IOP Conference Series: Materials
Science and Engineering

Detection of partially occluded objects – A comparative analysis based on Haar Classifier and K-Means Clustering

To cite this article: Aparajita Karmakar and D Aju 2021 *IOP Conf. Ser.: Mater. Sci. Eng.* **1145** 012043

IOP Conf. Ser.: Mater. Sci. Eng. **1145** (2021) 012162 https://doi.org/10.1088/1757-899X/1145/1/012162

Retraction

Retraction: Detection of partially occluded objects – A comparative analysis based on Haar Classifier and K-Means Clustering (*IOP Conf. Ser.: Mater. Sci. Eng.* **1145** 012043)

Published 23 February 2022

This article (and all articles in the proceedings volume relating to the same conference) has been retracted by IOP Publishing following an extensive investigation in line with the COPE guidelines. This investigation has uncovered evidence of systematic manipulation of the publication process and considerable citation manipulation.

IOP Publishing respectfully requests that readers consider all work within this volume potentially unreliable, as the volume has not been through a credible peer review process.

IOP Publishing regrets that our usual quality checks did not identify these issues before publication, and have since put additional measures in place to try to prevent these issues from reoccurring. IOP Publishing wishes to credit anonymous whistleblowers and the Problematic Paper Screener [1] for bringing some of the above issues to our attention, prompting us to investigate further.

[1] Cabanac G, Labbé C and Magazinov A 2021 arXiv:2107.06751v1

Retraction published: 23 February 2022

Detection of partially Occluded Objects – A comparative Analysis based on Haar Classifier and K-Means Clustering

Aparajita Karmakar[1] and Aju D[2 *]

[1]School of Information Technology and Engineering, Vellore Institute of Technology, Tamil Nadu, India.

[2]School of Computer Science and Engineering, Vellore Institute of Technology, Tamil Nadu, India.
daju@vit.ac.in*

Abstract. This paper addresses the problem of detecting partially occluded objects from 2D images. The detection of partially occluded objects is performed and compared using feature-based training and color-based object segmentation. The occluded objects are very difficult to be detected based only on their features since, all the essential features may not be visible to the learned model due to occlusion. Haar cascade classifier has been utilised for feature-based training and the k-means clustering is utilized for color-based tracking. Various input images are provided Haar classifier as well as the K-means clustering to detect the objects in the 2D images and the subsequent results are compared and analysed. For segmenting the 2D objects using k-means clustering, the average recall and average precision varies from 0.70 to 0.98. The variation is based upon the veracity of the occluded objects. The average precision rate for detecting the occluded 2D objects through the developed method is between 0.24 and 0.60. And it is noted that the average recall for the respective detection lies between 0.25 and 0.70.

Keywords: Occlusion, Object detection, Haar Cascade, K-means

1. Introduction

Object detection has been one of the most challenging and difficult tasks in computer vision. Many methods have been proposed in the past for efficiently and effectively detecting the objects in the 2D images. Some of these techniques involve training or segmentation, and edge detection. However, detecting partially occluded objects is still far more difficult and challenging than detecting non-occluded objects.

Object detection by training can be performed by utilizing neural networks or machine learning techniques that involves developing a model based on the features extracted [1] from the digital image. Segmentation on the other hand, detects object based on edge tracking or intensity differences [2].

Object detection algorithms take an image as an input and produce bounding box values based on the location of the objects [3]. One of the major drawbacks of object detection method are that they cannot determine the shape, area and perimeter [4] of an object.

In image segmentation, the target objects that are using pixel masks are marked. It is more granular than object detection as it helps us to determine the shape of the object. Basically, Image segmentation separates the image into regions of different shapes and colors and subsequently highlights the areas of importance for further processing. It provides pixel based details of an image that makes it more efficient than the object detection. After detection, it can be marked with the bounding box. Different objects can be labelled [5][6] with different colors based on their boundaries, colors or textures.

Figure 1. (a) Object Detection (b) Image Segmentation

The object detection by marking the objects with bounding box is shown in Figure1a. And the second image, Figure 1b shows the segmented objects with masks [7] of different colors. Haar cascade classifier is used to train the feature based detection model and the results obtained from the Haar cascade classifier based method is compared and analysed with another color based segmentation method using K-means clustering. Finally, by comparing both these object detection methods, the respective result analysis is presented here.

2. Materials and Methods

The image dataset is obtained from Carnegie Mellon's Kitchen dataset. This dataset contains 1600 images of 8 household items (i.e. cup, pitcher, shaker, thermos, shaker, scissors, and baking pan) that are partially visible under occluded kitchen environments [8]. For the comparative analysis, the single-view images are considered and experimented.

2.1 Object detection using Haar Cascade Classifier

Paul Viola and Michael Jones proposed this effective object detection method using machine learning in 2001. A cascade function is trained with a number of positive and negative images. Since an object can be clearly distinguished from its surroundings based on positive and negative images, the haar cascade classifier can also be used for edge detection. Once the classifier is trained, it can successfully detect objects and their edges from the validation set.
The Haar Cascade Feature Detection algorithm first considers a fixed size detection window. This window consists of adjacent rectangular regions to the current region of interest. It uses three types of features: two-rectangle features, three rectangle features and four rectangle features. The intensities of all the region's in this window are summed up, and compared to a classifying threshold, which categorizes an object and a non-object [9].
We will be using this haar cascade classifier for edge detection in a Grayscale version of our image, as object detection has more to do with edges and less to do with color. Hence, computation for object detection is significantly faster on a Grayscale image, while returning the same accuracy as you would expect from a RGB image.

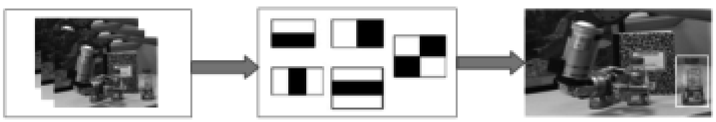

Figure 2. Haar Cascade Object detection architecture

The architecture in Figure 2 depicts our model implementation. The grayscale images are given as an input to the created Haar classifier model which returns the bounding box each time an object is detected. The model is trained with 1000 images of each object and tested on 100 images having the object in an occluded environment [10].

2.1.1 Haar Cascade Classifier implementation algorithm: Target object Detection

Step 1: Start.
Step 2: Read all 'jpg' images and resize to (100 X 100).
Step 3: Convert the images to grayscale for faster processing.
Step 4: Save all the negative images in a directory.
Step 5: Create a text file to refer the negative images.
Step 6: For all positive images, resize the images to (50 X 50) size.
Step 7: Create training samples.
Step 8: Create a list of positive samples by superimposing the resized positives on to the negative backgrounds in different angles.
Step 9: Create positive vectors with 20 height by 20 width.
Step 10: Train the cascade for 10 epochs through the positive vectors.
Step 11: Create xml file for the trained output.
Step 12: Create cascade objects through the trained xml file.
Step 13: Import the test images.
Step 14: Convert the test images to grayscale.
Step 15: Detect the target object from the test images through the cascade object.
Step 16: Draw the resulting bounding box values on the test image.
Step 17: Repeat steps 2 to step 15 for all kitchen items.
Step 18: Stop.

The above steps describe the implementation of Haar Cascade on our dataset.

2.2 Object detection using K-means Clustering

Segmentation is a widely used method in digital image processing that can represent an object and compute on further for object identification. Image pixels can be grouped based on its intensity, colors and textures [11]. It is often used in detection of region of interest or recognizing a pattern.
K-means is one of the most popular clustering algorithms that can be used to cluster and classify objects. Clustering is an unsupervised technique used to group data points of similar characteristics in same clusters. Therefore, it is used in processing a digital image to group similar pixels. K-means groups pixels into K (pre-determined) clusters and subsequently, it assigns one data point to one cluster only. The number of cluster and the initial centre is given by the user. The algorithm calculates the squared distance between the pixels and the centroid and assigns the pixel to the nearest centroid. Once all pixels are grouped, it again calculates the new centroid and repeats the whole method until a maximum iteration or error value is reached [12].
The K-Means clustering used here is performed on the RGB image, and the clusters are formed based on color and shade.

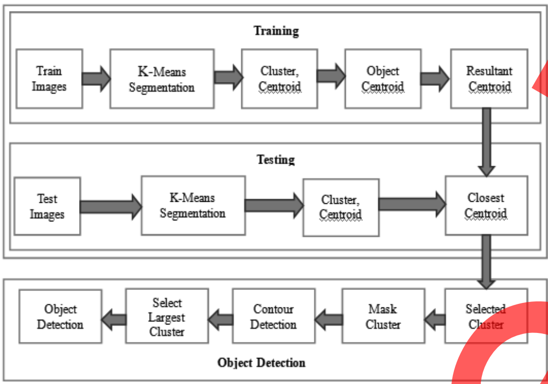

Figure 3. K-means Object detection architecture

The pipeline in Figure 3 depicts our working flow of the K-means algorithm. K-means is applied on the training images which returns the cluster centroids. From the centroids found, we select the centroid of our object and store it for further processing. Next, the test images are clustered from which we get the resulting centroids. We compare each centroid and select the cluster with the closest value to our previously stored training object centroid. The cluster is masked out and contours are found. From the contours we select the largest one and apply a bounding box around the object.

2.2.1 K means: Training Images

Step 1: Start.
Step 2: Import training image.
Step 3: Resize the training image to (1 X 3) array.
Step 4: Convert the resultant array to a 32-bit floating point image.
Step 5: Ensure maximum color separation by setting the number of cluster to 10.
Step 6: Maximum iteration is set to 100
Step 7: Set the value of epsilon to 0.2 of accuracy.
Step 8: Set the number of attempts to 10 and Initialize the centroid.
Step 9: Convert the centroids to integer type.
Step 10: Flatten the labels to one dimension.
Step 11: Copy the original image and resize to (1 X 3) array.
Step 12: Store the centroid value of the cluster to which the target object belongs.
Step 13: End.

2.2.2 K means: Test Images

Step 1: Start
Step 2: Import testing image.
Step 3: Resize the testing image to (1 X 3) array.
Step 4: Convert the resultant array to a 32-bit floating point image.
Step 5: Ensure maximum color separation by setting the number of cluster to 10.
Step 6: Maximum iteration is set to 100
Step 7: Set the value of epsilon to 0.2 of accuracy.
Step 8: Set the number of attempts to 10 and Initialize the centroid.
Step 9: Convert the centroids to integer type.
Step 10: Flatten the labels to one dimension.
Step 11: Copy the original image and resize to (1 X 3) array.

Step 12: Select the cluster number based on the centroid value that is closest to the training centroid of ROI.
Step 13: Mask the image copy based on the region of interest.
Step 14: Find the target object based upon the largest contour from the masked image.
Step 15: Repeat the steps for each type of objects.
Step 16: End

The training was done using the training image dataset provided for each object. The centroid for each object cluster was stored. Then the testing was executed on 100 images of each object type. The resulting clusters of testing phase were compared to the training cluster centroid and the closest cluster was selected from which the largest contour was detected with the bounding box as our target object.

3. Results and Discussion

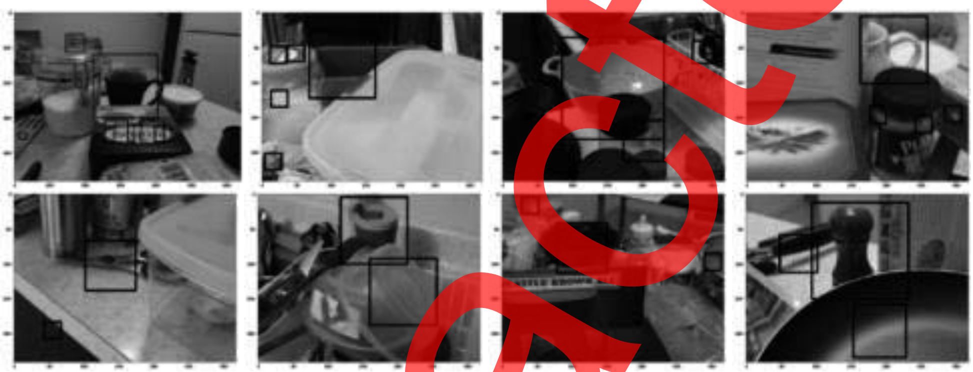

Figure 4. Occluded object detection by Haar Cascade Classifier.

In the above cases (Figure 4) Haar cascade model's performance is demonstrated. The test images were provided as an input to the trained model for each object. It detects the object but, also gives false positives along with the true positives. This indicates that the average recall of the algorithm is very low due to many false positives.

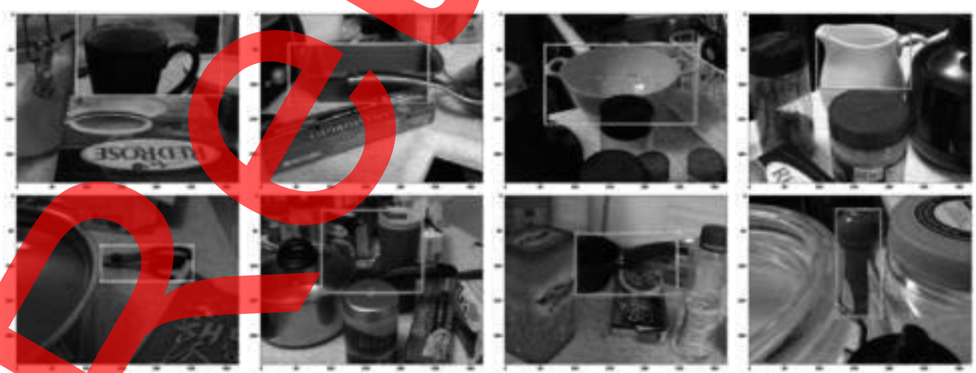

Figure 5. Occluded object detection by K-means segmentation.

In the above cases (Figure 5), the K-means algorithm's performance is demonstrated. It detects the objects successfully based on their RGB combinations as we are implementing color based tracking instead of feature based detection. The algorithm has been designed to detect one object at a time.

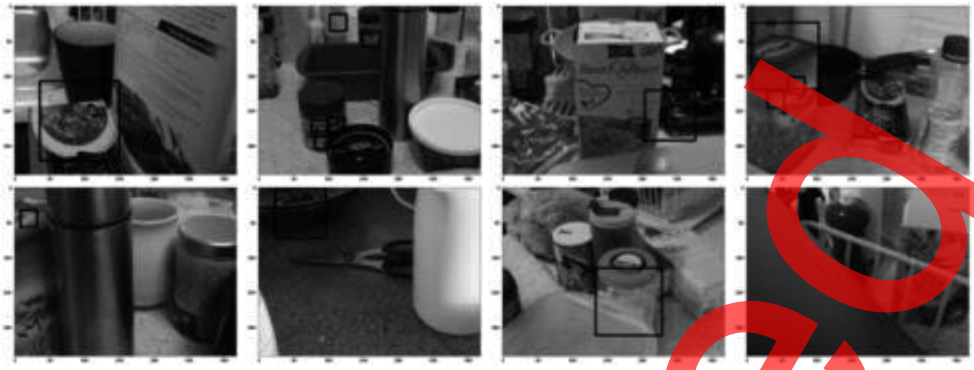

Figure 6. Failure of Haar Cascade Classifiers.

In the above cases, (figure 6) our model has failed to detect the target object due to presence of heavy occlusion that prevents the model from extracting and analysing the features properly. The model also seems to be confusing other objects with our desired target object due to similarity in their features.

Figure 7. Failure of K-means algorithm

The above cases (Figure 7) demonstrate the failure of K-means segmentation. The K-means algorithm highly operates on the basis of centroid values and if the background or occlusion fall in the same cluster as the object, it shall be detected as the object.

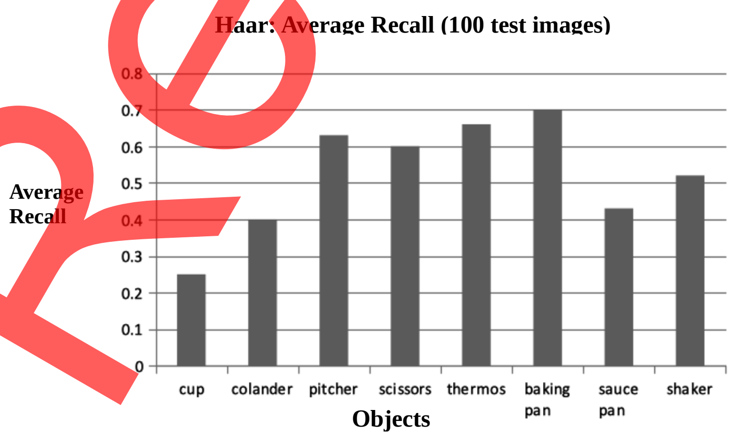

Figure 8: Average Recall of Haar Cascade for 100 test images of each object type.

IOP Conf. Series: Materials Science and Engineering 1145 (2021) 012043 doi:10.1088/1757-899X/1145/1/012043

Figure 8 shows that the Recall of the Haar classifier lies between 25% and 70% which is quite low. This is due to the detection of many false positives along with the true positives.

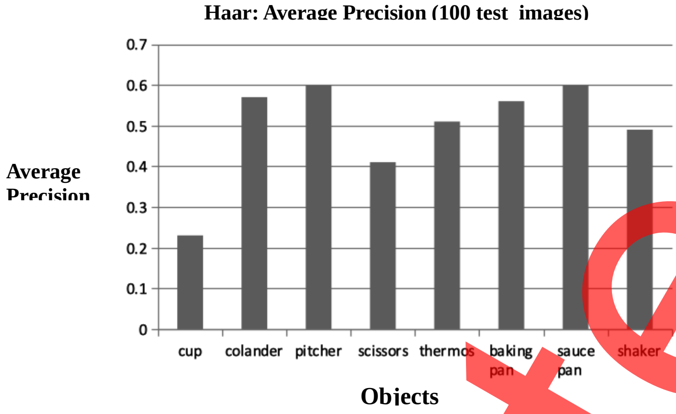

Figure 9. Average Precision of Haar Cascade for 100 test images of each object type.

The Precision values from Figure 9, range from 24% to 60% again which is low due its failure in detecting objects under heavy occlusion.

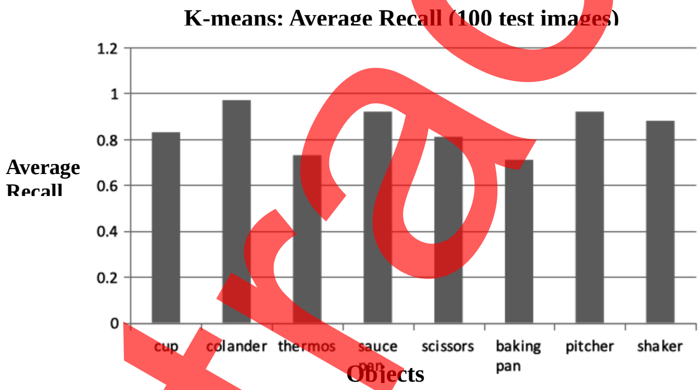

Figure 10. Average recall of K-means Segmentation for 100 test images of each object type.

The Recall of K-means algorithm as seen from Figure 10, lies between 70% and 98% which is quite high and is due to detection done on the basis of color values.

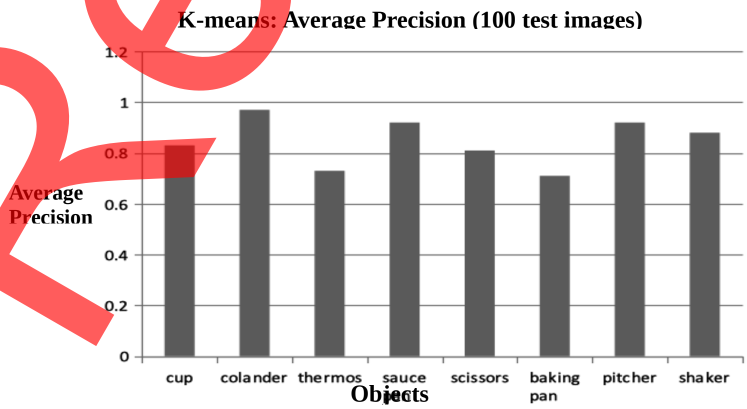

Figure 11. Average recall of K-means Segmentation for 100 test images of each object type

The precision graph, from Figure 11, also lies between 70% and 98% and the precision is same as recall because our algorithm is designed to detect only one object at a time as a result of which the detected object is either true positive or a false positive.

Table 1. Execution time of models

Model	Time (in min)
Haar Cascade average training time	59.37
Haar Cascade average execution time on test data	0.03
K-means average execution time on test data	0.65

Haar Cascade Classifer [13] detects objects based on Haar-like-features (digital image features). It is primarily known for its high precision and recall compared to Histogram of Oriented Gradients (HOG) features [14]. On the other hand, it is known that the Convolutional Neural Network (CNN) [15] fails to detect small objects. Therefore, Haar Cascade Classifier is utilized for the detection of occluded objects.

Image segmentation is the method of partitioning the images into multiple set of clusters or segments. This technique is used to simplify the image representation [16] so that it can be used for further analysis. It is a common technique used to identify objects and object boundaries. There are several methods that can be used for image segmentation, among which K-means is one of the simple and better algorithms that can be used for color based clustering [17]. K-means is known for its accuracy and simplicity as a segmentation technique. Hence, K-means clustering is used to detect the occluded objects based on color features[18][19].

4. Conclusion

A comprehensive analysis based on detecting partially occluded objets using Haar classifier and K-means clustering has been performed. Keeping in mind the limitations of Haar training (false alarm rates) and the segmentation techniques (over segmentation or under segmentation), the occluded object images are utilized to train and test the model using the respective algorithms. And, the color based segmentation is performed using the K-means clustering. On analyzing the resultant images from figure 4, it is clearly noted that the Haar cascade classifiers yield many false positives along with the true positive and sometimes even failed to detect the actual object under heavy occlusion (figure 6). On the other hand for color based segmentation, it is observed from Figure 5, that it accurately detects the actual positive inspite of heavy occlusion. The color based method is however slow as compared to Haar cascade method.

It is observed that the average precision of Haar cascade in detecting occluded objects ranges from 0.24 to 0.60 and average recall varies from 0.25 to 0.70. Also, the average recall and average precision varies from 0.70 to 0.98 for the k-means segmentation of the occluded object. And it is noted that the color based detection algorithm detects only one object at a time based on the contour color and area. Thus the detected object may either be a true positive or a false positive which results in precision being equal to the recall.

In conclusion, from the above results Haar cascade classification works well when the object is not occluded. However, it fails to detect the occluded objects in several cases as it can be seen in figure 6. In future work, this may be improvised. On the other hand, color based segmentation gives good results however; it fails to yield on certain conditions. The desired results are not been obtained when occluded or when the objects have the same color and shades, as it is observed in figure 7. Also, it is noted that when the surrounding objects and target object has the same color, it detects them along with the target object. The Haar cascade however executes much faster than K-means algorithm after training as it is shown in Table 1. K-means on the other hand need no prior training as it is an unsupervised learning technique, it only matches the cluster centroid with training object's centroid value. In future, color-based segmentation can be combined along with Haar cascade or other feature based detection architectures to increase the accuracy of detection in any surrounding or amount of occlusion.

References

[1] Szegedy C, Toshev A and Erhan D 2013 Deep neural networks for object detection. In *Advances in neural information processing systems* 2553-61.
[2] Gupta S, Girshick R, Arbeláez P and Malik J 2014, September Learning rich features from RGB-D images for object detection and segmentation. In *European conference on computer vision* 345-360. Springer, Cham.
[3] Papageorgiou C P, Oren M and Poggio T 1998, January A general framework for object detection. In *Sixth International Conference on Computer Vision (IEEE Cat. No. 98CH36271)* 555-562. IEEE.
[4] Ferrari V, Jurie F and Schmid C, 2010 From images to shape models for object detection. *International journal of computer vision* **87**(3) 284-303.
[5] Li Y, Hou X, Koch C, Rehg J M and Yuille A L, 2014 The secrets of salient object segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* 280-287.
[6] Raja Y, McKenna S J and Gong S 1998, January Segmentation and tracking using color mixture models. In *Asian Conference on Computer Vision* 607-614. Springer, Berlin, Heidelberg.
[7] Pulkit Sharma. 2019 *Object Detection Image segmentation* [Photograph]. Medium.https://miro.medium.com/max/540/0*VX1UNfrNgjRD24Kq.jpg
[8] Hsiao E and Hebert M 2014 Occlusion reasoning for object detection under arbitrary viewpoint. *IEEE transactions on pattern analysis and machine intelligence* **36**(9) 1803-15.
[9] Viola P and Jones M 2001, December Rapid object detection using a boosted cascade of simple features. In *Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001* Vol. **1** I-I. IEEE.
[10] Python Programming. "Creating your own Haar Cascade OpenCV Python Tutorial". Python Programming. Accessed January 5, 2020.https://pythonprogramming.net/haar-cascade-object-detection-python-opencv-tutorial/.
[11] Sarkar A and Kumar R 2017 Study of Various Image Segmentation Methodologies: An Overview. In *Applied Video Processing in Surveillance and Monitoring Systems* 1-27..
[12] Dhanachandra N, Manglem K and Chanu Y J 2015 Image segmentation using K-means clustering algorithm and subtractive clustering algorithm. *Procedia Computer Science* **54** 764-771.
[13] Haldorai, A. Ramu, and S. Murugan, Social Aware Cognitive Radio Networks, Social Network Analytics for Contemporary Business Organizations, pp. 188–202. doi:10.4018/978-1-5225-5097-6.ch010
[14] R. Arulmurugan and H. Anandakumar, Region-based seed point cell segmentation and detection for biomedical image analysis, International Journal of Biomedical Engineering and Technology, vol. **27**, no. 4, p. 273, 2018..
[15] Iscen A, Tolias G, Avrithis Y, Furon T and Chum 2017 Efficient diffusion on region manifolds: Recovering small objects with compact cnn representations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* 2077-86.

[16] Haralick R M and Shapiro L G 1985 Image segmentation techniques. *Computer vision, graphics, and image processing* **29**(1) 100-132.
[17] Ray S and Turi R H 1999, December Determination of number of clusters in k-means clustering and application in color image segmentation. In *Proceedings of the 4th international conference on advances in pattern recognition and digital techniques* 137-143.
[18] Lienhart R and Maydt J 2002, September An extended set of haar-like features for rapid object detection. In *Proceedings International conference on image processing* Vol. **1** I-I. IEEE.
[19] Liu D and Xia F 2010 Assessing object-based classification: advantages and limitations. *Remote Sensing Letters* **1**(4) 187-194.