SMART COW - ASSIGNMENT

1. <u>Problem Statement :-</u> Given a video stream, many UFO's are visible on each frame with a different number ranging from 0 - 9 written on them. The **task** is to **find** the **location** of those **UFO's** on each frame, sort them from top to bottom by their position and return numbers written on them.

2. UNDERSTANDING THE CONTEXT OF THE TASK
   - Our task is characterised as a **Computer vision Task** which in general is about understanding images.
   - Most applications of computer vision today are around images, with less focused on **sequences of RELATED images(i.e. video frames).**
   - While Video allows for deeper **contextual** understanding, (information about action), **in our task the video frames are independent in terms of the <u>context</u> ,and we need to just independently analyse each video frame and locate UFO's Position.**

3. TO SOLVE THE TASK - DETECT OBJECTS USING PYTHON AND OPENCV
   - Object Identification - A video is a sequence of related frames. In our **task** we are studying a **video stream,** to **identify the LOCATION of UFO'S in each video frame, data given is in the form of video frames (raw image data) as generally Videos take up a lot of storage space and are usually not already using AI.**
   - OPEN CV - helps to process images and videos to identify objects, find similar images from an image database etc..
   - We will now detect UFOs in any video file combining OpenCV with Python.
   - Lets first start working with a single video frame(image) :-

– Process the below image to identify UFOs with numbers

– Most popular way to save this is to build a classifier to identify the object type(UFO with NUMBERS). The classifier has lot of definitions/patterns inside. When some object matches with those defined patterns ,then our code will identify and categorises that object.
– HOW TO BUILD A CUSTOM HAAS CASCADE CLASSIFIER? :- we will not use this method as it was prohibited to use background images.

4. CODE TO SOLVE THE TASK - find the location of the UFO's, and crop that part and pass it through a classifier to find the number that UFO contains.

- We will first read the image file using OpenCV and create an instance.
- Now we will do **object localisation** to identify the entity (UFO) by contour detection method and sort the contours from top to bottom.
- Then crop the sorted contours and save them in a new directory "cropped".
- Next comes the **object classification** to recognise the image and classify it into different classes from 0-9 using an svm classifier.