# Multiple One-Class Classifier Combination for Multi-Class Classification

Bilal Hadjadji, Youcef Chibani and Yasmine Guerbai
Speech Communication and Signal Processing Laboratory,
Faculty of Electronics and Computer Science
University of Science and Technology Houari Boumediene (USTHB),
32, El Alia, Bab Ezzouar, 16111, Algiers, Algeria
{bhadjadji, yhibani, yguerbai}@usthb.dz

***Abstract*—The One-Class Classifier (OCC) has been widely used for solving the one-class and multi-class classification problems. Its main advantage for multi-class is offering an open system and therefore allows easily extending new classes without retraining OCCs. However, extending the OCC to the multi-class classification achieves less accuracy comparatively to other multi-class classifiers. Hence, in order to improve the accuracy and keep the offered advantage we propose in this paper a Multiple Classifier System (MCS), which is composed of different types of OCC. Usually, the combination is performed using fixed or trained rules. Generally, the static weighted average is considered as straightforward combination rule. In this paper we propose a dynamic weighted average rule that calculates the appropriate weights for each test sample. Experimental results conducted on several real-world datasets proves the effective use of the proposed multiple classifier system where the dynamic weighted average rule achieves the best results for most datasets versus the mean, max, product and the static weighted average rules.**

***Keywords—one class classifier, multiple classifier system, multi-class classification, dynamic weighted average rule.***

## I. INTRODUCTION

One Class Classifier (OCC) has been designed for training only patterns belonging to the target class distribution. Its main goal is to detect anomaly or a state other than the one for the target class [1], [2]. The assumed hypothesis is that only information of the target class is available. Therefore no information about the potential nature of other classes is needed to derive the decision boundary.

In the last decades, OCC has attractive much attention for many researchers leading to use it for solving the multi-class classification problem [3], [4], [5], [6]. Indeed, extending the classifier to new classes does not require retraining it again on all classes. In addition, the OCC trains only on the target class that allows avoiding the unbalanced data. This is usually appears when the training data of the target class are significantly outnumbered by the other training instances. In this case, separating the target class among the remaining classes is a difficult task.

However, using OCC for the multi-class classification usually achieves less accuracy than the usual multi-class classifiers [5]. Furthermore, due to the high diversity of existing OCC [7] choosing a specific classifier for various applications is a difficult task. Therefore, combining different OCCs is suitable since it can produce a better system in terms of robustness and accuracy. In addition, it allows keeping the offered advantage for achieving an extensible multi-class system. In this case, the most difficult problem is finding the best combination rule.

In order to perform the combination, a Multiple Classifier System (MCS) of diverse classifier must be created, for which different ways are possible. The most popular ways are based on different initialization, different parameter choices, different architectures, different classifiers, different training sets or different feature sets [8]. In the field of combining OCCs for achieving a multi-class classification system, Juszczak et al. [9] use Parzen OCC ensembles for classifying missing data in multi-class problems. In a related work, Muñoz-Marí et al. [10] demonstrate that using a simple combination rules (e.g. average or product) to combine OCCs trained on different feature sets are able to improve the classification accuracy for classifying image remote sensing. More recently, Abbas et al.[11] used the Dezert-Smarandache theory to achieve one-class support vector machine (OC-SVM) ensemble, trained on different feature sets for handwritten digits recognition.

The previous methods are based on using the same type of OCC trained on different feature sets or by different training sets to yield complementary OCCs. However, different feature sets are not always available. Moreover, in some applications, training samples are often reduced, which does not allow generating different training sets.

Hence, we propose in this paper to use an alternative approach, which relies on creating a multiple one-class classifier that is achieved by combining different types of OCC, trained on the same feature set by the same training set for solving the multi-class classification problem. The combination step is performed through the use of fixed and trained rules. For the latter rule, in addition to the widely used weighted average that has been investigated for classifiers combination [12], [13], [14], we propose a dynamic weighted average rule to measure the importance of the used classifiers through calculating the suitable weights for each test sample.

The remaining of this paper is organized as follows. In section 2, we review the used OCCs and their extension to the multi-class classification. Section 3 describes the propose MCS based on OCCs. In order to evaluate the effective use of the proposed approach, experimental results conducted on various

1051-4651/14 $31.00 © 2014 IEEE
DOI 10.1109/ICPR.2014.488

datasets are presented in section 4. Finally, the conclusion is provided in the last section.

## II. Overview of One-Class Classifiers for Multi-Class classification

Different algorithms are addressed for designing one class classification. In this work, three types of OCC are selected, which are the One-Class Nearest Neighbor (OC-NN), one-class neural network which is usually referred to as auto-encoder and also as Auto-Associative Neural Networks (AANN) and One-Class Support Vector Machine (OC-SVM). In the following, we briefly review properties of the used OCC and their extension for multi-class classification.

### A. One-Class Nearest Neighbor Classifier

The one-class nearest neighbor (OC-NN) is a particular case of the OC-KNN, such $K$ is set to one. According to [1,7], OC-NN finds the distance of a test object $x$ to its nearest neighbor in the training set, and the method estimates the density as:

$$dnn(x) = \frac{1/n}{V(\|x - NN^{tr}(x)\|)} \quad (1)$$

Where $n$ is the number of training samples and $V$ defines the smallest volume value with the centre in $x$ surrounding the observation vector nearest to $x$.

A test sample $x$ may either be rejected as being an outlier, or accepted as being part of the target class according to the threshold value defined in the training step.

### B. Auto-Associative Neural Network Classifier

Neural networks are composed of interconnected processing units arranged in one or several layers that can be used to implement a complex functional mapping between input and output variables. The weights of the neural network are adjusted using training data so that an error function would be minimized over the training set.

The basic design of the AANN is ''bottleneck''. Which assumes that the data represented in an $p$-dimensional space is mapped to less dimension and then reproduced for testing the reproduction ability of the model. Usually, the AANN is composed of three layers is designed having $p$ inputs, $p$ outputs and $k$ neurons on the hidden layer, where $k < p$. The AANN is then trained using the standard back-propagation algorithm to learn the identity function over the training set. This design has been used successively by Cottrell and Zipser [15] to produce a compression algorithm and Japkowicz et al. [16] for novelty detection.

Let $S$ defines a training set $S = \{x_1, \dots, x_n\}$, the AANN is trained on each sample in order to produce an identity function $f$ that assigns for each input $x_i$ an output $f(x_i)$, $i = 1, \dots, n$ taking ideally the following form:

$$f(x_i) = x_i \quad (2)$$

The principle of the AANN is to adjust its weights according to the error of reconstruction, which is defined as the distance between output and its corresponding input. Formally, the error of reconstruction is defined as:

$$Er(x) = ||f(x) - x|| \quad (3)$$

Such that, $x \in S$

A test sample $x$ may either be rejected or accepted according to the threshold value defined in the training step.

### C. One-Class Support Vector Machine Classifier

The concept of the OC-SVM consists to find an hyper sphere in which the most of training samples are included into a minimum volume. More specifically, the objective of the OC-SVM is to estimate a function $f(x)$ that encloses the most of training samples into a hyper sphere $R_x = \{x \in R^d \setminus f(x) > 0\}$ with a minimum volume where $d$ is the size of feature vector [17]. $f(x)$ is the decision function, which takes the form as:

$$f(x) = sgn\{\sum_{i=1}^{n} \alpha_i K(x, x_i) - \rho\} \quad (4)$$

Denoting by $\alpha_i$ the Lagrange multipliers computed by optimizing the following equations:

$$\min_\alpha \left\{\frac{1}{2} \alpha_i \alpha_j K(x_i, x_j)\right\} \quad (5)$$

Subject to

$$0 \le \alpha_i \le \frac{1}{\nu n} \quad (6)$$

$$\sum_i^n \alpha_i = 1 \quad (7)$$

$\rho$ defines the distance of the hyper sphere from the origin. $\nu$ is the percentage of data considered as outliers. $K(.,.)$ defines the OC-SVM kernel that allows projecting data from the original space to the feature space.

A pattern $x$ is then accepted when $f(x) > 0$. Otherwise, it is rejected. Various kernel functions can be used as polynomial or Radial Basis Function or multilayer perceptron [18]. Usually, the RBF is the most used kernel, which allows determining the radius of the hyper sphere according to the parameter $\gamma$. It is defined by:

$$K(x, x_i) = \exp(-\gamma\ (d(x, x_i))) \quad (8)$$

Such that,

$$d(x, x_i) = \|x - x_i\|^2 \quad (9)$$

The extension of the OC-SVM to the multi class is proposed by [3], where they use a logarithmic function for calibrating the outputs which is defined as follows:

$$d(x, x_i) = -\log\left(\sum_{i=1}^{n} \alpha_i K(x, x_i)\right) + \log(\rho) \quad (10)$$

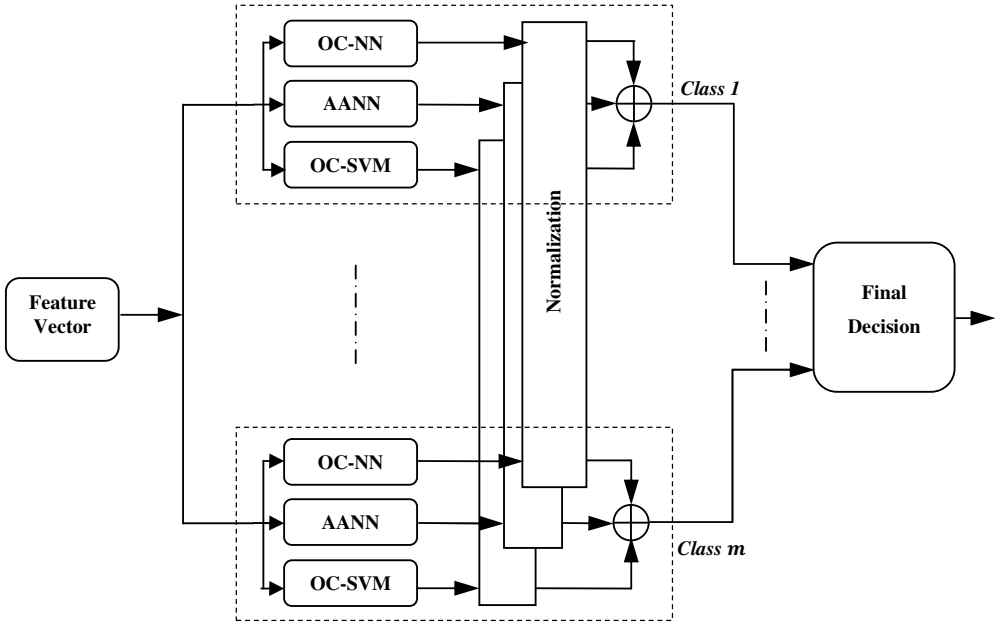


Fig. 1. The proposed MCS architecture

### D. Extension of OCCs for Multi-Class Classifiaction

Extending the OCC to the multi-class is straightforward. Since, for a defined set of $m$ classes $\mathrm{C} = \{c_1, \dots, c_m\}$, each class has its corresponding OCC. After achieving the OCC model of each class, a test sample is assigned to its corresponding class that generates the maximum prediction. The class label $y(x)$ of a test sample $x$ is defined mathematically as follows:

$$y(x) = \arg max\left(OCC_j(x)\right), \text{with } j = 1, \dots, m \qquad (11)$$

Such $OCC_j$ is the one of the OCCs (OC-NN, AANN or OC-SVM).

## III. One-Class Classifiers Combinatiion for Multi-Class Classification

The basic structure of the proposed MCS based on OCC is depicted in figure 1. The achieved MCS is composed of three different types of OCC, which operate in parallel at the same data. A description of each stage of the MCS is given in the following sections.

### A. Normalization of OCC Outputs

Several combination rules are possible to achieve the MCS, but all these rules assume a unique interpretation of the confidence values as a posteriori probabilities for each test sample $x$. Hence, transformation by means of the normalization of each classifier outputs into posteriori probability is required for performing correctly the combination.

Thus, we propose to use the softmax normalization method [19] which is adopted due to its simplicity and effectiveness. This function allows mapping the outputs in the range [0, 1]. It is used for the three classifiers as follows:

- For the OC-NN, the density $dnn_j$ response of each class $c_j$ from $m$ classes are transformed to $P(c_j/x)$ through:

$$P_1(c_j/x) = \frac{exp\left(dnn_j(x)\right)}{\sum_{j=1}^{m} exp\left(dnn_j(x)\right)} \qquad (12)$$

- For the AANN, the reconstruction error $Er_j$ of each class $c_j$ are transformed as follows:

$$P_2(c_j/x) = \frac{exp\left(\frac{1}{Er_j(x)}\right)}{\sum_{j=1}^{m} exp\left(\frac{1}{Er_j(x)}\right)} \qquad (13)$$

- For the OC-SVM outputs $d_j$ of each class $c_j$ are transformed by the following equation:

$$P_3(c_j/x) = \frac{\exp\left(d_j(\mathrm{x},\mathrm{x_i})\right)}{\sum_{j=1}^{m} exp\left(d_j(\mathrm{x},\mathrm{x_i})\right)} \qquad (14)$$

### B. Combination Rules

Various combination rules are possible for achieving an enhanced MCS. In our case, two groups are considered which

are based on fixed [8] and trained combination rules, respectively.

The MCS is defined as a set of $L$ of classifiers which are combined through the following rules:

*1) Fixed Combination Rules*

*a) Mean combination rule:*

$$y(x) = \arg max(1/L \sum_{i=1}^{L} P_i\,(c_j/x)), j = 1, \dots, m. \quad (16)$$

*b) Product combination rule:*

$$y(x) = \arg max(\prod_{i=1}^{L} P_i\,(c_j/x)), j = 1, \dots, m. \quad (17)$$

*c) Max combination rule:*

$$y(x) = \arg max(max(P_i\,(c_j/x))), j = 1, \dots, m. \quad (18)$$

*2) Trained combination rules*

*a) Static weighted average combination rule (SWA):*

This rule has been widely used for combining classifiers [8], [12], [13], [14]. In our case weights are assigned to each individual classifier which represents the importance of each one for achieving the MCS. However, this importance is assumed the same for all test samples. Indeed, the individual classifiers can differ from each other in terms of performance which is measured using a validation dataset. The obtained results can be used for selecting the weights which will be assigned to the individual classifiers

This method relies on the average error rates (AERs) of classifiers which are calculated in the validation step. Denote the AER of the classifier $i$ as $r_i$, $i = 1, \dots, L$ where $L$ defines the number of classifier. Then, the weight $w_i$ assigned to classifier $i$ is calculated as follows:

$$w_i = \frac{1/\sum_{i=1}^{L} \frac{1}{r_i}}{r_i} \quad (19)$$

Such that, $0 \le w_i \le 1$ and $\sum_{i=1}^{L} w_i = 1$.

The class label $y(x)$ of a test sample $x$ is defined mathematically as follows:

$$y(x) = \arg max(\sum_{i=1}^{L} w_i P_i\,(c_j/x)), j = 1, \dots, m. \quad (20)$$

*b) Dynamic weighted average combination rule (DWA):*

The main drawback of the SWA is that all samples are weighted by the same values. This approach is not efficient since each sample has its own importance. Indeed, each classifier allows recognizing well a set of samples which is not well recognized by the others. Therefore, the contribution of each classifier for each test sample must be calculated.

In order to overcome this drawback, we propose to calculate the weight assigned to each classifier for each test sample according to the classifier maximum response kept from the validation step.

More precisely, when the response of the classifier (i.e. the maximum value of all classes) is near to its maximum response, we assign a high contribution is assigned to the classifier comparatively versus other classifiers. In contrast, when the response of the classifier is less than its maximum response, the contribution will be minimized.

Denote $w_i^x$ the weight assigned to the classifier $i$ of a test sample $x$, and denote by $P_i^{max}$ the maximum response of the classifier $i$ calculated in the validation step, the weight assigned to each sample of the $i^{th}$ classifier is defined as:

$$w_i^x = \frac{\max\,(P_i(c_j/x))/P_i^{max}}{\sum_{i=1}^{L} \max\,(P_i(c_j/x))/P_i^{max}}\ ,\ j = 1, \dots, m \quad (21)$$

Such that, $0 \le w_i^x \le 1$ and $\sum_{i=1}^{L} w_i^x = 1$ (22)

The class label $y(x)$ of a test sample $x$ is defined mathematically as follows:

$$y(x) = \arg max(\sum_{i=1}^{L} w_i^x P_i\,(c_j/x)), j = 1, \dots, m. \quad (23)$$

Finally, a test sample $x$ is assigned to the corresponding class when the Combined-OCC provides the maximum prediction value.

As each class is presented by Combined-OCC, adding a new class to the system does not require retraining it again on all classes, but it needs only adding new OCCs, where each OCC is trained on the data of the new class.

## IV. Experimental Results

### A. Dataset Description

For evaluating the proposed MCS, four datasets are selected from ELENA project [20], which represent real applications: iris, phoneme, satimage and texture. In addition, we use Breast cancer [21], Crab gender [22], and handwritten digits [23] datasets. All these datasets are summarized in Table I.

We randomly partition each dataset into three subsets as they are reported in table I. The first subset is used for training the classifiers, the second subset is used for selecting the optimal parameters of each classifier, and the last subset is used for testing the MCS.

### B. Tuning of Parameters for Multiple Classifier System

The MCS is composed of three classifiers which are trained separately on the same feature set using the same training set. However, each classifier has its own parameters which must be tuned.

For the OC-NN classifier, no parameters are required to be tuned unlike to other classifiers (AANN and OC-SVM).

The AANN has two parameters to be tuned which are the number of epochs and the number of nodes. In order to select the optimal parameters for each class the AANN is trained on the training dataset with different parameter values. Then the

optimal parameters are selected when the AANN achieves the best reconstruction of the validation dataset.

The OC-SVM has also two parameters ($\nu$, $\gamma$), which are fixed for each class through the training and validation steps. In the training step, different couples of parameters are generated in order to achieve the best recognition rate of the training dataset. Validation step has been done in order to select the couple of parameters which provides the highest recognition rate.

### C. Results and Descusion

Results for the individual classifiers and MCS with different combination rules conducted on the used datasets are reported in Table II. Firstly, we can note that combining the achieved MCS allows improving the recognition rates than the best single system for all datasets. Secondly, when observing carefully results, we can note that DWA allows improving the recognition rates whatever the selected application. In contrast, the remaining combination rules (Mean, Max, Prod, SWA) depend on the selected dataset. For instance, Mean and Prod rules are suitable for Iris, texture and Breast Cancer datasets, respectively. In contrast, the OC-NN classifier is well suitable for all applications except the Breast Cancer dataset, which requires the use AANN.

It is interesting to note that when using the Satimage dataset, the DWA is the only combinations rule that allows improving the recognition rate. This proves the usefulness of weighting each classifier dynamically according to its response value.

## V. CONCLUSION

This paper aims to propose a new MCS for solving the multi-class classification problem based on OCCs. This MCS is composed of different types of OCC trained in the same feature space using the same trained set. In order to combine classifiers, fixed and trained combination rules are compared for finding the most suitable rule.

Experimental results conducted on several real-world datasets show that the proposed MCS achieves better results than the best individual classifier, when using the dynamic weighted average against the mean, max, product and static weighted average rules.

It is clear that combining all classifiers is not necessary for all datasets. Hence, the extension of this work consists to select for each class the most suitable OCCs which lead to achieve a robust MCS. This work is considered as challenging task which is decomposed into two main problems. The first problem is to find the best diversity measure for selecting the suitable classifiers for each class. The second problem is the need to a calibration function for calibrating outputs originating from different types of classifiers for each class.

TABLE I. DATASETS USED FOR EVALUATING THE PROPOSED MCS

| **Dataset** | **# Classes** | **# Features** | **# Training samples** | **# Validation samples** | **# Test samples** |
|---|---|---|---|---|---|
| Phoneme | 2 | 5 | 200 | 200 | 5004 |
| Iris | 3 | 4 | 45 | 45 | 60 |
| Texture | 11 | 40 | 732 | 731 | 4037 |
| Satimage | 6 | 36 | 600 | 600 | 5235 |
| Breast Cancer | 2 | 9 | 234 | 233 | 232 |
| Crab | 2 | 6 | 66 | 66 | 68 |
| Digits (USPS) | 10 | 14 | 3646 | 3641 | 2007 |

TABLE II. CLASSIFIACTION ACCURACY OF INDVIDUAL CLASSIFIER AND COMBINATION METHODS

| Dataset Accuracy (%) | **Classifier** | | | **Combination rules** | | | | |
|---|---|---|---|---|---|---|---|---|
| | *OC-NN* | *AANN* | *OC-SVM* | *Mean* | *Max* | *Prod* | *SWA* | *DWA* |
| Phoneme | **80.94** | 73.20 | 75.50 | 80.98 | 80.97 | 80.97 | 80.99 | **81.63** |
| Iris | **95.00** | 70.00 | 90.00 | **96.67** | 90.00 | **96.67** | **96.67** | **96.67** |
| Texture | **97.08** | 91.01 | 93.39 | **97.18** | 91.29 | **97.18** | **97.18** | **97.18** |
| Satimage | **86.77** | 70.26 | 78.61 | 85.88 | 70.35 | 86.32 | 84.05 | **87.46** |
| Breast Cancer | 96.26 | **96.41** | 95.65 | **98.44** | 98.08 | **98.44** | **98.44** | **98.44** |
| Crab Gender | **83.82** | 82.35 | 82.35 | 85.29 | 85.29 | 85.29 | 86.76 | **88.24** |
| Digits (USPS) | **81.56** | 79.47 | 71.20 | 82.71 | 79.92 | 82.61 | 82.61 | **83.01** |

## REFERENCES


[1] D.M.J. Tax, and R. P. W. Duin, "Characterizing one-class datasets", In Proceedings of the Sixteenth Annual Symposium of the Pattern Recognition Association of South Africa. pp 21–26, 2005.

[2] B. Krawczyk, and M. Wozniak, " Experiments on distance measures for combining one-class classifiers", in 2012 Federated Conference on Computer Science and Information Systems, pp.89–92. FedCSIS 2012

[3] A. Rabaoui, D. Manuel, Z. Lachiri, and N. Ellouze, "Improve One-Class SVM Classifier for Sounds Classification", Proceedings of IEEE Conference on Advanced Video and Signal Based Surveillance,pp. London, United Kingdom , 2007.

[4] T. Ban, and S. Abe, "Implementing Multi-class Classifiers by One-class Classification Methods", International Joint Conference on Neural Networks Sheraton Vancouver Wall Centre Hotel,Vancouver, BC, Canada,2006

[5] O. Boehm, D. R. Hardoon, and L. M. Manevitz, " Classifying cognitive states of brain activity via one-class neural networks with feature selection by genetic algorithms", Int. J. Mach. Learn. & Cyber 2:125–134, 2011.

[6] K. Goh, and E. Y. Chang, " Using One-Class and Two-Class SVMs for Multiclass Image Annotation". IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING vol.17, pp. ?? –??, 2005

[7] D.M.J. Tax, "One-class classification. PhD Thesis, Delft University of Technology". ISBN: 90-75691-05-x, 2001.

[8] R. P. W. Duin, "The combining classifier: to train or not to train?", In Proc. 16th International Conference on Pattern Recognition, ICPR'02, Canada, 2002, pp. 765–770.

[9] P. Juszczak, and R. P. W. Duin, "Combining One-Class Classifiers to Classify Missing Data", Multiple Classifier Systems., pp. 92-101, 2004

[10] J. Muñoz-Marí, G. Camps-Valls. L. Gómez-Chova. and J. Calpe-Maravilla, " Combination of one class remote sensing image classifiers", IGARSS , pp. 1509-1512, 2007

[11] N. Abbas, Y. Chibani, Z. Belhadi, and M. Hedir, "A DSmT Based Combination Scheme for MultiClass Classiffication", the 16th Inernational Confernece on Information FUSION: ICIF'13, Instanbul, Turkey, July 9-12, 2013.

[12] A. Al-Ani, and M. Deriche, "A New Technique for Combining Multiple Classifiers using The Dempster Shafer Theory of Evidence" , Journal of Artificial Intelligence Research, vol. 17, pp. 333-361, 2002.

[13] A. Verikas, A. Lipnickas, K. Malmqvist, M. Bacauskiene, and A. Gelzinis, "Soft combination of neural classifiers: A comparative study", Pattern Recognition Letters, vol. 20, pp. 429-444, 1999.

[14] W. Wang, A. Brakensiek, and G. Rigoll, "Combination of Multiple Classifiers for Handwritten Word Recognition", IWFHR IEEE Computer Society. New York, 2002

[15] G.W. Cottrell, P.W. Munro, D. Zipser, "Image compression by back propagation: a demonstration of extensional programming", In: Sharkey NE (ed) Advances in cognitive science, vol 2. Abbex, Nrwood, (N J) (in press), 1988

[16] N. Japkowicz, C. Myers. and M. Gluck, "A novelty detection approach to classification", Proc. 14 th International Joint Conference on Artificial Intelligence, pp. Montreal, Canada, 1995.

[17] B. Scholkopf, J.C. Platt, J. Shawe-Taylor, A.J. Smola, and R.C. Williamson, "Estimating the support of a high-dimensional distribution", Neural Computation, Vol. 13. (Also a Microsoft Research technical report, MSR-TR-99-87, 1999.) 2001

[18] R. L. Larkins, "Off-line Signature Verification", The University of Waikato, 2009.

[19] R. O. Duda, P. E. Hart, and D. G. Stork, "Pattern Classification", 2nd ed. John Wiley &Sons, NY, 2001.

[20] ftp.dice.ucl.ac.be in the directory pub/neural-nets/ELENA/databases.

[21] C.J. Merz, P.M. Murphy, "UCI Repository of Machine Learning Databases', Dept. of Information and Computer Science, Univ. of California, Irvine, CA (1998). (2012), http://archive.ics.uci.edu/ml

[22] N. A. Campbell, and R. J. Mahon, "A multivariate study of variation on two species of rock crab of genus Leptograspus", Australian Journal of Zoology, 22, 417–425, 1974.

[23] J. J. Hull, "A Database for Handwritten Text Recognition Research", Pattern Analysis and Machine Intelligence, 16(5):550-554, 1993.