Advances in Classification of EEG Signals via Evolving Fuzzy Classifiers and Dependant Multiple HMMs

C. Xydeas, P. Angelov, C. Shih-Yang, M. Reoullas

Digital Signal Processing Group

Department of Communication Systems

Infolab21, Lancaster University

Lancaster, LA1 4WA, UK

Phone: +44-1524-510310

Fax: +44-1524-592713

Abstract: *Two novel approaches to the problem of brain signals (EEG) classification are introduced in the paper. The first method is based on a modular probabilistic network architecture that employs Multiple Dependant Hidden Markov Models (DM-HMM-D) on the input features (channels). The second method, eClass, is based on an on-line evolvable fuzzy rule base of EEG signal prototypes that represent each class and take into consideration the spatial proximity between input signals. Both approaches use supervised learning but differ in their mode of operation. eClass is designed recursively, on-line, and has an evolvable structure, while DM-HMM-D is trained off-line, in a block-based mode, and has a fixed architecture. Both methods have been extensively tested on real EEG data that is recorded during several experimental sessions involving a single female subject who is exposed to mild pain induced by a laser beam. Experimental results illustrate the viability of the proposed approaches and their potential in solving similar classification problems. © Elsevier 2006*

1. INTRODUCTION

1.1 EEG in the context of brain-computer interaction

The human brain is the most complex information processing system known to science. It is this complexity that has attracted the interest of many scientists over the past years in studying the physiological activity of the brain. Human brain activity can be recorded in the form of electroencephalogram (EEG) signals. These provide an important source of information which can be useful in the study of underlying brain processes as well as in a variety of medical applications i.e. the use of EEG signals for psychiatric/physiological diagnosis as well as for evaluation of sensory experiences [1]. A particularly interesting aspect of EEG signal analysis is that related to pain experiences. From a medical point of view, the pain sensation is important either because of its mere, unpleasant presence or because it inhibits physiological functions or because it can be related to certain pathological conditions. These are a major source of disability, poor health status and mortality. The ability to analyze EEG signals and to interpret accurately different aspects of the pain sensation can be a major advance in pain research and clinical pain management. Previous work has shown that analysis of pain can be possible with the help of powerful techniques based on coherence analysis and topographic mapping. Thus EEG coherence analysis provides a measure of functional correlations between EEG signals [1–5] whereas functional imaging techniques have identified the matrix of brain structures that are responsible for the elaboration of pain experience. The precise division of function within this matrix is unclear.

The general thrust of the work presented in this paper is to develop practically feasible "*intelligent*" computer based systems for processing EEG brain activity information. Furthermore, the specific EEG signal analysis issue addressed here is that of pain detection, a capability that can find extensive use in the medical area, for example in real-time pain monitoring of patients under anesthesia, or in the safety/defense field, for example in *real–time* pain detection monitoring of human operators performing critical tasks. The following two subsections provide an introduction to

two and substantially different EEG signal classification methodologies (i.e. DM-HHM-D and *e-Class*) which have been developed and employed within the context of *real-time* pain detection. These classification techniques employ supervised learning in their classification system design, with the DM-HHM-D technique relying on the *a priori* (**off-line**) provision of representative training data and has a **fixed** system architecture whereas the *e-Class* method is capable of formulating the classification system architecture and associated fuzzy rules/parameters **on-line** and "**from scratch**" while processing and classifying given input EEG signals. Furthermore, given the fuzzy rules designed from previous data, *e-Class* is capable of updating and generalizing its set of rules by further recursive on-line training using new data.

1.2 HMM and DM-HMM-D

The theory of Hidden Markov Models (HMMs) [7]-[9] is a rigorous probabilistic classification framework that has been successfully applied to several applications domains [10]-[12]. Furthermore, their natural capability of dealing with time varying patterns of arbitrary lengths is attractive due to the expected variability in the time lengths of dynamic signal patterns. In most *real-time* classification applications (i.e. pain/no pain EEG signals classification) several different signals or features are observed (i.e. 64 EEG channels from the brain cap) in order to increase recognition/classification performance.

When dealing with multiple features having discrete observation densities, two HMM system design approaches can be used [8,10]: The first, HMM-VQ, employs Vector Quantization (VQ) to account for any dependencies that may exist between the input features whereas the second approach, IM-HMM-D, is based on the "separation" of input features and the assumption of no significant dependency between input features. The main drawback of the VQ based approach is that classification system performance can be reduced significantly if an inappropriate VQ method (in terms of quantization accuracy) is chosen [11]. Furthermore, due to system complexity limitations, codebooks are of relatively small size and are constructed by considering as few input features as

possible, which in turn leads to a feature selection search process. In addition when a new feature is added or an old feature is removed, codebooks must be re-designed and HMM networks retrained/redefined.

The second method [10][13] is based on a "multi-HMM-D" system formulation (hereafter referred to as IM-HMM-D) that employs in parallel separate HMM networks per input feature and combines their outputs in order to formulate an overall classification result. Furthermore a novel multiple HMM-D system architecture is presented in this paper that computes dynamically with time "weights" associated to the observed values of different features and employs them in the formulation of the overall classification result, according to the varying-with-time importance of the input features. This proposed classification structure, named as DM-HMM-D, aims to fully and efficiently exploit any inter-dependencies that may exist between input features.

1.3 Evolving Classifier (e-Class)

An important aspect in the analysis of EEG signals is the relevance of the classification system results and the system's generalization capability. Very often [6] EEG data are extracted from a small number of subjects or even a single subject, as in the case of the present study. Experimental configuration may vary, including the mental and physical condition of the subject (expectation, anxiety etc.). As a result, classifiers which are trained in a "**batch**" mode with a **fixed** set of training data may become irrelevant or imprecise when applied to scenarios characterized by new operating conditions. One possible solution to this problem is to design classifiers that can be *recursively* updated, or *self-organizing classifiers*.

Recently proposed schemes concerning evolving un-supervised clustering [26] and evolving Self Organizing Maps (SOM) [15] can be extended to the case of supervised learning (labeled outputs). These self-organizing classifiers are also called *evolving*, because they develop their structure of prototypical samples starting "from scratch", using the input EEG signals and their accumulated proximity measure. In this paper, a novel fuzzy classifier is build. The proposed design approach is

generic and can be applied to other classification problems, such as *on-line* classification of difficult targets [25].

Paper Organization

The remaining of this paper is organized as follows. Section 2 gives an insight on the procedure used for EEG data collection and includes some technical information on data acquisition and data preprocessing techniques. Section 3 discusses HMM systems in general and the novel DM-HMM-D scheme in particular, emphasizing its main characteristics and resulting advantages as compared to conventional HMM-D. Section 4 introduces a novel on-line recursive approach to the classification of EEG signals, called *eClass*, where the structure of the classifier evolves to take into account the changing characteristics of the processed EEG signals. This classification method is generic and is ideally suited for use in *on-line* type of applications where the architecture of the classification system is not predetermined [25]. Section 5 gives an account and analysis of the experimental results obtained from both methods operating on the same real data. Finally, section 6 concludes this paper with a summary of the comparative investigations presented in previous sections and identifies future work aimed at achieving further improvements in EEG pain classification results.

2. EXPERIMENTAL CONFIGURATION

Repeated heat stimuli in the form of laser pulses were delivered in a controlled manner by laser cannon (CO_2 laser) to the right forearm of the subject. The duration of each pulse is set to 150ms, and each such stimulus is repeated at regular intervals of 10s (epochs). Each EEG continuous recording included 61 stimuli. Note that EEG (channel) signal responses to the first stimulus were routinely discarded, as they were considerably higher in amplitude, due to an element of "surprise" that is often exhibited by the subject and associated artefacts in the EEGs. Thus, 60 stimuli were taken into account for each recording and nine continuous EEG data files were produced from a healthy female

6

subject. These data files were recorded on three different days (i.e. 08.06.00, 15.06.00 and 02.08.00) with three EEG recording sessions taken on each of the above three days.

More specifically, this EEG data collection experimental procedure can be described as follows:

1) The subject was seated comfortably in a chair, placed his/her arm on a table and the laser cannon was positioned at a fixed distance from the subject's hand.

2) Electrodes were placed around the scalp, the eyes (to test whether the subject blinks) and on each ear (which are used as "ground" for EEG), see Figure 2.1.

Figure 2.1 here

3) A computer is placed in front of the subject, showing a table with the pain intensity scale. The subject was asked to rate the pain intensity of each stimulus using a scale from 0 to 10, where 0 represents no sensation and 10 represents unbearable pain. In these experiments the rating of 4 ("just painful") was used as a pain threshold (i.e. 1–3: non–painful and 4–10: painful). Subjective pain ratings were also recorded so that they could be used as "target" classification data during the training and testing procedures of the proposed classification algorithms.

4) EEG readings taken from the electrodes were then stored automatically in a file using a control program that also operated the laser cannon.

Thus recordings were made using a 64–electrode cap (see Figure 2.1) with 62 head electrodes while two face electrodes (vEOG and hEOG) were used to monitor artifacts from eye movement. It must be mentioned here that EEG artifacts associated with signal activity in electrodes vEOG and hEOG were appropriately removed *prior* to the pain classification experiments.

Figure 2.2 here

EEG signals were band–pass filtered at 0.15–30Hz and sampled at a frequency of 500Hz, with a gain of 500 (150 for the EOG channels). Following the above described acquisition process of Pain/No Pain EEG signal files, signal segments of 1s duration were extracted from the original EEG files,

corresponding to (a) "Pain" signal states, represented by 1s intervals starting from the time of the pain stimulus, and (b) "No Pain" signal states represented by 1s intervals centred at 1s before the application of each pain stimulus.

3. DM-HMM-D

The general structure of a one-feature discrete observation HMM network is shown diagrammatically in Figure 3.1.

Figure 3.1 here

There are N hidden states (nodes) $\{S_1, S_2, ..., S_N\}$ in the model and M possible observations can be generated by the model. At every time step one of the states, say S_j, is entered based on the state transition probability $\{a_{ij}\}$ that depends on the previous state S_i. After each transition is made, an observation, say the m-th observation o_m, is produced from S_j with corresponding observation probabilities $\{b_j(o_m)\}$, note that the initial state probabilities are defined as $\{\pi_i\}$. A compact notation $\lambda = \{\{a_{ij}\}, \{b_j(o_m)\}, \{\pi_i\}\}$ is set to indicate the complete model parameters. Therefore, the probability of an given observation sequence $O = \{o_1, o_2, ..., o_T\}$ in the period of time T can be calculated by tracing the paths $Q = \{q_1, q_2, ..., q_T\}$ (Viterbi paths) which offer the maximum likelihood probability $P(O|\lambda)$. Equation 3.1 shows the result of multiplying all the probabilities that the Viterbi path passes through. Note that $f(\pi, a)$ is a function of initial probabilities and state transition probabilities.

$$
\begin{aligned}
P(O \mid \lambda) &= \pi_{q_1} b_{q_1}(o_1) \quad a_{q_1 q_2} b(o_2) \cdots a_{q_{T-1} q_T} b_{q_T}(o_T) = \\
&= \pi_{q_1} b_{q_1}(o_1) \prod_{(k)=2}^{T} a_{q_{(k)-1} q_{(k)}} b_{q_{(k)}}(o_{(k)}) = f(\pi, a) \prod_{(k)=1}^{T} b_{q_{(k)}}(o_{(k)})
\end{aligned}
\tag{3.1}
$$

A novel multiple HMMs system architecture (named DM-HMM-D, Figure 3.2) is introduced that computes the weights attached to different sequences of observations prior to the operation of HMM models. Figure 3.2 (a) shows the conventional IM-HMM-D model framework and the final probability likelihood in the IM-HMM-D is computed as shown in Equation 3.2 which assumes that the i-th observation sequence $O^{(i)}$ and the remaining observation sequences $\{O^{(1)}, O^{(2)}, ..., O^{(c)}, ..., O^{(C)}\}$,

$c \neq i$ are independent.

Figure 3.2 here

In this case C features $\{o_{(k)}^{(1)}, o_{(k)}^{(2)}, ... o_{(k)}^{(C)}\}$ are available, at a given time k, the system employs C HMM parameter sets $\{\lambda_1, \lambda_2, ..., \lambda_C\}$ and the total likelihood probability $P(O|\lambda)$ is given as:

$$P(O|\lambda) = \prod_{i=1}^{C} P(O^{(i)}|\lambda_i) \tag{3.2}$$

This conditional independence is stated as

$$P(O^{(i)}|\lambda_i, Y) = P(O^{(i)}|\lambda_i) \tag{3.3(a)}$$

where $O^{(i)} = \{o_1^{(i)}, ..., o_T^{(i)}\}$ is the observation sequence of the i-th feature and $Y = \{O^{(1)}, O^{(2)}, ..., O^{(c)}, ..., O^{(C)}\}$, $c \neq i$ are the observation sequences of the remaining features.

In Figure 3.2 (b), different sequences of observations are considered to be "linked" in a vertical manner by assuming that a weighting function is introduced to each model. The output probability of the i-th model is rewritten as $P(O^{(i)}|\hat{\lambda}_i) \ w_i(O)$, where $\hat{\lambda}_i$ is the new HMM parameter set for the i-th feature. Equation (3.2) can be rewritten as:

$$P(O|\hat{\lambda})' = \prod_{i=1}^{C} (P(O^{(i)}|\hat{\lambda}_i) \ w_i(O)) \tag{3.3(b)}$$

where $w_i(O)$ is designed to be the conditional probability of $O^{(i)}$ given Y, i.e. the probability of the observation sequence of the i-th feature given the observation sequences of the remaining features.

$$w_i(O) = p(O^{(i)}|Y) \tag{3.3(c)}$$

The system shown in Figure 3.2(b) now takes the form shown in Figure 3.2(c) that can be also depicted as in Figure 3.2 (d). This new Multi HMM model structure is named as DM-HMM-D, to distinguish it from the conventional IM-HMM-D scheme. Since the weight function $w_i(O)$ and the conventional HMM structure are now effectively combined, the HMM training and testing procedures must be adjusted appropriately.

Considering Equations (3.3(b)), (3.3(c)) and (3.2), the conditional probability $p(O^{(i)}|\hat{\lambda}_i)'$ can be rewritten as:

$$P(O^{(i)} \mid \hat{\lambda}_i)' = P(O^{(i)} \mid \hat{\lambda}_i) \quad P(O^{(i)} \mid Y)$$

$$= f(\pi^{(i)}, a^{(i)}) \quad \prod_{(k)=1}^{T} b^{(i)}_{q_{(k)}}(o^{(i)}_{(k)}) \quad \prod_{(k)=1}^{T} p(o^{(i)}_{(k)} \mid y_{(k)}) \qquad (3.4)$$

$$= f(\pi^{(i)}, a^{(i)}) \quad \prod_{(k)=1}^{T} b^{(i)}_{q_{(k)}}(o^{(i)}_{(k)}) \, p(o^{(i)}_{(k)} \mid y_{(k)})$$

where the product terms represent the transitional probabilities of the new model, i.e.

$$b^{(i)}_j(o^{(i)}_{(k)})' = b^{(i)}_j(o^{(i)}_{(k)}) \quad p(o^{(i)}_{(k)} \mid y_{(k)}) \qquad (3.5)$$

It can be seen that the conditional independent probability $P(O^{(i)}|Y)$ will only affect observation transition probability $\{b_j^{(i)}(o_{(k)}^{(i)})\}$. Therefore DM-HMM-D can be implemented by replacing $\{b_j^{(i)}(o_{(k)}^{(i)})\}$ with the probability $\{b_j^{(i)}(o_{(k)}^{(i)})'\}$ at each time step (k).

$\{b_j^{(i)}(o_{(k)}^{(i)})'\}$ is calculated using Equation (3.5) with the help of a pre-defined (during the training procedure) "dependency" codebook that contains $p(o_{(k)}^{(i)}|y_{(k)})$ estimates. In particular, $p(o_{(k)}^{(i)}|y_{(k)})$ estimates are obtained using:

$$p(o_{(k)}^{(i)}|y_{(k)}) = p(o_{(k)}^{(i)} \mid (\{o_{(k)}^{(1)}, o_{(k)}^{(2)}, \cdots, o_{(k)}^{(c)}, \cdots, o_{(k)}^{(C)}\})) = \frac{p(o_{(k)}^{(i)}, y_{(k)})}{p(y_{(k)})}$$

$$= \frac{\displaystyle\sum_{k=1}^{K}\sum_{(k)'=1}^{T_k} h(O_{k,(k)'}, O_{(k)}) / \sum_{k=1}^{K} T_k}{\displaystyle\sum_{k=1}^{K}\sum_{(k)'=1}^{T_k} h(U_{k,(k)'}(i), V_{(k)}(i)) / \sum_{k=1}^{K} T_k} = \frac{\displaystyle\sum_{k=1}^{K}\sum_{(k)'=1}^{T_k} h(O_{k,(k)'}, O_{(k)})}{\displaystyle\sum_{k=1}^{K}\sum_{(k)'=1}^{T_k} h(U_{k,(k)'}(i), V_{(k)}(i))}, z \neq i \qquad (3.6)$$

where $U_{k,(k)'}(i) = \{o_{k,(k)}^{(1)}, o_{k,(k)}^{(2)}, ..., o_{k,(k)}^{(c)}, ..., o_{k,(k)}^{(C)}\}$ and $V_{(k)}(i) = \{o_{(k)}^{(1)}, o_{(k)}^{(2)}, ..., o_{(k)}^{(c)}, ..., o_{(k)}^{(C)}\}$ with $c \neq i$ are calculated as the expected number of times in observing $V_{(k)}(i)$ for all $U_{k,(k)'}(i)$ in K training data sets, $k=\{1,2,...,K\}$. The counting function $h(a,b)$ is equal to one if and only if $\{a=b\}$, otherwise its value is zero.

The model evaluation and estimation procedures used in DM-HMM-D are effectively those developed for conventional HMM-D structures with the simple replacement of $\{b_j^{(i)}(o_{(k)}^{(i)})\}$ with $\{b_j^{(i)}(o_{(k)}^{(i)})p(o_{(k)}^{(i)}|y_{(k)})\}$.

3.1 Model Evaluation

This involves an efficient forward-backward procedure, which takes only $O(TN^2)$ operations with

sequence length T and state number N.

The forward algorithm incorporates the following steps:

- Initialization:

$$\alpha_1(i) = \pi_i b_i(o_1) p(o_1 \mid y_1), \qquad 1 \le i \le N \tag{3.7}$$

- Induction:

$$\alpha_{(k)+1}(j) = [\sum_{i=1}^{N} \alpha_{(k)}(i) \quad a_{ij}] \quad b_j(o_{(k)+1}) p(o_{(k)+1} \mid y_{(k)+1}), \qquad 1 \le (k) \le T, 1 \le j \le N \tag{3.8}$$

- Termination:

$$P(O \mid Y, \lambda) = \sum_{i=1}^{N} \alpha_T(i) \tag{3.9}$$

where the forward variable $\alpha_{(k)}(i)$ is defined as $\alpha_{(k)}(i) = P(o_1 \, o_2 \dots \, o_{(k)} \, , \, q_{(k)} = S_i / , \lambda) p(o^{(i)} / y_{(k)})$. This formulation of the forward probability calculation is based on a lattice structure and is efficient since the calculation of the forward variable $\alpha_{(k)}(i)$ involves only N previous values of $\alpha_{(k)-1}(i)$ [4].

The backward part of the process is similar to the Forward procedure with,

- Initialization:

$$\beta_T(i) = 1, \qquad 1 \le i \le N \tag{3.10}$$

- Induction:

$$\beta_{(k)}(i) = [\sum_{j=1}^{N} a_{ij} \quad b_j(o_{(k)+1}) p(o_{(k)+1} \mid y_{(k)+1})] \beta_{(k)+1}(j), \qquad 1 \le i \le N; (k) = T \quad 1, T \quad 2, \dots, 1 \tag{3.11}$$

- Termination:

$$P(O \mid Y, \lambda) = \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_{(k)}(i) a_{ij} b_j(o_{(k)+1}) p(o_{(k)+1} \mid y_{(k)+1}) \beta_{(k)+1}(j) \tag{3.12}$$

where the backward variable $\beta_{(k)}(i)$ is defined as $\beta_{(k)}(i) = P(o_{(k)+1} \dots \, o_{T+2} \dots \, o_T \mid q_{(k)} = S_i, \lambda) \, p(o^{(i)} / y_{(k)})$.

3.2 **Model Estimation**

The probability of $P(O \mid Y, \lambda)$ is maximized via an iterative estimation process. Thus HMM parameter sets (models) are calculated using the EM algorithm [7] with E and M steps operating over an initial

model, noted as $\lambda = \{A, B, \pi\}$ and a re-estimated model $\overline{\lambda} = \{\overline{A}, \overline{B}, \overline{\pi}\}$. The expectation step E involves the calculation of Baum's auxiliary function $Q(\lambda, \overline{\lambda}))$ whereas the M (modification) step is the maximization over $\overline{\lambda}$. Re-estimation of a parameter set using K training data streams involves:

$$\overline{\pi}_i = \sum_{k=1}^{K} \frac{\alpha_1^{(k)}(i)\beta_1^{(k)}(i)}{P(O^{(k)}|Y^{(k)}, \lambda)} \tag{3.13}$$

$$\overline{a}_{ij} = \frac{\sum_{k=1}^{K}\frac{1}{P_k}\sum_{t=1}^{T_k-1}\alpha_t^{(k)}(i)a_{ij}b_j^{(k)}(o_{t+1}^{(k)})p(o_{t+1}^{(k)}|y_{t+1}^{(k)})\beta_{t+1}^{(k)}(j)}{\sum_{k=1}^{K}\frac{1}{P_k}\sum_{t=1}^{T_k-1}\alpha_t^{(k)}(i)\beta_t^{(k)}(i)} \tag{3.14}$$

$$\overline{b}_j(\ell) = \frac{\sum_{k=1}^{K}\frac{1}{P_k}\sum_{\substack{t=1 \\ s.t.o_t^{(k)}=\ell}}^{T_k-1}\alpha_t^{(k)}(i)\beta_t^{(k)}(i)}{\sum_{k=1}^{K}\frac{1}{P_k}\sum_{t=1}^{T_k-1}\alpha_t^{(k)}(i)\beta_t^{(k)}(i)} \tag{3.15}$$

In DM-HMM-D parameter sets $\hat{\lambda}_i = \{\pi^{(i)}, a^{(i)}, b^{(i)}\}$, $i=1,...,C$, are determined during training while using $\{b_j^{(i)}(o_t^{(i)})p(o_t^{(i)}|y_t)\}$ instead of $\{b_j^{(i)}(o_t^{(i)})\}$ [14,15]. Following training, and during testing (i.e. when using the derived system to perform classification of input signals) the required probability $P(O^{(i)}|Y)$ for each input testing data stream is obtained from a pre-designed dependency codebook.

4. E-CLASS

As mentioned earlier, the relevance of the results produced by an *off-line* classification technique will be limited to the degree of representativeness of the training data. The applicability of such an *off-line* trained classifier to new data sets is limited and therefore, the design of incrementally *evolvable* classifiers is an attractive alternative. *Evolvable* fuzzy-rule-bases have been recently developed and successfully applied to clustering [26], time-series prediction [14,15], and neuro-fuzzy systems [24]. In this paper this concept [19] is extended to the *on-line* signal classification. The resulting novel approach called *eClass* is based on an *evolvable* rule base, which is composed of fuzzy rules of the following form:

Rule^j: ***IF** (EEG_1 is EEG_1^{j*}) **AND** ...**AND** (EEG_n is EEG_n^{j*}) **THEN** (**Class** is Pain/No Pain)*   (4.1)

where EEG_i denotes the electroencephalogram signal produced by the ith channel, $i=1,2,...,n$; in this particular application $n=2$, i.e. only the two most informative channels are employed (see diagram representing the importance of all 64 channels in Figure 5.2). These channels have been identified according to the $w_i(O)$ weight values of input features (indicating the relative importance of corresponding channels) as estimated by the system described in section 3;

EEG_i^{j*} denotes the jth prototypical *EEG* signal of the ith input (channel), $i=1,2,...n$; $j=1,2,...,$ R_k rules, $k=1,2,...,m$

R is the number of fuzzy rules;

m is the number of classes (in this particular application $m=2$, namely: "Pain" and "No Pain")

Class is the output of the classifier (in this particular application it is a binary variable with values Pain/No Pain.

This overall rule base comprises of *m* sets of fuzzy rules – one per class, see Figure 4.1 where two sets of prototypical EEG signals are depicted – one for the class "Pain" (Figure 4.1 (a)) and another for the class "No Pain" (Figure 4.2 (b)).

Figure 4.1 (a) and (b) here.

The fuzzy rule base is designed in *on-line* mode via supervised learning starting "*from scratch*". It selects the first measured *EEG* signal as a prototype. Then, starting from the next measured *EEG* signal, an accumulated proximity measure (called *potential*, [14, 19]) is calculated recursively and the rule-base is incrementally updated. The potential, *P* is inversely proportional to the sum of Euclidean distances between a particular *EEG* signal and *all* other *EEG* signals. The value of the potential will be higher for these *EEG* signals that are similar to a large number of other *EEG* signals. It should be noted that in contrast to evolving clustering [26], evolving time series prediction [14], and evolving

modeling [15,22] techniques, in *eClass* the *potential* is calculated with respect to the inputs (discrete *EEG* signals) only. Class labels (classifier outputs) are not included in the calculation of *P*.

The overall classification is performed based on the so called '*winners take all*' principle [27], which corresponds to the MAX t-norm used to produce a defuzzification (note that the same is also used in Mamdani type fuzzy models) [18].

$$y = y_k; k = \arg \max_{j=1}^{R} (\lambda_j) \tag{4.2a}$$

$$y = \sum_{j=1}^{R} \lambda_j y_j \tag{4.2b}$$

where y_k represents the kth class, $k=1,2,...m$

$\lambda_j = \dfrac{\tau_j}{\sum\limits_{i=1}^{R} \tau_i}$ is the normalized firing level of the jth rule, $j=1,2,...R$.

$\tau_j = e^{-\alpha \sum\limits_{i=1}^{n} \sum\limits_{l=1}^{L} \left(x_i^l - x_j^{l*} \right)^2}$ is the firing rule of the jth fuzzy rule; $j=1,2,...,R;$

x_j^l is the jth sampled *EEG* signal;

x_j^* is the jth prototypical *EEG* signal based on which the jth fuzzy rule antecedent is formed;

$\alpha=4/r^2$ is a positive constant which defines the spread of the membership function of the fuzzy sets which are of Gaussian type;

r is the radius that defines the zone of influence of the fuzzy rules;

L denotes the length of the discrete *EEG* signal.

In Figure 4.2 a snap-shot, at given time instant k, of a 2-feature space is presented and two classes (Pain and No Pain) are shown with different types of lines. It is important to mention that the classification of *EEG* signals is performed based on similarity using the whole length of the *EEG* signal ($l=1,2,...,L$), whereas this Figure represents information related to a certain time instant.

Figure 4.2 here

The *input* data space in *eClass* is clustered on a **per class** *basis*. For each class the algorithm forms a partial rule-base, which consists of R_k rules ($k=1,2,...,m$). These class-related partial rule-bases are then combined to form the overall rule-base R of the *eClass* process. In this way, the total number of fuzzy rules that form the evolvable classifier R is equal to the sum of fuzzy rules that form the partial rule-sets associated with each class, see Figure 4.3:

$$R = R_1 + R_2 + ... + R_m \tag{4.3}$$

Figure 4.3 here

It should be mentioned that the system learning and the testing procedures are performed in *on-line* mode. *EEG* signals are first presented to *eClass* for classification and then (given the ground truth/label) the same *EEG* signals are used to update or upgrade the partial rule-base, R_k of the (k^{th}) class**.**

In *eClass* training, *EEG* signals are collected continuously. Some signals reinforce and confirm the information contained in the classifier. Other input signals provide new information, which may be important enough *to form a new fuzzy rule or to modify* an existing one. The value of the information they contain can be measured by their *potential*, *P*. Two main potentials are calculated *recursively*:

   a)   the *potential* of a *EEG* signal that is to be used as a new prototype;

   b)   the *potential* of the existing prototype *EEG* signals.

Thus the *potential* of a new *EEG* signal to be a prototype of class *j* can be calculated recursively by [26]:

$$P_k[x(k)] = \frac{k-1}{(k-1)(1+b(k)) - 2h(k) + g(k)} \ ; k=2,3,... \tag{4.4}$$

where $P_k(x(k))$ denotes the potential of the *EEG* signal *x(k)* calculated at the moment *k*;

$$b(k) = \sum_{j=1}^{n+1} \sum_{l=1}^{L} (x_j^l(k))^2 \ ; \qquad\qquad g(k) = \sum_{i=1}^{k-1} \sum_{j=1}^{n+1} \sum_{l=1}^{L} \left(x_j^l(i)\right)^2$$

$$h(k) = \sum_{j=1}^{n+1}\sum_{l=1}^{L} x_j^l(k)p_j^l(k); \qquad p_j^l(k) = \sum_{i=1}^{k-1}\sum_{l=1}^{L} x_j^l(i)$$

Parameters $b(k)$ and $h(k)$ are calculated from the current *EEG* signal $x(k)$, while $p_j^l(k)$ and $g(k)$ are recursively updated by

$$g(k) = g(k-1) + \sum_{j=1}^{n+1}\sum_{l=1}^{L}(x_j^l(k-1))^2 \quad ; \quad p_j^l(k) = p_j^l(k-1) + x_j^l(k-1)$$

A new input *EEG* signal is influencing the potentials of the established prototypes (x_j^*, $j=1,2,...,R$), because by definition *potentials* depend on the distance to *all* of the input signals, including the new one. The *potential* of a prototype (x_j^*) at the moment k can be calculated as [26]:

$$P\left(x_j^*(k)\right) = \frac{1}{1 + \dfrac{1}{k-1}\sum_{p=1}^{k-1}\sum_{i=1}^{n+1}\sum_{l=1}^{L}(d_l^i(j,p))^2} \; ; \; k=2,3,... \tag{4.5}$$

where $P\left(x_j^*(k)\right)$ is the potential of the at the moment k of the cluster, which is a prototype of the jth rule;

$d_l^i(j,p)$ .is the distance calculated at the lth sample between the pth EEG signal and the jth prototype (cluster centre) for the ith channel.

Similarly, for the time instant *k-1* we have:

$$P\left(x_j^*(k-1)\right) = \frac{1}{1 + \dfrac{1}{k-2}\sum_{p=1}^{k-2}\sum_{i=1}^{n+1}\sum_{l=1}^{p}(d_l^i(j,p))^2} \; ; \; k=3,4,... \tag{4.6}$$

Thus the potential of an existing prototype EEG signal can be expressed *recursively* from its potential value at the previous time instant (i.e. before the new data sample is available) as:

$$P\left(x_j^*(k)\right) = \frac{(k-1)P(x_j^*(k-1))}{k-2 + P\left(x_j^*(k)\right) + P\left(x_j^*(k-1)\right)\sum_{i=1}^{n+1}\sum_{l=1}^{L}\left(d_l^i(k,k-1)\right)^2} \tag{4.7}$$

16

The *on-line* classification procedure can be summarised as follows:

1. Accept the first *EEG* signal as the first prototype. This is used to form the antecedent part of the fuzzy rule and its *potential* is set to 1.

2. Starting from the next *EEG* signal for all subsequent *EEG* signals the *potential* of each new signal is calculated *recursively* using equation (4.4).

3. The potentials of existing prototypes are *recursively* updated using equation (4.7).

4. The *potential* of the *new EEG* input signal is compared to the updated *potentials* of the existing prototypes. Then

   (a) If the potential of the **new** *EEG* signal is **higher** than the potential of the existing prototypes then the **new** *EEG* signal is added as a **new prototype** and a **new rule** is formed ($x_R^* = x(k)$ and the number of rules in the rule-base gradually increases ($R := R+1$). The condition used in this case i.e. of having a "higher" potential, limits the generation of excessively large rule base;

   (b) If in addition to the previous condition the **new** *EEG* signal is close to an old prototype then the **new** *EEG* signal, *x(k)* replaces this prototype ($x_j^* := x(k)$).

This *on-line* clustering approach results in an *evolving* rule-base, by recursively upgrading and modifying the rule-base at every instant of time while inheriting the bulk of the rules from the previous time instant (*R*- of the rules are preserved even when a modification or an upgrade take place).

5. EXPERIMENTAL RESULTS AND DISCUSSION

As mentioned in section 2, EEG data were recorded on three different occasions from a healthy female subject and on each of these days; three EEG recordings were taken by directing a laser beam on the right arm of the subject. In total, nine EEG data files were obtained. The first eight files were used to obtain the required "training" data whereas the last file provided data for "testing" the classification performance of the proposed systems..

The overall HMM based experimental procedure is shown in Figure 5.1 and involves the network training/design and system classification performance evaluation processes for the conventional IM-HMM-D structure and the new DM-HMM-D system. A useful by-product of the second technique is the instantaneous "weight" information that is attached by DM-HMM-D to each input signal. This information reflects the importance of each EEG channel/signal for achieving maximum classification performance. Note that the number of hidden states of each HMM network in both the IM-HMM-D and DM-HMM-D systems is N. This was experimentally fixed to $N=10$ whereas the resolution of the input scalar quantization process used assumed values $M=100, 50, 20$ and 10 possible values. Notice that M is also equal to the number of different discrete observation values that can be produced from a network state.

Table 5.1 provides a comparative list of the classification performance results obtained from the IM-HMM-D and DM-HMM-D schemes. When performance is calculated as an average value using both classes, the IM-HMM-D system delivers 69.11765% with $M=20$; and a maximum 72.0588% with $M=10$. The relatively low performance of IM-HMM-D in classifying pain can be explained in terms of the substantial inter-dependencies which exist between certain channels and which this system fails to take into consideration. Considerably higher classification accuracy rates are obtained by applying the DM-HMM-D system, see Table 5.1. This improved performance is obtained for all values of M and the system operates best (with a performance in excess of 95%) with M values in the region of 20 to 50. Note that when several different models provide similar classification performance, structure with lower values of M and N are preferred due to lower system complexity. Thus the model with $M=20$ and $N=10$ is used below to obtain the "usage rate" for each EEG channel and hence an indication of the significance of each channel.

Note that, some sequences can be completely blocked out, if necessary, from the resulting likelihood probability P by setting a minimum threshold $W_{threshold}$ that operates during classification on $P(O^{(i)}|Y)$ values stored in the predefined "dependency" codebook. In the reported experiments,

$W_{threshold} = 10^{-5}$. Figure 5.2 shows the usage rate of each input channel (feature) as calculated from a classification experiment with DM-HMM-D. In this figure certain channels (for example, Channels 4, 6, 15, 16, 39, 48, 49, 51, and 58) are heavily involved in the classification of both *Pain* and *No Pain* conditions. In general, this input channel categorization methodology can be particularly useful to researchers interested in the reduction of the number of input channels (features) presented at the input of a classifier with a carefully controlled effect on classification performance.

Experimentation with the novel *eClass* system involved a data-set of 355 Pain and 355 No-Pain epochs, in order to produce the fuzzy rule-base of the classifier. As before, these epochs came from the first 8 out of the available 9 *EEG* data-sets (experiments). Thus 710 signal epochs were introduced to the system together with their respective Pain/No Pain labels, (ground-truth). It should be noted that the *eClass* approach does not need to be pre-trained, but in order to compare the performance of the two methods; the same experimental conditions were used here. In this way the system produced a "rule-set" for each of the Pain and No-Pain conditions, by comparing spatial proximities between signals of the same class. Following the merging of these two individual "rule-sets", the system was ready to classify unknown signals based on the "global rule-set" that contained a total of 168 prototypical EEG signals (rules).

In order to test the classification efficiency of the resulting system, 34 epochs from each class taken from the 9th data-set that was used for testing the system, were presented to *eClass*. Classification was thus performed by comparing the distances formed between the input test EEG signals and the prototypes stored in the "rule-set".

Two different approaches were then used in order to arrive at the required classification decision:

1. A "winner takes all" classification approach (see equation 4.2a).

2. A weighted average approach (see equation 4.2b).

Using the first approach, *eClass* yielded a respectable 79.45% average classification accuracy rate. Overall classification accuracy is significantly lower (64%) when using the second approach. Thus

experimental results illustrated that employing the "aggressive" winner-takes-all type of decision outperforms the version that is based on the weighted type of classification. It should be noted that a classification accuracy of 88.24% is achieved for the No-Pain class whereas the corresponding result for the Pain condition is 69.21%. This can be associated to the observation that the number of rules formed by the system when modelling No-Pain EEG signals is significantly larger than that of the Pain case. Note that the *eClass* approach does not require the number of prototypes to be pre-specified; instead prototypes are formed according to the characteristics of the input signals and as a result of an evolving design procedure. The reason why more clusters and hence prototypes are obtained in the No-Pain case can be traced to the higher variance of the EEG signals recorded under the Pain condition, as compared to those recorded under No-Pain. Higher variance (Figure 4.1b) leads to lower potentials (equation (4.4)) which in turn restricts the process of generating new prototypes.

6. CONCLUSIONS

Two novel and substantially different approaches to the problem of automatically deciding on the Pain/No Pain condition of a subject using *EEG* signals are introduced in the paper. The first approach, DM-HMM-D, is based on a new formulation of a bank of HMM with discrete density classifiers where each HMM operates on a different channel. Furthermore DM-HMM-D exploits any inter-dependencies that may exist between the EEG signals of different channels via the introduction per HHM model of a varying with time weighting function that represents the instantaneous importance of each channel. This system performs substantially better than the conventional IM-HMM-D approach and with a Pain/No Pain classification accuracy of 94% and 97% respectively. Whereas this HMM based approach to *EEG* Pain/No Pain condition classification requires **off-line** supervised training of the system, in order to specify the model parameters whose architecture is **fixed** (and should be therefore determined via experimentation), the second classification approach, i.e. *eClass*, is far more **flexible** and defines its fuzzy rule-based structure **on line** and in response to the input *EEG* signals presented to the system during its training process. This *evolving* rule based characteristic is a

major asset of *eClass* and an important differentiator, with respect to other *off-line*, fixed structure classification systems, since it opens up significantly the way that the classifier can be used in practical ***real-time*** applications. Furthermore experimental results clearly demonstrated the potential of the *eClass* system whose underlining methodology bears the promise of further significant research developments on supervised *on line* classification systems.

7. ACKNOWLEDGEMENTS

REFERENCES

[1] J.R. Wolpaw, D.J McFarland, Multichannel EEG-based brain-computer communication, *Electroencephalography and Clinical Neurophysiology* **90,** 444-449 (1994).

[2] S.W.G. Derbyshire, Jones A.K.P., "Cerebral responses to a continual tonic pain stimulus measured using positron emmision tomography", Pain 76 (1998) 127 – 135.

[3] P. Rappelsberger, Petsche H: "Probability Mapping: Power and Coherence Analyses of Cognitive Processes", Brain Topography, 1988, 1: 46-54.

[4] Chen A.-C.N., Rappelsberger P: "Brain and Human Pain: Topographic and Coherence Mapping", Brain Topography, 1994, 7, No 2: 129-140.

[5] L.K.Stergioulas, Reoullas M., Xydeas C.S., Baltas E., Bentley D., Youell P., Jones A.: "Coherence Pattern Classification Using LVQs For Pain Detection", Analysis of Biomedical Signals and Images, 16th Biennial International EURASIP Conference BIOSIGNAL 2002 Proceedings, Vutium Press 2002: 56 – 58.

[6] D.E. Bentley., P.D. Youell., A.K.P. Jones.: "Anatomical localization and intra-subject reproducibility of laser evoked potential source in cingulated cortex, using a realistic head model", Clinical Neurophysiology 113 (2002) 1351 – 1356.

[7] J. Bilmes, A Gentle Tutorial of the EM Algorithm and its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models, *Tech. Report*, International Computer Science Institute, Berkeley, 1997.

[8] L. R. Rabiner, A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition, *Proc. of the IEEE,* **77 (2),** 257-286 (1989).

[9] Z. Ghahramani, An Introduction to Hidden Markov Models and Bayesian Networks, *International Journal of Pattern Recognition and AI*, **15 (1),** 9-42 (2001).

[10] Y. Bengio, Markovian Models for Sequential Data, *Neural Computer Surveys*, Dept. of Information and Research Operation, **2,** 129-162 (1999).

[11] M. A. Mohamed, P. Gader, Generalized Hidden Markov Models-Part I: Theoretical Frameworks, *IEEE Transaction on Fuzzy Systems*, **8 (1),** (February 2000).

[12] M. A. Mohamed, P. Gader, Generalized Hidden Markov Models-Part II: Application to Handwritten Word Recognition, *IEEE Transaction of Fuzzy Systems*, **8 (1),** (February 2000).

[13] K. F. Lee, Automatic Speech Recognition: the development of the SPHINX system, Kluwer Academic Publish, (1989).

[14] P. Angelov, D. Filev, An approach to on-line identification of evolving Takagi-Sugeno models, *IEEE Trans. on Systems, Man and Cybernetics, part B*, **34 (1)**, 484-498 (2004).

[15] N. Kasabov, Q. Song, DENFIS: Dynamic Evolving Neural-Fuzzy Inference System and Its Application for Time-Series Prediction, *IEEE Trans. on Fuzzy Systems* **10(2)**, 144-154 (2002).

[16] S.-Y. Chiao, C. S. Xydeas, Using Hierarchical HMMs in Dynamic Behaviour Modelling, *Proceedings of Seventh International Conference on Information Fusion, Fusion 2004,* Stockholm Sweden, 576-582 (2004).

[17] S.-Y. Chiao, C. S. Xydeas, Behaviour Modelling Using a Hierarchical HMM Approach, *IEEE Computer Society Press Proceedings*, (May 2005 to appear).

[18] R. R. Yager, D. P. Filev, *Essentials of Fuzzy Modeling and Control*, NY: John Wiley (1994).

[19] P.P. Angelov, *Evolving Rule-based Models:* A Tool for Design of Flexible Adaptive Systems, Springer, Physica-Verlag, Heidelberg, Germany (2002).

[20] R. O. Duda, D. G. Stork, P. E. Hart, *Pattern Classification and Scene Analysis*: Pattern Classification, 2nd edition, John Wiley, NY (2001).

[21] J. Bezdek, Cluster Validity with Fuzzy Sets, *Journal of Cybernetics*, **3 (3),** 58-71 (1974).

[22] I. Gath, A.B. Geva, Unsupervised optimal fuzzy clustering, *IEEE Trans, Pattern Analysis and Machine Intelligence,* **7,** 773-781 (1989).

[23] S.L. Chiu, Fuzzy Model Identification based on Cluster Estimation, *Journal of Intelligent and Fuzzy Systems*, **2,** 267-278 (1994).

[24] N. Kasabov, Evolving fuzzy neural networks for on-line supervised/unsupervised knowledge-based learning, *IEEE Trans. SMC - part B, Cybernetics* **31**, 902-918 (2001).

[25] D. Carline , P. Angelov, R. Clifford, Agile Collaborative agents for classification of under-water targets, *Undersea Defence Technology*, 21-23 June 2005, Amsterdam, The Netherlands.

[26] P. Angelov, An Approach for Fuzzy Rule-base Adaptation using On-line Clustering, *International Journal of Approximate Reasoning*, **Vol. 35** (3), March 2004, Pages 275-289.

[27] L. Kuncheva, How Good are Fuzzy If-Then Classifiers, IEEE Transactions on Systems, Man and Cybernetics-part B, **Vol** 30 (4) 501-509.

Figure 2.1: EEG cap in its physical form mounted on the scalp of a subject
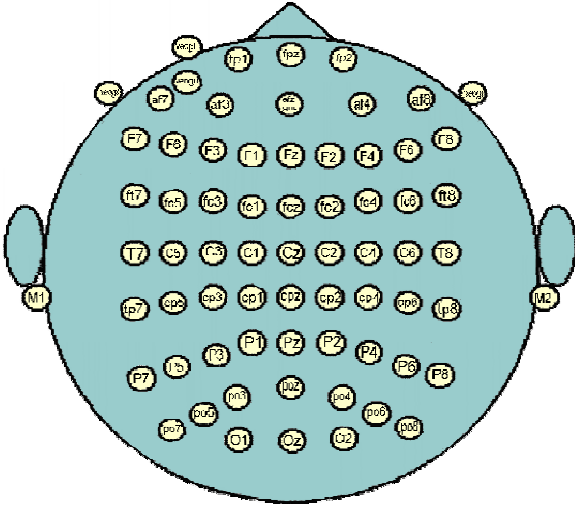

Figure 2.2: The 64-electrode cap and its relevant positions together with the channel names
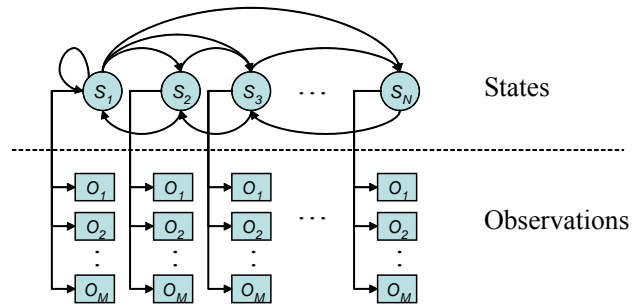
Figure 3.1: A general HMM structure with *N* hidden states and *M* possible observations per state.
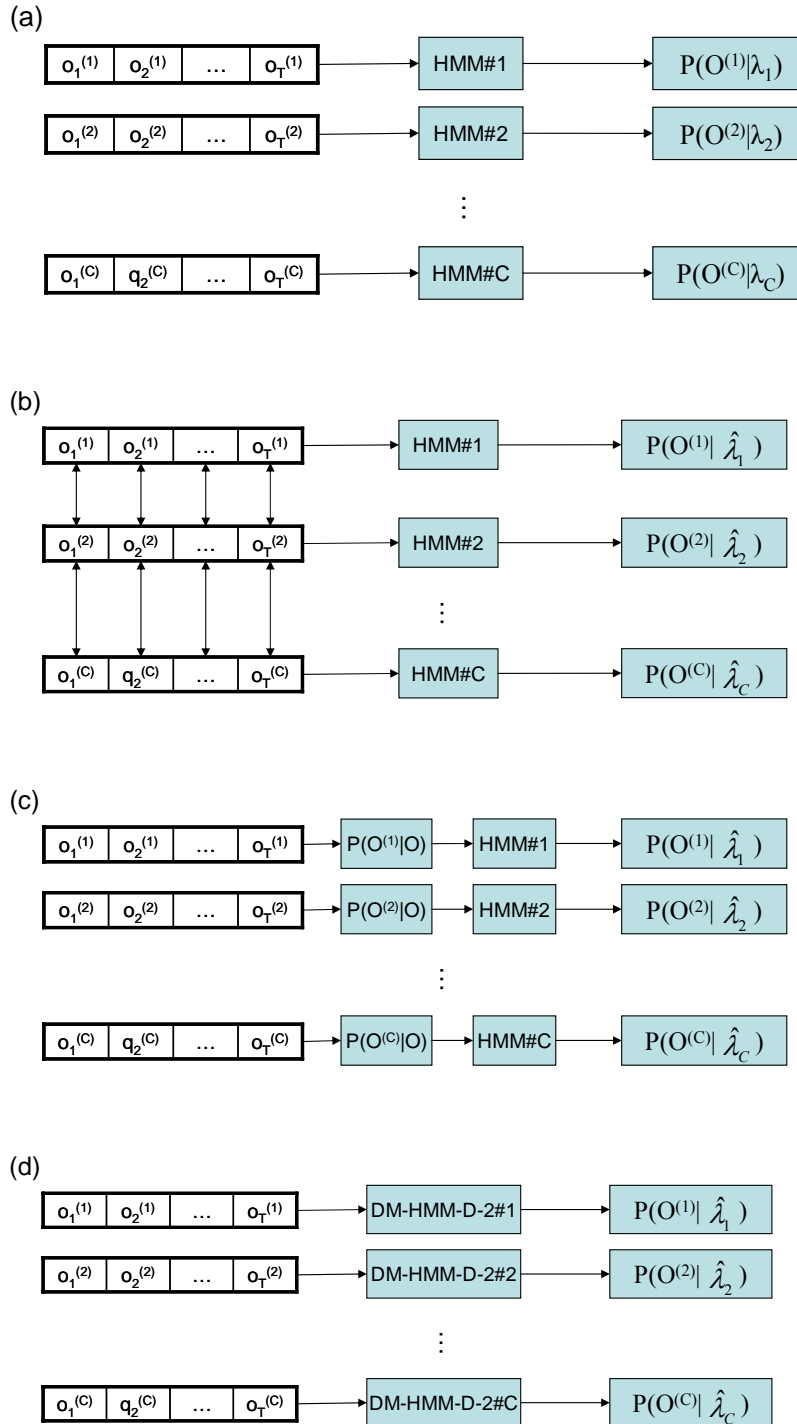
(a)

(b)

(c)

(d)

Figure 3.2: (a) a conventional IM-HMM-D, (b) (c) and (d) DM-HMM-D equivalent
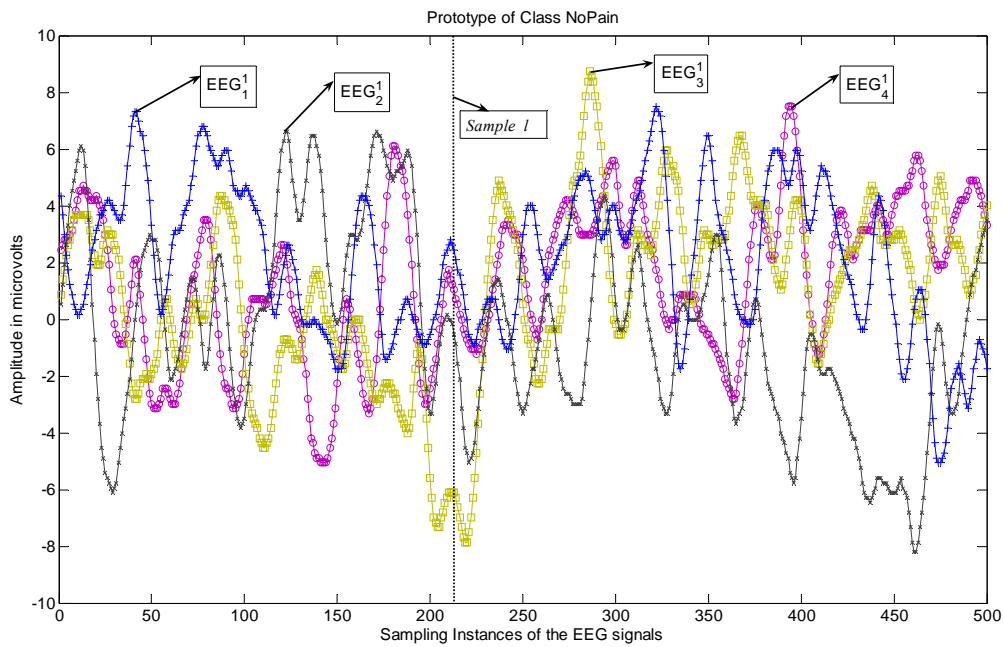
structures.

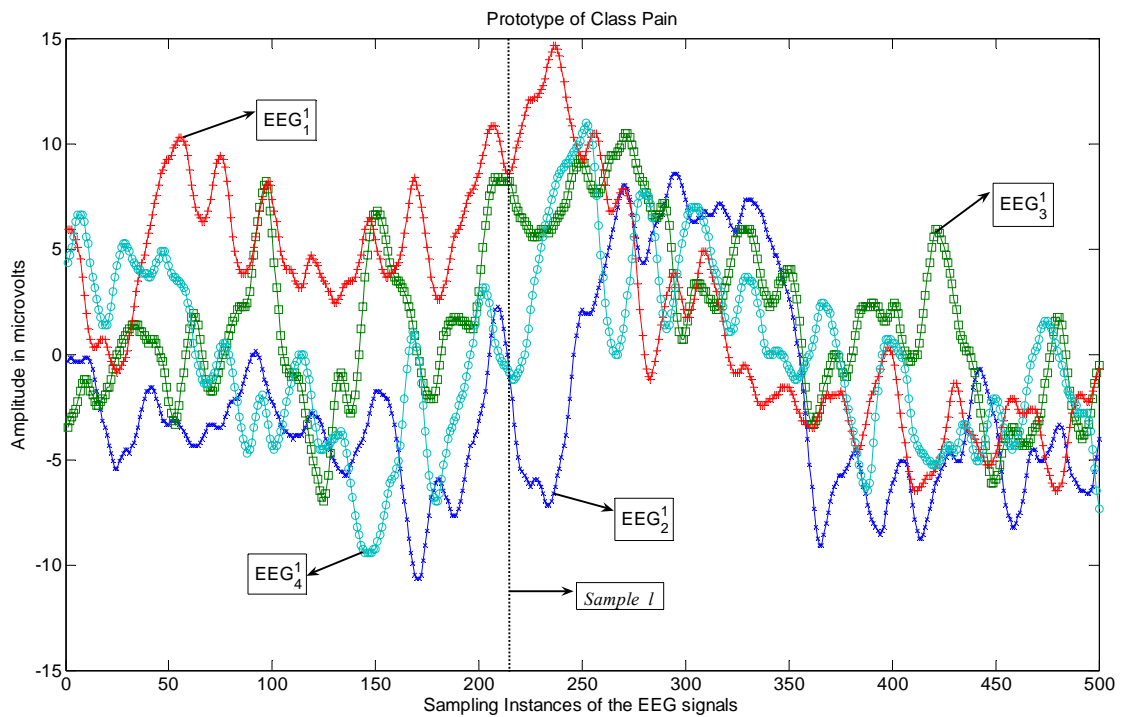Figure 4.1 (a) Prototype EEG signals for the class No Pain for one of the channels

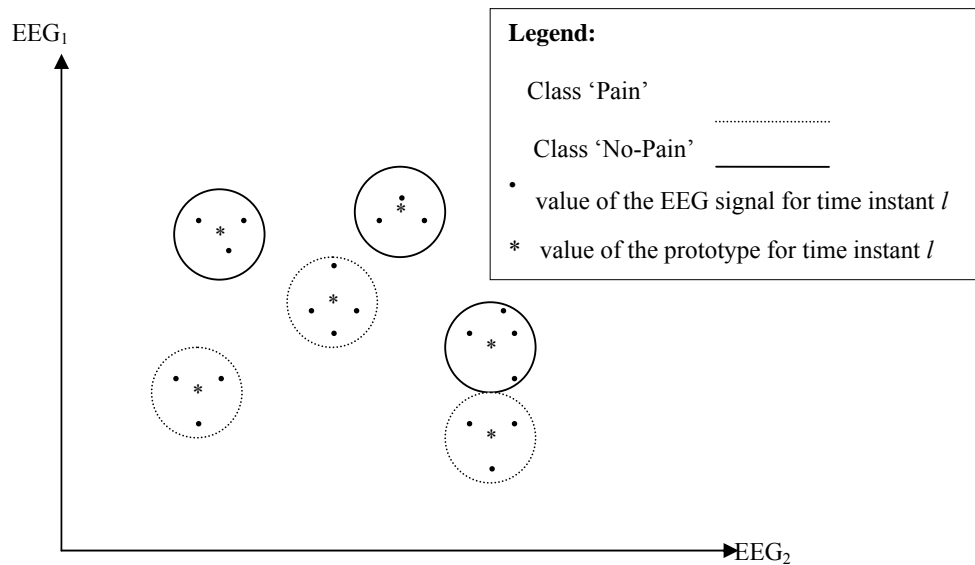Figure 4.1 (b) Prototype EEG signals for the class Pain for the same channel

Figure 4.2   A snap-shot of the clustering for certain moment of time l after discretization
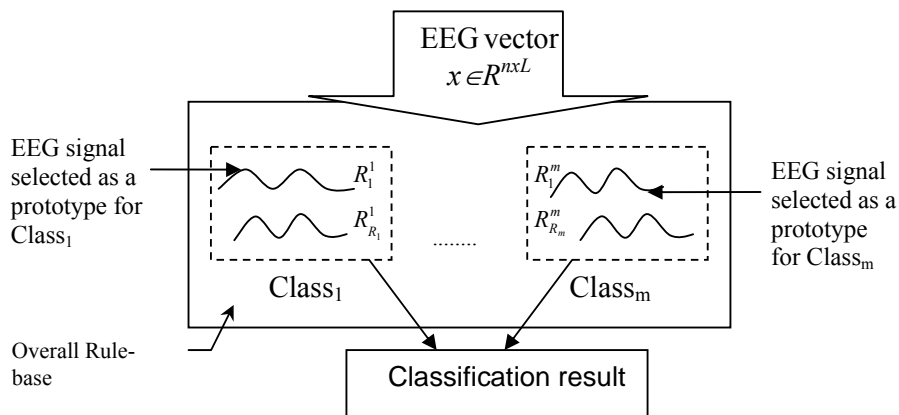
of EEG signals for the two channels

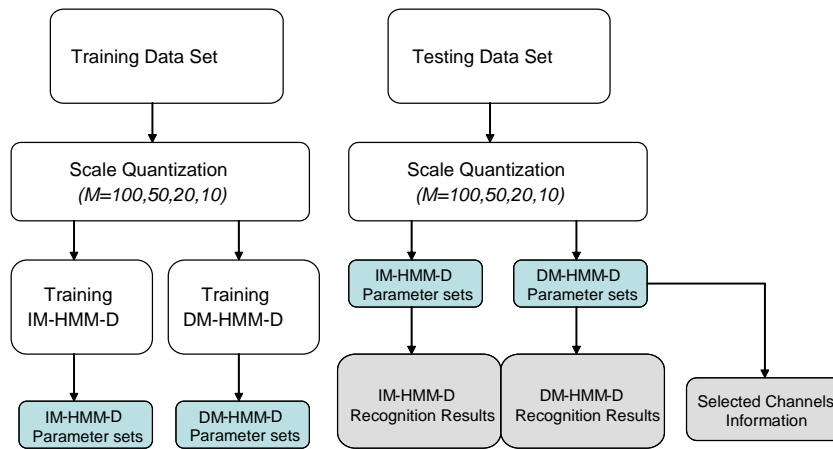Figure 4.3: General Model of the eClass scheme.

Figure 5.1: Design and evaluation procedures in EEG classification experiments using

HMMs

Scale Model	100		50		20		10	
	Pain	No Pain	Pain	No Pain	Pain	No Pain	Pain	No Pain
IM-HMM-D	41.1765%	97.0588%	40.5882%	97.0588%	47.0588%	91.1765%	52.9412%	91.1765%
DM-HMM-D	91.1765%	97.0588%	97.0588%	94.1176%	97.0588%	94.1176%	55.8824%	92.9412%

Formatted: Font: (Default) Arial, 11 pt

Table 5.1: Classification results for the IM-HMM-D and DM-HMM-D systems. Classification performance is expressed as the percentage of correctly classified EEG segments
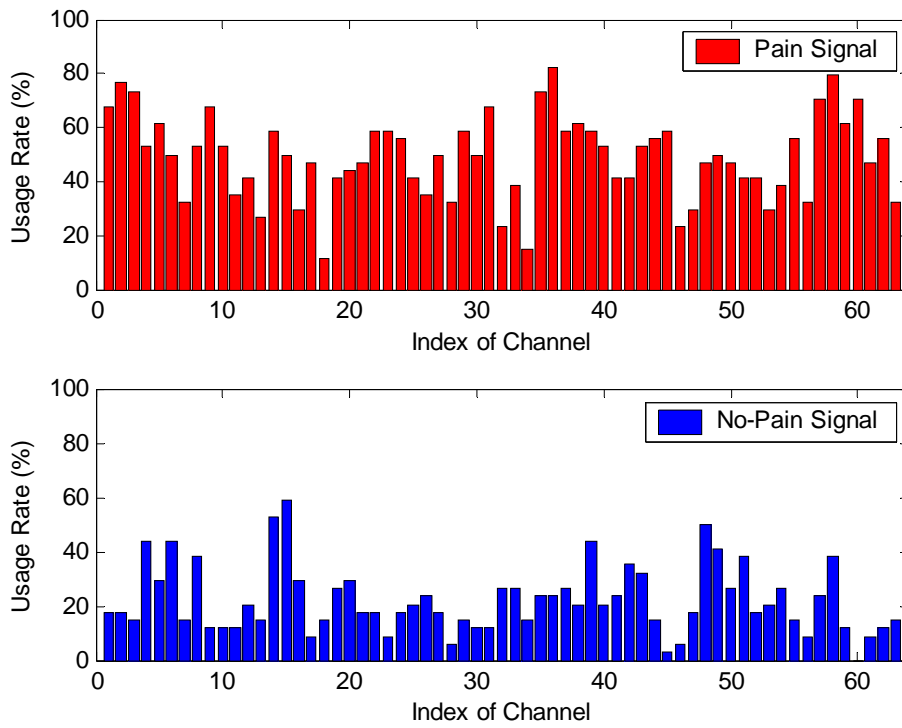
Figure 5.2: Channel usage rates in DM-HMM-D Pain/No-Pain classification