Description of Pixit script parameters

Gabriel Stela Ballester

June 27, 2016

This is the parameter description for each argument of the Pixit program. Note that the position of the arguments start with the position 0, wich is the first single number. The parameter you need to adapt to your environment are: 4, 7, 34 and 44. The parameters that might have strong impact on the recognition result are: 5, 10, 11, 12, 22 and 23.

Example for extract 100 subwindows of sizes included between 50% and 100% of image sizes, resize them to 16x16 and encode them in graylevels, takes half of the images per class for training (cross-validation), remaining half for testing and repeat 10 runs; build 10 extra-trees using nmin=2 and random tests=16, using simple thresholding within tree nodes:

```
java -Xmx2500m -classpath iclass.jar;ifilter.jar;utilities.jar;libsvm.jar;.
Script_cv 1 0 1 1 F:\YourImages\ 100 16 F:\YourImages\ 100 16 10 16 2 90
90 1 1 0 0 0 0 CV-STRAT-PROP 50 100 0 0 0 0 GRAY 0 50 10 0.0 100
C:\tmp\ZEBRACROP 0 C 0 0 CLASSIC 0 1 NONE C:\pixit\libsvm\ 0
NONE 1 1 SIMPLETHRES 0
```

Argument number: description

0: set 1 if you want to build the database of subwindows extracted in images in the training directory, 0 otherwise. it is usually set to 0 to to run experiments with the same subwindow extraction scheme but with different model parameters

1: set 1 if you want to build the database of subwindows extracted in images in the testing directory, 0 otherwise.

2: set 1 if you want to build the classification model (extra-trees...), 0 otherwise

3: set 1 if you want to evaluate the recognition performances of the model on the test set, 0 otherwise

4: to specify the main directory where you put training images (this directory has to contains as much subdirectories as there are classes, eg: maindirectory/class0 ; maindirectory/class1 ; ...)

5: the number of subwindows extracted in each training image that will be written in the binary db file. recommended value > 100

6: the fixed-sized used to resize subwindows extracted in the training images (recommended value: 16 if random sizes mode)

7: to specify the main directory where you put test images. it has to have the same structures (ie. subdirectories) than the training directory. only useful in evaluation protocol "S".

8: the number of subwindows extracted in each test image that will be written in the binary db file. note that in CV evaluation mode, this parameter is not taken into account as there is no additional database. recommended value > 100.

9: the fixed-sized used to resize subwindows extracted in the test images (recommended value: 16 if random sizes mode).

10: the number of trees to build (T) or random vectors recommended value: >= 10.

11: the number of random tests evaluated at each tree node (k) recommended value: in gray-encoded subwindows ¿= 16, in color-encoded subwindows: ¿= 28. In mode "FERNS", for random vectors, it denotes the number of tests per vector (recommended value: 40).

12: the stop criteria based on the minimum node sample size (nmin), ie. the minimum number of objects in a tree node we accept before splitting this node. in model mode "C" the recommended value is 2., in mode "BAGS" the recommended value is around 5000 if the total number of training subwindows is 1 million (divide or multiply it proportionnaly to this total number). It has no influence in mode "FERNS".

13: the number of images we will use from training directory to build the model (in mode "S"). in general we use the total number of training images.

14: the number of images we will use from test directory to test the model. in general we use the total number of test images.

15: set 1 if you want to extract subwindows of random sizes in training images (random sizes modes), 0 if you want to extract subwindows of fixed-size). recommended value: 1.

16: set 1 if you want to extract subwindows of random sizes in test images (random sizes modes), 0 if you want to extract subwindows of fixed-size). recommended value: 1.

17: set 1 if you want to rotate subwindows (random angles) in training images, 0 otherwise. recommended value: 0 except if your images have high rotation variance and you are able to extract very large set of subwindows.

18: set 1 if you want to rotate subwindows (random angles) in test images, 0 otherwise. recommended value: 0 except if your images have high rotation variance and you are able to extract very large set of subwindows.

19: set 1 if you want to apply a shear transformation in training images, 0 otherwise. recommended value: 0.

19: set 1 if you want to apply a shear transformation in test images, 0 otherwise. recommended value: 0.

21: evaluation protocol. set "S" for independant test set mode (ie. model built on training

set of subwindows; evaluated on independant test set of subwindows). "LOO" for leave-one-out-mode. CV for % of training images randomly choosen. "CV-STRAT-NB" to use NB randomly choosen images from each class for training (remaining images for test); "CV-STRAT-PROP" to use PROP% of images from each class for training (remaining for test).

Recommended value: if your database (< 1000) is small CV-STRAT-PROP, if you database is big then you can split it into separate training and test directories.

22: minwidth, the minimum size (in % of image size) of random subwindow sizes. recommended value: this is the parameter which is the most dependant on the problem at hand. so evaluate the default values for minimum and maximum sizes 0-100 (random subwindows of all sizes), then small subwindows 0-50, then large subwindows 50-100, then fine-tune according to results.

23: maxwidth, the maximum size (in % of image size) of random subwindow sizes. recommended value: see minimum size.

24: nb. of additional features. recommended value 0. set 3 if each subwindow is also described by posx,posy,width, =2 if posx,posy.

25: this parameter is deprecated. (model no used if we build one tree per execution)

26: the minimum rotation angle (between 0 and 360) to apply to training subwindows before resizing if in rotate mode. recommended value: depends on the rotation variance in your data, but in general no rotation is applied (0).

27: the maximum rotation angle (between 0 and 360) to apply to test subwindows before resizing if in rotate mode. recommended value: depends on the rotation variance in your data, but in general no rotation is applied (0).

28: the image color space used to encode subwindows. "GRAY" for graylevels (using CIE709 recommendation), GRAYV (using max(r,g,b)), GRAYL (using (0.5*(min(r,g,b)+ max(r,g,b)))), "HSV", "RGB", "TRGB" (normalized color distributions for each channel within each resized patch), "OPP" (opponent color space). recommended values: HSV or TRGB for color images where color could be discriminant; GRAY for other images.

29: set 1 to storeCoordinates and subwindows in text file, 0 otherwise. 0 is the recommended value.

30: set the proportion (%) of training images per class to use in cross-validation mode CV-STRAT-PROP, or the number of training images per class in CV-STRAT-NB.

31: set the number of cross-validation runs. recommended value: if the error rate standard deviation is high, you should have a large number of runs so difference in results could be considered significant.

32: set the filtering threshold for individual subwindows. by default use 0.0 (values: -1.0: expected mean probability, 10.0: use maximum probability, 11.0 vote for maximum class, otherwise double value between 0.0 - 1.0)

33: the number of test subwindows that are effectively used for testing (¡= number of extracted test subwindows). by default set the same number than extracted test subwindows.

34: set the temp directory where to output temporary files (ex: /tmp/). This directory has to contain two subdirectories: /tmp/model/ and /tmp/bd/

35: the prefix to identify database name that will be used for temporary filenames.

36: set 1 if you want to highlight subwindows classification in original images (in ”C” classification mode), 0 otherwise. This will create image files in the temp directory.

37: problem mode: ”C” for image classification using extra-trees (et-dic), ”BAGS” for image classification using extra-trees as ”codebook” generation. ”R” for content-based image retrieval using trees. ”FERNS” for content-based image retrieval using random vectors. recommended value: both modes ”C” and ”BAGS” should be evaluated on each dataset.

38: this parameter is deprecated and only specific to the UKBENCH database. use default value 0. (nrblock: number of images per class block to retrieve)

39: the number of images to show (top-rank list) in retrieval. Note that in retrieval mode, HTML files are generated in the temp directory containing retrieval results.

40: Feature encoding in BAGS mode (CLASSIC by default, BINARY, HIERARCHICALBINARY, HIERARCHICAL), in C mode: set ”CLASSIC” by default.

41: this parameter is deprecated, use default value 0. (for kernel mode, the number of subwindows per image used to build the kernel-tree (to identify file))

42: this parameter is depcrecated, use default value 1. (the number of subwindows that are concurrently extracted ie. one subwindow descriptor contains multiple subwindows)

43: the file format to output the image ”bag-of-words” representation in problem mode ”BAGS” and ”R” (JAVADB or LIBSVM). Use NONE for C mode and LIBSVM for BAGS mode.

44: the directory where are libsvm (or other external software) binaries (eg: ”/home/grid/maree/libsvm/”).

45: in CV evaluation protocol mode, specify a fixed maximum number of test images per class. use default value 0 to take all remaining images in the class; use e.g. 25 in caltech-256 standard protocol so that highly populated classes do not influence final results too much.

46: the final learning algorithm used in mode ”BAGS”: LIBSVM or LIBLINEAR. Recommended value: Use NONE for C mode, for BAGS mode use LIBLINEAR for faster execution times. (Note: LIBSVM/LIBLINEAR parameters are hard-coded in the Script_cv.java file)

47 number of pages to use in multi-page tiff files, so each subwindow is described by the pixel values from the different pages. Recommended value: 1 if you don’t have pages.

48 algorithm to resize subwindows to fixed-size patches: 1=nearest neighbor, 2=binlinear interpolation, 3=bicubic interpolation. Recommended value: 1 (faster).

49 tree node test type (default: simplethres, other possible values includes diff, difftwopixlbp, random)

50 deep mode (default: 0): deep mode 1 generates new image representation based on spatial distribution of bof (bof maps)