İZMİR KÂTİP ÇELEBİ ÜNİVERSİTESİ
2010
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

Traffic Sign Detection

EGE ÖZBENDER – Y210234094

AYTUĞ ONAN

DATE: 18/06/2023

ABSTRACT

The goal of this project is to create a system that can identify a road sign's location in a picture and categorize it. We created functions to reduce dynamic range, modify image resolution, and convert RGB to grayscale for this purpose. Based on the outcomes of various clustering techniques, we chose to use the K-Means algorithm with five clusters. Additionally, 32x32 resolution photos have been selected as acceptable. A bounding box is created around the sign after selecting the appropriate structure to use. We created the code to operate on every input image because these procedures were initially performed on a single test image. Finally, by inheriting the feature extraction portion of VGG19, a transfer learning model has been created.

TABLE OF CONTENTS

ABBREVIATIONS

IEEE : Institute of Electrical and Electronics Engineers

MoM : Method of Moments

TABLE OF FIGURES

1. INTRODUCTION

These days, AI and machines are ingrained in one another. Additionally, the circumstance has an impact on the vehicles that people use for transition, particularly cars. Intriguing AI technologies like self-driving systems, active collision preventers, road assistants, and user habit learning systems in automobiles are being developed by the largest car provider firms, which have been working on this issue for years. In that situation, we sought to create a comparable system that might work well in those systems. So, we began to create a system that could function as a traffic sign detector in those systems. In that approach, a system for self-driving car applications might be developed further. We need to figure out how to apply dynamic range reduction (DRR), how to convert RGB images to grayscale, how to make bounding boxes for traffic signs in images with roads, and how to recognize the signs, among other problems, for this project. We used a step-by-step approach to solve those issues, starting with the conversion of images to greyscale, followed by the calculation of entropy, the application of dynamic range reduction, the generation of images with various pixel values, the development and selection of the most optimized algorithm, the generation of bounding boxes, the generalization of the sign detection process, and finally the classification of the contents of road and traffic signs using transfer learning.

2. PROBLEM DEFINITON

Numerous issues have arisen as a result of our work on the traffic sign detection project. Converting RGB images to grayscale, obtaining information from entropy for dynamic range reduction, altering image dimensions for testing, identifying specific grayscale levels for dynamic range reduction, selecting the best clustering algorithm to use, creating a bounding box around the road sign, generalizing all of the previous operations to make this project work on any image, and categorizing the road sign are the main problems that could be categorized.

2.1. RGB to Gray Scale Conversion

Red, Green, and Blue, or RGB, are the primary colors in the image. These primary colors serve as the foundation for all additional hues. Each channel in a three-channel matrix used to depict an RGB image is a 2D array. These numbers range from 0 to 255. A grayscale image, in contrast to RGB images, is only represented by a single channel 2D array. It is recommended to conduct channel reduction when converting RGB photos to grayscale.

2.2. Using Entropy Information

Entropy is a physically quantifiable property that is frequently associated with chaos, unpredictable behavior, or uncertainty [1]. An algorithm was required for this project in order to generalize the entire procedure and make it applicable to every traffic sign. The image's entropy value was employed for this process, and it is important to utilise this information properly.

2.3. Dynamic Range Reduction

A process known as dynamic range reduction modifies an image's pixel values to a set of specified pixel levels. In order to improve the effectiveness of clustering algorithms for locating the road sign on the image, dynaic range reduction has been required to lower the complexity of the image.

2.4. Changing Images’ Dimensions

By utilizing a technique to cut the number of pixels or interpolate the missing pixels, changing image dimensions means replicating the present image with different dimension values. Various methods have been created or modified during the process in accordance with the requirements and objectives of the project. This transformation is required to generate the same-sized images for deep learning model training and to evaluate the performance of DRR.

2.5. Choosing Proper Algorithm for Clustering

The technique of grouping a set of elements so that those in the same group are more similar than those in other groups is known as clustering, often referred to as cluster analysis [2]. Additionally, as clustering is an unsupervised machine learning approach, this project establishes clusters by examining the grayscale values of each pixel. Out of all the applied algorithms, the right algorithm should be picked.

2.6. Generating Bounding Box

For the purpose of object detection, a bounding box is a hypothetical rectangle that also creates a collision box for the item [3]. According to the definition provided, a bounding box has been used in this project to indicate where the road sign is located within the image. One of the key objectives of the traffic sign detecting project is to assess the correctness of the selected class, making this challenge one of the most crucial problems.

2.7. Generalizing All Performed Operations

Figure 1: Test image.

Since the project's objective is to locate each road sign on the image, every component that functions should be updated to reflect that it does so without external interference. Finding the amount of pixel levels to perform DRR and class determination to generate a bounding box should both be automated for generalizing all of the operations.

2.8. Classifying Road Signs

First, since the starting data collection only has 7 photos, a better data set should be identified. A deep learning model was then selected to classify road signs. Before building and altering the model, input photos need to be scaled in order for the deep learning method to function. Images should be flattened after adjusting their size in order to separate them into train and test data. Finally, the right algorithm should be selected.

3. PROPOSED SOLUTION

3.1. Solution of RGB to Gray Scale Conversion

As was previously said, switching a picture from RGB to grayscale simply entails altering the matrix size of the image. Various methods have been tested for grayscale image conversion. In the beginning, a function that already existed was employed. Resize is a cv2 library function that is used in this function. Writing our own codes to solve the challenges was one of our primary objectives when we started this project. So, a custom converter function was created to achieve our purpose. A new list has first been formed specifically for the conversion of photos, and the conversion procedure is then carried out using equation 1.

$$Gray\ Scale\ Pixel = R * 0.2126 + G * 0.7152 + B * 0.0722 \text{ [4]} \qquad (1)$$

In formula 1, R, G, and B stand in for the values of the red, green, and blue channels, respectively. When pixel values were multiplied by the provided normalized coefficients and added, a grayscale value ranging from 0 to 255 was produced. A dimension reduction operation has been used in this operation.

3.2. Making Sense of Entropy Value

Entropy is a measure of the data's randomness, hence it has been attempted to calculate the number of color levels by computing the entropy value. The test image's entropy value was first calculated, and then the image's pixel values were plotted as shown in figure 2 to examine the link between pixel values and entropy value.

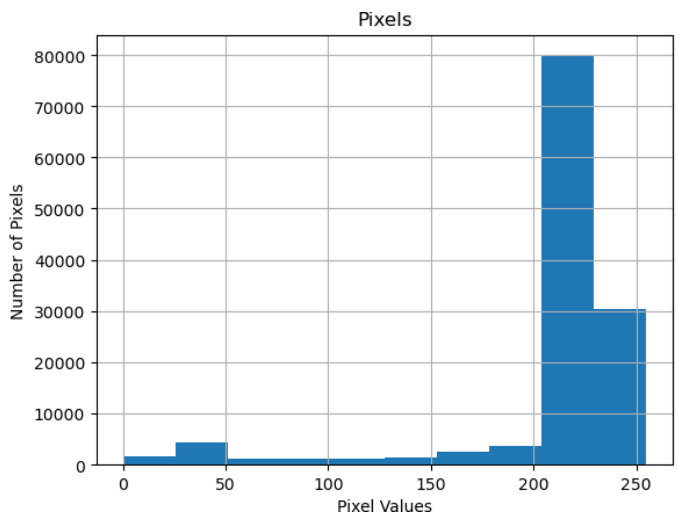


Figure 2: Histogram of pixel value distribution.

The association between 1/ entropy and the number of peak values in the pixel distribution has been noticed after examining the relationship between the pixel value distribution and entropy value. We first chose the pixel values to reduce and the pixel values to reduce, however in the generalizing phase, DRR levels have been decided by utilizing 1 / entropy.

3.3. Dynamic Range Reduction

The goal of dynamic range reduction is to lower the number of distinctive pixel values in the image matrix. Initially created to function on a test image, then generalized to function on all images during the generalization of the entire process section.

By examining the histogram of pixel distribution, some specific values have been chosen when working on the test image. The chosen numbers are 40, 160, 190, 225, and 245. Rounding is done to the closest pixel level for the remaining pixel values.

3.4. Image Dimension Change

For testing the performance of the clustering algorithm under various settings, picture dimension change components have been built to produce the identical DRR produced images with varied resolutions, which are 16x16, 32x32, 64x64, and 128x128. Three alternative algorithms for altering image dimensions have been examined. In order to achieve the purpose of our custom code, various algorithms have been devised, and the first method evaluated is one from CV2.

A moving average filter was used in the second algorithm. The Gaussian filter has been applied as a filter. first multiplying the picture matrix by a matrix factor of two and then splitting the output into 16 equal halves.

$$New\ Pixel\ Value = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \cdot \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} * \frac{1}{16} \tag{2}$$

However, because of the float results produced by this process, we now have new, distinct pixel values. Additionally, relocating the average filter makes it more susceptible to unexpectedly noisy values. A different algorithm has so been tested. The moving median filter is the third and final algorithm that has been evaluated. The pixel value is replaced by two steps using the median filter. The surrounding pixel values were sorted in the first stage, and the value in the middle was then selected as the output value. Last but not least, the moving median filter has been used because it performs better against noise than the moving average filter. The entire body of our written code is another benefit of choosing.

3.5. Choosing The Best Clustering Algorithm

An edge detection approach was initially considered to function instead of a clustering algorithm at the beginning of the project. Clustering algorithms have been chosen to deal with since edge detection approaches have difficulty recognizing sharp edges and rectangle shapes. In this study, two alternative clustering techniques were employed. The K-Means method from the scikit-learn toolkit was the first algorithm used.

We used the code directly because we were unable to write it ourselves. The connected region algorithm, which we developed, is the second algorithm. Starting in the image's left-top corner, this method scans pixels all the way down to the right-bottom corner. various identically valued pixels take on distinct cluster values due to various skips. The identical procedure has been used twice more in an attempt to solve this issue. An organized image is what is ultimately left. K-Means have been chosen to work with because of some potential issues like various colored sections on the image. We conducted tests after determining the ideal algorithm. The effectiveness of the algorithms used for testing from 1-Mean to 10-Means was examined. The original image was used for the initial tests during the testing phase, however the results could not locate the place on the sign. Because of that same test was done on the images with different dimensions and the best result was observed in 32x32 sized images with the 5-Means algorithm.

3.6. Generating Bounding Box

Before generalizing earlier procedures, bounding box generation comes last. In this section, the sign clusters were manually assigned; however, in the generalization, the selection of the appropriate cluster is automated. The edge coordinates of the selected cluster have been gathered in order to create the bounding box around it. The corner values are then surrounded by lines.

3.7. Generalizing Operations

The majority of the actions that are carried out on the test image can be applied to every image that is provided to the system. The dynamic range reduction and producing bounding box sections underwent some unique adjustments. When deciding how many levels to reduce and how much to reduce each level by, the dynamic range reduction is altered. As previously stated, the entropy value is utilized to calculate the number of levels. The entropy function's output ranges from 0 to 1, hence the number of levels is specified by 1/entropy. The values of these levels should be determined automatically as the generalization of the process has eliminated the user's external impact. To get the step size, divide the range of values from 0 to 255 by the entropy value directly. By selecting the appropriate cluster, the bounding box generation can be made more universal. Our concept is designed to work with photographs where the road sign is the most prominent element. As a result, we calculated the road sign cluster's proportion relative to the total number of pixels in the image. The threshold values of greater than 20% and less than 30% have been chosen.

3.8. Classifying Road Signs

Since the starting data collection only had 7 photos, we have to find a fresh data set before identifying road signs. We evaluated Kaggle's data sets initially, but they weren't sufficient for our needs, so we moved on to Google. We finally obtained the TSRD data set. There are 58 distinct classifications of Chinese traffic signs in this data set.

Preprocessing has begun after locating the appropriate data set. A resizing process is carried out once again in this section. But this time, some of the photos should have also undergone interpolation. Therefore, we were unable to use the selected algorithm in the picture dimension modification section. The cv2 library's resize function has been selected to work with since it is a faster and more prepared solution given the more challenging nature of the challenge and the data set's around 6000 photos.

Finally, we began developing a deep learning model for categorizing traffic signs. A shallow model has initially been tested. As was to be expected, the results were difficult to use, thus a 14 deeper model was developed. Once more, the outcomes were disappointing. A model for transfer learning has been created. Figure 3 illustrates how the feature extractor, VGG19, was employed.

```
Model: "CNN"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 input_1 (InputLayer)        [(None, 224, 224, 3)]     0

 vgg19 (Functional)          (None, 7, 7, 512)         20024384

 reshape (Reshape)           (None, 1, 25088)          0

 flatten (Flatten)           (None, 25088)             0

=================================================================
Total params: 20,024,384
Trainable params: 0
Non-trainable params: 20,024,384
_________________________________________________________________
None
```

Figure 3: Feature extraction model.

dense layers added by us to classify the output number of classes, which is shown in the figure 4.

```
Model: "Classifier-Model"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 input_3 (InputLayer)        [(None, None, 25088)]     0

 dense (Dense)               (None, None, 512)         12845568

 dense_1 (Dense)             (None, None, 128)         65664

 dense_2 (Dense)             (None, None, 58)          7482

=================================================================
Total params: 12,918,714
Trainable params: 12,918,714
Non-trainable params: 0
_________________________________________________________________
None
```

Figure 4: Classification model.

These two models are gathered under the complete model.

```
Model: "Complete-Model"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 input_4 (InputLayer)        [(None, 224, 224, 3)]     0

 CNN (Functional)            (None, 25088)             20024384

 Classifier-Model (Functiona  (None, None, 58)         12918714
 l)

=================================================================
Total params: 32,943,098
Trainable params: 12,918,714
Non-trainable params: 20,024,384
_________________________________________________________________
None
```

Figure 5: The complete model.

Number of layers of the dense layers determined by performing different structures. Final structure chosen as 2 dense layers with ReLU activation function and 1 dense layer with softmax activation function with 58-bit output.

4. RESULTS AND DISCUSSIONS

The test image, which is depicted in figure 6, was used for the initial experiments.

First, the RGB to grayscale conversion was done using equation 1. Figure 7 displays the resulting image.

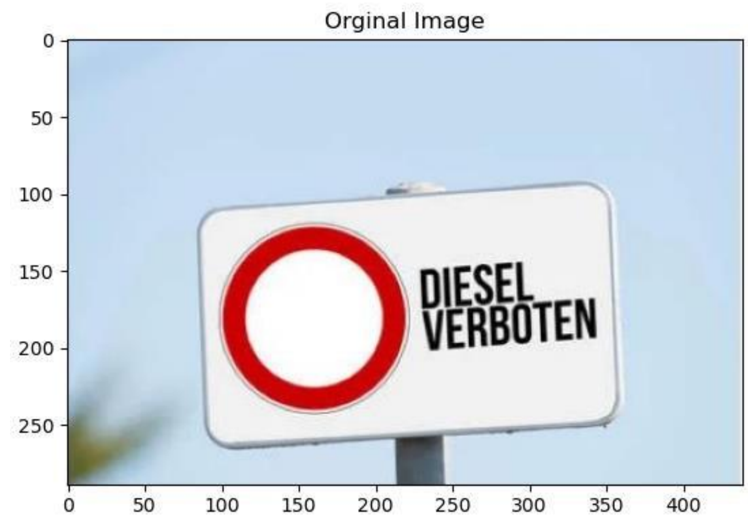


Figure 6: Original test image.

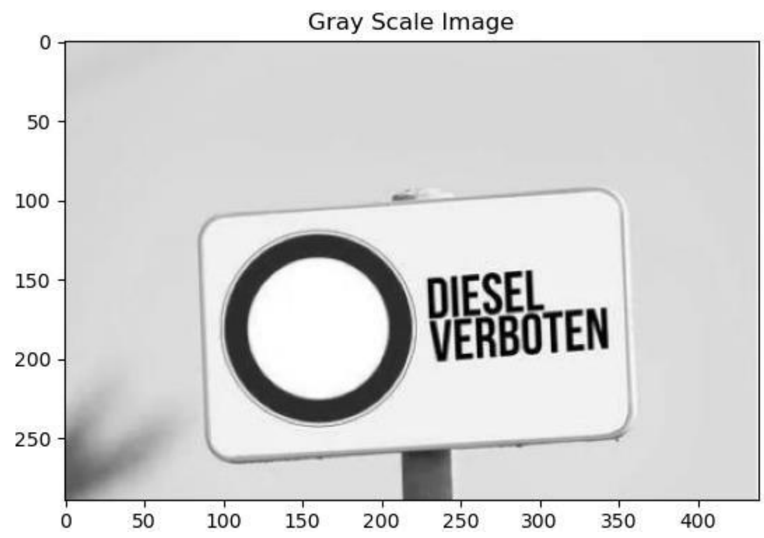


Figure 7: Gray scale converted version of test image.

Dynamic range reduction was done to the test image after the RGB image was converted to grayscale. In order to narrow the range of pixel values, 5 levels—40, 160, 190, 225, and 245—have been selected. Figure 8 displays the end outcome.

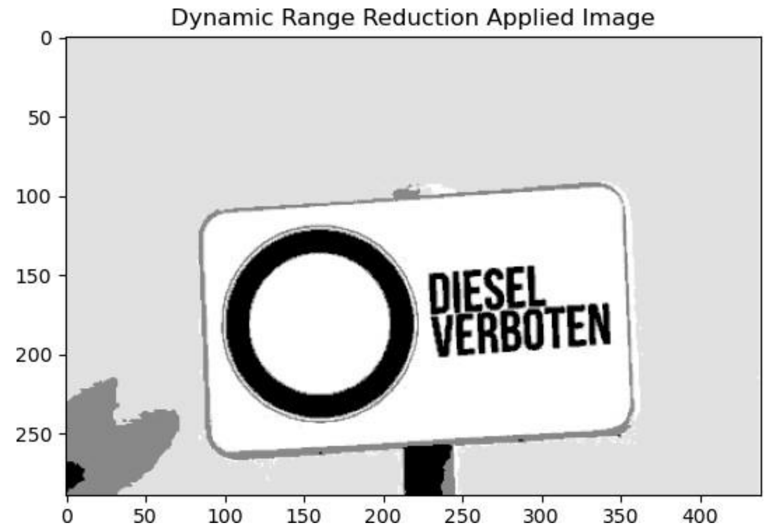


Figure 8: DRR applied test image.

Figures 8 and 7 are identical, with the exception that alterations in figure 8 occurred in shadowy areas. This stage improves the development of our image so that clustering algorithms can work with it. Dimension change techniques have been used after the number of unique pixels has been decreased. Figures 9 through 12 display the outcomes.

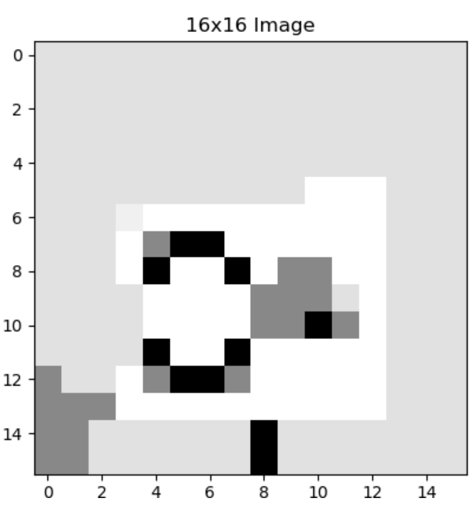


Figure 9: 16x16 DRR applied test image.

Figure 10: 32x32 DRR applied test image.

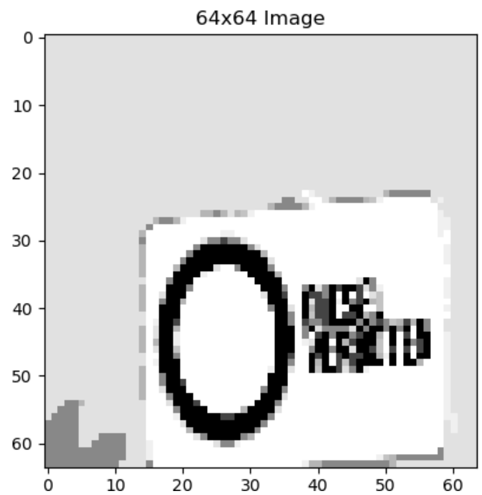


Figure 11: 64x64 DRR applied test image.

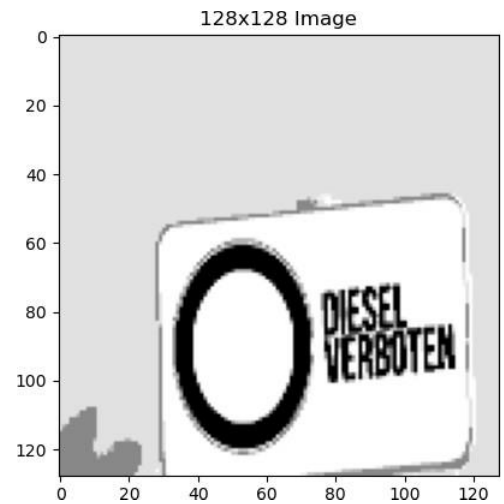


Figure 12: 128x128 DRR applied test image.

More logical visuals have been seen as expected when image resolution has increased. After 64x64 photos, mistakes began to appear due to the algorithm being employed, which causes the signs to be larger than the original image. This is one another compelling argument in favor of using the 32x32 picture when applying clustering methods. Tests on the clustering have begun after photos were prepared for evaluating clustering algorithms under various scenarios. First tests are carried out on the DRR-applied test image with the original resolution, as was already stated in the previous section. The 7- Means algorithm has produced the best results. The 5-Means and 7-Means performances are displayed in figures 13 and 14, respectively.

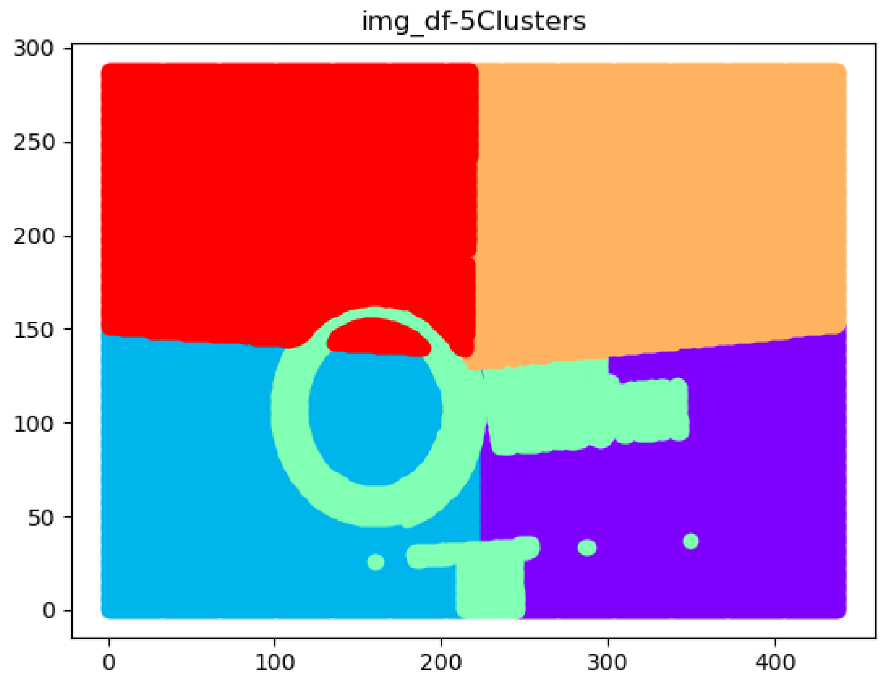


Figure 13: 5-Means result on test image.

Figure 14: 7-Means result on test image.

The best result was seen at 32x32 resolution with 5-Means since this technique is insufficient to find the placement of the sign at that resolution.

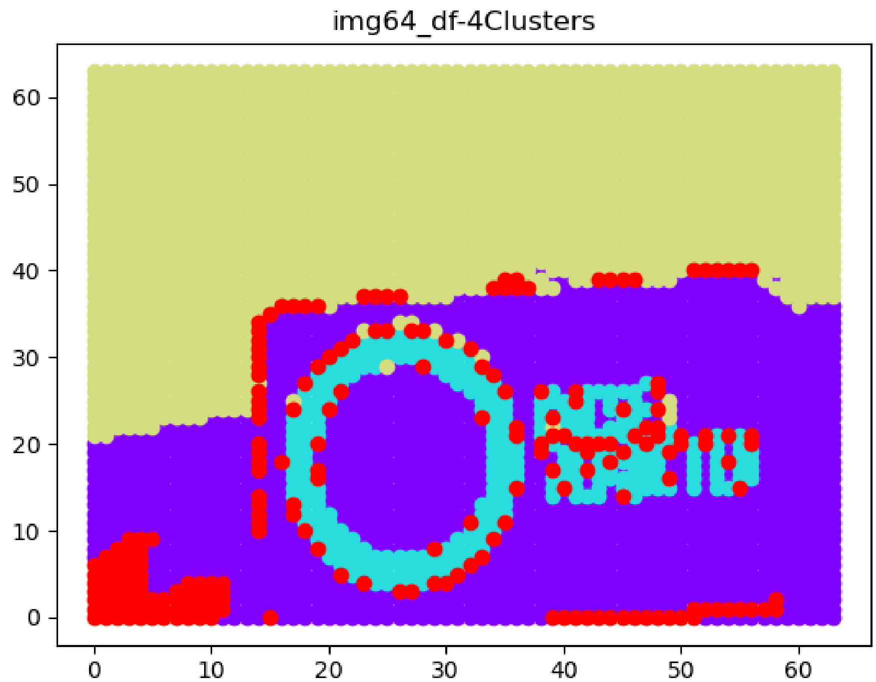


Figure 15: 64x64 resolution with 4-Means.

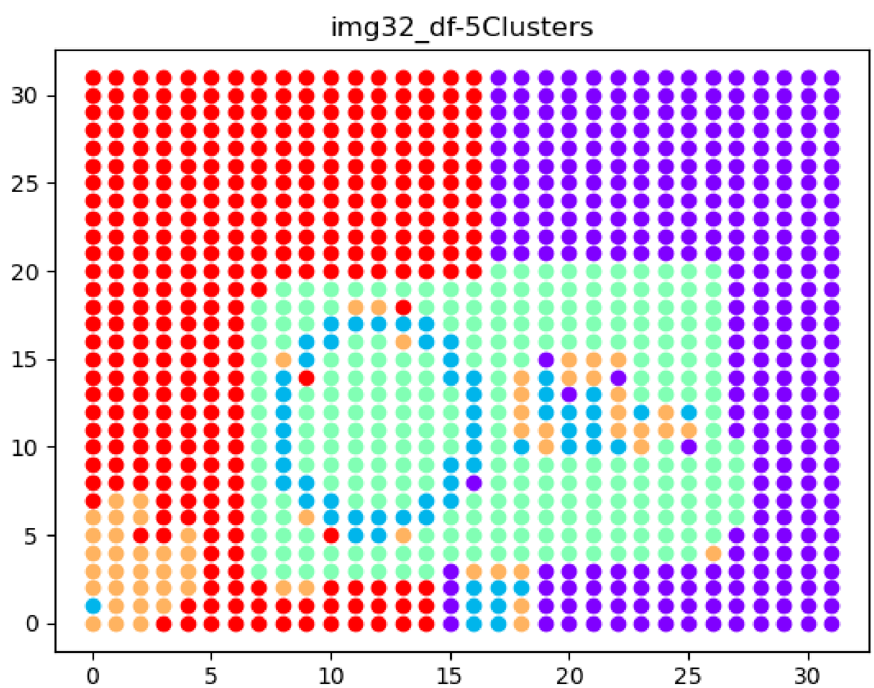
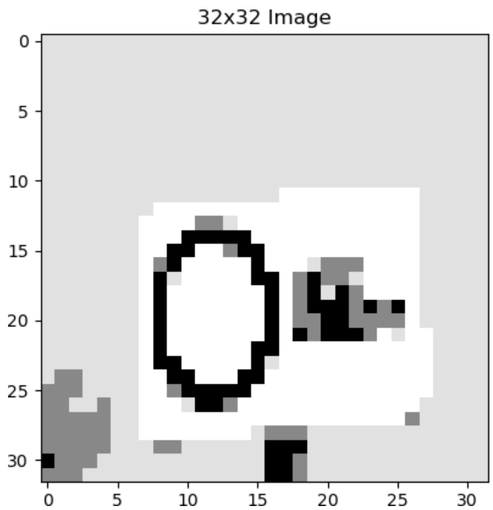


Figure 16: 32x32 resolution with 5-Means.

Figure 16 illustrates how the sign cluster may be clearly distinguished from other classes. A bounding box is created around the road sign cluster when it has been located. Figures 17 and 18 show the image without a bounding box and the figure 17 with the produced bounding box, respectively



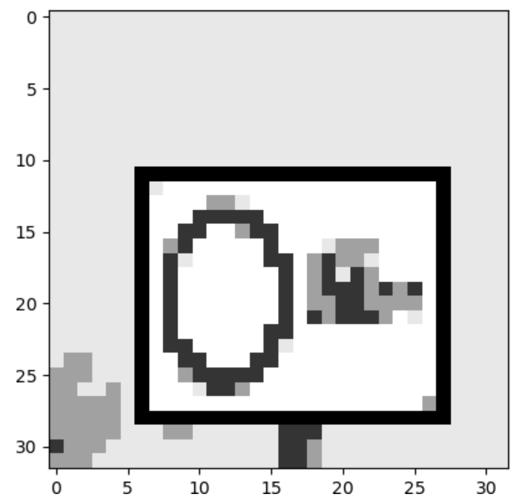


Figure 17: DRR applied test image with 32x32 resolution. **Figure 18:** Bounding box generated version.

The operation of generalization has also begun. As previously mentioned in section 3.7, the first DRR operation is made more broad by utilizing the image's entropy. The test image in 32x32 resolution has been exhibited in figures 19 and 20, respectively, with differences between generalized DRR and not generalized DRR.

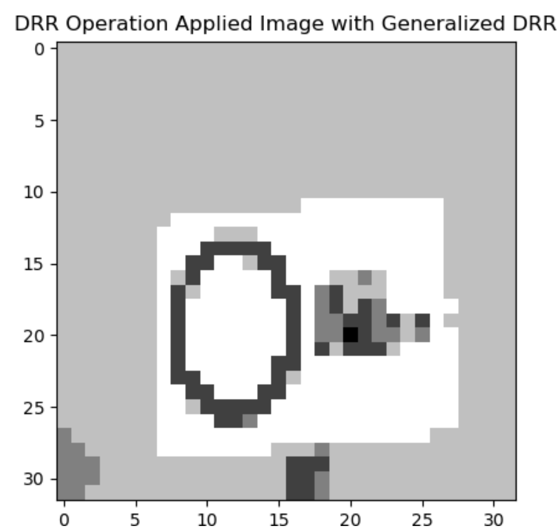


Figure 20: Generalized DRR applied test image.

Figure 19: Non-generalized DRR applied test image.

Select the appropriate cluster using the percentage completed after completing DRR generalization. The cluster is selected, as previously said, when the percentage of the cluster is larger than 20% and less than 30%. A created bounding box for the sign is displayed in figure 21.

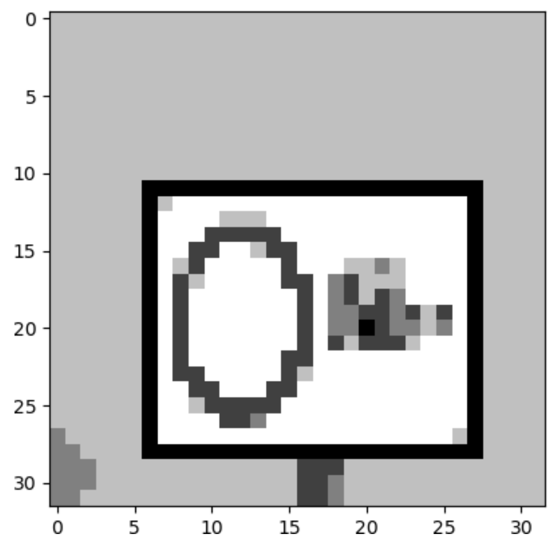


Figure 21: Generalized bounding box generated image.

The performance of the generalization operation can be seen in the following figures.

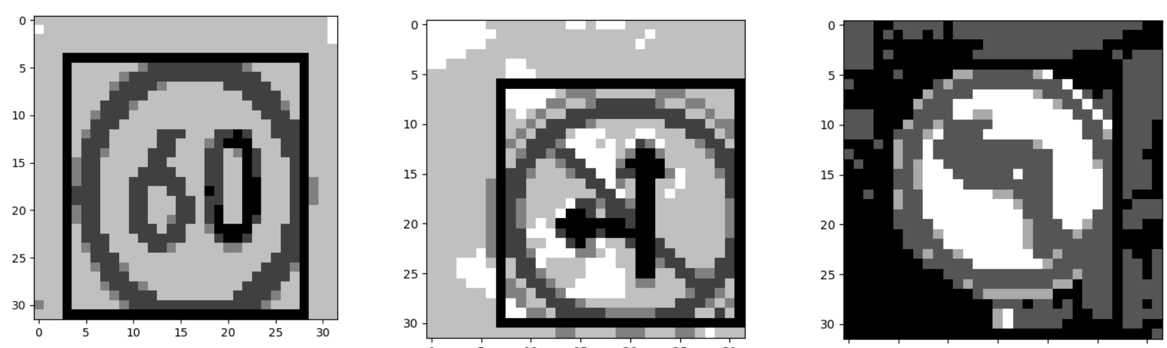
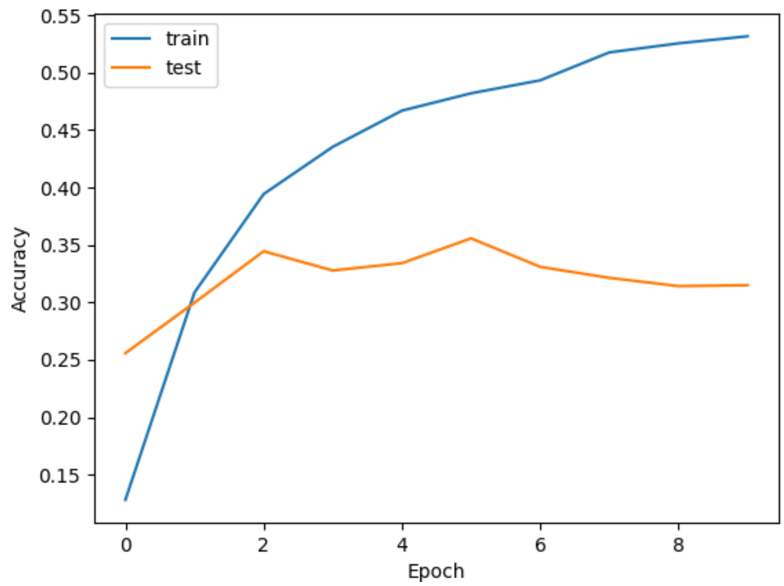


Figure 22,23, and 24: Results of generalized operations applied on test images from other data set.

Finally, a transfer learning model has been developed after fully generalizing all operations. Figure 22 displays the model's loss and accuracy value.



Since our objective is to have an accuracy value of about 90%, the results of the used model were not satisfactory. However, given the large number of classes in the data set, this is an expected outcome.

5. CONCLUSIONS

Our initial objective is to create a system that might be used in self-driving cars to identify the information and locate the road sign. We had to deal with a wide range of issues in order to design this kind of system. The first issue we encountered was converting RGB photos to grayscale. Various techniques for fixing this issue were evaluated, and in the end, we decided to work by multiplying equations with normalized coefficients, and by adding them, we were able to produce a grayscale image. Understanding entropy value presented the second challenge. This challenge was successfully done by understanding the relationship between 1/entropy and the number of levels to lower the unique pixel values. Following that, we began to work on DRR to develop a suitable working space for clustering methods. We performed this task manually at first, but in the succeeding steps, it was automated. Another approach, called the dimension changer algorithm, has been developed for producing additional images to test clustering algorithms after running DRR on the test image. A moving median filter has been employed to alter the image's dimensions. The sizes of the new photos are 16 x 16, 32 x 32, 64 x 64, and 128 x 128. Performance of clustering techniques has been evaluated using newly produced photos. As a result, we decided to use the 5-Means model with 32x32 pixel images. Then, a bounding box is drawn around the selected right cluster, which is the cluster of the sign. After completing each of these steps on a single test image, the process is generalized to work with all images by using the entropy value to determine the number of levels of DRR operation and by examining the percentage of the clusters of road signs. Using this knowledge, a bounding box is created around the sign, and the number of clusters has been determined. The classification of photos has also been done using a transfer learning model feature extractor components that came from VGG19 and classifying components that we created.

REFERENCES

[1] Einstein, A., Podolsky, B., & Rosen, N. (1935). Can quantum-mechanical description of physical reality be considered complete?. *Physical review*, *47*(10), 777.

[2] Cover, T. M., & Thomas, J. A. (2012). *Elements of information theory*. John Wiley & Sons.

[3] Grossglauser, M., & Tse, D. (2001). Mobility increases the capacity of ad-hoc wireless networks. In INFOCOM 2001. *Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies*. Proceedings. (Vol. 3, pp. 1360-1369).

APPENDIX