Mixture Feature Model of Gender Identification from Off-line Handwritings

Kaili Zhang
M.S Computer Science
Columbia University
kz2203@columbia.edu

Hebo Yang
M.S Computer Science
Columbia University
hy2326@columbia.edu

Tony Jebara
Associate Professor
Columbia University
jebara@cs.columbia.edu

Abstract

In this paper, we build a comprehensive model to identify gender from a series of handwriting images. Firstly we build a mixture feature model including both local features and global features to represent potential sexual differences. Then, we use a typical Random Forests model to generate label and probability for each training samples, also a frequency is generated for each feature. We use labels and probabilities to build importance weights for samples while use frequency to build weights for features in order to improve performance of our final Weighted Support Vector Machine model. Experimental results indicate that this mixture model for both features and weights performs well in gender identification.

1. Introduction

It is well known from everyday life, that sex, handedness, mood state and some other factors have impact on the appearance of handwritten text. Persons that are in contact with many handwritten documents (such as teachers), quickly develop the ability to distinguish male and female handwriting. Many studies have been conducted in this field [1]-[6], and it has been shown that even untrained human examiners are able to guess the sex of the writer based on text samples, with the percentage rate up to 70%. Reports of handwriting find that female handwriting has greater circularity [1], and it is more 'delicate and decorative' than for man [2]. Typically, handwriting attributes such as 'large', 'rounded', 'neat', 'regular/consistent' and 'carefully executed' writing belongs to females and 'sloping', 'spiky', 'confident', 'hurried' and 'untidy/scruffy' are significantly more often used to describe male scripts [3]. This is also a common understanding of handwriting of gender differences of most people.[7] Males and Females across cultures are also rated as different. Some research has shown that even hormonal influences affect male and female handwriting performance [4]. The identification of writer gender from handwriting is an important element of the field of forensic studies [8].

However, our paper do not discuss much about whether handwritings containing sexual differences or not, we only assume they do and then build a mixture model to identify gender from them.

Previous researches about gender identification including writer identification from off-line handwritings are mainly concentrating on feature extraction from single level: local or global. For example, in [8], author only used shape descriptors (local features) as model features; in [9], author use character local feature to describe differences between individuals; in [10], author build fuzzy features for characters which corresponding to our global features. However, in our paper, we extract features from both local and global level and build a more comprehensive feature model for all handwritings in our database.

Furthermore, we also do a lot effort on classifier used, especially a modification to introduce weight in our model. The mixture not only comes from a mixture of different weights, but also comes from a mixture of different popular classification models.

2. Dataset

Our dataset original dataset is a set of images and their corresponding chain code features. We have to pre-process the data before extracting feature and building the model.

2.1 Description

Our dataset consists of images of the offline handwritten documents produced by 282 writers, upon which 139 are male and 143 are female. Each writer has four documents: 1. Some random text writing in Arab which varies from one writer to another. 2. Same text writing in Arab for all writers. 3. Some random text writing in English varies from one writer to another. 4. Same text writing in English for all writers. Figure 1 is an illustration of the images from our dataset. We used Arab and English writings so that our model can be generally applied to different but similar languages.

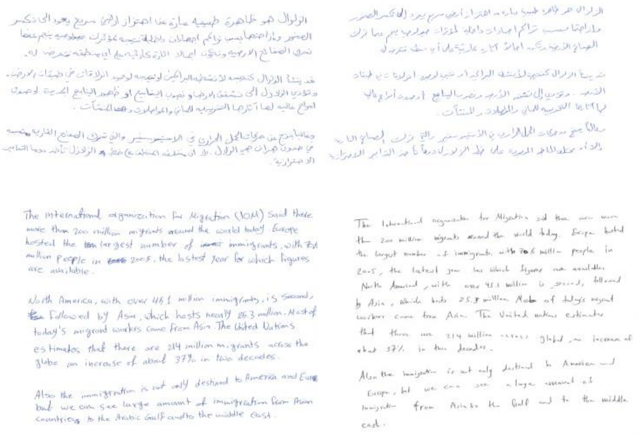

Figure 1: The left images are the same text writing by a female writer in Arab and English. The right are the corresponding ones from a male writer.

2.2 Pre-processing Data

In this section, we give three level of segmentations used to segment an integrated handwriting picture into strokes and blocks before extracting feature from segmentations.

We have the RBF images of handwriting. These kind of images are used to extract feature ’Pressure’ in section 3.3. Then, we calculate the gray image from RBF for all handwritings in database. Integrated gray image are used to extract feature ‘Gravity’. Also, binary images can be generated by Otsu’s Method.

In order to minimize size of images, we cut the blank area of and only reserve rectangle areas covering handwritings. To implement this idea, we calculate sum value for each row and column in binary images. Those blank areas are all continuous rows and columns with sum value equal to zero. Meanwhile, pixel’s sum value can also be used to segment rows and columns which could generate separated blocks in this way.

As a whole, we have three format kind of image (RBF, gray, binary) and four size level of segmentations (integrated, blank-cut-off, rows, blocks). Following feature extraction section are based on this new dataset described above.

3. Feature Extraction

We primarily focus on our research in the area of feature extraction because we believe that the selection of model defines the upper limit of the prediction accuracy but well-designed features give us a high lower limit of the prediction accuracy. The following subsections described the local and global features of interest.

3.1 Local Feature Extraction

3.1.1 Directions

Several studies have shown that directions or slant is a prominent feature to categorize writing style. We first binarize all the images data using Otsu’s Method so that each pixel is represented by either 1 or 2.

Since each stroke is more than one pixel thick, we then thin the writings into skeletons by removing all the other pixels. The thinning is achieved by two steps suggested by Suen and Zhang [11]. This algorithm has been proven to be very efficient and fast in thinning of digital patterns.

We consider the image as a matrix and define the connected eight neighbors of a pixel $P_{i,j}$ as

$P_1 = P_{i-1,j-1}$	$P_2 = P_{i-1,j}$	$P_3 = P_{i-1,j+1}$
$P_8 = P_{i,j-1}$	$P_{i,j}$	$P_4 = P_{i,j+1}$
$P_7 = P_{i+1,j-1}$	$P_6 = P_{i+1,j}$	$P_5 = P_{i+1,j+1}$

First we remove the pixel $P_{i,j}$ if it satisfies the following conditions:

(a) $2 \leq$ N $(P_{i,j}) \leq 6$
(b) A $\left(P_{i,j}\right) = 1$
(c) $P_2 * P_4 * P_6 = 0$
(d) $P_4 * P_6 * P_8 = 0$

where $P_{i,j}$ is a pixel of the whole image in a matrix, N $(P_{i,j})$ is the number of non-zero neighbors of $P_{i,j}$ in the 8 connected neighbors, A $\left(P_{i,j}\right)$ is the number of 01 patterns in ordered set $P_2, P_3, \ldots P_{8,}P_1$

Then we run another round of iteration and remove the pixel $P_{i,j}$ if satisfies the following conditions:

(a) $2 \leq$ N $(P_{i,j}) \leq 6$
(b) A $\left(P_{i,j}\right) = 1$
(c) $P_2 * P_4 * P_8 = 0$
(d) $P_2 * P_6 * P_8 = 0$

The skeleton is then segmented at its junction pixels, which are just pixels having three or more neighboring pixels. The segmentation is improved by using the algorithm described by Jayarathna and Bandara [12].

Then we moved along the pixels of the obtained segments of the skeleton using an order favoring the four neighbors in the order of P_4, P_6, P_8, P_2 and then the other four pixels P_3, P_5, P_7, P_1. For each pixel, we used eleven neighboring pixels. A simple linear regression can be used on these pixels to obtain its tangent, which is a good estimation for the direction feature we need. The process is demonstrated in figure 2.The PDF of the resulting directions for each pixel is computed as a probability vector. We construct the features with 4, 8, 12 and 16 directions.

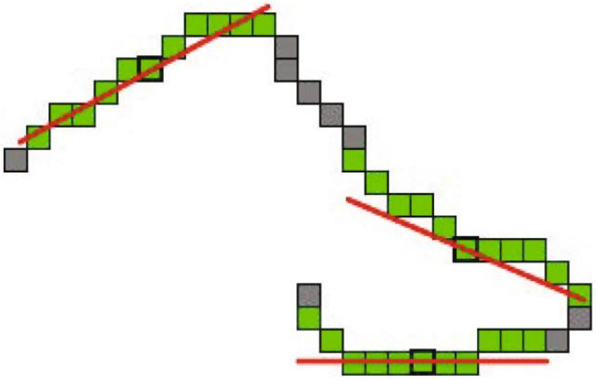

Figure 2: The heighted squared green points are junction pixels and the green points around them are the identified neighboring used in regression. The red lines are the tangent lines.

3.1.2 Curvature

Curvature is another commonly accepted characterizing feature in forensic document examination. We first used Moore Neighbor-Tracing algorithm to identify the contour of the words. Note that the identified contour is useful in many other feature extraction processes later on. The Moore Neighbor-Tracing algorithm is as following:

(a) Start from the most bottom-left black pixel s, pop s into the set of the contour points.
(b) Backtrace s according to the defined order (P_1 *to* P_8) to the white pixel before s.
(c) Search around s in the defined clockwise order, visiting each its neighbor pixel until finding a black pixel.
(d) Pop the black pixel into the set of contour points and backtrace the black pixel.
(e) Repeat (c) and (d) until reaching the start point s.

For each pixel p belonging to the contour, we consider a square neighboring window of size 10*10. Inside the window, we compute the number of background pixel n_1 and the foreground pixels n_2. The curvature C can thus be measured as:

$$\mathrm{C} = \frac{\mathrm{n}_1 - \mathrm{n}_2}{\mathrm{n}_1 + \mathrm{n}_2}$$

The computation of curvature is demonstrated in figure 3. The PDF of curvatures is computed in a probability vector of size 100.

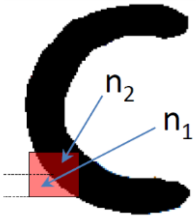

Figure 3: The heighted squared window in red is the 10*10 neighbor window of pixels.

3.1.3 Chain code

Chain code features characterize detailed distribution of curvatures in the handwriting and are generated by browsing the contour of the text as described in section 3.1.2 and assigning a number to each pixel according to its location with respect to the previous pixel. We used the chain-code of 8 directions and order 4 which lead to a probability vector of size 4096. This feature set is already provided with the dataset we have.

3.1.4 Tortuosity

Tortuosity is also a greatly used feature which distinguishes between fast writers who produce smooth handwriting and slow writers who produce tortuous or twisted handwriting. We considered but didn't implement its extraction since we believe it is not significant in the case of gender identification.

3.2 Feature selection for Local Features

Due to the large number of features we have for local features, we have to implement a vast and efficient algorithm to reduce the dimension of features.

We converted our dataset into ARFF(Attribute-Relation File Format) and used FST3lib, which is a standalone widely applicable C++ library for feature selection capable of reducing problem dimensionality to maximize the accuracy of data models, performance of automatic decision rules as well as to reduce data acquisition cost. The toolbox framework allowed us to choose and combine various criteria to implement feature selection. We used a sub-optimal algorithm which runs fast on large feature space provided in an example. Features are selected using the Sequential Forward Selection (SFS) procedure to maximize the

Generalized Mahalanobis probabilistic class distance. Generalized Mahalanobis is evaluated on the first 50% of data samples. The selected subset is eventually verified by means of 3-NN classifier accuracy estimation on the second 50% of the data [13].

3.3 Global Feature Extraction

Besides local features, global features are also important indexes in representation of gender differences. We developed most global characters on our own and build eight global features: Gravity, Area, Compactness, Creation, Row size, Column size, Gradient, Pressure.

Center of Gravity: Treats 2D image as a real object and find its center of gravity by using Fourier Transformation;

Area: Represents the ratio of handwriting area to whole page area;

Compactness: Consisted of Row spacing and Column spacing (including variance of these four features). Those together indicate the compactness degree of handwritings. Row spacing represents the average space between each two rows and Column spacing represents the average space between each two blocks;

Creation: Represents the amount of creation blocks. When people write wrong things, they place crosses and then blocks of masses generated. We calculate these blocks as an indicator of neat degree of handwritings. Creation blocks can be found simply by compare the average pixel value of blocks in binary type, and then pick those high valued blocks as creation blocks.

Row size: Represents the average height of each row;

Column size: Represents the average width of each block;

Gradient: Consisted of three indexes representing three different directions, up, down and horizon. We calculate the gradient of every row and classified them into three different directions. Then calculate the ratio of each direction to form a three column array.

Pressure: Represents the change in pen writing pressure which means the stability of a writer using a pen. Pressure here is equals to the variance of pixels' value in original gray image.

Together, these features formulate an array with fifteen columns. We call them global features.

4. Model

In this section, we mainly described our learning model which is a combination of Random Forests (RF) model and Weighted Support Vector Machine (WSVM) model. We use typical RF model to generate labels and weight for samples, frequencies for features. Then we use WSVM to generate the final labels for gender identification. Also, detailed ideas are described in this section.

4.1 Random Forest Model

Random Forest (RF) is an ensemble learning method for classification and regression that operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes ouput by individual trees. The algorithm for inducing a random forest was developed by Leo Breiman and Adele Cutler, and 'Random Forests' is their trade mark. The term came from random decision forests that were first proposed by Tin Kam Ho of Bell Labs in 1995. The method combines Breiman's bagging idea and the random selection of features, introduced independently by Ho and Amit and Geman inorder to construct a collection of decision trees with controlled variation. The selection of a random subset of features is an example of the random subspace method, which, in Ho's formulation is a way to implement stochastic discrimination proposed by Eugene Kleinberg. It is better to think of RF as a framework rather than as a particular model. The framework consists of several interchangeable parts which can be mixed and matched to create a large number of particular models, all built around the same central theme [14].

Using RF model, we can easily gain following advantages. Firstly, decision label and probability can be acquired at same time. These two kinds of results are very important for our model since they are used to design weight of samples for WSVM, the next step and the final step for gender identification. Secondly, by RF's own bagging algorithm, it can provide different feature combinations for each decision tree, so RF model can at some degree reduce the influence of feature selection. Thirdly, with the quick running time, we can use RF to improve accuracy of single WSVM model with little cost.

Therefore, the output of RF for each sample x_i, we have a probability p_i and a decision label k_i.

Also, we can get a statistical result of appearance frequency for the best accuracy case after k-fold validation method, which is set as f_i.

4.2 WSVM Model

The WSVM is used to model this two-class problem. Usually, the number of instances for female and male are balanced. So there exists no class imbalance problem. In this binary classification problem, the parameters of

WSVM model is chosen by the grid search based on k-fold cross validation. The model's parameters include C and kernel parameters. For example, they are C and g if the kernel function is GRBF. It is unknown which pair of parameters are best for a given problem. Therefore, we must carry out the parameter selection before training process. At present, the most commonly used parameter selection method is the grid search based on k-fold cross validation. In k-fold across validation, the training set is first divided into k subsets of equal size, then one subset is tested using the classifier trained on the remaining k-1 subsets sequentially. In grid search, the parameter space is first divided into a series of grids, in which a grid corresponds to a pair of parameters. Then an exhaustive search through a specified range is used to find the pair of parameters with the highest cross validation accuracy or the lowest testing error.

After the parameters are selected, the training set is inputted to train WSVM model. Because WSVM model is a quadratic programming (QP) problem, training a WSVM model means the solution of a QP problem. The sequential minimal optimization proposed by Platt is a fast algorithm in the solution of large QP problem. The main two parts of SMO are the solution of two Lagrange multipliers and working set selection. Many scholars proposed some improvement rules in working set selection.

Once the model for binary classification is trained, it is used to create the final gender label for handwritings. This model is implemented by modifying the SVM lib [15].

In the features of the sample vector, some features of strongly correlated with the classification, some features of weakly correlated with the classification, or uncorrelated. If we do not consider the different importance of different features for the classification, kernel function may be dominated by weakly related or not related to features. Particularly, in our case, volume of micro features is much bigger than volume of global and fuzzy features. It will reduce the performance of SVM. So, we introduce weight to features according to some works.

For a general kernel, it is difficult to interpret the SVM weights, however for linear SVM there actually is a useful interpretation. Recall that in linear SVM, the result is a hyper plane that separates the classes as best as possible. The weight represents this hyper plane, by giving the coordinates of a vector which is orthogonal to the hyper plane. These are the coefficients given by SVM and let's call itw. What can we do with the weight vector w? Its direction gives us the prediction class, that is to say that if we take the dot product of any point with this vector, we can tell on which side it is: if the dot product is positive, it belongs to the positive class; otherwise, it belongs to the negative class. Moreover, we can even learn something about the importance of each feature. Suppose SVM would find only one feature useful for separating data, then the hyper plane would be orthogonal to that axis. So, we could say that the absolute size of the coefficient relative to the other ones gives an indication of how important the feature was for the separation. For example, if only the first coordinate is used for separation, w will be of the form $(\mathrm{x}, 0)$ where x is some nonzero number and then $|\mathrm{x}| > 0$.

In our general gender identification model, we use the above described linear kernel to set weight for features. The way to set features is to change value of coefficient vector of hyper plane (just like what we did in Project 1 and Project 2, especially the part to change kernels). We give those global features a higher weight than micro features based on the unbalanced amount.

According to the above-mentioned method, we get the weight value of each feature, then we construct feature weighting matrix P as follows:

$$\mathrm{P} = \begin{bmatrix} w_1 & & \\ & \ddots & \\ & & w_n \end{bmatrix}$$

P is an n rank opposite angles array, $P_{ii} = w_i$ represents the weight of feature i.

$$w_i = \begin{cases} f_i & for\ micro\ feature \\ f_i \times 5 & for\ global/fuzzy\ feature \end{cases}$$

When we predict category of a sample, the different training samples have different importance for classification. In our case, the higher probability the RF model give, the more important this training sample is to the classification (Perfect samples). We need to reduce the interference of the noise samples and isolated points. To solve this problem, we give a greater weight to the training samples with high probability like 'Perfect samples', 'Good samples', and give a smaller weight to the training samples with low probability like 'Bad samples'. Here we added a membership function in the conventional SVM sample. All samples are fuzzed by membership function. Fuzzy training set is expressed as follows:

$$\mathrm{D} = \{(x_1, y_1, u_1), \dots (x_l, y_l, u_l)\},$$
$$\mathrm{x} \in \mathrm{R}_n, 1 \le i \le l$$
$$\mathrm{y} \in \{1, 0\}$$
$$0 \le u_i \le 1$$

Here u_i is membership function of sample x_i, each sample has its u_i. For this training set, through Lagrange multiplier and kernel function, the problem that find the best separating hyper plane of SVM can be transformed into the dual problem as follows:

$$\max W(\alpha) = \sum_{i=1}^{l} \alpha_i - \frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{i} \alpha_i Q(i,j)\alpha_j$$

$$\mathrm{s.t.} \sum_{i=1}^{l} y_i \alpha_i = 0,$$

$$0 \le \alpha_i \le u_i \times C\ , i = 1, \dots, l$$

Punishment parameter C is a constant, and u_i will be fuzzed C. $u_i \times C$ will set different punish parameter for each sample. It can be drawn from the function above that the greater $u_i \times C$ is, the smaller the possibility of misclassification of sample x_i will be. To the important samples, we set a greater value for its u_i. To the noise data and isolated points, we set a smaller value for its u_i and $u_i \times C$ also will be small [16].

In our paper, we set u_i for samples based on p_i given by RF model and label k_i.

$$u_i = \begin{cases} 0.5 & for\ k_i \neq y_i \\ \dfrac{(p_i - 0.5)}{0.5} & for\ k_i = y_i \end{cases}$$

5. Experiments and Result

In this section we describe the experimental setting we designed and the consequent results obtained from the comparison. Our previous assumptions are supported and they imply that our model has a reasonable performance.

5.1 Experiments

Our database contains handwritings 71different individuals; each individual has two English writings and two Arabic writings. We extract all features and form a feature set. We tried to separate our database into four parts; however, the results of experiments of the four parts differ little. So we denied this separation and just treat all handwritings as integrity.

Then, randomly select three quarter samples as training set and a quarter samples as testing set. Then use our model to train and set k-fold cross validation as 3-fold. Finally, we repeat this procedure for 20 times so that a more general result could be acquired.

All the feature extraction programs are coded by our team and we implement RF model simply by a RF toolbox on Matlab. For WSVM model, we use Lib-SVM as prototype and make complex modification on it.

5.2 Result

Since our final model is designed based on SVM, we use SVM to test our experimental trials.
Following is the predict accuracy for different feature types by typical SVM with not weight parameters.

Table 1. Accuracy for Different Feature Types

Feature Type	Accuracy
Local	61.03%
Global	66.25%
Combined	67.11%

This result can strongly support our settings for feature weight which gives higher weight for global feature than local features.

By applying random forest model, we gain an accuracy of 69.37%. Also, the distribution of probability can also be found as the following table shows.

Table 2. Distribution of Label Probability

Sample Type	Probability	Ratio
Perfect	$p_i \geq 85\%$	21%
Good	$85\% > p_i \geq 65\%$	62%
Bad	$65\% > p_i \geq 50\%$	17%

We can treat these bad samples as unidentified handwritings which lacks characters of sexual differences. If we drop off those bad samples, we can improve the accuracy to 78.77%. This improvement indicates that those Bad samples are really hard for identification and maybe even person could not tell the right category of them.

Table 3. Accuracy for Different SVM Type

SVM Type	Accuracy
Non-weight	67.11%
Feature weights	72.23%
Sample weights	74.98%
Both weights	76.17%

Without weight, if we only implement the typical support vector binary classifying machine, the accuracy becomes a terrible 67%, only a little higher than random classification. And then, accuracy is accelerated prominently by adding weight for features and samples individually into our model and reaches highest when both weights are added.

Our experiments proved all our assumptions and give a reasonable result.

6. Conclusion and Future works

In summary, our works are concentrated in three mixtures. One is the mixture of local and global features; one is the mixture of two classifiers; one is the mixture of two weights. All of these three mixture works well even though we could not give concrete mathematical prove so far. However, that's the rule of engineering: when everything works, no one knows why.

For future works, we have several ideas. Firstly, we need to construct concrete mathematical justification for our model settings. Until now, all settings come from experimental experience and intuitive inspirations. For example, actually, we could not make sure weather the result of Random Forests is correct or not. So when we generate weights according to result of RF, it is not credible in math even if we gained satisfying accuracy. Secondly, some new features should be extracted from original handwriting images. Features are always the most important part defining the upper bound of a classifier. More features extracted more space to select a better combination. Also, the weight in our model could also be modified in future improvement.

Our thanks to ACM SIGCHI for allowing us to modify templates they had developed.

Reverences

[1] D. Lester, S. McLaughlin, R. Cohen, and L. Dunn, 'Sex-deviant handwriting, femininity, and homosexuality,' Perceptual and Motor Skills, vol. 45, p. 1156, 1977.

[2] S. Hamid and K. Loewenthal, 'Inferring gender from handwriting in urdu and English,' The Journal of Social Psychology, vol. 136, pp. 778-782, 1996.

[3] V. Burr, 'Judging gender from samples of adult handwriting: Accuracy and use of cues,' The Journal of Social Psychology, vol. 142, pp. 691700, 2002

[4] J. R. Beech and I. C. Mackintosh, 'Do differences in sex hormones affect handwriting style? Evidence from digit ratio and sex role identity as determinants of the sex of handwriting,' Personality and Individual Differences, vol. 39, pp. 459-468, 2005

[5] J. Hartley, 'Sex differences in handwriting: A comment on spear,' British Educational Research Journal, vol. 17, pp. 141-145, 1991.

[6] W. Hayes, 'Identifying sex from handwriting,' Perceptual and Motor Skills, vol. 83, pp. 791-800, 1996.

[7] Emir Sokic, Almir Salihbegovic, Melita Ahic-Djokic, 'Analysis of Off-line Handwritten Text Samples of Different Gender using Shape Descriptors,' 2012 IX International Symposium on Telecommunications, October 25-27, 2012, Sarajevo, Bosnia and Herzegovina

[8] S. N. Srihari, S. –H. Cha, H. Arora, and S. Lee, 'Individuality of descriptor,' Signal Processing: Image Communication, vol. 17, no. 10, pp. 825-848, 2002

[9] Bin Zhang, Sargur N. Srihari, 'Analysis of Handwriting Individuality Using Word Features,' 7th International Conference on Document Analysis and Recognition, Edinburgh, Scotland, August 3-6, 2003

[10] Ashutosh Malaviya, Liliane Peters, 'Fuzzy Feature Description of Handwriting Patterns,' Pattern Recognition, vol. 30, No.10, pp. 1591-1604, 1997

[11] T. Y. Zhang and C. Y. SUEN, 'A Fast Parallel Algorithm for Thinning Digital Patterns', Commun. ACM, 27.3, March 1984

[12] U. K. S. Jayarathna and G. E. M. D. C. Bandara, 'A Junction Based Segmentation Algorithm for Offline Handwritten Connected Character Segmentation', 2006 International Conference on Computational Intelligence for Modeling Control and Automation

[13] UTIA, Institute of Information Theory and Automation. http://fst.utia.cz/?fst3

[14] Prinzie, Anita; Van Den Poel, Dirk (2008). "Random Forests for multiclass classification: Random MultiNomial Logit". Expert Systems with Applications 34 (3): 1721.

[15] Linkai Luo, Xi Chen, 'Integrating Piecewise Linear Representation and Weighted Support Vector Machine for Stock Trading Signal Prediction,' Applied Soft Computing, vol. 13, No. 2, Feb. 2013, pp. 806-816

[16] Qiongsheng Zhang, Dong Liu, Zhidong Fan, Ying Lee, and Zhoujun Li 'Feature and Sample Weighted Support Vector Machine')