COMPUTER VISION
CAC II

Rakesh Pv
2147228
MCA B

1. Analyze different object detection, localization and classification methods.

METHODS FOR OBJECT DETECTION:

Locating instances of semantic things of a specific class (such as people, buildings, or cars) in digital movies and images is the aim of object detection, a computer vision-related technology. Object detection has established itself as a critical component for several important applications, such as face detection, autonomous driving, and video surveillance. Invariant to Scale Feature transformed and accelerated A real-time application of any complexity cannot use Robust Features or other effective methods for producing high-quality features because they consume too much processing power. Support vector machines and back-propagation neural network training are used for efficient object recognition based on the normalized corner information.

1) Scale-Invariant Feature Transform (SURF):
The SIFT technique can successfully recognize objects even among clutter and under partial occlusion since the SIFT feature descriptor is invariant to size, orientation, and affine distortion.

The following steps are used in SIFT algorithms to produce feature information: Scale-space extrema detection; Key-point Localization; Orientation assignment; Key-point Descriptor.

The Harris corner detector is used to extract features. During Scale-space Extrema detection, the interest spots (key points) are located at recognizable areas in the image. In keypoint localization, distinct key points are selected from among keypoint candidates by comparing each pixel of the detected feature to its neighbors.

The Keypoint descriptor yields SIFT descriptions that can withstand local affine distortion.This enables the keypoint description, which has a wide range of orientations and scales, to locate objects in photographs. The SIFT approach does not support real-time object recognition due to the expensive computation required for feature detection and keypoint descriptor generation.

2) Speeded-Up Robust Feature (SURF):
SURF techniques employ comparable detection strategies to SIFT algorithms. The scale space is created via SURF methods through distribution changes rather than the Difference of the Gaussian (DoG) filter. SURF uses a box filter model to simulate the Gaussian Laplacian. They are much faster than SIFT algorithms because of the simplification of scale-space extrema detection, which speeds up feature extraction. To speed up computation, SURF uses integral images for image convolutions.

Within the interest point neighborhood, the descriptor describes a distribution of Haar-wavelet responses:
Descriptor generation, orientation and size assignment, and feature extraction.

An image descriptor is generated by measuring an image gradient. Image descriptor-based SURF approaches are impervious to various image alterations and occlusion-related image disturbance. Even though feature calculation and matching require less time, they still need help providing real-time object recognition in embedded system contexts due to the available resources.

3) Features of the corner detector from the Accelerated Segment Test (FAST):
Corners in an input image stand out from the surrounding pixels in specific ways. Corners can be consistently spotted and tracked even when geometric defects exist in the photos. Therefore, most object recognition algorithms use corner information to extract features. Although the Harris corner detector of the SIFT approach performs well, its slow computation time makes it unsuitable for real-time object detection.

The FAST corner detector is ten times faster than the Harris corner detector without sacrificing performance. It can find corners by examining a sixteen-pixel circle surrounding the corner candidate.

Suppose the intensities of a predetermined number of adjacent pixels are uniformly above or uniformly below the power of the center pixel by a predetermined threshold. In that case, this candidate is identified as a corner. The areas of the image with the highest contrast are where the interest points were taken from

4) Fast R-CNN:
A Fast R-CNN network receives as inputs an entire image plus a list of suggested objects. The network first processes the whole image using several convolutional and max pooling layers before producing a convolutional feature map. The feature map is then used to create a fixed-length feature vector for each object proposition by a region of interest (RoI) pooling layer.
Each feature vector is fed into a series of fully connected (FC) layers, each of which branches into two sibling output layers: one layer outputs four real-valued numbers for each of the K object classes, and another layer generates softmax probability estimates over the K object classes.

Each set of 4 values for a particular K-class encodes precise bounding-box coordinates.

Fast R-CNN offers the following benefits:

* PPnet (Spatial Pyramid Pooling) has higher detection quality (mean Average Precision) than R-CNN. Training is one-stage and uses a multi-task loss.
* All network layers can be updated by training.
* There is no need for disc storage for feature caching.

5) You Only Look Once (YOLO):

This innovative approach to object detection is new and original. Using classifiers to detect objects has been done before. YOLO frames object identification as a regression problem by utilizing spatially different bounding boxes and accompanying class probabilities.

A single neural network can directly predict bounding boxes and class probabilities from entire images in a single evaluation. The detection performance may be modified from start to finish because the whole detection pipeline is made up of a single network.

Unlike sliding window and region proposal-based approaches, YOLO sees the entire image during training and testing, so it implicitly stores contextual information about classes in addition to their appearance.

Because it lacks context awareness, the popular object detection method Fast R-CNN sees background patches in an image as objects. YOLO generates fewer background errors than Fast R-CNN, which is less than half as many.

METHODS FOR OBJECT LOCALIZATION :

Image localization is a branch of standard CNN vision techniques. These algorithms predict the discrete number classes. The object localization approach predicts a collection of 4 serial numbers, namely the x coordinate, y coordinate, height, and width, to create a boundry box around an object of interest.

Initial layers in CNN-based classifiers are convolutional neural network layers. The number of layers utilized can range from a few to 100 (for example, ResNet 101), depending on the application, the amount of input, and the available processing capacity. The quantity of layers is a crucial subject for investigation. The CNN layers are followed by a pooling layer and one to two ultimately linked layers.

An object's likelihood of appearing in an image is provided by the output layer, which is the last layer. Consider a scenario where an algorithm identifies 100 unique objects in a snap. The final layer then produces an array with a length of 100 and values between 0 and 1, indicating the likelihood of an object appearing in an image.

Any machine learning algorithm's objective is to produce predictions that are as accurate as feasible. A loss function is a necessary component of any supervised machine-learning technique, and algorithms learn by minimizing loss through optimizing weights or parameters.

Since object localization is a regression problem, any regression loss function suitable for an N-dimensional array can be used—Huber loss, L1 and L2 distances, for instance, and so forth. As previously mentioned, L2 distance loss is commonly used by both the corporate and scientific communities.

Distance L2

L2 distance is also referred to as euclidean distance. Using their cartesian coordinates, the Pythagorean theorem is used to find the distance between two endpoints in N-dimensional space. With this approach, any N-dimensional space can be used.

Let's assume two points, P and Q, in three-dimensional space to better comprehend L2 distance. In the cartesian coordinate systems, points P and Q are denoted by (p1, p2, p3) and (q1, q2, q3), respectively. These coordinates' separation is given as:

Distance (P, Q) is equal to sqrt((p1-q1) + p2-q2) + p3-q3)).

The lower the L2 distance between the prediction and the ground truth, the better the method; the optimization algorithm seeks to minimize the Euclidean distance between the ground truth and the forecasted value.

OBJECT CLASSIFICATION METHODS:

A computer vision technique called object detection is used to find occurrences of objects in pictures or movies. To generate useful results, object detection algorithms frequently use machine learning or deep learning. Humans can quickly identify and pinpoint objects of interest when viewing photos or videos. Using a computer, object detection aims to simulate this intelligence.

1) Unsupervised classification : A fully automatic methodology called unsupervised classification does not use training data. This means that without the need for human participation, machine learning algorithms are utilised to evaluate and cluster unlabeled datasets by identifying hidden patterns or data groups.

The unsupervised categorization algorithm K-means divides items into k categories according to their traits. Another name for it is "clusterization." One of the most straightforward and well-liked unsupervised machine learning algorithms is K-means clustering.

Iterative Self-Organizing Data Analysis Technique, or ISODATA, is an unsupervised approach for classifying images. The ISODATA approach incorporates iterative techniques that divide data components into various classes using Euclidean distance as the similarity metric. The ISODATA approach allows for a varying number of clusters, but the k-means algorithm presumes that the number of clusters is known a priori (in advance).

2) Supervised classification : In order to train the classifier and subsequently classify fresh, unknown data, supervised image classification algorithms require previously classified reference samples (the ground truth). The method of visually selecting training data samples from inside the image and assigning them to pre-selected categories, such as flora, roads, water resources, and buildings, is hence known as the supervised classification technique. To establish statistical measurements that can be used to analyse the overall image, this is done.

Prior to a few years ago, security applications were the principal use cases for picture categorization. However, picture classification use cases are becoming more common in a variety of industries today, including health care, industrial manufacturing, smart cities, insurance, and even space exploration. The exponential increase in visual data availability and the quick development of superior computing technology are two factors contributing to the rise in applications. This data can be valuablely extracted via image classification. Visual data has equity when used strategically because the value derived from applications throughout the company exceeds the expense of storing and maintaining it.

2. Identify the limitations of the existing methods and propose a robust and consistent model that automatically detect and classify the different objects in an image.

There are several limitations to the existing object detection methods. One such limitation is the computational cost of the methods. The current methods are very resource intensive and require a lot of processing power. This makes them impractical for many applications. Another limitation is the accuracy of the methods.

The current methods could be more accurate and often fail to detect objects accurately. This limits their usefulness in many applications. Finally, the current methods could be more robust and often susceptible to false positives and negatives. This limits their usefulness in many applications.

Most existing object detection methods are based on hand-crafted features, such as Haar-like features, SIFT, or HOG. These features are designed to be robust to common image transformations, such as translation, rotation, and scale. However, they are only sometimes well-suited to some objects and all images. In addition,these features are often designed with a specific object category in mind, such as faces or pedestrians. As a result, they may need help to generalize to other object categories.

Deep learning methods have recently been proposed for object detection. These methods learn a set of features from data instead of using hand-crafted features. Deep learning methods have the potential to be more generalizable than hand-crafted features, but they are still limited in several ways:
Deep learning methods require a large amount of data to learn good features. This is a problem for many real-world applications where there may need to be more data available.
Deep learning methods are often slow, which makes them impractical for many real- time applications.
Deep learning methods are not yet well-understood, and it is not easy to design suitable architectures for deep neural networks.
Despite these limitations, deep learning methods have shown promise for object detection, and they are likely to improve as more research is done.

Object Detection - MXNet - a consistent model that automatically detects and classifies the different objects in an image

The Amazon SageMaker Object Detection:
MXNet algorithm detects and classifies objects in images using a single deep neural network. It is a supervised learning algorithm that takes images as input and identifies all instances of objects within the image scene. The object is categorized into one of the classes in a specified collection with a confidence score that it belongs to the class. Its

location and scale in the image are indicated by a rectangular bounding box. It uses the Single Shot multi-box Detector (SSD) framework and supports two base networks: VGG and ResNet.
The network can be trained from scratch or trained with models that have been pre-trained on the ImageNet dataset.

Input/Output Interface for the Object Detection Algorithm

The SageMaker Object Detection algorithm supports both Record IO (application/x recorded) and image (image/Png, image/jpeg, and application/x-image) content types for training in file mode and supports Record IO (application/x-recorded) for training in pipe mode. However, you can also train in pipe mode using the image files (image/Png, image/jpeg, and application/x-image), without creating Record IO files, by using the augmented manifest format. The recommended input format for the Amazon SageMaker object detection algorithms is Apache MXNet Record IO.

However, you can also use raw images in .jpg or Png format. The algorithm supports only application/x-image for inference.

Train with the Record IO Format
If you use the Record IO format for training, specify both train and validation channels as values for the Input Data Config parameter of the Create Training Job request. Specify one Record IO (.rec) file in the train channel and one Record IO file in the validation channel. Set the content type for both channels to application/x- recorded. An example of generating a Record IO file can be found in the object detection sample notebook. You can also use tools from MXNet's Gluon CV to generate Record IO files for popular datasets like the PASCAL Visual Object Classes and Common Objects in Context (COCO).

How Object Detection Works
The object detection algorithm identifies and locates all instances of objects in an image from a known collection of object categories. The algorithm takes an image as input and outputs the category that the object belongs to, along with a confidence score that it belongs to the category. The algorithm also predicts the object's location and scale with a rectangular bounding box. Amazon SageMaker Object Detection uses the Single Shot multi-box Detector (SSD) algorithm, which takes a convolutional neural network (CNN) pretrained for classification as the base network. SSD uses the output of intermediate layers as features for detection.

Various CNNs, such as VGG and ResNet, have achieved excellent performance on the image classification task. Object detection in Amazon SageMaker supports both VGG-16 and ResNet-50 as a base network for SSD. The algorithm can be trained in full training mode or in transfer learning mode. In full training mode, the base network is initialized with random weights and then trained on user data. The base network weights are loaded from pre-trained models in transfer learning mode. The object detection algorithm uses standard data augmentation operations, such as flip, rescale, and jitter, on the fly internally to help avoid overfitting.

Tune an Object Detection Model
Automatic model tuning, also known as hyperparameter tuning, finds the best version of a model by running many jobs that test a range of hyper parameters on your dataset. You choose the tunable hyperparameters, a range of values for each, and an objective metric. You choose the objective metric from the metrics that the algorithm computes. Automatic model tuning searches the hyperparameters chosen to find the combination of values that result in the model that optimizes the objective metric.

There are a few potential advantages to using AWS MXNet for object detection:

1. MXNet is a compelling and efficient deep-learning framework that can help you train accurate models quickly.
2. AWS provides a managed service for MXNet, which can take care of your infrastructure and scalability issues.
3. Object detection is a relatively new field, and MXNet is one of the most popular frameworks for it, so a large community of users and developers can help you if you need assistance.

However, there are also a few potential disadvantages to using AWS MXNet for object detection:

1. MXNet can be challenging to learn and use, especially if you have yet to become familiar with deep learning frameworks.
2. AWS MXNet is a managed service, so it may be more expensive than a self-hosted solution.
3. Object detection is a relatively new field, so some kinks still need to be ironed out. For example, MXNet does not currently support real-time object detection, so you need to use a different solution if you need that functionality.