with stored image features, enabling efficient retrieval. Unlike text, which can be easily processed, algorithms require additional capabilities to interpret image content.

Traditionally, the information conveyed by images and videos has relied heavily on manually provided meta descriptions. To overcome this limitation, there is a growing need for computer systems to visually perceive and comprehend images to extract meaningful information from them. This involves enabling computers to "see" images and decipher their content, thereby bridging the gap in understanding and indexing visual data. This poses a simple challenge for humans, evident in the common practice of teaching children to associate an image, such as an apple, with the letter 'A'. 

Humans can easily make this connection. However, enabling computers to comprehend images presents a different dilemma. Similarly to how children learn by repeatedly viewing images to memorize objects or people, we need computers to develop similar capabilities to effectively analyse our images and videos.

3.1. HOW MACHINES SEE?

Computer Vision, commonly referred to as CV, enables systems to see, observe, and understand. Computer Vision is similar to human vision as it trains machines with cameras, data, and algorithms similar to retinas, optic nerves, and a visual cortex as in human vision. CV derives meaningful information from digital images, videos and other visual input and makes recommendations or takes actions accordingly.

Computer Vision systems are trained to inspect products, watch infrastructure, or a production asset to analyse thousands of products or processes in real-time, noticing defects or issues. Due to its speed, objectivity, continuity, accuracy, and scalability, it can quickly surpass human capabilities. The latest deep learning models achieve above human-level accuracy and performance in real-world image recognition tasks such as facial recognition, object detection, and image classification.

Computer Vision is a field of artificial intelligence (AI) that uses Sensing devices and deep learning models to help systems understand and interpret the visual world.

Computer Vision is sometimes called Machine Vision.

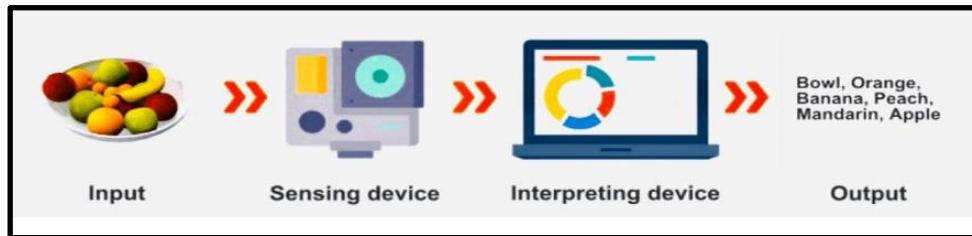


Fig.3.1: Process flow of computer vision

image source: <https://www.ciopages.com/wp-content/uploads/2020/07/vision-work.jpg>

3.2. WORKING OF COMPUTER VISION

At its core, computer vision is the field of study that focuses on processing and analysing digital images and videos to comprehend their content. A fundamental aspect of computer vision lies in understanding the basics of digital images.

3.2.1. Basics of digital images

A digital image is a picture that is stored on a computer in the form of a sequence of numbers that computers can understand. Digital images can be created in several ways like using design software (like Paint or Photoshop), taking one on a digital camera, or scan one using a scanner.

3.2.2. Interpretation of Image in digital form



Fig.3.2: How pixel affects the image

When a computer processes an image, it perceives it as a collection of tiny squares known as *pixels*. Each pixel, short for "picture element," represents a specific color value. These pixels collectively form the digital image. During the process of digitization, an image is converted into a grid of pixels. The resolution of the image is determined by the number of pixels it contains; the higher the resolution, the more detailed the image appears and the closer it resembles the original scene.

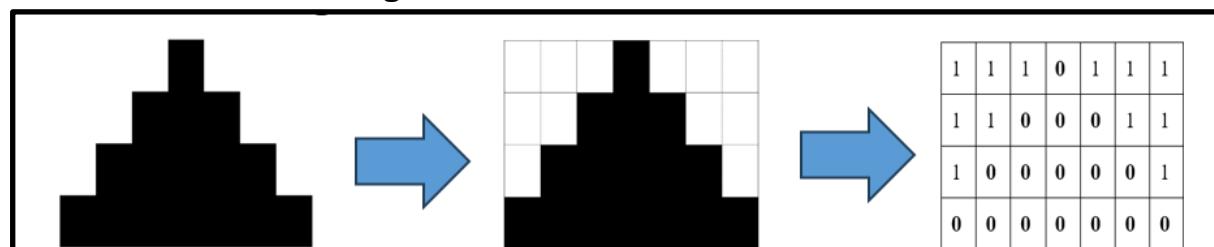


Fig. 3.3: Representing an image using 0's and 1's

In representing images digitally, each pixel is assigned a numerical value. For monochrome images, such as black and white photographs, a pixel's value typically ranges from 0 to 255. A value of 0 corresponds to black, while 255 represents white.

ACTIVITY 3.1 - Binary Art: Recreating Images with 0s and 1s

Step 1: Choose an Image

- Select any image to work with.
- You can find free images on open-source websites like: Pixabay, Unsplash, pexels, etc.

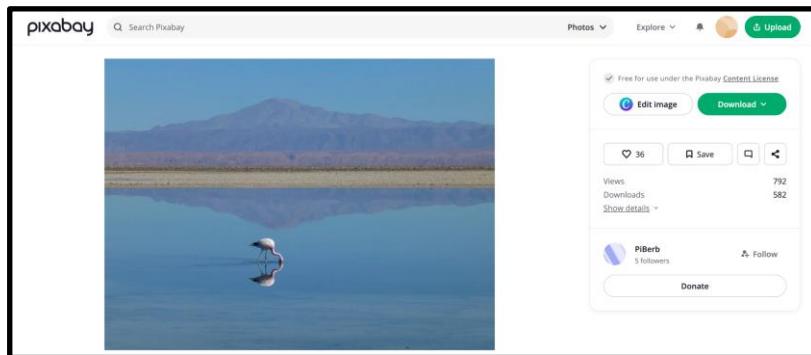


Fig. 3.4: choose an image

Step 2: Resize the Image

- To simplify the activity, resize the image to smaller dimensions (recommended size: width and height between 200 to 300 pixels).
- Use any online resizing tool, such as: Image resizer - <https://imageresizer.com/>
- Ensure the resized image is saved to your computer.

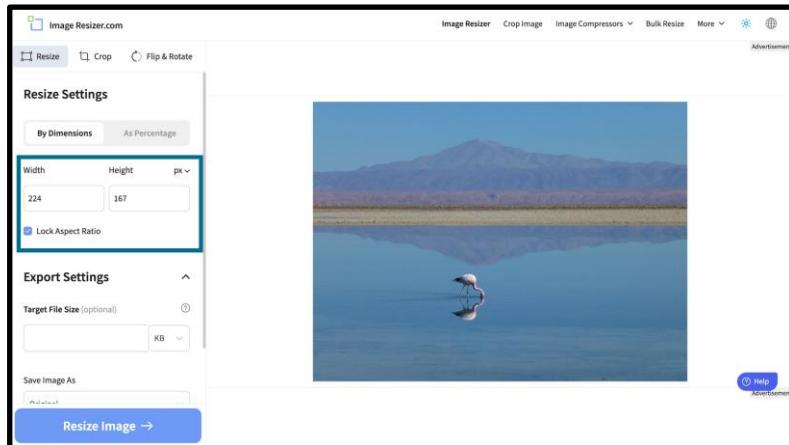


Fig. 3.5: resize of image

Step 3: Convert to Grayscale

- Transform the image into grayscale so it contains only shades of gray (1 channel).
- Use an online grayscale converter, such as Pine tools- <https://pinetools.com/grayscale-image>
- Upload your resized image, convert it to grayscale, and download the resulting image.

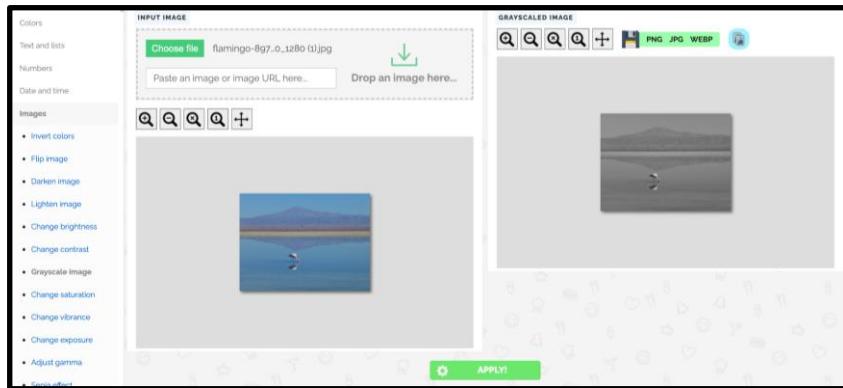


Fig. 3.6: grayscale conversion

Step 4: Extract Pixel Values

- The grayscale image needs to be converted into numerical pixel values (e.g., 0 and 1 for black and white tones).
- Use a pixel value extractor tool, such as: Boxentriq Pixel Value Extractor - <https://www.boxentriq.com/code-breaking/pixel-values-extractor>
- Upload your grayscale image and press the “Extract” button to generate pixel values.

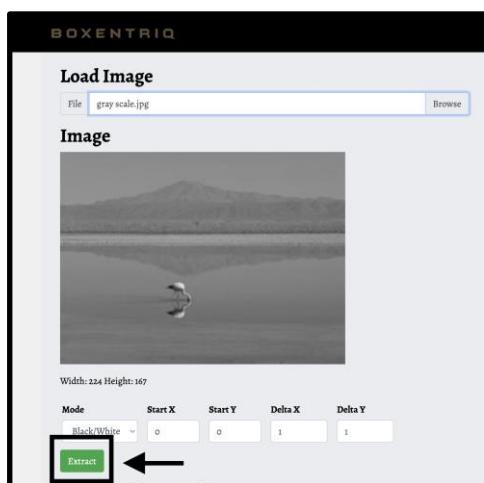


Fig. 3.7

Step 5: Copy the Pixel Values

- Once the pixel values are extracted, select all the values from the tool and copy them.

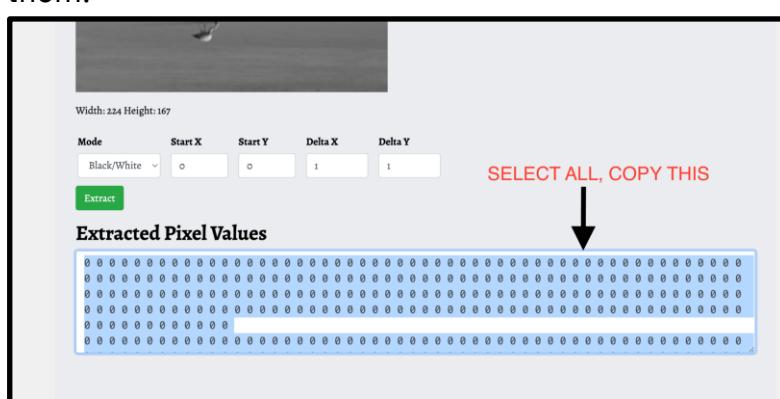


Fig. 3.8: copy this pixel value

Step 6: Paste into a Word Document

- Open a Word document (Google Docs or Microsoft Word).
- Paste the copied pixel values into the document.

Step 7: Adjust the Font Size

- Select all the pasted pixel values in the document.
- Change the font size to **1** for better visualization.
- **Observe the image formation as 0s and 1s recreate the original grayscale image.**

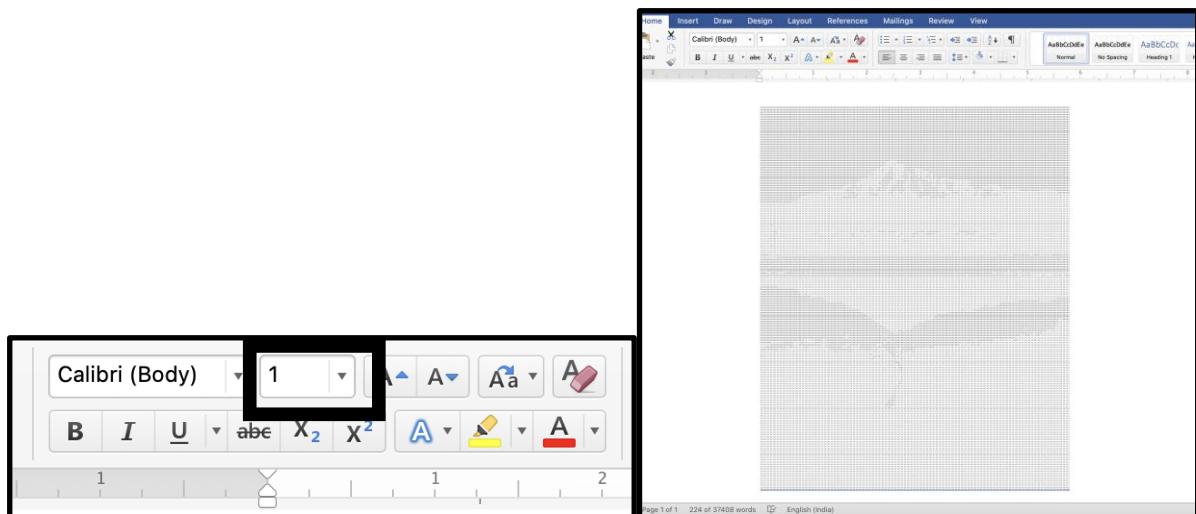


Fig. 3.9: image formation as 0s and 1s recreate the original grayscale image

In coloured images, each pixel is assigned a specific number based on the RGB colour model, which stands for Red, Green, and Blue.

1 byte= 8 bits so the total number of binary numbers formed will be $2^8=256$.

$$\boxed{0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0} = 2^7X0 + 2^6X0 + 2^5X0 + 2^4X0 + 2^3X0 + 2^2X0 + 2^1X0 + 2^0X0 = 0$$

$$\boxed{1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1} = 2^7X1 + 2^6X1 + 2^5X1 + 2^4X1 + 2^3X1 + 2^2X1 + 2^1X1 + 2^0X1 = 255$$

By combining different intensities of red, green, and blue, a wide range of colours can be represented in an image, each colour channel can have a value from 0 to 255, resulting in over 16 million possible colours.

3.3. COMPUTER VISION – PROCESS:

The Computer Vision process often involves five stages. They are explained below.

3.3.1. Image Acquisition: *Image acquisition is the initial stage in the process of computer vision, involving the capture of digital images or videos.* This step is crucial as it provides the raw data upon which subsequent analysis is based. Digital images can be acquired through various means, including capturing them with digital cameras, scanning physical photographs or documents, or even generating them using design software.

The quality and characteristics of the acquired images greatly influence the effectiveness of subsequent processing and analysis. It is important to understand that the capabilities and resolutions of different imaging devices play a significant role in determining the quality of acquired images. Higher-resolution devices can capture finer details and produce clearer images compared to those with lower resolutions. Moreover,

various factors such as lighting conditions and angles can influence the effectiveness of image acquisition techniques. For instance, capturing images in low-light conditions may result in poorer image quality, while adjusting the angle of capture can provide different perspectives of the scene.

In scientific and medical fields, specialized imaging techniques like MRI (Magnetic Resonance Imaging) or CT (Computed Tomography) scans are employed to acquire highly detailed images of biological tissues or structures. These advanced imaging modalities offer insights into the internal composition and functioning of biological entities, aiding in diagnosis, research, and treatment planning.

3.3.2. Preprocessing:

Preprocessing in computer vision aims to enhance the quality of the acquired image. Some of the common techniques are-

- Noise Reduction:** Removes unwanted elements like blurriness, random spots, or distortions. This makes the image clearer and reduces distractions for algorithms.

Example: Removing grainy effects in low-light photos.

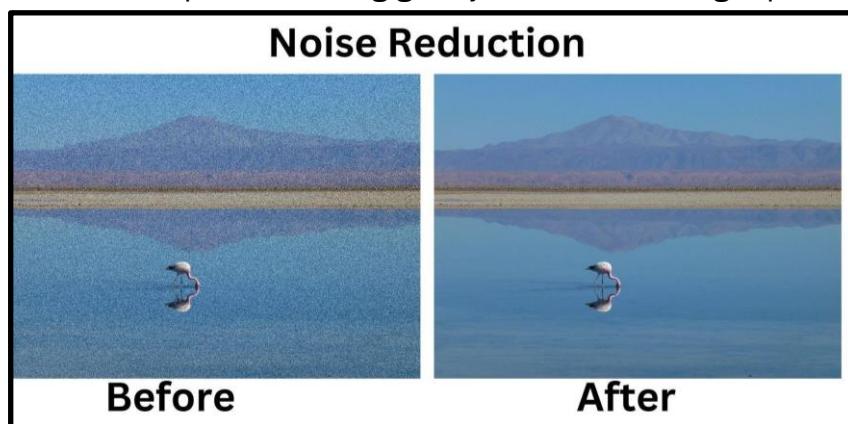


Fig. 3.10: before image is noise image, after image is noise reduced

- Image Normalization:** Standardizes pixel values across images for consistency. Adjusts the pixel values of an image so they fall within a consistent range (e.g., 0–1 or -1 to 1). Ensures all images in a dataset have a similar scale, helping the model learn better. Example: Scaling down pixel values from 0–255 to 0–1.

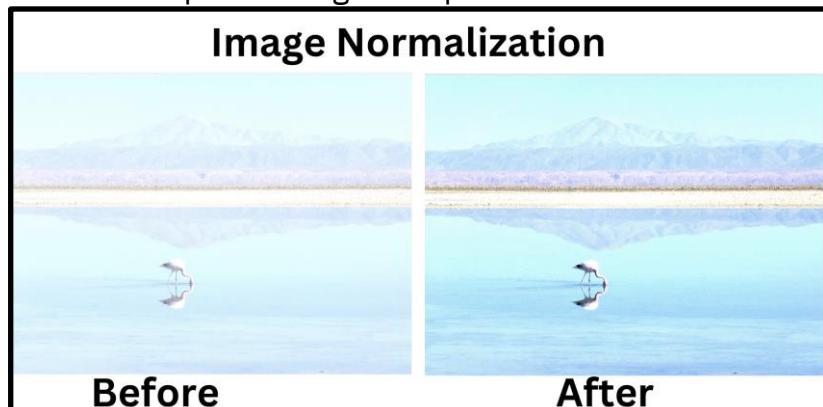


Fig. 3.11: before is distorted with no normalization

- Resizing/Cropping:** Changes the size or aspect ratio of the image to make it uniform. Ensures all images have the same dimensions for analysis. Example: Resizing all images to 224×224 pixels before feeding them into a neural network.

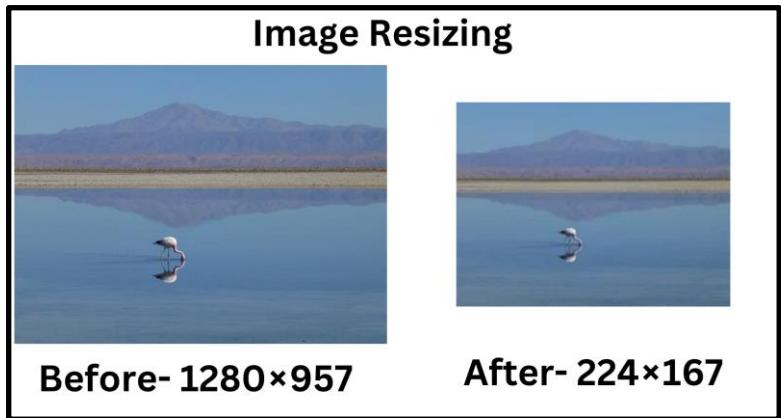


Fig. 3.12: Resizing of image

- d. **Histogram Equalization:** Adjusts the brightness and contrast of an image. Spreads out the pixel intensity values evenly, enhancing details in dark or bright areas. Example: Making a low-contrast image look sharper and more detailed.

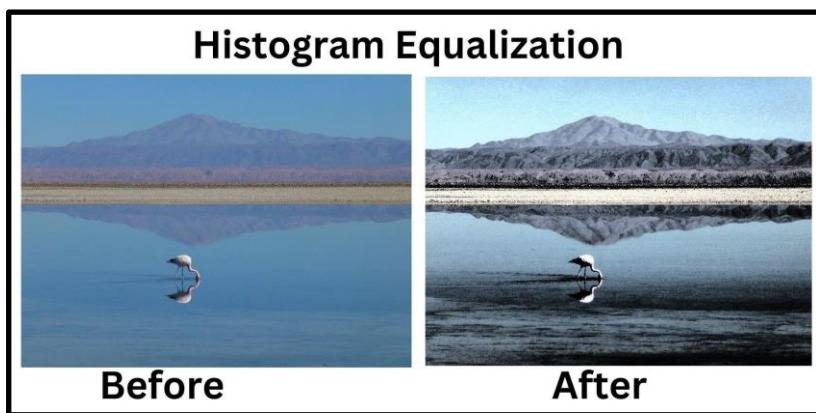


Fig. 3.13

The main goal for preprocessing is to prepare images for computer vision tasks by:

- Removing noise (disturbances).
- Highlighting important features.
- Ensuring consistency and uniformity across the dataset.

3.3.3. Feature Extraction:

Feature extraction involves identifying and extracting relevant visual patterns or attributes from the pre-processed image. Feature extraction algorithms vary depending on the specific application and the types of features relevant to the task. The choice of feature extraction method depends on factors such as the complexity of the image, the computational resources available, and the specific requirements of the application.

- **Edge detection** identifies the boundaries between different regions in an image where there is a significant change in intensity
- **Corner detection** identifies points where two or more edges meet. These points are areas of high curvature in an image, focused on identifying sharp changes in image gradients, which often correspond to corners or junctions in objects.
- **Texture analysis** extracts features like smoothness, roughness, or repetition in an image

- **Colour-based feature extraction** quantifies colour distributions within the image, enabling discrimination between different objects or regions based on their colour characteristics.

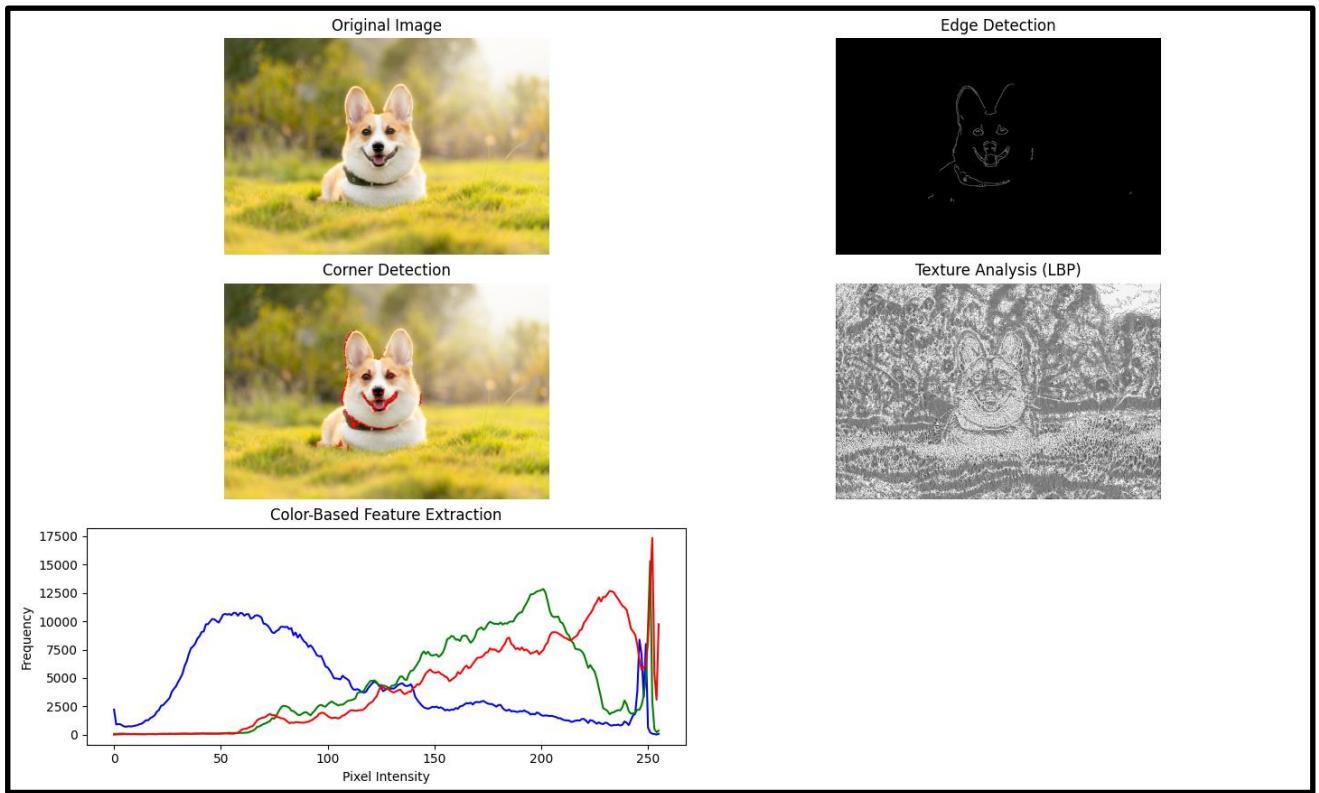


Fig. 3.14-Edge detection, corner detection, texture analysis, color-based feature extraction

In deep learning-based approaches, feature extraction is often performed automatically by convolutional neural networks (CNNs) during the training process.

3.3.4. Detection/Segmentation:

Detection and segmentation are fundamental tasks in computer vision, focusing on identifying objects or regions of interest within an image. These tasks play a pivotal role in applications like autonomous driving, medical imaging, and object tracking. This crucial stage is categorized into two primary tasks:

1. Single Object Tasks
2. Multiple Object Tasks

Single Object Tasks: Single object tasks focus on analysing/or delineate individual objects within an image, with two main objectives:

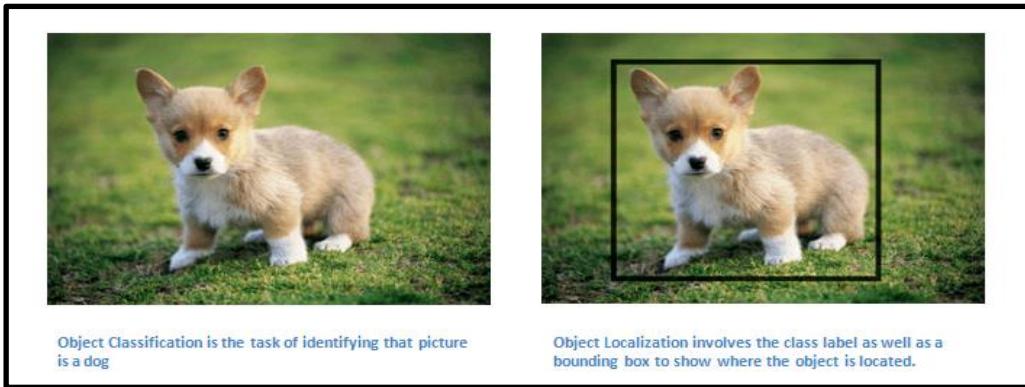


Fig.3.15: classification, classification+localization

- i) **Classification:** This task involves determining the category or class to which a single object belongs, providing insights into its identity or nature. KNN(K-Nearest Neighbour)algorithm may be used for supervised classification while K-means clustering algorithm can be used for unsupervised classification.
- ii) **Classification + Localization:** In addition to classifying objects, this task also involves precisely localizing the object within the image by predicting bounding boxes that tightly enclose it.

Multiple Object Tasks: Multiple object tasks deal with scenarios where an image contains multiple instances of objects or different object classes. These tasks aim to identify and distinguish between various objects within the image, and they include:

- i) **Object Detection:** Object detection focuses on identifying and locating multiple objects of interest within the image. It involves analysing the entire image and drawing bounding boxes around detected objects, along with assigning class labels to these boxes. The main difference between classification and detection is that classification considers the image as a whole and determines its class whereas detection identifies the different objects in the image and classifies all of them.

In detection, bounding boxes are drawn around multiple objects and these are labelled according to their particular class. Object detection algorithms typically use extracted features and learning algorithms to recognize instances of an object category. Some of the algorithms used for object detection are: R-CNN (Region-Based Convolutional Neural Network), R-FCN (Region-based Fully Convolutional Network), YOLO (You Only Look Once) and SSD (Single Shot Detector).

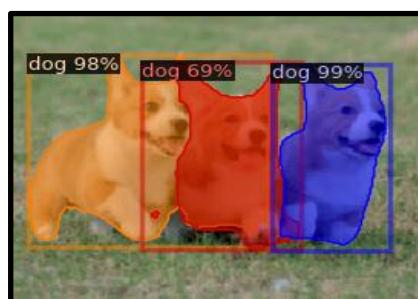


Fig.3.16- object detection

- ii) **Image segmentation:** It creates a mask around similar characteristic pixels and identifies their class in the given input image. Image segmentation helps to gain a better understanding of the image at a granular level. Pixels are assigned a class and for each object, a pixel-wise mask is created in the image. This helps to easily identify each object separately from the other. Techniques like Edge detection which works by detecting discontinuities in brightness is used in Image segmentation. There are different types of Image Segmentation available.

Two of the popular segmentation are:

- a. **Semantic Segmentation:** It classifies pixels belonging to a particular class. Objects belonging to the same class are not differentiated. In this image for example the pixels are identified under class animals but do not identify the type of animal.
- b. **Instance Segmentation:** It classifies pixels belonging to a particular instance. All the objects in the image are differentiated even if they belong to the same class. In this image for example the pixels are separately masked even though they belong to the same class.

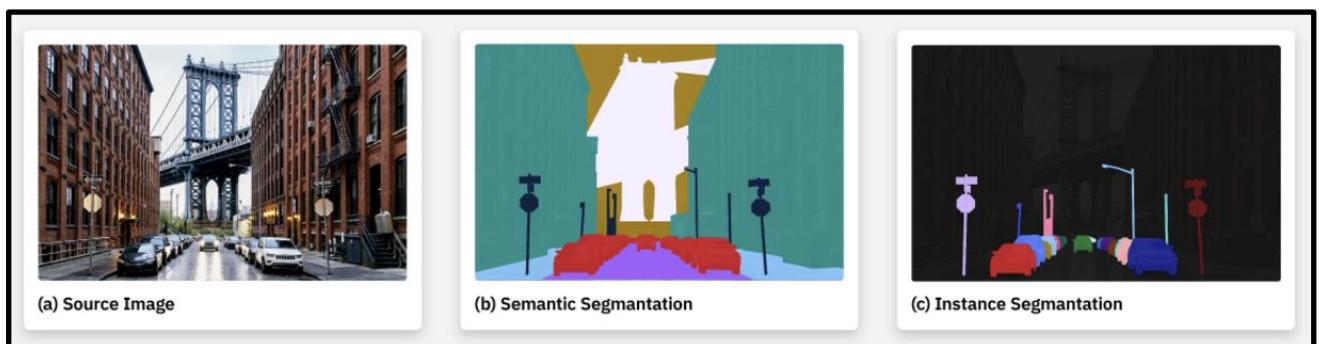
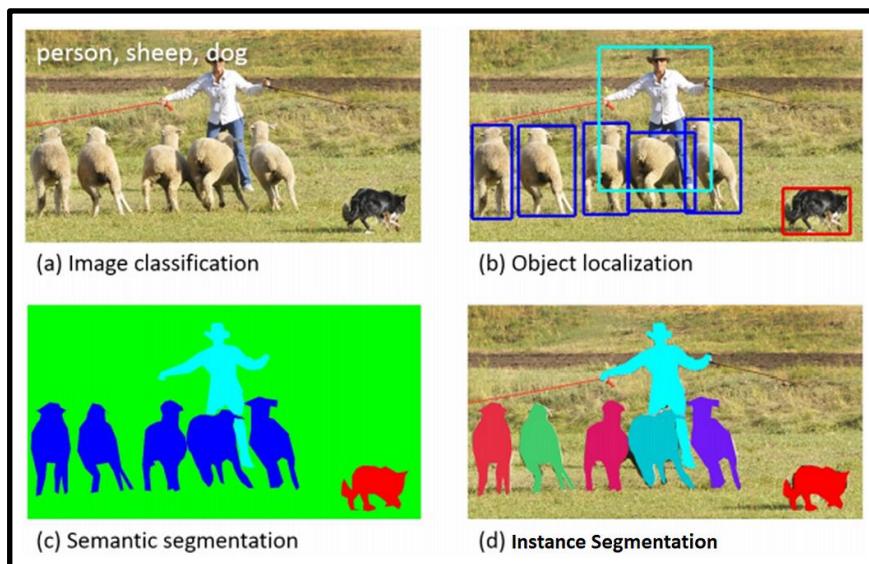


Fig.3.17

3.3.5. High-Level Processing: In the final stage of computer vision, high-level processing plays a crucial role in interpreting and extracting meaningful information from the detected objects or regions within digital images. This advanced processing enables computers to achieve a deeper understanding of visual content and make informed decisions based on the visual data. Tasks involved in high-level processing include recognizing objects, understanding scenes, and analysing the context of the visual content. Through sophisticated algorithms and machine learning techniques, computers can identify and categorize objects, infer relationships between elements in a scene, and derive insights from complex visual data. Ultimately, high-level processing empowers computer vision systems to extract valuable insights and drive intelligent decision-making in various applications, ranging from autonomous driving to medical diagnostics.

3.4. APPLICATIONS OF COMPUTER VISION

Computer vision is one of the areas in Machine Learning whose principle is already integrated into major products that we use every day. Some of the applications are listed below which you might have already learned in lower classes.

- **Facial recognition:** Popular social media platforms like Facebook uses facial recognition to detect and tag users.
- **Healthcare:** Helps in evaluating cancerous tumours, identifying diseases or abnormalities. Object detection & tracking in medical imaging.
- **Self-driving vehicles:** Makes sense of the surroundings by capturing video from different angles around the car. Detect other cars and objects, read traffic signals, pedestrian paths, etc.
- **Optical character recognition (OCR):** Extract printed or handwritten text from visual data such as images or documents like invoices, bills, articles, etc.
- **Machine inspection:** Detects a machine's defects, features, and functional flaws, determines inspection goals, chooses lighting and material-handling techniques, and other irregularities in manufactured products.
- **3D model building:** Constructing 3D computer models from existing objects which has a variety of applications in various places, such as Robotics, Autonomous driving, 3D tracking, 3D scene reconstruction, and AR/VR.
- **Surveillance:** Live footage from CCTV cameras in public places helps to identify suspicious behaviour, identify dangerous objects, and prevent crimes by maintaining law and order.
- **Fingerprint recognition and biometrics:** Detects fingerprints and biometrics to validate a user's identity.

3.5. CHALLENGES OF COMPUTER VISION

Computer vision, a vital part of artificial intelligence, faces several hurdles as it strives to make sense of the visual world around us. These challenges include:

1. **Reasoning and Analytical Issues:** Computer vision relies on more than just image identification; it requires accurate interpretation. Robust reasoning and analytical skills are essential for defining attributes within visual content. Without such capabilities, extracting meaningful insights from images becomes challenging, limiting the effectiveness of computer vision systems.
2. **Difficulty in Image Acquisition:** Image acquisition in computer vision is hindered by various factors like lighting variations, perspectives, and scales. Understanding complex scenes with multiple objects and handling occlusions adds to the complexity. Obtaining high-quality image data amidst these challenges is crucial for accurate analysis and interpretation.
3. **Privacy and Security Concerns:** Vision-powered surveillance systems raise serious privacy concerns, potentially infringing upon individuals' privacy rights. Technologies like facial recognition and detection prompt ethical dilemmas regarding privacy and security. Regulatory scrutiny and public debate surround the use of such technologies, necessitating careful consideration of privacy implications.
4. **Duplicate and False Content:** Computer vision introduces challenges related to the proliferation of duplicate and false content. Malicious actors can exploit vulnerabilities in image and video processing algorithms to create misleading or fraudulent content. Data breaches pose a significant threat, leading to the dissemination of duplicate images and videos, fostering misinformation and reputational damage.

3.6. THE FUTURE OF COMPUTER VISION

Over the years, computer vision has evolved from basic image processing tasks to complex systems capable of understanding and interpreting visual data with human-like precision. Breakthroughs in deep learning algorithms, coupled with the availability of vast amounts of labelled training data, have propelled the field forward, enabling machines to perceive and analyse images and videos in ways previously thought impossible.

As we look to the future, the possibilities by computer vision are awe-inspiring. From personalized healthcare diagnostics to immersive AR experiences, the impact of computer vision on society is set to be profound and far-reaching. By embracing innovation, fostering collaboration, and prioritizing ethical considerations, we can unlock the full potential of computer vision and harness its transformative power for the benefit of humanity.

ACTIVITY 3.2 CREATING A WEBSITE CONTAINING AN ML MODEL

1. Go to the website <https://teachablemachine.withgoogle.com/>

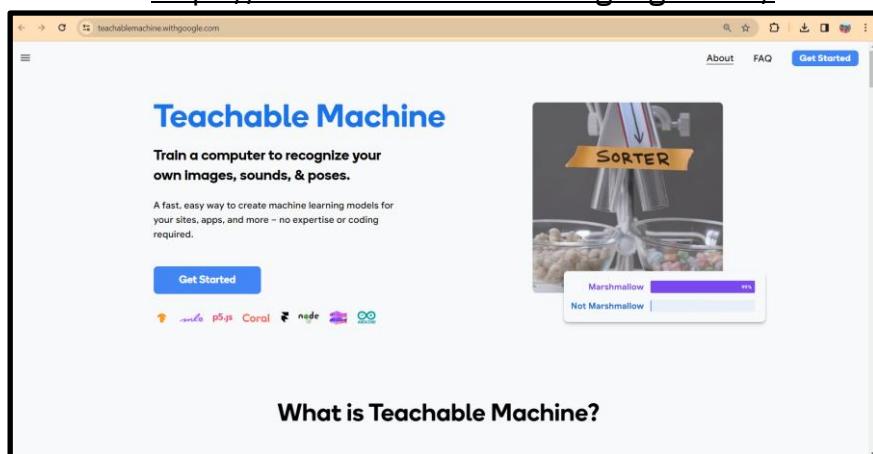


Fig.3.18

2. Click on Get Started.
3. Teachable machine offers 3 options as you can see.

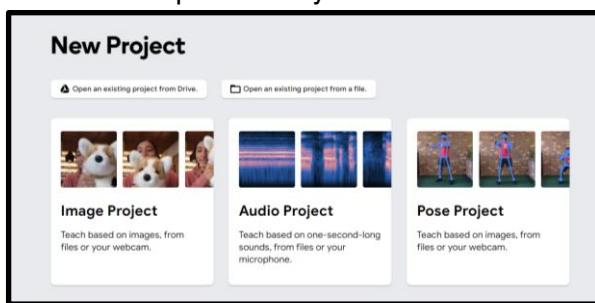


Fig.3.19

4. Choose the 'Image' project.
5. Select 'Standard Image Model'.

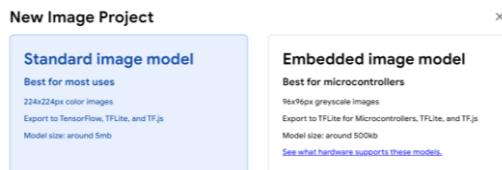


Fig.3.20

6. You will get a screen like this.

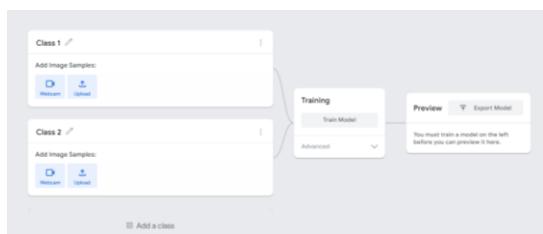


Fig.3.21

You have the option to choose between two methods: using your **webcam** to capture images or **uploading** existing images. For the webcam option, you will need to position the image in front of the camera and hold down the record button to capture the image. Alternatively, with the upload option, you have the choice to upload images either from your local computer or directly from Google Drive.

7. Let us name ‘Class 1’ as Kittens and upload pictures already saved on the computer.

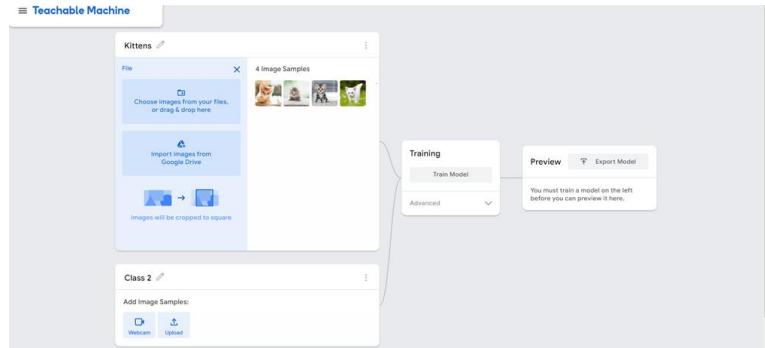


Fig.3.22

8. Now, let us add another class of images “Puppies” saved on the computer.

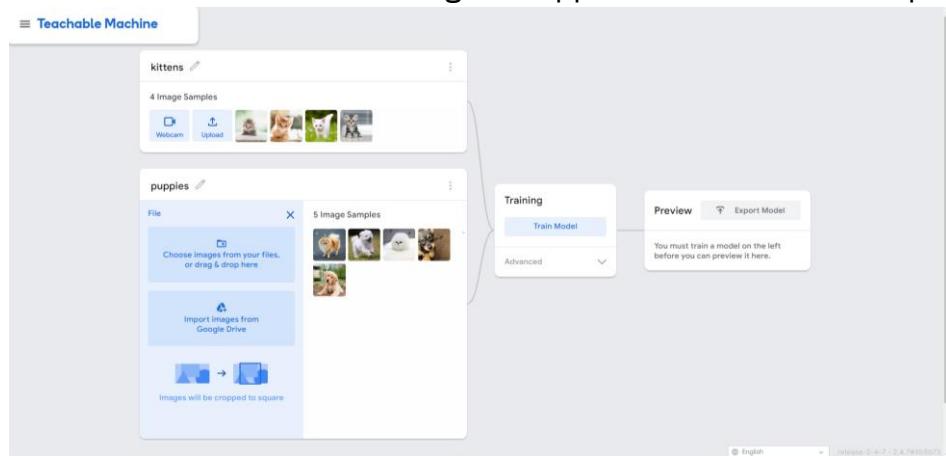


Fig.3.23

9. Click on Train Model.

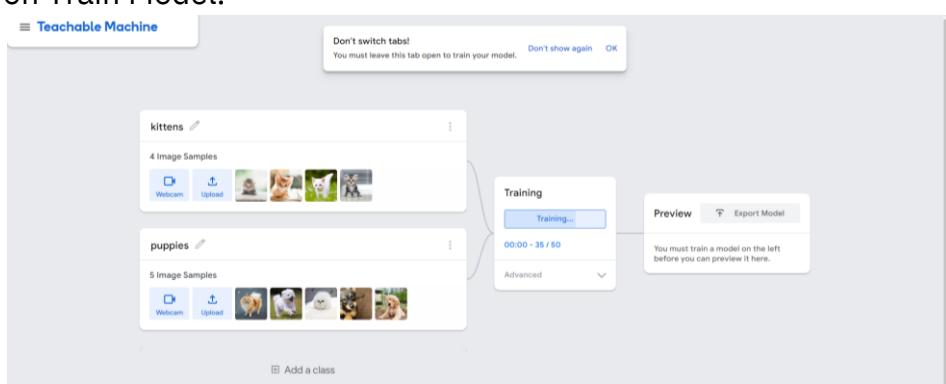


Fig.3.24

Once the model is trained, you can test the working by showing an image in front of the web camera. Else, you can also upload the image from your local computer / Google drive.

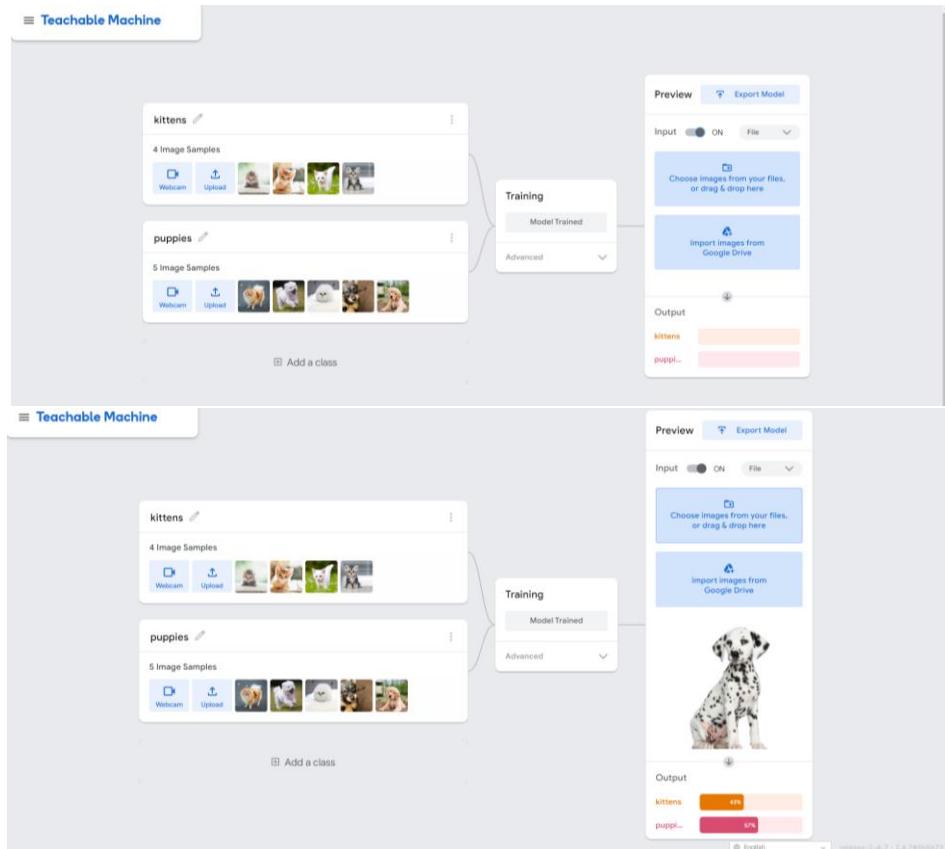


Fig.3.25

10. Now click on ‘Export Model’. A screen will open up as shown. Now, click on Upload my model.

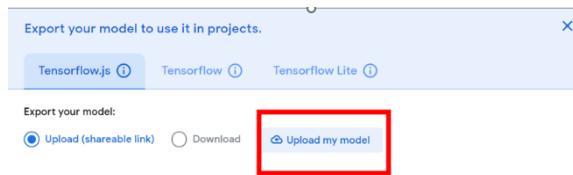


Fig.3.26

11. Once your model is uploaded, Teachable Machine will create a URL which we will use in the Javascript code. Copy the Javascript code by clicking on Copy.

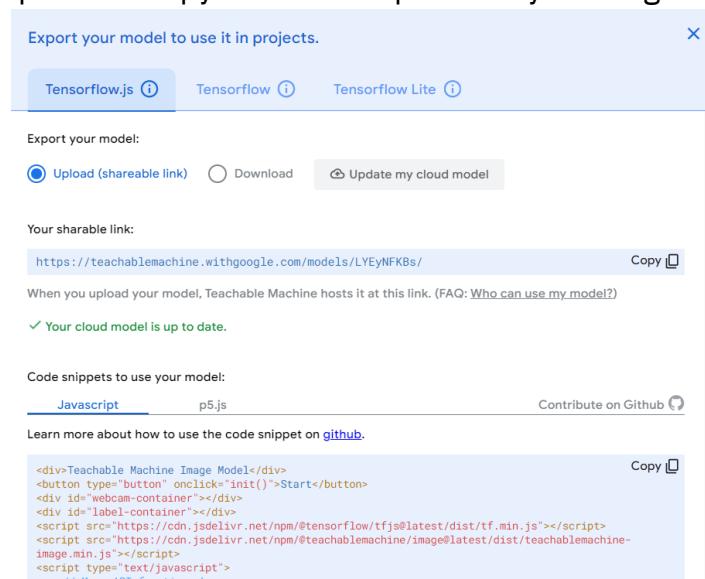


Fig.3.27

12. Open Notepad and paste the JavaScript code and save this file as web.html.
13. Let us now deploy this model in a website.
14. Once you create a free account on Weebly, go to Edit website and create an appealing website using the tools given.



Fig.3.28

15. Click on Embed Code and drag and place it on the webpage.

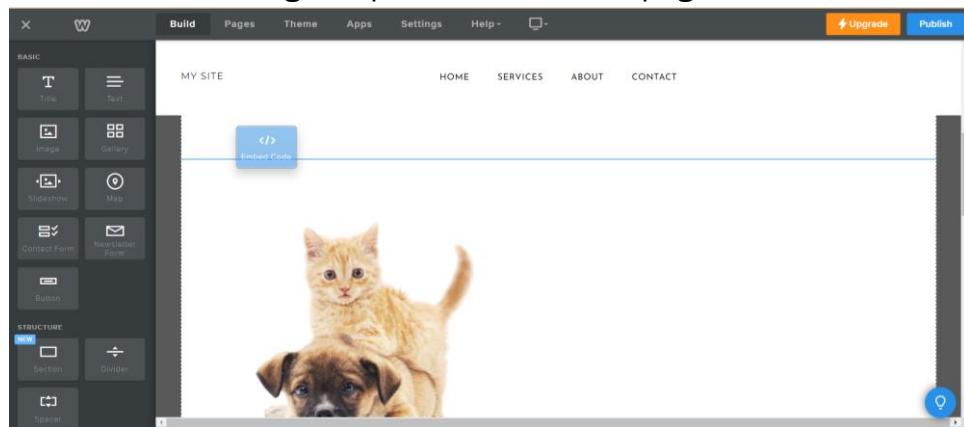


Fig.3.29

16. Click on 'Edit HTML Code' and copy the Javascript code from the html file (web.html) and paste it here as shown.

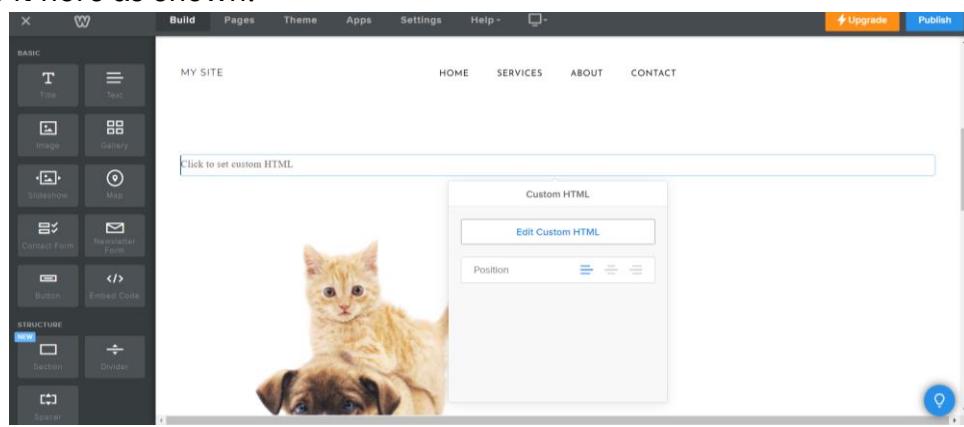


Fig.3.30

17. Click 'Publish' (keep on Weebly subdomain for free version).

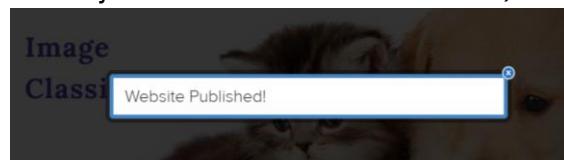


Fig.3.31

18.Copy the URL and paste it into a new browser window to check the working of your model.



Fig.3.32

19.Click on start and you can show pictures of kitten and puppy to check the predictions of your model.

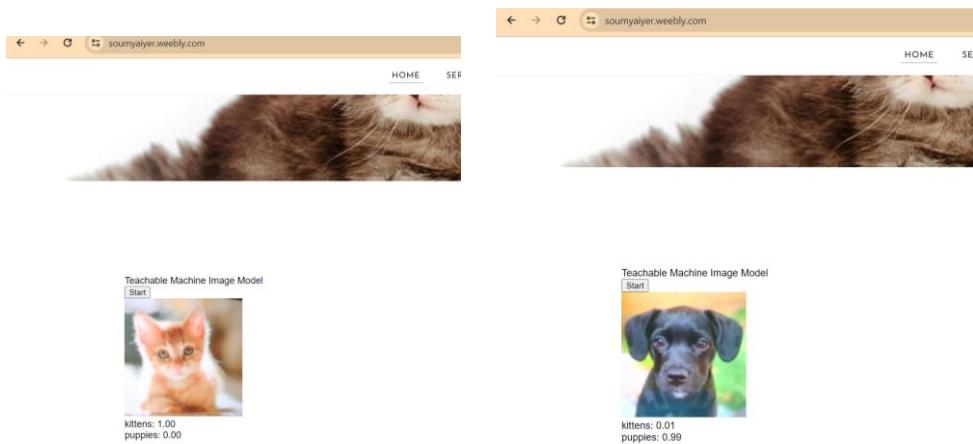


Fig.3.33

3.7. Working with OpenCV: (**For Advanced Learners)

3.7.1. Introduction to OpenCV: OpenCV or Open-Source Computer Vision Library is a cross-platform library using which we can develop real-time computer vision applications. It mainly focuses on image processing, video capture, and analysis including features like face detection and object detection. It is also capable of identifying objects, faces, and even handwriting.

To use OpenCV in Python, you need to install the library. Use the following command in your terminal or command prompt:

pip install opencv-python

3.7.2. Loading and Displaying an Image: Let us understand the loading and displaying using a scenario followed by a question.

Scenario- You are working on a computer vision project where you need to load and display an image. You decide to use OpenCV for this purpose.

Question:

What are the necessary steps to load and display an image using OpenCV? Write a Python code snippet to demonstrate this.

sol -

Here's a simple Python script to load and display an image using OpenCV:

```
import cv2  
  
image = cv2.imread('example.jpg') # Replace 'example.jpg' with the path to  
your image  
cv2.imshow('original image', image)  
cv2.waitKey(0)  
cv2.destroyAllWindows()
```



Fig.3.34: loading and displaying image

- `cv2.imread('example.jpg')` loads the image into a variable. Replace `example.jpg` with your image's file name or path.
- `cv2.imshow()` opens a new window to display the image.
- `cv2.waitKey(0)` waits indefinitely for a key press to proceed
- `cv2.destroyAllWindows()` closes any OpenCV windows.

3.7.3. Resizing an image:

Scenario- Imagine we are working on an application where we need to process images of varying sizes. To standardize the input, all images must be resized to 300x300 pixels before further processing. We are using OpenCV to achieve this. Additionally, we want to ensure the aspect ratio is maintained in some cases.

Question:

1. Write a Python script using OpenCV to resize an image named “example.jpg” to a fixed size of 300x300 pixels.

Sol -

Here we will use the “`cv2.resize()`” function to resize the image. We can directly specify the width and height of the new image

```

new_width = 300
new_height = 300
# Resize the image to the new dimensions
resized_image = cv2.resize(image, (new_width, new_height))

```

So, the full code will look like this -

```

import cv2
image = cv2.imread('example.jpg') # Replace 'example.jpg' with the path
to your image
new_width = 300
new_height = 300
# Resize the image to the new dimensions
resized_image = cv2.resize(image, (new_width, new_height))
cv2.imshow('Resized Image', resized_image)
cv2.waitKey(0)
cv2.destroyAllWindows()

```

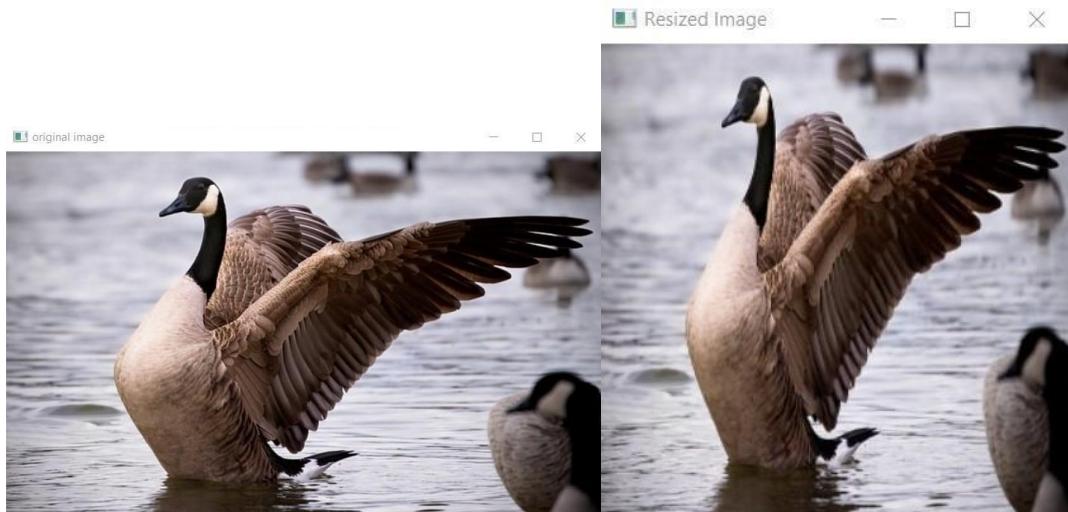


Fig.3.35: Resizing an Image

3.7.4. Converting an Image to Grayscale

Scenario:

Imagine we are working on an application where color images need to be converted to grayscale before further processing, as grayscale images reduce computational complexity. We are using OpenCV to achieve this.

Question: Write a Python script using OpenCV to convert an image named “example.jpg” into a grayscale.

Solution:

Here we will use the `cv2.cvtColor()` function to convert the image into grayscale. The function requires the source image and a color conversion code as inputs. The code for grayscale conversion is `cv2.COLOR_BGR2GRAY`.

```
grayscale_image = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
```

So, the full code will look like this:

```
import cv2  
image = cv2.imread('example.jpg') # Replace 'example.jpg' with the path to  
your image  
grayscale_image = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)  
cv2.imshow('Grayscale Image', grayscale_image)  
cv2.waitKey(0)  
cv2.destroyAllWindows()
```

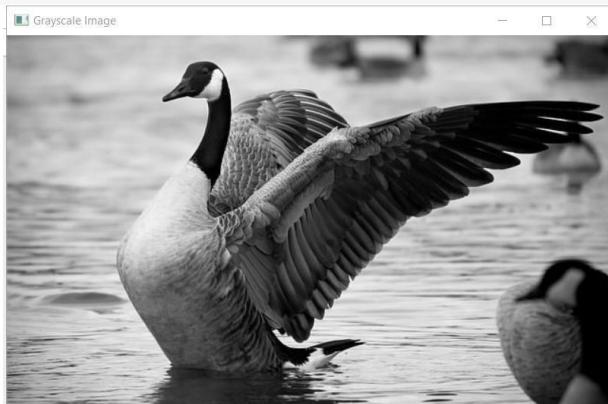


Fig.3.36: converting an image to grayscale

EXERCISES

A. Multiple Choice Questions:

1. The field of study that helps to develop techniques to help computers “see” is _____.
a. Python b. Convolution c. Computer Vision d. Data Analysis
2. Task of taking an input image and outputting/assigning a class label that best describes the image is _____.
b. Image classification b. Image localization
c. Image Identification d. Image prioritization
3. Identify the incorrect option
(i) computer vision involves processing and analysing digital images and videos to understand their content.
(ii) A digital image is a picture that is stored on a computer in the form of a sequence of numbers that computers can understand.
(iii) RGB colour code is used only for images taken using cameras.
(iv) Image is converted into a set of pixels and less pixels will resemble the original image.
a. ii b. iii c. iii & iv d. ii & iv
4. The process of capturing a digital image or video using a digital camera, a scanner, or other imaging devices is related to _____.
a. Image Acquisition b. Preprocessing
c. Feature Extraction d. Detection
5. Which algorithm may be used for supervised learning in computer vision?
a. KNN b. K-means c. K-fold d. KEAM
6. A computer sees an image as a series of _____
a. colours b. pixels c. objects d. all of the above

7. _____ empowers computer vision systems to extract valuable insights and drive intelligent decision-making in various applications, ranging from autonomous driving to medical diagnostics.

- a. Low level processing
- b. High insights
- c. High-level processing
- d. None of the above

8. In Feature Extraction, which technique identifies abrupt changes in pixel intensity and highlights object boundaries?

- a. Edge detection
- b. Corner detection
- c. Texture Analysis
- d. boundary detection

9. Choose the incorrect statement related to preprocessing stage of computer vision

- a. It enhances the quality of acquired image
- b. Noise reduction and Image normalization is often employed with images
- c. Techniques like histogram equalization can be applied to adjust the distribution of pixel intensities
- d. Edge detection and corner detection are ensured in images.

10. 1 byte = _____ bits

- a. 10
- b. 8
- c. 2
- d. 1

B. Short Answer Questions

1. What is Computer Vision?
2. What is the main difference between classification and detection?
3. Write down any two algorithms which can be used for object detection.
4. Write down the process of object detection in a single object.
5. Write any four applications of computer vision.

C. Long Answer Questions

1. What do you mean by Image segmentation? Explain the popular segmentations.
2. Explain the challenges faced by computer vision.

COMPETENCY BASED QUESTIONS:

1. A group of students is participating in a photography competition. As part of the competition, they need to submit digitally captured images of various landscapes. However, one of the students, Aryan, is unsure about how to ensure the best quality for his images when digitizing them. Explain Aryan how the resolution of his images can impact their quality and detail when viewed on a computer screen or printed.

2. The Red Fort is hosting a grand cultural event, and keeping everyone safe is top priority! A state-of-the-art security system utilizes different "FEATURE EXTRACTION" to analyse live video feeds and identify potential issues. Identify the feature extraction technique that can be used in the following situation.

- a. A large bag is left unattended near a crowded entrance.
- b. A person tries to climb over a wall near a blind spot.
- c. A group of people starts pushing and shoving in a congested area.
- d. A wanted person with a distinctive red scarf enters the venue.