

# Project Report

## Garbage Classification Using Deep Learning

**BY**

**Mohammed Marzuk Ali .S**

**Akash Kumar .S**

**Sridhar .S**

**Nikesh .C**

# **INDEX**

1. INTRODUCTION
    - 1.1 Overview
    - 1.2 Purpose
  2. LITERATURE SURVEY
    - 2.1 Existing Problem
    - 2.2 Proposed Solution
  3. THEORITICAL ANALYSIS
    - 3.1 Block Diagram
    - 3.2 Hardware/Software Designing
  4. EXPERIMAENTAL INVESTIGATION
  5. FLOWCHART
  6. RESULTS
  7. ADVANTAGES & DISADVANTAGES
  8. APPLICATIONS
  9. CONCLUSION
  10. FUTURE SCOPE
  11. BIBILOGRAPHY
- APPENDIX
- A. User Interface
  - B. Source Code

# **INTRODUCTION**

## **1.1 Overview**

A computer vision approach to classify garbage into recycling categories could be an efficient way to process waste. The project aims to take garbage waste images and classify them into six classes: cardboard, glass, paper, metal, plastic and trash. We use garbage image database that contains around 2696 images for each i.e. 450 approx. In this project, we import the images in data generator library and configure the class and then apply the image data generator functionality then we import the model building libraries. Initialize the model, then we add the layers, first CNN and then dense.

Then we have built the HTML web page and for doing this we used:

- Git
- Git Hub
- Python 3.7
- Anaconda Spyder
- Html5
- CSS3
- Java Script
- Flask Framework

## **1.2 Purpose**

This code set shows how to differentiate the garbage type by pulling it aside on the basis of 2696 images we have uploaded and described which will be recognized by CNN. the six type of garbage's are: cardboard, glass, paper, metal, plastic and trash.

On the basis of recognition, the model differentiates the garbage. which is possibly harmful or hazardous to humans labours.

The main purpose of garbage collection system is to provide another way for the labours for reducing the hand-picking method and giving the e-waste materials.

The goals for the system are:

- to make the waste separate by image identification
- to rescue electronic material by recycling or deploy
- to decrease the mixing of waste material from every household

# **LITERATURE SURVEY**

## **2.1 Existing Problems**

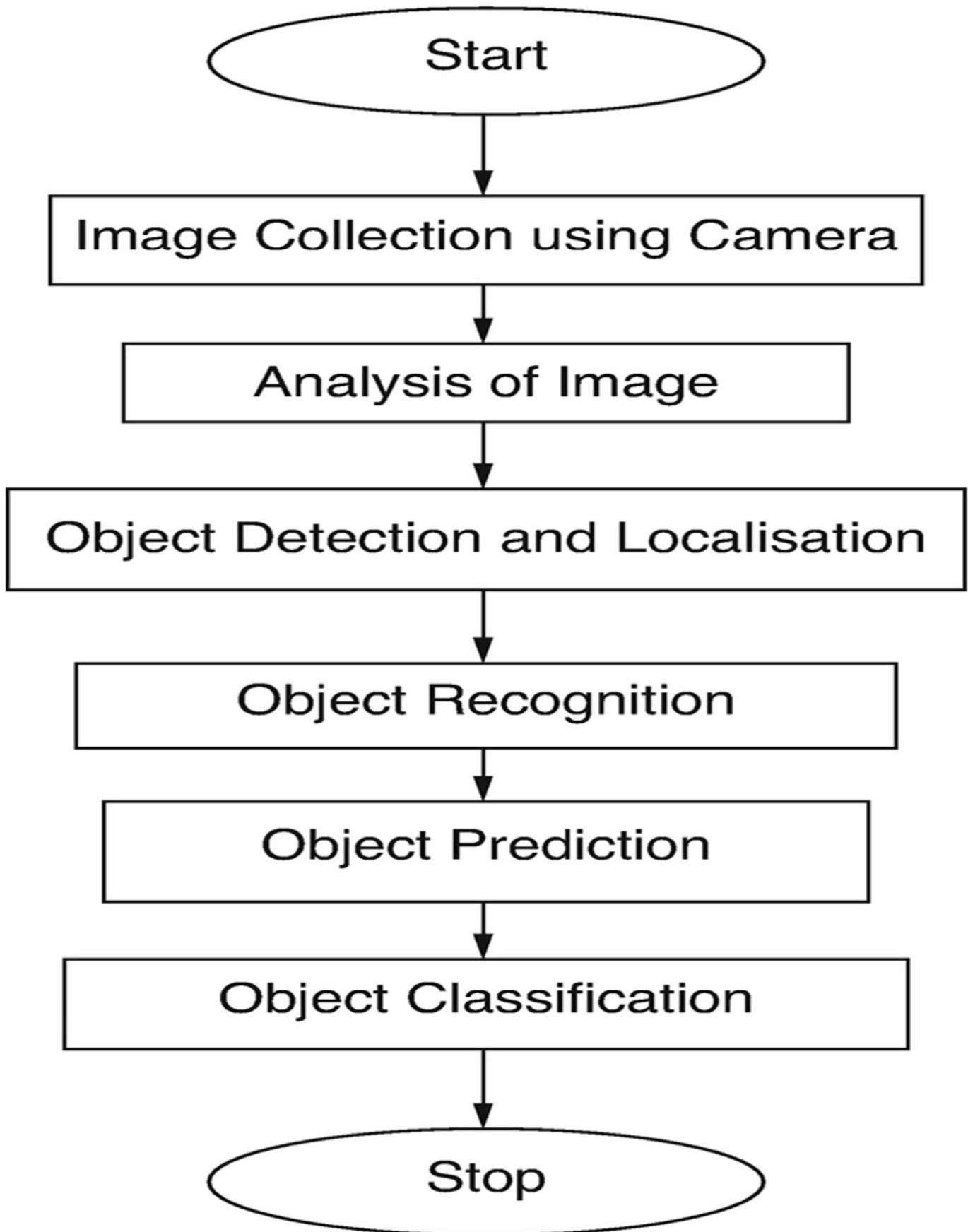
The accumulation of solid waste in the urban area is becoming a great concern, and it would result in environmental pollution and may be hazardous to human health if it is not properly managed. It is important to have an advanced/intelligent waste management system to manage a variety of waste materials. One of the most important steps of waste management is the separation of the waste into the different components and this process is normally done manually by hand-picking. Convolutional Neural Network model which is used to classify the waste into different groups/types such as glass, metal, paper, and plastic etc. The proposed system is tested on the trash image data set which was developed by Gary Thung and Mindy Yang, and is able to achieve an accuracy of 87% on the data set. The separation process of the waste will be faster and intelligent using the proposed waste material classification system without or reducing human involvement.

## **2.2 Proposed Solution**

The present way of separating waste/garbage is the hand-picking method, whereby someone is employed to separate out the different objects/materials. The person, who separate waste, is prone to diseases due to the harmful substances in the garbage. With this in mind, it motivated us to develop an automated system which is able to sort the waste and this system can take short time to sort the waste, and it will be more accurate in sorting than the manual way. With the system in place, the beneficial separated waste can still be recycled and converted to energy and fuel for the growth of the economy. The system that is developed for the separation of the accumulated waste is based on the combination of Convolutional Neural Network (CNN) with recognition and classification.

## THEORITICAL ANALYSIS

### 3.1 Block Diagram



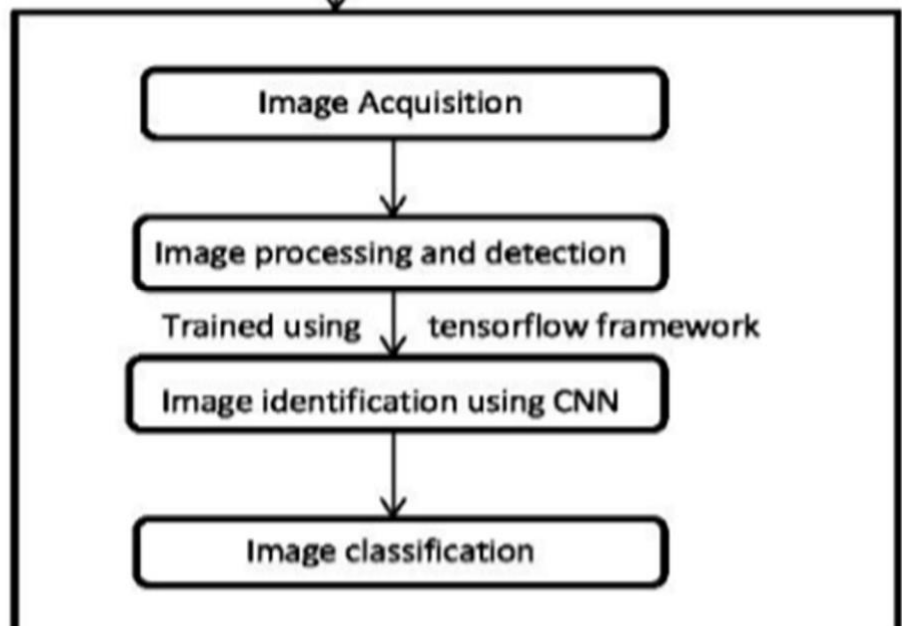
### 3.2 Hardware/Software Designing

CAPTURING OF IMAGE



Input image

SOFTWARE UNIT

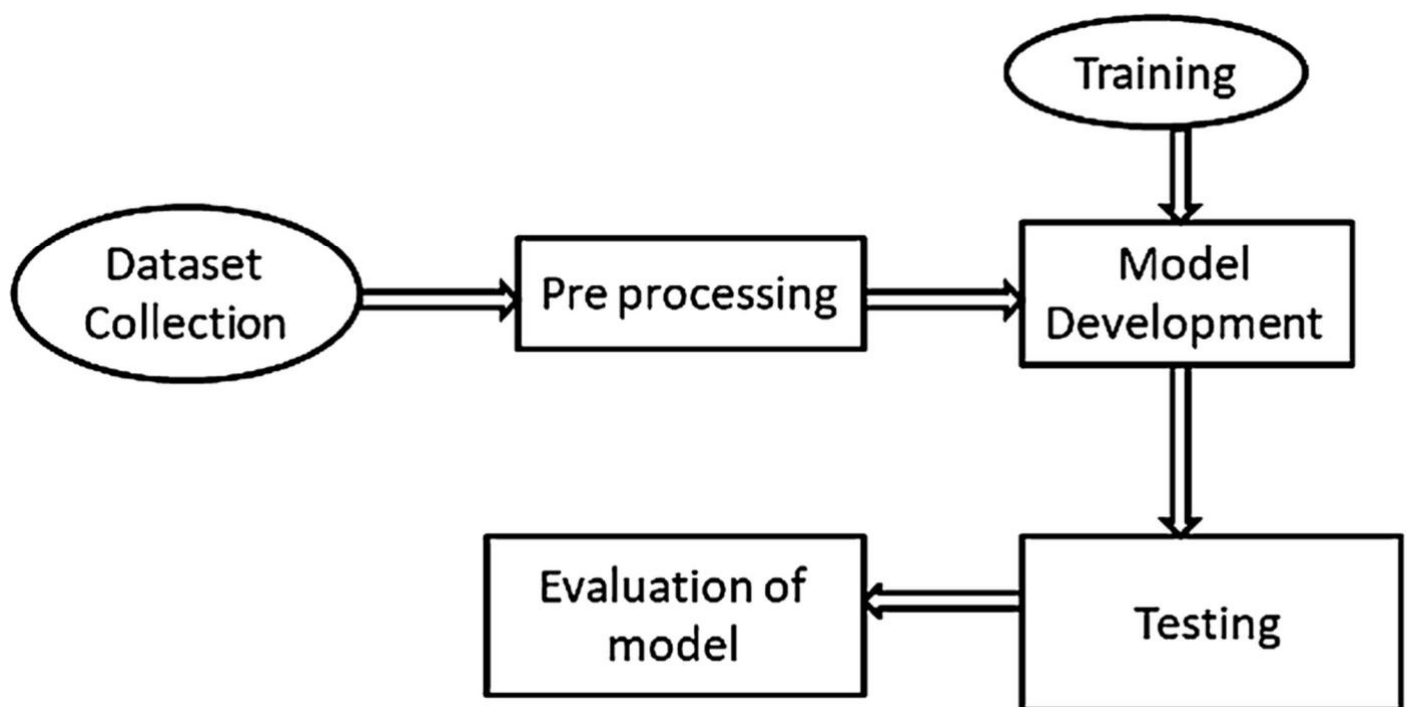


## **EXPERIMENTAL INVESTIGATION**

In this process of developing the project we have undergone many investigation processes to learn and understand the concept of smart garbage classification on the basis of CNN and to make it successfully. For this we have to learn and investigate following:

- Convolution Neural Network
- ZOHO Writer
- Anaconda Spyder
- Slack
- Git Hub
- Git
- Alex net
- Sublime Text for CSS3

## **FLOWCHART**



# **RESULTS**

## **1) Data Collection:**

In our data sets, non-organic recyclable waste has been categorized in 6 groups. These are- cardboard, plastic, metal, glass, paper, and trash. For plastic, we have collected 480 images of each plastic bottles and boxes. In a similar fashion, for each group, we have chosen six objects and for each of them, we have collected 20 images. Hence, our total data sets have 2696 images of many different objects along with their recycle values.

## **2) Object Detection:**

The process initiates by introducing a small region in the whole image and gradually merging them with hierarchical grouping. By analyzing the similarities between all neighboring regions, two most similar regions are selected and then grouped together. After that, the resulting group region is compared with all other neighboring regions and a new group is formed. The whole process continues until it becomes a single region. A demo of object detection and segmentation.

## **3) Object Recognition:**

After detection and segmentation, the images of the objects are fed into the trained CNN for recognition and labeling purposes. The output of the CNN-based model along with a few labeled objects.



1). Cardboard



2). Glass



3). Metal



4). Paper



5). Plastic



6). Trash



## **ADVANTAGES**

- Reduce the cost and complexity of the edge devices.
- Labours life will be safe from harmful and hazardous waste.
- In household waste, there are some substances that are not easy to be degraded and these substances will pollute the soil badly. By garbage classification, we can remove substances that are recyclable and substances that are hard to be degraded. Thus reducing more than 50% of the waste.
- Reduce environmental pollution. The waste battery contains metal mercury, cadmium and other toxic substances that will do harm to human, the waste plastics in the soil will reduce the production of crop and the waste plastics can also lead to the death of animals. Thus recycling these wastes can reduce the damage.

## **DISADVANTAGE**

- High cost
- Need skilled personals
- Labours that earn small amount of money by collecting the hand picked garbage's for their living will be effected .

## **APPLICATION**

This garbage classification application powered by CNN is used to classify the type of garbage mainly glass, paper, cardboard and metal. Which recognized by the CNN image recognition process that will help us to classify the garbage and the type of recycling process is to be processed for different type of garbage classified.

## **CONCLUSION**

In conclusion, we proposed a garbage classification system that is able to separate different components of waste using the Machine learning tools. This system can be used to automatically classify waste and help in reducing human intervention and preventing infection and pollution. From the result, when tested against the trash data set, we got an accuracy of 87%. The separation process of the waste will be faster and intelligent using our system without or reducing human involvement. If more image is added to the data set, the system accuracy can be improved In the future, we will tend to improve our system to be able to categories more waste items, by turning some of the parameters used.

## **FUTURE SCOPE**

The product developed can also find application in various food industries, fabrication industries and many more where all the process of segregation is involved.

According to the need, the images of the materials to be separated will be entered in the data set and the algorithm is trained.

## **BIBLIOGRAPHY**

**Name:** Mohammed Marzuk Ali .S

**College Name:** Coimbatore Institute Of Technology

**Work Title:** Garbage Classification Using Deep Learning

**Name:** Akash Kumar .S

**College Name:** Coimbatore Institute Of Technology

**Work Title:** Garbage Classification Using Deep Learning

**Name:** Sridhar .S

**College Name:** Coimbatore Institute Of Technology

**Work Title:** Garbage Classification Using Deep Learning

**Name:** Nikesh .C

**College Name:** Coimbatore Institute Of Technology

**Work Title:** Garbage Classification Using Deep Learning

## Reference:

### 1. Train and Test Image Data Set

[https://drive.google.com/drive/folders/1\\_RZkL0ZcFfSk6jnysyQ1UVDgMp\\_2ssm5?usp=sharing](https://drive.google.com/drive/folders/1_RZkL0ZcFfSk6jnysyQ1UVDgMp_2ssm5?usp=sharing)

### 2. HTML, CSS, JavaScript and Python

<https://stackoverflow.com/>

### 3. HTML, CSS and JavaScript

<https://www.w3schools.com/>

### 4. Python CNN Issues

<https://github.com/>

### 5. Python Preprocessor and Model Building

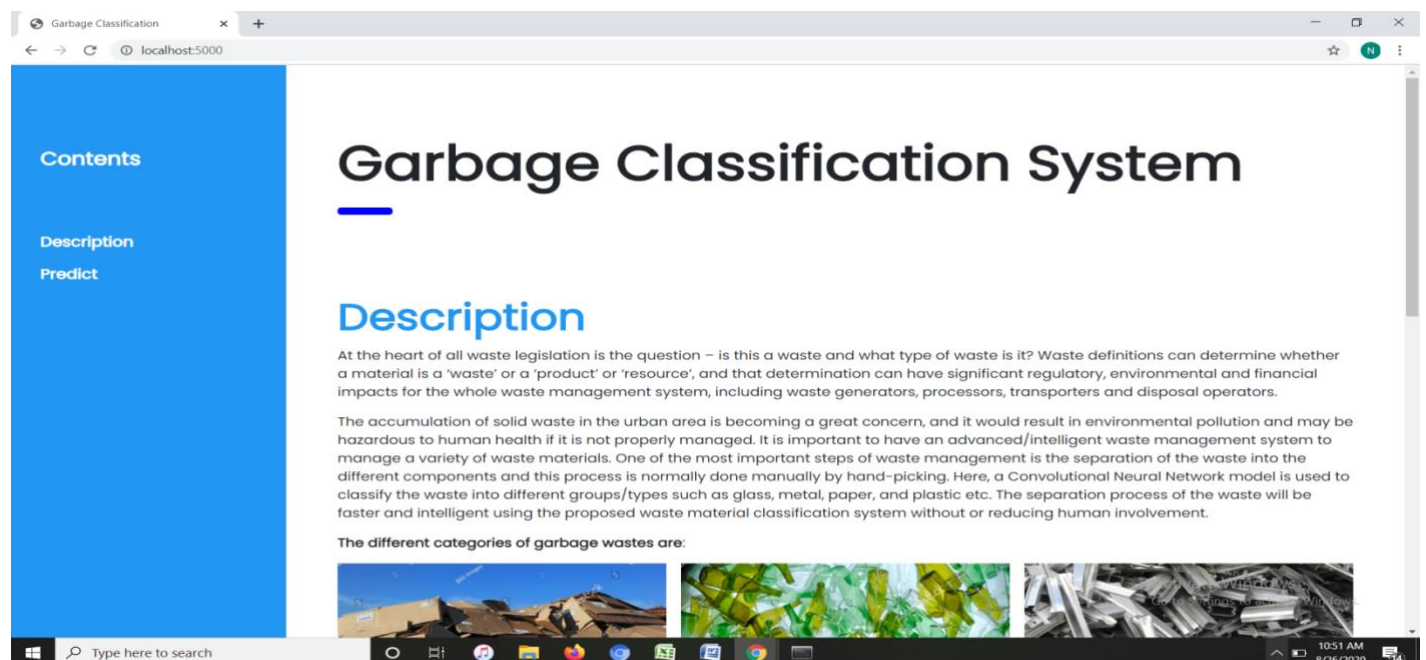
<https://www.kaggle.com/rkuo2000/garbage-cnn>

### 6. Working with Flask Framework

<https://palletsprojects.com/p/flask/>

## APPENDIX

### A. User Interface:



Garbage Classification


localhost:5000

Contents


Description

Predict


The different categories of garbage wastes are:




Cardboard Waste




Glass Waste




Metal Waste



Paper Waste



Plastic Waste



General Trash

Activate Windows  
Go to Settings to activate Windows.

Garbage Classification

localhost:5000

Contents


Description

Predict

## Predict

Upload a picture to predict the type

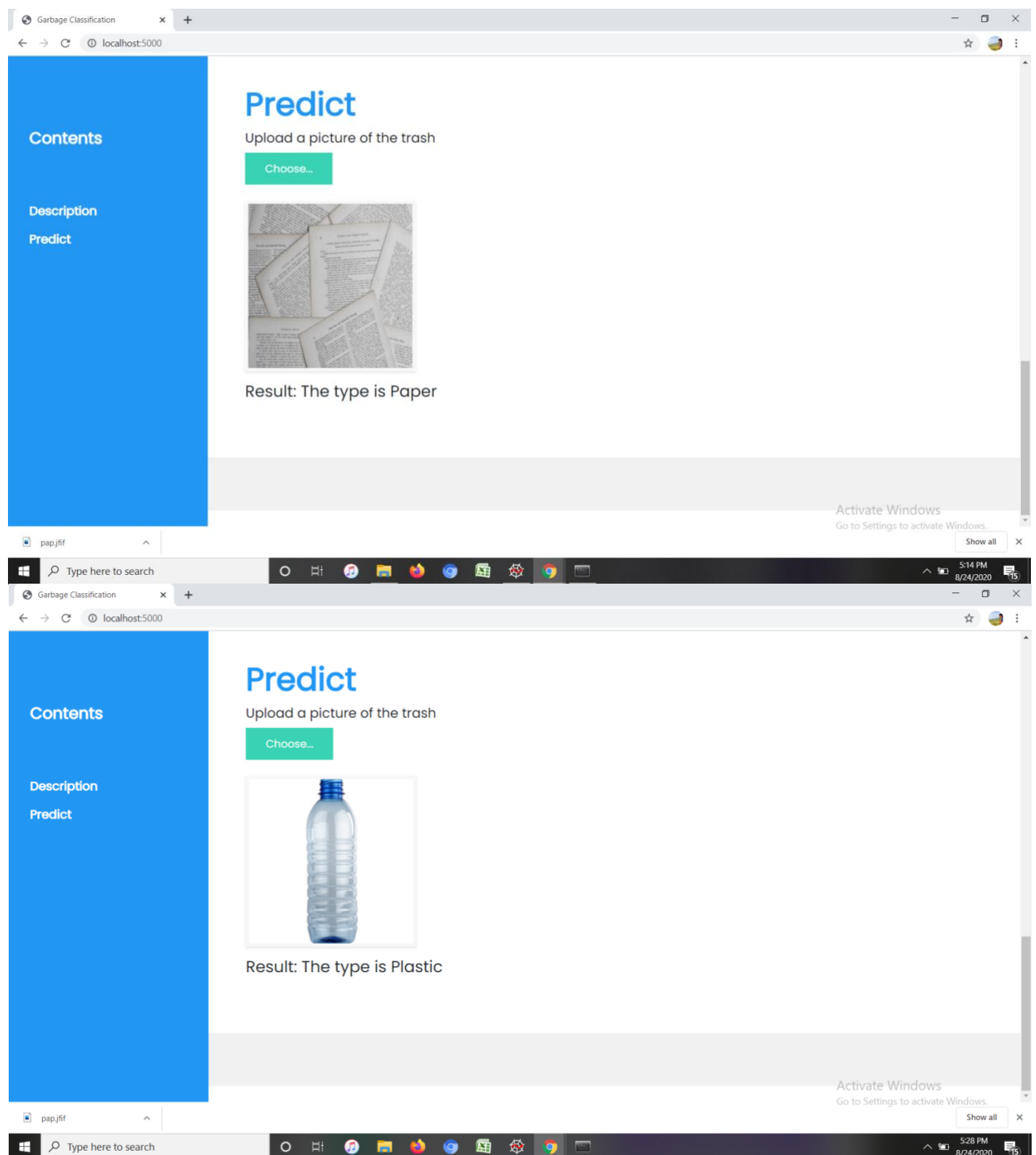
Choose...



Result: The type is Cardboard

Activate Windows  
Go to Settings to activate Windows.

## → Predicting the type of garbage:



## B. Source Code:

### 1. garbage.html

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <title>Garbage Classification</title>
4 <meta charset="UTF-8">
5 <meta name="viewport" content="width=device-width, initial-scale=1">
6 <meta http-equiv="X-UA-Compatible" content="ie=edge">
7 <link href="https://cdn.jsdelivr.net/npm/bootstrap@4.0.0/css/bootstrap.min.css" rel="stylesheet">
8 <script src="https://cdn.jsdelivr.net/npm/popper.js@1.12.9/umd/popper.min.js"></script>
9 <script src="https://cdn.jsdelivr.net/npm/jquery@3.3.1/jquery.min.js"></script>
10 <script src="https://cdn.jsdelivr.net/npm/bootstrap@4.0.0/js/bootstrap.min.js"></script>
11 <link href="{ url_for('static'), filename='css/main.css' }" rel="stylesheet">
12 <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
13 <link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Poppins">
14 <style>
15 body,h1,h2,h3,h4,h5 {font-family: "Poppins", sans-serif}
16 body {font-size:16px;}
17 .w3-half img{margin-bottom: -6px;margin-top:16px;opacity:0.8;cursor:pointer}
18 .w3-half img:hover{opacity:1}
19 </style>
20 <body>
21
22 <!-- Sidebar/menu -->
23 <nav class="w3-sidebar w3-blue w3-collapse w3-top w3-large w3-padding" style="z-index:3;width:300px;font-weight:bold;" id="mySidebar"><br>
24 <a href="javascript:void(0)" onclick="w3_close()" class="w3-button w3-hide-large w3-display-left" style="width:100%;font-size:22px">Close Menu</a>
25 <div class="w3-container">
26 <h3 class="w3-padding-64"><b>Contents</b></h3>
27 </div>
28 <div class="w3-bar-block">
29 <a href="#" onclick="w3_close()" class="w3-bar-item w3-button w3-hover-white">Description</a>
30 <a href="#" onclick="w3_close()" class="w3-bar-item w3-button w3-hover-white">Predict</a>
31 </div>
32 </nav>
33
34 <!-- Top menu on small screens -->
35 <header class="w3-container w3-top w3-hide-large w3-blue w3-xlarge w3-padding">
36 <a href="javascript:void(0)" class="w3-button w3-blue w3-margin-right" onclick="w3_open()">☰</a>
37 <span>Contents</span>
38 </header>
39
40 <!-- Overlay effect when opening sidebar on small screens -->
41 <div class="w3-overlay w3-hide-large" onclick="w3_close()" style="cursor:pointer" title="close side menu" id="myOverlay"></div>
42
43 <!-- IPAGE CONTENT! -->
44 <div class="w3-main" style="margin-left:340px;margin-right:40px">
45
46 <!-- Header -->
47 <div class="w3-container" style="margin-top:80px" id="heading">
48 <h1 class="w3-jumbo"><b>Garbage Classification System</b></h1>
49 <hr style="width:50px;border:5px solid blue" class="w3-round">
50 </div>
51 <!-- Description -->
52 <div class="w3-container" id="services" style="margin-top:75px">
53 <h1 class="w3-xxxlarge w3-text-blue"><b>Description</b></h1>
54 <p>At the heart of all waste legislation is the question - is this a waste and what type of waste is it? Waste definitions can determine whether a
55 <p>The accumulation of solid waste in the urban area is becoming a great concern, and it would result in environmental pollution and may be hazardous
56 <p><b>The different categories of garbage wastes are</b></p>
57 <p><b>Cardboard</b></p>
58 </div>
59 <div class="w3-row-padding">
60 <div class="w3-col m4 w3-margin-bottom">
61 <div class="w3-light-grey">
62 
63 <p class="w3-opacity">Cardboard Waste</p>
64 </div>
65 </div>
66 </div>
67 </div>
68 </div>
69 <div class="w3-col m4 w3-margin-bottom">
70 <div class="w3-light-grey">
71 
72 <p class="w3-opacity">Glass Waste</p>
73 </div>
74 </div>
75 </div>
76 </div>
77 </div>
78 </div>
79 <div class="w3-col m4 w3-margin-bottom">
80 <div class="w3-light-grey">
81 
82 <p class="w3-opacity">Metal Waste</p>
83 </div>
84 </div>
85 </div>
86 </div>
87 </div>
88 </div>
89 </div>
90 </div>
91 </div>
92 </div>
93 </div>
94 </div>
95 </div>
96 </div>
97 </div>
98 </div>
99 </div>
100 </div>
```



Spyder (Python 3.8)

File Edit Search Source Run Debug Consoles Projects Tools View Help

C:\Users\LENOVO

D:\webapplication\ISPS-BIT-3587-Garbage-Classification-master\templates\garbage1.html

```
101 
102 <div class="w3-container">
103   <p class="w3-opacity">Metal Waste</p>
104 </div>
105 </div>
106 </div>
107 <div class="w3-row-padding">
108   <div class="w3-col m4 w3-margin-bottom">
109     <div class="w3-light-grey">
110       
111       <div class="w3-container">
112         <p class="w3-opacity">Paper Waste</p>
113       </div>
114     </div>
115 </div>
116 <div class="w3-col m4 w3-margin-bottom">
117   <div class="w3-light-grey">
118     
119     <div class="w3-container">
120       <p class="w3-opacity">Plastic Waste</p>
121     </div>
122 </div>
123 <div class="w3-col m4 w3-margin-bottom">
124   <div class="w3-light-grey">
125     
126     <div class="w3-container">
127       <p class="w3-opacity">General Trash</p>
128     </div>
129 </div>
130 </div>
131 <div class="w3-container" id="prediction" style="margin-top:75px">
132   <!-- Predict -->
133 </div>
```

Usage

Here you can get help of any object by pressing Ctrl+H in front of it, either on the Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in Preferences > Help.

New to Spyder? Read our tutorial

Variable explorer Help Plots Files

Console 1/A

Python 3.8.3 (default, Jul 2 2020, 17:30:36) [MSC v.1916 64 bit (AMD64)] Type "copyright", "credits" or "license()" for more information.

IPython 7.16.1 -- An enhanced Interactive Python.

In [1]:

Activate Windows

Go to Settings to activate Windows.

Python console History

Kite: unsupported conda: base (Python 3.8.3) Line 140, Col 1 UTF-8-GUESSED CRLF RW Mem 69%

Type here to search

11:00 AM 8/26/2020

Spyder (Python 3.8)

File Edit Search Source Run Debug Consoles Projects Tools View Help

C:\Users\LENOVO

D:\webapplication\ISPS-BIT-3587-Garbage-Classification-master\templates\garbage1.html

```
121 <div class="w3-container" id="prediction" style="margin-top:75px">
122   <h1 class="w3-xxxlarge w3-text-blue"><b>Predict</b></h1>
123   <h4>Upload a picture to predict the type</h4>
124   <form action="http://localhost:5000/predict" id="upload-file" method="post" enctype="multipart/form-data">
125     <label for="imageUpload" class="upload-label">
126       Choose...
127     </label>
128     <input type="file" name="image" id="imageUpload" accept=".png, .jpg, .jpeg">
129   </form>
130   <div class="image-section" style="display:none">
131     <div class="img-preview">
132       <div id="imagePreview">
133     </div>
134   </div>
135   <div>
136     <button type="button" class="btn btn-info btn-lg" id="btn-predict">Click to know the type</button>
137   </div>
138 </div>
139 <div class="loader" style="display:none"></div>
140 <h3>
141   <span id="result"></span>
142 </h3>
143 </div>
144 <!-- End page content -->
145 </div>
146 <!-- W3.CSS Container -->
147 <div class="w3-light-grey w3-container w3-padding-32" style="margin-top:75px;padding-right:58px"><p class="w3-right"><a href="https://www.w3schools.com" />
148 </p>
149 </div>
150 <script>
151 // Script to open and close sidebar
152 function w3_open() {
153   document.getElementById("mySidebar").style.display = "block";
154   document.getElementById("myOverlay").style.display = "block";
155 }
156 function w3_close() {
157   document.getElementById("mySidebar").style.display = "none";
158   document.getElementById("myOverlay").style.display = "none";
159 }
160 </script>
161 </body>
162 <footer>
163   <script src="{{ url_for('static', filename='js/main.js') }}" type="text/javascript"></script>
164 </footer>
165 </html>
```

Usage

Here you can get help of any object by pressing Ctrl+H in front of it, either on the Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in Preferences > Help.

New to Spyder? Read our tutorial

Variable explorer Help Plots Files

Console 1/A

Python 3.8.3 (default, Jul 2 2020, 17:30:36) [MSC v.1916 64 bit (AMD64)] Type "copyright", "credits" or "license()" for more information.

IPython 7.16.1 -- An enhanced Interactive Python.

In [1]:

Activate Windows

Go to Settings to activate Windows.

Python console History

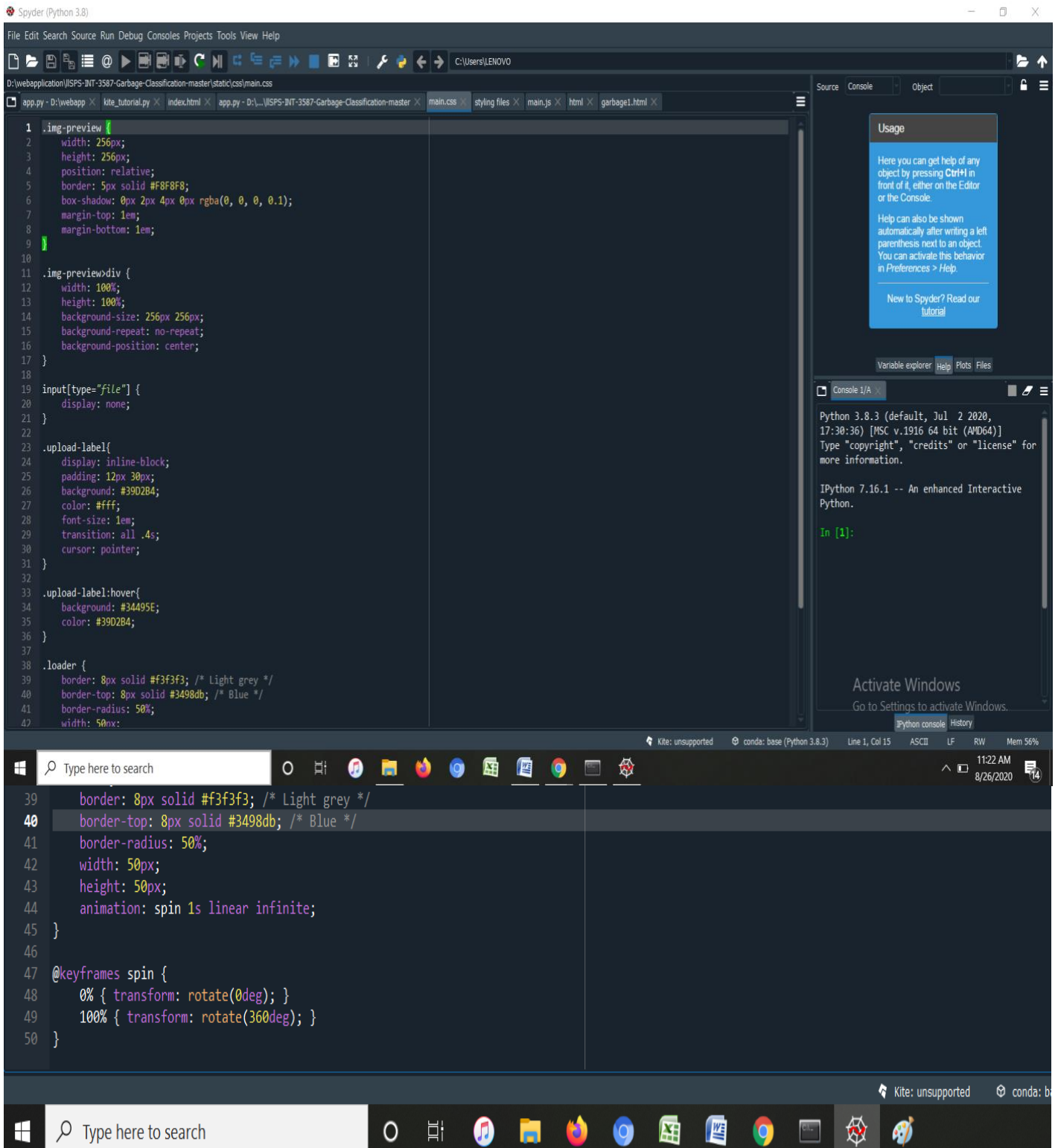
Kite: unsupported conda: base (Python 3.8.3) Line 117, Col 9 UTF-8-GUESSED CRLF RW Mem 69%

Type here to search

11:01 AM 8/26/2020



## 2. main.css



```
1 .img-preview {
2     width: 256px;
3     height: 256px;
4     position: relative;
5     border: 5px solid #F8F8F8;
6     box-shadow: 0px 2px 4px 0px rgba(0, 0, 0, 0.1);
7     margin-top: 1em;
8     margin-bottom: 1em;
9 }
10
11 .img-preview>div {
12     width: 100%;
13     height: 100%;
14     background-size: 256px 256px;
15     background-repeat: no-repeat;
16     background-position: center;
17 }
18
19 input[type="file"] {
20     display: none;
21 }
22
23 .upload-label{
24     display: inline-block;
25     padding: 12px 30px;
26     background: #390284;
27     color: #fff;
28     font-size: 1em;
29     transition: all .4s;
30     cursor: pointer;
31 }
32
33 .upload-label:hover{
34     background: #34495E;
35     color: #390284;
36 }
37
38 .loader {
39     border: 8px solid #f3f3f3; /* Light grey */
40     border-top: 8px solid #3498db; /* Blue */
41     border-radius: 50%;
42     width: 50px;
43     height: 50px;
44     animation: spin 1s linear infinite;
45 }
46
47 @keyframes spin {
48     0% { transform: rotate(0deg); }
49     100% { transform: rotate(360deg); }
50 }
```

### 3. main.js

The screenshot displays the Spyder Python IDE interface. The main editor window shows the `main.js` file with the following JavaScript code:

```
1 $(document).ready(function () {
2     // Init
3     $('#image-section').hide();
4     $('#loader').hide();
5     $('#result').hide();
6
7     // Upload Preview
8     function readURL(input) {
9         if (input.files && input.files[0]) {
10             var reader = new FileReader();
11             reader.onload = function (e) {
12                 $('#imagePreview').css('background-image', 'url(' + e.target.result + ')');
13                 $('#imagePreview').hide();
14                 $('#imagePreview').fadeIn(650);
15             }
16             reader.readAsDataURL(input.files[0]);
17         }
18     }
19     $("#imageUpload").change(function () {
20         $('#image-section').show();
21         $('#btn-predict').show();
22         $('#result').text('');
23         $('#result').hide();
24         readURL(this);
25     });
26
27     // Predict
28     $('#btn-predict').click(function () {
29         var form_data = new FormData($('#upload-file')[0]);
30
31         // Show loading animation
32         $(this).hide();
33         $('#loader').show();
34
35         // Make prediction by calling api /predict
36         $.ajax({
37             type: 'POST',
38             url: '/predict',
39             data: form_data,
40             contentType: false,
41             cache: false,
42             processData: false,
43             async: true,
44             success: function (data) {
45                 // Get and display the result
46                 $('#loader').hide();
47                 $('#result').fadeIn(600);
48                 $('#result').text(' Result: ' + data);
49                 console.log('Success!');
50             },
51         });
52     });
53
54 });
55
```

The right sidebar contains a 'Usage' panel with a blue box explaining how to get help by pressing `Ctrl+H` in front of an object, and a 'Console' panel showing the IPython 7.16.1 prompt `In [1]:`. The bottom status bar indicates 'Kite: ready' and 'conda: base (Python 3.8.3)'. The Windows taskbar at the bottom shows the search bar and various application icons.

## 4. app.py

Spyder (Python 3.8)

File Edit Search Source Run Debug Consoles Projects Tools View Help

C:\Users\LENOVO

D:\webapplication\ISPS-3587-Garbage-Classification-master\app.py

app.py - D:\webapp x kite\_tutorial.py x index.html x app.py - D:\...ISPS-3587-Garbage-Classification-master x main.css x styling files x main.js x html x garbage1.html x

```
1 import numpy as np
2 import os
3 from keras.models import load_model
4 from keras.preprocessing import image
5 import tensorflow as tf
6 global graph
7 from flask import Flask, request, render_template
8 from werkzeug.utils import secure_filename
9 from event.pywsgi import WSGIServer
10 graph = tf.get_default_graph()
11 import theano
12 theano.config.optimizer="None"
13 os.environ['THEANO_FLAGS'] = 'optimizer=None'
14 app = Flask(__name__)
15 model = load_model("garbagenew.h5")
16
17
18 @app.route('/')
19 def index():
20     return render_template("garbage1.html")
21
22 @app.route('/predict', methods = ['GET', 'POST'])
23 def upload():
24     if request.method == 'POST':
25         f = request.files['image']
26         print("current path")
27         basepath = os.path.dirname(__file__)
28         print("current path", basepath)
29         filepath = os.path.join(basepath, 'uploads', f.filename)
30         print("upload folder is ", filepath)
31         f.save(filepath)
32
33         img = image.load_img(filepath, target_size = (64, 64))
34         x = image.img_to_array(img)
35         x = np.expand_dims(x, axis = 0)
36
37         with graph.as_default():
38             preds = model.predict_classes(x)
39             print("prediction", preds)
40
41         index = ['Cardboard', 'Glass', 'Metal', 'Paper', 'Plastic', 'Trash']
42         text = "The type is "+index[preds[0]]
43
44         return text
45
46
47 if __name__ == '__main__':
48     app.run(debug=True, threaded=False)
49
50
```

Usage

Here you can get help of any object by pressing **Ctrl+H** in front of it, either on the Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in Preferences > Help.

New to Spyder? Read our [tutorial](#)

Variable explorer Help Plots Files

Console 1/A

Python 3.8.3 (default, Jul 2 2020, 17:30:36) [MSC v.1916 64 bit (AMD64)]  
Type "copyright", "credits" or "license" for more information.

IPython 7.16.1 -- An enhanced Interactive Python.

In [1]:

Activate Windows  
Go to Settings to activate Windows.

Python console History

LSP Python: ready Kite: indexing conda: base (Python 3.8.3) Line 10, Col 1 ASCII CRLF RW Mem 60%

Type here to search

```
40
41
42     index = ['Cardboard', 'Glass', 'Metal', 'Paper', 'Plastic', 'Trash']
43     text = "The type is "+index[preds[0]]
44
45     return text
46
47
48 if __name__ == '__main__':
49     app.run(debug=True, threaded=False)
50
```

LSP Python: ready Kite: indexing conda: b

## 5. Training.ipynb

The screenshot displays a Jupyter Notebook interface with the title "training1". The notebook is running on a local host (localhost:8888). The first cell, labeled "In [1]:", contains the following code:

```
import keras
from tensorflow.keras.layers import Conv2D, Flatten, MaxPooling2D, Dropout, BatchNormalization
from keras.models import Sequential
from keras.layers import Dense
from keras.layers import Convolution2D
from keras.layers import MaxPooling2D
from keras.layers import Flatten
from keras.layers import Dropout
```

Below the code, a message indicates the use of the TensorFlow backend:

```
Using TensorFlow backend.
C:\Users\marzu\anaconda3\envs\tensorflow\lib\site-packages\tensorflow\python\framework\dtypes.py:516: FutureWarning: Passing (type, 1) or '1type' as a synonym of type is deprecated; in a future version of numpy, it will be understood as (type, (1,)) / '(1,)type'.
_np_qint8 = np.dtype [("qint8", np.int8, 1)]
C:\Users\marzu\anaconda3\envs\tensorflow\lib\site-packages\tensorflow\python\framework\dtypes.py:517: FutureWarning: Passing (type, 1) or '1type' as a synonym of type is deprecated; in a future version of numpy, it will be understood as (type, (1,)) / '(1,)type'.
_np_quint8 = np.dtype [("quint8", np.uint8, 1)]
C:\Users\marzu\anaconda3\envs\tensorflow\lib\site-packages\tensorflow\python\framework\dtypes.py:518: FutureWarning: Passing (type, 1) or '1type' as a synonym of type is deprecated; in a future version of numpy, it will be understood as (type, (1,)) / '(1,)type'.
```

The second cell, labeled "In [12]:", contains the following code:

```
model.add(MaxPooling2D(pool_size=(2,2)))
```

Below the code, a warning message is displayed:

```
WARNING:tensorflow:From C:\Users\marzu\anaconda3\envs\tensorflow\lib\site-packages\keras\backend\tensorflow_backend.py:3976: The name tf.nn.max_pool is deprecated. Please use tf.nn.max_pool2d instead.
```

The subsequent cells show the addition of more layers to the model:

```
In [13]: model.add(Convolution2D(64,(3,3), input_shape=(64,64,3), strides=(1,1), padding='same', activation = 'relu'))
In [14]: model.add(MaxPooling2D(pool_size=(2,2)))
In [15]: model.add(Convolution2D(128,(3,3), input_shape=(64,64,3), strides=(1,1), padding='same', activation = 'relu'))
In [16]: model.add(MaxPooling2D(pool_size=(2,2)))
In [17]: model.add(Convolution2D(128,(3,3), input_shape=(64,64,3), strides=(1,1), padding='same', activation = 'relu'))
In [18]: model.add(MaxPooling2D(pool_size=(2,2)))
In [19]: model.add(Dropout(0.25))
```

A warning message is also displayed at the bottom of the notebook:

```
WARNING:tensorflow:From C:\Users\marzu\anaconda3\envs\tensorflow\lib\site-packages\keras\backend\tensorflow_backend.py:133: The name tf.nn.max_pool is deprecated. Please use tf.nn.max_pool2d instead.
```



localhost:8888/notebooks/training1.ipynb

jupyter training1 Last Checkpoint: Last Sunday at 9:03 PM (autosaved)

File Edit View Insert Cell Kernel Help Trusted Python 3

In [2]: `from keras.preprocessing.image import ImageDataGenerator  
train_datagen = ImageDataGenerator(rescale = 1./255, shear_range=0.2, zoom_range=0.2, horizontal_flip=True)  
test_datagen = ImageDataGenerator(rescale = 1./255)`

In [3]: `x_train = train_datagen.flow_from_directory(r'D:\smartinern_garbage_class\data_set\train', target_size=(64,64), batch_size=64, class_mode='categorical')`  
Found 2056 images belonging to 6 classes.

In [4]: `x_train.class_indices`  
Out[4]: {'Cardboard': 0, 'Glass': 1, 'Metal': 2, 'Paper': 3, 'Plastic': 4, 'Trash': 5}

In [7]: `x_test = test_datagen.flow_from_directory(r'D:\smartinern_garbage_class\data_set\Test', target_size=(64,64), batch_size=64, class_mode='categorical')`  
Found 471 images belonging to 6 classes.

In [8]: `x_train.image_shape`  
Out[8]: (64, 64, 3)

localhost:8888/notebooks/training1.ipynb

jupyter training1 Last Checkpoint: Last Sunday at 9:03 PM (autosaved)

File Edit View Insert Cell Kernel Help Trusted Python 3

In [20]: `model.add(Flatten())`

In [21]: `model.add(Dense(output_dim=512, activation='relu'))`  
C:\Users\marzu\anaconda3\envs\tensorflow\lib\site-packages\ipykernel\_launcher.py:1: UserWarning: Update your `Dense` call to the Keras 2 API: `Dense(activation="relu", units=512)`  
""Entry point for launching an IPython kernel.

In [22]: `model.add(Dense(output_dim= 6, activation='softmax'))`  
C:\Users\marzu\anaconda3\envs\tensorflow\lib\site-packages\ipykernel\_launcher.py:1: UserWarning: Update your `Dense` call to the Keras 2 API: `Dense(activation="softmax", units=6)`  
""Entry point for launching an IPython kernel.

In [23]: `model.compile(loss='categorical_crossentropy', optimizer = 'adam', metrics= ['accuracy'])`  
WARNING:tensorflow:From C:\Users\marzu\anaconda3\envs\tensorflow\lib\site-packages\keras\optimizers.py:790: The name tf.train.Optimizer is deprecated. Please use tf.compat.v1.train.Optimizer instead.  
WARNING:tensorflow:From C:\Users\marzu\anaconda3\envs\tensorflow\lib\site-packages\keras\backend\tensorflow\_backend.py:3295: The name tf.log is deprecated. Please use tf.math.log instead.

Browser tabs: IISPS-INT-27, Microsoft W, Microsoft W, Student Das, Home Page, training1 - Ju, testing1 - Ju, Meet - s, +

Address bar: localhost:8888/notebooks/training1.ipynb

Navigation: Apps, Gmail, YouTube, Maps, News, Translate, Learn the basics of..., SmartBridge, Student Dashboard, Slack | 29-july | thes..., Dashboard, (1) WhatsApp

Jupyter training1 Last Checkpoint: Last Sunday at 9:03 PM (autosaved) Logout

File Edit View Insert Cell Kernel Help Trusted Python 3

In [25]: `model.fit_generator(x_train, steps_per_epoch=len(x_train), epochs=100, validation_data=x_test, validation_steps=len(x_test))`

```
WARNING:tensorflow:From C:\Users\marzu\anaconda3\envs\tensorflow\lib\site-packages\tensorflow\python\ops\math_grad.py:1250: a
dd_dispatch_support.<locals>.wrapper (from tensorflow.python.ops.array_ops) is deprecated and will be removed in a future ver
sion.
Instructions for updating:
Use tf.where in 2.0, which has the same broadcast rule as np.where
Epoch 1/100
33/33 [=====] - 54s 2s/step - loss: 1.6852 - acc: 0.2592 - val_loss: 1.6272 - val_acc: 0.2739
Epoch 2/100
33/33 [=====] - 25s 745ms/step - loss: 1.5198 - acc: 0.3688 - val_loss: 1.6061 - val_acc: 0.3461
Epoch 3/100
33/33 [=====] - 25s 758ms/step - loss: 1.4579 - acc: 0.3912 - val_loss: 1.5905 - val_acc: 0.3864
Epoch 4/100
33/33 [=====] - 25s 759ms/step - loss: 1.3403 - acc: 0.4636 - val_loss: 1.4169 - val_acc: 0.4628
Epoch 5/100
33/33 [=====] - 25s 760ms/step - loss: 1.2966 - acc: 0.4854 - val_loss: 1.2263 - val_acc: 0.5053
Epoch 6/100
33/33 [=====] - 25s 761ms/step - loss: 1.1816 - acc: 0.5356 - val_loss: 1.2040 - val_acc: 0.5350
Epoch 7/100
33/33 [=====] - 25s 752ms/step - loss: 1.1256 - acc: 0.5648 - val_loss: 1.0903 - val_acc: 0.5478
Epoch 8/100
```

saving the model

Project Report (1).pdf Canceled Project Report.pdf Show all

Browser tabs: IISPS-INT-27, Microsoft W, Microsoft W, Student Das, Home Page, training1 - Ju, testing1 - Ju, Meet - s, +

Address bar: localhost:8888/notebooks/training1.ipynb

Navigation: Apps, Gmail, YouTube, Maps, News, Translate, Learn the basics of..., SmartBridge, Student Dashboard, Slack | 29-july | thes..., Dashboard, (1) WhatsApp

Jupyter training1 Last Checkpoint: Last Sunday at 9:03 PM (autosaved) Logout

File Edit View Insert Cell Kernel Help Trusted Python 3

```
Epoch 5/100
33/33 [=====] - 25s 759ms/step - loss: 1.3403 - acc: 0.4636 - val_loss: 1.4169 - val_acc: 0.4628
Epoch 5/100
33/33 [=====] - 25s 760ms/step - loss: 1.2966 - acc: 0.4854 - val_loss: 1.2263 - val_acc: 0.5053
Epoch 6/100
33/33 [=====] - 25s 761ms/step - loss: 1.1816 - acc: 0.5356 - val_loss: 1.2040 - val_acc: 0.5350
Epoch 7/100
33/33 [=====] - 25s 752ms/step - loss: 1.1256 - acc: 0.5648 - val_loss: 1.0903 - val_acc: 0.5478
Epoch 8/100
```

saving the model

In [26]: `model.save('final_garbage.h5')`

In [ ]:

Project Report (1).pdf Canceled Project Report.pdf Show all

## 6. Testing .ipynb

The screenshot shows a Jupyter Notebook titled 'testing1' in a web browser. The browser's address bar shows 'localhost:8888/notebooks/testing1.ipynb'. The notebook interface includes a menu bar (File, Edit, View, Insert, Cell, Kernel, Help) and a toolbar with icons for file operations and running code. The first code cell, labeled 'In [1]:', contains the following Python code:

```
from keras.models import load_model
from keras.preprocessing import image
import numpy as np
```

Below the code, the output is displayed, showing several TensorFlow warnings about deprecated functions and types. The warnings include:

- FutureWarning: Passing (type, 1) or '1type' as a synonym of type is deprecated; in a future version of numpy, it will be understood as (type, (1,)) / '(1,)type'.
- FutureWarning: Passing (type, 1) or '1type' as a synonym of type is deprecated; in a future version of numpy, it will be understood as (type, (1,)) / '(1,)type'.
- FutureWarning: Passing (type, 1) or '1type' as a synonym of type is deprecated; in a future version of numpy, it will be understood as (type, (1,)) / '(1,)type'.
- FutureWarning: Passing (type, 1) or '1type' as a synonym of type is deprecated; in a future version of numpy, it will be understood as (type, (1,)) / '(1,)type'.

At the bottom of the notebook, there are two tabs: 'Project Report (1).pdf' (Canceled) and 'Project Report.pdf'. A 'Show all' button is visible on the right.

The screenshot shows the same Jupyter Notebook interface, but now the second code cell, labeled 'In [2]:', is active. The code in this cell is:

```
model=load_model('final_garbage.h5')
```

The output of this cell shows several TensorFlow warnings about deprecated functions and types, similar to the first cell. The warnings include:

- WARNING:tensorflow:From C:\Users\marzu\anaconda3\envs\tensorflow\lib\site-packages\keras\backend\tensorflow\_backend.py:517: The name tf.placeholder is deprecated. Please use tf.compat.v1.placeholder instead.
- WARNING:tensorflow:From C:\Users\marzu\anaconda3\envs\tensorflow\lib\site-packages\keras\backend\tensorflow\_backend.py:4138: The name tf.random\_uniform is deprecated. Please use tf.random.uniform instead.
- WARNING:tensorflow:From C:\Users\marzu\anaconda3\envs\tensorflow\lib\site-packages\keras\backend\tensorflow\_backend.py:3976: The name tf.nn.max\_pool is deprecated. Please use tf.nn.max\_pool2d instead.
- WARNING:tensorflow:From C:\Users\marzu\anaconda3\envs\tensorflow\lib\site-packages\keras\backend\tensorflow\_backend.py:131: The name tf.get\_default\_graph is deprecated. Please use tf.compat.v1.get\_default\_graph instead.
- WARNING:tensorflow:From C:\Users\marzu\anaconda3\envs\tensorflow\lib\site-packages\keras\backend\tensorflow\_backend.py:133: The name tf.placeholder\_with\_default is deprecated. Please use tf.compat.v1.placeholder\_with\_default instead.
- WARNING:tensorflow:From C:\Users\marzu\anaconda3\envs\tensorflow\lib\site-packages\keras\backend\tensorflow\_backend.py:3445: calling dropout (from tensorflow.python.ops.nn\_ops) with keep\_prob is deprecated and will be removed in a future version. Instructions for updating: Please use 'rate' instead of 'keep\_prob'. Rate should be set to 'rate = 1 - keep\_prob'.
- WARNING:tensorflow:From C:\Users\marzu\anaconda3\envs\tensorflow\lib\site-packages\keras\backend\tensorflow\_backend.py:133: The name tf.get\_default\_session is deprecated. Please use tf.compat.v1.get\_default\_session instead.

At the bottom of the notebook, there are two tabs: 'Project Report (1).pdf' (Canceled) and 'Project Report.pdf'. A 'Show all' button is visible on the right.

A notification box in the bottom right corner states: 'Screenshot saved. The screenshot was added to your OneDrive.'



Browser tabs: IISPS-INT-27, Microsoft W, Microsoft W, Student Das, Home Page, training1 - Ju, testing1 - Ju, Meet - s, +

Address bar: localhost:8888/notebooks/testing1.ipynb

Navigation bar: Apps, Gmail, YouTube, Maps, News, Translate, Learn the basics of..., SmartBridge, Student Dashboard, Slack | 29-july | thes..., Dashboard, (1) WhatsApp


Jupyter interface: jupyter testing1 Last Checkpoint: Last Sunday at 9:03 PM (autosaved) Logout

Menu: File Edit View Insert Cell Kernel Help Trusted Python 3

Code cells:

```
In [5]: img=image.load_img(r'C:\\Users\\marzu\\trash8.jpg',target_size=(64,64))
```

```
In [6]: img
```

Out[6]: 

```
In [7]: x=image.img_to_array(img)
```

```
In [8]: x=np.expand_dims(x,axis=0)
```

```
In [9]: model.predict_classes(x)
```

Out[9]: array([5], dtype=int64)

```
In [15]: img=image.load_img(r'C:\\Users\\marzu\\glass19.jpg',target_size=(64,64))
```

```
In [16]: img
```

Project Report (1).pdf Canceled Project Report.pdf Show all

Browser tabs: IISPS-INT-27, Microsoft W, Microsoft W, Student Das, Home Page, training1 - Ju, testing1 - Ju, Meet - s, +

Address bar: localhost:8888/notebooks/testing1.ipynb


Navigation bar: Apps, Gmail, YouTube, Maps, News, Translate, Learn the basics of..., SmartBridge, Student Dashboard, Slack | 29-july | thes..., Dashboard, (1) WhatsApp

Jupyter interface: jupyter testing1 Last Checkpoint: Last Sunday at 9:03 PM (autosaved) Logout

Menu: File Edit View Insert Cell Kernel Help Trusted Python 3

Code cells:

```
In [16]: img
```

Out[16]: 

```
In [17]: x=image.img_to_array(img)
```

```
In [18]: x=np.expand_dims(x,axis=0)
```

```
In [19]: model.predict_classes(x)
```

Out[19]: array([1], dtype=int64)

```
In [ ]:
```

```
In [ ]:
```

Project Report (1).pdf Canceled Project Report.pdf Show all