Color Classification and Recycling Bin Detection

Jiayang Zhou
Department of Electrical and Computer Engineering
University of California, San Diego
La Jolla, U.S.A.
jiz141@ucsd.edu

Nikolay Atanasov
Department of Electrical and Computer Engineering
University of California, San Diego
La Jolla, U.S.A.
natanasov@eng.ucsd.edu

***Abstract*—This paper focuses on using the image pixel classification methods and image reinforcement methods to segment images and then detected the position of the bin in different images. As bins may appear in different places, it is necessary to design a classification algorithm that can segment the bin from a complex scenario. My method achieved a reliable outcome on detecting the location of bins in different scenarios.**

Keywords—Logistic Linear Regression, Roipoly, Morphologic Operation

I. Introduction

With the advent of the era of automation and intelligence, people are trying to make everything in their life easier and faster with the help of machines and robotics.

Nowadays, with the increasing population and the development of industrialization, people are making garbage much more than ever. Therefore, developing a robust algorithm for machines and robotics to detect the position of bins in life will help the automatic process of recycling bins and make people's life easier.

In this article, the main approach is to use a pixel classification model to segment the bins in the image and detect the position of the trash bin in the image based on the segmented image generated before.

The training sample is gathered manually by using the Roipoly method to select and filter local features in the image and manually label those areas. Based on the training sample, train a logistic linear regression model to classify each pixel in a real image to get the segmented image.

With the segmented Image, the complex image gets simplified to only a few classes of color. Based on the simplified Image and some image reinforcement techniques, the segmented image can be transferred to a larger area with higher contrast. Then detect the position of the bin in the image by computing the ratio between length and width of the bounding box and the saturation level of the blue pixel.

II. Problem Formulation

A. *Logistic Regression – pixel classification*

First, we need to deal with the fundamental issue, which is to classify the pixel in the image. For the fundamental task, we only have three classes of pixel, red, green, and blue, which is relatively easy. For the bin Detecting system, there are more than three classes, but still the same solution.

To use logistic regression to classify pixels. First we need to construct the input vector with bias, which is shown as follows:

$$X_{input} = [1 \quad r \quad g \quad b]$$

While we also have to initialize the corresponding weight vector to perform the Gradient Descent Algorithm. The initial weight vector goes as follows:

$$Weight = [\theta_1 \ \theta_2 \ \theta_3 \ \theta_4]$$

The original output is computed as follows:

$$Y = X_{input} * Weigth.T$$

As we are using Logistic Regression here for classification purpose, the initial output has to be transferred to a domain between 0 to 1. Therefore, we use the sigmoid function as the Logistic function.

$$g(z) = \frac{1}{1 + e^{-z}}$$

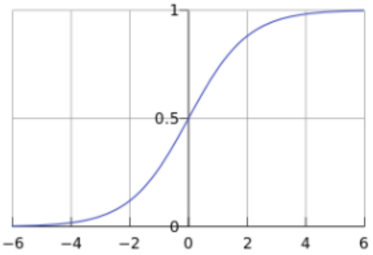

Fig. 1. Logistic Function - sigmoid

After being transferred to a domain between 0 to 1, the output can be labeled as 1, if it is higher than 0.5 and labeled as 0 if below 0.5. So, we have the following distribution:

$$P(y = 1 \mid x; \theta) = h_\theta(x)$$
$$P(y = 0 \mid x; \theta) = 1 - h_\theta(x)$$

$$P(y \mid x; \theta) = h_\theta(x)^y * (1 - h_\theta(x))^{1-y}$$

Gradient Descent

To make the prediction robust, we have to optimize the parameters in the weight vector.

Therefore, based on the maximum likelihood estimation, the basic Loss Function goes as follows:

$$L(\theta) = -\frac{1}{m} * \sum_{i=1}^{i=m} \left(yi * \log\ h_\theta(xi) + (1-yi) * \log\left(1 - h_\theta(xi)\right)\right)$$

But in order to prevent overfitting from happening, we still must add the regularization term. Therefore, the final Loss Function goes as follows;

$$L(\theta) = -\frac{1}{m} * \sum_{i=1}^{i=m} \left(yi * \log h_\theta(xi) + (1-yi) * \log\left(1 - h_\theta(xi)\right)\right) + \frac{1}{2}\sum_{i=1}^{i=m} \theta^2$$

With the Loss Function, we can compute the gradient of weight parameters and update all of them as follows:

$$\frac{\partial L(\theta)}{\partial \theta} = = \frac{1}{m}\sum_{i=1}^{i=m} \left[(h_\theta(xi) - yi) * xi_j\right] + \sum_{i=1}^{i=m} \theta$$

$$\theta = \theta - \alpha * \frac{\partial L(\theta)}{\partial \theta}$$

III. Technical approach

Both tasks are based on the Linear Logistic Regression model.

A. *Pixel Classification Task*

As Linear Logistic Regression model is basically for binary classification tasks. To make the model being able to deal with multi-class tasks. Here we just simply make several binary linear regression classification models.

Each color class corresponds to a binary linear regression model, which classifies themselves from the rest color. By combining several of these kinds of models, it can achieve the goal of multiclass classification task with logistic regression.

With the gradient descent algorithm, the model is trained to become robust in pixel classification after 200 iterations with a learning rate and regularization term both equal to 0.1.

B. *Bin Detection Task*

- First, choose 5 kinds of color for the image segmentation, which are blue, dark green, brown, black, and gray. The reason for choosing such a combination goes as follows. The blue bin always appears on the road, on the grass, on the ground with some bins with other colors. Therefore, it is necessary to distinguish them from their surroundings. Gray and dark can distinguish from the black bin, road, and blue bin. Yellow can distinguish the bin from the ground, while green can distinguish the bin from the grassland. The manually labeling process is fulfilled with the Roipoly method.

- Considering that the light intensity varies and the presence of shadow will also cause some influence on the final prediction of color class. Therefore, the model here transfers the RGB data to HSI first to separate the influence of intensity from other color information, so that the linear regression model can percept that and ignore it easily.

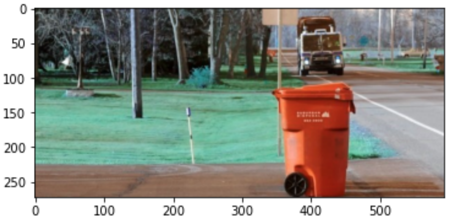

Fig. 2. HIS Image

- After done with the training sample, use the same way with the Pixel Classification tasks to train a classification model to classify the 5 kinds of color to do color segmentation.

- With the trained model, it is easy to segment every pixel in the image into different classes to form the segmented image.

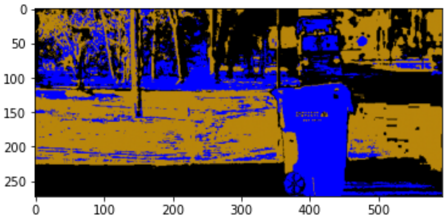

Fig. 3. Segmented Imag Output

- Then, perform erosion and dilation of the segmented image for 1 iteration with kernel size (9,6), because (9,6) is close to the ratio of the bin (length/width) to augment the image.

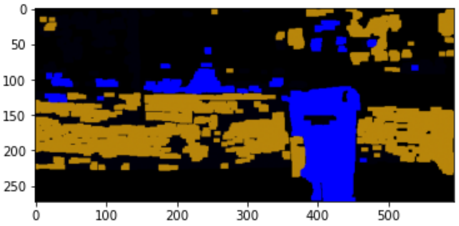

Fig. 4. Segmented Image After Erosion and dilation

- Finally, set up a threshold for area and a boundary for length to width ratio, then find areas whose area is larger than the threshold and length to width ratio is also with the boundary to be the location of bin.

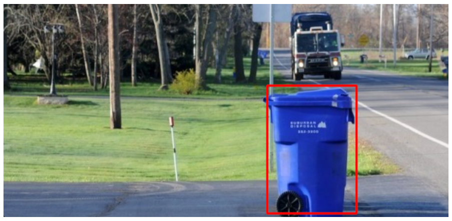

Fig. 5. Detected Bin position

IV. Results

After training the model in both tasks becomes robust, the final results go as follows.

A. Pixel Classification Task

For the RGB classification task, my linear regression model achieved 100% accuracy in the validation set and achieved 99.34% accuracy in the test set on Gradescope.

The weights of my linear regression mode go as follows:

[−1.17227135, 6.38322718, −3.30960737, −3.19249442]
[−0.95560608, −3.63335187, 6.13819129, −3.20317297]
[−0.97357361, −3.56814519, −3.2084153, 6.03092903]

B. Bin Detection Task

For the Bin Detection task, the model achieves 98% accuracy in the test images, the only two wrong classified images do not contain a bin, but the model still detects something like a bin. That may be caused by the limited amount of training samples. In the training process, the model only has five label area in five images, which is not enough and lack diversity.

The final weights of the bin detection model go as follows:

[−19.831939, − 15.19792, 18.2241467, −4.8996555]
[0.96083018, −0.9745223, 0.82823063, −0.6210963]
[0.59994694, −7.9805637, − 2.7002224, 5.29974882]
[0.94091389, 10.6324033, − 2.9779833, −4.1070537]
[0.16690198, −8.0142612, −13.401788, −12.734435]

References

[1] Ng, A.Y. & Jordan, M. I. (2002). On Discriminative vs. Generative Classifiers: A comparison of Logistic Regression and Naive Bayes, *Neural Information Processing Systems*, Ng, A.Y., and Jordan, M. (2002).