**Barre Image Classification Program**

Xin Shane Li (LSU-GVC) v1

Walker Shurlds (USDA SRRC) v2

Thomas Miller (USDA SRRC) v3

2016-2018

# Setting up the Environment

**Step 1.** To run this program, Python 2.7 should be installed in the computer. First, install python 2.7:

<https://www.python.org/ftp/python/2.7.11/python-2.7.11.msi>

Then, run the following commands in a Command Prompt Window:

* **pip install numpy**
* **pip install Tkinter**
* **pip install ttk**

**Step 2.** After installing Python, download the following file from:

[www.ece.lsu.edu/xinli/Release/cv2.pyd](http://www.ece.lsu.edu/xinli/Release/cv2.pyd)

Then, copy this downloaded file ‘cv2.pyd’ to the following folder:

C:\Python27\lib\site-packeges.

# Starting the classifier program

1. Copy all the images into the folder ‘data’ (at the same directory of ‘ProcessBarreV1.py’).
2. To run classifier, in a command window, go to folder where you put ProcessBarreV1.py, type: **python ProcessBarreV1.py**

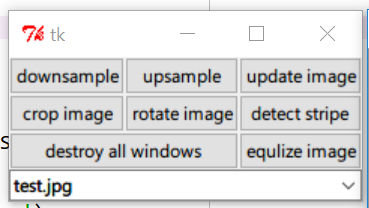
A GUI window will pop out.

* Actually, **python** is not in our PATH unless Hans fixed it. So type:

C:\Python27\python.exe ProcessBarreV1.01.py

# Using the classifier program

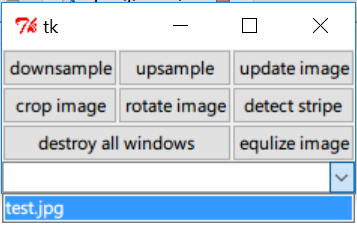
The controlling panel looks like this:



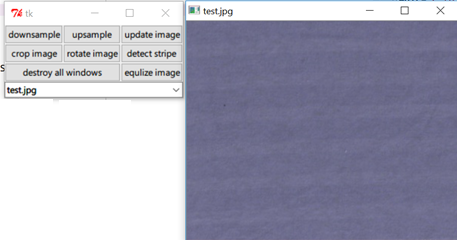
There are eight buttons and a listbox. Use the listbox to select the image to show . All the corresponding operations from the bottoms will be applied to this picture accordingly.

The following example illustrates how the entire classifier pipeline works.

1. select the test.jpg from the listbox:



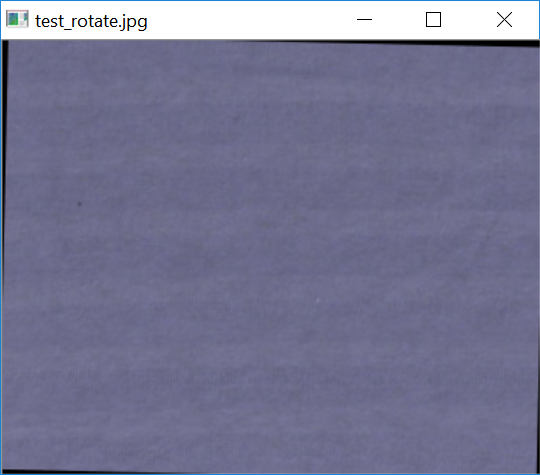
1. After the selection, the image will be shown:



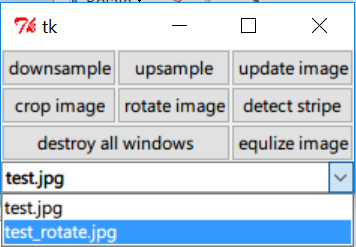
1. To detect stripes reliably, we need to rotate the image to make the stripes horizontal. (This is done using two points on a same line that should be horizontal after rotation, such as the following two points )



Press the ‘rotate image’ button, then pick the first point by click down the mouse button, hold on and move to the location of the second point. Once you release it the rotated image will pop up:



Press the ‘update image’ button, this rotated image will be added into the listbox, as follows:



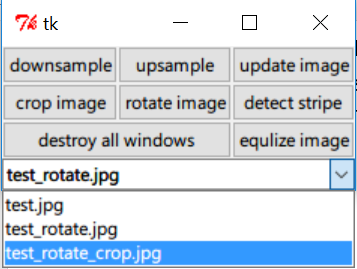
1. After rotation, some region on the boundary is black, which should be cropped out. To do this, first pick the ‘test\_totate.jpg’ in the listbox; then, press the “crop image” button, you need to draw a canonical window by specifying its two corner points in the image ‘test\_rotate.jpg’.



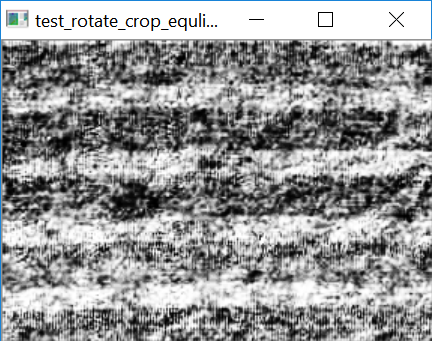
To specify the top-left and right-bottom corner points, similarly, pick the first point by pressing down the left bottom of the mouse, move to the second point then release it. The cropped the image will pop up:



Now, again, press the “update image” button, the “test\_rotate\_crop.jpg” will be added into the listbox.



1. “Equalize Image” button is used to enhance the contrast of the image, so that you can see the potential stripe pattern more clearly. Press the “equalize image” button, you will see an image like the following.



1. Finally, to run the stripe detection, first select the preprocessed image “test\_rotate\_crop.jpg” from the listbox, then press the ‘detect stripe’ button. The pattern, if detected, will be rendered as follows:



# Note: More about Image Data

As mentioned above, all the barre images should be put in the ‘data’ folder where the “ProcessBarreV1.py” locates. During the above processing procedure, all calculated image results are stored in the same ‘data’ folder in ‘jpg’ format.

To add more images, simply copy them into the “data” folder and press “update image”. This will detect all the images placed in the “data” folder and update the list accordingly.

If you would like to delete any temporary images, simply delete those images from the folder. If you re-run a step, the newly generated image will replace the old one.