**Paper: Objects detection using deep learning Algorithm CNN**

Name: Milon Hossain

ID: 1610497042

**Abstract:** Object recognition is an advanced computer vision and image processing application that deals with the identification of objects in visual images of a specific class. One of difficult and demanding tasks CNN in computer vision has been considered in this project. In order to identify objects in an image, many machine learning-based approaches such as SIFT (Scale-invariant feature transformation) and HOG (Histogram of directed gradients) are commonly used. For grouping, these methods use to help vector machine. For these methods, the main obstacles are that they are computationally intensive for use in real-time applications, even for large database, these approaches do not perform well. For this a Deep Learning based Convolutional Neural Network (CNN) applied in this project.

**Introduction:** Humans, including photograph and audio files, are very good at decoding unstructured data. Computers are much better at processing unstructured data relative to a few years earlier, one of the strongest things about the emergence of deep learning. This provides opportunities for many exciting new applications, such as object detection, speech recognition, image recognition etc. Object detection deals with the recognition of individual objects along with their location in an image.

**Convolutional Neural Network (CNN**): Convolutional Neural Network (CNN) model has been used in this project. Convolutional Neural Network is a deep learning-based algorithm that can take an image as input, assign classes for the objects in the image. It differentiates the objects one from another. The processing in ConvNet is much lower when compared to other classification algorithms. ConvNet is used to reduce the images without losing features, which helps in getting the right prediction. It can successfully capture the spatial and temporal dependencies in an image through the application of relevant filters. This model was chosen due to three advantages such as detection accuracy, fewer training parameters and runs on minimum computation hardware.

**Dataset:** The dataset used in this model contains 2500 images belonging to 5 different classes, which are car, cycle, motorbike, dog and flowers. The dataset downloaded from TensorFlow Datasets, which is a collection of datasets ready to use. The dataset has divided into Training Set and Testing Set. To train the model, here use images in training set and the testing set contains the images used for performance evaluation.

**Analysis:** PyQt5 have been implemented for the front-end interface. When the user chooses to build a CNN model, the given dataset trained according to the CNN algorithm, that implemented 5 datasets or classes in this project. To build the CNN model, the training data split into Training Set and Validation set. Validation set is a part of the training data that can be used to build a model. Validation set is used to avoid overfitting. Training on validation loss is the error on the training set of data. In contrast, validation loss is the error after running the validation set of data through the trained network. After that user will give an image as an input into the system and the system will differentiate all the objects into the picture.

**Proposed Algorithm:**

1. Input Layer
2. Convolutional Layer
3. ReLU Activation Function
4. Pooling Layer
5. Flattening
6. Fully Connected Layer
7. Soft Max

**Conclusion:** This paper proposes a new approach to recognize and detects objects in an image. This proposed approach can be useful for the system, which seeks object detection as a prime feature. The proposed system CNN gives accurate results in detecting objects in an image along with the location. CNN is an efficient way to work with more massive datasets. Sometime this system can give more than 96% accurate information while it detecting an image.

**Reference:**

1. https://www.academia.edu/43726642/Object\_Detection\_using\_Deep\_Learning\_Algorithm\_CNN