

**SELECTION AND OPTIMIZATION OF PEAK FEATURES
FOR EVENT-RELATED EEG SIGNALS
CLASSIFICATION**

ASRUL BIN ADAM

**FACULTY OF ENGINEERING
UNIVERSITY OF MALAYA
KUALA LUMPUR**

2017

**SELECTION AND OPTIMIZATION OF PEAK
FEATURES FOR EVENT-RELATED EEG SIGNALS
CLASSIFICATION**

ASRUL BIN ADAM

**THESIS SUBMITTED IN FULFILMENT OF THE
REQUIREMENTS FOR THE DEGREE OF DOCTOR
OF PHILOSOPHY**

**FACULTY OF ENGINEERING
UNIVERSITY OF MALAYA
KUALA LUMPUR**

2017

UNIVERSITY OF MALAYA

ORIGINAL LITERARY WORK DECLARATION

Name of Candidate: Asrul Bin Adam (I.C/Passport No: 851028-04-5063)

Registration/Matric No: KHA130017

Name of Degree: Doctor of Philosophy

Title of Thesis ("this Work"): Selection and Optimization of Peak Features for Event-related EEG Signals Classification

Field of Study: Signal and Systems

I do solemnly and sincerely declare that:

- (1) I am the sole author/writer of this Work;
- (2) This Work is original;
- (3) Any use of any work in which copyright exists was done by way of fair dealing and for permitted purposes and any excerpt or extract from, or reference to or reproduction of any copyright work has been disclosed expressly and sufficiently and the title of the Work and its authorship have been acknowledged in this Work;
- (4) I do not have any actual knowledge nor do I ought reasonably to know that the making of this work constitutes an infringement of any copyright work;
- (5) I hereby assign all and every rights in the copyright to this Work to the University of Malaya ("UM"), who henceforth shall be owner of the copyright in this Work and that any reproduction or use in any form or by any means whatsoever is prohibited without the written consent of UM having been first had and obtained;
- (6) I am fully aware that if in the course of making this Work I have infringed any copyright whether intentionally or otherwise, I may be subject to legal action or any other action as may be determined by UM.

Candidate's Signature

Date:

Subscribed and solemnly declared before,

Witness's Signature

Date:

Name:

Designation:

ABSTRACT

The classification of desired peaks in event-related electroencephalogram (EEG) signals becomes a challenging problem for brain signals researchers. The reasons are mainly because of the peak in the signals have been contaminated with various noises, the nature of non-stationary EEG signals, many peaks candidates in the signals, and the peak features relative to the baseline amplitude, time, and different users. Many peak classification algorithms have been introduced for various EEG signals applications. However, the developed algorithms only consider the selected features from a peak model based on the understanding of the EEG signals characteristics. The utilization of different existing models cannot assure giving the best classification performance for other event-related EEG signals applications. For a fair performance evaluation, the selection of the best peak model requires experimental exploration by using a common and unbiased classification approach. This thesis aims to provide a high and good generalized peak classification performance through the application of an optimization approach with the advantageous of a common classification method for finding a new optimal combination of peak features. At first, a peak classification algorithm is developed based on the general following processes including peak candidate identification, feature extraction, and classification. Four different existing peak models with the associated features and full features set model are considered as inputs to the classifier. The four peak models are named as Dumpala, Acir, Liu, and Dingle models whereas the full features set model consists of 16 peak features. Three event-related EEG signals that recorded from 30 voluntary of healthy subjects, namely as a single eye blink, double eye blink, and eye movement signals are employed. All subjects are instructed to direct their eye blinks and horizontal gaze in response to a voice cue. In the preliminary study, the algorithm is evaluated on the four different peak models of the

three EEG signals using the artificial neural network (ANN) with particle swarm optimization (PSO) as learning algorithm. Unfortunately, the ANN classification method cannot provide the fast learning speed once it integrates with the PSO. The study continued with other classification technique which is neural network with random weights (NNRW). Next, four recently introduced optimization algorithms are employed as feature selector, namely as 1) angle modulated simulated Kalman filter (AMSKF), 2) binary simulated Kalman filter (BSKF), 3) local optimum distance evaluated simulated Kalman filter (LocalDESKF), and 4) global optimum distance evaluated simulated Kalman filter (GlobalDESKF). This study resulted in a new generalized model based on the best performance among the four novel simulated Kalman filter (SKF) approaches. The new generalized models with the associated features that are selected using the novel feature selection approaches have substantially improved the performance of the existing models. The proposed models and NNRW method in this thesis perform at par with the existing related studies of epileptic EEG events classification.

ABSTRAK

Pengesanan titik puncak dalam isyarat elektroensefalogram (EEG) menjadi masalah yang sangat mencabar untuk dikesan kepada penyelidik isyarat otak. Ini disebabkan kerana puncak pada isyarat telah dicemari dengan pelbagai hingar, sifat isyarat EEG yang tidak bergerak, banyak puncak palsu pada isyarat, dan ciri-ciri puncak pada isyarat EEG relatif kepada amplitud garis dasar, masa, dan pengguna yang berbeza. Terdapat pelbagai algoritma pengesanan puncak yang telah diperkenalkan untuk pelbagai jenis aplikasi dalam isyarat EEG. Walau bagaimanapun, algoritma yang telah dibangunkan hanya mengambil kira ciri-ciri terpilih daripada model puncak berdasarkan pemahaman asas ciri-ciri dari isyarat EEG. Penggunaan model yang sedia ada tidak menjamin memberi prestasi pengesanan yang terbaik kepada aplikasi EEG isyarat yang lain. Untuk menilai secara saksama, pemilihan model puncak yang terbaik memerlukan penerokaan secara eksperimen dengan menggunakan pendekatan klasifikasi umum dan tidak berat sebelah. Dalam kajian ini, kami berhasrat untuk membina pengesanan puncak yang mempunyai prestasi yang terbaik. Pada mulanya, kami telah membangunkan algoritma pengesanan puncak berdasarkan proses berikut termasuk mengenalpasti puncak calon, pengekstrakan ciri, dan klasifikasi. Empat model sedia ada dengan ciri-cirinya adalah dianggap sebagai input kepada pengelas. Empat model puncak tersebut adalah Dumpala, Acir, Liu, dan Dingle dan semua ciri model terdiri daripada 16 ciri-puncak. Tiga aktiviti daripada isyarat EEG direkodkan daripada 30 peserta kajian yang sihat, iaitu aktiviti mata berkelip satu kali, mata berkelip dua kali, dan isyarat pergerakan mata digunakan dalam kajian. Semua peserta kajian telah diarahkan untuk menggerakkan mata mereka berkelip dan pergerakan secara mendatar sebagai tindak balas kepada isyarat suara. Pada kajian awal, algoritma dinilai menggunakan model-model puncak untuk setiap aktiviti isyarat EEG dengan menggunakan kecerdikan rangkaian neural (ANN). Malangnya, klasifikasi kaedah ANN

tidak dapat memberi kelajuan pembelajaran yang pantas apabila ia digabungkan dengan teknik pengoptimuman. Kajian ini diteruskan dengan menggunakan teknik lain iaitu rangkaian neural yang berwajaran rawak (NNRW). Seterusnya, empat teknik terbaru diperkenalkan untuk mengenalpasti ciri-ciri puncak yang terbaik. Teknik tersebut adalah sudut modulat simulasi turas Kalman (AMSKF), perduaan simulasi turas Kalman (BSKF), optimum setempat jarak ternilai simulasi turas Kalman (LocalDESKF), dan optimum global jarak ternilai simulasi turas Kalman (GlobalDESKF). Kajian ini menghasilkan model umum baru berdasarkan prestasi yang terbaik di kalangan empat teknik tersebut. Model umum baru bersama ciri-ciri puncaknya yang dipilih menggunakan empat teknik tersebut telah berupaya meningkatkan prestasi model puncak yang sedia ada. Model-model yang dicadangkan dan teknik NNRW di dalam tesis ini mempunyai prestasi yang setara dengan kajian sedia ada daripada klasifikasi acara isyarat EEG untuk penyakit epilepsi.

ACKNOWLEDGEMENTS

First and foremost, the deepest gratitude of all shall be bestowed to Allah the Almighty and The Merciful for all the insight which He gave to us that lead to the completion of this study.

I would like to express my gratitude and appreciation to my main supervisor and co-supervisor, Dr. Norrima Mokhtar and Associate Professor Dr. Zuwairie Ibrahim, respectively. They gave me a good guidance, support, and motivation throughout the completion of this study. Without they encouragement, enthusiasm and positive critics, this thesis will not even be possible.

Also, special words of thanks are due to all my research colleagues, Dr. Ibrahim Shapiai for his patience in editing revisions of the manuscripts and friends, especially Mr. Zulkifli Mohd Yusof, Dr. Marizan Mubin, and Dr. Ismail Ibrahim for providing help and support in accomplishing this study.

Finally, mention should be made to the forbearance and patience of my beloved father, mother, wife, and sons Mr. Adam bin Abdullah, Mrs. Noraini binti Hj. Ali, Mrs. Khadijah binti Kamal, Wafiy bin Asrul, and Ahsan bin Asrul in which they are my main source of purpose and inspiration, during the period of this research and also throughout whole my life. Last but not least, an expression and gratitude to all individuals who are involved either directly or indirectly in making this project. Many thanks to High Impact Research Fund (UM.C/HIR/MOHE/ENG/16 Account code: D000016-16001) for financially support the research activities. Also, I would like to thank the Ministry of Higher Education Malaysia for supporting my study by awarding me a MyPhD scholarship.

TABLE OF CONTENTS

ABSTRACT	III
ABSTRAK	V
ACKNOWLEDGEMENTS	VII
TABLE OF CONTENTS	VIII
LIST OF FIGURES	XII
LIST OF TABLES.....	XIV
LIST OF SYMBOLS AND ABBREVIATIONS.....	XVII
CHAPTER 1: INTRODUCTION	22
1.1 Research Background.....	22
1.2 The Significance of Peak Detection Algorithm in Event-related Signals Classification.....	23
1.3 The Definition of a Peak in the EEG Signals	24
1.4 Motivation and Problem Statements	25
1.5 Objectives of the Study	27
1.6 Main Contributions.....	28
1.6.1 First Contribution	28
1.6.2 Second Contribution.....	29
1.6.3 Third Contribution.....	29
1.7 Outline of Thesis.....	30
CHAPTER 2: LITERATURE REVIEW	32
2.1 Introduction.....	32
2.2 The Existing Algorithms for Peak Classification	32
2.3 The Existing Peak Models based on the Time Domain Analysis.....	34
2.4 The Existing Feature Selection Methods using Optimization Algorithms for EEG Signals Peak Classification	37

2.5	Summary	39
CHAPTER 3: PEAK CLASSIFICATION USING THE EXISTING MODELS ...		41
3.1	Introduction.....	41
3.2	Experimental Protocols and Data Collection.....	43
3.3	Performance Measure.....	49
3.4	Feature Extraction.....	49
3.5	Peak Model Selection	54
3.6	Classification Methods	54
3.6.1	Artificial Neural Network with Particle Swarm Optimization as Learning Algorithm for Artificial Neural Network Classifier	54
3.6.1.1	Experiment 1: Evaluation on Different Peak Models for EEG Signals Peak Detection using Artificial Neural Network	60
3.6.1.2	Experimental Results	61
3.6.2	Neural Network with Random Weights Classifier.....	67
3.6.2.1	Experiment 2: Evaluation on Different Peak Models using NNRW Classifier	70
3.6.2.2	Experimental Results	71
3.7	Performance Comparison of Different Peak Models between the ANNPSO and NNRW Classifiers	76
3.8	Summary	77
CHAPTER 4: THE PROPOSED GENERALIZED MODEL FOR PEAK CLASSIFICATION OF EVENT-RELATED EEG SIGNALS.....		78
4.1	Introduction.....	78
4.2	Research Methodology.....	79
4.3	Preparing Training, Validation, and Testing Sets.....	80
4.4	Performance of NNRW under Various Number of Hidden Neurons	80
4.5	The Proposed Generalized Model I based on Angle Modulated Simulated Kalman Filter (AMSKF) Algorithm.....	84

4.5.1	Simulated Kalman Filter (SKF) Algorithm	85
4.5.2	Angle Modulated Simulated Kalman Filter (AMSKF) Algorithm	88
4.5.3	Feature Selection using Angle Modulated Simulated Kalman Filter....	90
4.5.4	Experiment 1: To Find a Generalized Model based on AMSKF Feature Selection Approach	93
4.5.4.1	Experimental Results and Discussions	93
4.5.4.2	Statistical Significance Analysis	96
4.6	The Proposed Generalized Model II based on Binary Simulated Kalman Filter ..	98
4.6.1	Binary Simulated Kalman Filter (BSKF) Algorithm	98
4.6.2	Feature Selection using Binary Simulated Kalman Filter	100
4.6.3	Experiment 2: To Find a Generalized Model based on BSKF Feature Selection Approach.....	103
4.6.3.1	Experimental Results and Discussions	103
4.6.3.2	Statistical Significance Analysis	105
4.7	The Proposed Generalized Model III based on Local Optimum Distance Evaluated Simulated Kalman Filter.....	108
4.7.1	Local Optimum Distance Evaluated Simulated Kalman Filter	108
4.7.2	Feature Selection using LocalDESKF	109
4.7.3	Experiment 3: To Find a Generalized Model based on LocalDESKF- NNRW Feature Selection Approach.....	112
4.7.3.1	Experimental Results and Discussions	112
4.7.3.2	Statistical Significance Analysis	114
4.8	The Proposed Generalized Model IV: Global Optimum Distance Evaluated Simulated Kalman Filter	117
4.8.1	Global Optimum Distance Evaluated Simulated Kalman Filter	117
4.8.2	Feature Selection using GlobalDESKF.....	118
4.8.3	Experiment 4: To Find a Generalized Model IV based on GlobalDESKF Feature Selection Approach.....	121
4.8.3.1	Experimental Results and Discussions	121
4.8.3.2	Statistical Significance Analysis	123
4.9	Performance Comparison of the Proposed Generalized Models and the Existing Models.....	126
4.10	Summary	128

CHAPTER 5: APPLICATION OF THE PROPOSED PEAK MODELS TO EPILEPTIC CLASSIFICATION EVENTS IN EEG SIGNALS.....	130
5.1 Introduction.....	130
5.2 Data Description	131
5.2.1 Preparing Training and Testing Sets	132
5.3 Results and Discussions	133
5.3.1 Experiment 1: Performance Evaluation of Various Number of Hidden Neuron	133
5.3.2 Experiment 2: Epileptic and Non-epileptic Event Classification.....	135
5.4 Summary	138
CHAPTER 6: CONCLUSIONS AND SUGGESTIONS FOR FUTURE RESEARCH	140
6.1 Introduction.....	140
6.2 Conclusions.....	140
6.3 Future Research	142
REFERENCES	144
LIST OF PUBLICATIONS AND PAPERS PRESENTED	153
APPENDIX A – MEDICAL ETHICS COMMITTEE APPROVAL.....	155

LIST OF FIGURES

Figure 3.1: Flowchart of peak classification using the selection of the existing models	42
Figure 3.2: Data collection session.....	44
Figure 3.3: The filtered EEG signals: (a) single eye blink (2 peak points per signal), (b) double eye blink (8 peak points per signal), and (c) eye movement (one peak point per signal)	48
Figure 3.4: Eight point locations of a peak	50
Figure 3.5: Architecture of a single layer feedforward ANN	55
Figure 3.6: ANN with PSO algorithm for peak detection algorithm	59
Figure 3.7: Average testing accuracy with respect to the number of hidden neurons: (a) single eye blink signals, (b) double eye blink signals, and (c) eye movement signals ...	74
Figure 4.1: Variation of testing accuracy of NNRW classifier with respect to number of hidden neurons.....	82
Figure 4.2: The simulated Kalman filter (SKF) algorithm	86
Figure 4.3: The angle modulated simulated Kalman filter (AMSKF) algorithm	89
Figure 4.4: An example of $g(x)$ function with $a = 0$, $b = 1$, $c = 1$, and $d = 0$	90
Figure 4.5: Flowchart of the proposed AMSKF feature selection algorithm	92
Figure 4.6: Example of the convergence curve of AMSKF	95
Figure 4.7: Flowchart of the proposed BSKF feature selection algorithm	102
Figure 4.8: Example of the convergence curve of BSKF	105
Figure 4.9: Flowchart of the proposed LocalDESKF feature selection algorithm.....	111
Figure 4.10: Example of the convergence curve of LocalDESKF	114
Figure 4.11: Flowchart of the proposed GlobalDESKF feature selection algorithm....	120
Figure 4.12: Example of the convergence curve of GlobalDESKF	123
Figure 5.1: Example of five different sets of EEG signals taken from Bonn University EEG database (Polat & Gunes, 2008)	132

Figure 5.2: Selection of optimal number of hidden neuron from training accuracy 134

Figure 5.3: Example of epileptic event classification using record Z001 and S001 137

Figure 5.4: Example of misclassification of epileptic event in record Z083 and S083 137

LIST OF TABLES

Table 2.1: Summary of the previous research works using various types of peak detection algorithms on different peak models for various applications	37
Table 3.1: Description of the signals.....	48
Table 3.2: Equations and descriptions of peak features	53
Table 3.3: List of different peak models and sets of feature.....	54
Table 3.4: Representation of particle position, particle velocity, p_{best} , and g_{best}	59
Table 3.5: Parameters setting of ANN	61
Table 3.6: Parameters setting of PSO.....	61
Table 3.7: Performance of peak detection models for single eye blink signals	64
Table 3.8: Sensitivity and specificity test results for single eye blink signals	64
Table 3.9: Performance of peak detection models for double eye blink signals	64
Table 3.10: Sensitivity and specificity test results for double eye blink signals	65
Table 3.11: Performance of peak detection models for eye movement signals	65
Table 3.12: Sensitivity and specificity test results for eye movement signals	65
Table 3.13: Average ranking of Friedman's test with $p < 0.01$ for single and double blink signals.....	66
Table 3.14: Average ranking of Friedman's test with $p < 0.01$ for eye movement signals	67
Table 3.15: Pos-hoc analysis for Friedman's test for single and double eye blink signals	67
Table 3.16: Pos-hoc analysis for Friedman's test for eye movement signals.....	67
Table 3.17: Parameters setting of NNRW	71
Table 3.18: Average training accuracy with respect to the number of hidden neurons for single eye blink signals	72
Table 3.19: Average training accuracy with respect to the number of hidden neurons for double eye blink signals.....	73

Table 3.20: Average training accuracy with respect to the number of hidden neurons for eye movement signals	73
Table 3.21: Comparison of testing accuracy in terms of average values for three different signals over five peak models	76
Table 3.22: Performance comparison of average test results between the NNRW and PSOANN classifiers	77
Table 4.1: Classification accuracy results for NNRW classifier under different number of hidden neurons	83
Table 4.2: The average ranking of the Dumpala, Acir, Liu, and Dingle, achieved by Friedman	84
Table 4.3: Adjusted p-value for $N \times N$ comparisons of algorithms over 30 runs.....	84
Table 4.4: Parameters setting of AMSKF algorithm.....	93
Table 4.5: Best results over 30 runs using AMSKF feature selection algorithm	95
Table 4.6: Comparison of the classification accuracy between the existing peak models and AMSKF model	96
Table 4.7: The average ranking of the full feature set, Dumpala, Acir, Liu, Dingle, and AMSKF, achieved by Friedman.....	97
Table 4.8: Adjusted p-value for $N \times N$ comparisons of algorithms over 30 runs.....	98
Table 4.9: Best results over 30 runs using BSKF feature selection algorithm	104
Table 4.10: Comparison of the classification accuracy between the existing peak models and BSKF model	105
Table 4.11: The average ranking of the full feature set, Dumpala, Acir, Liu, Dingle, and BSKF, achieved by Friedman.....	107
Table 4.12: Adjusted p-value for $N \times N$ comparisons of algorithms over 30 runs for BSKF.....	107
Table 4.13: Best results over 30 runs using LocalDESKF feature selection algorithm	113
Table 4.14: Comparison of the classification accuracy between the existing peak models and LocalDESKF model	114

Table 4.15: The average ranking of the full feature set, Dumpala, Acir, Liu, Dingle, and LocalDESKF, achieved by Friedman.....	116
Table 4.16: Adjusted p-value for $N \times N$ comparisons of algorithms over 30 runs for LocalDESKF.....	116
Table 4.17: Best results over 30 runs using GlobalDESKF feature selection algorithm	122
Table 4.18: Comparison of the classification accuracy between the existing peak models and GlobalDESKF model	123
Table 4.19: The average ranking of the full feature set, Dumpala, Acir, Liu, Dingle, and GlobalDESKF, achieved by Friedman.....	125
Table 4.20: Adjusted p-value for $N \times N$ comparisons of algorithms over 30 runs for GlobalDESKF	125
Table 4.21: Performance comparisons among peak models.....	128
Table 5.1: Class distribution of the peak candidate sample and event.....	133
Table 5.2: Training and testing results of various number of hidden neuron on EEG Epileptic database.....	134
Table 5.3: Confusion matrix of epileptic and non-epileptic events classification	136
Table 5.4: Performance comparison of other methods	136
Table 5.5: Performance comparisons and ranking among peak models	139

LIST OF SYMBOLS AND ABBREVIATIONS

EEG	:	Electroencephalogram
PSO	:	Particle Swarm Optimization
ANN	:	Artificial Neural Network
SLFN	:	Single Layer Feedforward Neural Network
RVFL	:	Random Vector Functional Link
ANNPSO	:	PSO-based learning algorithm for ANN
SKF	:	Simulated Kalman Filter
AMSKF	:	Angle Modulated Simulated Kalman Filter
BSKF	:	Binary Simulated Kalman Filter
LocalDESKF	:	Local Optimum Distance Evaluated SKF
GlobalDESKF	:	Global Optimum Distance Evaluated SKF
NNRW	:	Neural Network with Random Weights
ECA	:	Gastric Electrical Activity
KEEL	:	Knowledge Extraction based on Evolutionary Learning
<i>Gmean</i>	:	Geometric Mean
PPG	:	Photoplethysmogram
ECG	:	Electrocardiogram
k-NEO	:	k-point nonlinear energy operator
P300	:	A neural evoked potential component of the EEG
QRS	:	The components of the ECG
RBFN	:	Radial Basis Function Network
MAC	:	Moving Average Curve
ACR	:	Applied Control and Robotics Laboratory
FWHM	:	Full Width with Half Maximum
BCI	:	Brain-Computer Interface
HMI	:	Human-Machine Interface

TP	:	True Peak
TN	:	True Non-peak
FP	:	False Peak
FN	:	False Non-peak
TNR	:	True Non-peak Rate
TPR	:	True Peak Rate
PP_i	:	The i th candidate peak point
$VP1_i$:	The associated valley point at first half wave
$VP2_i$:	The associated valley point at second half wave
$HP1_i$:	The half point at first half wave
$HP2_i$:	The half point at second half wave
$TP1_i$:	The turning point at first half wave
$TP2_i$:	The turning point at second half wave
$MAC(PP_i)$:	The moving average curve point
f_1	:	Peak-to-peak amplitude of the first half wave
f_2	:	Peak-to-peak amplitude of the second half wave
f_3	:	Turning point amplitude of the first half wave
f_4	:	Turning point amplitude at the second half wave
f_5	:	Moving average amplitude
f_6	:	Peak width
f_7	:	First half wave width
f_8	:	Second half wave width
f_9	:	Turning point width
f_{10}	:	First half wave turning point width
f_{11}	:	Second half wave turning point width
f_{12}	:	FWHM
f_{13}	:	Peak slope at the first half wave

f_{14}	:	Peak slope at the second half wave
f_{15}	:	Turning point slope at the first half wave
f_{16}	:	Turning point slope at the second half wave
w_{ij}	:	The input weights of ANN located between i th and j th neurons
w_{jk}	:	The output weights of ANN located between j th and k th neurons
n	:	The total number of input neurons of ANN
m	:	The total number of output neurons of ANN
l	:	The total number of hidden neurons of ANN
y	:	The binary output for ANN classifier
y_k	:	The yield of k th output neuron
y_j	:	The yield of j th hidden neuron
θ	:	Decision threshold of ANN
\tanh	:	The hyperbolic tangent function
net_k	:	The k th output neuron
net_j	:	The j th hidden neuron
ω	:	Inertia weight of PSO
c_1	:	The cognitive coefficient of PSO
c_2	:	The social coefficient of PSO
r_1	:	Random number of PSO
r_2	:	Random number of PSO
k	:	Number of iteration of PSO
i	:	Number of particle of PSO
$pbest$:	Personal best of PSO
$gbest$:	Global best of PSO
v_i^k	:	Current velocity of i th particle in PSO

v_i^{k+1}	:	Updated velocity of i th particle in PSO
s_i^k	:	Current position of particle in PSO
s_i^{k+1}	:	Updated position of i -th particle in PSO
ω_{\max}	:	The maximum values of inertia weight for PSO
ω_{\min}	:	The minimum values of inertia weight for PSO
k_{\max}	:	The maximum iteration of PSO
pb_i^k	:	The $pbest$ of particle i for PSO
gb^k	:	The $gbest$ of particle i for PSO
d	:	The dimension of input weights for PSO
D	:	The dimension of output weights for PSO
nl	:	The total number of input weights for ANN
lm	:	The total number of output weights for ANN
e	:	The e th dimension of decision threshold for PSO
STDEV	:	Standard Deviation
x	:	The given unknown sample
X_{best}	:	The best fitness value among agents at every iteration for SKF
X_{true}	:	The best solution compared to the current X_{best} for SKF
t	:	The number of iteration for SKF
t_{\max}	:	The maximum number of iteration for SKF
P	:	The value of error covariance estimate for SKF
$P(0)$:	The initial value of error covariance estimate for SKF
Q	:	The process noise value for SKF
R	:	The measurement noise value for SKF
T	:	The sampling time
$X_i(t)$:	The next state estimate for SKF

$X_i(t-1)$:	The previous state for SKF
$X_i(t t-1)$:	The transition state for SKF
$P(t t-1)$:	The previous error covariant estimate for SKF
$P(t-1)$:	The transition error covariant estimate for SKF
$Z_i(t)$:	The measurement value for SKF
$K(t)$:	The Kalman gain for SKF
$rand$:	The random value for SKF
$g(x)$:	The continuous signals for AMSKF
Δ_i	:	The multiplication equation for BSKF
\sum_i^d	:	Generated random bit string for BSKF
$X_{\text{best-so-far}}^d$:	The best-so-far solution for LocalDESKF and GlobalDESKF

CHAPTER 1: INTRODUCTION

1.1 Research Background

Electroencephalogram (EEG) is a well known type of non-invasive recorded brain signals. The EEG signals are acquired using electrodes which are placed using the 10-20 international electrode placement system (Klem et al, 1999). The EEG signals contain an electrical activity arising from the brain response to the cerebral activities such as mental task, event-related desynchronization, evoke potential, and slow cortical potential. Other human activities also can be recorded using EEG through brain response from horizontal eye movements, eye blinks, head movements, left and right hand movements, left and right leg movements and finger movements.

EEG signals have become a growing interest in research that are widely used for various real applications such as brain-computer interface (BCI) (Nicolas-Alonso & Gomez-Gil, 2012), human-machine interface (HMI) (Ramli et al, 2015), diagnosing and monitoring epilepsy (Acir, 2005), diagnosing stroke patients (Zappasodi et al, 2014), tracking eye gaze (Adam, Shapiai, et al, 2014), and continuous monitoring critically ill patients in coma (Claassen et al, 2013). The HMI, BCI, and tracking eye gaze applications are specifically developed to assist physical impaired people on verbal communication and controlling devices. Nowadays, the utilization of an advanced processing method makes the EEG signals has efficiently been used for those particular applications. However, there are still many researches required to be done for the further development process.

1.2 The Significance of Peak Detection Algorithm in Event-related Signals Classification

The utilization of peak classification algorithm has become the most significant approach in several physiological signals applications such as the detection of epileptic EEG signals (Acir et al, 2005; Y. C. Liu et al, 2013), the detection of P300 response in the EEG signals (N. Xu et al, 2004), photoplethysmogram (PPG) monitoring (M. Elgendi et al, 2013), electrocardiogram (ECG) monitoring (Mohamed Elgendi et al, 2016; Kim & Shin, 2016; Tafreshi et al, 2014), the analysis of gastric electrical activity (ECA) (Dumpala et al, 1982), and the detection of eye gaze direction applications (Adam, Shapiai, Mohd Tumari, Mohamad, & Mubin, 2014). In those applications, peak classification algorithm is typically located in the first step of a classification process. For example, in epilepsy detection application, epilepsy may occur when recurrent peaks are detected in the EEG recording during a given time interval by any immediately identifiable cause. A similar approach is used for the detection of horizontal eye gaze direction application. Once one true peak is identified, a subject may have shift once to the left or right direction. P300 response also triggers a peak in the EEG signals. P300 is a brain response measured by electrodes covering the parietal lobe in the presence of visual and auditory stimuli. Also in PPG signals monitoring, peak classification algorithm is employed for the analysis of heart rate variability in evaluating vascular effects. For the ECG signals, peak detection is typically used to detect a QRS complex. The QRS complex is a peak model for ECG signals including Q-peak point, R-peak point, and S-peak point. Another important peak points in ECG signals are P-peak point and T-peak point. The detection of the QRS complex is a critical part in numerous ECG signal processing system. Note that, the peak classification is just a first step in detecting an event for the variation of signals such as PPG, EEG, ECG, and ECA. The main goal of event detection is to determine the event,

not the whole peaks. Therefore, the classification performances of the highlighted applications are not the performance of interest of peak detection research.

1.3 The Definition of a Peak in the EEG Signals

In general, a peak point in a signal holds the highest value located at a specific time and location. A peak point can exist in the signals as the response of brain on human activities or noise. Some examples of the response of brain on human activities that triggers a peak in the signals are epilepsy, eye blink, and the horizontal and vertical eye movements. Some researchers focused on research to define the characteristics of a peak in the EEG signals. For example, in the epileptic EEG signals point of view, Gloor (1975) has defined a peak as follows: (1) a restricted triangular transient clearly distinguishable from background activity, (2) having an amplitude of at least twice that of the preceding five seconds of background activity in any channel of EEG signals, and (3) a peak signals have a duration of lower and equal than 200ms. From the response of eye blink and eye movements in the EEG signals, Iwasaki et al (2005) have pointed out that the amplitude of peak points are different from one subject to another and it can vary from $600\mu\text{V}$ to $1100\mu\text{V}$. Another research work by Sovierzoski et al (2008), have analyzed the electrical behavior of EEG eye blink events. The research work has recorded the minimum, maximum, and the average of the peak amplitude. The minimum value of amplitude was $55\mu\text{V}$. The maximum value of amplitude was $533\mu\text{V}$. The average of peak amplitude was $170\mu\text{V}$. These findings showed that the peak amplitude can vary from $55\mu\text{V}$ up to $533\mu\text{V}$ and it depends on subjects. Sometimes, the amplitude is higher than usual due to various noises. From the various definition of a peak in EEG signals, it can be understood that a peak definition is not similar to different events. Also, different subjects often do not produce the same peaks. As such, this kind of knowledge has to be considered in the further research works.

1.4 Motivation and Problem Statements

Prominently, one of the actively undergoing research works in EEG signals is the classification of peaks in the time domain analysis. The classification of desired peak points becomes a challenging problem to automatically recognized due to a peak in the signals have been contaminated with various noises, the nature of non-stationary EEG signals, and its relative to the baseline amplitude, time, and different users. In the time domain analysis, previously, there are four different peak models with associated peak features, namely Dumpala, Acir, Liu, and, Dingle models that have been used as inputs for classification in peak detection algorithm. Three peak models, which are Acir, Liu, and Dingle, have been successfully employed in EEG epilepsy application while Dumpala peak model has successfully been used for Electric Control Activity (ECA) application (Acir, Oztura, Kuntalp, Baklan, & Guzelis, 2005; Dingle et al, 1993; Dumpala, Reddy, & Sarna, 1982; H. S. Liu et al, 2002). The utilization of a particular peak model is to detect true peaks in a particular application. This method has proved to achieve good detection performance especially for epilepsy application. As pointed out by Dumpala, Reddy, and Sarna (1982), Acir, Oztura, Kuntalp, Baklan, and Guzelis (2005), H. S. Liu, Zhang, and Yang (2002), and Dingle, Jones, Carroll, and Fright (1993), their detection system with the selected peak model and subsequent processes contributed to the good detection performance. Although good detection performance has been achieved in (Acir, Oztura, Kuntalp, Baklan, & Guzelis, 2005; Dingle, Jones, Carroll, & Fright, 1993; Dumpala, Reddy, & Sarna, 1982; H. S. Liu, Zhang, & Yang, 2002), yet these approaches use various types of peak detection algorithms on different peak models. The defined models, however, cannot assure will give the best detection performance for other EEG signals applications. Therefore, there are needs for further research on the evaluation of detection performance using the combination of different

types of peak models in a common and unbiased classification approach for a fair evaluation in finding the best model.

The classification approaches that have frequently been used for EEG signals peak classification algorithm consist of rule-based (Adam et al, 2015; Dingle, Jones, Carroll, & Fright, 1993; Dumpala, Reddy, & Sarna, 1982), AdaBoost (H. S. Liu, Zhang, & Yang, 2002), radial basis function network (Acir, Oztura, Kuntalp, Baklan, & Guzelis, 2005), support vector machine (Y. C. Liu, Lin, Tsai, & Sun, 2013), radial basis support vector machine (Acir, Oztura, Kuntalp, Baklan, & Guzelis, 2005), artificial neural network (ANN) (H. S. Liu, Zhang, & Yang, 2002), and expert system (Dingle, Jones, Carroll, & Fright, 1993; H. S. Liu, Zhang, & Yang, 2002). The best peak model can be determined by using those particular classification methods. However, the classification methods cannot provide the fast learning speed once it integrates with metaheuristic-based feature selection techniques for selecting the best peak model. The classification method that provides the fast learning speed and learning without iterative tuning becomes more superiority to be used in this study. The common classification approach that follows those criteria is a neural network with random weights (NNRW) classifier (Schmidt, 1992). Generally in NNRW, the input weights (the weights that are located between input and hidden layer) and biases at hidden layer are randomly chosen. The output weights (the weights that are situated between hidden and output layer) can be determined by using Moore-Penrose generalized inverse. This kind of approach has been proven to achieve a good and generalized classification performance in different applications (Cao et al, 2015; J. Lu et al, 2014). Therefore, this study aims to produce a good and generalized performance for EEG signal peak classification application using the advantageous of the NNRW approach.

From the preliminary experimental results on evaluation of the existing peak models into the individual NNRW classifier, none of the existing models perform adequately in all event-related EEG signals peak classification (e.g., single eye blink, double eye blink, and eye movement signals). For example, Liu model only performed on single eye blink and double eye blink signals but the classification performance dropped when evaluating to eye movement signals. Recently, to properly determine the best and generalized peak model in all event-related EEG signals are still open problems for further research. Through this study in finding the best and generalized peak models, new approaches are proposed based on four novel optimization algorithms, namely as angle modulated simulated Kalman filter (AMSKF), binary simulated Kalman filter (BSKF), local optimum distance evaluated simulated Kalman filter (LocalDESKF), and global optimum distance evaluated simulated Kalman filter (GlobalDESKF). In the conducted experiment, 11781 samples of peak candidate are employed in this study for the validation purpose. The samples are collected from the three different event-related EEG signals.

1.5 Objectives of the Study

Without precisely determining the peak model from the set of features, the best classification performance only depends on the design of subsequent processes, for example, classification process. To find the best and generalized peak model, therefore, five main directions are highlighted in this study:

- i. To fairly evaluate the performance of the four different peak models including Dumpala, Acir, Liu, and Dingle models and then, suggest the best peak model using individual ANNPSO and NNRW classifiers.

- ii. To introduce four new feature selection methods using AMSKF, BSKF, LocalDESKF, and GlobalDESKF algorithms for achieving the highest performance of peak detection of EEG signals in the time domain approach.
- iii. To introduce four new generalized peak models generated by the proposed AMSKF, BSKF, LocalDESKF, and GlobalDESKF feature selection algorithms.
- iv. To compare the classification performance of the existing peak models and the new generalized models.
- v. To apply the four proposed models to epileptic EEG events classification.

1.6 Main Contributions

Preliminary, three main contributions of this study are determined. The three main contributions are expressed in the following subsection.

1.6.1 First Contribution

At first, the best peak model for three different EEG signals (single eye blink, double eye blink, and eye movement signals) have been identified using ANN classifier with particle swarm optimization (PSO) as learning algorithm. It has been observed that the best test performance, in average, is 91.94%, 87.47%, and 87.6% for single eye blink, double eye blink, and eye movement signals, respectively. These results indicate that the Acir and Dingle peak models offer high accuracy of peak detection as compared to other models for the two eye blink signals and eye movement signals, respectively. The result of statistical analysis indicates that the Acir peak model is better than Dingle and Dumpala peak models for single and double eye blink signals. Moreover, the result of statistical analysis for eye movement signals proves that the Dingle peak model is better than Dumpala, Acir, and Liu models.

1.6.2 Second Contribution

Secondly, a fair evaluation of the detection performance of the four different peak models and full feature set using NNRW classifier has been accomplished. In this study, the NNRW is firstly employed into peak detection algorithm of EEG signals for classification. Based on the final experimental results, it was found that the Dingle model gave the best performance, with 72% accuracy in the analysis of real EEG data. Statistical analysis conferred that the Dingle model afforded significantly better mean testing accuracy than did the Acir and Liu models, which were in the range 37-52%. Meanwhile, the Dingle model has no significant difference compared to Dumpala model. This study also observes that defining more peak features does not guarantee in producing better accuracy on EEG-based horizontal eye movement signal application. Based on the experimental results, the mean of testing accuracy only can achieve at 36.9%. However, determining the optimal model from the selected features associated with the advantageous of common classification platform is the best approach to gain the accuracy of detection performance.

1.6.3 Third Contribution

The third contribution of this study are expressed as follows: (1) to employ a recently introduced population-based metaheuristic optimization algorithm for feature selection in EEG signals peak classification using AMSKF, BSKF, LocalDESKF, and GlobalDESKF, (2) to firstly employ the NNRW into peak detection algorithm for classification and feature selection, and (3) to propose new generalized peak models for EEG signals peak classification based on the features selected by AMSKF, BSKF, LocalDESKF, and GlobalDESKF. For the benchmarking purpose, four different existing peak models and full features set are considered in this study. The experimental results indicate that the four new combinations of peak features that are produced by the

proposed AMSKF, BSKF, LocalDESKF, and GlobalDESKF techniques perform better accuracy compared to the NNRW with the conventional peak models. The four proposed peak models are then used for classification of epileptic EEG events application.

1.7 Outline of Thesis

In following five chapters, the development of a feature selection approach to produce a generalized model for event-related EEG signals peak classification is presented. Chapter 2 reviewed the research works related to peak detection algorithm in event-related EEG signals classification that have been published so far. The review was focused on three main aspects: (1) a review of the existing algorithm for peak classification, (2) a review of the existing peak models in time domain approach, and (3) a review of the existing feature selection methods in the EEG signals peak detection algorithm.

Chapter 3 detailed the step by step approaches of EEG signals peak detection algorithm. Two famous ANN classifiers were discussed in this chapter which is ANNPSO and NNRW. The associated features of the existing peak models are considered as inputs to the ANNPSO and NNRW. Two experiments were conducted to both classifiers in order to evaluate the classification performance of the existing models. Discussion on the experimental results obtained to find an appropriate classification approach for the algorithm and at the meantime determine the best model among the existing models.

Chapter 4 describes the proposed generalized models, which are produced by the proposed AMSKF, BSKF, LocalDESKF, and GlobalDESKF feature selection algorithms. The descriptions in Chapter 4 also include the basic process flow of the

variant of the SKF algorithms and the implementation of the algorithms as a feature selector for peak classification. The experimental setups are also presented to demonstrate the selection and classification processes of peaks in EEG signals. Experimental results and discussions indicating the performance of these feature selection techniques are also shown. To measure the effectiveness of the proposed models, the performance of the proposed models and the existing models are discussed.

Chapter 5 presents the application of the proposed work on EEG epileptic classification. Finally, the conclusions and suggestions for the future research are described in Chapter 6.

CHAPTER 2: LITERATURE REVIEW

2.1 Introduction

The literature review consists of three groups which are (1) a review of the existing algorithms for peak classification, (2) a review of the existing peak models in time domain approach, and (3) a review of the existing feature selection methods in the EEG signals peak detection algorithm.

2.2 The Existing Algorithms for Peak Classification

To date, variety approaches of peak classification algorithms have been introduced. The algorithms can be categorized into four main approaches based on time domain (Acir & Guzelis, 2004; Acir, Oztura, Kuntalp, Baklan, & Guzelis, 2005; Barea et al, 2012; Dingle, Jones, Carroll, & Fright, 1993; Dumpala, Reddy, & Sarna, 1982; H. S. Liu, Zhang, & Yang, 2002; W. Lu et al, 2006; Manikandan & Soman, 2012; L. Xu et al, 2008), frequency domain (Juozapavi et al, 2011), time-frequency domain (H. S. Liu, Zhang, & Yang, 2002; Senhadji & Wendling, 2002), and nonlinear (Putignano et al, 2012). In time domain approach, the peaks are analyzed against time. In frequency domain approach, the peaks are analyzed against frequency. In time-frequency domain approach, the peaks are analyzed in both time and frequency domain. In nonlinear approach, some statistical parameters of the peaks are analyzed.

The general framework of peak classification algorithm usually involves several processes which are signal pre-processing, peak candidate detection, feature extraction, and classification. Various signal pre-processing methods have been employed such as data compression (Bonner et al, 1972), wavelet transform (Indiradevi et al, 2008), Kalman filter (Oikonomou et al, 2007), and Hilbert transform (Manikandan & Soman, 2012). Two methods for peak candidate identification have been used which are three

points sliding window method (Dumpala, Reddy, & Sarna, 1982) and k-point nonlinear energy operator (k-NEO) method (Y. C. Liu, Lin, Tsai, & Sun, 2013). Various feature extraction techniques have been proposed which are model-based (Y. C. Liu, Lin, Tsai, & Sun, 2013), wavelet analysis (Sinno & Tout, 2008), template matching (Ji et al, 2011), and power spectra analysis (Exarchos et al, 2006). The classification approaches that have frequently been used for EEG signals peak detection consist of rule-based (Adam, Ibrahim, Mokhtar, Shapiai, & Mubin, 2015; Dingle, Jones, Carroll, & Fright, 1993; Dumpala, Reddy, & Sarna, 1982), AdaBoost (H. S. Liu, Zhang, & Yang, 2002), radial basis function network (Acir, Oztura, Kuntalp, Baklan, & Guzelis, 2005), support vector machine (Y. C. Liu, Lin, Tsai, & Sun, 2013), radial basis support vector machine (Acir, Oztura, Kuntalp, Baklan, & Guzelis, 2005), artificial neural network (H. S. Liu, Zhang, & Yang, 2002), and expert system (Dingle, Jones, Carroll, & Fright, 1993; H. S. Liu, Zhang, & Yang, 2002).

In this study, several peak classification algorithms in the time domain analysis are highlighted. Dumpala, Reddy, and Sarna (1982) have introduced the utilization of three points sliding window and threshold-based classification. The theory of maxima and minima using three-point sliding window approach has been applied to detect a candidate peak. Two flowcharts of peak detection have been proposed. A predicted peak can be identified if the feature values are satisfied the decision threshold values. The authors claimed that the proposed peak classification algorithm can be used to other biological signals.

Dingle, Jones, Carroll, and Fright (1993), use two-threshold systems to detect a candidate peak. An expert system which considered both spatial and temporal contextual information has been used to reject the artifacts and classify the transient events.

Wavelet transform has been used to decompose the EEG signals by H. S. Liu, Zhang, and Yang (2002). Based on the decomposed signals, seven peak features are calculated. These features are used as the input of ANN classifier. An expert system which considered both spatial and temporal contextual information has been used to reject the artifact. Several heuristic rules have been employed to distinguish the type of artifact. After all artifacts are recognized and rejected, the decision will be made to classify the epileptic events.

Acir, Oztura, Kuntalp, Baklan, and Guzelis (2005), have introduced a three stages procedure based on ANN for the detection of epileptic peaks. The EEG signal is transformed into the time-derivative signals. Several rules have been used to detect a peak candidate. The features of peak candidate are calculated and fed into two discrete perceptron classifiers to classify into three groups: definite peak, definite non-peak, and possible/possible non-peak. The peak that belongs in the third group is going further process by nonlinear classifier. Different peak detection algorithm based on a modified radial basis function network (RBFN) and discrete perceptron classifiers has also been invented by Acir (2005) for the detection of epileptic spikes. k-NEO method has also been used by Y. C. Liu, Lin, Tsai, and Sun (2013) to detect a candidate peak. The peak features are calculated and then used as the input of the AdaBoost classifier.

2.3 The Existing Peak Models based on the Time Domain Analysis

The first conventional peak model in the time domain analysis has been introduced in Dumpala's peak detection research (Dumpala, Reddy, & Sarna, 1982). The defined peak model comprises of four features, which are (1) the amplitude of the magnitude of peak point and the magnitude of valley point at the first half wave, (2) the width between valley point of first half point and valley point at second half wave, (3) and (4) two slopes between a peak point and valley point in the first half wave and second half

wave. A peak point is a point that holds the maximum value located at a specific time and location on the signals. However, a valley point is a vice versa a peak point. A similar definition of the peak amplitude and slopes are also been used in (Acir & Guzelis, 2004; Acir, Oztura, Kuntalp, Baklan, & Guzelis, 2005; L. Xu, Meng, Liu, & Wang, 2008).

An additional feature of peak amplitude and two features of the peak width have been introduced by Acir, Oztura, Kuntalp, Baklan, and Guzelis (2005) to detect a peak of EEG epileptic signals. The additional peak amplitude is the amplitude of the magnitude of peak point and the magnitude of valley point of the second half wave. The peak widths are the width between peak point and valley point of first half wave and second half wave. The total features that are introduced by are six features. Acir, Oztura, Kuntalp, Baklan, and Guzelis (2005) did not use the width feature that was introduced by Dumpala, Reddy, and Sarna (1982). A similar definition of the peak amplitudes, widths, and slopes are also been used in (Y. C. Liu, Lin, Tsai, & Sun, 2013). In (Y. C. Liu, Lin, Tsai, & Sun, 2013), an additional peak feature is added to a set of feature that is introduced in (Acir & Guzelis, 2004; Acir, Oztura, Kuntalp, Baklan, & Guzelis, 2005), which is the area of the peak. However, the definition of area integration is not presented in the paper.

Also, H. S. Liu, Zhang, and Yang (2002) have introduced 11 peak features. The peak model consists of four amplitudes; (1) the amplitude of the magnitude of peak point and the magnitude of valley point at the first half wave, (2) the amplitude of the magnitude of peak point and the magnitude of valley point of the second half wave, (3) the amplitude of the magnitude of peak and the magnitude of turning point at the first half wave, and (4) the amplitude of the magnitude of peak and the magnitude of turning point at the second half wave. The turning point is defined as the point where the slope

decreases more than 50% as compared to the slope of the preceding point. The model also consists of three widths; (1) the width between valley point at first half point and valley point at second half wave, (2) the width between turning point at first half wave and turning point at second half wave, and (3) the width between half point at first half wave and half point at second half wave. Four slopes are also measured; (1) and (2) two slopes between a peak point and valley point in the first half wave and second half wave, (3) and (4) two slopes between peak point and turning point at first half wave and second half wave.

Another peak model consists of four features, which has been introduced by Dingle, Jones, Carroll, and Fright (1993). The peak amplitude is the difference between the peak point and the floating mean. The floating mean is the average EEG that is centered at the peak point that is also called moving average curve (MAC) (W. Lu, et al., 2006). The width is calculated based on the difference between the valley point at the first half wave and the valley point at the second half wave. The two slopes are the slopes between a peak point and valley point in the first half wave and second half wave. Recently, M. Elgendi, Norton, Brearley, Abbott, and Schuurmans (2013) also used MAC in his study to detect systolic peak for heart rate analysis.

Based on the literature study of peak detection, almost all researchers focus on the problem of an epileptic EEG signal. A review of peak detection algorithms that is employed to the epileptic EEG signal is presented by Wilson and Emerson (2002) and Webber and Lesser (2017). Details of the different peak detection algorithms on different peak models are tabulated in Table 2.1. Note that, the detection performances of the highlighted applications are not the performance of interest of peak detection research. The detection performance is just to show that the utilization of peak detection algorithm in the events classification has provided the best performance.

Table 2.1: Summary of the previous research works using various types of peak detection algorithms on different peak models for various applications

Peak Model	Input signals	Event	Classification method	Accuracy test of event (%)
M. Elgendi, Norton, Brearley, Abbott, and Schuurmans (2013)	PPG	Heart rate analysis	Thresholds, Rule-based	Sensitivity: 99.89 Selectivity: 99.84
Y. C. Liu, Lin, Tsai, and Sun (2013)	EEG	Epilepsy	AdaBoost	93.5
Acir (2005)	EEG	Epilepsy	Radial basis function network (RBFN)	Sensitivity: 91.1 Selectivity: 89.2
Acir, Oztura, Kuntalp, Baklan, and Guzelis (2005)	EEG	Epilepsy	Radial basis support vector machine (RB-SVM)	Sensitivity: 89.1 Selectivity: 85.9
H. S. Liu, Zhang, and Yang (2002)	EEG	Epilepsy	ANN, Expert system	90
Dingle, Jones, Carroll, and Fright (1993)	EEG	Epilepsy	Thresholds, Rule-based, Expert system	80
Dumpala, Reddy, and Sarna (1982)	ECA	Gastric activity	Thresholds, Rule-based	100

2.4 The Existing Feature Selection Methods using Optimization Algorithms for EEG Signals Peak Classification

One approach for improving the peak classification performance is to identify the best combination of peak features. Previously, several authors have defined a variant of peak models based on the characteristic of the peak of EEG signals in the time domain analysis. In one peak of EEG signals, there are several signal parameters including different amplitudes, widths, and slopes. A variety of peak features can be calculated based on those signal parameters. For instance, the peak-to-peak amplitude of the first and second half waves, peak width, ascending peak slopes at the first half wave, and descending peak slope at the second half wave. All these features are used as inputs to the classification process to differentiate between the peak and non-peak of the signals. To the best of our knowledge, there are very reports a few studies have used feature selection method to find the best peak model for EEG signals peak detection

application. Two methods that have been found in the literature are particle swarm optimization (Adam, Shapiai, Mohd Tumari, Mohamad, & Mubin, 2014) and gravitational search algorithm (Adam, Ibrahim, et al, 2014). Both of the methods use the same classification approach which is a rule-based classifier. There are a limitation has been pointed out, which the classifier tend to have poor performance with peak models defining many peak features. The classification performance declined to nil when the classifier employed all 11 features from Liu model (Adam, Shapiai, Mohd Tumari, Mohamad, & Mubin, 2014).

An adequate solution was achieved in a shorter time by utilizing population-based metaheuristic optimization algorithms. Many complicated real-world issues can be ironed out by using these algorithms. These algorithms can also be practiced to solve almost any optimization problems (Xiong et al, 2015). There are a variety of population-based metaheuristic optimization algorithms which have been created such as genetic algorithm (Hooker, 1995), simulated annealing (Johnson et al, 1989), particle swarm optimization (James Kennedy & Russell Eberhart, 1995), ant colony optimization (Dorigo et al, 1996), big bang-big crunch optimization (Erol & Eksin, 2006), intelligent water drops algorithm (Shah-Hosseini, 2007), honey bee mating optimization (Marinakis et al, 2011), firefly algorithm (X. S. Yang, 2010), gravitational search algorithm (Rashedi et al, 2009), harmonic search optimization (Yang, 2009), bat algorithm (X.-S. Yang, 2010), and black hole algorithm (Hatamlou, 2013). Thus far, these optimization algorithms have been widely applied in fields such as power system (Ahila et al, 2015), manufacturing (Zhang et al, 2015), and medical (Adam, Shapiai, Mohd Tumari, Mohamad, & Mubin, 2014; Bababdani & Mousavi, 2013) as a practical technique for feature selection.

A new metaheuristic optimization algorithm was recently introduced by Ibrahim *et al.* (Zuwairie Ibrahim et al, 2015) and this algorithm was inspired by the state estimation process of Kalman filter. The new optimizer is entitled simulated Kalman filter (SKF) algorithm. There are three main processes in the principle of Kalman filter which are states prediction, state measurement, and state estimation. Each agent acts as an individual Kalman filter and holds a vector state in the SKF algorithm. New states are predicted and new locations of agents are revised from the prediction, measurement, and estimation state processes. The processes are iteratively looped until the maximum iteration is achieved. The SKF algorithm has the capability to find the most optimal solution efficiently while the performance is comparable to gravitational search algorithm and black hole algorithm for unimodal optimization problems based on the final experimental results in (Zuwairie Ibrahim, et al., 2015). The original SKF algorithm, however, cannot be used for solving discrete optimization problems. In order to eradicate this problem, various binary-based SKF algorithms were introduced such as Angle modulated SKF (AMSKF) (Md Yusof, Ibrahim, et al, 2016a), Binary SKF (BSKF) (Md Yusof et al, 2015), Local Optimum Distance evaluated SKF (LocalDESKF) (Md Yusof, Ibrahim, Ibrahim, Abd Aziz, et al, 2016), and Global Distance Evaluated SKF (GlobalDESKF) (Md Yusof, Ibrahim, et al, 2016b) algorithms. Based on the capability of the Binary-based SKF algorithms, they have potential to be developed as a feature selection method.

2.5 Summary

The existing peak classification algorithms have all been used successfully in various applications. However, almost no comparisons of these algorithms have been performed so far. For that reason, it appears as difficult to choose which one the best. To address this difficulty, the similarity of these algorithms in the time domain analysis point of

view are observed. As presented in this chapter, a group of researchers have used a different style of frameworks but similar processes, for example, there are a variety of methods in signal pre-processing, peak candidate identification, feature extraction, and classification. This is the reason why this study focused on the similar processes in this thesis work.

The literature review also showed that every existing algorithm employed a different peak model in specific event-related EEG signals. The selection of these peak models with the associated features are based on the characteristics of the EEG signals. Consequently, a good performance of peak classification is obtained in the past work. However, the utilization of the existing peak models are not guaranteed to achieve higher performance in other event-related EEG signals peak classification. Moreover, to the best of our knowledge, none of the techniques based on experimental exploration to find the best model have been performed so far. This motivated my research explores a good experimental technique that can produce the best and generalized peak model for any event-related EEG signals peak classification. The best approach so far is feature selection.

CHAPTER 3: PEAK CLASSIFICATION USING THE EXISTING MODELS

3.1 Introduction

The EEG signals peak classification algorithm is typically implemented in the first step of the signal classification process. That means the desired peaks have to be firstly identified, then, it translate to a specific activity such as epileptiform and eye gaze direction activities. For example, in an important application for clinical neurology, epileptiform activity in the cerebral cortex is identified from recurrent spikes in the electroencephalogram (EEG) recording during a given time interval. A similar approach is used in procedures for detecting horizontal eye gaze direction, which has applications for brain-machine interfacing. Furthermore, in the case of PPG signal monitoring, peak detection algorithms serve to measure heart rate variability, which can be predictive of risk for heart disease (Shaffer et al, 2014). In all these applications, it is essential to have peak classification algorithms with high performance.

In this study, a generalized peak classification algorithm that suits for any peak event-related EEG signals has been developed. The procedure for the collection of EEG data will present at the beginning of this chapter. This chapter will also introduce the step by step processes of the peak detection algorithm. The processes consist of four main stages including peak candidate identification, feature extraction, peak model selection, and classification. For peak candidate identification, eight location points of a peak candidate are recognized. The information of the eight points will be used in feature extraction process in which will produce 16 time domain peak features. Then, the selection of features is guided based on the existing models i.e., Dumpala, Dingle, Acir, Liu, and full feature set. For classification, two famous classification approaches are introduced into peak classification algorithm, which are ANNPSO and NNRW

classifiers. The flowchart of the proposed peak classification process is shown in Figure 3.1.

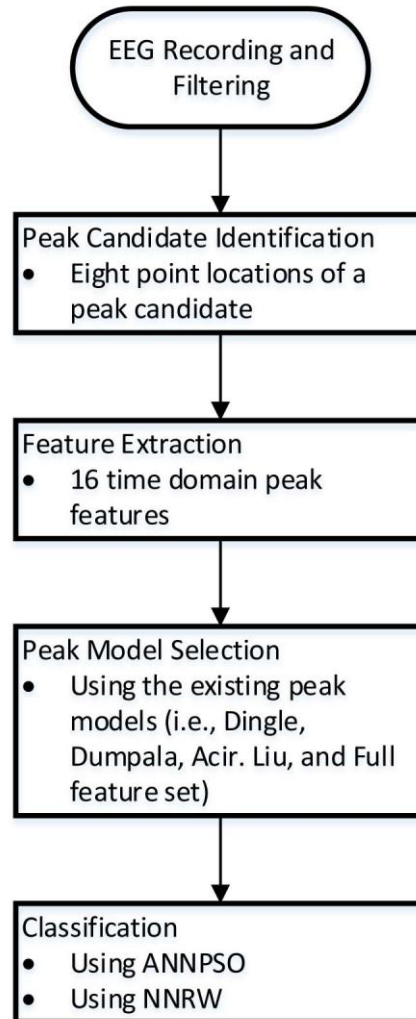


Figure 3.1: Flowchart of peak classification using the selection of the existing models

3.2 Experimental Protocols and Data Collection

The EEG signals in this study were obtained in the Applied Control and Robotic (ACR) Laboratory, Department of Electrical Engineering, Faculty of Engineering, University of Malaya, Malaysia. Thirty healthy subjects were involved voluntarily in these data collection sessions which were undergraduate and postgraduate students in the Faculty of Engineering. The experimental protocol was approved by the medical ethics committee of the University of Malaya Medical Centre (see Appendix A). All subjects are informed to sign a consent form in advance.

This study involves three different cases of the EEG signals. The first case is labeled as single eye blink signals. The second case is labeled as double eye blink signals. The third case is labeled as eye movement signals. The first and second cases of EEG signals recording were conducted using the g.USBamp biological signals acquisition system. While, the third case of EEG signals recording were conducted using the g.MOBILab portable biological signals acquisition system. The scalp electrodes arrangement of the three different signals is placed using the 10-20 international electrode placement system (Klem, Luders, Jasper, & Elger, 1999). The sampling frequency for those signals was set to 256 Hz.

In the data collection session as shown in Figure 3.2, the subjects were told to prepare for the external voice cue within up to 4 seconds. Appearance of the cue is voice command or verbal reminder for the subject to move his eyes initially forward fixation to the left, right, single eye blink, or double eye blink. At exactly 5 seconds from the beginning session, the external voice cue appears instructing the subject to follow the command.



Figure 3.2: Data collection session

The single blink and double blink signals were recorded from F9 channel. The reference electrode was located on the ear. The ground electrode was located on channel AFz. In total, only three electrodes were used. The electrodes from the F9 channels are positioned for detecting EEG peaks associated with the brain response of commanded single and double eye blink. Single means the eye are blinking once while double means the eye are blinking twice.

For the data collection of single eye blink signals, the commands will appear one by one in the duration of 10 seconds. The sequences of the commands are single eye blink, shift gaze to the right direction, shift gaze the forward direction, single eye blink, shift gaze to the right direction, shift gaze to the forward direction. For the data collection of double eye blink signals, the commands will appear one by one in the duration of 80 seconds. The sequences of the commands are shift gaze to the right direction, shift gaze

to the forward direction, double eye blink, shift gaze to the right direction, shift gaze to the forward direction, shift gaze to the left direction, shift gaze to the forward direction, double eye blink, shift gaze to the left direction, shift gaze to the forward direction, double eye blink, shift gaze to the left direction, shift gaze to the forward direction, shift gaze to the right direction, shift gaze to the forward direction, and double eye blink. The eyes blink that produces some peaks in the signals on channel F9 is archived as raw data for analysis.

The eye movement signals were recorded from C3 and C4 channels. The channel CZ was used as a reference. The ground electrode was located on FPz channel. In total, only four electrodes were used. The electrodes from the C3 and C4 channels are positioned for detecting EEG peaks associated with the brain response of commanded horizontal eye gaze direction. For the data collection of eye movement signals, the subjects have only to follow the command to shift gaze to the left or right direction, and hold the new eye position from 5 until 10 s, which is the end of the EEG recording. The eye gaze directions that produce some peaks in the signals on channels C3 and C4 are archived as raw data for analysis.

Figure 3.3 shows three different filtered EEG signals that were named as a single eye blink, double eye blink, and eye movement signals. The dotted red vertical lines show the actual peak point location, as manually assigned by a researcher. The descriptions of those EEG signals are tabulated in Table 3.1.

The single eye blink signals have 30 signals, 10-second length per signal, 2560 sampling points per signal, and each signal containing two known peak points and various additional signal patterns. In total, this study was collecting 76800 sampling points' data. The additional signal patterns are the edge transitions which represent the

eye movements. The known peak pattern in this signal represents a single eye blink. The peak pattern of a single eye blink is useful as an additional feature for controlling an electric wheelchair (Lin & Yang, 2012). From the total sampling points (76800), 3238 sampling point locations are identified as the locations of peak candidates, 60 sampling point locations are identified as the locations of true peaks, and 3178 sampling point locations are identified as the locations of false peaks.

The double eye blink signals have five signals, 80-second length per signal, 20480 sampling points per signal, and each signal containing eight known peak points and some additional signal patterns. The total sampling points that were collecting are 102400. The additional signal patterns are the edge transitions that represent the horizontal eye movements. The signals occasionally contain a peak of the single eye blink. The peak pattern of the double eye blink is also useful as an additional feature for controlling a wheelchair (Ahmed, 2011). From the total sampling points (102400), 4662 sampling point locations are identified as the locations of peak candidates, 40 sampling point locations are identified as the locations of true peaks, and 4622 sampling point locations are identified as the locations of false peaks.

In general, the peak amplitude of EEG signal is different from one subject to another where it can vary between 600 μV and 1100 μV (Iwasaki, et al., 2005). Another research work by Sovierzoski, Argoud, and de Azevedo (2008), have analyzed the electrical behavior of EEG eye blink events. The research work has recorded the minimum, maximum, and the average of the peak amplitude. The minimum value of amplitude was 55 μV . The highest value of amplitude was 533 μV . The average of peak amplitude was 170 μV . These findings showed that the peak amplitude can vary from 55 μV up to 533 μV and it depends on subjects. Sometimes, the amplitude is higher than usual due to containing various noises.

In this study, the single and double eye blink signals that are shown in Figure 3.3(a) and Figure 3.3(b) were recorded from different subjects. It is also demonstrated that both figures consist of different baseline. Based on this reason, the peak amplitude of double blink is much lower than the single eye blink. However, for one particular subject, the same value of peak amplitude between double and single eye blinks can be observed as shown in Figure 3.3(b). Moreover, the peak amplitude can be measured for both figures where the values are approximately around 300 μ V to 500 μ V.

Figure 3.3(c) shows the third case of EEG signals that was labeled as eye movement signals. The eye movement signals have 40 signals of C3 and C4 channels, 10-second length per signal, 2560 sampling points per signal, and each signal containing one known actual peak point location. The known peak pattern in this signal represents the horizontal eye gaze direction, either to the left or the right.

Several research works have used C3 and C4 channels to record the response of eye gaze direction in EEG signals. Also, CZ channel is commonly used as a reference for EEG signals. The C3, C4, and CZ channels are used because of they have relatively little less contamination from EEG artifacts due to eye blinking (Klados et al, 2011). The C3 and C4 channels also have been proven to be used on human-machine interface (HMI) application such as wheelchair navigation (Ramli, Arof, Ibrahim, Mokhtar, & Idris, 2015).

In total, the data collection has 40-second length and 102400 sampling points. From 102400 sampling points, 3881 candidate peak locations were recognized where the known actual peak point locations are 40 and the remaining sampling points are the known actual non-peak point location.

Table 3.1: Description of the signals

Type of signal	No. of signals	No. of sampling points per signal	Length per signal (second)	No. of peaks per signal	Class distribution per signal (peak point/non-peak point)	Total number of (candidate peaks/true peaks/false peaks)
Single eye blink	30	2560	10	2	2/2558	3238/60/3178
Double eye blink	5	20480	80	8	8/20472	4662/40/4622
Eye movement	40	2560	10	1	1/2559	3881/40/3841

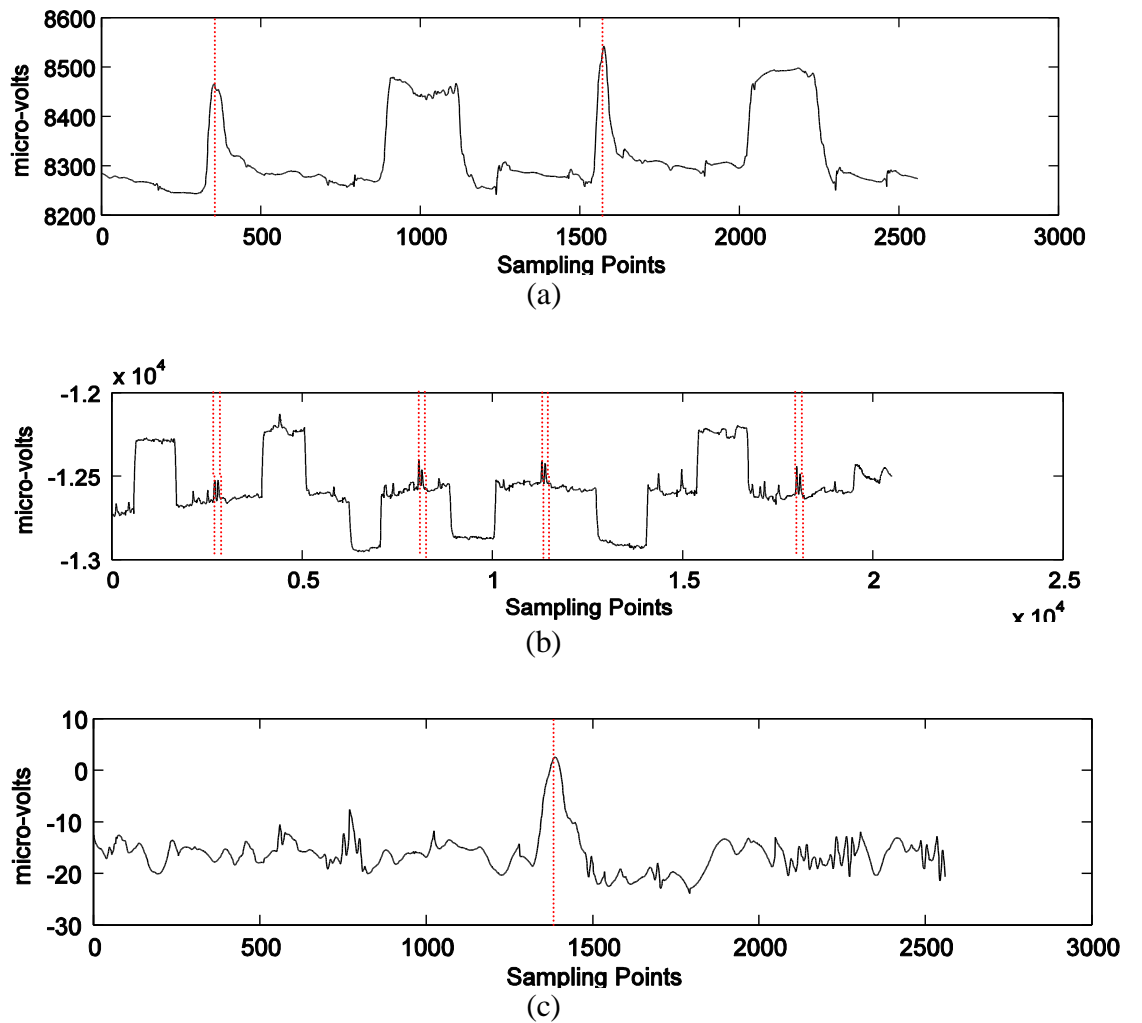


Figure 3.3: The filtered EEG signals: (a) single eye blink (2 peak points per signal), (b) double eye blink (8 peak points per signal), and (c) eye movement (one peak point per signal)

3.3 Performance Measure

The *Gmean* is calculated as follows:

$$TPR = \frac{TP}{TP + FN} \quad (3.1)$$

$$TNR = \frac{TN}{TN + FP} \quad (3.2)$$

$$Gmean = \sqrt{TPR \times TNR} \quad (3.3)$$

where any true peak (*TP*) is the correctly detected apex point of a peak candidate, a true non-peak (*TN*) is any correctly detected non-peak point of a peak candidate, a false peak (*FP*) is an incorrectly designated non-peak point of a peak candidate, a false non-peak (*FN*) is any incorrectly detected true peak point of peak candidate, *TPR* is the true peak rate, and *TNR* is the true non-peak rate.

3.4 Feature Extraction

A peak model of the time domain analysis can be defined based on the selection of eight parameter points as shown in Figure 3.4. The set of parameter points comprised of the *i*th candidate peak point, PP_i , the two associated valley points, $VP1_i$ and $VP2_i$, the half point at first half wave ($HP1_i$), the half point at second half wave ($HP2_i$), the turning point at first half wave ($TP1_i$), the turning point at second half wave ($TP2_i$), and the moving average curve point ($MAC(PP_i)$).

The *i*th candidate peak point, PP_i , are identified using three-points sliding window method (Billauer, 2012; Dumpala, Reddy, & Sarna, 1982). By considering discrete-time signals, $x(I)$, of L points, those three-points are denoted as $x(I-1)$, $x(I)$ and $x(I+1)$ for $I = 1, 2, 3, \dots, L$. A candidate peak point is identified when

$x(PP_i - 1) < x(PP_i) > x(PP_i + 1)$ and two associated valley points, $VP1_i$ and $VP2_i$, are in between. Both valley points exist when $x(VP1_i - 1) > x(VP1_i) < x(VP1_i + 1)$ and $x(VP2_i - 1) > x(VP2_i) < x(VP2_i + 1)$. The half point at first half wave is defined as the point located in the middle between the PP_i and $VP1_i$ while the half point at second half wave is defined as the point based in the midst between the PP_i and $VP2_i$. The turning point is defined as the point where the slope decreases more than 50% as compared to the slope of the preceding point. The $MAC(PP_i)$ point is located at the intersection between the PP_i and $MAC(PP_i)$ points. The window length of the moving averaging is 100 sampling points.

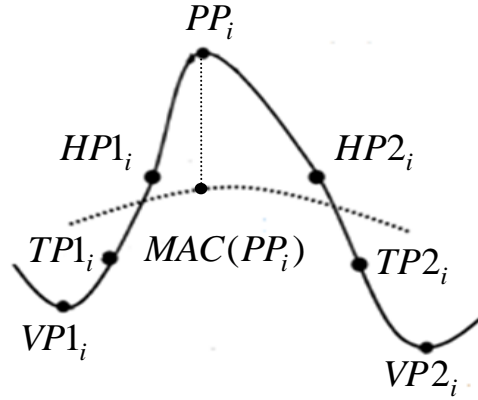


Figure 3.4: Eight point locations of a peak

The time domain-based features of a candidate peak can be categorized into three groups, namely amplitude, width, and slope as shown in Table 3.2. Referring to Table 3.2, there are five different amplitudes, seven different widths, and four different slopes that can be calculated based on the eight defined points, resulting in a total of 16 features, which can be defined as follows:

- 1) The peak-to-peak amplitude at the first half wave, f_1 , is the peak amplitude between the magnitude of the peak and the magnitude of the valley of the first half wave.
- 2) The peak-to-peak amplitude at the second half wave, f_2 , is the peak amplitude between the magnitude of the peak and the magnitude of the valley of the second half wave.
- 3) The turning point amplitude at the first half wave, f_3 , is the peak amplitude between the magnitude of the peak and the magnitude of the turning point at the first half-wave. The turning point is defined as the point where the slope decreases more than 50% compared to the slope of the preceding point.
- 4) The turning point amplitude at the second half wave, f_4 , is the peak amplitude between the magnitude of the peak and the magnitude of the turning point at the second half wave.
- 5) The moving average amplitude, f_5 , is the peak amplitude between the magnitude of the peak and the magnitude of the moving average.
- 6) The peak width, f_6 , is the peak width between the valley point of the first half point and the valley point of the second half wave.
- 7) The first half wave width, f_7 , is the peak width between the peak point and the valley point of the first half wave.
- 8) The second half wave width, f_8 , is the peak width between the peak point and the valley point of the second half wave.
- 9) The turning point width, f_9 , is the peak width between the turning point at the first half wave and the turning point at the second half wave.
- 10) The first half-wave turning point width, f_{10} , is the peak width between the turning point at the first half-wave and the peak point.

- 11) The second half wave turning point width, f_{11} , is the peak width between the turning point at the second half-wave and the peak point.
- 12) The half point width, f_{12} , is the peak width between the half point of the first half-wave and the half point of the second half-wave.
- 13) The peak slope at the first half wave, f_{13} , is the maximal slope between the peak point and the valley point of the first half wave.
- 14) The peak slope at the second half-wave, f_{14} , is the peak slope between the peak point and the valley point of the second half wave.
- 15) The turning point slope at the first half-wave, f_{15} , is the peak slope between the peak point and the turning point of the first half-wave.
- 16) The turning point slope at the second half wave, f_{16} , is the peak slope between the peak point and the turning point of the second half-wave.

Table 3.2: Equations and descriptions of peak features

Peak Feature	Feature Name	Equation	Description
Amplitudes	Peak-to-peak amplitude of the first half wave	$f_1 = x(PP_i) - x(VP1_i) $	Amplitude between the magnitude of peak and the magnitude of valley at the first half wave
	Peak-to-peak amplitude of the second half wave	$f_2 = x(PP_i) - x(VP2_i) $	Amplitude between the magnitude of peak and the magnitude of valley of the second half wave
	Turning point amplitude of the first half wave	$f_3 = x(PP_i) - x(TP1_i) $	Amplitude between the magnitude of peak and the magnitude of turning point at the first half wave
	Turning point amplitude at the second half wave	$f_4 = x(PP_i) - x(TP2_i) $	Amplitude between the magnitude of peak and the magnitude of turning point at the second half wave
	Moving average amplitude	$f_5 = x(PP_i) - MAC(PP_i) $	Amplitude between the magnitude of peak and the magnitude of moving average
Widths	Peak width	$f_6 = VP1_i - VP2_i $	Width between valley point of first half point and valley point at second half wave
	First half wave width	$f_7 = PP_i - VP1_i $	Width between peak point and valley point at first half wave
	Second half wave width	$f_8 = PP_i - VP2_i $	Width between peak point and valley point of second half wave
	Turning point width	$f_9 = TP1_i - TP2_i $	Width between turning point at first half wave and turning point at the second half wave
	First half wave turning point width	$f_{10} = PP_i - TP1_i $	Width between turning point at first half wave and peak point
	Second half wave turning point width	$f_{11} = PP_i - TP2_i $	Width between turning point at second half wave and peak point
	FWHM	$f_{12} = HP1_i - HP2_i $	Width between half point of first half wave and half point of second half wave
Slopes	Peak slope at the first half wave	$f_{13} = \left \frac{x(PP_i) - x(VP1_i)}{PP_i - VP1_i} \right $	Slope between a peak point and valley point at the first half wave
	Peak slope at the second half wave	$f_{14} = \left \frac{x(PP_i) - x(VP2_i)}{PP_i - VP2_i} \right $	Slope between a peak point and valley point at the second half wave
	Turning point slope at the first half wave	$f_{15} = \left \frac{x(PP_i) - x(TP1_i)}{PP_i - TP1_i} \right $	The slope between peak point and turning point at the first half wave
	Turning point slope at the second half wave	$f_{16} = \left \frac{x(PP_i) - x(TP2_i)}{PP_i - TP2_i} \right $	The slope between peak point and turning point at the second half wave

3.5 Peak Model Selection

Peak model selection is a conventional approach in detecting the desired peaks of EEG signals. This approach has successfully been applied in various applications as highlighted in the literature review. Each peak model has its own set of features. The list of different peak models and their sets of the feature are tabulated in Table 3.3.

Table 3.3: List of different peak models and sets of feature

Peak Model	Set of Feature	Number of Features
Dingle	f_5, f_6, f_{13}, f_{14}	4
Dumpala	f_1, f_6, f_{13}, f_{14}	4
Acir	$f_1, f_2, f_7, f_8, f_{13}, f_{14}$	6
Liu	$f_1, f_2, f_3, f_4, f_6, f_9, f_{12}, f_{13}, f_{14}, f_{15}, f_{16}$	11
Full feature set	$f_1, f_2, f_3, f_4, f_5, f_6, f_7, f_8, f_9, f_{10}, f_{11}, f_{12}, f_{13}, f_{14}, f_{15}, f_{16}$	16

3.6 Classification Methods

3.6.1 Artificial Neural Network with Particle Swarm Optimization as Learning

Algorithm for Artificial Neural Network Classifier

The architecture of a single layer feedforward ANN is shown in Figure 3.5. In the network architecture, the input weights w_{ij} are located on the connection between the input layer and the hidden layer, while the output weights, w_{jk} are located on the link between the hidden layer and the output layer. The total number of weights is dependent on the number of inputs, n , number of outputs, m , and number of neurons, l , which can be calculated using the following equation:

$$Total \text{ Weights} = (n \times l) + (l \times m) \quad (3.4)$$

As shown in Figure 3.5, the hyperbolic tangent (tanh) function is chosen for all neurons at hidden and linear function for two neurons in the output layer. The output of each neuron at hidden layer is denoted as y_j and the output of neuron at output layer is denoted as y_k . The output, y is a classifier output. The outputs, y_j , y_k , and y can be defined as follows,

$$y = \begin{cases} 0 & \text{if } y_k \leq \theta \\ 1 & \text{if } y_k > \theta \end{cases} \quad (3.5)$$

where,

$$y_k = \tanh(net_k) \quad (3.6)$$

$$y_j = \tanh(net_j) \quad (3.7)$$

$$\tanh(net) = \frac{e^{net} - e^{-net}}{e^{net} + e^{-net}} \quad (3.8)$$

$$net_k = \sum_{j=1}^l y_j w_{jk} \quad (3.9)$$

Based on Equation (3.5), the classifier can classify the output into two classes (output): class 0 and class 1. The classifier produces 1 if y_k is greater than the decision threshold, θ . Otherwise, the classifier produces 0.

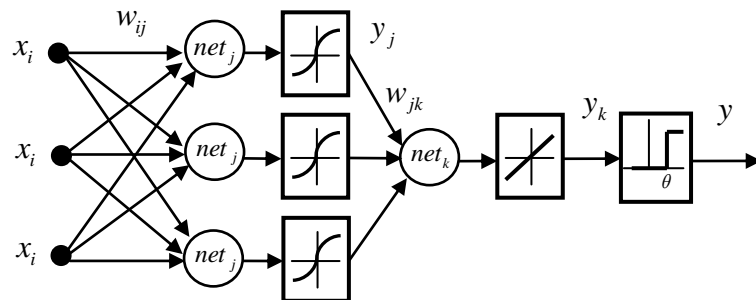


Figure 3.5: Architecture of a single layer feedforward ANN

PSO is a stochastic population-based optimization algorithm introduced by J. Kennedy and R. Eberhart (1995). This algorithm is based on the movement and information sharing of particles in a multi-dimensional search space. The PSO algorithm has been numerously enhanced fundamentally (Lim et al, 2013; Mohamad et al, 2013) and applied in many fields (Adam et al, 2010; Ayob et al, 2010; Z. Ibrahim et al, 2012). A pseudo code of PSO algorithm is described in Algorithm 1.

In the preliminary stage of PSO algorithm, some parameters are initialized: (1) PSO parameters, (2) position of particles, and (3) velocity of particles. PSO parameters include the maximum number of particles, the inertia weight, ω , the cognitive component, c_1 , the social component, c_2 , the random values r_1 and r_2 , and the maximum number of iterations, k . The position of particles is randomly located in the search space, and the velocity of particles is set to zero.

Algorithm 1: PSO Algorithm

```

1: Initialization
2: while not stopping criteria do
3:   for each ith particle in a population do
4:     calculate fitness evaluation function
5:     update pbest and gbest
6:   end for
7:   for each particle in a population do
8:     update the ith particle's velocity and
9:     update the ith particle's position
10:  end for
11: end while

```

After the initialization stage is done, the fitness evaluation function is calculated. The fitness evaluation function is problem dependent. Then, the algorithm follows by updating the personal best (*pbest*) and the global best (*gbest*). The *pbest* is the best solution found by each particle, while *gbest* is the best solution among the *pbest*.

After PSO algorithm progresses through the initialization and the fitness evaluation function calculation is used for updating *pbest* and *gbest*, the velocities and the positions of each particle are calculated and updated. The velocity of a particle is updated using the following equation:

$$v_i^{k+1} = \omega v_i^k + c_1 r_1 (pbest - s_i^k) + c_2 r_2 (gbest - s_i^k) \quad (3.10)$$

where v_i^k is the velocity particle i at iteration k , r_1 and r_2 are random numbers $[0, 1]$, and c_1 and c_2 denote the cognitive and social coefficients, respectively. The particle's new velocity is then used to update the particle's position using Equation (3.11).

$$s_i^{k+1} = s_i^k + v_i^{k+1} \quad (3.11)$$

where s_i^k is the position of a particle i at k th iteration. In this study, the linear dynamic inertia weight (Shi & Eberhart, 1998a, 1998b) is used and calculated according to Equation (3.12) as follows:

$$\omega = \omega_{\max} - \frac{\omega_{\max} - \omega_{\min}}{k_{\max}} \times k \quad (3.12)$$

where ω_{\max} and ω_{\min} denote the maximum and minimum values of inertia weight, respectively, and k_{\max} is the maximum iteration. The linear dynamic inertia weight takes a value between $\omega_{\max} = 0.9$ and $\omega_{\min} = 0.4$. Then, PSO algorithm is terminated based on a stopping criterion. For example, if the algorithm reaches maximum iterations, then the algorithm is stopped.

PSO algorithm adapts the neural network parameters during a training process. The primary concern is to find, during the training process, the value of the input and output weights and the decision threshold for producing the best classification performance. A process flow of ANN with PSO algorithm for peak detection algorithm is schematically illustrated in Figure 3.6.

Table 3.4 illustrates the representation of particle position, particle velocity, $pbest$, and $gbest$. The position of particle i at iteration k is denoted as

$$s_i^k = \{w_{i,1}^k, w_{i,2}^k, w_{i,3}^k, \dots, w_{i,d}^k, w_{i,1}^k, w_{i,2}^k, w_{i,3}^k, \dots, w_{i,D}^k, \theta_{i,e}^k\}. \quad (3.13)$$

While the velocity of particle i at iteration k is denoted as

$$v_i^k = \{v_{i,1}^k, v_{i,2}^k, v_{i,3}^k, \dots, v_{i,d}^k, v_{i,1}^k, v_{i,2}^k, v_{i,3}^k, \dots, v_{i,D}^k, v_{i,e}^k\}. \quad (3.14)$$

The $pbest$ of particle i is represented as

$$pb_i^k = \{pb_{i,1}^k, pb_{i,2}^k, pb_{i,3}^k, \dots, pb_{i,d}^k, pb_{i,1}^k, pb_{i,2}^k, pb_{i,3}^k, \dots, pb_{i,D}^k, v_{i,e}^k\} \quad (3.15)$$

and the $gbest$ is denoted as

$$gb^k = \{gb_1^k, gb_2^k, gb_3^k, \dots, gb_d^k, gb_1^k, gb_2^k, gb_3^k, \dots, gb_D^k, gb_e^k\}. \quad (3.16)$$

The $d = 1, 2, 3, \dots, nl$ is a d th dimension of input weights, the $D = nl + 1, nl + 2, nl + 3, \dots, nl + lm$ is a D th dimension of output weights, and the $nl + lm + 1$ is an e th dimension of decision threshold. nl is the total number of input weights. lm is the total number of output weights. The total number of weight is denoted as $nl + lm$. The total number of particle dimension is denoted as $nl + lm + 1$.

Table 3.4: Representation of particle position, particle velocity, $pbest$, and $gbest$

Particle	Notations	Neural Network Parameters								
		Input Weights				Output Weights				Decision Threshold
Dimension		1	2	...	nl	$nl + 1$	$nl + 2$...	$nl + lm$	$nl + lm + 1$
Position	s_i^k	$w_{i,1}^k$	$w_{i,2}^k$...	$w_{i,d}^k$	$w_{i,1}^k$	$w_{i,2}^k$...	$w_{i,D}^k$	$\theta_{i,e}^k$
Velocity	v_i^k	$v_{i,1}^k$	$v_{i,2}^k$...	$v_{i,d}^k$	$v_{i,1}^k$	$v_{i,2}^k$...	$v_{i,D}^k$	$v_{i,e}^k$
$pbest$	pb_i^k	$pb_{i,1}^k$	$pb_{i,2}^k$...	$pb_{i,d}^k$	$pb_{i,1}^k$	$pb_{i,2}^k$...	$pb_{i,D}^k$	$pb_{i,e}^k$
$gbest$	gb^k	$gb_{i,1}^k$	$gb_{i,2}^k$...	gb_d^k	gb_1^k	gb_2^k	...	$gb_{i,D}^k$	gb_e^k

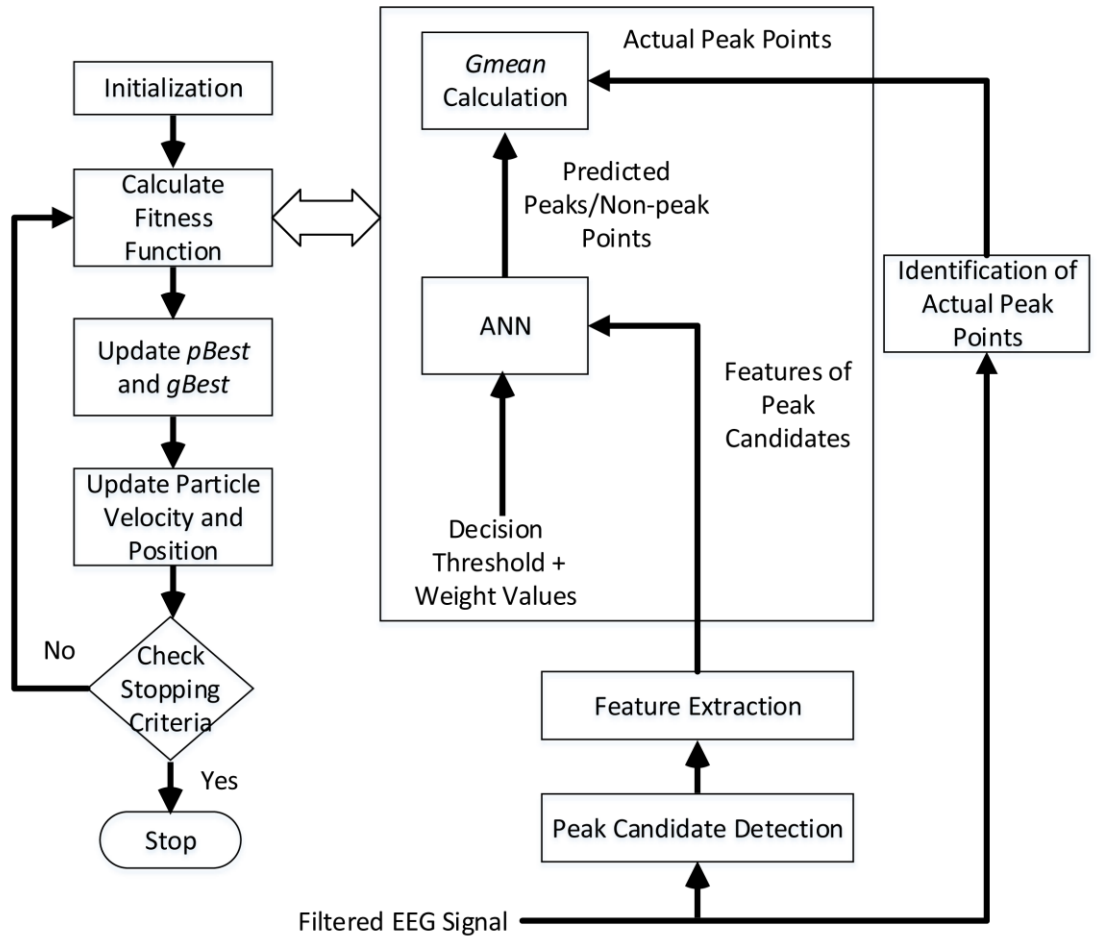


Figure 3.6: ANN with PSO algorithm for peak detection algorithm

3.6.1.1 Experiment 1: Evaluation on Different Peak Models for EEG Signals Peak Detection using Artificial Neural Network

The objective of this experiment is to evaluate the performance of every peak model for peak detection algorithm using ANNPSO classifier. The performance is evaluated using Gmean accuracy. For experimental setup, the first 50% of the EEG signals is divided for training data, and the remaining 50% is for testing data. For the ANN classifier, as shown in Table 3.5, the numbers of neurons are selected using a trial and error method which is set to 10. The hyperbolic tangent $[-1, 1]$ is used as an activation function in the hidden layer for normalization, while a linear function is located inside the neuron in the output layer. Other settings for the ANN classifier, such as the number of neurons in the input layer and the total number of weights, are dependent on the dataset as calculated using Equation (3.4). PSO algorithm is employed as the learning algorithm which is to produce the optimal weights and decision threshold value. The PSO parameter values are based on the suggested values by Shi and Eberhart (1999). The parameters setting of PSO are tabulated in Table 3.6. For each run, 10 particles are used, and the maximum iteration was set to 150. This experiment was repeated 10 trials and the average, maximum, minimum, and STDEV of 10 times is taken as the final results.

Table 3.5: Parameters setting of ANN

Parameters	Value
Number of neuron at hidden layer	10
Activation function at hidden layer	Hyperbolic tangent [-1, 1]
Activation function at output layer	Linear function
Number of neurons in the input layer	Depends on number of features
Total number of weights	$(n \times l) + (l \times m)$

Table 3.6: Parameters setting of PSO

Parameters	Value
Decrease inertia weight, ω	0.9 ~ 0.4
Cognitive component, c_1	2
Social component, c_2	2
Random value, r_1 and r_2	Random number [0,1]
Initial velocity vector for each particle	0
Initial $pbest$ score for each particle	0
Initial $gbest$ score	0

3.6.1.2 Experimental Results

Table 3.7 shows the performance of each peak model for a single eye blink signal. For the single eye blink signal experiments, the best average test performance is 91.94%, which was obtained by Acir peak model. On the other hand, the worst test performance is 73.86%, which was obtained by Dingle peak model. Based on Acir peak model test performance, the maximum test performance reached 97.08%, the minimum test performance dropped to 85.26%, and the standard deviation value is 4.12%. The best and worst test performance is also highlighted in the table. The sensitivity and

specificity of Acir peak model are 94.91% and 89.38%, respectively. The whole sensitivity and specificity of test performance are shown in Table 3.8.

Table 3.9 shows the performance of each peak model for the double eye blink signal. Based on the end results in Table 3.9, the best average test performance is 87.47%, which was obtained by Acir peak model. The worst test performance is 75.79%, which was obtained by Dingle peak model. Based on Acir peak model test performance, the maximum test performance reached 92.10%, the minimum test performance dropped to 82.72%, and the standard deviation value is 3.17%. The best and worst test performance is also highlighted in the table. The sensitivity and specificity of Acir peak model are 88.74% and 86.33%, respectively. The whole sensitivity and specificity of test performance for double eye blinking problem are shown in Table 3.10.

Table 3.11 shows the performance of each peak model for the eye movement signals. At the testing phase, the Dumpala model produced 80.2%, 85.1%, 72.2%, and 4.2% accuracies for average, maximum, minimum, and STDEV, respectively. The Acir model produced 73.7%, 84.1%, 65.6%, and 5.9% accuracies for average, maximum, minimum, and STDEV, respectively. The Liu model produced 75.9%, 80.6%, 69.4%, and 3.8% accuracies for average, maximum, minimum, and STDEV, respectively. The Dingle model produced 87.6%, 92.4%, 83.8%, and 3.2% accuracies for average, maximum, minimum, and STDEV, respectively. It can be observed that the Dingle model gave the best average testing result, with accuracy of 87.59%. In contrast, the worst average testing result is 73.67% which produced by the Acir model. The best and worst test performance is also highlighted in the table. The sensitivity and specificity of Dingle peak model are 85.4% and 90%, respectively. The whole sensitivity and specificity of test performance for double eye blinking problem are shown in Table 3.12.

Overall, Acir peak model offers the highest classification performance rate on the single eye blink and double eye blink signals. Whereas, Dingle model offers the highest classification performance rate on the eye movement signals. In general, it may be due to the small variation of peak features of Acir and Dingle peak models. Therefore, the small variation of peak features relaxes the learning process of the classifier in the algorithm.

Moreover, the main concern in developing any techniques resorts to the ability of generalizing the model. The main index to measure the generalization capability is best to indicate by minimum *Gmean*. Referring in Table 3.7 and Table 3.9, Acir peak model obtained the highest classification rate for the minimum testing result on the single eye blink (85.26%) and double eye blink (82.72%) as compared to the Dumpala, Liu, and Dingle peak models. Referring to Table 3.11, Dingle peak model obtained the highest classification rate for the minimum testing result on the eye movement signals (83.8%) as compared to another model. Therefore, based on these experimental results, Acir peak model provided a good generalized model on the single eye blink and double eye blink signals whereas Dingle model provided a good generalized model on the eye movement signals.

The ANNPSO classifier is used to overcome the problem of the conventional ANN which is developed for balanced dataset problem. In this study, the collected EEG data is categorized under the problem of imbalanced dataset as presented in Table 3.1. The ratio for the three signals between false peak and true peak are very imbalanced. For example, the ratio between false peak and true peak for single eye blink signal is 60:3178. For this case, the right classifier for imbalanced dataset should be chosen at the first stage of this study. Hence, the ANNPSO classifier is chosen once it has been developed for imbalanced dataset problem.

Table 3.7: Performance of peak detection models for single eye blink signals

Peak Model	Training Results (%)				Testing Results (%)			
	Average	Max	Min	STDEV	Average	Max	Min	STDEV
Dumpala	91.23	92.30	89.86	00.69	81.05	84.48	76.35	3.18
Acir	97.61	99.21	95.86	1.07	91.94 ^{best}	97.08	85.26	4.12
Liu	93.10	95.56	89.75	2.05	88.11	94.82	76.79	5.81
Dingle	84.21	87.32	79.62	2.37	73.86 ^{worst}	81.18	64.76	5.41

Table 3.8: Sensitivity and specificity test results for single eye blink signals

Peak Model	Sensitivity (%) =	Specificity (%) =
	$\frac{TP}{TP + FN} \times 100$	$\frac{TN}{TN + FP} \times 100$
Dumpala	94.11	70.00
Acir	94.91	89.38
Liu	90.02	86.88
Dingle	76.16	72.50

Table 3.9: Performance of peak detection models for double eye blink signals

Peak Model	Training Results (%)				Testing Results (%)			
	Average	Max	Min	STDEV	Average	Max	Min	STDEV
Dumpala	85.42	87.46	82.64	1.43	80.14	83.28	76.26	2.18
Acir	89.05	91.04	87.20	1.15	87.47 ^{best}	92.10	82.72	3.17
Liu.	89.75	91.25	86.24	1.64	86.10	89.65	81.33	2.64
Dingle	83.79	87.65	80.56	2.40	75.79 ^{worst}	83.22	64.48	5.91

Table 3.10: Sensitivity and specificity test results for double eye blink signals

Peak Model	Sensitivity (%) =	Specificity (%) =
	$\frac{TP}{TP + FN} \times 100$	$\frac{TN}{TN + FP} \times 100$
Dumpala	86.57	74.33
Acir	88.74	86.33
Liu	89.08	83.33
Dingle	75.20	77.00

Table 3.11: Performance of peak detection models for eye movement signals

Peak Model	Training Results (%)				Testing Results (%)			
	Average	Max	Min	STDEV	Average	Max	Min	STDEV
Dumpala	84.1	87	80.9	1.8	80.2	85.1	72.2	4.2
Acir	88.6	92.1	85	1.9	73.7 ^{worst}	84.1	65.6	5.9
Liu	85.2	90.3	81.1	2.7	75.9	80.6	69.4	3.8
<i>Dingle</i>	<i>90.9</i>	<i>92.7</i>	<i>87.7</i>	<i>1.9</i>	<i>87.6 ^{best}</i>	<i>92.4</i>	<i>83.8</i>	<i>3.2</i>

Table 3.12: Sensitivity and specificity test results for eye movement signals

Peak Model	Sensitivity (%) =	Specificity (%) =
	$\frac{TP}{TP + FN} \times 100$	$\frac{TN}{TN + FP} \times 100$
Dumpala	83.16	78.00
Acir	77.00	71.50
Liu	79.21	73.50
Dingle	85.40	90.00

Also, the comparison of the average test accuracy between the four peak models is proved using Friedman's test as statistical analysis. The analysis indicates that there is a difference in test accuracies between the peak models with p -value is lower than 0.01 for the two signals. The average ranking of Friedman's test for the single and double eye blink signals are tabulated in Table 3.13. For both signals, the first ranked is Acir peak model, then it is followed by Liu, Dumpala, and Dingle peak models. The post-hoc analysis for Friedman's test is then evaluated to find the relationship through the

comparison of the four peak models. The post-hoc analysis is based on Holm-Bonferroni method using two difference confidence intervals, $\alpha=0.05$ and $\alpha=0.10$. Both Friedman's test and Holm-Bonferroni post-hoc analysis are carried out by using KEEL software tool (Alcala-Fdez et al, 2009). The results in Table 3.15 show similar patterns for $\alpha=0.05$ and $\alpha=0.10$ for the two signals where Acir peak model offers significantly better test accuracies than Dingle and Dumpala peak models. In addition, there is no significant difference in test accuracies for Acir and Liu peak models, Dumpala and Dingle peak models, and Dumpala and Liu peak models. In other words, the Acir peak model is proved through statistical analysis to offer the highest accuracy as compared to previous results tabulated in Table 3.7, Table 3.8, Table 3.9, and Table 3.10.

The average ranking of Friedman's test for the eye movement signals is tabulated in Table 3.14. Referring to Table 3.14, the first ranked is Dingle peak model, then it is followed by Dumpala, Liu, and Acir peak models. The post-hoc analysis for Friedman's test for eye movement signals is presented in Table 3.16. The post-hoc analysis shows that Dingle peak model offers significantly better test accuracies than Acir, Liu, and Dumpala peak models, which the p -value for condition 4, 5, and 6 are lower than 0.01 for $\alpha=0.05$ and $\alpha=0.10$.

Table 3.13: Average ranking of Friedman's test with $p<0.01$ for single and double blink signals

Peak Model	Average Ranking		Rank
	Single Eye Blink Signals	Double Eye Blink Signals	
Dumpala	2.9	3.1	3
Acir	1.3	1.3	1
Liu	2	1.8	2
Dingle	3.8	3.8	4

Table 3.14: Average ranking of Friedman's test with $p < 0.01$ for eye movement signals

Peak Model	Average Ranking	Rank
Dumpala	2.5	2
Acir	3.3	4
Liu	3.2	3
Dingle	1	1

Table 3.15: Pos-hoc analysis for Friedman's test for single and double eye blink signals

<i>i</i>	Condition	Single Eye Blink Signal				Double Eye Blink Signal			
		$\alpha = 0.05$		$\alpha = 0.10$		$\alpha = 0.05$		$\alpha = 0.10$	
		<i>p</i>	Holm	<i>p</i>	Holm	<i>p</i>	Holm	<i>p</i>	Holm
6	Acir vs. Dingle	0.00002	0.00833	0.00002	0.01667	0.00002	0.00833	0.00002	0.01667
5	Liu vs. Dingle	0.00182	0.01	0.00182	0.02	0.00053	0.01	0.00053	0.02
4	Dumpala vs. Acir	0.00558	0.0125	0.00558	0.025	0.00182	0.0125	0.00182	0.025
3	Dumpala vs. Liu	0.11903	0.01667	0.11903	0.03333	0.02434	0.01667	0.02434	0.03333
2	Dumpala vs. Dingle	0.11903	0.025	0.11903	0.05	0.22535	0.025	0.22535	0.05
1	Acir vs. Liu	0.22535	0.05	0.22535	0.1	0.38648	0.05	0.38648	0.1

Table 3.16: Pos-hoc analysis for Friedman's test for eye movement signals

<i>i</i>	Condition	$\alpha = 0.05$		$\alpha = 0.10$	
		<i>p</i>	Holm	<i>p</i>	Holm
6	Acir vs. Dingle	0.000068	0.008333	0.000068	0.016667
5	Liu vs. Dingle	0.000139	0.01	0.000139	0.02
4	Dumpala vs. Dingle	0.009375	0.0125	0.009375	0.025
3	Dumpala vs. Acir	0.165857	0.016667	0.165857	0.033333
2	Dumpala vs. Liu	0.225346	0.025	0.225346	0.05
1	Acir vs. Liu	0.86249	0.05	0.86249	0.1

3.6.2 Neural Network with Random Weights Classifier

NNRW is a fast learning algorithm of a single layer feedforward neural network (SLFN). NNRW was firstly introduced by Schmidt (1992). The network of NNRW consists of three layers that are input, hidden, and output layers. The input weights are located between the input layer and hidden layer. Whereas in between of the hidden layer and output layer, are the output weights. The learning concept of NNRW is that the input weights and the biases at the hidden layer in the network are chosen randomly with a specific interval, whereas the output weights are estimated by the Moore-Penrose

generalized inverse method (C.R. Rao & Mit, 1971). The input weights are assigned randomly between -1 and 1. Also, the biases in the hidden layer are assigned randomly between 0 and 1. Both parameters follow the setup parameters in (Cao, Ye, & Wang, 2015). A similar concept of NNRW was further developed by Pao and Takefuji (1992), knowingly as random vector functional-link (RVFL) nets. Variations of extended RVFL were introduced to establish the theoretical results of the RVFL concept (Igelnik & Pao, 1995; Pao et al, 1994).

The output function of NNRW classifier of a given unknown sample, x can be mathematically described as

$$f(x) = h(x)\beta \quad (3.17)$$

The output matrix of the hidden layer, H , is calculated as follows:

$$H = \begin{bmatrix} h(x_1) \\ \vdots \\ h(x_N) \end{bmatrix} = \begin{bmatrix} g\left(\sum_{i=1}^d a_{i1}x_{1i} + b_1\right) & \cdots & g\left(\sum_{i=1}^d a_{iL}x_{1i} + b_L\right) \\ \vdots & \ddots & \vdots \\ g\left(\sum_{i=1}^d a_{i1}x_{Ni} + b_1\right) & \cdots & g\left(\sum_{i=1}^d a_{iL}x_{Ni} + b_L\right) \end{bmatrix}_{N \times L} \quad (3.18)$$

where g is an activation function of the hidden neuron, x is the $N \times L$ matrix of inputs, a is the $d \times L$ matrix of random input weights, b is the $1 \times L$ matrix of random biases in the hidden layer, N is an arbitrary distinct sample, L is the number of hidden neurons, and d is the number of inputs. The i th column of H is the output of the i th hidden neuron with respect to inputs x_1, x_2 , until x_d .

The NNRW can be represented as a linear system. The linear system of NNRW is mathematically modeled as

$$H\beta = T \quad (3.19)$$

where β is the $L \times m$ matrix of output weights and T is the $N \times m$ matrix of target outputs. m is the number of output neurons. The β and T matrixes are denoted as

$$\beta = \begin{bmatrix} \beta_1^T \\ \vdots \\ \beta_L^T \end{bmatrix}_{L \times m} \quad (3.20)$$

and

$$T = \begin{bmatrix} t_1^T \\ \vdots \\ t_N^T \end{bmatrix}_{N \times m} \quad (3.21)$$

, respectively. To find the least square solution, β of the linear system, $H\beta = T$, the minimum-norm least-squares solution is computed as follows:

$$\|H(a_1, \dots, a_L, b_1, \dots, b_L)\beta - T\| = \min_{\beta} \|H(a_1, \dots, a_L, b_1, \dots, b_L)\beta - T\| \quad (3.22)$$

It is well known that the smallest norm least-squares solution of Equation (3.22) is

$$\beta = (H^T H)^{-1} H^T T = H^+ T \quad (3.23)$$

where H^+ is the Moore-Penrose pseudo-inverse of H . The summary of the training stages of the NNRW classifier is listed as follows:

Stage 1: Assign randomly the input weights, a_i and biases in the hidden neurons, b_i .

Stage 2: Calculate the output matrix of the hidden layer, H .

Stage 3: Calculate the output weights, $\beta = H^+T$.

In the output layer, two neurons are used in the network to classify the output into two classes (output): class 1 and class 0. For two classes ($m > 1$), the predicted class label is the i th number of the output neurons which the maximum value of output neuron. The predicted class label of a given unknown sample x is defined as follows.

$$label(x) = \arg \max_{i \in \{1, \dots, m\}} f_i(x) \quad (3.24)$$

3.6.2.1 Experiment 2: Evaluation on Different Peak Models using NNRW Classifier

The aim of this experiment is to evaluate every peak model for peak detection algorithm using the NNRW classifier. For the NNRW classifier of each experiment, all parameters setup are shown in Table 3.17. The sigmoidal function $g(x) = 1/(1 + e^{-x})$ was used as an activation function in the hidden layer for normalization while a linear function is located inside the neuron in the output layer. Other settings for the NNRW classifier, such as the number of neurons in the input layer, is dependent on the number of selected features of the particular peak model. The number of output neurons was set to 2. Notice that the input weights and the biases in the hidden layer remain fixed during the training but the value of these two NNRW parameters are changed for each run.

Table 3.17: Parameters setting of NNRW

Parameters	Value
Activation function at hidden layer	Sigmoid [-1, 1]
Activation function at output layer	Linear function
Number of neurons in the input layer	Depends on number of features
Number of neurons in the output layer	2
Number of neurons in the feature scaling layer	Depends on number of features

One advantage of the NNRW classifier is that the learning algorithm is less difficult than other conventional neural network classifier (i.e., gradient descent, levenberg marquart, and particle swarm optimization-based learning algorithms). So that, with an enormous number of hidden neurons is possible to perform using the NNRW classifier. However, the optimal number of neurons of the NNRW classifier is required to be firstly identified for offering better generalization ability of the NNRW classifier. To find the optimal number of hidden neuron, an experiment is executed by varying the number of hidden neuron from 100 to 4000 in steps of 500. In this experiment, the first 50% of each signal is divided for training and the remaining 50% of each signal for testing. The experiment was repeated 30 trials and the average of 30 times is taken as the final results.

3.6.2.2 Experimental Results

The average of training accuracy of five different peak models with respect to the number of hidden neurons for single eye blink, double eye blink, and eye movement signals are tabulated in Table 3.18, Table 3.19, and Table 3.20, respectively. Referring to Table 3.18, Table 3.19, and Table 3.20, generally, the classifier produces the highest training accuracy and the lowest value of standard deviation when the numbers of

hidden neurons are between 1500 and 4000 for all peak models. Hence, for further investigation on the classification performance of the five different models, the number of hidden neurons for single eye blink, double eye blink, and eye movement signals was set to 1500.

Figure 3.7 shows the final test results with respect to the number of hidden neurons for single eye blink signals (see Figure 3.7(a)), double eye blink signals (see Figure 3.7(b)), and eye movement signals (see Figure 3.7(c)), respectively. Referring to Figure 3.7(a), 5(b), and 5(c), the testing accuracy improves up to a certain level and starts slowly decreasing afterwards. The NNRW classifier offers the optimal accuracy when the numbers of hidden neurons are between 1500 and 3000 for all peak models and signals. Moreover, the final results in Figure 3.7 indicate that the selection of relevant features is necessary for providing the best and generalizes performance in EEG signals peak classification.

Table 3.18: Average training accuracy with respect to the number of hidden neurons for single eye blink signals

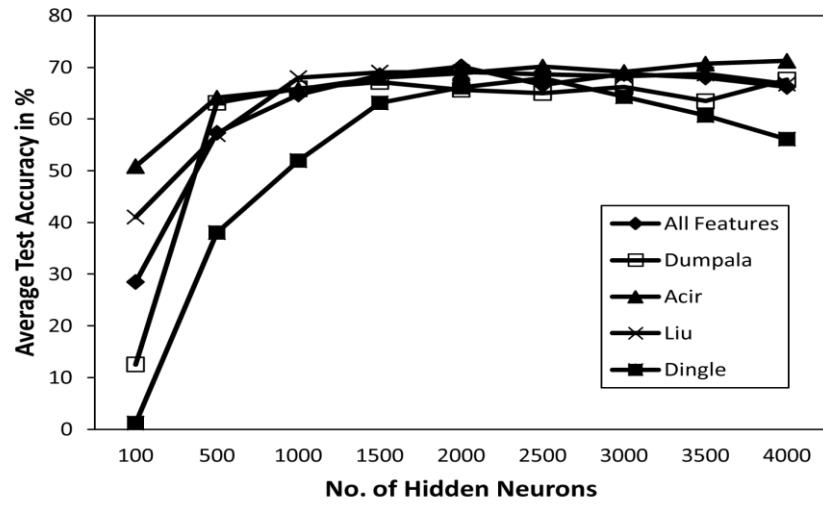
Peak model		No. of hidden node								
		100	500	1000	1500	2000	2500	3000	3500	4000
Full feature set	Training acc (%)	46	93	97	98	98	98	98	98	98
	STDEV (%)	9.6	1.6	0.8	0.6	0.3	0	0	0	0.4
Dumpala	Training acc (%)	17	92	100	100	100	100	100	100	100
	STDEV (%)	19.4	2.7	0	0	0	0	0	0	0
Acir	Training acc (%)	40	86	97	100	100	100	100	100	100
	STDEV (%)	9.9	2.7	0.9	0	0	0	0	0	0
Liu	Training acc (%)	47	93	97	97	97	97	97	97	97
	STDEV (%)	9.1	1.1	0	0	0	0	0	0	0
Dingle	Training acc (%)	1	84	99	100	100	100	100	100	100
	STDEV (%)	5.8	4.6	1.4	0	0	0	0	0	0

Table 3.19: Average training accuracy with respect to the number of hidden neurons for double eye blink signals

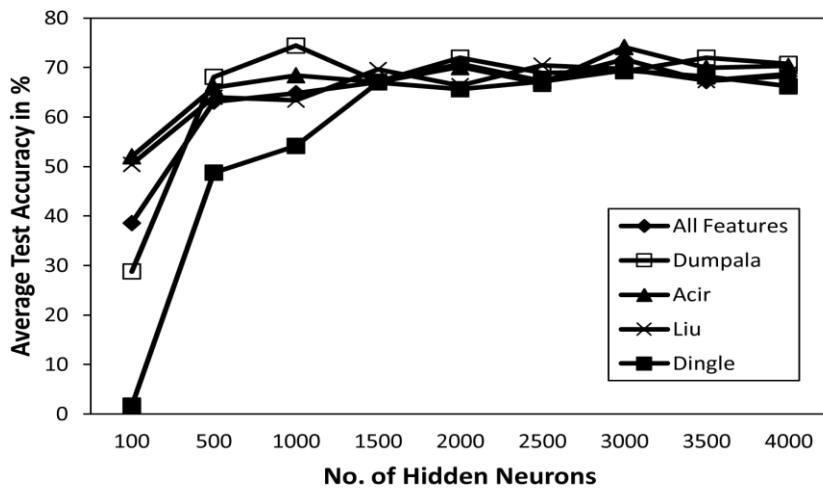
Peak model		No. of hidden node								
		100	500	1000	1500	2000	2500	3000	3500	4000
Full feature set	Training acc (%)	43	78	83	83	83	83	83	83	83
	STDEV (%)	10.4	1.4	0.8	0.5	0.7	0.9	0.8	0.9	0.9
Dumpala	Training acc (%)	34	81	85	87	87	87	87	87	87
	STDEV (%)	11.4	1.1	0	0.3	0	0	0	0	0
Acir	Training acc (%)	52	79	85	85	85	87	87	87	87
	STDEV (%)	8.9	2.8	0	0	0	0	0	0	0
Liu	Training acc (%)	53	79	83	83	83	83	83	83	83
	STDEV (%)	7.8	0.9	0.5	0	0.3	0	0	0.6	0.3
Dingle	Training acc (%)	4	78	85	87	87	87	87	87	87
	STDEV (%)	7.8	3.1	0	0	0	0	0	0	0

Table 3.20: Average training accuracy with respect to the number of hidden neurons for eye movement signals

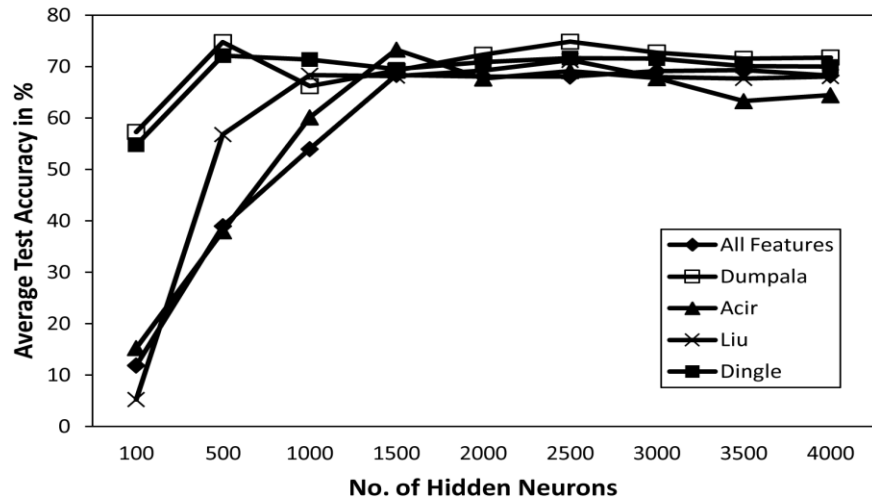
Peak model		No. of hidden node								
		100	500	1000	1500	2000	2500	3000	3500	4000
Full feature set	Training acc (%)	5	87	95	95	95	95	95	95	95
	STDEV (%)	9.9	4	0	0	0	0	0	0	0
Dumpala	Training acc (%)	55	85	91	100	100	100	100	100	100
	STDEV (%)	14.3	1.5	1.4	0.8	0	0	0	0	0
Acir	Training acc (%)	33	88	92	100	100	100	100	100	100
	STDEV (%)	20.4	1.4	0.8	0	0	0	0	0	0
Liu	Training acc (%)	2	80	95	95	95	95	95	95	95
	STDEV (%)	6.9	2.9	0	0	0	0	0	0	0
Dingle	Training acc (%)	66	100	100	100	100	100	100	100	100
	STDEV (%)	11.1	0.9	0	0	0	0	0	0	0



(a)



(b)



(c)

Figure 3.7: Average testing accuracy with respect to the number of hidden neurons: (a) single eye blink signals, (b) double eye blink signals, and (c) eye movement signals

The comparison results of average testing accuracy for five different peak models on the three signals are exhibited in Table 3.21. Please be noted that this investigation is using the previous results of 1500 number of hidden neurons for single eye blink, double eye blink, and eye movement signals. From Table 3.21, Liu model obtained the best accuracy for single and double eye blink signals, with 69.1% and 69.6% of accuracy, respectively. The Acir model obtained 73%, which the best performance for eye movement signals. Overall, based on the average accuracy of all signals, the best peak model is produced by Acir model (69.3%). However, Friedman's test statistical analysis indicates that there is no significance difference in test accuracies between the peak models with 1500 number of hidden neurons for the three signals.

From Table 3.21, it can be observed that the all peak models have not contributed to give the best detection performance for all signals when using the NNRW classifier. It may due to the combinations of peak feature contain some irrelevant features. This study also showed that all subjects did not produce the same peak features during data collection session. One subject itself did not produce a small variation of features due to the lack of concentration during data collection. To overcome this case, this study attempted to find the relevant features for every signal through the evaluation of the different peak models.

Table 3.21: Comparison of testing accuracy in terms of average values for three different signals over five peak models

Peak model	Single eye blink (%)	Double eye blink (%)	Eye movement (%)	Average (%)
Full feature set	68.5	67.1	68	67.9
Dumpala	67.2	67.1	69	67.8
Acir	68	67	73	69.3
Liu	69.1	69.6	68	68.9
Dingle	63.1	67	69	66.4

3.7 Performance Comparison of Different Peak Models between the ANNPSO and NNRW Classifiers

Table 3.22 shows the classification performance in term of average test accuracy between the NNRW and ANNPSO classifiers. Based on the average result, the classification performance of the ANNPSO classifier is better than the NNRW classifier, with 81.9% accuracy. However, the ANNPSO took a longer time to compute than the NNRW. The NNRW performed lower than 75% due to several contributing factors such as the collected data is affected by various noises and the peak features have a large different value from one subject to another subject. These factors are the cause to the high variation of peak features of the four peak models. Moreover, it might be caused by the effect imbalanced classes of the dataset.

In term of computational time between ANNPSO and NNRW methods, an average time is recorded and tabulated in Table 3.22. The NNRW classifier performs almost ten times faster than the PSOANN method. The NNRW provides a fast learning mechanism rather than the ANNPSO. Therefore, the NNRW classifier is used in this study for future development to improve the performance of peak classification algorithm.

Table 3.22: Performance comparison of average test results between the NNRW and PSOANN classifiers

Peak model	NNRW				ANNPSO			
	Single eye blink (%)	Double eye blink (%)	Eye movement (%)	Average (%)	Single eye blink (%)	Double eye blink (%)	Eye movement (%)	Average (%)
Dumpala	67.2	67.1	69	67.8	81.1	80.1	80.2	80.5
Acir	68	67	73	69.3	91.9	87.5	73.7	84.4
Liu	69.1	69.6	68	68.9	88.1	86.1	75.9	83.4
Dingle	63.1	67	69	66.4	73.9	75.8	87.6	79.1
Average				68.1				81.9
Average computational time				10 minutes				130 minutes

3.8 Summary

In this chapter, a generalized peak detection algorithm for three different cases of EEG signals (i.e. single eye blink, double eye blink, and eye movement signals) was presented. The fundamental aspects of the four main processes, i.e. peak candidate identification, feature extraction, peak model selection, and classification were explained. Four peak models have been considered in this chapter to evaluate the peak detection algorithm (i.e. Dumpala, Acir, Liu, and Dingle models). The full feature set was also considered as a performance comparison to other models. In classification process, two famous ANN techniques were used which are ANNPSO and NNRW classifiers.

From the two experimental results, in general, it was found that the ANNPSO gave better performance than the NNRW. However, the NNRW provides the fast learning speed rather than the ANNPSO. Therefore, due to this complexity on learning speed, it is impossible to improve the detection performance using the ANNPSO classifier. Based on this finding, the NNRW was selected as classification method into the peak detection algorithm.

CHAPTER 4: THE PROPOSED GENERALIZED MODEL FOR PEAK CLASSIFICATION OF EVENT-RELATED EEG SIGNALS

4.1 Introduction

The existing peak classification algorithms tend to employ a particular model of specific event-related EEG signals rather than use one model of the relationships between several event-related EEG signals. The previously experimental results indicated that none of the existing peak models perform efficiently in all EEG signals. In consequence, the utilization of the existing models is not sufficient for achieving a good result for peak classification of several event-related EEG signals. In this chapter, this problem is addressed by employing feature selection technique. To produce a new generalized model, the developed feature selection techniques are evaluated using 11781 samples of peak candidate that was collected from three different peak event-related EEG signals of 30 healthy subjects; 1) single eye blink, 2) double eye blink, and 3) eye movement signals.

At the beginning of the feature selection process, the collected data of peak candidate samples are randomly divided by using four-fold cross-validation process. Then, the training, validation, and testing sets are prepared for experiments.

In feature selection technique, the NNRW classification performance is examined from various combinations of 16 peak features by four novel estimation-based optimization algorithms: 1) angle modulated simulated Kalman filter (AMSKF), 2) binary simulated Kalman filter (BSKF), 3) local optimum distance evaluated simulated Kalman filter (LocalDESKF), and 4) global optimum distance evaluated simulated Kalman filter (GlobalDESKF). Every algorithm produces the best and generalized peak model. The peak models are defined as generalized peak model I, generalized peak

model II, generalized peak model III, generalized peak model IV for AMSKF, BSKF, LocalDESKF, and GlobalDESKF algorithms, respectively.

4.2 Research Methodology

In the following sections comprise of the details for introduction of the proposed four novel estimation-based optimization algorithms including AMSKF, BSKF, LocalDESKF, and Global DESKF methods. At first, the approach of preparing training, validation, and testing sets are explained in Subchapter 4.3.

The details explanation of SKF algorithm, AMSKF algorithm, and the proposed feature selection approach using AMSKF algorithm are presented in Subchapter 4.5.1, 4.5.2, and 4.5.3, respectively.

The following Subchapter 4.6.1 and 4.6.2 are the details description of BSKF algorithm and the proposed feature selection approach using BSKF algorithm, respectively.

The following Subchapter 4.7.1 and 4.7.2 are the details description of LocalDESKF algorithm and the proposed feature selection approach using LocalDESKF algorithm, respectively.

Next, the details explanation of GlobalDESKF algorithm, and the proposed feature selection approach using GlobalDESKF algorithm are presented in Subchapter 4.8.1 and 4.8.2, respectively.

4.3 Preparing Training, Validation, and Testing Sets

From the collected raw data of the three EEG signals as previously introduced in Section 3.2, 11781 peak candidate samples with their associated features were archived as EEG data for experiments. From 11781 peak candidate samples, 140 were assigned as true peaks and the other 11461 were assigned as false peaks.

To prepare the experiment data, the four-fold cross-validation process is used to produce four groups of EEG data: each group consists of training and testing sets. Next, the training set is randomly divided into two: training and validation sets. Both datasets were set to equally distribute the two-class ratio. The ratio size of training and validation data was set to 0.5:0.5. The testing set is utilized as unseen EEG data. After all four groups are evaluated by the algorithm, the maximum value of testing results from the four groups is measured and the best peak model with the associated features is recorded. This entire four-fold cross validation process is repeated 30 times to obtain the final statistical results (e.g., average, maximum, minimum, and standard deviation) for all experiments.

4.4 Performance of NNRW under Various Number of Hidden Neurons

The optimal number of neurons of the NNRW classifier is required to be firstly identified for offering better generalization ability of the NNRW classifier. To find the optimal number of hidden neuron, this experiment is executed by varying the number of hidden neuron from 100 to 1200 in steps of 100.

The variation of training and testing accuracies with respect to a different number of hidden neurons is graphically illustrated in Figure 4.1. Referring to Figure 4.1, the training accuracy of all four peak models gradually increased up to 84.5%, 81.7%,

80.2%, and 77.2% at 1200 hidden neurons for Dumpala, Acir, Liu, and Dingle, respectively. The computational time takes too long time when the number of hidden neurons increased more than 1200 neurons. Hence, the numbers of hidden neurons for the next experiments were set to 1000.

The results in Figure 4.1 shows the testing accuracy of all four peak models also gradually increased around 45% to 55% at 1200 neurons. Three peak models (e.g., Dumpala, Acir, and Liu models) except Dingle model offer the optimal testing accuracy (48% to 55%) when the numbers of hidden neurons are between 900 and 1200. The final results in Figure 4.1 indicate that the selection of the best combination features is necessary for providing the best and generalizes performance in EEG signals peak classification.

Table 4.2 shows the average ranking of Friedman's test of the algorithm with different models: Dumpala, Acir, Liu, and Dingle models. The statistical results show that the lowest average ranking is obtained by Acir model that represents ranking first among the four models for EEG data. While, the algorithm with Dumpala model ranking second, the algorithm with Liu model ranking third, and the algorithm with Dingle model ranking fourth.

Next, p -values for unadjusted values and adjusted p -values for Nemenyi, Holm, Shaffer, and Bergmann-Hommel test for $N \times N$ comparisons for all possible 6 pairs of the algorithm with the peak models are presented in Table 4.3. The p -values below 0.05 represent that the particular algorithm with the peak model differs significantly in testing accuracy. The p -values below 0.05 were marked with the italic font.

From Table 4.3, it can be observed that p -values for unadjusted values, Nemenyi, Holm, Shaffer, and Bergmann-Hommel offer for eliminating five hypotheses. Based on

unadjusted p-values and adjusted p-values for Nemenyi, Holm, Shaffer, and Bergmann-Hommel test, there are a significance different in average testing accuracy between Acir and Dingle, Dumpala and Dingle, Acir and Liu, Liu and Dingle, and also Dumpala and Liu.

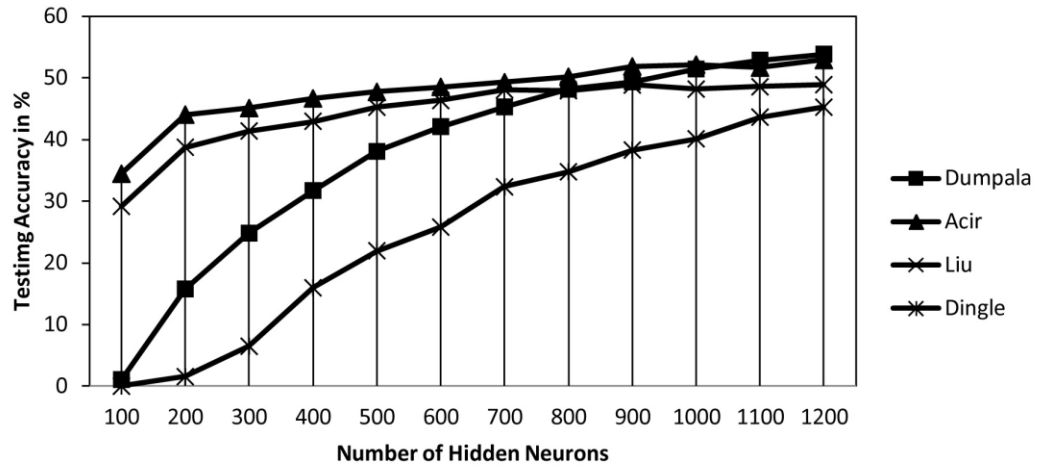


Figure 4.1: Variation of testing accuracy of NNRW classifier with respect to number of hidden neurons

Table 4.1: Classification accuracy results for NNRW classifier under different number of hidden neurons

Peak model	Result	No. of hidden neurons											
		100	200	300	400	500	600	700	800	900	1000	1100	1200
Dumpala	Train	5.15	30.1	43.61	53.39	60.26	66.51	71.27	75.55	78.63	80.86	82.96	84.54
	Test	1.09	15.77	24.83	31.75	38.09	42.12	45.31	48.17	49.37	51.46	52.9	53.87
Acir	Train	37.69	48.95	53.37	56.87	59.82	63.27	66.41	70.06	73.69	76.3	79.38	81.73
	Test	34.46	44.05	45.11	46.67	47.74	48.55	49.3	50.2	51.86	52.16	51.67	52.91
Liu	Train	35.61	48.54	54.83	60.38	65.41	69.09	71.94	73.99	75.52	77.18	78.62	80.16
	Test	29.18	38.76	41.4	42.97	45.25	46.34	48.07	47.94	48.85	48.19	48.57	48.91
Dingle	Train	0	6.19	19	31.13	41.89	49.91	57.07	61.96	68.14	71.39	75.12	77.22
	Test	0	1.55	6.48	15.97	21.97	25.81	32.34	34.78	38.31	40.13	43.65	45.26

Table 4.2: The average ranking of the Dumpala, Acir, Liu, and Dingle, achieved by Friedman

Peak Model	Average Ranking	Rank
Dumpala	1.8	2
Acir	1.6	1
Liu	2.8	3
Dingle	3.9	4
Statistic	59.2	
<i>p</i> -value	5.07E-11	

Table 4.3: Adjusted p-value for $N \times N$ comparisons of algorithms over 30 runs

Algorithm vs. Algorithm	pUnadj	pNeme	pHolm	pShaf	pBerg
Acir vs. Dingle	0	0	0	0	0
Dumpala vs. Dingle	0	0	0	0	0
Acir vs. Liu	0.000465	0.002792	0.001861	0.001396	0.001396
Liu vs. Dingle	0.000967	0.005801	0.002901	0.002901	0.001934
Dumpala vs. Liu	0.0027	0.016199	0.0054	0.0054	0.0027
Dumpala vs. Acir	0.617075	3.70245	0.617075	0.617075	0.617075

4.5 The Proposed Generalized Model I based on Angle Modulated Simulated Kalman Filter (AMSKF) Algorithm

This section is further divided into four sub-sections that describe a technique to produce a generalized model I using AMSKF algorithm. The first sub-section begins with the explanation of the theories and formulations of SKF algorithm. The SKF algorithm was further expanded to be AMSKF algorithm for discrete optimization problems and feature selection, in which will be explained in Sub-section 4.5.2 and Sub-section 4.5.3, respectively. The conducted experiments that to find a generalized model I based on AMSKF feature selection approach are demonstrated in Sub-section 4.5.4.

4.5.1 Simulated Kalman Filter (SKF) Algorithm

The SKF algorithm (Zuwairie Ibrahim, et al., 2015) was originally invented for solving continuous optimization problems. The algorithm follows several steps as shown in Figure 4.2: (1) generate an initial population, (2) calculation of the fitness evaluation function for each agent, (3) update the best fitness value among agents at every iteration (X_{best}) and the best solution compared to the current X_{best} (X_{true}), (4) perform state prediction, measurement, and estimation, and (5) perform termination based on a stopping criterion.

In the initialization step, several initial SKF parameters such as the initial value of error covariance estimate, $P(0)$, the process noise value, Q , and the measurement noise value, R are required. Further settings, such as, the number of n agents and maximum number of iterations, t_{max} , are also determined. The states values of each agent are given randomly within a specific interval.

Next, the fitness evaluation function is computed for every agent to obtain initial solutions. The best fitness value among each agent at every iteration t , $X_{\text{best}}(t)$ can be either in the maximization problem, $\max_{i \in 1, \dots, n} \text{fit}((X(t)))$ or minimization problem $\min_{i \in 1, \dots, n} \text{fit}((X(t)))$.

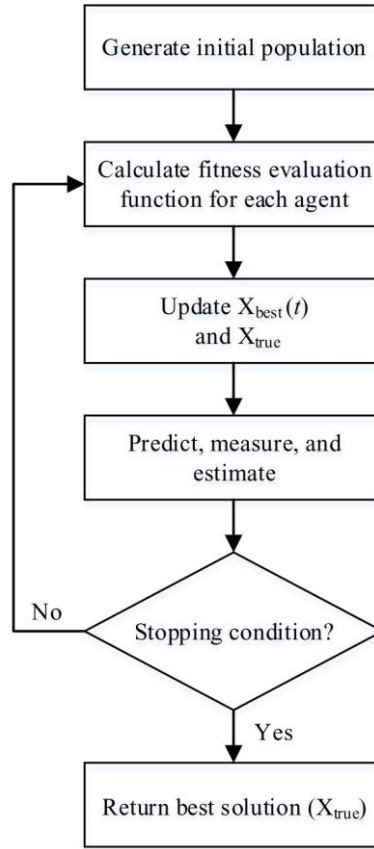


Figure 4.2: The simulated Kalman filter (SKF) algorithm

The $X_{\text{best}}(t)$ value at every iteration t is compared and the best among the $X_{\text{best}}(t)$ value, which is X_{true} is updated. For a maximization problem, X_{true} is only updated when $X_{\text{best}}(t)$ at current iteration is greater than X_{true} . Whereas, for a minimization problem, X_{true} is only updated when $X_{\text{best}}(t)$ at current iteration is lower than X_{true} .

Referring to Figure 4.2, the next following steps include the state prediction, measurement, and estimation. The state prediction follows the following equations:

$$X_i(t|t-1) = X_i(t-1) \quad (4.1)$$

$$P(t|t-1) = P(t-1) + Q \quad (4.2)$$

where, $X_i(t-1)$ and $X_i(t|t-1)$ are the previous state and transition state, respectively.

$P(t|t-1)$ and $P(t-1)$ are previous error covariant estimate and transition error covariant estimate, respectively. From Equation (4.2), it noted that the transition error covariant estimate is controlled by the process noise, Q .

In the state measurement step, the following equation, $Z_i(t)$, is used, which gives some feedbacks to the estimation process.

$$Z_i(t) = X_i(t|t-1) + \sin(rand \times 2\pi) \times |X_i(t|t-1) - X_{true}| \quad (4.3)$$

From Equation (4.3), the $\sin(rand \times 2\pi)$ term offered the stochastic element of SKF algorithm which having a random probability distribution to the measurement value and $rand$ is a uniformly distributed random number in the range of [0 1].

Next, the Kalman gain, $K(t)$, is computed based on the calculated value of the transition error covariant estimate, $P(t|t-1)$ and the measurement noise value, R . The equation of $K(t)$ is given as follows.

$$K(t) = \frac{P(t|t-1)}{P(t|t-1) + R} \quad (4.4)$$

Here, the equation for estimating the next state, $X_i(t)$, is given in Equation (4.5) and the error covariant is updated based on Equation (4.6). Finally, the processes are iteratively looped until the maximum number of iteration is reached.

$$X_i(t) = X_i(t|t-1) + K(t) \times (Z_i(t) - X_i(t|t-1)) \quad (4.5)$$

$$P(t) = (1 - K(t)) \times P(t|t-1) \quad (4.6)$$

4.5.2 Angle Modulated Simulated Kalman Filter (AMSKF) Algorithm

For solving discrete optimization problems, the angle modulated concept is embedded into SKF algorithm. The AMSKF is firstly introduced by Md Yusof, et al. (2016a). Referring to Figure 4.3, additional two steps of the angle modulated into SKF are described as follows. After the initialization step, the continuous signals, $g(x)$ with four coefficient parameters (a , b , c and d) are generated for each agent. So, the state of the i th agent in a population at iteration t is denoted as $X_i(t) = \{a_i, b_i, c_i, d_i\}$. As mentioned before, the state values which are a , b , c , and d are given randomly in an initial stage. The function $g(x)$ with the four coefficient parameters is defined as follows,

$$g(x) = \sin(2\pi(x-a) \times b \times \cos(2\pi(x-a) \times c)) + d \quad (4.7)$$

An example plot of function, $g(x)$ for the case of $a = 0$, $b = 1$, $c = 1$, and $d = 0$ is given in Figure 4.4. From the signals, the sampling time, T , is chosen to generate a bit string of length n in the next step. The bit 1 is generated when $g(x)$ value is greater than 0 while, the bit 0 is generated when $g(x)$ value is lower than 0. The length of the bit string depends on the given problem. For example, if the length of the full feature set is 100, so the length of the bit string is 100. The generated bit string of each agent is employed to calculate the fitness value for each agent. Then, AMSKF follows similar steps as SKF until it returns the final solution. Using the angle modulated approach, the

AMSKF algorithm only tunes the four coefficient parameters for getting the best solution.

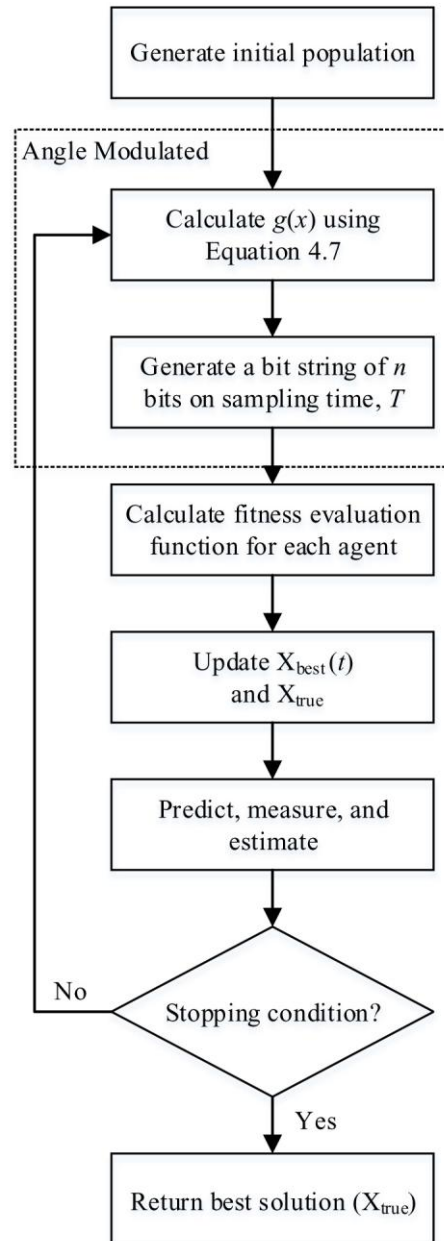


Figure 4.3: The angle modulated simulated Kalman filter (AMSKF) algorithm

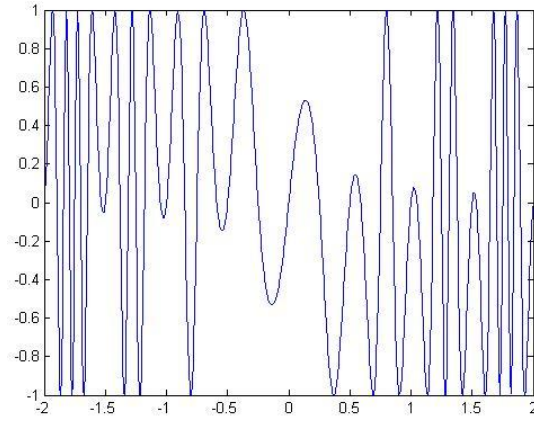


Figure 4.4: An example of $g(x)$ function with $a = 0$, $b = 1$, $c = 1$, and $d = 0$

4.5.3 Feature Selection using Angle Modulated Simulated Kalman Filter

The proposed feature selection algorithm for EEG signals peak detection is based on AMSKF algorithm. In addition, NNRW classifier is employed for peak classification in the algorithm. The combination of both methods is illustrated in the flowchart as shown in Figure 4.5.

From Figure 4.5, the proposed AMSKF technique begins with initialization of a population and then calculation of a $g(x)$ function. The maximum number of iteration was set to 500 and the number of agents was set to 10. The initial value of the error covariance estimate, P , process noise value, Q , and measurement noise value, R , are 10000, 0.5, and 0.5, respectively. To employ AMSKF algorithm for feature selection in EEG peak classification, a total of the 16-bit string is generated since the selection of one feature is determined by one-bit value. If AMSKF assigns bit value 1 to an i th feature, the i th feature is selected. Otherwise, the i th feature is not selected.

In the calculation process of the fitness evaluation function, the selected features are used to prepare the training and validation sets, as shown in Figure 4.5. To calculate the

fitness evaluation function, at first, the classifier has to be trained by the given training data. Then, the trained classifier is tested using the validation set. The detection performance of the training and validation sets are computed based on *Gmean* (Guo et al, 2008). The *Gmean* of validation set is set as fitness value for AMSKF algorithm.

From Figure 4.5, subsequent to the calculated fitness value, the process is continued to the next following processes; update $X_{\text{best}}(t)$ and X_{true} , state measurement, state prediction, and state estimation. Next, new 16 bits are determined and those processes are looped until it reaches maximum iterations. Finally, the best peak model associated with the trained NNRW was determined by this feature selection approach.

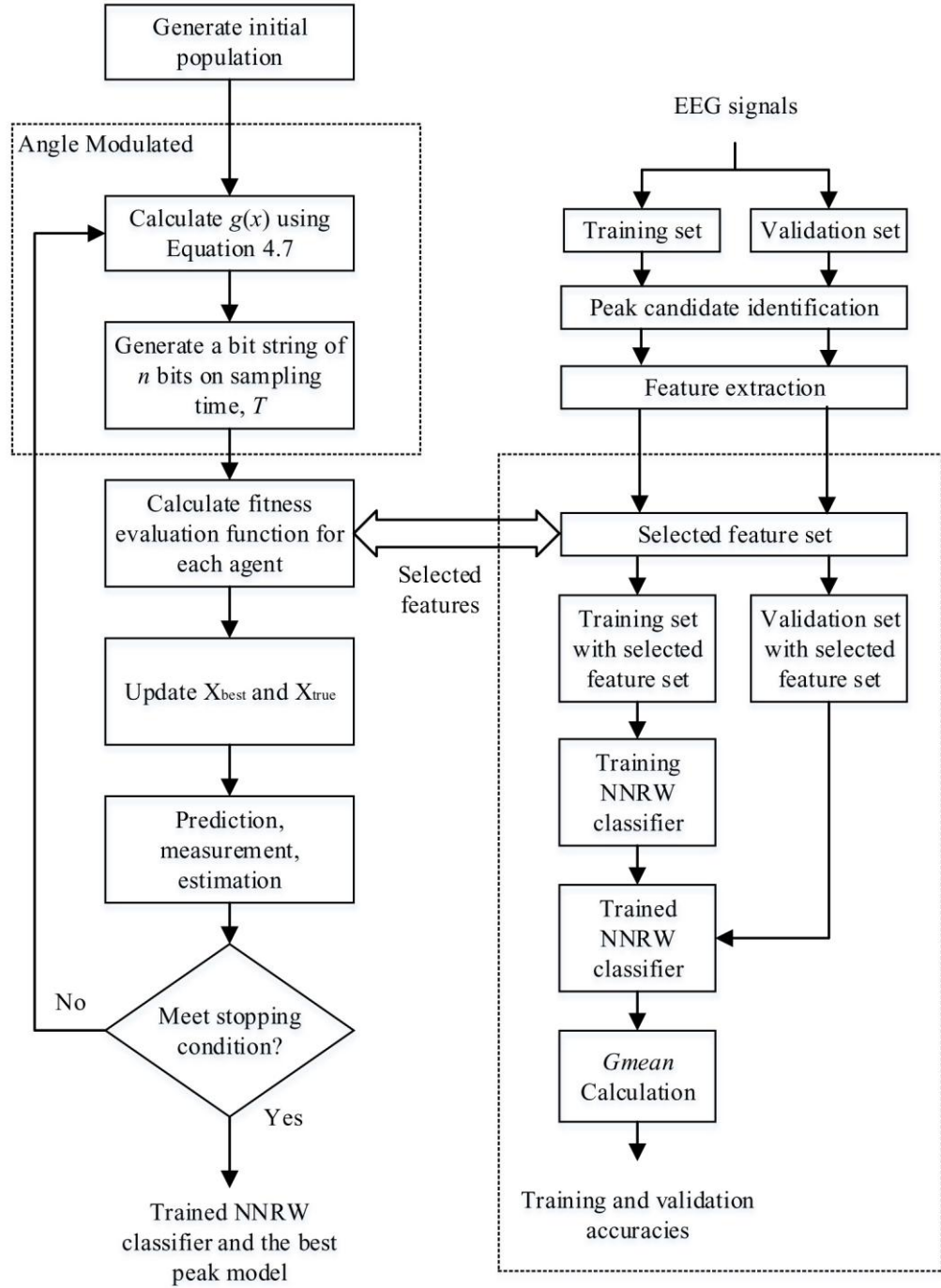


Figure 4.5: Flowchart of the proposed AMSKF feature selection algorithm

4.5.4 Experiment 1: To Find a Generalized Model based on AMSKF Feature Selection Approach

For this experiment, the AMSKF algorithm is applied as a feature selector to find the best combination of peak features. The parameters setting of AMSKF algorithm are tabulated in Table 4.4. The maximum iteration for AMSKF algorithm was set to 500. The number of agents was set to 10. The initial value of the error covariance estimate, P , process noise value, Q , and measurement noise value, R , are 1000, 0.5, and 0.5, respectively.

Table 4.4: Parameters setting of AMSKF algorithm

Parameters	Value
Maximum iteration, t_{max}	500
Number of agents	10
Initial error covariance estimate value, P	10000
Initial process noise value, Q	0.5
Initial measurement noise value, R	0.5
$rand$	Random number [0,1]

4.5.4.1 Experimental Results and Discussions

Table 4.5 shows the 30 independent runs experimental results of the proposed AMSKF feature selection algorithm using the EEG data that collected from the three recorded EEG signals (i.e., single eye blink, double eye blink, and eye movement signals). The experimental results were used the 50:50 ratios setting of training and validation data sets. Table 4.5 gives the best peak model with the highest training, validation, and testing accuracies for the NNRW classifier at every run. In this

experiment, the best generalized peak model is chosen based on the maximum accuracy of testing data over 30 runs.

From Table 4.5, it is found that the feature set of the best peak model is $f_1, f_2, f_7, f_8, f_9, f_{10}, f_{11}, f_{12}, f_{13}, f_{14}$, and f_{15} , with 72.7% of testing accuracy. From those associated features, two of features are peak amplitudes (e.g., f_1 and f_2), six of features are peak widths (e.g., $f_7, f_8, f_9, f_{10}, f_{11}$, and f_{12}), and three of features are peak slopes (e.g., f_{13}, f_{14} , and f_{15}). For overall of testing accuracy, the average, maximum, minimum, and STDEV over 30 runs are 61.7%, 72.7%, 53%, and 4.1%, respectively.

In this experiment, the proposed AMSKF algorithm was iteratively executed with maximum 500 iterations. To observe the result of the convergence of the proposed AMSKF, one example is taken from this experiment, as illustrated in Figure 4.6. From Figure 4.6, it can be seen that the AMSKF algorithm can reach convergence within 20 iterations.

In order to evaluate the effectiveness of the proposed algorithm and the associated best features, some comparisons are performed in terms of percentage of the testing classification accuracy between the results of the NNRW without and with feature selection algorithm. The comparison results are comparatively presented in Table 6. For the results of the NNRW without feature selection, we evaluated using five existing peak models with their associated features (i.e., full feature set, Dumpala, Acir, Liu, and Dingle models). As seen from Table 4.5, the performance of the proposed generalized AMSKF-based model exceeds the performance of the other existing five models.

Table 4.5: Best results over 30 runs using AMSKF feature selection algorithm

Run	Training (%)	Validation (%)	Testing (%)	Best peak model															
1	87.52	63.88	69.19	1	3	4	6	7	8	9	10	11	12	13	14	15	16		
2	90.14	63.92	62.89	1	2	3	4	5	6	7	8	9	15	16					
3	95.12	61.30	55.78	1	2	3	4	5	6	7	8	9	10	11	16				
4	91.77	61.68	72.71	1	2	7	8	9	10	11	12	13	14	15					
5	78.33	65.99	56.51	13	14	15	16												
6	89.44	71.36	62.21	3	6	7													
7	93.81	67.50	66.78	1	2	8	9	10	11	12	13	14	15	16					
8	96.61	67.19	60.02	1	5	9	13												
9	94.65	64.64	66.50	1	2	14	15												
10	92.20	60.68	57.87	2	3	8	9	10	13	14									
11	95.74	66.54	62.55	1	11	12	15												
12	82.57	65.36	61.47	12	13	14	15	16											
13	92.20	71.06	64.64	1	2	5	13	14	15	16									
14	91.50	71.13	59.16	3	6	14													
15	89.44	58.06	60.60	1	2	3	7	8	10	11	13	15	16						
16	88.19	65.65	60.32	1	2	5	6	7	8	9	10	13	14	15	16				
17	90.83	70.24	55.20	1	2														
18	86.92	67.34	60.51	1	2	5	6	7	8	9	10	13	14	15					
19	95.24	62.63	61.98	1	2	3	4												
20	88.80	68.93	66.51	1	2	3	15	16											
21	85.54	66.92	61.66	9	10	11	12	13	14	15	16								
22	94.15	66.02	57.85	1	3	4	7	9	11	14	16								
23	82.12	62.33	61.34	12	13	14	15	16											
24	95.59	65.14	62.30	1	2	3	9	10											
25	83.67	68.40	62.37	1	2														
26	92.08	66.54	61.75	3	9	15	16												
27	80.18	63.01	61.96	14	15	16													
28	94.15	66.95	52.96	1	10	11	12	13	14										
29	87.60	60.47	63.47	12	13	14	15	16											
30	89.92	71.94	62.34	3	4														

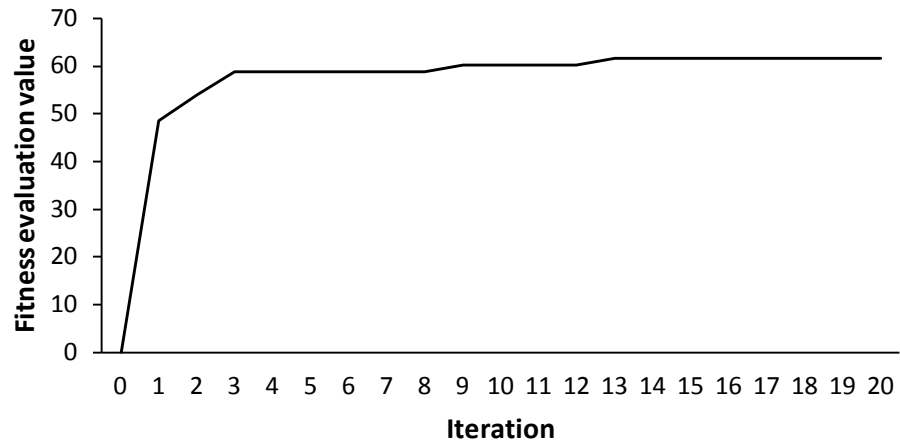


Figure 4.6: Example of the convergence curve of AMSKF

Table 4.6: Comparison of the classification accuracy between the existing peak models and AMSKF model

Peak model	Feature subset length	Selected features	Training Accuracy (%)	Testing Accuracy (%)
Full feature set	16	$f_1, f_2, f_3, f_4, f_5, f_6, f_7, f_8, f_9, f_{10}, f_{11}, f_{12}, f_{13}, f_{14}, f_{15}, f_{16}$	73.8	49.4
Dumpala	4	f_1, f_6, f_{13}, f_{14}	80.9	51.5
Acir	6	$f_1, f_2, f_7, f_8, f_{13}, f_{14}$	76.3	52.2
Liu	11	$f_1, f_2, f_3, f_4, f_6, f_9, f_{12}, f_{13}, f_{14}, f_{15}, f_{16}$	77.2	48.2
Dingle	4	f_5, f_6, f_{13}, f_{14}	71.4	40.1
AMSKF (proposed work)	11	$f_1, f_2, f_7, f_8, f_9, f_{10}, f_{11}, f_{12}, f_{13}, f_{14}, f_{15}$	91.8	72.7

4.5.4.2 Statistical Significance Analysis

The peak detection algorithm with the five different peak models and the AMSKF technique with the best model are further analyzed by using nonparametric Friedman statistical analysis. The statistical analysis is required to demonstrate the significant difference in testing accuracy in terms of average value for the algorithm with the existing five models and the best model of AMSKF technique. The experiments were conducted on statistical procedures designed especially for multiple $N \times N$ comparisons with six models executed in the KEEL data mining system (Alcala-Fdez, et al., 2009).

Table 4.7 shows the average ranking of Friedman's test of the algorithm with different models: full feature set, Dumpala, Acir, Liu, Dingle, and AMSKF models. The statistical results show that the lowest average ranking is obtained by AMSKF model that represents ranking first among the six models for EEG data. While, the algorithm with Acir model ranking second, the algorithm with Dumpala model ranking third, the algorithm with full feature set model ranking fourth, the algorithm with Liu model ranking fifth, and the algorithm with Dingle model ranking sixth.

Next, p -values for unadjusted values and adjusted p -values for Nemenyi, Holm, Shaffer, and Bergmann-Hommel test for $N \times N$ comparisons for all possible 15 pairs of the algorithm with the peak models are presented in Table 4.8. The p -values below 0.05 represent that the particular algorithm with the peak model differs significantly in testing accuracy. The p -values below 0.05 were marked with the italic font.

From Table 4.8, it can be observed that p -values for unadjusted values offer for eliminating twelve hypotheses whereas Nemenyi and Holm ones reject only ten. However, Shaffer and Bergmann-Hommel lets for eliminating only eleven hypotheses. Based on unadjusted p -values and adjusted p -values for Nemenyi, Holm, Shaffer, and Bergmann-Hommel test, the AMSKF model revealed significantly better performance than other peak models.

Table 4.7: The average ranking of the full feature set, Dumpala, Acir, Liu, Dingle, and AMSKF, achieved by Friedman

Peak models	Average ranking	Rank
Dingle	5.867	6
Liu	4.3	5
Full feature set	3.833	4
Dumpala	3.067	3
Acir	2.833	2
AMSKF (proposed work)	1.1	1
Statistic	109.2381	
p -value	8.651E-11	

Table 4.8: Adjusted p-value for $N \times N$ comparisons of algorithms over 30 runs

Algorithm vs. Algorithm	pUnadj	pNeme	pHolm	pShaf	pBerg
Dingle vs. AMSKF	0	0	0	0	0
Liu vs. AMSKF	0	0	0	0	0
Acir vs. Dingle	0	0	0	0	0
Dumpala vs. Dingle	0	0	0	0	0
full feature set vs. AMSKF	0	0	0	0	0
full feature set vs. Dingle	0.000026	0.000384	0.000256	0.000256	0.000154
Dumpala vs AMSKF	0.000047	0.000701	0.000421	0.000327	0.000187
Acir vs. AMSKF	0.000333	0.004992	0.002662	0.00233	0.001331
Liu vs. Dingle	0.001182	0.017723	0.008271	0.008271	0.004726
Acir vs. Liu	0.002395	0.035927	0.014371	0.014371	0.014371
Dumpala vs. Liu	0.010673	0.160088	0.053363	0.04269	0.032018
full feature set vs. Acir	0.038434	0.576509	0.153736	0.153736	0.115302
full feature set vs. Dumpala	0.112478	1.687177	0.337435	0.337435	0.115302
full feature set vs. Liu	0.333998	5.009974	0.667997	0.667997	0.667997
Dumpala vs. Acir	0.629063	9.435947	0.667997	0.667997	0.667997

4.6 The Proposed Generalized Model II based on Binary Simulated Kalman Filter

This section is divided into three sub-sections that describe a technique to produce a generalized model II using BSKF algorithm. The process flow of BSKF algorithm is described in Sub-section 4.6.1. The BSKF algorithm was further developed to be a feature selection technique, in which will be given details in Sub-section 4.6.2. The experiments that to find a generalized model II based on BSKF feature selection method are reported in Sub-section 4.6.3.

4.6.1 Binary Simulated Kalman Filter (BSKF) Algorithm

BSKF is one of SKF variant for solving the discrete optimization problem (Md Yusof, et al., 2015). In the BSKF, the algorithm is similar to the SKF in Figure 4.2, but it requires some modifications to convert the search space from continuous to discrete. The modifications are required in the initialization stage and conversion of the solution to a bit string.

One additional parameter in the initialization stage is a generated random bit string, Σ_i^d , for every agent in a population. The length, d , of the bit string, Σ_i^d , is a problem dependent. For example, in this study, the length, d , is 16 due to the total number of full features are 16. Other initial parameters such as the initial value of error covariance estimate, $P(0)$, the process noise value, Q , the measurement noise value, R , the number of n agents, the maximum number of iterations, t_{\max} , and the states values of each agent are required.

The conversion of the solution to the bit string for each agent is based on the multiplication of the Kalman gain, $K(t)$, and measurement value, $Z_i(t)$ and computed after the evaluation of state prediction, measurement, and estimation. The multiplication equation is denoted as, Δ_i . Then, the term, Δ_i , is mapped into a probabilistic value with interval value within 0 and 1. The equation of Δ_i and the probabilistic function of Δ_i are formulated as follows,

$$\Delta_i = K(t) \times Z_i(t) \quad (4.8)$$

$$S(\Delta_i(t)) = |\tanh \Delta_i(t)| \quad (4.9)$$

After getting the probabilistic value of $S(\Delta_i(t))$, the next step is to update the bit string, Σ_i^d . The update step of every dimension d of an i th agent, Σ_i^d , requires the following rule.

$$\begin{aligned}
& \text{if } rand < S(\Delta_i(t)) \\
& \quad \text{then } \Sigma_i^d(t+1) = \text{complement } \Sigma_i^d(t) \\
& \quad \text{else } \Sigma_i^d(t+1) = \Sigma_i^d(t) \\
& \text{end}
\end{aligned} \tag{4.10}$$

4.6.2 Feature Selection using Binary Simulated Kalman Filter

To perform a feature selection using the BSKF algorithm, the following main steps (as shown in Figure 4.7) must be proceeded. The detailed explanation of the steps is as follows:

Step (1) – Generate an initial population:

- i. Initialize all the SKF parameters (i.e., the initial value of error covariance estimate, $P(0)$, the process noise value, Q , and the measurement noise value, R are required. Further settings, such as, the number of n agents and a maximum number of iterations, t_{\max}).
- ii. Initialize randomly within a specific interval the 16 states values of each agent.
- iii. Initialize randomly 16-bit string, Σ_i^d , for every agent in a population.

Step (2) – Calculation of the fitness evaluation function for each agent:

- i. Using the generated 16-bit string, Σ_i^d , for every agent in a population to select a feature set.
- ii. Prepare the training and validation sets based on the selected feature set.
- iii. Then, train the NNRW classifier using the training set.

- iv. Use the validation set with the trained NNRW and compute the corresponding fitness evaluation value of validation set for every agent.

Step (3) – Update X_{best} and X_{true} :

- i. In the current iteration, set the best fitness evaluation value among agents, $\max_{i \in 1, \dots, n} \text{fit}(X(t))$ and update X_{best} .
- ii. Compare the $X_{\text{best}}(t)$ at the current iteration with X_{true} . X_{true} is only updated when $X_{\text{best}}(t)$ at current iteration is greater than X_{true} .

Step (4) – The state prediction, measurement, and estimation:

- i. Calculate the state prediction using Equation (4.1 and Equation (4.2.
- ii. Calculate the state measurement using Equation (4.3 and Equation (4.4.
- iii. Calculate the state estimation using Equation (4.5 and Equation (4.6.

Step (5) – The conversion of solution to the bit string for each agent:

- i. Convert and update the 16-bit string for each agent using Equation (4.8 to Equation (4.10.

Step (6) – Repeat the Step (2) to Step (5) until reached the number of maximum iterations.

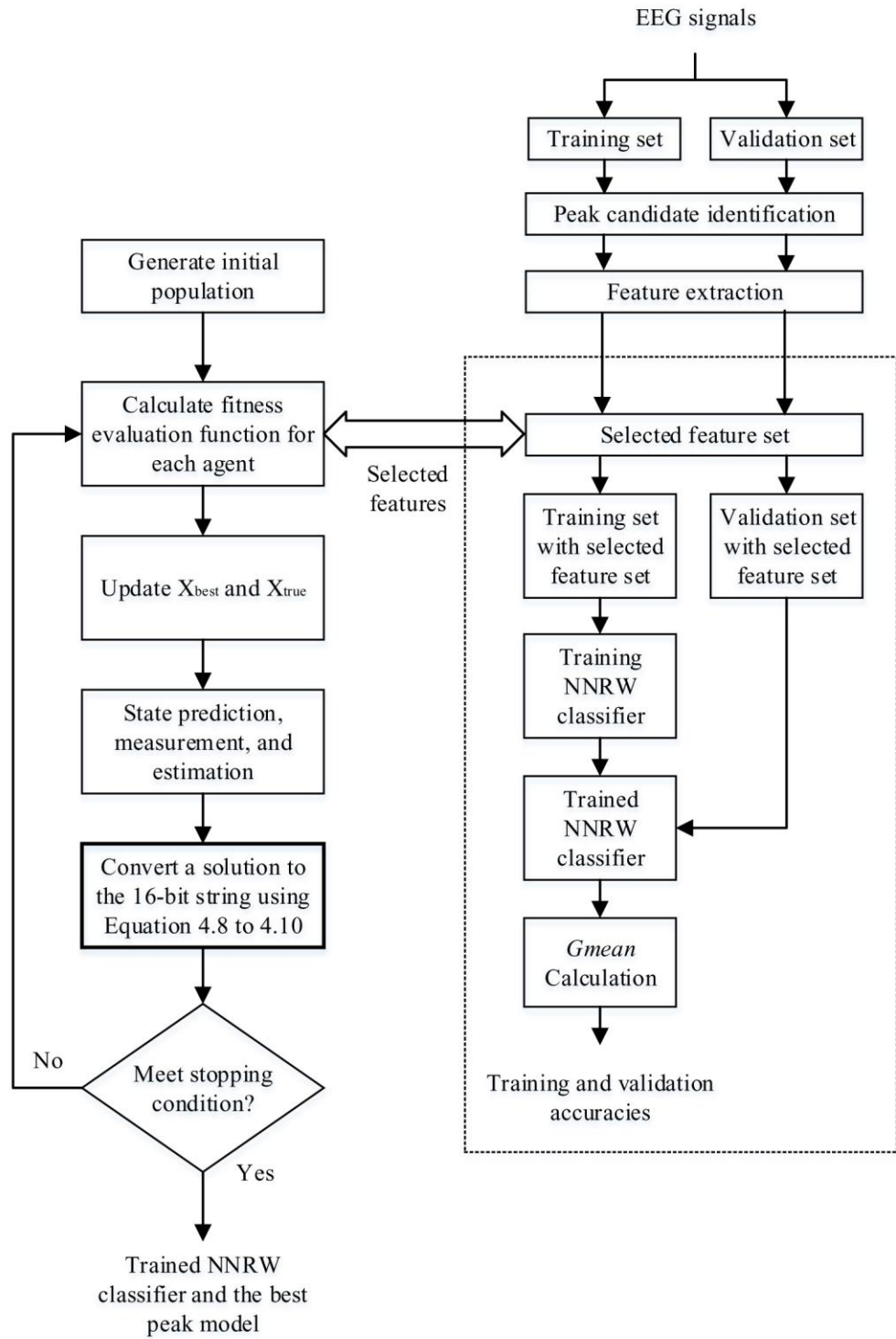


Figure 4.7: Flowchart of the proposed BSKF feature selection algorithm

4.6.3 Experiment 2: To Find a Generalized Model based on BSKF Feature Selection Approach

In this experiment, the BSKF algorithm is employed as a feature selector to find the best combination of peak features. The initial parameters setting for this experiment were set similar to the previous experiment. The four-fold cross validation with 30 runs is performed to analyze the EEG data of the proposed BSKF algorithm.

4.6.3.1 Experimental Results and Discussions

Table 4.9 demonstrates the experimental results of 30 runs using BSKF technique. The best testing accuracy among the four groups of each run indicates the best combination of peak features that searched by the BSKF. The training and validation accuracies of the best testing accuracy are also tabulated in Table 4.9.

From Table 4.9, it is found that the best peak model over 30 runs is $f_1, f_4, f_7, f_9, f_{11}, f_{13}$, and f_{16} , with 72.9% of testing accuracy. From those associated features, two of features are peak amplitudes (e.g., f_1 and f_4), three of features are peak widths (e.g., f_7, f_9 , and f_{11}), and two of features are peak slopes (e.g., f_{13} , and f_{16}). For overall of testing accuracy, the average, maximum, minimum, and STDEV over 30 runs are 65.1%, 72.9%, 57.3%, and 4.2%, respectively.

Figure 4.8 shows the example of convergence curve of the BSKF feature selection. This experiment was set 500 of the maximum iteration. The example of convergence curve demonstrates that the BSKF algorithm can reach convergence within 140 iterations.

The effectiveness of the BSKF technique and the best peak model can be measured by comparing with the existing models. As can be seen in Table 4.10, the classification performance of testing for full feature set, Dumpala, Acir, Liu, and Dingle models are 49.4%, 51.5%, 52.2%, 48.2%, and 40.1%, respectively. The testing performance of the BSKF model achieves 72.9%, with more than 20% of accuracy better than the five existing peak models.

Table 4.9: Best results over 30 runs using BSKF feature selection algorithm

Run	Training (%)	Validation (%)	Testing (%)	Best peak model											
1	93.40	66.16	70.78	3	4	10	11	13	15	16					
2	91.89	67.62	62.12	1	6	13	14	16							
3	95.35	65.69	66.43	1	4	7	13	15							
4	96.45	68.17	64.26	1	4	6	15	16							
5	89.44	66.17	72.57	4	5	9	14	16							
6	91.09	73.88	57.31	3	4	15									
7	86.27	66.76	57.43	4	6	8	14								
8	94.41	69.29	68.31	1	3	7	13	14							
9	92.25	70.13	64.40	1	3	4	9	15	16						
10	92.18	64.13	62.72	3	5	7	10	13	14						
11	90.45	70.70	67.55	3	9	14	16								
12	95.59	73.86	68.56	2	3	9	10	13							
13	92.14	69.34	61.98	1	7	9	10	13	14	15	16				
14	86.95	65.70	68.23	1	3	5	15								
15	92.58	69.87	60.05	2	3	6	10	13	15	16					
16	89.71	65.95	72.00	3	10	15	16								
17	84.91	65.51	62.23	3	12										
18	92.72	72.58	62.17	1	10	11	13								
19	93.86	65.04	68.27	3	4	15									
20	92.74	71.52	60.36	2	3	7	10	14	15						
21	91.77	68.76	64.35	2	10	14	15								
22	94.28	63.28	66.88	1	4	5	9	15							
23	88.47	62.63	60.27	1	2	7	8	9	10	11	12	14	15		
24	93.25	70.86	64.16	1	6	9	11	13	15	16					
25	95.45	63.70	66.50	3	4	11	13	14	16						
26	86.99	69.47	66.37	2	10	13	16								
27	96.45	66.21	72.88	1	4	7	9	11	13	16					
28	91.82	70.42	61.38	4	9	13	15	16							
29	89.19	74.07	66.17	2	7	15									
30	90.55	73.85	65.70	1	3	9	14	15							

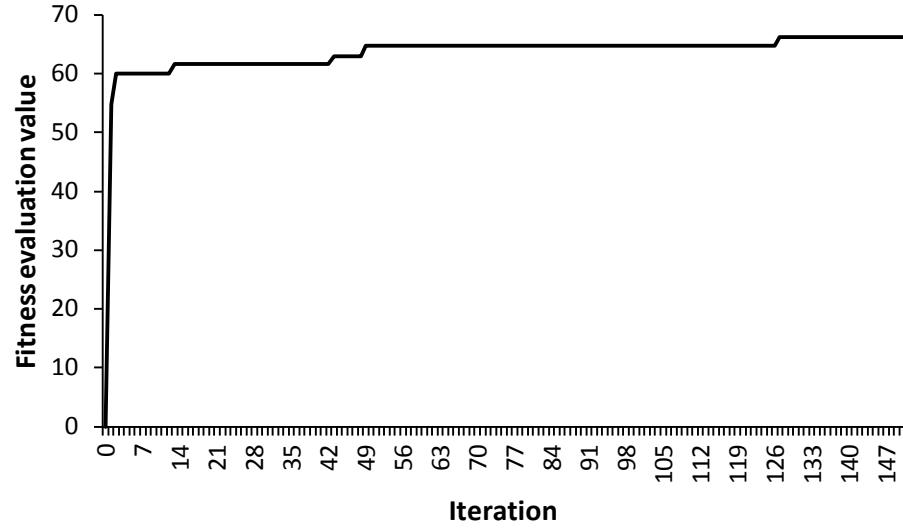


Figure 4.8: Example of the convergence curve of BSKF

Table 4.10: Comparison of the classification accuracy between the existing peak models and BSKF model

Peak model	Feature subset length	Selected features	Training Accuracy (%)	Testing Accuracy (%)
Full feature set	16	$f_1, f_2, f_3, f_4, f_5, f_6, f_7, f_8, f_9, f_{10}, f_{11}, f_{12}, f_{13}, f_{14}, f_{15}, f_{16}$	73.8	49.4
Dumpala	4	f_1, f_6, f_{13}, f_{14}	80.9	51.5
Acir	6	$f_1, f_2, f_7, f_8, f_{13}, f_{14}$	76.3	52.2
Liu	11	$f_1, f_2, f_3, f_4, f_6, f_9, f_{12}, f_{13}, f_{14}, f_{15}, f_{16}$	77.2	48.2
Dingle	4	f_5, f_6, f_{13}, f_{14}	71.4	40.1
BSKF (proposed work)	8	$f_1, f_4, f_7, f_9, f_{11}, f_{12}, f_{13}, f_{16}$	96.5	72.9

4.6.3.2 Statistical Significance Analysis

The detection algorithm with the existing five different peak models and the best model of BSKF feature selector are further analyzed by using nonparametric Friedman statistical analysis. The statistical analysis is required to demonstrate the significant difference in testing accuracy in terms of average value for the algorithm with the existing five models and the best model of BSKF technique. Similar to the previous

experiment, the experiments were conducted on statistical procedures designed especially for multiple $N \times N$ comparisons with six models executed in the KEEL data mining system (Alcala-Fdez, et al., 2009).

Table 4.11 presents the average ranking of Friedman's test of the existing six different models classification performance: full feature set, Dumpala, Acir, Liu, Dingle, and BSKF models. Referring to Table 4.11, the lowest average ranking is obtained by BSKF model that represents ranking first among the six models. While, the algorithm with Acir model ranking second, the algorithm with Dumpala model ranking third, the algorithm with full feature set model ranking fourth, the algorithm with Liu model ranking fifth, and the algorithm with Dingle model ranking sixth.

Additionally, p -values for unadjusted values and adjusted p -values for Nemenyi, Holm, Shaffer, and Bergmann-Hommel test for $N \times N$ comparisons for all possible 15 pairs of the algorithm with the peak models are presented in Table 4.12. The p -values below 0.05 represent that the particular algorithm with the peak model differs significantly in testing accuracy. The p -values below 0.05 were marked with the italic font.

As can be seen in Table 4.12, the p -values for unadjusted values and Bergmann-Hommel offer for eliminating eleven hypotheses whereas Holm and Shaffer ones reject only ten. However, Nemenyi eliminates only nine hypotheses. Based on unadjusted p -values and adjusted p -values for Nemenyi, Holm, Shaffer, and Bergmann-Hommel test, the AMSKF model revealed significantly better performance than other existing peak models.

Table 4.11: The average ranking of the full feature set, Dumpala, Acir, Liu, Dingle, and BSKF, achieved by Friedman

Peak model	Average ranking	Rank
Dingle	5.867	6
Liu	4.3	5
Full feature set	3.833	4
Dumpala	3.1	3
Acir	2.9	2
BSKF (proposed work)	1	1
Statistic	112.4762	
<i>p</i> -value	5.385E-11	

Table 4.12: Adjusted p-value for $N \times N$ comparisons of algorithms over 30 runs for BSKF

Peak model vs. Peak model	pUnadj	pNeme	pHolm	pShaf	pBerg
Dingle vs. BSKF	0	0	0	0	0
Liu vs. BSKF	0	0	0	0	0
Acir vs. Dingle	0	0	0	0	0
full feature set vs. BSKF	0	0	0	0	0
Dumpala vs. Dingle	0	0	0	0	0
Dumpala vs BSKF	0.000014	0.000207	0.000138	0.000138	0.000083
full feature set vs. Dingle	0.000026	0.000384	0.00023	0.000179	0.000102
Acir vs. BSKF	0.000084	0.001256	0.00067	0.000586	0.000335
Liu vs. Dingle	0.001182	0.017723	0.008271	0.008271	0.004726
Acir vs. Liu	0.003752	0.056283	0.022513	0.022513	0.022513
Dumpala vs. Liu	0.0129833	0.194745	0.064915	0.051932	0.038949
full feature set vs. Acir	0.053337	0.800053	0.213347	0.213347	0.160011
full feature set vs. Dumpala	0.128978	1.934671	0.386934	0.386934	0.160011
full feature set vs. Liu	0.333998	5.009974	0.667997	0.667997	0.667997
Dumpala vs. Acir	0.678845	10.182679	0.678845	0.678845	0.678845

4.7 The Proposed Generalized Model III based on Local Optimum Distance Evaluated Simulated Kalman Filter

The purpose of this section is to explain the process flow of the proposed feature selection technique using LocalDESKF algorithm, in which to produce a new generalized model III. This section consists of three sub-sections: (1) a detailed description of LocalDESKF algorithm, (2) a feature selection method using LocalDESKF algorithm, and (3) the experiments to find a generalized model based on LocalDESKF feature selection approach.

4.7.1 Local Optimum Distance Evaluated Simulated Kalman Filter

Another variant of SKF algorithm for solving discrete optimization problem is LocalDESKF (Md Yusof, Ibrahim, Ibrahim, Abd Aziz, & Ab Aziz, 2016). The LocalDESKF is the modified version of the BSKF algorithm. The process flow of the LocalDESKF is similar to the BSKF, as explained in Sub-section 4.6.1. The stages of the algorithms including: (1) generate an initial population, (2) calculation of the fitness evaluation function for each agent, (3) update the best fitness value among agents at every iteration (X_{best}) and the best solution compared to the current X_{best} (X_{true}), (4) perform state prediction, measurement, and estimation, (5) convert a new solution to a bit string, and (6) perform termination based on a stopping criterion. The only one different between the BSKF and the LocalDESKF is the formulation in the conversion stage in which to alter a new solution to a bit string for each agent.

In the conversion stage, the BSKF employs the probabilistic value of a new solution from the multiplication of the Kalman gain, $K(t)$, and measurement value, $Z_i(t)$, to update a bit string, as formulated in Equation (4.8, Equation (4.9, and Equation (4.10. While the LocalDESKF uses the probabilistic value of the distance between the state

value of i th agent and the best-so-far solution. In the LocalDESKF algorithm, the best-so-far solution is X_{best} .

After completion of updating X_{best} and X_{true} and calculation of the state prediction, state measurement, and state estimation, the conversion stage of the LocalDESKF begins with the following equations:

$$X_{\text{best-so-far}}^d = X_{\text{best}}^d \quad (4.11)$$

$$D_i^d(t) = x_i^d(t) - X_{\text{best-so-far}}^d(t) \quad (4.12)$$

$$S(D_i^d(t)) = |\tanh D_i^d(t)| \quad (4.13)$$

$$\begin{aligned} & \text{if } rand < S(D_i^d(t)) \\ & \quad \text{then } \Sigma_i^d(t+1) = \text{complement } \Sigma_i^d(t) \\ & \quad \text{else } \Sigma_i^d(t+1) = \Sigma_i^d(t) \\ & \text{end} \end{aligned} \quad (4.14)$$

Then, the LocalDESKF algorithm proceeds to the next iteration until the maximum iterations are reached.

4.7.2 Feature Selection using LocalDESKF

The process flow of feature selection using the LocalDESKF algorithm is similar to the BSKF feature selector, as illustrated in Figure 4.7. As mentioned in the previous sub-section, the only one different between BSKF and LocalDESKF is the formulation of conversion a solution to a bit string.

Similar to the BSKF in Sub-section 4.6.2, the LocalDESKF feature selector consists of six stages. All the six stages including: (1) generate an initial population, (2) calculation of the fitness evaluation function for each agent, (3) update X_{best} and X_{true} ,

(4) the state prediction, measurement, and estimation, (5) the conversion of solution to the bit string for each agent, and (6) repeat the stage (2) to stage (5) until the maximum iterations are reached. In LocalDESKF feature selection method, all the processes in each stage are similar to the BSKF except the stage (5), in which the process to update a new 16-bit string. In stage (5), the 16-bit string for each agent is converted and updated using Equation (4.11 to Equation (4.14). The flowchart of the proposed LocalDESKF feature selection algorithm is illustrated in Figure 4.9.

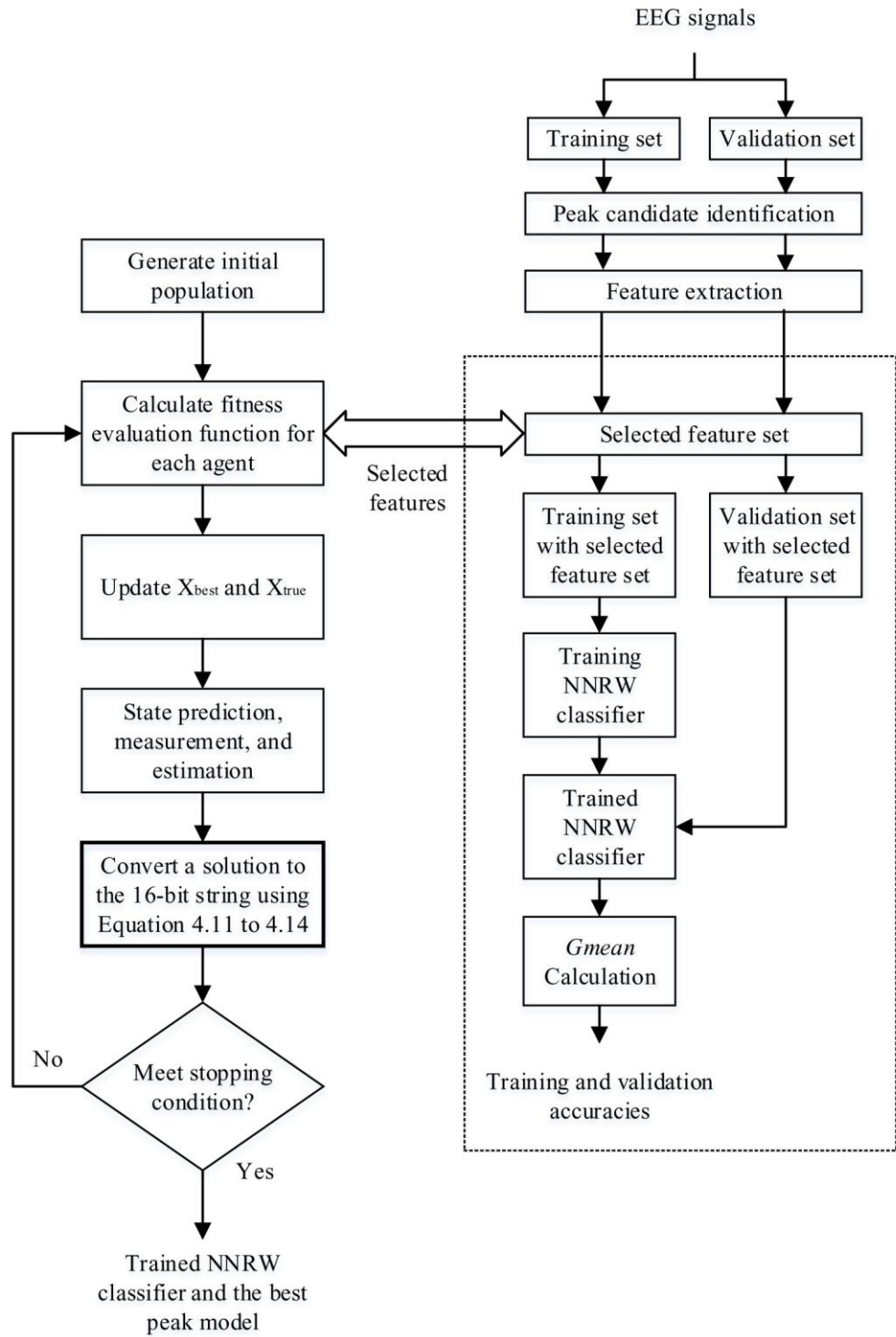


Figure 4.9: Flowchart of the proposed LocalDESKF feature selection algorithm

4.7.3 Experiment 3: To Find a Generalized Model based on LocalDESKF-NNRW Feature Selection Approach

In this experiment, the LocalDESKF algorithm is used as a feature selector to find the best combination of peak features. The initial parameters setting for this experiment were set similar to the previous experiment. The four-fold cross validation with 30 runs is performed to analyze the EEG data of the proposed LocalDESKF algorithm.

4.7.3.1 Experimental Results and Discussions

Table 4.13 shows the best testing result of the four-fold cross validation process that repeated 30 times using the LocalDESKF feature selection method. The best testing result indicates the maximum classification performance of testing data among four groups of data. The training and validation accuracies of the best testing result are also recorded and tabulated in the table.

In Table 4.13, it can be seen that the peak model with three associate peak features: f_1 , f_{11} , and f_{16} , obtains 72.3% of accuracy, in which it the best classification performance using the LocalDESKF approach. From the three associated features, f_1 is a peak-to-peak amplitude of the first half wave, f_{11} is a second half wave turning point width, and f_{16} is a turning point slope at the second half wave. For overall of testing accuracy, the average, maximum, minimum, and STDEV over 30 runs are 64.3%, 72.3%, 52.6%, and 4.4%, respectively.

In this experiment, the proposed LocalDESKF feature selector was initially set the maximum 500 iterations. One example is taken from this experiment to observe the result of the convergence of the proposed LocalDESKF, as shown in Figure 4.10. In Figure 4.10, it can be observed that the LocalDESKF method can reach convergence within 50 iterations.

To measure the effectiveness of the best peak model of the LocalDESKF technique, the classification performance of the existing models and the best model are compared. The comparisons of testing results are tabulated in Table 4.14. In Table 4.14, the classification performance of testing for the full feature set, Dumpala, Acir, Liu, and Dingle models are 49.4%, 51.5%, 52.2%, 48.2%, and 40.1%, respectively. The testing performance of the LocalDESKF model achieves 72.9%, with more than 20% of accuracy better than the five existing peak models.

Table 4.13: Best results over 30 runs using LocalDESKF feature selection algorithm

Run	Training (%)	Validation (%)	Testing (%)	Best peak model															
1	92.58	71.33	60.24	1	4	6	8	9	13	16									
2	88.53	70.78	61.46	2	4	9	15												
3	98.80	71.28	71.03	1	3	4	9	10	11	12	14	15							
4	93.40	70.04	60.32	1	2	3	8	9	11	13	14	15							
5	83.53	73.91	63.42	7	13	15	16												
6	91.11	73.35	68.35	1	5	13	14												
7	91.28	72.45	64.51	1	4	11	14	16											
8	78.68	72.68	70.21	10	13	16													
9	89.44	67.94	70.63	2	3	10	15	16											
10	88.35	69.06	70.87	2	3	6	9	10	14	15	16								
11	88.64	70.96	64.80	1	2	4	6	7	11	15									
12	92.58	67.89	59.68	4	5	9													
13	93.70	71.45	63.56	1	3	9	14	16											
14	93.09	56.27	60.59	1	2	5	9	11	12	13	14								
15	92.42	71.69	60.32	1	2	4	9	10	11	12	13	15	16						
16	93.37	70.66	60.45	1	3	4	6	7	8	9	12	13	14						
17	89.16	71.49	52.59	1	3	4	6	9	16										
18	90.05	66.52	62.83	1	4	5	6	7	9	11	13	14	15						
19	94.28	67.10	65.74	3	9	16													
20	91.70	70.91	64.98	1	2	5	10	13	15	16									
21	83.94	70.37	60.06	1	6	7	9	13	14	15	16								
22	92.58	66.06	61.57	2	9	13													
23	93.09	69.78	64.71	1	2	3	6	9	10	13	14	16							
24	94.28	69.84	64.07	2	3	9	15												
25	88.85	64.99	68.35	4	7	15													
26	95.06	69.27	72.26	1	11	16													
27	89.24	73.44	66.38	1	4														
28	92.76	68.45	66.10	4	9	12	13												
29	87.29	64.28	66.67	5	7	10	13	16											
30	88.32	71.15	62.26	2	3	6	13	15											

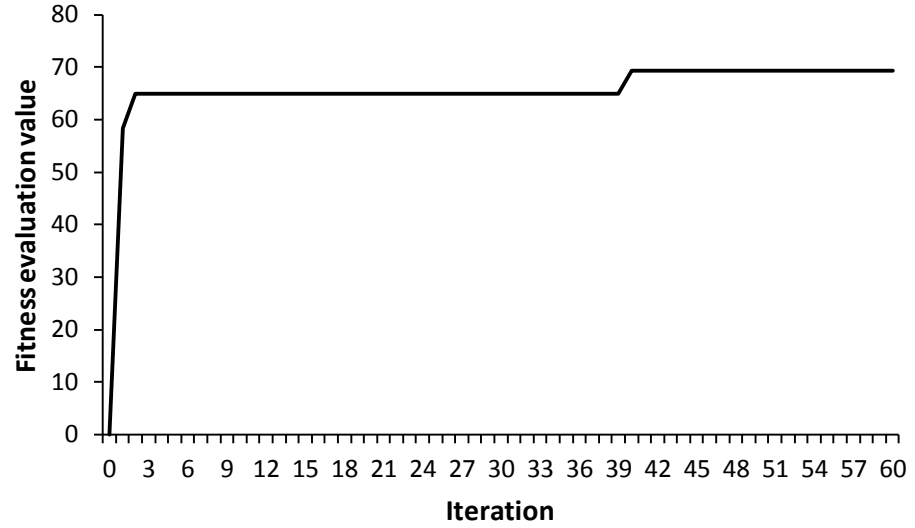


Figure 4.10: Example of the convergence curve of LocalDESKF

Table 4.14: Comparison of the classification accuracy between the existing peak models and LocalDESKF model

Peak model	Feature subset length	Selected features	Training Accuracy (%)	Testing Accuracy (%)
Full feature set	16	$f_1, f_2, f_3, f_4, f_5, f_6, f_7, f_8, f_9, f_{10}, f_{11}, f_{12}, f_{13}, f_{14}, f_{15}, f_{16}$	73.8	49.4
Dumpala	4	f_1, f_6, f_{13}, f_{14}	80.9	51.5
Acir	6	$f_1, f_2, f_7, f_8, f_{13}, f_{14}$	76.3	52.2
Liu	11	$f_1, f_2, f_3, f_4, f_6, f_9, f_{12}, f_{13}, f_{14}, f_{15}, f_{16}$	77.2	48.2
Dingle	4	f_5, f_6, f_{13}, f_{14}	71.4	40.1
LocalDESKF (proposed work)	3	f_1, f_{11}, f_{16}	95.06	72.3

4.7.3.2 Statistical Significance Analysis

The existing five different peak models and the best model of the LocalDESKF feature selector are further analyzed by using nonparametric Friedman statistical analysis. The statistical analysis is required to demonstrate the significant difference in testing accuracy in terms of average value for the existing five models and the best

model of the LocalDESKF technique. Similar to the previous experiment, the experiments were conducted on statistical procedures designed especially for multiple $N \times N$ comparisons with six models executed in the KEEL data mining system (Alcala-Fdez, et al., 2009).

Table 4.15 shows the average ranking of Friedman's test of the existing six different models classification performance: full feature set, Dumpala, Acir, Liu, Dingle, and LocalDESKF models. From Table 4.15, the lowest average ranking is obtained by the LocalDESKF model that represents ranking first among the six models. While, the algorithm with Acir model ranking second, the algorithm with Dumpala model ranking third, the algorithm with full feature set model ranking fourth, the algorithm with Liu model ranking fifth, and the algorithm with Dingle model ranking sixth.

Table 4.16 presents p -values for unadjusted values and adjusted p -values for Nemenyi, Holm, Shaffer, and Bergmann-Hommel test for $N \times N$ comparisons for all possible 15 pairs of the algorithm with the peak models. The p -values below 0.05 represent that the particular algorithm with the peak model differs significantly in testing accuracy. The p -values below 0.05 were marked with the italic font.

Referring to Table 4.16, the p -values for unadjusted values offers for eliminating twelve hypotheses whereas the p -values for adjusted p -values for Holm and Shaffer ones reject only ten. However, Nemenyi and Bergmann-Hommel eliminates only nine and eleven hypotheses, respectively. Based on unadjusted p -values and adjusted p -values for Nemenyi, Holm, Shaffer, and Bergmann-Hommel test, the LocalDESKF model revealed significantly better performance than other existing peak models.

Table 4.15: The average ranking of the full feature set, Dumpala, Acir, Liu, Dingle, and LocalDESKF, achieved by Friedman

Peak model	Average ranking	Rank
Dingle	5.8667	6
Liu	4.3	5
Full feature set	3.8333	4
Dumpala	3.0667	3
Acir	2.8667	2
LocalDESKF (proposed work)	1.0667	1
Statistic	110.2476	
<i>p</i> -value	7.13399E-11	

Table 4.16: Adjusted *p*-value for $N \times N$ comparisons of algorithms over 30 runs for LocalDESKF

Peak model vs. Peak model	pUnadj	pNeme	pHolm	pShaf	pBerg
Dingle vs. LocalDESKF	0	0	0	0	0
Liu vs. LocalDESKF	0	0	0	0	0
Acir vs. Dingle	0	0	0	0	0
Dumpala vs. Dingle	0	0	0	0	0
full feature set vs. LocalDESKF	0	0	0	0	0
full feature set vs. Dingle	0.000026	0.000384	0.000256	0.000256	0.000154
Dumpala vs LocalDESKF	0.000035	0.00052	0.000312	0.000256	0.000154
Acir vs. LocalDESKF	0.000194	0.002914	0.001554	0.00136	0.000777
Liu vs. Dingle	0.001182	0.017723	0.008271	0.008271	0.004726
Acir vs. Liu	0.003004	0.056283	0.022513	0.022513	0.022513
Dumpala vs. Liu	0.010673	0.194745	0.064915	0.051932	0.038949
full feature set vs. Acir	0.045372	0.800053	0.213347	0.213347	0.160011
full feature set vs. Dumpala	0.112478	1.687177	0.337435	0.337435	0.136116
full feature set vs. Liu	0.333998	5.009974	0.667997	0.667997	0.667997
Dumpala vs. Acir	0.678845	10.182679	0.678845	0.678845	0.678845

4.8 The Proposed Generalized Model IV: Global Optimum Distance Evaluated Simulated Kalman Filter

The purpose of this section is to explain the process flow of the proposed feature selection technique using GlobalDESKF algorithm, in which to produce a new generalized model IV. This section consists of three sub-sections: (1) a detailed description of GlobalDESKF algorithm, (2) a feature selection method using GlobalDESKF algorithm, and (3) the experiments to find a generalized model based on GlobalDESKF feature selection approach.

4.8.1 Global Optimum Distance Evaluated Simulated Kalman Filter

Another variant of SKF algorithm for solving discrete optimization problem is GlobalDESKF (Md Yusof, et al., 2016b). The GlobalDESKF is the modified version of the LocalDESKF algorithm. The process flow of the GlobalDESKF is similar to the BSKF and LocalDESKF, as explained in Sub-section 4.6.1 and 4.7.1, respectively. The stages of the algorithms including: (1) generate an initial population, (2) calculation of the fitness evaluation function for each agent, (3) update the best fitness value among agents at every iteration (X_{best}) and the best solution compared to the current X_{best} (X_{true}), (4) perform state prediction, measurement, and estimation, (5) convert a new solution to a bit string, and (6) perform termination based on a stopping criterion. The only one difference between the BSKF, LocalDESKF and GlobalDESKF is the formulation in the conversion stage in which to alter a new solution to a bit string for each agent.

In the conversion stage, the BSKF employs the probabilistic value of a new solution from the multiplication of the Kalman gain, $K(t)$, and measurement value, $Z_i(t)$, to update a bit string, as formulated in Equation (4.8, Equation (4.9, and Equation (4.10.

The LocalDESKF and GlobalDESKF employ the probabilistic value of the distance between the state value of i th agent and the best-so-far solution. In the LocalDESKF algorithm, the best-so-far solution is X_{best} . While, in the GlobalDESKF algorithm, the best-so-far solution is X_{true} .

After completion of updating X_{best} and X_{true} and calculation of the state prediction, state measurement, and state estimation, the conversion stage of the GlobalDESKF begins with the following equations:

$$X_{\text{best-so-far}}^d = X_{\text{true}} \quad (4.15)$$

$$D_i^d(t) = x_i^d(t) - X_{\text{best-so-far}}^d(t) \quad (4.16)$$

$$S(D_i^d(t)) = |\tanh D_i^d(t)| \quad (4.17)$$

$$\begin{aligned} & \text{if } rand < S(D_i^d(t)) \\ & \quad \text{then } \Sigma_i^d(t+1) = \text{complement } \Sigma_i^d(t) \\ & \quad \text{else } \Sigma_i^d(t+1) = \Sigma_i^d(t) \\ & \text{end} \end{aligned} \quad (4.18)$$

Then, the GlobalDESKF algorithm proceeds to the next iteration until the maximum iterations are reached.

4.8.2 Feature Selection using GlobalDESKF

The process flow of feature selection using the GlobalDESKF algorithm is similar to the BSKF and LocalDESKF feature selectors, as illustrated in Figure 4.7. As mentioned in the previous sub-section, the only one difference between BSKF, LocalDESKF, and GlobalDESKF is the formulation of conversion a solution to a bit string.

Similar to the LocalDESKF in Sub-section 4.7.2, the GlobalDESKF feature selector consists of six stages. All the six stages including: (1) generate an initial population, (2) calculation of the fitness evaluation function for each agent, (3) update X_{best} and X_{true} , (4) the state prediction, measurement, and estimation, (5) the conversion of solution to the bit string for each agent, and (6) repeat the stage (2) to stage (5) until the maximum iterations are reached. In GlobalDESKF feature selection method, all the processes in each stage are similar to the BSKF and LocalDESKF except the stage (5), in which the process to update a new 16-bit string. In stage (5), the 16-bit string for each agent is converted and updated using Equation (4.15 to Equation (4.18). The flowchart of the proposed LocalDESKF feature selection algorithm is illustrated in Figure 4.11.

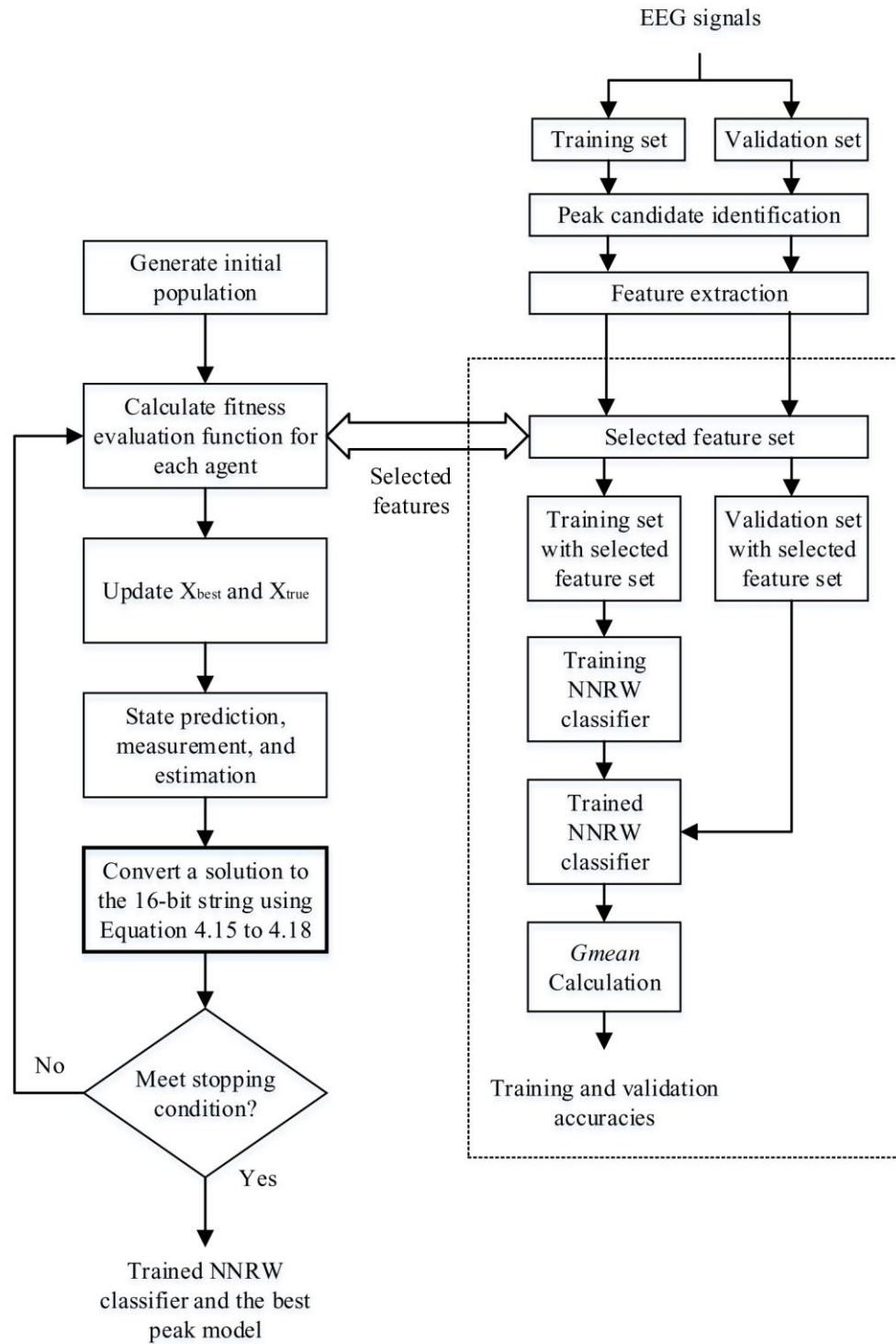


Figure 4.11: Flowchart of the proposed GlobalDESKF feature selection algorithm

4.8.3 Experiment 4: To Find a Generalized Model IV based on GlobalDESKF Feature Selection Approach

In this experiment, the GlobalDESKF algorithm is employed to be a feature selector to find the best combination of peak features. The initial parameters setting of the NNRW and GlobalDESKF for this experiment were set similar to the previous experiment. The four-fold cross validation with 30 runs is performed to analyze the EEG data of the proposed GlobalDESKF feature selection algorithm.

4.8.3.1 Experimental Results and Discussions

Table 4.17 presents the experimental results based on 30 times repeated of four-fold cross validation process using the GlobalDESKF feature selection technique. The results of every four-fold cross validation process indicate the maximum testing accuracy and the best combination of features among the four groups of data. The training and validation accuracies of the best testing result are also recorded and tabulated in the table.

From Table 4.17, it can be shown that the peak model with three associate peak features: f_1 , f_9 , and f_{16} , obtains 73.8% of accuracy, in which it the best classification performance using the GlobalDESKF technique. From the three associated features, f_1 is a peak-to-peak amplitude of the first half wave, f_9 is a turning point width, and f_{16} is a turning point slope at the second half wave. For overall of testing accuracy, the average, maximum, minimum, and STDEV over 30 runs are 64.4%, 73.8%, 55.3%, and 5.3%, respectively.

In this experiment, the proposed GlobalDESKF feature selector was initially set the maximum 500 iterations. One example is taken from this experiment to observe the result of convergence of the proposed GlobalDESKF, as shown in Figure 4.12. From

Figure 4.12, it can be seen that the GlobalDESKF method can reach convergence within 60 iterations.

To measure the effectiveness of the best peak model of the GlobalDESKF technique, the classification performance of the existing models and the best model are compared. The comparisons of testing results are tabulated in Table 4.18. In Table 4.18, the classification performance of testing for the full feature set, Dumpala, Acir, Liu, and Dingle models are 49.4%, 51.5%, 52.2%, 48.2%, and 40.1%, respectively. The testing performance of the GlobalDESKF model achieves 73.8%, with more than 20% of accuracy better than the five existing peak models.

Table 4.17: Best results over 30 runs using GlobalDESKF feature selection algorithm

Run	Training (%)	Validation (%)	Testing (%)	Best peak model											
1	95.92	68.20	55.33	1	3	10	11	16							
2	89.44	77.84	73.79	1	9	16									
3	88.35	71.16	60.21	1	10	11	13								
4	90.83	70.60	62.49	4	7	11	13	16							
5	93.09	67.62	68.24	5	9	13	15								
6	91.06	66.95	59.59	1	8	15									
7	88.19	69.64	66.00	2	9	13	15	16							
8	84.10	80.50	67.48	4	9	13	14	15							
9	91.65	67.81	58.27	1	4	5	8	9	15	16					
10	90.89	67.13	70.46	1	3	4	9	15							
11	93.37	67.96	72.27	3	4	6	9	14	15	16					
12	86.28	67.80	59.69	5	8	13									
13	88.47	71.60	57.67	1	6	7	8	9	13	14	15				
14	88.92	67.90	73.17	1	4	6	10	11	13	14					
15	90.45	65.70	59.64	4	7	13	16								
16	93.09	69.01	73.53	3	10	16									
17	95.97	71.58	64.50	1	5	13	16								
18	94.41	68.43	64.98	1	2	3	4	7	8	9	13	16			
19	96.08	65.57	57.61	3	4	5	9	15	16						
20	87.90	68.04	59.50	3	6	13	14								
21	92.25	77.90	64.23	1	9	15	16								
22	92.42	68.99	64.53	1	8	10	15								
23	94.87	74.01	59.71	1	9	11	15								
24	96.08	73.61	68.53	3	11	13	14	15							
25	89.44	69.26	67.44	3	13	16									
26	95.65	76.99	59.60	1	2	3	9	14							
27	89.69	74.07	61.77	3	7	14	15								
28	85.97	66.36	68.42	2	6	12	14	15	16						
29	91.91	64.34	66.98	1	2	5	7	9	11	12	15	16			
30	86.90	68.45	67.68	10	14	15	16								

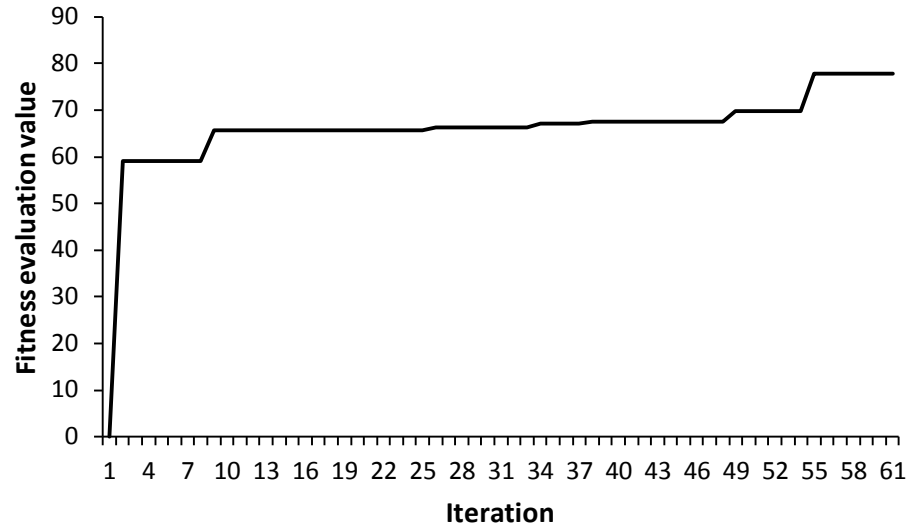


Figure 4.12: Example of the convergence curve of GlobalDESKF

Table 4.18: Comparison of the classification accuracy between the existing peak models and GlobalDESKF model

Peak model	Feature subset length	Selected features	Training Accuracy (%)	Testing Accuracy (%)
Full feature set	16	$f_1, f_2, f_3, f_4, f_5, f_6, f_7, f_8, f_9, f_{10}, f_{11}, f_{12}, f_{13}, f_{14}, f_{15}, f_{16}$	73.8	49.4
Dumpala	4	f_1, f_6, f_{13}, f_{14}	80.9	51.5
Acir	6	$f_1, f_2, f_7, f_8, f_{13}, f_{14}$	76.3	52.2
Liu	11	$f_1, f_2, f_3, f_4, f_6, f_9, f_{12}, f_{13}, f_{14}, f_{15}, f_{16}$	77.2	48.2
Dingle	4	f_5, f_6, f_{13}, f_{14}	71.4	40.1
GlobalDESKF (proposed work)	3	f_1, f_9, f_{16}	89.4	73.8

4.8.3.2 Statistical Significance Analysis

The existing five different peak models and the best model of the GlobalDESKF feature selector are further analyzed by using nonparametric Friedman statistical analysis. The statistical analysis is required to demonstrate the significant difference in testing accuracy in terms of average value for the existing five models and the best model of the GlobalDESKF technique. Similar to the previous experiment, the

experiments were conducted on statistical procedures designed especially for multiple $N \times N$ comparisons with six models executed in the KEEL data mining system (Alcala-Fdez, et al., 2009).

Table 4.19 shows the average ranking of Friedman's test of the existing six different models classification performance: full feature set, Dumpala, Acir, Liu, Dingle, and GlobalDESKF models. From Table 4.19, the lowest average ranking is obtained by the GlobalDESKF model that represents ranking first among the six models. While, the algorithm with Acir model ranking second, the algorithm with Dumpala model ranking third, the algorithm with full feature set model ranking fourth, the algorithm with Liu model ranking fifth, and the algorithm with Dingle model ranking sixth.

Table 4.20 presents p -values for unadjusted values and adjusted p -values for Nemenyi, Holm, Shaffer, and Bergmann-Hommel test for $N \times N$ comparisons for all possible 15 pairs of the algorithm with the peak models. The p -values below 0.05 represent that the particular algorithm with the peak model differs significantly in testing accuracy. The p -values below 0.05 were marked with the italic font.

Referring to Table 4.20, the p -values for unadjusted values offers for eliminating eleven hypotheses. The p -values for adjusted p -values for Bergmann-Hommel test also offers for eliminating eleven hypotheses. The p -values for adjusted p -values for Holm and Shaffer ones reject only ten. However, Nemenyi eliminates only nine. Based on unadjusted p -values and adjusted p -values for Nemenyi, Holm, Shaffer, and Bergmann-Hommel test, the GlobalDESKF model revealed significantly better performance than other existing peak models.

Table 4.19: The average ranking of the full feature set, Dumpala, Acir, Liu, Dingle, and GlobalDESKF, achieved by Friedman

Peak model	Average ranking	Rank
Dingle	5.8667	6
Liu	4.3	5
Full feature set	3.8333	4
Dumpala	3.1	3
Acir	2.9	2
GlobalDESKF (proposed work)	1	1
Statistic	112.47619	
<i>p</i> -value	5.388589E-11	

Table 4.20: Adjusted *p*-value for $N \times N$ comparisons of algorithms over 30 runs for GlobalDESKF

Peak model vs. Peak model	pUnadj	pNeme	pHolm	pShaf	pBerg
Dingle vs. GlobalDESKF	0	0	0	0	0
Liu vs. GlobalDESKF	0	0	0	0	0
Acir vs. Dingle	0	0	0	0	0
Full feature set vs. GlobalDESKF	0	0	0	0	0
Dumpala vs. Dingle	0	0	0	0	0
Dumpala vs GlobalDESKF	0.000014	0.000207	0.000138	0.000138	0.000083
Full feature set vs. Dingle	0.000026	0.000384	0.00023	0.000179	0.000102
Acir vs. GlobalDESKF	0.000084	0.001256	0.00067	0.00586	0.000335
Liu vs. Dingle	0.001182	0.017723	0.008271	0.008271	0.004726
Acir vs. Liu	0.003752	0.056283	0.022513	0.022513	0.022513
Dumpala vs. Liu	0.012983	0.194745	0.064915	0.051932	0.038949
Full feature set vs. Acir	0.053337	0.800053	0.213347	0.213347	0.160011
Full feature set vs. Dumpala	0.128978	1.934671	0.386934	0.386934	0.160011
Full feature set vs. Liu	0.333998	5.009974	0.667997	0.667997	0.667997
Dumpala vs. Acir	0.678845	10.182679	0.678845	0.678845	0.678845

4.9 Performance Comparison of the Proposed Generalized Models and the Existing Models

This section discusses the performance of the proposed four different generalized models and the existing models. The generalized models (AMSKF, BSKF, LocalDESKF, and GlobalDESKF models) are obtained by the four different feature selection processes, which are the AMSKF, BSKF, LocalDESKF, and GlobalDESKF algorithms. The existing models with the associated features including Dumpala, Acir, Liu, Dingle, and full feature set models are chosen based on the existing research works that have been discussed in Section 2.3.

Table 4.21 shows the performance comparison of the proposed generalized models (AMSKF, BSKF, LocalDESKF, and GlobalDESKF models) and the existing models (Dumpala, Acir, Liu, Dingle, and full feature set models). All the results in Table 4.21 are obtained from the previous experiments.

From Table 4.21, in term of testing accuracy, it is found that the GlobalDESKF model is the highest performance compared to other models, with 73.8%. Overall, the performances of the proposed all generalized models are 20% greater than the existing peak models. Among the peak models, the lowest testing accuracy is 40.1% from the Dingle model.

Non-parametric statistical analysis has been made using Friedman test between AMSKF, BSKF, LocalDESKF, and GlobalDESKF in order to demonstrate the significant difference in testing accuracy of the results. The result of the Friedman test indicates that there is no significant difference in testing accuracy between AMSKF, BSKF, LocalDESKF, and GlobalDESKF methods.

From Table 4.21, it can be seen that there is a large different value between training and testing accuracies. The proposed method of the GlobalDESKF model has only achieved 73.8% of testing accuracy. In this study, the ratio between true peaks and false peaks are 140:11461. That means the dataset has extremely imbalanced dataset ratio. In this case, the conventional NNRW classifier may fail to offer high accuracy of performance for imbalanced dataset problem. Other contributing factor is the collected data is affected by various noises and the peak features have a large different value from one subject to another subject. Moreover, the EEG data that was used in this study generalized to eye event-related signals which are the combination of three different event-related signals. (e.g., single eye blink, double eye blink, and eye movement signals). These factors are the cause to the high variation of peak features. The consequent of this factor is the NNRW classifier may fail to correctly classify the true peak and false peak.

The average computational time of each method is also recorded in Table 4.21. For the method that not involve feature selection technique (i.e. full feature set, Dumpala, Acir, and Liu models), they obtained an average computational time below than 10 minutes. The feature selection techniques (i.e. AMSKF, BSKF, Local DESKF, and GlobalDESKF algorithms) performed an average computational time around 150 minutes due to its involving the process of selection of the best peak model.

Table 4.21: Performance comparisons among peak models

Peak model	Feature subset length	Selected features	Training Accuracy (%)	Testing Accuracy (%)	Average Computational Time (minutes)
Full feature set	16	$f_1, f_2, f_3, f_4, f_5, f_6, f_7, f_8, f_9, f_{10}, f_{11}, f_{12}, f_{13}, f_{14}, f_{15}, f_{16}$	73.8	49.4	7
Dumpala	4	f_1, f_6, f_{13}, f_{14}	80.9	51.5	5
Acir	6	$f_1, f_2, f_7, f_8, f_{13}, f_{14}$	76.3	52.2	5
Liu	11	$f_1, f_2, f_3, f_4, f_6, f_9, f_{12}, f_{13}, f_{14}, f_{15}, f_{16}$	77.2	48.2	6
Dingle	4	f_5, f_6, f_{13}, f_{14}	71.4	40.1	5
AMSKF	11	$f_1, f_2, f_7, f_8, f_9, f_{10}, f_{11}, f_{12}, f_{13}, f_{14}, f_{15}$	91.8	72.7	150
BSKF	8	$f_1, f_4, f_7, f_9, f_{11}, f_{12}, f_{13}, f_{16}$	96.5	72.9	150
LocalDESKF	3	f_1, f_{11}, f_{16}	95.1	72.3	150
GlobalDESKF	3	f_1, f_9, f_{16}	89.4	73.8	150

4.10 Summary

This chapter introduced the implementation of the four different SKF algorithms for feature selection of EEG signals peak classification. The four variants of SKF algorithm include the AMSKF, BSKF, LocalDESKF, and GlobalDESKF, which are specially invented for discrete optimization problems.

All experiments in this chapter used 11781 peak candidate samples with their associated features, which are collected from the three different peak event-related EEG signals of 30 healthy subjects; 1) single eye blink, 2) double eye blink, and 3) eye movement signals.

The fundamental aspect of each algorithm is described in details in which the flow of the process and the technique to be implemented as a feature selector for EEG signals peak classification are discussed. Each experiment for the particular proposed algorithm in this chapter focused to find the best combination of peak features. The highest

performance of testing accuracy with the associated features is highlighted in the experimental result. Furthermore, the statistical Friedman test analysis is conducted for every algorithm to find the significance difference from the testing accuracy of the proposed model and the existing models.

The major finding of this chapter is that all the proposed generalized models perform better than the existing peak models. At the end of this chapter, the performances of the proposed generalized models are compared with the existing peak models based on the Gmean value of training and testing accuracy.

CHAPTER 5: APPLICATION OF THE PROPOSED PEAK MODELS TO EPILEPTIC CLASSIFICATION EVENTS IN EEG SIGNALS

5.1 Introduction

Diagnosing and monitoring epileptic application is one of the most common area in medical that has used EEG signals which enable us to deduce that epileptic EEG signals have been gaining significant amount of attention and interest. Till now, the utilization of an advanced processing method such as feature selection and classification enable the EEG signals to be effectively used in classification of epileptic and non-epileptic EEG events.

The diagnosing of epileptic application share a common peak detection problem as briefly discussed in the previous chapter. Because of that reason, the classification of epileptic and non-epileptic EEG events is chosen as peak-event related feature selection application. In this application, an epileptic event is response of the brain from human activities that cause a number of peaks in EEG signals.

A number of epileptic peak in EEG signals located at certain region have significantly differ in terms of time domain characteristics compared with non-epileptic peak in EEG signals. Because of that, a medical expert can manually foresee in the EEG signals for the patient with epileptic events. However, this kind of clinical practice is time consuming and fatigue for medical experts to be feasible. One of the solutions to overcome this problem is to develop an automated system for recognizing epileptic and non-epileptic events. To date, various research works have been done to develop an efficient automated system for epileptic EEG signals application (Acir, 2005; Acir & Guzelis, 2004; Gabor & Seyal, 1992; Juozapavi, Bacevi, Bugelskis, & Samaitien, 2011; Nigam & Graupe, 2004; Obeid & Wolf, 2004; Valenti et al, 2006; Yalcin et al, 2015).

In this chapter, all the proposed generalized models (i.e., model I, model II, model III, and model IV) with their associated feature are used as inputs to the NNRW classifier for epileptic classification events.

5.2 Data Description

The data used in this study is available and published on Bonn University EEG database (Andrzejak et al, 2001). The EEG recording was prepared using standard 10-20 electrode placement system. The datasets have five different sets, which are named as set A, set B, set C, set D, and set E. Each set contains 100 EEG segments that were selected from continuous multi-channel EEG recordings after removing muscle activity or eye movement artifacts. Each EEG segment consists of 4097 sampling points and the duration is about 23.6 seconds. Sets A and B consist of EEG segments taken from surface EEG recording collected from five healthy subjects. Subjects were relaxed in an awoken state with eyes open (A) and eyes closed (B), respectively. Sets C, D, and E were taken from EEG archive of presurgical diagnosis. Segments in set D were recorded from the epileptogenic zone. Set C is recorded from hippocampal formation of opposite hemisphere of brain. Sets C and D contain only activity measured during epileptic-free intervals. Set E contains only epileptic events. Data is recorded within 128-channel amplifier system and digitized at 173.61 Hz sampling rate and 12 bit A/D resolution. To select the EEG signal of desired band a band-pass filter having a pass band of 0.53–40 Hz (12 dB/oct) was used. In this study, only set A and set E were used. Set A represents as non-epileptic peak events while set E denotes as epileptic peak events. The examples of EEG signals (set A - set E) taken from Bonn University EEG database are shown in Figure 5.1.

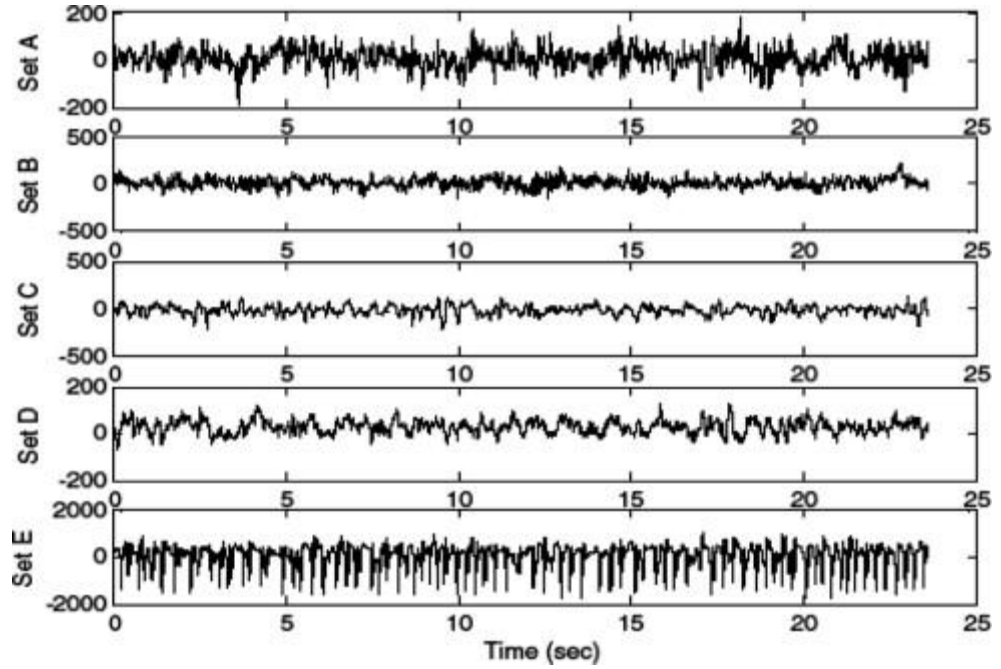


Figure 5.1: Example of five different sets of EEG signals taken from Bonn University EEG database (Polat & Gunes, 2008)

5.2.1 Preparing Training and Testing Sets

From the collected EEG raw data of the two sets EEG signals (set A and set E), 20000 peak candidate samples with their associated features were archived as EEG data for experiments. Note that, the feature of peak candidates are extracted using the method that has been discussed in subsection 3.4. From 20000 peak candidate samples, 10000 were assigned as epileptic peaks event from set E. The other 10000 were assigned as non-epileptic peaks event from set A. 100 peak candidate samples were randomly selected from each segment of both set. The four-fold cross-validation process is used to produce four groups of EEG data. The class distribution of the peak candidate sample and event is summarized in Table 5.1. The preparation of training and testing sets for the individual NNRW classifier are similar with the previous experiments that can be referred in Chapter 3.

Table 5.1: Class distribution of the peak candidate sample and event

Class	No. of peak candidate sample	No. of event	Partition of EEG data
Epileptic	10000	100	4-fold cross validation
Non-epileptic	10000	100	
Total	20000	100	

5.3 Results and Discussions

Two main experiments were conducted in this section. The first experiment was conducted to investigate the classification performance of the individual NNRW under a various number of hidden neurons. This experiment also aims to assess the performance of the individual NNRW over the 16 peak features. An ideal number of hidden neurons were chosen. Meanwhile, the second experiment was assigned to study the classification capability of the proposed models.

5.3.1 Experiment 1: Performance Evaluation of Various Number of Hidden Neuron

This experiment is conducted by varying the number of hidden neuron from 10 to 700 in steps of 50 in order to find ideal number of hidden neuron. The EEG dataset is randomly divided into four groups equally distributes the two-class ratio, by four-fold cross-validation process to prepare the experiment data of the individual NNRW classifier. Every group is alternately assigned as the testing set and the other three groups are combined to be a training set. The mean value of testing results from the four groups is then calculated. As shown in Table 5.2, this experiment is repeated 30 times, so that the mean of the training and testing results can be measured.

Table 5.2 illustrates the variation of training and testing accuracy with respect to a different number of hidden neurons. From Table 5.2, the mean training accuracy has increased up to 96% at 700 neurons. It can be observed that when the number of neuron

is up to 700 neurons, the computational time becomes slower. In this experiment, 500 neurons were set as shown in Figure 5.2. The mean of testing accuracies that has been shown in Table 5.2 also has increased up to 95% at 700 neurons. At 500 to 600 neurons, the testing accuracy maintained up to 94%.

Table 5.2: Training and testing results of various number of hidden neuron on EEG Epileptic database

Number of hidden neuron	Training Result				Testing Result			
	Mean accuracy (%)	Maximum accuracy (%)	Minimum accuracy (%)	STDEV (%)	Mean accuracy (%)	Maximum accuracy (%)	Minimum accuracy (%)	STDEV (%)
10	62	80	48	7.29	62	80	48	7.33
50	76	83	65	4.39	76	83	65	4.42
100	84	89	77	2.98	83	89	77	3.01
150	88	91	85	1.56	88	90	84	1.58
200	90	93	85	1.65	89	92	85	1.67
250	92	93	90	0.84	91	93	90	0.84
300	93	94	91	0.74	92	94	91	0.75
350	93	94	92	0.72	92	94	91	0.75
400	94	95	93	0.46	93	94	92	0.48
450	94	96	93	0.63	93	95	92	0.66
500	95	96	94	0.46	94	95	93	0.47
550	95	96	94	0.34	94	95	93	0.35
600	95	96	95	0.30	94	95	94	0.31
650	96	97	95	0.32	95	95	94	0.34
700	96	96	95	0.30	95	95	94	0.31

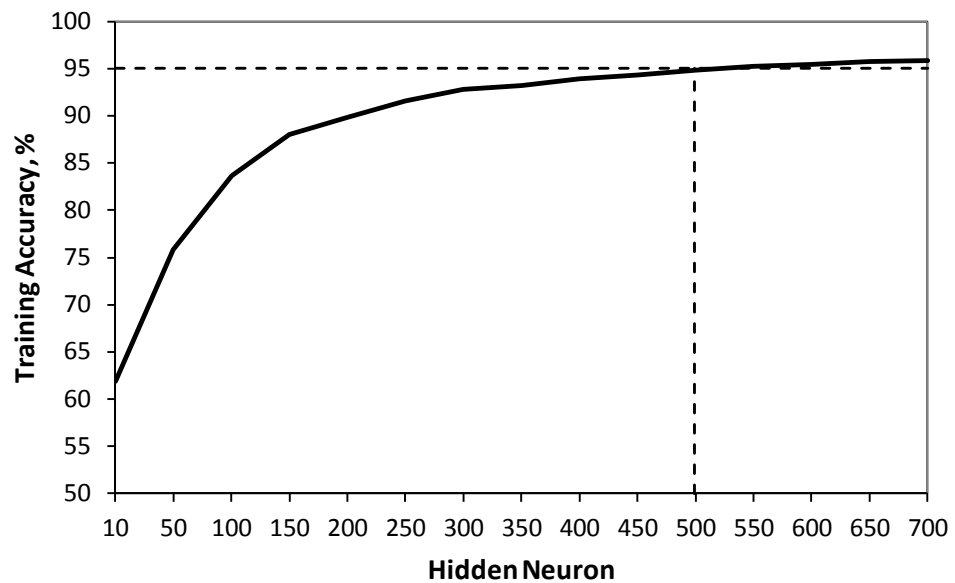


Figure 5.2: Selection of optimal number of hidden neuron from training accuracy

5.3.2 Experiment 2: Epileptic and Non-epileptic Event Classification

In this study, two EEG events have been assigned which are epileptic and non-epileptic events. 100 non-epileptic events are collected from set A while 100 epileptic peak events from set E. Each EEG event is a segment that consists of 4097 sampling points and the duration is about 23.6 seconds. The best combination of peak feature and the trained NNRW classifier are used to perform the classification. To distinguish between epileptic and non-epileptic events, the voting method is used. The epileptic event is recognized when more than 50 peaks are identified in within an event. Whereas, the non-epileptic event is identified once the peaks is lower than 50.

Table 5.3 demonstrates the confusion matrix of epileptic and non-epileptic event classification using the proposed models. It can be observed that the AMSKF model obtains the highest accuracy, with 97.9% of total accuracy, 100% of the non-epileptic event rate, and 96% of the epileptic event rate. There are four misclassifications of epileptic event.

Non-parametric statistical analysis has been made using Friedman test between AMSKF, BSKF, LocalDESKF, and GlobalDESKF in order to demonstrate the significant difference in testing accuracy of the results. The result of the Friedman test indicates that there is no significant difference in testing accuracy between AMSKF, BSKF, LocalDESKF, and GlobalDESKF methods.

The performance comparisons have been done to observe the efficiency of the proposed method. Table 5.4 gives the classification accuracies of this study and the existing methods on Bonn University EEG database. Referring to Table 5.4, the classification accuracy of this study using the NNRW method is lower than AIRS-PCA-

FFT and Wavelet-ANFIS methods. However, the classification accuracy of the NNRW using AMSKF model is higher than other methods.

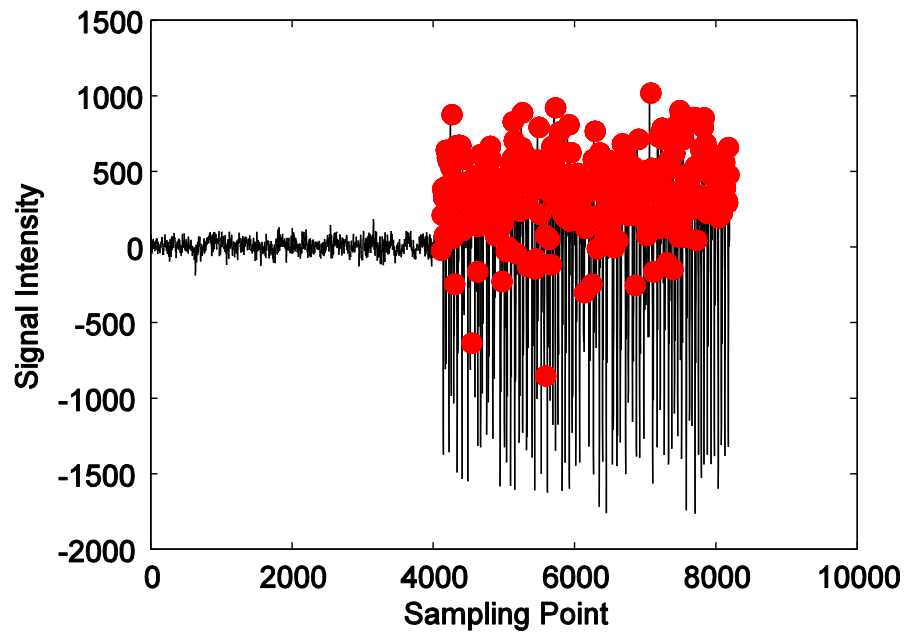
An example of epileptic and non-epileptic events classification is illustrated in Figure 5.3. There are more than 50 peaks (red dotted) identified in the epileptic segment (the right side) within the region from 4000 and 8000 sampling points. Figure 5.4 shows an example of misclassification of epileptic event in record S083. The number of detected peaks obviously can be seen is lower than 50. Consequently, the actual epileptic event is classified as non-epileptic event.

Table 5.3: Confusion matrix of epileptic and non-epileptic events classification

Peak Model	Output/Desired	Result (Non-epileptic event)	Result (Epileptic event)	Total Accuracy (%)
AMSKF	Result (Non-epileptic event)	100	4	97.9
BSKF	Result (Epileptic event)	0	96	96.9
	Result (Non-epileptic event)	100	6	
LocalDESKF	Result (Epileptic event)	0	94	96.4
	Result (Non-epileptic event)	100	7	
GlobalDESKF	Result (Epileptic event)	0	93	97.5
	Result (Non-epileptic event)	100	5	
	Result (Epileptic event)	0	95	

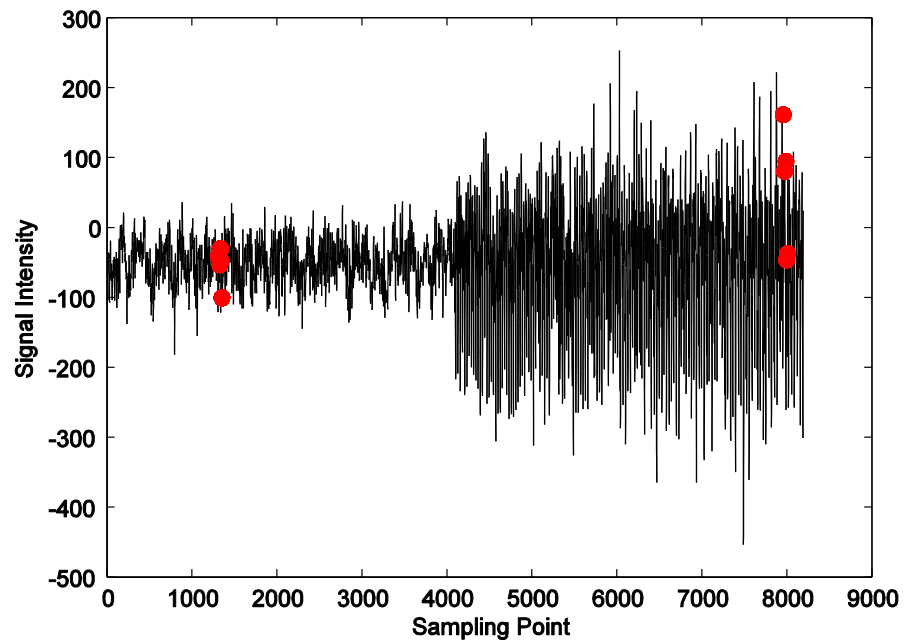
Table 5.4: Performance comparison of other methods

Author (Year)	Method	Type of Analysis	Accuracy (%)
Proposed work (2016)	AMSKF-NNRW	Time domain	98
Polat and Gunes (2008)	AIRS-PCA-FFT	Frequency domain	100
Subasi (2007)	Wavelet-ANFIS	Frequency domain	98.7
	Wavelet-MLPNN	Frequency domain	93.6
Subasi (2007)	Wavelet-ME	Frequency domain	95
Kannathal et al (2005)	ANFIS	Frequency domain	95
Guler and Ubeyli (2005)	Lyapunov exponent-Recurrent neural networks	Nonlinear	96.8



Epileptic EEG Signals

Figure 5.3: Example of epileptic event classification using record Z001 and S001



Epileptic EEG Signals

Figure 5.4: Example of misclassification of epileptic event in record Z083 and S083

5.4 Summary

This chapter exhibited the application of the proposed peak event-related models and NNRW classification method on epileptic EEG signals to classify between epileptic and non-epileptic events. A published EEG database from Bonn University was selected to evaluate the proposed method and at the same time apply the relevant combination of peak features for epileptic EEG signals application. From set A and set E of the published EEG database, 20000 peak candidate samples consist of epileptic peak and non-epileptic peak points were archived as EEG data for analysis. The major finding of this chapter is that all the proposed generalized models and NNRW classifier perform at par than the existing methods.

The final comparison results and ranking among all peak models are tabulated in Table 5.5. In general, the performances of the proposed all generalized models are greater than the existing four peak models. The first rank among all the peak models is GlobalDESKF model. The second rank is BSKF model. The third rank is AMSKF model.

Table 5.5: Performance comparisons and ranking among peak models

Peak model	Feature subset length	Selected features	Training Accuracy (%)	Testing Accuracy (%)	Average Computational Time (minutes)	Rank
GlobalDESKF	3	f_1, f_9, f_{16}	89.4	73.8	150	1
BSKF	8	$f_1, f_4, f_7, f_9, f_{11}, f_{12}, f_{13}, f_{16}$	96.5	72.9	150	2
AMSKF	11	$f_1, f_2, f_7, f_8, f_9, f_{10}, f_{11}, f_{12}, f_{13}, f_{14}, f_{15}$	91.8	72.7	150	3
LocalDESKF	3	f_1, f_{11}, f_{16}	95.1	72.3	150	4
Acir	6	$f_1, f_2, f_7, f_8, f_{13}, f_{14}$	76.3	52.2	5	5
Dumpala	4	f_1, f_6, f_{13}, f_{14}	80.9	51.5	5	6
Full feature set	16	$f_1, f_2, f_3, f_4, f_5, f_6, f_7, f_8, f_9, f_{10}, f_{11}, f_{12}, f_{13}, f_{14}, f_{15}, f_{16}$	73.8	49.4	7	7
Liu	11	$f_1, f_2, f_3, f_4, f_6, f_9, f_{12}, f_{13}, f_{14}, f_{15}, f_{16}$	77.2	48.2	6	8
Dingle	4	f_5, f_6, f_{13}, f_{14}	71.4	40.1	5	9

CHAPTER 6: CONCLUSIONS AND SUGGESTIONS FOR FUTURE RESEARCH

6.1 Introduction

This chapter presents some concluding remarks about the research works that have been conducted in this thesis. The beginning of section begins with a brief summarization of the main contributions, proposed methods, and experimental results that were reported in this thesis. The next section focuses on discussing the directions for future research.

6.2 Conclusions

The selection of the best combination of peak features for event-related EEG signals classification is an essential approach to ensure a higher performance for peak classification. In general, this approach has been proven in this thesis by using two main methods: 1) evaluation on the existing peak models using one common peak classification algorithm and 2) implementation of feature selection techniques.

Through the performance evaluation of the four different peak models, the best model with the highest accuracy is selected. In Sub-section 3.6.1, ANNPSO algorithm has been employed as a classifier for peak classification. From this, three case studies of eye event-related EEG signals involving eye movements, single and double eye blink signals were considered for performance evaluation of all peak models. The finding of the performance evaluations indicates that the Acir model is the best model for single and double eye blink EEG signals with 91.94% and 87.47%, respectively. While, the Dingle model is the best model for eye movement signals with 87.6%. In general, from the experimental results, it can be observed that the relevant peak model and the chosen *Gmean* function are the main factors in achieving higher classification rate. The

utilization of PSO learning algorithm with ANN for peak detection algorithm is to balance the ratio between false peak and true peak detection rates. The utilization of the ANNPSO method in peak classification can provide the best classification performance that has been proven by the experimental results. However, the ANNPSO cannot provide the fast learning speed once it integrates with feature selection technique for selecting the best peak model.

Then, the NNRW classifier is employed in peak classification. From experimental results in Sub-section 3.6.2.2, it was found that the Dumpala peak model to be the best for reliably detecting voluntary horizontal eye movement signal peaks, delivering a mean performance of 74.8% for the testing set. It also was observed that the Liu model obtained the best accuracy for single and double eye blink signals, with 69.1% and 69.6% of accuracy, respectively. This study also observes that defining more peak features on model is not a guarantee in producing better accuracy on all eye event-related EEG signals applications. However, determining the optimal model from the selected features associated with the advantageous of common classification platform is the best approach to gain the accuracy of detection performance.

In Sub-section 4.5, a new generalized peak model for EEG signals peak classification has been identified using a novel AMSKF feature selection approach. The proposed algorithm considered 11781 peak candidate samples of real EEG data, which were collected from 30 healthy subjects instructed to direct their single eye blink, double eye blink, and horizontal eye gaze. The detection performance of the NNRW with four different peak detection models and new AMSKF model are compared. In general, the experimental results showed that the accuracy of the NNRW with new AMSKF model is better than the NNRW with other models. The statistical analysis showed that the

detection performance of the NNRW with the new AMSKF model is significantly better in terms of testing accuracy compared to other models.

Other three new generalized models for EEG signals peak classification has been identified using the proposed feature selection algorithms, which are BSKF, LocalDESKF, and GlobalDESKF. In general, the performances of the proposed all generalized models are greater than the existing four peak models. Referring to the performance comparisons in Sub-section 4.9, the proposed GlobalDESKF model is the highest performance compared to other models, with 73.8%.

The four proposed models have been applied for classification of epileptic EEG signals application. The results that reported in Sub-section 5.3.2 indicated that the highest performance is obtained by AMSKF model. Compared to the existing studies in epileptic EEG events classification that utilized similar dataset, the AMSKF model and NNRW method performed at par, with 97.9%.

Although the GlobalDESKF and AMSKF models produced the highest classification performance for eye and epileptic event-related EEG datasets, however, the obtained statistical results indicated that there are no significant differences in term of average testing accuracy among the proposed models.

6.3 Future Research

The application of peak classification does not only focus on eye event-related EEG signals. Peak classification research may also provide a significant contribution to medical diagnostic, human-machine interface (HMI), brain-computer interface (BCI), and harmonic detection in digital and audio signal processing as these applications share a common peak classification problem. For example, an EEG peak in response to a change of horizontal eye gaze direction might be useful for patients with locked-in

syndrome or other disabilities for controlling the direction of computer cursor in BCI applications (Belkacem et al, 2014). This approach might also be translatable for EEG-based command of the movement of a robotic arm or wheelchair in HMI applications (Aziz et al, 2014; Postelnicu et al, 2011; Ramli, Arof, Ibrahim, Mokhtar, & Idris, 2015). It is also useful for medical expert to diagnose patients with stroke using EEG signals (Low et al, 2014).

REFERENCES

- Acir, N. (2005). Automated system for detection of epileptiform patterns in EEG by using a modified RBFN classifier. *Expert Systems with Applications*, 29(2), 455-462. doi: 10.1016/j.eswa.2005.04.040.
- Acir, N., & Guzelis, C. (2004). Automatic spike detection in EEG by a two-stage procedure based on support vector machines. *Comput Biol Med*, 34(7), 561-575. doi: 10.1016/j.combiomed.2003.08.003.
- Acir, N., Oztura, I., Kuntalp, M., Baklan, B., & Guzelis, C. (2005). Automatic detection of epileptiform events in EEG by a three-stage procedure based on artificial neural networks. *IEEE Trans Biomed Eng*, 52(1), 30-40. doi: 10.1109/TBME.2004.839630.
- Adam, A., Ibrahim, Z., Mokhtar, N., Shapiai, M. I., Mohd Tumari, M. Z., & Mubin, M. (2014, 3-5 December). *Feature Selection and Classifier Parameter Estimation for EEG Signal Peak Detection using Gravitational Search Algorithm*. Paper presented at the 4th International Conference on Artificial Intelligence with Applications in Engineering and Technology, Kota Kinabalu, Sabah, Malaysia. doi: 10.1109/ICAIET.2014.26.
- Adam, A., Ibrahim, Z., Mokhtar, N., Shapiai, M. I., & Mubin, M. (2015, 10-12 January). *Dingle's Model-based EEG Peak Detection using a Rule-based Classifier*. Paper presented at the International Conference on Artificial Life and Robotics, Oita, JAPAN.
- Adam, A., Shapiai, M. I., Mohd Tumari, M. Z., Mohamad, M. S., & Mubin, M. (2014). Feature selection and classifier parameters estimation for EEG signals peak detection using particle swarm optimization. *Scientific World Journal*, 2014(Article ID 973063), 973063. doi: 10.1155/2014/973063.
- Adam, A., Zainal Abidin, A. F., Ibrahim, Z., Hussain, A. R., Md Yusof, Z., & Shapiai, M. I. (2010, 26-28 May). *A Particle Swarm Optimization Approach to Robotic Drill Route Optimization*. Paper presented at the Fourth Asia International Conference on Mathematical/Analytical Modelling and Computer Simulation (AMS) Kota Kinabalu, Malaysia. doi: 10.1109/ams.2010.25.
- Ahila, R., Sadasivam, V., & Manimala, K. (2015). An integrated PSO for parameter determination and feature selection of ELM and its application in classification of power system disturbances. *Applied Soft Computing*, 32, 23-37. doi: 10.1016/j.asoc.2015.03.036.
- Ahmed, K. S. (2011). Wheelchair Movement Control VIA Human Eye Blinks. *American Journal of Biomedical Engineering*, 1(1), 55-58. doi: 10.5923/j.ajbe.20110101.09.

- Alcala-Fdez, J., Sanchez, L., Garcia, S., del Jesus, M. J., Ventura, S., Garrell, J. M., . . . Herrera, F. (2009). KEEL: a software tool to assess evolutionary algorithms for data mining problems. *Soft Computing*, 13(3), 307-318. doi: 10.1007/s00500-008-0323-y.
- Andrzejak, R. G., Lehnertz, K., Mormann, F., Rieke, C., David, P., & Elger, C. E. (2001). Indications of nonlinear deterministic and finite-dimensional structures in time series of brain electrical activity: dependence on recording region and brain state. *Phys Rev E Stat Nonlin Soft Matter Phys*, 64(6 Pt 1), 061907. doi: 10.1103/PhysRevE.64.061907.
- Ayob, M. N., Yusof, Z. M., Adam, A., Abidin, A. Z., Ibrahim, I., Ibrahim, Z., . . . Hani, M. K. (2010, 28-30 July 2010). *A Particle Swarm Optimization Approach for Routing in VLSI*. Paper presented at the Second International Conference on Computational Intelligence, Communication Systems and Networks (CICSyN). doi: 10.1109/CICSyN.2010.42.
- Aziz, F., Arof, H., Mokhtar, N., & Mubin, M. (2014). HMM based automated wheelchair navigation using EOG traces in EEG. *J Neural Eng*, 11(5), 056018. doi: 10.1088/1741-2560/11/5/056018.
- Bababdani, B. M., & Mousavi, M. (2013). Gravitational search algorithm: A new feature selection method for QSAR study of anticancer potency of imidazo[4,5-b]pyridine derivatives. *Chemometrics and Intelligent Laboratory Systems*, 122, 1-11. doi: 10.1016/j.chemolab.2012.12.002.
- Barea, R., Boquete, L., Ortega, S., Lopez, E., & Rodriguez-Ascariz, J. M. (2012). EOG-based eye movements codification for human computer interaction. *Expert Systems with Applications*, 39(3), 2677-2683. doi: 10.1016/j.eswa.2011.08.123.
- Belkacem, A. N., Hirose, H., Yoshimura, N., Shin, D., & Koike, Y. (2014). Classification of Four Eye Directions from EEG Signals for Eye-Movement-Based Communication Systems. *Journal of Medical and Biological Engineering*, 34(6), 581-588. doi: 10.5405/jmbe.1596.
- Billauer. (2012). peakdet: Peak detection using MATLAB. <http://billauer.co.il/peakdet.html>.
- Bonner, R. E., Crevasse, L., Ferrer, M. I., & Greenfield, J. C., Jr. (1972). A new computer program for analysis of scalar electrocardiograms. *Comput Biomed Res*, 5(6), 629-653. doi: 10.1016/0010-4809(72)90043-2.
- C.R. Rao, & Mit, S. K. (1971). *Generalized Inverse of Matrices and Its Applications*. New York: Wiley.
- Cao, F. L., Ye, H. L., & Wang, D. H. (2015). A probabilistic learning algorithm for robust modeling using neural networks with random weights. *Information Sciences*, 313(C), 62-78. doi: 10.1016/j.ins.2015.03.039.

- Claassen, J., Taccone, F. S., Horn, P., Holtkamp, M., Stocchetti, N., Oddo, M., & Neurointensive Care Section of the European Society of Intensive Care, M. (2013). Recommendations on the use of EEG monitoring in critically ill patients: consensus statement from the neurointensive care section of the ESICM. *Intensive Care Med*, 39(8), 1337-1351. doi: 10.1007/s00134-013-2938-4.
- Dingle, A. A., Jones, R. D., Carroll, G. J., & Fright, W. R. (1993). A multistage system to detect epileptiform activity in the EEG *IEEE Trans Biomed Eng* (Vol. 40, pp. 1260-1268).
- Dorigo, M., Maniezzo, V., & Colorni, A. (1996). Ant system: optimization by a colony of cooperating agents. *IEEE Trans Syst Man Cybern B Cybern*, 26(1), 29-41. doi: 10.1109/3477.484436.
- Dumpala, S. R., Reddy, S. N., & Sarna, S. K. (1982). An algorithm for the detection of peaks in biological signals. *Comput Programs Biomed*, 14(3), 249-256. doi: 10.1016/0010-468X(82)90030-7.
- Elgendi, M., Meo, M., & Abbott, D. (2016). A Proof-of-Concept Study: Simple and Effective Detection of P and T Waves in Arrhythmic ECG Signals. *Bioengineering*, 3(4), 26.
- Elgendi, M., Norton, I., Brearley, M., Abbott, D., & Schuurmans, D. (2013). Systolic peak detection in acceleration photoplethysmograms measured from emergency responders in tropical conditions. *PLoS One*, 8(10), e76585. doi: 10.1371/journal.pone.0076585.
- Erol, O. K., & Eksin, I. (2006). A new optimization method: Big Bang Big Crunch. *Advances in Engineering Software*, 37(2), 106-111. doi: 10.1016/j.advengsoft.2005.04.005.
- Exarchos, T. P., Tzallas, A. T., Fotiadis, D. I., Konitsiotis, S., & Giannopoulos, S. (2006). EEG Transient Event Detection and Classification Using Association Rules. *IEEE Transaction on Information Technology in Biomedicine*, 10(3), 451-457. doi: 10.1109/TITB.2006.872067
- Gabor, A. J., & Seyal, M. (1992). Automated Interictal Eeg Spike Detection Using Artificial Neural Networks. *Electroencephalography and Clinical Neurophysiology*, 83(5), 271-280. doi: 10.1016/0013-4694(92)90086-W.
- Gloor, P. (1975). Contributions of electroencephalography and electrocorticography to the neurosurgical treatment of the epilepsies. *Advances in Neurology*, 8, 59-105.
- Guler, I., & Ubeyli, E. D. (2005). Adaptive neuro-fuzzy inference system for classification of EEG signals using wavelet coefficients. *Journal of Neuroscience Methods*, 148(2), 113-121. doi: 10.1016/j.jneumeth.2005.04.013.
- Guo, X., Yin, Y., Dong, C., Yang, G., & Zhou, G. (2008, 25-27 August). *On the Class Imbalance Problem*. Paper presented at the Fourth International Conference on Natural Computation (ICNC 08), Jinan, China. doi: 10.1109/ICNC.2008.871.

- Hatamlou, A. (2013). Black hole: A new heuristic optimization approach for data clustering. *Information Sciences*, 222, 175-184. doi: 10.1016/j.ins.2012.08.023.
- Hooker, C. A. (1995). Adaptation in Natural and Artificial Systems - Holland, Jh. *Philosophical Psychology*, 8(3), 287-299. doi: Doi 10.1080/09515089508573159.
- Ibrahim, Z., Abdul Aziz, H., Abdul Aziz, A., Razali, S., Shapiai, M. I., Nawawi, S. W., & Mohamad, M. S. (2015). A Kalman filter approach for solving unimodal optimization problems. *ICIC Express Letters*, 9(12), 3415-3422.
- Ibrahim, Z., Khalid, N. K., Mukred, J. A. A., Buyamin, S., Yusof, Z. M., Saa'id, M. F. M., . . . Engelbrecht, A. R. (2012). A DNA Sequence Design for DNA Computation Based on Binary Vector Evaluated Particle Swarm Optimization. *International Journal of Unconventional Computing*, 8(2), 119-137.
- Igel'nik, B., & Pao, Y. H. (1995). Stochastic choice of basis functions in adaptive function approximation and the functional-link net. *IEEE Trans Neural Netw*, 6(6), 1320-1329. doi: 10.1109/72.471375.
- Indiradevi, K. P., Elias, E., Sathidevi, P. S., Dinesh Nayak, S., & Radhakrishnan, K. (2008). A multi-level wavelet approach for automatic detection of epileptic spikes in the electroencephalogram. *Comput Biol Med*, 38(7), 805-816. doi: 10.1016/j.compbiomed.2008.04.010.
- Iwasaki, M., Kellinghaus, C., Alexopoulos, A. V., Burgess, R. C., Kumar, A. N., Han, Y. H., . . . Leigh, R. J. (2005). Effects of eyelid closure, blinks, and eye movements on the electroencephalogram. *Clin Neurophysiol*, 116(4), 878-885. doi: 10.1016/j.clinph.2004.11.001.
- Ji, Z., Wang, X., Sugi, T., Goto, S., & Nakamura, M. (2011). *Automatic spike detection based on real-time multi-channel template*. Paper presented at the 4th International Conference on Biomedical Engineering and Informatics (BMEI), Shanghai. doi: 10.1109/BMEI.2011.6098388.
- Johnson, D. S., Aragon, C. R., Mcgeoch, L. A., & Schevon, C. (1989). Optimization by Simulated Annealing - an Experimental Evaluation .1. Graph Partitioning. *Operations Research*, 37(6), 865-892. doi: DOI 10.1287/opre.37.6.865.
- Juo'zapavi, A., Bacevi, G., Bugelskis, D., & Samaitien, R. (2011). EEG analysis – automatic spike detection. *Journal of Nonlinear Analysis: Modelling and Control*, 16(4), 375-386.
- Kannathal, N., Choo, M. L., Acharya, U. R., & Sadasivan, P. K. (2005). Entropies for detection of epilepsy in EEG. *Computer Methods and Programs in Biomedicine*, 80(3), 187-194. doi: 10.1016/j.cmpb.2005.06.012.
- Kennedy, J., & Eberhart, R. (1995, 27 November- 1 December). *Particle Swarm Optimization*. Paper presented at the Proceedings of the IEEE international Conference on Neural Networks (ICW), Perth, Western Australia.

- Kennedy, J., & Eberhart, R. (1995, 27 November- 1 December). *Particle Swarm Optimization*. Paper presented at the Proceedings of the IEEE International Conference on Neural Networks (ICW), Perth, Western Australia.
- Kim, J., & Shin, H. (2016). Simple and Robust Realtime QRS Detection Algorithm Based on Spatiotemporal Characteristic of the QRS Complex. *PLOS ONE*, 11(3), e0150144. doi: 10.1371/journal.pone.0150144.
- Klados, M. A., Papadelis, C., Braun, C., & Bamidis, P. D. (2011). REG-ICA: A hybrid methodology combining Blind Source Separation and regression techniques for the rejection of ocular artifacts. *Biomedical Signal Processing and Control*, 6(3), 291-300. doi: 10.1016/j.bspc.2011.02.001.
- Klem, G. H., Luders, H. O., Jasper, H. H., & Elger, C. (1999). The ten-twenty electrode system of the International Federation. The International Federation of Clinical Neurophysiology. *Electroencephalogr Clin Neurophysiol Suppl*, 52, 3-6.
- Lim, K. S., Ibrahim, Z., Buyamin, S., Ahmad, A., Naim, F., Ghazali, K. H., & Mokhtar, N. (2013). Improving Vector Evaluated Particle Swarm Optimisation by incorporating nondominated solutions. *Scientific World Journal*, 2013, 510763. doi: 10.1155/2013/510763.
- Lin, J. S., & Yang, W. C. (2012). Wireless Brain-Computer Interface for Electric Wheelchairs with EEG and Eye-Blinking Signals. *International Journal of Innovative Computing Information and Control*, 8(9), 6011-6024.
- Liu, H. S., Zhang, T., & Yang, F. S. (2002). A multistage, multimethod approach for automatic detection and classification of epileptiform EEG. *IEEE Trans Biomed Eng*, 49(12 Pt 2), 1557-1566. doi: 10.1109/TBME.2002.805477.
- Liu, Y. C., Lin, C. C., Tsai, J. J., & Sun, Y. N. (2013). Model-based spike detection of epileptic EEG data. *Sensors (Basel)*, 13(9), 12536-12547. doi: 10.3390/s130912536.
- Low, E., Mathieson, S. R., Stevenson, N. J., Livingstone, V., Ryan, C. A., Bogue, C. O., . . . Boylan, G. B. (2014). Early postnatal EEG features of perinatal arterial ischaemic stroke with seizures. *PLoS One*, 9(7), e100973. doi: 10.1371/journal.pone.0100973.
- Lu, J., Zhao, J. W., & Cao, F. L. (2014). Extended feed forward neural networks with random weights for face recognition. *Neurocomputing*, 136, 96-102. doi: 10.1016/j.neucom.2014.01.022.
- Lu, W., Nystrom, M. M., Parikh, P. J., Fooshee, D. R., Hubenschmidt, J. P., Bradley, J. D., & Low, D. A. (2006). A semi-automatic method for peak and valley detection in free-breathing respiratory waveforms. *Med Phys*, 33(10), 3634-3636. doi: 10.1118/1.2348764.

- Manikandan, M. S., & Soman, K. P. (2012). A novel method for detecting R-peaks in electrocardiogram (ECG) signal. *Biomedical Signal Processing and Control*, 7(2), 118-128. doi: 10.1016/j.bspc.2011.03.004.
- Marinakakis, Y., Marinaki, M., & Dounias, G. (2011). Honey bees mating optimization algorithm for the Euclidean traveling salesman problem. *Information Sciences*, 181(20), 4684-4698. doi: 10.1016/j.ins.2010.06.032.
- Md Yusof, Z., Ibrahim, I., Ibrahim, Z., Abd Aziz, N. H., & Ab Aziz, N. A. (2016). *Local Optimum Distance Evaluated Simulated Kalman Filter for Combinatorial Optimization Problems (Accepted)*. Paper presented at the National Conference for Postgraduate Research (NCON-PGR), Universiti Malaysia Pahang, Campus Pekan.
- Md Yusof, Z., Ibrahim, I., Satiman, S. N., Ibrahim, Z., Abd Aziz, N. H., & Ab Aziz, N. A. (2015, 2 - 4 December 2015). *BSKF: Binary Simulated Kalman Filter*. Paper presented at the Third International Conference on Artificial Intelligence, Modelling and Simulation, Kota Kinabalu, Sabah (Malaysia).
- Md Yusof, Z., Ibrahim, Z., Ibrahim, I., Mohd Azmi, K. Z., Abd Aziz, N. A., Abd Aziz, N. H., & Mohamad, M. S. (2016a). Angle modulated simulated Kalman filter algorithm for combinatorial optimization problems. *ARPJ Journal of Engineering and Applied Sciences* 11(7), 4854-4859.
- Md Yusof, Z., Ibrahim, Z., Ibrahim, I., Mohd Azmi, K. Z., Abd Aziz, N. A., Abd Aziz, N. H., & Mohamad, M. S. (2016b). Distance Evaluated Simulated Kalman Filter for Combinatorial Optimization Problems. *ARPJ Journal of Engineering and Applied Sciences*, 11(7), 4911-4916.
- Mohamad, M. S., Omatu, S., Deris, S., Yoshioka, M., Abdullah, A., & Ibrahim, Z. (2013). An enhancement of binary particle swarm optimization for gene selection in classifying cancer classes. *Algorithms Mol Biol*, 8(1), 15. doi: 10.1186/1748-7188-8-15.
- Nicolas-Alonso, L. F., & Gomez-Gil, J. (2012). Brain computer interfaces, a review. *Sensors (Basel)*, 12(2), 1211-1279. doi: 10.3390/s120201211.
- Nigam, V. P., & Graupe, D. (2004). A neural-network-based detection of epilepsy. *Neurol Res*, 26(1), 55-60. doi: 10.1179/016164104773026534.
- Obeid, I., & Wolf, P. D. (2004). Evaluation of spike-detection algorithms for a brain-machine interface application. *IEEE Trans Biomed Eng*, 51(6), 905-911. doi: 10.1109/TBME.2004.826683.
- Oikonomou, V. P., Tzallas, A. T., & Fotiadis, D. I. (2007). A Kalman filter based methodology for EEG spike enhancement. *Comput Methods Programs Biomed*, 85(2), 101-108. doi: 10.1016/j.cmpb.2006.10.003.

- Pao, Y. H., Park, G. H., & Sobajic, D. J. (1994). Learning and Generalization Characteristics of the Random Vector Functional-Link Net. *Neurocomputing*, 6(2), 163-180. doi: Doi 10.1016/0925-2312(94)90053-1.
- Pao, Y. H., & Takefuji, Y. (1992). Functional-Link Net Computing - Theory, System Architecture, and Functionalities. *Computer*, 25(5), 76-79. doi: 10.1109/2.144401.
- Polat, K., & Gunes, S. (2008). Artificial immune recognition system with fuzzy resource allocation mechanism classifier, principal component analysis and FFT method based new hybrid automated identification system for classification of EEG signals. *Expert Systems with Applications*, 34(3), 2039-2048. doi: 10.1016/j.eswa.2007.02.009.
- Postelnicu, C. C., Talaba, D., & Toma, M. I. (2011). Controlling a Robotic Arm by Brainwaves and Eye Movement *Technological Innovation for Sustainability* (Vol. 349, pp. 157-164).
- Putignano, M., Intermite, A., & Welsch, C. P. (2012). A non-linear algorithm for current signal filtering and peak detection in SiPM. *Journal of Instrumentation*, 7(08), P08014-P08014. doi: 10.1088/1748-0221/7/08/p08014.
- Ramli, R., Arof, H., Ibrahim, F., Mokhtar, N., & Idris, M. Y. I. (2015). Using finite state machine and a hybrid of EEG signal and EOG artifacts for an asynchronous wheelchair navigation. *Expert Systems with Applications*, 42(5), 2451-2463. doi: 10.1016/j.eswa.2014.10.052.
- Rashedi, E., Nezamabadi-Pour, H., & Saryazdi, S. (2009). GSA: A Gravitational Search Algorithm. *Information Sciences*, 179(13), 2232-2248. doi: 10.1016/j.ins.2009.03.004.
- Schmidt, W. F. (1992). *Feed forward neural networks with random weights*. Paper presented at the 1th IAPR International Conference on Pattern Recognition Methodology and Systems, The Hague doi: 10.1109/ICPR.1992.201708.
- Senhadji, L., & Wendling, F. (2002). Epileptic transient detection: wavelets and time-frequency approaches. *Neurophysiol Clin*, 32(3), 175-192. doi: 10.1016/S0987-7053(02)00304-0.
- Shaffer, F., McCraty, R., & Zerr, C. L. (2014). A healthy heart is not a metronome: an integrative review of the heart's anatomy and heart rate variability. *Front Psychol*, 5(1040), 1040. doi: 10.3389/fpsyg.2014.01040.
- Shah-Hosseini, H. (2007). Problem solving by intelligent water drops. *2007 Ieee Congress on Evolutionary Computation, Vols 1-10, Proceedings*, 3226-3231.
- Shi, Y., & Eberhart, R. C. (1998a, 4-9 May). *A Modified Particle Swarm Optimizer*. Paper presented at the Proceedings of the IEEE International Conference on Evolutionary Computation, Anchorage, Alaska, USA.

- Shi, Y., & Eberhart, R. C. (1998b, 25-27 March). *Parameter Selection in Particle Swarm Optimization*. Paper presented at the Proceedings of the 7th Annual Conference on Evolutionary Programming, San Diego, C.A.
- Shi, Y., & Eberhart, R. C. (1999, July 6-9). *Empirical Study of Particle Swarm Optimization*. Paper presented at the IEEE Congress on Evolutionary Computation (CEC), Washington D.C, USA. doi: 10.1109/CEC.1999.785511.
- Sinno, N., & Tout, K. (2008). Analysis of Epileptic Events Using Wavelet Packets. *International Arab Journal of Information Technology*, 5(4), 421-425.
- Sovierzoski, M. A., Argoud, F. I. M., & de Azevedo, F. M. (2008). *Identifying eye blinks in EEG signal analysis*. Paper presented at the International Conference on Information Technology and Applications in Biomedicine (ITAB) Shenzhen. doi: 10.1109/ITAB.2008.4570605
- Subasi, A. (2007). EEG signal classification using wavelet feature extraction and a mixture of expert model. *Expert Systems with Applications*, 32(4), 1084-1093. doi: 10.1016/j.eswa.2006.02.005.
- Tafreshi, R., Jaleel, A., Lim, J., & Tafreshi, L. (2014). Automated analysis of ECG waveforms with atypical QRS complex morphologies. *Biomedical Signal Processing and Control*, 10, 41-49. doi: 10.1016/j.bspc.2013.12.007.
- Valenti, P., Cazamajou, E., Scarpettini, M., Aizemberg, A., Silva, W., & Kochen, S. (2006). Automatic detection of interictal spikes using data mining models. *Journal of Neuroscience Methods*, 150(1), 105-110. doi: 10.1016/j.jneumeth.2005.06.005.
- Webber, W. R. S., & Lesser, R. P. (2017). Automated spike detection in EEG. *Clinical Neurophysiology*, 128(1), 241-242. doi: 10.1016/j.clinph.2016.11.018.
- Wilson, S. B., & Emerson, R. (2002). Spike detection: a review and comparison of algorithms. *Clin Neurophysiol*, 113(12), 1873-1881. doi: 10.1016/S1388-2457(02)00297-3.
- Xiong, N., Molina, D., Ortiz, M. L., & Herrera, F. (2015). A Walk into Metaheuristics for Engineering Optimization: Principles, Methods and Recent Trends. *International Journal of Computational Intelligence Systems*, 8(4), 606-636. doi: 10.1080/18756891.2015.1046324.
- Xu, L., Meng, M. Q.-H., Liu, R., & Wang, K. (2008, August 20-24). *Robust Peak Detection of Pulse Waveform Using Height Ratio*. Paper presented at the 30th Annual International IEEE EMBS Conference, Vancouver, British Columbia, Canada. doi: 10.1109/IEMBS.2008.4650051.
- Xu, N., Gao, X., Hong, B., Miao, X., Gao, S., & Yang, F. (2004). BCI Competition 2003--Data set IIB: enhancing P300 wave detection using ICA-based subspace projections for BCI applications. *IEEE Trans Biomed Eng*, 51(6), 1067-1072. doi: 10.1109/TBME.2004.826699.

- Yalcin, N., Tezel, G., & Karakuzu, C. (2015). Epilepsy diagnosis using artificial neural network learned by PSO. *Turkish Journal of Electrical Engineering and Computer Sciences*, 23(2), 421-432. doi: 10.3906/elk-1212-151.
- Yang, X.-S. (2009). Harmony Search as a Metaheuristic Algorithm. In Z. Geem (Ed.), *Music-Inspired Harmony Search Algorithm* (Vol. 191, pp. 1-14): Springer Berlin Heidelberg.
- Yang, X.-S. (2010). A New Metaheuristic Bat-Inspired Algorithm. In J. González, D. Pelta, C. Cruz, G. Terrazas & N. Krasnogor (Eds.), *Nature Inspired Cooperative Strategies for Optimization (NICSO 2010)* (Vol. 284, pp. 65-74): Springer Berlin Heidelberg.
- Yang, X. S. (2010). Firefly Algorithm, Levy Flights and Global Optimization. *Research and Development in Intelligent Systems Xxvi*, 209-218. doi: 10.1007/978-1-84882-983-1_15.
- Zappasodi, F., Olejarczyk, E., Marzetti, L., Assenza, G., Pizzella, V., & Tecchio, F. (2014). Fractal dimension of EEG activity senses neuronal impairment in acute stroke. *PLoS One*, 9(6), e100199. doi: 10.1371/journal.pone.0100199.
- Zhang, X. L., Chen, W., Wang, B. J., & Chen, X. F. (2015). Intelligent fault diagnosis of rotating machinery using support vector machine with ant colony algorithm for synchronous feature selection and parameter optimization. *Neurocomputing*, 167, 260-279. doi: 10.1016/j.neucom.2015.04.069.

LIST OF PUBLICATIONS AND PAPERS PRESENTED

- i. Adam, A., Shapiai, M. I., Mohd Tumari, M. Z., Mohamad, M. S., & Mubin, M. (2014). Feature selection and classifier parameters estimation for EEG signals peak detection using particle swarm optimization. *Scientific World Journal*, 2014(Article ID 973063), 973063. doi: 10.1155/2014/973063.
- ii. Adam, A., Ibrahim, Z., Mokhtar, N., Shapiai, M. I., & Mubin, M. (2016). Evaluation of different peak models of eye blink EEG for signal peak detection using artificial neural network. *Neural Network World*, 26(1), 67-90. doi: 10.14311/NNW.2016.26.004.
- iii. Adam, A., Ibrahim, Z., Mokhtar, N., Shapiai, M. I., Cumming, P., & Mubin, M. (2016). Evaluation of Different Time Domain Peak Models using Extreme Learning Machine-based Peak Detection for EEG signal. *SpringerPlus*, 5(1036), 1-14. doi: 10.1186/s40064-016-2697-0.
- iv. Adam, A., Ibrahim, Z., Mokhtar, N., Shapiai, M. I., Mubin, M., & Saad, I. (2016). Feature selection using angle modulated simulated Kalman filter for peak classification of EEG signals, *SpringerPlus*. 5(1580), 1-24. doi: 10.1186/s40064-016-3277-z.
- v. Adam, A., Ibrahim, Z., Mokhtar, N., Shapiai, M. I., Mubin, M., & Saad, I. (2017). Improving EEG signal peak detection using feature weight learning of a neural network with random weights for eye event-related applications, *SADHANA-Academy Proceedings in Engineering Sciences*. doi: 10.1007/s12046-017-0633-9.
- vi. Adam, A., Ibrahim, Z., Mokhtar, N., Shapiai, M. I., & Mubin, M. (2015, 10-12 January). Dingle's Model-based EEG Peak Detection using a Rule-based Classifier. Paper presented at the International Conference on Artificial Life and Robotics, Oita, JAPAN.

- vii. Adam, A., Ibrahim, Z., Mokhtar, N., Shapiai, M. I., Mohd Tumari, M. Z., & Mubin, M. (2014, 3-5 December). Feature Selection and Classifier Parameter Estimation for EEG Signal Peak Detection using Gravitational Search Algorithm. Paper presented at the 4th International Conference on Artificial Intelligence with Applications in Engineering and Technology, Kota Kinabalu, Sabah, Malaysia. doi: 10.1109/ICAIET.2014.26.

APPENDIX A – MEDICAL ETHICS COMMITTEE APPROVAL



**UNIVERSITI
MALAYA**

PUSAT PERUBATAN UM

**MEDICAL ETHICS COMMITTEE
UNIVERSITY MALAYA MEDICAL CENTRE**

ADDRESS: LEMBAH PANTAI, 59100 KUALA LUMPUR, MALAYSIA
TELEPHONE: 03-79493209 / 2251 FAXIMILE: 03-79492030

NAME OF ETHICS COMMITTEE/IRB: Medical Ethics Committee, University Malaya Medical Centre	ETHICS COMMITTEE/IRB REFERENCE NUMBER: 974.21
ADDRESS: LEMBAH PANTAI 59100 KUALA LUMPUR	
PROTOCOL NO (if applicable): - TITLE: Human machine interface via brain signals	VERSION NO.: -
PRINCIPAL INVESTIGATOR: Dr. Norrima Mokhtar	SPONSOR: HIR-MOHE

The following item [✓] have been received and reviewed in connection with the above study to be conducted by the above investigator.

- | | |
|--|---------------------|
| [✓] Application to Conduct Research Project (form) | Ver date: 05 Mar 13 |
| [✓] Study Protocol | Ver date: |
| [] Investigator Brochure | Ver date: |
| [] Patient Information Sheet | Ver date: |
| [] Consent Form | Ver date: |
| [] Questionnaire | Ver date: |
| [✓] Investigator's CV (Dr. Norrima Mokhtar) | |

and the decision is [✓] :

- [✓] Approved
[] Modification requested (item specified below or in accompanying letter)
[] Rejected (reasons specified below or in accompanying letter)

Comments:

Investigator are required to:

- 1) follow instructions, guidelines and requirements of the Medical Ethics Committee.
- 2) report any protocol deviations/violations to Medical Ethics Committee.
- 3) provide annual and closure reports to the Medical Ethics Committee.
- 4) comply with International Conference on Harmonization – Guidelines for Good Clinical Practice (ICH-GCP) and the Declaration of Helsinki.
- 5) obtain permission from the Director of UMMC before starting research that involves recruitment of UMMC patients.
- 6) ensure that if the research is sponsored, the usage of consumable items and laboratory tests from UMMC services are not charged in the patient's hospital bills but are borne by the research grant.
- 7) note that he/she can appeal to the Chairman of MEC for studies that are rejected.
- 8) note that Medical Ethics Committee may audit the approved study.
- 9) ensure that the study does not take precedence over the safety of subjects.

Date of meeting : 20th MARCH 2013

Date of approval: 20th MARCH 2013

c.c Head
Department of Electrical Engineering
Faculty of Engineering, UM

Deputy Dean (Research)
Faculty of Medicine

Secretary
Medical Ethics Committee
University Malaya Medical Centre

PROF. DATUK LOOI LAI MENG
Chairman
Medical Ethics Committee