CHAPTER 3
RESEARCH METHODOLOGY

3.1. Data Collection

This research required fruits and vegetables images dataset for training the algorithm. The dataset that are being used in this research were obtained through Kaggle by title Fruits 360 dataset: A dataset of images containing fruits and vegetables as with 2020.05.18.0. for the version. The dataset set can be obtained simply by downloading the dataset bundle and the 1.31 GB zip package already consists of all the mentioned below. Available fruits and vegetables image dataset cover 360 degree images with original size and scaled size which is 100x100 pixels for the image size all with extracted background.

Training set and test set have already been incorporated once you download the package. This data set comprises 131 kinds of fruit with the total number of images 90483. It is divided into a training set which consists of a total 67692 images and a test set with a total 22688 images. As for the filename format "r" stands for rotated fruit as an example imageindex100.jpg (e.g. 32100.jpg) or rimageindex100.jpg (e.g. r32100.jpg) or r2imageindex100.jpg or r3imageindex100.jpg. "r2" stands for the fruit was rotated in the 3rd axis and "100" comes from image pixels size which are 100x100 pixels. Dataset images come in RGB values. In this research only the training set is used.

As for the density table, fruits and vegetables density obtained from various sources which has the best accuracy for particular fruits and vegetables. Density table formed in csv file format. 2D input images in this research are obtained by using the iPhone X built-in camera as our device.

3.2. Algorithm

Clustering and classification takes an important part of this research as to achieve an accurate object detection. The author is going to use both algorithms for clustering in order to make classification training simpler where classification only trained data where the cluster has similar features with the input image. At first, the author sought K-means for clustering algorithm but K-means usage is overrated already and depends on ‘k’ value while DBSCAN doesn’t require a specific number of clusters. Algorithms that featured in this research are DBSCAN and k-NN as DBSCAN for clustering and k-NN for the classification. Although DBSCAN is less popular among the other algorithms, DBSCAN is great at clustering.

Reasons behind the DBSCAN algorithm is a great help in clustering is because noise doesn't affect the algorithm, DBSCAN can find a cluster completely surrounded by another cluster, and DBSCAN can discover an arbitrarily shaped cluster. In this paper, the clustering algorithm of DBSCAN which relies on a density-based notion of clusters is being presented. The DBSCAN algorithm requires only two input parameter and the algorithm supports in determining an appropriate value for it. Another advantage of DBSCAN is that there is no requirement for using training data. DBSCAN doesn't perform prediction, the algorithm only tries to cluster similar data points so it performs clustering on the actual data. DBSCAN also looks into the spatial density of data points which makes DBSCAN different among other clustering algorithms. Additionally, DBSCAN is robust toward outlier detection all based on the data points density matrix, low density areas are separated as the outliers.

After the data has already been clustered it becomes classification training so that image classification can be done easily and fast. The clustered part where the cluster is similar to the related input image is taken for classification training data. As for the classification, the author uses k-NN because a lot of research before that has been conducted using k-NN has great results besides k-NN is a great classification algorithm, there's disadvantages with using k-NN because k-NN doesn't perform well with large dataset which the author covered by clustering the dataset first using DBSCAN and k-NN is famous for lazy learner which there is no training period in as a result fast prediction can be accomplished.

Both of the algorithm applications in order to obtain accurate object detection, after the object has successfully detected the result will be matched into a density table and by using a mathematical formula along with OpenCV for volume estimation, the object's mass can be found.

3.3. Design

Clustering and classification takes an important part of this research as to achieve an accurate object detection. The author is going to use both algorithms for clustering in order to make classification training simpler where classification only trained data where the cluster has similar features with the input image. At first, the author sought K-means for clustering algorithm but K-means usage is overrated already and depends on 'k' value while DBSCAN doesn't require a specific number of clusters. Algorithms that featured in this research are DBSCAN and k-NN as DBSCAN for clustering and k-NN for the classification.

Although DBSCAN is less popular among the other algorithms, DBSCAN is great at clustering. Reasons behind the DBSCAN algorithm is a great help in clustering is because noise doesn't affect the algorithm, DBSCAN can find a cluster completely surrounded by another cluster, and DBSCAN can discover an arbitrarily shaped cluster. In this paper, we presented the clustering algorithm DBSCAN which relies on a density-based notion of clusters. It requires only one input parameter and supports the user in determining an appropriate value for it. Another advantage of DBSCAN is that there is no requirement for using training data. DBSCAN doesn't perform prediction, the algorithm only tries to cluster similar data points so it performs clustering on the actual data. DBSCAN also looks into the spatial density of data points which makes DBSCAN different among other clustering algorithms. Additionally, DBSCAN is robust toward outlier detection all based on the data points density matrix, low density areas are separated as the outliers.

After the data has already been clustered it becomes classification training so that image classification can be done easily and fast. The clustered part where the cluster is similar to the related input image is taken for classification training data. As for the classification, the author uses k-NN because a lot of research before that has been conducted using k-NN has great results besides k-NN is a great classification algorithm, there's disadvantages with using k-NN because k-NN doesn't perform well with large dataset which the author covered by clustering the dataset first using DBSCAN and k-NN is famous for lazy learner which there is no training period in as a result fast prediction can be accomplished.

Both of the algorithm applications in order to obtain accurate object detection, after the object has successfully detected the result will be matched into a density table and by using a mathematical formula along with OpenCV for volume estimation, the object's mass can be found.

3.4. Coding

Python ver 3.8.2 becomes this research main computing language. Major reasons behind the author choosing python are because python is a really compatible machine learning language with various tools provided by python. Python simplicity and consistency has proven by the popularity of this computing language among the AI and machine learning communities. Wide variety of great libraries can be accessed easily to help develop AI and machine learning models.

Besides, python has incredible visualization tools where users can visualize their dataset and by visualization tools it accommodates data analysis better. Besides various features provided by python, python is human-friendly, easy to understand and implement makes python the right choice for machine learning.

As for the density table the author chose to use in csv form in order for easier human readability and relatively easy to manipulate csv file. With the help of pandas csv can be read fast by python.

3.5. Analysis

The author uses 131 kinds of fruits with a total of 67692 images. Further analysis required in order to measure the success of the algorithm for object detection. Compatible dataset is needed to achieve high accuracy. The author will use the analysis of 10, 20, 30,and 40 kinds of fruits and vegetables. How many kinds of fruits and vegetables the algorithm can read accurately. Besides, the DBSCAN algorithm has particular variables such as eps and minPts which need to be defined and support the clustering accuracy. On the other hand, input images have major roles. Another analysis is needed to find if different lighting and background affect the image clarity and detection or not which we discussed deeper in the next section. As for evaluation, accuracy measure about comparison between actual mass and estimated mass will be applied.