Expt_9

Prathmesh Malvi - 38

Aim: To Creating and Training an Object Detector

Objective: Bag of Words BOW in computer version Detecting cars in a scene

Theory:

Creating and Training an object detector:

Creating and training an object detector from scratch can be a complex and resource-intensive task. It typically involves the use of deep learning frameworks like TensorFlow or PyTorch. Training object detectors from scratch typically requires significant computational resources, including powerful GPUs, and a good understanding of deep learning concepts. Many object detection tasks benefit from leveraging pre-trained models on large datasets and fine-tuning them on a smaller, domain-specific dataset.

Following are the steps for creating and training an object detector:

1. **Data Collection and Annotation:** Collect a dataset of images that contain the objects you want to detect and annotating them.
2. **Data Preprocessing**: Resize and normalize the images. Augment the data to increase the diversity of the dataset.
3. **Choose a Deep Learning Framework:** Select a deep learning framework like TensorFlow, PyTorch, or a specialized object detection library like Detectron2 (built on PyTorch) or TensorFlow Object Detection API.
4. **Model Architecture:** Choose an object detection architecture, such as Faster R-CNN, YOLO (You Only Look Once), or SSD (Single Shot MultiBox Detector).
5. **Loss Function:** Define a loss function appropriate for object detection.
6. **Training:** Train the model on your annotated dataset.
7. **Evaluation:** Evaluate the trained model on a separate validation dataset to assess its performance. Monitor the loss and accuracy metrics during training to ensure the model is learning correctly.

8. **Deployment:** Deploy the trained model for real-time or batch inference in your desired application.

Bag - Of - Words:

The Bag of Words (BoW) is a commonly used technique in natural language processing and computer vision for text and image analysis. It's a simple and effective way to represent data for various tasks like text classification, document retrieval, and image classification. BoW is used to convert variable-length sequences (such as sentences or images) into fixed-length feature vectors.

BOW in Computer Vision:

In computer vision, the bag-of-words model (BoW model) sometimes called bag-of-visual-words model can be applied to image classification or retrieval, by treating image features as words. In document classification, a bag of words is a sparse vector of occurrence counts of words; that is, a sparse histogram over the vocabulary. In computer vision, a bag of visual words is a vector of occurrence counts of a vocabulary of local image features.

Here's how BoW is typically applied in computer vision:

- Feature Extraction
- Clustering
- Feature Quantization
- Histogram Generation
- Normalization
- Image Classification
- Image Retrieval

Detecting Cars Example:

To detect cars using the Bag of Words (BoW) model in computer vision, you'll begin by collecting a labelled dataset containing images of cars and non-car images. Next, you'll extract local features from these images, such as Histogram of Oriented Gradients (HOG) or other distinctive patterns. These features serve as the basis for identifying cars.

The BoW approach involves clustering these local features into visual words or codewords using a technique like K-Means clustering. The resulting clusters form a visual vocabulary or codebook, where each cluster centre represents a visual word.

Once you have the visual vocabulary, you can assign the local features from each image to their nearest visual word in the codebook, essentially quantizing the features into discrete visual words. This step is crucial for transforming the image data into a format suitable for machine learning.

For each image, you generate a histogram that counts the occurrences of each visual word. Car images will have characteristic histograms that correspond to the distribution of visual words associated with cars, making them distinguishable from non-car images.

To train a car detection model, you'll use these histograms as feature vectors, labelling each training sample as "car" or "non-car." Machine learning classifiers like Support Vector Machines (SVMs) or Random Forests are commonly employed for this purpose.

In practice, to detect cars in new images, you extract local features, create histograms, and feed them into the trained classifier. The classifier then outputs predictions, effectively identifying cars in the images. Post-processing techniques like non-maximum suppression can help refine the detected car regions and eliminate duplicates or overlapping detections.

Conclusion:

The Bag of Words Model is a useful computer vision tool for object recognition and automobile detection. In the context of car detection, the BoW model can be used to identify the presence and location of cars within an image or video frame. It is a flexible solution for a variety of applications, including traffic surveillance, autonomous vehicles, and more, because it makes use of the power of visual features, language building, and machine learning to recognize cars in image or scenarios.

In this experiment, we successfully studied the approach for detection of cars using Bag Of words Model.