SCORE-BASED ADAPTIVE TRAINING FOR P300 SPELLER BRAIN-COMPUTER INTERFACE

Yuan Zou, Omid Dehzangi, Roozbeh Jafari

Department of Electrical Engineering, University of Texas at Dallas, Richardson, TX 75080, USA

ABSTRACT

The primary aim of a Brain-Computer Interface (BCI) is to provide communication capabilities through brain signals recorded from the scalp for those with brain disorders to be able to interact with the outside world. In order to properly decode the electroencephalographic (EEG) brain signals, the BCI needs to adapt to the subject via calibration to ensure stable performance. One of the major challenges in realization of the EEG signals is the long calibration time required since they show significant variations between recording sessions even for the same subject within the same experimental condition. This paper proposes a score-based adaptive training algorithm that maximally utilizes relevant information from prior recording sessions and significantly shortens the calibration time. Also the proposed method is suitable to develop real-time, wearable, and low-power BCI embedded devices. The BCI developed in this work is based on the P300 word speller application introduced by Farwell and Donchin in 1988. The experimental results show that by employing few letters for calibration, the proposed adaptive training algorithm can achieve 100% classification accuracy.

Index Terms— BCI, EEG signal, P300 speller, Calibration, Adaptive training

1. INTRODUCTION

Encouraged by new understanding of brain function over the past two decades, many researchers have explored BCI technology as a new communication/control channel for those with severe neuromuscular disorders [1, 2]. The goal is to provide basic communication capabilities through the EEG brain signals recorded from implanted electrodes on their scalp so that they can express their wishes, operate word processing programs, or even control neuroprostheses.

During the recent years, many research efforts have been made to improve the performance of BCI technology by introducing advanced and computationally expensive machine learning [3], signal processing [4], and classification techniques [5]. Therefore, high accuracy is achieved at the expense of increased computational complexity. However, in real-time and long-term recording applications, it is highly desirable to consider simpler and more efficient mathematical models to reduce the computational time and power consumption while maintaining adequate classification accuracy. With recent advances in embedded systems and signal processing techniques, there is a major interest in developing real-time, wearable, and low-power BCIs with embedded systems [6].

One major limitation in BCI applications, especially the EEG-based BCIs, is the requirement for long calibration and training sessions (e.g., more than one hour) to collect sufficient training signals for constructing specific features and classifiers. This time-consuming calibration process is necessary for each new recording session and even for the same subjects that are beyond novices within the same experimental environment. Two main reasons the EEG patterns vary strongly from one session to another are 1) subjects have different psychological pre-conditions and 2) electrode coupling conditions vary during different recording sessions (between electrodes and subject's scalp).

There are several previous studies on reducing the calibration time in the BCI technology and different approaches are proposed. In [7] and [8], subject transfer algorithms were proposed to shorten the calibration time for motor imagery BCI applications. Subject transfer is accomplished by constructing a prototype of spatial filters from other subjects and adapting the prototype to the new subject. Also in [9], a similar algorithm based on Support Vector Machine (SVM) was introduced for P300 speller BCI application. However, due to large inter-subject variability, subject transfer algorithms require EEG data from multiple subjects in order to generate a robust prototype spatial filter, but those data may not be available for personal BCI devices. Another strategy is to focus on session transfer instead. In [10] and [11], an algorithm was proposed to skip the calibration process targeted towards long-term BCI users. It is achieved by generalizing a common spatial filter across sessions estimated via training data from prior sessions of the same subject and clustering of prior spatial filters. In [12], a method was presented that allowed reducing the calibration time for both long-term and novel users. Their approach is based on an ensemble of prior classifiers that are transferred to the current session. Both approaches require a large number of historic sessions be available and involve intensive training. Further approaches for reducing calibration processing time, especially for the P300 speller BCI application, are threshold-based adaptive training [13] and semi-supervised learning [14]. In [13], an adaptive training procedure was presented that aimed to estimate the amount of data needed for calibration based on two threshold values. The limitation of this approach is that the adaptive training is highly sensitive to the value of these thresholds. In [14], it was suggested to initially use a BCI with few labeled training samples, and then to incrementally adapt it with unlabeled online data using an iterative semi-supervised learning algorithm. However, the unsupervised learning algorithm needs a large amount of unlabeled data in order to achieve a robust BCI with a satisfactory performance.

In this paper, we investigate the problem of reducing the required calibration time for the P300 speller BCI application [2]. We present a score-based adaptive training method to maximally extract relevant data from prior recording sessions.

To achieve this goal, minimal new recorded data are combined with the complementary data from prior sessions in a two-stage procedure. First, the low quality recorded data caused by artifacts or distraction are rejected based on a similarity score so that the quality of data for further processing is guaranteed. Then, a log-likelihood score-based elimination algorithm is designed to select the most complementary data from prior sessions to train the classifier. Our proposed method significantly shortens the calibration time and at the same time, it does not involve high computational signal processing such as Independent Component Analysis (ICA). Also, the EEG signals in our experiments are recorded from a dry electrode system which is more convenient to wear compared to gel-based electrodes used in previous studies. Therefore, it is suitable for long-term recording applications and is easy to implement in real-time, wearable, and low-power BCI embedded devices. The experimental results show that by using a few letters during the calibration procedure, the proposed adaptive training algorithm achieves 100% accuracy. Even with no calibration data provided, the proposed algorithm can achieve 79% accuracy.

The paper is organized as follows. Section 2 describes the P300 speller experimental setting. Section 3 proposes the motivation and principle of our proposed score-based adaptive training method. Then, section 4 explains the experimental results and conclusions are presented in section 5. Finally, section 6 shows the relation to prior works

2. TASK AND DATA ACQUISITION

The BCI application investigated in this paper is the P300 speller introduced in [2]. It enables users to spell a word from a 6×6 matrix that includes all the alphabet letters as well as other useful symbols (Fig. 1). The rows or columns intensify sequentially in a random order. To spell a word, the subjects are instructed to focus on the letter they wish to communicate by counting the number of times it intensifies. In response, a P300 evoked potential is elicited in the brain which is a positive deflection in the EEG after 300ms [15]. By identifying this P300 pattern, it is possible to infer the attended letter.

Six healthy subjects with no previous experience with the P300 speller participate in the experiment. EEG data are acquired using g.USBamp amplifier (g.tec Medical Engineering GmbH, Austria) and the BCI software platform BCI2000 [16] in a P300 speller scenario. Signals are recorded at 256 Hz sampling rate from 8 g.HASARA active dry electrodes from Fz, Cz, Pz, P3, P4, PO7, PO8, Oz and referenced at right mastoid.

For each subject, two to five sessions of data was recorded. In each session, the subject was instructed to choose between 20-30 letters. For each letter, the intensification lasts for 250 ms followed by a 125 ms blank interval. Twelve intensifications make up one epoch which covers all the rows and columns. 15 epochs are carried out for each letter. Thus for one letter, there are 15×12 =180 intensifications. The task of the P300 speller is to identify the subject's desired letter based on the EEG data collected during the 180 intensifications.

3. THE PROPOSED ALGORITHM

Our proposed approach is a two-stage score-based adaptive training algorithm. In the first stage, the low quality recorded data are rejected based on a similarity score as the preprocessing

Fig1. P300 speller matrix with one row intensified

procedure. In the second stage, an adaptive data selection approach, based on log-likelihood score, is presented in the classifier training procedure.

3.1 Preprocessing: trial rejection

The raw EEG data consist of useful trials carrying discriminative information to detect the target letter. They also include low quality recorded data caused by some artifacts (e.g., eye movement) or due to the subject losing attention, etc. Each trial is composed of all data samples between 0-800 ms from the beginning of target letter intensification. In order to achieve better classification performance, the bad trials need to be rejected in the preprocessing stage to improve the data quality. The most common method used for artifact removal is ICA [17]. However, ICA is not a good preprocessing candidate for real-time applications due to the high computation requirement. In this paper, we present a template matching algorithm based on a similarity score for trial rejection in the preprocessing stage that is straightforward, easy to implement and at the same time leads to high performance. The template is defined based on the average of k target trials in which the P300 pattern is most likely present (k=30 used in this paper),

$$Temp_h = \frac{1}{k}\sum_{i=1}^{k} Trial_i \qquad (1)$$

where $Temp_h$ is the template for the electrode channel h, and $Trial_i$ is the i-th target trial. Then, we calculate the similarity score between each single trial and the template as below,

$$\mu_h^i = 1 - \frac{\sqrt{\sum_{j=1}^{n}(Trial_i(j) - Temp_h(j))^2}}{\sqrt{n}} \qquad (2)$$

where i is the trial index, j = {1, 2, …, n} is the sample index in each trial, and μ_h^i is the similarity score between the i-th trial and the channel h. Assuming that each recorded sample is normalized to the interval [0, 1], the above measure maps the similarities between the trial i-th and the h-th template to a real number in the interval [0,1]. Then, we obtain a ranking according to the score, μ_h^i that rejects the trials corresponding to the λ lowest scores for each channel. Therefore, the trials corresponding to the eye movement, muscle artifacts, or distraction are rejected to guarantee the quality of data for further processing.

3.2. Score-based adaptive data selection

A major limitation in EEG-based BCIs is the requirement for collecting sufficient training data at the beginning of every session (i.e. calibration) for constructing robust features and classifiers. One way to resolve this issue is to employ previously recorded data to adapt with the new session. Even if we obtain data from multiple prior sessions, it might not be useful for this purpose due to the inter-session variability. Therefore, we need to design a mechanism to select the most relevant data from prior sessions and combine with minimal, currently-available data in the new session. In this section, we propose a score-based adaptive data selection algorithm based on the backward elimination procedure [18], described in Algorithm 1. In this algorithm, the function $[i_{max}, f(i_{max})] = max_{i\in N}\{f(i)\}$ returns i_{max}, the value of i for which $f(i)$ is maximum, and the maximum value is denoted by $f(i_{max})$. The function, *train&test*(*D_Training, D_Testing*) is conventionally supposed to return the accuracy obtained by training the classifier on *D_Training* data and testing on *D_Testing* data,

$$Accuracy_\varphi = TP_\varphi - FP_\varphi \quad (3)$$

where TP_φ and FP_φ are the set of true positive and false positive for the letter φ, respectively. However, the accuracy measure may be too coarse to capture the discriminative information among different letters. Therefore in this paper, we define the function *train&test*(*D_Training, D_Testing*) to return $\hat{S}core_\varphi$, as the soft accuracy measure for the test target letter φ within all rows and columns. In this way, the scores are obtained by Stepwise Linear Discriminant Analysis (SLDA) training on *D_Training* data and testing on *D_Testing* data based on the approach in [6]. Sc_p is the original score of the input **x** for the row/column p defined by,

$$Sc_p(\mathbf{x}) = \sum_{i=1}^{ep} S_i(\mathbf{x}) \quad (4)$$

where s_j is the decision score calculated by SLDA classifier for epoch j. ep is the total number of epochs. In this paper, ep=15. Assuming the target letter φ is located at row p, and column q, the likelihood of the letter φ is $Sc_\varphi=Sc_p+Sc_q$. As the distribution of the scores for each target letter may be different due to variability of the brain signals, the scores are less compatible across different letters. Hence, score normalization is a necessary step to provide consistency over the output scores of the classifier. The log-likelihood ratio score normalization for the letter φ is calculated as below,

$$llr_\varphi(\mathbf{x}) = Sc_\varphi(\mathbf{x}) - \log\left(\frac{1}{M-1}\sum_{i=1, i\neq p}^{M}\sum_{j=1, j\neq q}^{M}\exp\left(Sc_i(\mathbf{x}) + Sc_j(\mathbf{x})\right)\right) \quad (5)$$

where M=6 is the number of rows/columns in the matrix, llr_φ is the log-likelihood score for the letter φ, and exp(.) is the exponential function. Therefore, $\hat{S}core_\varphi$, the soft accuracy measure for the test target letter φ is calculated as,

$$\hat{S}core = \sum_{\forall \mathbf{x}_i \in TP}\max_{j=1,\dots,c}\left\{llr_j(\mathbf{x}_i)\right\} - \sum_{\forall \mathbf{x}_i \in FP}\max_{j=1,\dots,c}\left\{llr_j(\mathbf{x}_i)\right\} \quad (6)$$

where c is the number of target letters and TP and FP are the set of true positive and false positive data, respectively. If the index of the winner class for the input $\mathbf{x}_i$ is equal to φ (i.e. $\arg\max_{j=1,\dots,c}\left\langle llr_j(\mathbf{x}_i)\right\rangle = \varphi$),

$$\begin{aligned} \mathbf{x}_i \in TP &\iff True\_label(\mathbf{x}_i) = \varphi \\ \mathbf{x}_i \in FP &\iff True\_label(\mathbf{x}_i) \neq \varphi \end{aligned} \quad (7)$$

$\hat{S}core$ is a soft measure for the overall performance of the system that is expected to capture more discriminative information than the *Accuracy* measure.

Using Algorithm 1, a training set D_{tr} = $\{C_1, C_2, \dots, C_{N_s}\}$ including the total N_s letters is first generated. C_i is the subset data for the i-th letter. Then, it sequentially removes the letters from the current set of letters, in order to maximize $\hat{S}core$ as the overall performance measure on a development set D_{dev} which is a portion of available labeled data that is not contributed in the training. After the adaptive data selection procedure, a new data set S_{tr} is generated to train the SLDA classifier. To evaluate the performance of the proposed algorithm, a test set, D_{test}, including only the data in the new session not contributed in the training or the development sets are employed. The classification accuracy is calculated using LDA classifier trained using S_{tr}.

Algorithm 1: Adaptive data selection algorithm

Input: D_{tr}: original training data; D_{dev}: The development data
Output: S_{tr}: final selected subset of training data
Initialize: baseline score: $\hat{S}core_B$= *train&test* (D_{tr}, D_{dev});
Remaining$_0$ = D_{tr} ; n=1; $N_t = N_s$;
while $n< N_t$ **do**
  for i=1 to N_t
    Remove the data corresponding to the i-th letter from D_{tr}
    $\hat{S}core_i$= *train&test* ($D_{tr}-C_i$, D_{dev}) ;
  end
  Obtain a ranking of $\hat{S}core_i$, find the maximal score value
  (i_{max}, $\hat{S}core_{max}$) = max($\hat{S}core_i$) ;
  if $\hat{S}core_{max} > \hat{S}core_B$ **then**
    Remaining$_n$ = *Remaining*$_{n-1}$ - $C_{i_{max}}$;
    $\hat{S}core_B$= $\hat{S}core_{max}$;
    $N_t = N_t$ -1;
    n=n+1;
  else
    Break;
  end
end
S_{tr} = *Remaining*$_n$;

4. RESULTS

4.1 Trial Rejection Results

To assess the performance of the trial rejection method, we compare the classification accuracy of the signal with and without trial rejection. Three λ values, representing the number of rejected trials, are used in our experiments (i.e. λ = 5, 10, 15) as reported in Table 1. Subject #1, 3, 4, 6 can only achieve 100% accuracy after applying the trial rejection. Also Subject #2 and 5 achieve 100% accuracy with fewer epochs than before. As expected, the results prove that the trial rejection method considerably improves the quality of data, and as a result it leads to better performance. Table 1 also shows that λ=10 leads to the best results in our experiments. In general, λ can be adaptively selected by cross validation on the training data.

Table 1 Classification accuracy (in %) of the signal with/without trial rejection (100(*n*) means achieving 100% accuracy after *n* epochs)

Subject	Without Trial Rejection	With Trial Rejection		
		λ =5	λ =10	λ =15
#1	67	67	100 (11)	67
#2	100 (14)	67	100 (4)	100 (4)
#3	80	100 (9)	100 (11)	80
#4	75	75	100(4)	100(7)
#5	100 (10)	100(10)	100(6)	100(9)
#6	80	80	100(3)	80
Average	83.66	81.5	100(6)	87.8

4.2 Adaptive Training Results

In our experiments, we assume that data for five target letters are available in the current session. The rest of the data in the current session is used as the test set to assess the performance of our proposed algorithm. We compare the classification accuracy achieved via the score-based adaptive training approach, to the system with no adaptive training (no historical data from prior sessions are taken into account). The comparison is depicted in Figure 2 which shows that if no new data is available, the non-adaptive system cannot produce any output, whereas our proposed method already generates stable classification accuracy up to 79%. Using all five available target letters' data, the non-adaptive system achieves the same accuracy that the proposed adaptive training method generated without any data from the current session. With the five target letters' data, the proposed adaptive training method achieves 91% average accuracy within six subjects shown in Figure 1.

Then, we evaluate the effectiveness of the proposed score-based adaptive data selection criterion. To do so, we compare the adaptive training based on $\hat{Score}$ in Eq. (6) with the *Accuracy* criterion in Eq. (3). The results of the comparison between the accuracy-based and the score-based methods are reported in Table 2. The results achieved based on the $\hat{Score}$ value are constantly and considerably better than those based on the *Accuracy* value with different number of available letters.

The last two columns in Table 2 show the averaged classification results with different number of available letters in the new session under the accuracy-based and score-based adaptive training method. The proposed method achieves more than 80% average accuracy on all the subjects with only two available letters' data, and more than 90% average accuracy with five letters.

The results in Table 2 show that the proposed $\hat{Score}$ measure and the adaptive training algorithm are effective in improving the *Accuracy* measure by capturing more discriminative information during the training process. Finally, based on the results of the score-based adaptive training algorithm, five letters are enough for the calibration session to obtain good classification accuracy. Compared to the requirement of at least 30 letters recording in the typical calibration period (30*67.5=2025 seconds), our proposed algorithm dramatically reduces the calibration time (5*67.5s=337.5 seconds).

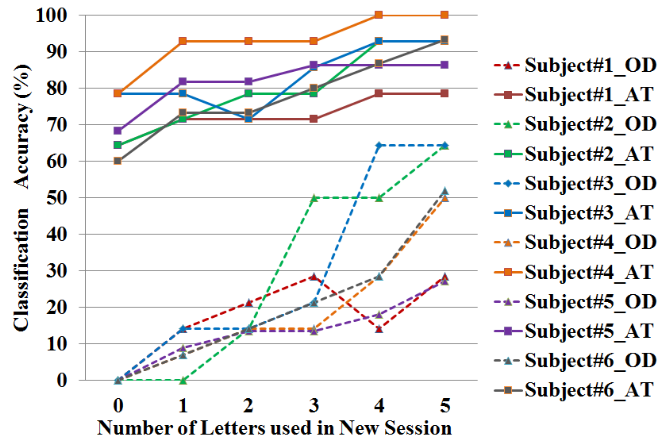

Fig 2 Comparison of the classification accuracy achieved by the non-adaptive method (OD, in dash line) and the score-based adaptive training method (AT, in solid line)

5. CONCLUSION

In this paper, we present a score-based adaptive training algorithm to efficiently shorten the calibration period, as well as achieve better classification performance. In the proposed approach, the data from several prior sessions are combined with few currently available data in the new session. The adaptive training process is accomplished by rejecting the redundant trials based on a similarity score in the preprocessing stage, and then selecting the most relevant data from prior sessions based on the log-likelihood score value. The experimental results validate the good performance of our proposed algorithm. With five available letters' data in the new session, the classification accuracy could achieve 100% (91% on average). Also with no available new data, our proposed method can achieve 79% accuracy (70% on average). Furthermore, in contrast to a non-adaptive method and the accuracy-based adaptive training algorithm, our proposed algorithm significantly outperforms for every subjects under different numbers of available letters' data. Finally, the results of the proposed algorithm show that five letters are enough for calibration to achieve good classification performance.

Table 2 Classification accuracy (in %) achieved by the accuracy-based (Acc) vs. score-based (Score) adaptive training method.

# of letters available in new session	Subjects												Averaged on all 6 subjects	
	#1		#2		#3		#4		#5		#6			
	Score	Acc	Score	Acc	Score	Acc	Score	Acc	Score	Acc	Score	Acc	Score	Acc
0	64.3	57.1	64.3	57.1	78.6	71.4	78.6	64.3	68.2	59.1	60	53.3	**69**	**60.4**
1	71.4	42.9	71.4	71.4	78.6	71.4	92.9	78.6	81.8	72.7	73.3	53.3	**78.2**	**65.1**
2	71.4	57.1	78.6	71.4	78.6	57.1	92.9	78.6	81.8	72.7	80	60	**80.5**	**66.2**
3	71.4	71.4	78.6	78.6	85.7	78.6	92.9	78.6	86.4	77.3	80	73.3	**82.5**	**76.3**
4	78.6	71.4	92.9	78.6	92.9	78.6	***100***	85.7	86.4	72.7	86.7	73.3	**89.6**	**76.7**
5	78.6	71.4	92.9	85.7	92.9	78.6	***100***	85.7	86.4	81.8	93.3	80.0	**90.7**	**80.5**
Average	**72.6**	**61.9**	**79.8**	**73.8**	**84.5**	**72.6**	**92.9**	**78.6**	**81.8**	**72.7**	**78.9**	**65.6**		

6. REFERENCES

[1] J. Wolpaw, N. Birbaumer, D. McFarland, G. Pfurtscheller, T. Vaughan *et al.*, “Brain-computer interfaces for communication and control,” *Clinical neurophysiology*, vol. 113, no. 6, pp. 767–791, 2002.

[2] L. Farwell and E. Donchin, “Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials,” *Electroencephalography and clinical Neurophysiology*, vol. 70, no. 6, pp. 510–523, 1988.

[3] K. Muller, M. Tangermann, G. Dornhege, M. Krauledat, G. Curio, and B. Blankertz, “Machine learning for real-time single-trial eeg-analysis: From brain-computer interfacing to mental state monitoring,” *Journal of neuroscience methods*, vol. 167, no. 1, pp. 82–90, 2008.

[4] A. Bashashati, M. Fatourechi, R. Ward, and G. Birch, “A survey of signal processing algorithms in brain–computer interfaces based on electrical brain signals,” *Journal of Neural engineering*, vol. 4, no. 2, p. R32, 2007.

[5] F. Lotte, M. Congedo, A. Lécuyer, F. Lamarche, B. Arnaldi *et al.*, “A review of classification algorithms for eeg-based brain–computer interfaces,” *Journal of neural engineering*, vol. 4, 2007.

[6] U. Hoffmann, JM. Vesin, T. Ebrahimi, K. Diserens, “An efficient P300-based brain-computer interface for disabled subjects”, *J Neurosci Methods*. 2008;167:115–25.

[7] W. Tu and S. Sun, “A subject transfer framework for eeg classification,” *Neurocomputing*, 2011.

[8] F. Lotte and C. Guan, “Learning from other subjects helps reducing brain-computer interface calibration time,” in *Acoustics Speech and Signal Processing* (*ICASSP*)*, 2010 IEEE International Conference on*. IEEE, 2010, pp. 614–617.

[9] M. Kaper and H. Ritter, “Generalizing to new subjects in brain-computer interfacing,” in *Engineering in Medicine and Biology Society, 2004. IEMBS’04. 26th Annual International Conference of the IEEE*, vol. 2. IEEE, 2004, pp. 4363–4366.

[10]M. Krauledat, M. Tangermann, B. Blankertz, and K. Müller, “Towards zero training for brain-computer interfacing,” *PLoS One*, vol. 3, no. 8, p. e2967, 2008.

[11] M. Krauledat, M. Schröder, B. Blankertz, and K. Müller, “Reducing calibration time for brain-computer interfaces: A clustering approach,” *Advances in Neural Information Processing Systems*, vol. 19, pp. 753–760, 2007.

[12] S. Fazli, F. Popescu, M. Danóczy, B. Blankertz, K. Müller, and C. Grozea, “Subject-independent mental state classification in single trials,” *Neural networks*, vol. 22, no. 9, pp. 1305–1312, 2009.

[13] B. Rivet, H. Cecotti, M. Perrin, E. Maby, and J. Mattout, “Adaptive training session for a p300 speller brain–computer interface,” *Journal of Physiology-Paris*, vol. 105, no. 1, pp. 123–129, 2011.

[14] Y. Li, C. Guan, H. Li, and Z. Chin, “A self-training semi-supervised svm algorithm and its application in an eeg-based brain computer interface speller system,” *Pattern Recognition Letters*, vol. 29, no. 9, pp. 1285–1294, 2008.

[15] S. Sutton, P. Tueting, and J. Zubin, “Information delivery and the sensory evoked potential.” *Science*, 1967.

[16] G. Schalk, D. McFarland, T. Hinterberger, N. Birbaumer, and J. Wolpaw, “Bci2000: a general-purpose brain-computer interface (bci) system,” *Biomedical Engineering, IEEE Transactions on*, vol. 51, no. 6, pp. 1034–1043, 2004.

[17] Y. Zou, J. Hart, and R. Jafari, “Automatic eeg artifact removal based on ica and hierarchical clustering,” in *Acoustics, Speech and Signal Processing* (*ICASSP*)*, 2012 IEEE International Conference on*. IEEE, 2012, pp. 649–652.

[18] P. Devijver and J. Kittler, *Pattern recognition: A statistical approach*. Prentice/Hall International, 1982.