Masa Hu
May 5, 2017
CSCE 470
Trash Classification Project

User Manual

All code used for the project can be found in the source directory, results can be found in the results directory, and documentation as well as posters can be found in the products directory. In order to use this software please take the following steps.

Step 1: Creating the Environment

Read the readme.txt file and install necessary software before continuing

Step 2: Usage of Software

Move to the Source directory. Although all python code can be used as they are, it is recommended that the bash scripts provided are used to ensure proper usage of the software. The only exception is exectute.py which allows for the user to see the segmentation and extraction process of individual images.

ClassifyImage.sh:

arg1 – group_name
arg2 – in group directory (e.g. categories/treematter/ingroup)
arg3 – out group directory (e.g. categories/treematter/outgroup)
arg4 – showFlag (show/noshow)

NOTE:

- training.txt file will not move properly if arg1 is not the same as the folder name in categories
- show flag is case sensitive and must be “show” or “noshow”
- The script creates training instances out of all images in the ingroup and outgroup and then creates a model only usable by svm_light

classifyImage_singlemodel.sh:

arg1 – path to image file to classify
arg2 – path to model to be used

NOTE:

- ingroup is colored white and outgroup is colored black
- classification accuracy of the model on each segment can be found in the classification directory

classifyImage.sh:

arg1 – path to image file to classify

NOTE:

- The models used are hard coded into the script. Group color output into the terminal are also into stitchit.py
- all classification accuracies can be found in the classification directory if you’re curious to see how each individual model did

‘

execute.py:

arg1 – path to image file for feature extraction
arg2 – name of file to hold instances
arg3 – showFlag (show/noshow)

NOTE:

- showFlag is case sensitive and must be show or noshow otherwise it will default to noshow

<u>Flow of execution:</u>

1. supply training images via categories, ingroup and outgroup directories
2. run create_model.sh to create the model for that group
3. classify an image using that model with classifyImage_singlemodel.sh
4. look at results on terminal

<u>How to add your own categories:</u>

1. go to categories directory and make a new folder with your desired category name
2. go into the folder and create the folders ingroup and outgroup
3. supply the images for training
4. edit classifyImage.sh and create a new variable model6 and the path to the model you will create
5. make the following 3 changes to classifyImage.sh
    - 1. ./svm_classify tmp2.txt $model6 classification6.txt > classification_accuracy6.txt
    - 2. change line 27 to: python stichit.py $imageFileIn classification1.txt classification2.txt classification3.txt classification4.txt classification5.txt classification6.txt
    - 3. add the line: mv classification_accuracy6.txt $CLASS_Dir/classification_accuracy6.txt after line 32

6. make the following changes to stitchit.py:
    - Create a new group color
    - add the color name to COLORS
    - add the group name to GNAMES
    - add the new group color to GROUPS