**Lab- 7**

**CSET340- Advanced Computer Vision and Video analytics**

**Task-1:- Blob detection and Image Quality Enhancement.**

**Blob detection**:- Blob detection is a basic method in computer vision used to locate areas of interest in a picture. Apply the **three** different blob detection techniques (LoG, DoG, HoG) **separately** on the same image.

A close-up of a microscope

Description automatically generated

**Image:-** Image from a [light microscope](https://en.wikipedia.org/wiki/Light_microscope) (500 ×) showing platelets (small **purple** dots) surrounded by [red blood cells](https://en.wikipedia.org/wiki/Red_blood_cell) (large **gray** circular structures)

A group of small packages of candy

Description automatically generated

**Image:- Candy image. How many candies are present ? (Optional task)**

A close up of a speckled surface

Description automatically generated

**Images:- Satellite images**

**Image Quality Enhancement:-**



* Perform this using the following techniques
  + Adjusting brightness and contrast
  + Sharpening images
  + Removing noise from images
  + Enhancing color in images
  + Image resizing and scaling
  + Inverse Transform
  + Equalizing histograms
  + Super-resolution
  + Color correction

**Task-2:- Image Classification using AlexNet and VGG16 on Cifar-100**

**Objective:**

* Compare the performance of AlexNet and VGG16 on an image classification task.
* Use **CIFAR-100**, a common dataset for image classification.
* Analyse model accuracy, loss, and inference time on a dataset.

Step 1: Installation of necessary libraries

Step 2: Load the dataset.

Step 3: Load the pretrained Model (AlexNet and VGG16)

Step 4: Train the Models

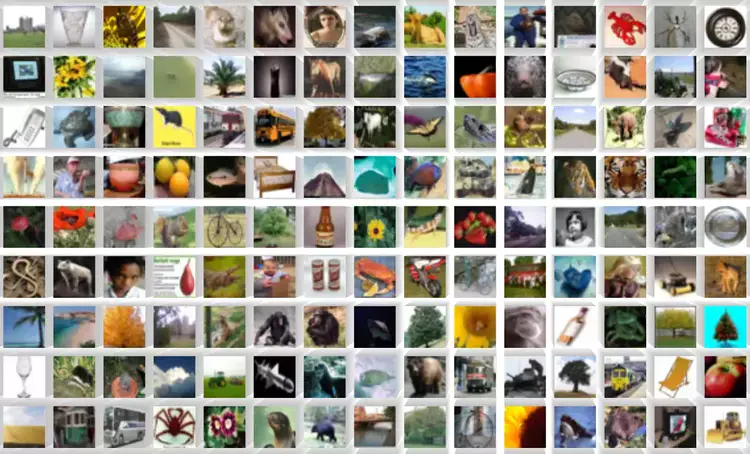
Step 5: Evaluate the Performances

Step 6: Compare the results.

**Datasets-**

1. CIFAR-100 dataset:- The CIFAR100 (Canadian Institute For Advanced Research) dataset consists of 100 classes with 600 color images of 32×32 resolution for each class.

This dataset is just like the CIFAR-10, except it has 100 classes containing 600 images each. There are 500 training images and 100 testing images per class. The 100 classes in the CIFAR-100 are grouped into 20 super classes. Each image comes with a "fine" label (the class to which it belongs) and a "coarse" label (the superclass to which it belongs).



1. Load CIFAR 100 Dataset Training Subset in Python
   1. import deeplake
   2. ds = deeplake.load("hub://activeloop/cifar100-train")
2. Load CIFAR 100 Dataset Testing Subset in Python
   1. import deeplake
   2. ds = deeplake.load("hub://activeloop/cifar100-test")
3. CIFAR 100 Dataset Structure
   1. **CIFAR 100 Data Fields**
      1. images: tensor containing images of the dataset.
      2. labels: tensor containing labels for their respective image.
      3. coarse\_labels: tensor containing superclass for their respective image.

**Alexnet:-**

* AlexNet consists of 5 convolution layers, 3 max-pooling layers, 2 Normalized layers, 2 fully connected layers and 1 SoftMax layer.
* Each convolution layer consists of a convolution filter and a non-linear activation function called “ReLU”.
* The pooling layers are used to perform **the max-pooling** function and the input size is fixed due to the presence of fully connected layers.
* The input size is mentioned at most of the places as 224x224x3 but due to some padding which happens it works out to be 227x227x3.
* AlexNet has over 60 million parameters.

A diagram of a rectangular object

Description automatically generated with medium confidence

**VGG-16:-**

* The 16 in VGG16 refers to 16 layers that have weights.
* In VGG16 there are 13 convolutional layers, 5 Max Pooling layers, and 3 Dense layers which sum up to 21 layers but it has only sixteen weight layers i.e., learnable parameters layer.
* VGG16 takes input tensor size as 224, 244 with 3 RGB channel
* Convolution layers of 3x3 filter with stride 1 and always used the same padding and maxpool layer of 2x2 filter of stride 2.
* Conv-1 Layer has 64 number of filters, Conv-2 has 128 filters, Conv-3 has 256 filters, Conv 4 and Conv 5 has 512 filters.