Computer Vision Tutorial 2

Aurko Routh, Edward Zhang, Simran Vaishya

February 2021

1 Features

In machine learning a feature is a measurable property or characteristic that defines a data point. In the case of computer vision, the lowest level features are the individual pixels that make up the image. On a higher level we can define a feature as a particular type of edge, or arc. On an even higher level, we can combine those edges and arcs into shapes. Finally, we can define a feature in an image to be an object contained within in image. The goal in computer vision is to train a model that can put together lower level features to form higher level features which will allow us to make predictions about an image. In the next section, we will be talking about the fundamental concept of edges in computer vision.

2 Edges

2.1 What is Edge detection?

The early stages of computer image processing identify features that are relevant to estimating both the structure and properties of images. One of the main methods to do this is through edge detection, which is an image processing technique which is commonly used in computer vision to find boundaries of objects within images. It does this by detecting certain discontinuities in the image revealing different levels of brightness, depth, and surface orientation. Edge detection can be applied to image segmentation and data extraction tasks by allowing us to focus on the most important structural properties in an image, while also significantly reducing the amount of data by filtering out useless information.

2.2 Edge properties

Detecting edges in an image is oftentimes the first step in image processing. Edges typically are found in the boundary zone between two different objects or regions in images. Edges in images are areas with strong intensity contrasts – a jump in intensity from one pixel to the next. These sharp local changes in image intensity are usually associated with some sort of discontinuity. Image intensity is represented in the image by changes in the intensity function which displays different kinds of variation in the image. These image intensity variations can be either classified as a step discontinuity, which represents an abrupt change in image intensity from one side of the discontinuity to the other, or a line discontinuity which is an abrupt change in the image intensity, but then falls back to its original value in a short timeframe. However, because of low frequency components or image smoothing caused by sensing devices, these sharp discontinuities rarely exist in real settings. Instead, step edges become ramp type discontinuities and line edges become roof type discontinuities.

Figure 1: Types of Edges (a) Step (b) Ramp (c) Line (d) Roof

2.3 Noise

Edges are represented in images as sudden disparities. An ideal edge would be a unit step from one pixel to the next. However, many factors must be taken into consideration such as noise, brightness, corners, and similar Edge-Like Features. Of these, arguably the most problematic is noise. Image noise is random variations of brightness or color in an image which can be produced by scanner or digital cameras. To circumvent this issue of image noise, we apply

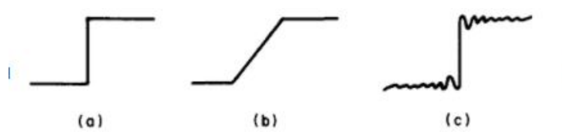


Figure 2: Effect of Noise on Edges(a) Step (b) Ramp (c) Noisy

something known as a Gaussian blur, which aims to smooth out the image. In blurring, we simply blur an image. By doing so, an image looks very sharp or more detailed because we are able to perceive all the edges and objects in the image. For example, An image of text looks clear when we are able to identify certain objects such as letters. This shape of an object is due to its edges. So in blurring, we simply reduce the edge content which makes the transition form one color to the other very smooth, thus mitigating noise.

2.4 Detection methods

Edge detection techniques are grouped into two categories: gradient of Laplacian. First order derivatives in an image are found using the gradient and second-order derivatives are found using the Laplacian. For the gradient edges are where the first derivative of the intensity is greater in magnitude than a specific threshold. Similarly, Laplacian is used by finding places where the second derivative of the intensity has a zero crossing (where the sign of a function changes.). We will be taking a look at two edge detection methods. The first is the Sobel edge detector, which uses a gradient, and the Canny edge detector which uses a Laplacian.

2.5 Sobel Edge detection

The derivative of a continuous image $f(x, y)$ assumes a local maximum in the direction of the edge. One technique is to measure the gradient of f in a particular location. This is accomplished by using a gradient operator. Such operators, also called kernels, provide finite-difference approximations of the orthogonal gradient vector f_x and f_y.

Sobel edge detection uses the Sobel Operator which is an approximation to a first derivative of an image. We use a 3x3 kernel matrix, one for the x direction (G_x) and one for the y direction (G_y). The gradient for x-direction has negative numbers on the left and positive numbers on the right. Similarly, the gradient for y-direction has negative numbers on the bottom and positive numbers on top. Both of the filters have zeros in the center

-1	0	+1
-2	0	+2
-1	0	+1

Gx

+1	+2	+1
0	0	0
-1	-2	-1

Gy

Here are the results of Sobel Edge Detection:

Figure 3: Sobel Edge Detection

2.6 Canny Edge Detection

Canny detection was devised by John Canny and is currently the state of the art method for edge detection. Canny operator is based on three criteria. Canny first uses a Gaussian function to smooth the image to get rid of noise. This is done because the Laplacian mask operator for the estimation of the second derivative is very sensitive to noise. Next we identify the areas in the image with the strongest intensity gradients by using a gradient such as the Sobel Kernel. The previous process produces results with relatively thick edges, so to thin out the edges, we must perform non maximum suppression. Maximum suppression works by finding the pixel with the maximum value in an edge. Double thresholding is then initiated, by accepting pixels as edges if the intensity gradient value exceeds an upper threshold and removing pixels if it doesn't. Finally edge tracking is performed, this means when the pixel is between the two thresholds then it will be accepted as an edge if it is adjacent to a pixel that is above the threshold.

Figure 4: Canny Edge Detection

3 Object Recognition

Object recognition is a computer vision technique for identifying objects in images or videos. Object recognition is a key output of deep learning and machine learning algorithms. When humans look at a photograph or watch a video, we can readily spot people, objects, scenes, and visual details. The goal is to teach a computer to do what comes naturally to humans: to gain a level of understanding of what an image contains. Object recognition consists of recognizing, identifying, and locating objects within a picture with a given degree of confidence.

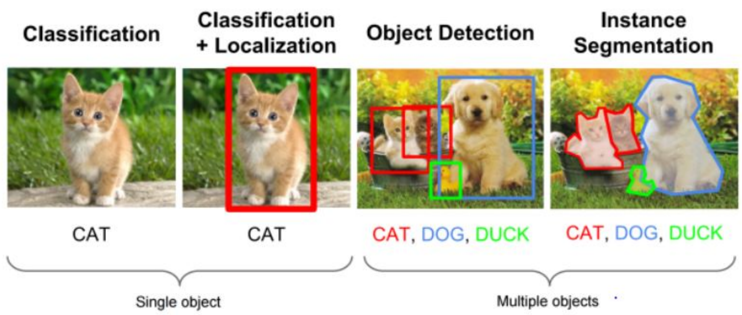
In this process, the four main tasks are:

- Image Classifiction - Predict the type or class of an object in an image. The input is an image with a single object, such as a photograph. The output is a class label (e.g. one or more integers that are mapped to class labels).

- **Object localization** - Locate the presence of objects in an image and indicate their location with a bounding box. The input is an image with one or more objects, such as a photograph. The output consists of one or more bounding boxes (e.g. defined by a point, width, and height).

- **Object Detection** - Locate the presence of objects with a bounding box and types or classes of the located objects in an image. The input consists of an image with one or more objects, such as a photograph. The output consists of one or more bounding boxes (e.g. defined by a point, width, and height), and a class label for each bounding box.

- **Object Segmentation** - we will talk about this in a later section.



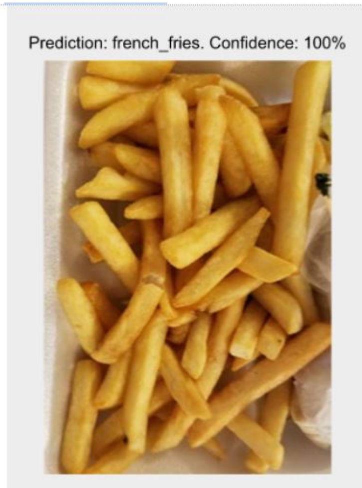
3.1 Object Recognition with deep learning

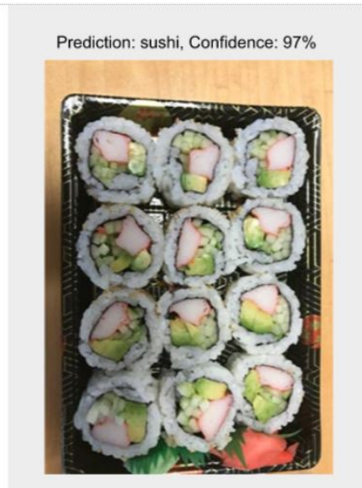
Deep learning techniques have become a popular method for doing object recognition. Deep learning models such as convolutional neural networks, or CNNs, are used to automatically learn an object's inherent features in order to identify that object. There are two approaches to performing object recognition using deep learning:

- **Training a model from scratch** : To train a deep network from scratch, you gather a very large labeled dataset and design a network architecture that will learn the features and build the model. The results can be impressive, but this approach requires a large amount of training data, and you need to set up the layers and weights in the CNN.

- **Using a pre trained deep learning model** : Most deep learning applications use the transfer learning approach, a process that involves fine-tuning a pretrained model. You start with an existing network, such as AlexNet or GoogLeNet, and feed in new data containing previously unknown classes. This method is less time-consuming and can provide a faster outcome because the model has already been trained on thousands or millions of images.

Deep learning offers a high level of accuracy but requires a large amount of data to make accurate predictions.





3.2 Object Recognition with Machine Learning

Machine learning techniques are also popular for object recognition and offer different approaches than deep learning. Common examples of machine learning techniques are:

- HOG (Histogram of oriented Gradients) feature Extractor and SVM (Support Vector M : Before the era of deep learning, it was a state-of-the-art method for object detection. It takes histogram descriptors of both positive (those images which contain objects) and negative(that image that does not contain objects) samples and trains our SVM model on that.

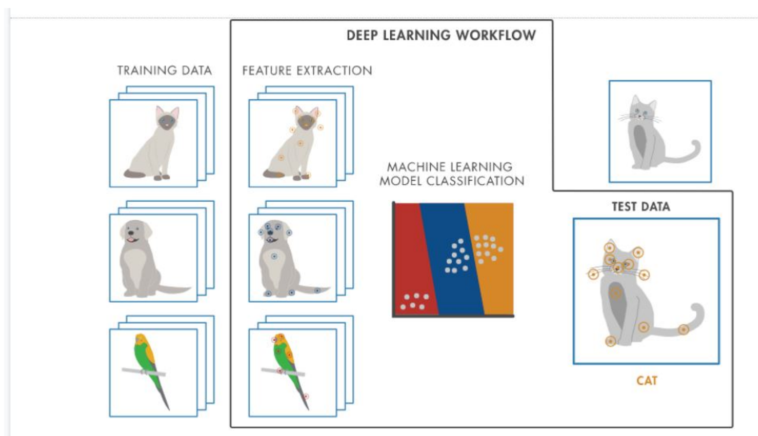
- Bag of features model : Just like a bag of words considers document as an orderless collection of words, this approach also represents an image as an orderless collection of image features. Examples of this are SIFT, MSER, etc.

- Viola-Jones algorithm : This algorithm is widely used for face detection in the image or real-time. It performs Haar-like feature extraction from the image. This generates a large number of features. These features are then passed into a boosting classifier. This generates a cascade of the boosted classifier to perform image detection. An image needs to pass to each of the classifiers to generate a positive (face found) result. The advantage of Viola-Jones is that it has a detection time of 2 fps which can be used in a real-time face recognition system.

3.3 Machine Learning Workflow

To perform object recognition using a standard machine learning approach, you start with a collection of images (or video), and select the relevant features in each image. For example, a feature extraction algorithm might extract edge or corner features that can be used to differentiate between classes in your data.These features are added to a machine learning model, which will separate these features into their distinct categories, and then use this information when analyzing and classifying new objects.You can use a variety of machine learning algorithms and feature extraction methods, which offer many combinations to create an accurate object recognition model.

Using machine learning for object recognition offers the flexibility to choose the best combination of features and classifiers for learning. It can achieve accurate results with minimal data.



3.4 Challenges faced in object Recognition

- Change in size, cropping out the background are some of the factors influencing the accuracy of the system. The accuracy of the model might change by scaling the image.
- Adjusting Brightness and Contrast of the image may also make it difficult for the system to recognize the objects in the image.
- There may be cases when the object might not be visible enough for the system to recognize it. The Object Recognition System must handle these cases of low visibility.
- The system may fail in cases where similar objects occur in groups and are too small in size.

- Various lightning conditions and shadows in the image may also pose difficulty for the system to recognize the object.

3.5 Applications of Object Detection

- Self-Driving Cars-Self Driving Cars may use Object detection and recognition systems to identify pedestrians and cars on the roads and then make the suitable decision in accordance.

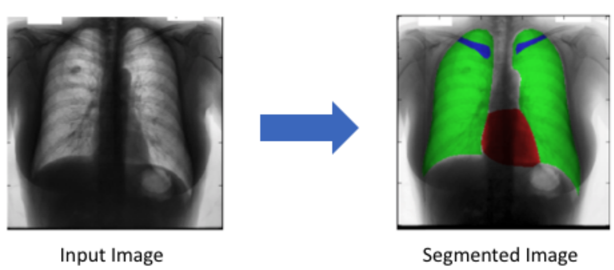
- Face Detection-Another application of Object detection and recognition is Face Detection .e.g.- Facebook recognizes people before they are tagged in images.

- Face Detection-Another application of Object detection and recognition is Face Detection .e.g.- Facebook recognizes people before they are tagged in images. Recognition-Text recognition deals with recognizing letters/symbols, individual words and series of words. ExRecognizing handwriting of a person.

- Hand Gesture Recognition- Hand Gesture Recognition deals with recognition of hand poses, and sign languages.

4 Semantic Segmentation

Semantic segmentation is a process in computer vision in which we divide an image into different segments. These segments usually correspond to different aspects of the input image. The goal of semantic segmentation is to form a segmentation map in which we classify each input pixel to a specific class label.



As shown in the figure above, different parts of the image are highlighted in different colors in order to distinguish between different parts of the body. The lungs are colored green, the heart is colored red, and the clavicles are colored blue. Now lets explore some different applications of semantic segmentation.

5 Applications of Semantic Segmentation

5.1 Autonomous Vehicle

An autonomous vehicle must take into account many dynamic variables in an open environment in order to ensure that the passenger and vehicle is able to travel safely. It must be able 1to perceive roads, cars, and traffic signs within its field of vision in order to plan the route of the vehicle for the next few seconds. To do this successfully, we use the process of semantic segmentation to make these distinctions.

5.2 Precision Agriculture

In order to reduce herbicide usage in farms, farmers have started using automated robots that automatically weed different areas of the farming fields. In this process, semantic segmentation is used to differentiate between crops and weeds.

5.3 Facial Segmentation

The modern day app snapchat uses semantic segmentation in its face filter features. Facial segmentation allows different parts of the face, such as the eyes, nose, mouth, and hair, to be distinguished, allowing face filters to make modifications to specific parts of the face.

6 References

https://www.cs.toronto.edu/ guerzhoy/320/lec/edgedetection.pdf

https://www.cse.usf.edu/ r1k/MachineVisionBook/MachineVision.files/MachineVision$_C$hapter5.pdf

https : //www.researchgate.net/publication/306394987$_E$DGE$_D$ETECTION−
APPLICATION$_O$F$_F$IRST$_A$ND$_S$ECOND$_O$RDER$_D$ERIVATIVE$_I$N$_I$MAGE$_P$ROCESSING : :
text = Edge%20detection%20is%20one%20of, st%20Order%20Derivative%20Filter)%20method.

https : //www.sciencedirect.com/science/article/pii/0146664X82900703

https : //www.cscjournals.org/library/manuscriptinfo.php?mc = IJIP − 15

https : //www.researchgate.net/profile/Shubhashree_Savant/publication/291975278_A_Review$_o$n$_E$dge_Det
Review−on−Edge−Detection−Techniques−for−Image−Segmentation.pdf

https : //www.jeremyjordan.me/semantic − segmentation/

https : //www.kdnuggets.com/2018/10/semantic − segmentation − wiki −
applications − resources.html

https : //machinelearningmastery.com/object − recognition − with − deep −
learning/

https : //www.mathworks.com/solutions/image−video−processing/object−
recognition.html

https : //www.ijedr.org/papers/IJEDR1704166.pdf