

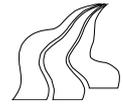
Non-invasively multimodal approach for decoding  
the intent of movement impaired individuals and  
low dimensional control of movement based on  
muscle synergies

By  
**JACOB KOCH PEDERSEN**

Thesis for the degree of Master of Science in Biomedical engineering

Department of Health Science and Technology  
The Faculties of Engineering, Science and Medicine





**Title:**

Non-invasively multimodal approach for decoding the intent of movement impaired individuals and low dimensional control of movement based on muscle synergies

**Project period:**

10<sup>th</sup> semester  
February 2<sup>nd</sup> - June 9<sup>th</sup> 2009

**Student:**

Jacob Koch Pedersen, group 1085h

**Supervisors:**

Professor Dario Farina  
Ph.D. student Silvia Muceli

Movement disability is one of the most invalidating conditions in consequence of a spinal cord injury. Consequences of SCI can be wide spread implying variety of complications, and several functions may be affected. SCI and can be devastating dependent on the level of injury on the spinal cord, causing a near total loss of independence in the case of high-level injury. Rehabilitation devices that enable paralyzed people to perform daily activities or restore even a modest level of independence have the potential to generate an enormous impact on the quality of life and lower the related costs to society.

In severely motor impaired individuals, only a few modalities remain under voluntary control, which complicates the opportunity for these individuals to use assistive technology to allow them to independently perform movements essential for many self-care activities.

Neuroprosthesis operate through a command interface that measures some modality over which voluntary control is maintained, and translates this to a specific operation of the prosthesis via electrical stimulation in case of FES and FET.

A configuration based on a Multi Layer Perceptron Neural Network was proposed to predict the users intent based on gaze-direction and the muscle activity recorded non-invasively from the proximal arm muscles. Additionally, a possibility of simplifying the control of electrical stimulation, was investigated based on muscle synergies extracted by non-negative matrix factorization

From 7 subject the users intent was predicted based on surface EMG and simulated gaze-direction, indicating a clear improvement compared to predictions based on the muscle activity only. The Neural Network outperformed the predictions achieved with a Kalman filter.

In 1 subject, the muscle activation level in 9 distal arm muscles was predicted continuously using the Neural network based on the muscle activity from the Deltoid muscle and recorded gaze-direction.

Further, the muscle activity in 12 muscles could be reconstructed based on a low dimensional muscle activation pattern, indicating a simplified approach for the stimulation control in a FES or FET system.

The data basis of this analysis should in future work be extended and additionally it should be demonstrated that individuals accurately can control the muscle synergies in a real-time FES/FET application.

**Number of Copies:** 4

**Number of pages in report:** 133



# Preface

This project report documents the work I have conducted during my 10<sup>th</sup> master thesis under the supervision of Dario Farina and Silvia Muceli, Spring 2009 at the Department of Health Science and Technology, at Aalborg University, Denmark. The report is intended for fellow students, supervisors and others with interest in this research area. The initiation of this work was inspired by my work at the Rehabilitation Institute of Chicago, Northwestern University during my 9<sup>th</sup> semester in Chicago under supervision of Eric Perreault.

I want to address a great thank to my supervisors Dario Farina and Silvia. To Dario for his inspiration and for sharing his high and impressive expertise during my work. To Silvia, who has contributed more than the expectable with her engagement, especially during conduction of the experimental work, including Sundays. Without doubt her help was of vital importance for me, to be able to conduct the experimental setup. Also for her thoroughness in her supervision. Thanks to the Oticon foundation for the grant I received during my 10<sup>th</sup> semester. I believe I have taken the full advantage of the grant by keeping my focus at the project throughout the entire period, and have had the privilege not to work beside my study. Indeed, this had an positive and direct effect on the effort I have been able to put into my master thesis. Thanks to Siemens for giving me a grant, and thereby provided me the opportunity to travel to Chicago to present my work at Rehabilitation Institute of Chicago and in addition to acquire equipment for my project.

This project documents not only the work I have prepared during my 10<sup>th</sup> semester, but it also rounds off 5 years of education, during which I have spent a great part of my time on my studies. Hence, despite the normal practice, I find it natural to thank my closest family and friends, who in a way have been affected and accepted my, occasional 'part time friendship', and even though been flexible in their planning to see me.

Especially thanks to mom, dad, and big brother for your support, understanding and acceptance of the many hours spend on my study, and the times when occasionally my work implied a bad mood. Your extensively positiveness have always had an impact on me and in the busy project periods, my visits at home have always given me new energy to keep my focus on finishing my work. Additionally your visit in Chicago during my 9<sup>th</sup> yielded precious and unforgettable memories. Also thanks for your contribution to my Master thesis in form of proof reading and acting subject. You deserve my utmost gratefulness.

To my friends, who have, if not understood the time I have spend on my projects, then at least accepted the circumstances. I have greatly appreciated the time with you all, though limited in periods. It has been valuable to me. Especially, it has been fantastic to sustain my friendship with you Christoffer Ulf, despite the facts of several thousands kilometers between us. Your visit in Chicago on my 9<sup>th</sup> semester is definitely printed in my memory as an unforgettable party and always will be.

Aalborg University, June 9<sup>th</sup> 2009

---

Jacob Koch Pedersen

2b || ~2b = ?



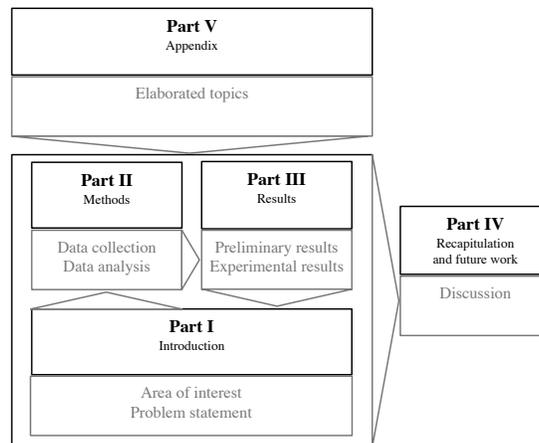
---

# Reading instructions

---

# 1

The report consist of 5 parts indicated with roman numbers in the table of contents and as depicted in figure 1.1.



**Figure 1.1:** Report structure, showing the 5 main parts in which the report is divided. The arrows indicates how the different parts are linked.

Part I introduces the area of interest in the project, and provides an overview of previous literature within the relevant research field, together with an results based on preliminary data. Thus Part I formulates the scope of the project, and is summarized in the problem statement.

Part II describes the methods used both for the data collection of the preliminary data (previously collected by others) and the conducted experiments, together with methods for the data analysis.

Part III presents the results based on the preliminary data and likewise for the collected data.

Part IV rounds of the main part of the report containing the discussion of the results and an outline of future work.

After the main report follows the appendix in part V, containing chapters that elaborate some of the presented information in the report, and additionally new information is introduced. No information is contained in the appendix, which is not referred to in the main report.

A CD is enclosed containing the application written for the experimental part together with the collected data on which the data analysis is based.

Selected terms throughout the report are explained in the glossary.



# Contents

<b>1 Reading instructions</b>	<b>iii</b>
<b>Glossary</b>	<b>5</b>
<b>I Introduction</b>	<b>7</b>
<b>2 Paralysis and rehabilitation</b>	<b>9</b>
2.1 Upper extremity movements in everyday living . . . . .	9
2.2 Causes and consequences of movement impairment . . . . .	9
2.3 Rehabilitation for movement impaired individuals . . . . .	11
2.4 Functional components of a FES system . . . . .	14
2.5 Decoding based on a multi-modal approach . . . . .	19
2.6 Reduction of the complexity for movement control . . . . .	24
<b>3 Problem statement</b>	<b>25</b>
<b>II Methods</b>	<b>27</b>
<b>4 Decoding of the users intent</b>	<b>29</b>
4.1 Extraction of muscle activation level from the EMG . . . . .	29
4.2 Decoding based on the Kalman filter . . . . .	30
4.3 Decoding based on a multi layer perceptron network . . . . .	32
4.4 Extraction of muscle synergies and motor commands . . . . .	37
4.5 Evaluation of predictions . . . . .	39
<b>5 Protocols</b>	<b>41</b>
5.1 Preliminary data . . . . .	41
5.2 Conducted experiments . . . . .	43
<b>III Results</b>	<b>55</b>
<b>6 Preliminary decoding results</b>	<b>57</b>
6.1 Prediction of kinematics and muscle activity . . . . .	57
<b>7 Experimental results</b>	<b>59</b>
7.1 Recorded data . . . . .	59
7.2 Prediction of muscle activity . . . . .	60
7.3 Prediction of motor commands . . . . .	63

7.4	Prediction of kinematics . . . . .	65
7.5	Summary . . . . .	67
<b>IV</b>	<b>Recapitulation and future work</b>	<b>69</b>
<b>8</b>	<b>Discussion</b>	<b>71</b>
	<b>Bibliography</b>	<b>75</b>
<b>V</b>	<b>Appendix</b>	<b>85</b>
<b>A</b>	<b>Literature overview</b>	<b>87</b>
<b>B</b>	<b>Upper limb movement</b>	<b>93</b>
B.1	Muscles responsible for arm reaching . . . . .	93
B.2	Remaining movement abilities under different levels of injury . . . . .	94
<b>C</b>	<b>Photos of experimental setup</b>	<b>95</b>
C.1	Equipment . . . . .	95
C.2	Montage of sEMG electrodes . . . . .	96
C.3	Placement of reflective markers for motion capture . . . . .	96
<b>D</b>	<b>Optimization of the neural network</b>	<b>99</b>
D.1	Prediction of muscle activity . . . . .	99
D.2	Prediction of kinematics . . . . .	101
<b>E</b>	<b>Selection of number of synergies</b>	<b>103</b>
E.1	Reconstruction performance as a function of number of synergies . . . . .	105
<b>F</b>	<b>Introduction to neural networks</b>	<b>109</b>
F.1	Multilayer perceptron neural network . . . . .	109
F.2	Structure of MLP NN . . . . .	110
F.3	Learning of a neural network . . . . .	112
<b>G</b>	<b>Configuration of applied eyetracker</b>	<b>119</b>
G.1	Eyetracker Tobii X120 . . . . .	119
<b>H</b>	<b>Trigger application</b>	<b>123</b>
H.1	Required software . . . . .	123
H.2	Interfacing the parallel port . . . . .	123
H.3	Tobii Studio . . . . .	124
H.4	Synchronization via software . . . . .	125

# List of Figures

1.1	Report structure . . . . .	iii
2.1	The vertebral column . . . . .	10
2.2	Schematic representation of the functional components of a FES system . . . . .	15
2.3	Prediction of muscle activity from preliminary data . . . . .	20
2.4	Prediction of muscle activity from preliminary data . . . . .	21
2.5	Prediction performance of muscle activity from preliminary data, single joint movement . . . . .	22
2.6	Prediction of kinematics from preliminary data, multi joint movement . . . . .	22
2.7	Prediction performance of the kinematics from preliminary data . . . . .	23
3.1	Project aims . . . . .	26
4.1	Decoding of the users intent . . . . .	30
4.2	Generating the linear envelope of the EMG . . . . .	30
4.3	Underlying model of the Kalman filter . . . . .	31
4.4	Kalman filter algorithm . . . . .	31
4.5	MLP NN with one hidden layer and one output layer . . . . .	34
4.6	Learning algorithm of a Neural Network . . . . .	35
4.7	Extraction of synergies and motor commands . . . . .	37
4.8	Reconstruction of muscle activity from muscle synergies and motor commands . . . . .	38
4.9	Simplified approach for neural control . . . . .	38
4.10	Evaluation of predictions . . . . .	39
5.1	Elbow movements at three different fixed shoulder angles, preliminary data . . . . .	41
5.2	Multi joint movements, preliminary data . . . . .	41
5.3	Position of reflective markers, preliminary data . . . . .	42
5.4	Outline of predictions of muscle activity and kinematics . . . . .	42
5.5	Experimental setup . . . . .	43
5.6	Audible movement cue indicating start and stop of movements . . . . .	44
5.7	Goal directed movements . . . . .	44
5.8	Electrode placement . . . . .	46
5.9	Position of reflective markers for motion tracking . . . . .	47
5.10	Setup of eyetracking . . . . .	47
5.11	Synchronization of the sEMG, motion capture, and eyetracking . . . . .	49
5.12	Prediction performance of MLP NN as a function of number of neurons (muscle activity) . . . . .	50
5.13	Prediction configurations . . . . .	52
5.14	Reconstruction performance as a function of number of extracted synergies and motor commands . . . . .	53
6.1	Comparison of preliminary results of predictions during single joint movement . . . . .	57
6.2	Comparison of preliminary results of predictions during multi joint movement . . . . .	58

7.1	Recorded muscle activity, subject 1 . . . . .	59
7.2	Recorded gaze-direction, subject 1 . . . . .	60
7.3	Evaluation of muscle activity predictions, subject 1 . . . . .	61
7.4	Prediction of muscle activity for all muscles, subject 1, trial 1 . . . . .	62
7.5	Predicted motor commands and extracted synergies, subject 1, trial 1 . . . . .	63
7.6	Predicted motor commands and extracted synergies, subject 1, trial 2 . . . . .	64
7.7	Predicted motor commands and extracted synergies, subject 1, trial 3 . . . . .	65
7.8	Evaluation of kinematic predictions, subject 1 . . . . .	66
7.9	Predicted position in $x$ , and $y$ -direction, subject 1 . . . . .	66
8.1	Recapitulation of project . . . . .	71
B.1	Anatomy of muscles responsible for arm reaching . . . . .	93
C.1	Experimental equipment . . . . .	95
C.2	Montage of the sEMG electrodes . . . . .	96
C.3	Placement of the reflective markers . . . . .	97
D.1	Prediction performance of MLP NN as a function of number of neurons (muscle activity) . . . . .	100
D.2	Prediction of muscle activity in Pectoralis using 3, 10, and 20 neurons in hidden layer . . . . .	101
D.3	Prediction performance of MLP NN as a function of number of neurons (kinematics) . . . . .	102
D.4	Prediction of kinematics using 1, 10, and 20 neurons in hidden layer . . . . .	102
E.1	Recorded muscle activity, subject 1, trial 1 . . . . .	103
E.2	Recorded muscle activity, subject 1, trial 2 . . . . .	104
E.3	Recorded muscle activity, subject 1, trial 3 . . . . .	105
E.4	Reconstruction performance as a function of number of extracted synergies and motor commands . . . . .	106
E.5	Reconstruction performance as a function of number of extracted synergies and motor commands . . . . .	107
F.1	Typical neuron of MLP network . . . . .	110
F.2	Single layer MLP NN with $S$ perceptrons in hidden and output layer . . . . .	111
F.3	MLP NN with one hidden layer and one output layer . . . . .	111
F.4	Learning algorithm of a Neural Network . . . . .	112
F.5	Extended multilayer network . . . . .	114
F.6	Backpropagation path from additional layer to output layer . . . . .	114
F.7	Backpropagation path from output layer to hidden layer . . . . .	115
G.1	Setup of eyetracking . . . . .	119
G.2	Active eyetracking display area . . . . .	120
G.3	Side offset configured in Tobii Studio . . . . .	120
G.4	Optimal configuration of eye-tracker . . . . .	120
G.5	Calibration grid in a perspective calibration . . . . .	121
G.6	Calibration plot from eye-tracker . . . . .	121
H.1	Detailed representation the synchronization of the recording systems . . . . .	123
H.2	Parallel port . . . . .	124
H.3	Possible synchronization delay between the recorded signals . . . . .	125

# List of Tables

2.1	Literature overview of studies using sEMG for decoding . . . . .	19
5.1	Variations of the input and output for the different prediction configurations . . . . .	51
7.1	Variations of the input and output for the different prediction configurations . . . . .	60
B.1	Primary muscles for reaching movements in the horizontal plane. . . . .	94
B.2	Remaining movement abilities under different levels of injury . . . . .	94



# Glossary

- ALS** Amyotrophic lateral sclerosis. A progressive and fatal neurodegenerative disease caused by the degeneration of motor neurons, the nerve cells in the central nervous system that control voluntary muscle movement, with the result of losing the ability of the brain and spinal cord to incite muscular activity and respiration, page 10
- ANN** Artificial Neural Network is an adaptive, and most often nonlinear system that learns to estimate a function from data or associate input vectors with specific output vectors and is based on the operation of biological neural networks, page 34
- ANN** Artificial Neural Network is an adaptive, and most often nonlinear system that learns to estimate a function from data or associate input vectors with specific output vectors and is based on the operation of biological neural networks, page 113
- BCI** Brain computer interface. A direct brain-computer interface is a device that provides the brain with a new, non-muscular communication and control channel [172], page 16
- Cerebral Palsy** Refers to one of a number of neurological disorders that appear in infancy or early childhood and permanently affect body movement and muscle coordination without worsen over time. Cerebral palsy is caused by abnormalities in parts of the brain that control muscle movements [100], page 11
- Co-contraction** Muscle co-contraction can be defined as the simultaneous activation of agonist and antagonist muscle groups crossing the same joint and acting in the same plane [83], page 25
- D.O.F** Degrees of freedom. Represent the number of independent movements an object can make. For a biological motor system it is the number of dimensions in which the system can independently vary., page 14
- DVT** Deep vein thrombosis. The formation of a blood clot, thrombus, in a deep vein with the most severe complication being a clot traveling to the lungs causing a pulmonary embolism, page 11
- EEG** Electroencephalography. The measurement of the electrical activity generated by the brain, page 16
- Endpoint stiffness** The relationship between externally imposed hand displacements and the resultant forces. Hence arm resistance to applied perturbations. The endpoint stiffness determines how strongly external disturbances are rejected during maintenance of posture or movement, page 14
- FES** Functional Electrical Stimulation. Electrically elicited muscle contractions coordinated in a manner that provides function[122], page 12
- FET** Functional electrical therapy, page 12
- Forward dynamics** Dynamics is the study of bodies in motion. Forward dynamics is based on determining the movement based on the forces., page 17
- Goal-directed movement** Goal-directed movement is a type of movement of high prevalence in our everyday life. Goal-directed movements are typically directed toward one of a (possibly large) number of discrete goals available in the subject's workspace. These goals may be visual targets presented on a computer screen or physical objects located near the subject. Furthermore repeated movements to the same goal are not identical. E.g. there may be variability in movement speed or curvature. Finally the trajectories generally start at rest, proceed out to the desired goal, and end at rest [178], page 9
- Inverse dynamics** Is the transformation from a desired motion (input is position, velocity, and acceleration of each limb segment together with external forces), to the muscle forces or activations that are needed to drive the limb according to the desired motion [10], page 23
- Jacobian matrix** The Jacobian is the matrix of all first-order partial derivatives of a vector-valued function, page 38
- Kinematics** Motion description. E.g. speed, position, or angular velocity., page 9
- Lower motor neurons** The motor neurons connecting the brainstem and spinal cord to muscle fibers, bringing the nerve impulses from the upper motor neurons out to the muscles. The axon's from the lower motor neurons terminates on an effector, the muscle. The upper motor neurons originates from the motor region of the cerebral cortex or the brain stem and carry motor information down to the final common pathway (the motor neurons not directly responsible for stimulating the target muscle), page 13
- MI** Primary motor cortex mediates voluntary movements of the limbs and trunk. It is called primary because it contains neurons that project directly to the spinal cord to activate somatic motor neurons, page 16
- MU** Motor unit consist of an  $\alpha$ -motor neuron (final point of summation for all descending and reflex input) in the spinal cord and the muscle fibers it innervates [89], page 25
- MUAP** Motor unit action potential is the combination of the muscle fiber action potentials from all the muscle fibers of a single motor unit, i.e. individual MU action potentials [171], page 31
- Multiple sclerosis** Multiple sclerosis, also known as disseminated sclerosis, is an autoimmune condition in which the immune system attacks the central nervous system, leading to demyelination [136], page 10
- Muscle synergy** A set of relative non-negative levels of muscle activation that recruits a group a muscles in a coordinated manner, page 26

<p><b>Muscular dystrophies</b> Genetic, hereditary muscle diseases that cause progressive muscle weakness. Muscular dystrophies are characterized by progressive skeletal muscle weakness, defects in muscle proteins, and the death of muscle cells and tissue, page 11</p> <p><b>NMF</b> Non-negative Matrix Factorization, page 39</p> <p><b>omnidirectional</b> Omnidirectional refers to the notion of existing in every direction, page 17</p> <p><b>Paraplegia</b> Paralysis that originate from the thoracic segments (T2-S5), page 10</p> <p><b>prehension</b> The act of grasping or seizing, page 14</p> <p><b>Proprioception</b> Transduce information about the relative configuration of the body segments. Proprioceptors induce changes in muscle activity that bypass consciousness (reflexes). They let us know where arm and legs are as well as how heavy or light. They help in the creation of an internal reference system that the brain uses to plan and execute movements [75], page 9</p> <p><b>SCI</b> Spinal cord injury, page 9</p>	<p><b>sEMG</b> Surface electromyogram comprises the sum of the electrical contributions made by the active motor units as detected by the electrodes placed on the skin overlying the muscle [33, 88], page 16</p> <p><b>Sigmoidal function</b> A sigmoid function refers to the mathematical function which produces a S-shaped sigmoid curve. The sigmoid function refers in this work to the logistic sigmoid function [132], page 114</p> <p><b>Taylor series</b> The Taylor series is a representation of a function as an infinite sum of terms calculated from the values of its derivatives at a single point, page 38</p> <p><b>Tendon transfer</b> Tendon transfer surgery is a type of surgery performed in order to improve a lost function e.g. due to a injury in the nervous system. A functioning tendon is shifted from its original attachment to a new one to restore the action that has been lost, page 14</p> <p><b>Tetraplegia</b> Paralysis that originate from the cervical segments (C1-T1), page 10</p>
---	--

---

# Introduction

---

**I**



Movement disability is one of the most invalidating conditions in consequence of a spinal cord injury (SCI). Paralysis often imply loss of independence and good health since the paralyzed individuals loose their abilities to interact with their surroundings and are unable to complete many activities critical for independent living, such as eating, reaching and acquiring objects. Tasks, which are considered simple and taken for granted by healthy people.

Paralysis can occur from different causes, among others stroke and fatal accidents resulting in SCI and can be devastating dependent on the level of injury on the spinal cord, causing a near total loss of independence in the case of high-level injury. Unfortunately, neurons generally exhibit little regenerative capacity, resulting in permanent damage to the nerves. Hence a lot of focus has been aimed at rehabilitation technologies to restore lost functions.

This chapter introduces upper extremity movements that are essential in our everyday living, different degrees of paralysis, and the consequences for motor impaired individuals and for the society. Also different rehabilitation technologies for paralyzed individuals are introduced.

## 2.1 Upper extremity movements in everyday living

One category of movements are goal-directed, which characterizes most upper limb movements, in the contrary to cyclic movements, e.g. as seen during gait [20]. Goal-directed movements are an essential part of everyday living such as reaching, grasping, pointing, throwing, drawing, handwriting, keyboarding, tracking, object manipulating etc. all being different modalities of goal-directed movements. This kind of movement is a planned change of arm and hand segments positions, leading to a task, and depends on a balance of initial programming and subsequent correction. Goal-directed movement is a complex sequence of kinematic events, which can be divided into different phases [81]. The initial programming is based partly on the visual perception of an object and partly on proprioception [122]. Generating movement requires the nervous system to integrate multi-modal sensory information, make complex transformations between different spatial and intrinsic coordinate systems, and to generate motor commands adequate to attain the arm movement [122].

Reaching in specific is described as an arm manipulation between two points. It is a complicated multi-joint movement directed to a defined point in space performed by coordinated rotation at the shoulder and elbow joints. Reaching movements have an initial acceleration phase and a final acceleration phase, resulting in a bell-shaped velocity profile. The hand follows relatively simple and often straight paths in space.

The reaching movement comprises several different activities, each having specific characteristics that require different feedback mechanisms. E.g. reaching in a tracking activity consists of movement of the hand along a prescribed trajectory. Visual information during reaching to a target is used to identify the target and its location in space, and for corrections of ongoing movement. [122]

Many of the basic movements that a paralyzed individual would desire are simple goal-directed movements, such as reaching, pointing and grasping, which can be considered to be the building blocks of more complex actions, such as eating.

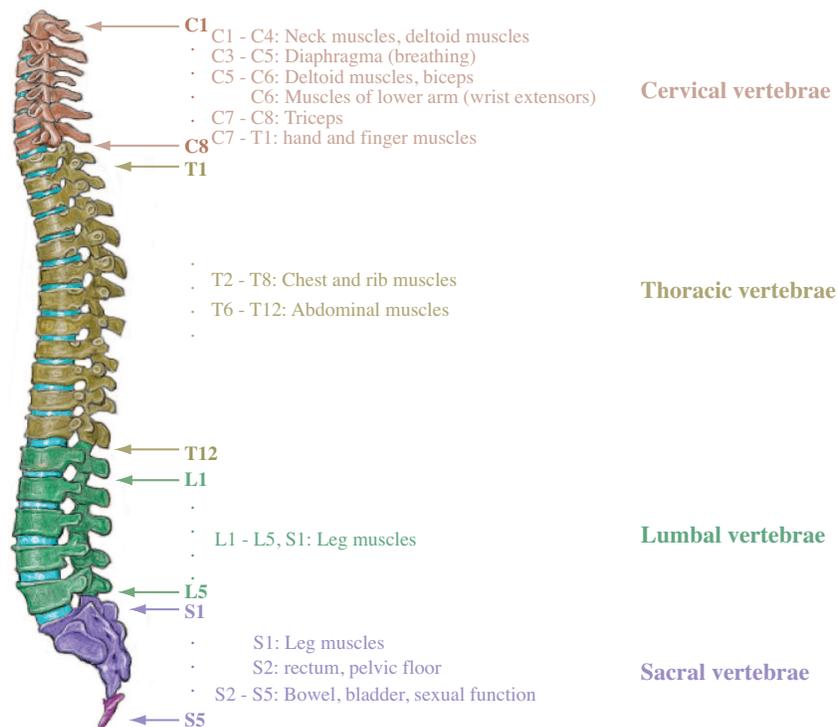
Movement impairment may have devastating consequences for individuals, who can no longer perform tasks, which previously were taken for granted. Causes and consequences of paralysis are the focus of the following section.

## 2.2 Causes and consequences of movement impairment

The spinal cord consists of spinal nerve fibers that transmit sensory information from the body to the brain and motor information from the brain to the body. Injury to the spinal cord results in interruption of the transmission of this information. SCI refers to any injury of the neural elements within the vertebral column and can occur from either a traumatic injury or from a disease to the vertebral column.

Traumatic injury can e.g. occur from motor vehicle crashes, falls, and acts of violence or sports activities. Often SCI's due to traumatic injury lead to the backbone pinching the spinal cord, causing bruising or swelling. The injury can tear the spinal cord and the nerve fibers. [101] Infectious conditions being infections involving the spinal canal (including epidural abscesses (infection in the epidural space), meningitis (infection of the meninges), subdural abscesses (infections of the subdural space), and intramedullary abscesses (infections within the spinal cord), tumors, multiple sclerosis, Amyotrophic lateral sclerosis (ALS), brain stem infarcts are some of the pathological conditions that can cause SCI. [101, 102, 114]

The neurological level (referring to the location of the vertebral column, numbered and named according to their location c.f. figure 2.1) and the extent of the lesion determines the consequences of a SCI. The higher the level of injury, the more extended the paralysis and sensory loss [101]. The highest level of injury is categorized from injuries at the cervical segments (C1-T1), as tetraplegia, which causes loss of functions in almost every extremity of the body. Individuals with high-level tetraplegia (defined from the fourth cervical level (C4) and above) would typically have limited upper extremity movements, retaining voluntary control over head and neck muscles and in some cases have minimal control of shoulder movement through trapezius activity [2, 101]. C4 tetraplegia is the highest level of paralysis without the individuals being ventilator dependent. In addition tetraplegic individuals with C3-C4 SCI exhibit extensive denervation of the shoulder and elbow muscles [122]. At a lower level of injury, the paralysis is categorized from injuries at the thoracic segments (T2-S5) as paraplegia causing paralysis of the lower extremities [101].



**Figure 2.1:** The vertebral column and spinal cord can be divided into four regions, Cervical, Thoracic, Lumbar, and Sacral based on the regions in which the vertebrae are positioned and where the nerves from the spinal cord leaves the vertebrae. Modified from [46, 2]

SCI's are classified, as a complete injury when all neurons are cut and no link between the brain and the body periphery exist, thereby lacking the possibility to recover the functions of the limbs below the level of injury. In a complete injury there is no motor or sensory function in the S4, S5, or anal area. In an incomplete injury, some control of the body may still remain and offers possibilities to recover some functions below the level of injury. [8, 101]

All types of paralysis affect the paralyzed individual's ability to interact with their environment and clearly a complete

injury resulting in high-level tetraplegia being the most severe case, with no control of either the lower or upper extremities and often they lack the possibility to breath voluntarily. Consequences of SCI can be wide spread, and several functions may be affected as well as several complications may occur as listed in the following [8, 127, 145]

- Paralysis with muscle atrophy and spasticity
- Risk of pressure ulcers
- Unable to control bladder and bowel, risk for urinary tract infections
- Changes in sexual function
- Endocrine and metabolic changes due to the physical inactivity
- Bone decalcification with extended risk of fractures, osteoporosis
- Affection of blood vessel and blood pressure (hypotension and severe hypertension (autonomous hyperreflexia), and extended risk of deep vein thrombosis (DVT))
- Depression
- Diabetes mellitus
- Increased body fat mass
- Cardiovascular deconditioning

According to [2] approx. 3000 individuals suffer from para- or tetraplegia in Denmark (the number is uncertain due to no central registration). Out of 1000 persons each year in Denmark subject to SCI approximately 10% will be cervical fractures [136]. Approx. 75% of the paralyzed individuals are young males with no cognitive impairment and with a long life expectancy, approaching 40 years for injuries incurred at the age of 20, after the time of injury [102]. These individuals are entirely dependent on outside assistance for largely all activities essential for daily living, including eating, personal hygiene, bladder and bowel care and transferring in and out of bed.

This independence has high economical cost for the society. According to [102] the average lifetime medical cost (average yearly health care and living expenses) for an individual injured at the age of 25 exceeds \$3 million in USA, excluding indirect costs such as lost wages and the impact on the individual's family<sup>1</sup>.

Other causes of movement impairment can be head injury, cerebral palsy, advanced-stage muscular dystrophies, and diseases of the neuromuscular junction [114].

The succeeding section provides insight into a range of rehabilitation options for persons suffering from paralysis focusing mainly on high-level tetraplegia.

## 2.3 Rehabilitation for movement impaired individuals

Damage to nerves is in most cases permanent, since neural tissue generally exhibit limited regenerative capacity. In contrast to advances in the care of SCI only little has been achieved clinically to reverse the neurological loss associated with SCI by protection or regeneration of axons within the spinal cord. [58] The neurological outcome of SCI is first and foremost determined by the extent of the damage caused at the moment of injury and still today no treatment can change that outcome [127]. Consequently, persons suffering from SCI must contend with multiple neurological sequelae of SCI such as paralysis, sensory loss, autonomic dysfunction, loss of bladder and bowel control [127].

Rehabilitation devices that enable paralyzed people to perform daily activities or restore even a modest level of independence have the potential to generate an enormous impact on quality of life and in addition lower the related costs to society [23]. The goal of rehabilitation is to enable movement impaired individuals to regain the highest possible level of independence.

---

<sup>1</sup>These statistics is not available in Denmark or Europe[14]

Numerous approaches have been taken to develop technologies useful in the area of rehabilitation of paralyzed individuals to facilitate independent communication and mobility, among these the mouth stick, sensors activated by eye blinks, respiration or head movement [77, 145]. In severely motor impaired individuals, only a few modalities remain under voluntary control, which complicates the opportunity for these individuals to use assistive technology to allow them to independently perform movements essential for many self-care activities.

The following subsection focuses on a promising approach to improve independence of severely movement impaired individuals, namely neurorehabilitation [145].

### **Neurorehabilitation**

Neurorehabilitation comprises methods and technology for maximizing the efficiency of preserved neuromuscular structures in individuals suffering from motor disabilities. Thus neurorehabilitation relates to developing new movement strategies based on preserved sensory and motor systems, which without training otherwise would remain unused. Individuals subject to SCI can have intrinsic resources that may still be available if they are developed appropriately and used in a functional manner. The effectiveness of neurorehabilitation activities is depended on the degree of disability and to the specificity of losses of neural connections. [122]

Spontaneous recovery of functions occurs through the process of compensation, substitution, and dynamic reorganization through training. Organization of learning within the cerebral cortex results from the repeated activity on the part of the individual and becomes organized into a functional system of behavior. The repeated activities can also change the neuronal connections and function at lower levels of the central nervous system. These processes lead to new types of integrated multiple sensory modalities, which directly affect the abilities for voluntary motor control.[122]

**Functional electrical therapy** Neuromuscular stimulation for motion can be applied for both therapeutic and functional purposes. Therapeutic purposes, functional electrical therapy (FET), includes clinical interventions from simple exercises for muscle conditioning through motor relearning.

FET is one strategy to assist individuals in executing functional movement after SCI. FET is suitable for individuals who have reduced voluntary control of muscles due to a prolonged fixation of the joint [128]. This approach facilitate voluntary movement by learning the diminished movement by electrical stimulation allowing regaining of voluntary control of some muscles [122].

The purpose of FET is to improve tissue health or voluntary function by inducing physiological changes, which remain after stimulation [114]. In addition, chronic use of FET has proved to have a positive effect on the bone density [92]. [58] has recommended regular FET to maintain the optimal physical and psychological condition of individuals suffering from SCI. Regular sensory stimulation through the skin maintain the retained neurological functions and returning reflexes closer to the normal physiological state. In addition [49] speculated that the normal neural circuitry within the spinal cord, necessary for gait and other motor functions, needs continuous maintenance by FES and/or body weight supported treadmill, training ambulation. Also the patterned neural activity that FET comprises has been suggested to be important for both development and recovery of neurological functions [86].

Furthermore, physical exercise both stimulates the productions of endorphins and contributes to the upregulation of brain derived neurotrophic factor, which may promote synaptic and functional plasticity within the brain and spinal cord [161].

This suggests that the therapeutically effects of electrical stimulation of paralyzed muscles may have an substantial effect on both the peripheral system in form of strengthening the muscles, increasing the range of movements, and decreasing spasticity, and contemporary on the functional organization of the CNS.

**Functional electrical stimulation** Functional electrical stimulation (FES) facilitate directly specific functions in daily living.

Paralyzed or paretic muscles can be made contract by applying low electrical currents to the intact peripheral motor nerves innervating them. When electrically elicited muscle contractions are coordinated in a manner that provides function, it is called FES. It is the application of low electrical currents to excitable (neuromuscular) tissue to supplement or replace function that is lost in neurologically impaired individuals [114].

The purpose of FES is to enable function by replacing or assisting a person's voluntary ability. Neuroprostheses are FES devices used as a substitution for lost neurological function [114, 122]. Both sensory, e.g. auditory, visual, and motor function can be restored with FES. However only neuromuscular FES, for restoration of motor function, is considered in this work.

FES for restoration of motor function has numerous physical advantages including preventing DVT [90], preventing pressure ulcers [22], increasing muscle strength by exercise of muscles post SCI [126, 131], increase muscle mass [138] thereby being effective in preventing muscle atrophy, rebuilding muscles from an atrophied stage, increase the range of motion, inhibit spasticity, and reeducate voluntary muscles, increase cardiac output, the peripheral venous blood return, and muscle oxidative capacity [114].

### Neuroprostheses

A neuroprosthesis replaces or augments a function that is lost or diminished because of an injury or disease to the nervous system. The basic principle of a neuroprosthesis is the stimulation of neuromuscular tissue.

Neuroprostheses operate through a command interface that measures some modality over which voluntary control is maintained, and translates this to a specific operation of the prosthesis [118]. Several control modalities have been applied such as residual movement, EMG, respiration, voice commands, and brain activity. FES devices used as a substitution for lost neurological function are called neuroprostheses. [114]

After SCI muscles below the level of injury are still connected to the CNS but will no longer be under voluntary control. Movement restoration can be achieved by the use of these muscles by by-passing the damaged motor structure, applying electrical currents to lower motor neurons using electrodes placed on or near the innervating nerve fibers for the specific muscles.

Thus, for FES to be effective, the lower motor neurons must be intact (excitable) from the anterior horns of the spinal cord to the neuromuscular junctions in the muscles, which are to be activated [114]. This is usually the case with SCI, stroke, head injuries, cerebral palsy, and multiple sclerosis [114]. Most neuroprostheses for restoration of motor function have been primarily targeted toward SCI's.

Stimulating the innervating axons appropriately can elicit action potentials, and the strength of the muscle contraction is regulated by modulating stimulus parameters (e.g. amplitude and duration of the stimulus pulses, while the frequency is often kept constant) [122]. Coordination of several electrically activated muscles can produce a functional limb movement.

A lot of development and progress have been seen in the field of FES in restoring upper limb functions but the main part of the studies have been demonstration studies [145]. The objective with upper limb FES is to promote function, such as reaching and grasping.

Especially upper limb FES applications for control of the hand in individuals with C5-C6 tetraplegia have been in focus. C5-C6 tetraplegic individuals generally retain enough voluntary control over shoulder and elbow joints to place the hand in positions below the level of the shoulder, thus retaining basic reaching abilities, which makes the target joints for FES in this population wrist, fingers, and thumb for grasp and release [96]. E.g. the FreeHand system has been developed to allow C5 or C6 tetraplegic individuals to restore abilities to grasp, hold, and release objects [61]. The system uses an implanted receiver-stimulator and eight-channel electric stimulation of the grasping muscles of one arm is controlled by using contra lateral shoulder movements. Results have shown positive effects in form of significantly improved pinch force and improvement in grasp-release abilities [66, 113]. [113] presented results showing that the majority (87% of 34 individuals) were satisfied with the FreeHand system and stated that it improved their activities in daily living. Additionally it was reported that the system had a positive impact on their life (88%).

[125] developed an external device, using three channels for stimulating finger flexors, extensors, thumb flexors, with the a control signal based on wrist position transducer. The Bionic glove FES system was shown in a study applied on 12 C5-C7 tetraplegic individuals to increase the subject's power grasp and range of movements [121]. [147] investigated the functional effects of the NESS Handmaster in C4-C6 tetraplegia individuals. Results showed that the Handmaster had functional benefit in C5 tetraplegic individuals, though with the requirement of sufficient shoulder

and biceps function combined with absent or weak wrist extensors. [82] tested the benefit of a transcutaneous FES system to improve grasp function in 11 C4-C7 tetraplegic individuals. Eight out of a total of nine who used the FES as neuroprosthesis showed improved grasp function and performance in activities of daily living.

In FES the placement of an electrode on the triceps has become a common option to improve the control of reaching, promoting elbow extension through triceps stimulation. [21, 42, 13] showed among others, that elbow extension enhances tetraplegic individuals ability to grasp and manipulate objects. [117] showed that triceps FES improved postural arm stability evaluated by endpoint stiffness, which characterizes the relationship between externally imposed hand displacements and the resultant forces (estimated during application of planar, stochastic perturbations).

[87] showed that successfully reaching and moving an object were improved and the time required to acquire an object while reaching was significantly decreased. 11 arms of persons with SCI received a triceps electrode as an addition to a hand-grasp neuroprosthesis. Triceps stimulation provided a significantly stronger elbow extension moment than a posterior deltoid to triceps tendon transfer. The elbow extension moment generated through simultaneous activation of the tendon transfer and triceps electrode was always greater than either method used alone. The elbow extension resulted in a decreased amount of time required to acquire an object while reaching and significantly increased the ability to successfully reach and move an object.

[168] showed that individuals suffering from tetraplegia are more independent in self-care tasks with active control of elbow extension compared to tetraplegic individuals with functioning wrist extension without active elbow extension.

However restoring of arm reaching in high-level tetraplegic individuals is more complicated due to the very few movements that remains under voluntary control (such as shoulder elevation, head movement, voice, and respiration) and to the fact of extensive denervation of shoulder and elbow muscles important for controlling the proximal limb. In addition FES systems for this population should provide control of the finger, thumb, wrist, elbow, and shoulder joints[96].

For high-level tetraplegic individuals to achieve the ability to position their hand in the workspace at a desired object, a simultaneous control of at least two degrees of freedom (D.O.F) (translation and grasping) is required[118]. Only few have used FES for restoration of arm function in this population, which might be attributable to the difficulties in determining subjects intent in multiple D.O.F. tasks and to select the appropriate patterns of muscle coordination for restoring the intended movements.

Multi-joint neuroprosthesis for high-level tetraplegic individuals have previously combined slings and balanced forearm orthoses with FES [146] allowing individuals to grasp an object from a horizontal workspace. [53] first demonstrated that FES could be used for the restoration of reach control in high-level tetraplegic individuals by pre-programmed control of elbow extension/flexion or hand grasp/release controlled by detection of respiration (inspiration/expiration).

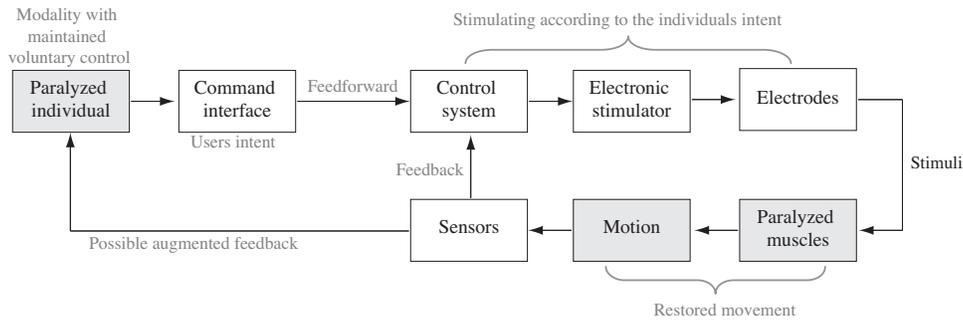
[146] developed a FES system that allowed one complete C4 tetraplegic individual to control the level of stimulation and choose between two pre-programmed stimulation patterns (both designed to couple hand and arm movements to allow hand-to-mouth activities) for hand grasp and release, elbow movement, and arm adduction, using contra lateral shoulder movement and combined the FES with a suspended sling to provide stability of the glenohumeral joint, since the rotator cuff and deltoid muscles (muscles acting to stabilize the shoulder) could not be stimulated due to denervation. [99] developed a similar FES system using vocal commands triggering preprogrammed hand prehensions, arm motion, and other functions.

In these approaches a pre-programmed set of muscle stimulation patterns was used and some also require assistance to place the object in the user's hand or at a specific location in the workspace, which limits the functional use of such devices.

The following section presents the functional components of a FES system.

## 2.4 Functional components of a FES system

Figure 2.2 shows a schematic representation of a FES system for restoration of movement.



**Figure 2.2:** Schematic representation of the functional components of a FES system. The users intent is decoded and converted into control commands in the command interface from signals originating from some modality of which the user has retained voluntary control. The control system sends command signals to the electronic stimulator, which in turn generates trains of pulses of electrical charge and deliver these to the excitable tissue via the electrodes. Eventually this creates the movement, and feedback can be returned to the control system and possibly to the paralyzed individual. Modified from [122, 127]

Two fundamental components of a FES system are the decoder (the command interface in figure 2.2) for inferring the users intent and the controller (control system in figure 2.2) for controlling the stimulation of the targeted paralyzed muscles according to users intent. Thus, the decoder is a transducer that monitor signals produced voluntarily by the user.[122]

Subject-controlled FES can be open- or closed-loop. In open-loop controlled FES, the electrical stimulator controls the output whereas closed-loop FES employs sensors (e.g. potentiometers and goniometers) to facilitate greater responsiveness to prevent certain conditions such as muscle fatigue, or to irregularities in the environment, e.g. external perturbations.

Electrodes act as interfaces between the electrical stimulator and the nervous system. Surface electrodes are used exclusively for routine physical therapy. FES electrodes can be external (surface) or surgically implanted depending on the application, the device, and the patient's needs. [122, 127] The interface between the FES system and the user should allow for simple and intuitive control (decoding), which also allows for compensation of disturbances (control). Furthermore such interfaces should limit the surgery required for implanting FES systems [137].

Decoding of the users intent has been based on a large range of different approaches, among which some are introduced in the following subsection.

### 2.4.1 Decoding the users intent

In a FES system the initial step is decoding, which previously has been based on numerous different approaches. Decoding of the users intent besides FES is seen in other interfaces for rehabilitation devices including artificial limbs, wheelchairs, and communication aids.

Amplitude, speed, and direction of arm motions vary between different upper limb goal-directed movements, which place demands on both the decoder (command interface) and the controller in a FES system [20]. First of all the basis for allowing a variety of different movements is to separate these in the decoding process of the users intent based on the available signals, of which voluntary control still exists. Contemporary, the controller needs to provide a rich variety of stimulation patterns.

Interfaces that can be controlled via the subconsciousness allow the user to be more attentive to the task at hand instead of being occupied by controlling the neuroprosthesis. Letting the command control be a natural extension of the users intact motor system will increase the likelihood that the user can achieve some subconscious control [137].

The following two subsections provide an insight into some of the signals and methods utilized for decoding, and are divided into invasive and noninvasive approaches respectively.

### Invasive approaches

It was observed by [39] that the firing rate of cells in the primary motor cortex (M1) were approximated by a cosine 'tuning function', i.e. the direction of arm movement during reaching is coded by neurons with a firing rate that varies as the cosine of the angle between a cell's preferred direction where it is maximally active and the actual movement direction.

Later they proposed the population vector hypothesis [40] suggesting that the direction of hand movements could be predicted by combining data from many neurons, i.e. a population vector constructed from the firing rates of many cortical neurons tends to point in the direction of the hand movement. Thus, the activity in the primary motor cortex is believed to be related to the movement of the hand (higher-level features) and not to lower level features of the individual joints, such as the shoulder and elbow and muscles that makes up the movements as suggested by [156].

This framework of relating the neural activity and the hand motion has since then been subject to much work in the field of decoding. E.g. [148, 175, 178, 174, 176, 169, 140] have exploited the feasibility of decoding the user intention based on intra-cortical recordings from populations of neurons by implanted electrodes in motor cortex in non-human primates and in humans. It is discussed whether the motor cortex explicitly codes for higher task-level parameters (e.g. direction of hand movement) or for intrinsic parameters (e.g. muscle force)[93, 38].

Beside focusing on prediction of the limb kinematics and dynamics, the accuracy with which the muscle activity of arm and hand muscles can be predicted based on intra-cortical activity has also been investigated [119].

### Noninvasive approaches

Invasive approaches are impeded by the surgical risks of surgical implantation and by the substantial problems in achieving and maintaining stable long term recordings [135]. Also, it is recommended that the interfaces in FES systems should limit the surgery required for implanting the systems [137].

Even though noninvasive recordings provide less fidelity compared to invasive methods, they have great potential for use in FES and FET systems. Several approaches have been taken, e.g. by [53] who demonstrated control of a FES system by detection of respiration (inspiration/expiration), and by [57] showing that impaired shoulder movements and proprioception reduce the utility of shoulder movements for higher dimensional control in the C5-C6 population. The Electroencephalography (EEG) has likewise been used for decoding, often in brain computer interfaces (BCI's), e.g. [12, 143]. The EEG is easily contaminated from artifacts caused by e.g. eye- or head movement since the electrical activity is measured on the scalp. Even though it has been shown that this contamination can be reduced by signal processing [44], the application of EEG in decoding is often limited to separate classes of movement and thus is not considered for continuous decoding, such as e.g. the hand trajectory or muscle activity during reaching.

Surface electromyography (sEMG) is non-invasively recorded from the surface of the skin and the equipment for recording the signals are relative simple, e.g. there is no need for straps and harnesses [110]. sEMG has been utilized in many studies, an overview of a fragment of the literature is provided in table 2.1<sup>2</sup>, and has been used for decoding and discriminating between different movements in FES systems.

It emerges that especially discrimination between movement classes for the control of prosthesis, in particular the upper limb, has received considerable attention in previous research, e.g. [63, 52, 153, 65, 142, 17, 56]. Many of these approaches yields simplistic control schemes consisting of a limited number of controllable movements of the upper limb. Thus movements are selected among a few stored programs of muscle stimulation activated by a control signal of which the user has voluntary control.

This is an obvious limitation compared to the functionality of the natural upper limb. The number of controllable D.O.F is limited to the number of classes that can be distinguished in the EMG by pattern recognition, and thus pre-determined. In contrast [35] has proposed a method for setting a limited number of reference directions in a defined

<sup>2</sup>For more details, see table A on page 87

plane, and from these determine infinitely movement directions based on movement probabilities in those directions. This approach enabled EMG control of an omnidirectional pointer and it was demonstrated that the system could provide smooth pointing for an upper limb amputee. Hence they showed that the developed system was capable of pointing in any direction by expressing infinitely many movement directions based on a combination of a limited number of reference directions.

A different focus has been to predict kinematics data (e.g. from elbow and shoulder), i.e. angle, angular velocity and acceleration, hand (endpoint) trajectory e.g. [5, 69, 70, 29, 18, 94] from the sEMG. Kinematic predictions in the context of FES can be used for shaping the stimulus patterns to paralyzed limbs that are to be controlled and in the context of prosthetics; dynamics can be used directly for control of the joints.

Others have estimated patterns of muscle activity from joint kinematics e.g. [139, 3, 119], which can be utilized directly for electrical stimulation. Though in [139, 3] a selection of the desired trajectory should be completed by the paralyzed individual, which is clearly an obstacle for the target population. [139] suggested that selection of a desired trajectory by the paralyzed individual could be achieved by providing the user with a menu of selectable stored movements. An obvious drawback is related to the limited movements available for high-level tetraplegics, preventing them for using any movements for this selection, thus such an interface will imply cumbersome control of the FES system.

[53] have used the EMG from averaged EMG signals obtained from normal subjects to identify patterns of muscle activity associated with a particular movement, which then were applied in tetraplegic subjects for the control of a FES system. However with this method, the available movements elicited by the control signals showed to be limited to the motor task, from which the EMG signals were originally recorded.

Study	Subjects	Input	Aim	Output	Application	Evaluation
[119]	Two male rhesus monkeys	Neural activity	Linear system with multiple inputs and a single output, i.e. linear filtering	EMG from arm and hand muscles (Medial deltoid, biceps, triceps, and above the flexor musculature for the hand)	Prediction for use in FES for tetraplegics	Magnitude of coefficient of determination $R^2$
[5]	six able bodied subject and two C5 tetraplegic	sEMG	Time-delayed artificial neural network (TDANN)	Shoulder and elbow kinematics (angle, angular velocity, and angular acceleration) of the four joint angles (elbow flexion-extension and shoulder horizontal flexion-extension, elevation-depression, and internal-external rotation)	Prediction for use in FES for tetraplegics	RMS error
[69]	One Macaca fuscata monkey	Neural activity	Linear summation of the neuron activities and an ANN model	EMG (predicted from neural activity) and trajectories of the hand and elbow position (predicted from the estimated EMG)	BMI	Correlation coefficient for both the EMG prediction and the trajectory reconstruction
[3]	Five healthy subjects	sEMG	Patterns of muscle activity (EMG)	Probabilistic method based on probability distributions computed from training from one subject and applied to the remaining four subjects	Prediction of muscle patterns in a FES system	VAF and RMS error
[63]	One subject	sEMG	Features: discrete Hopfield network, classification: a two layer perceptron network	Classes of movement	Myoelectric signal analysis	Correct classification of arm functions
[94]	One subject	sEMG	ANN (joint torque) and a dynamic hand model (from joint torque to joint angles)	Joint torque	Controlling of a prosthetic hand by torque control of each joint	Visual comparison of joint torque, position (true and estimated)
[35]	Five healthy subjects and one subject with the forearm amputated	sEMG	Recurrent Log-Linearized Gaussian Mixture Network (based on a hidden Markov model) and an impedance model (inertia and viscosity is included)	Omnidirectional pointing device	Error between movement direction of pointing device and instructed direction	Direction error
[52]	-	sEMG	Multilayer perceptron neural network	Discrimination of limb functions	Limb function identification	Identification rate
[97]	Two amputee subjects and three healthy subjects	sEMG	Recurrent Neural network based on a hidden Markov model	Discrimination of motions (wrist flexion/extension, supination/pronation) and hand grasping/opening))	Human-machine interface for prosthetic control	Discrimination rate
[70]	One healthy subject	sEMG, velocity and position	Forward dynamics model, artificial neural network	Endpoint trajectories (dynamic joint torques (from ANN) at the elbow and shoulder was input to forward dynamics model)	characterization of the relations between muscle activity and dynamical kinematics	Coefficient of determination
[29]	Seven healthy subjects	sEMG	Back Propagation Through Time-ANN	Movement trajectory (x and y coordinates)	Devise a computational model of how motor activity of upper-limb functional movements is controlled in normal subjects and altered by different neuromuscular injuries and diseases	RMS error

[18]	Four healthy subjects	sEMG	Dynamic recurrent neural networks	Arm kinematics (x and z coordinates)	Identification of the relationship between EMG activity and arm kinematic	Visual inspection and robustness of model in terms of applying perturbation and investigate the resulting difference between the actual and predicted kinematics.
[153]	One healthy subject	sEMG	ANN (Non-linear decoding filters designed using multi-layer, feed-forward Artificial Neural Networks) for classification, feature extraction was based on four time-domain EMG features; Mean of the absolute value, Willison Amplitude, Variance, and Waveform length	Classes of movements	EMG based control for upper-limb prostheses	Discrimination rate
[65]	–	sEMG	Features: time domain (mean absolute value (MAV), zero crossing (ZC), Wilson amplitude (WAMP), slope sign changes (SSC) and coefficients of autoregressive model (AR)), time-frequency (short time Fourier transform (STFT), wavelets transform (WT), and wavelet packets transform (WPT)), dimensionality reduction: PCA and class separability, classification: ANN and fuzzy inference system	Discrimination of eight hand movements (hand opening and closing, pinch, thumb flexion, wrist radial flexion and extension and wrist flexion and extension)	Control for hand prosthesis	Discrimination rate
[142]	Three healthy subjects	Static sEMG	Features: RMS of windowed steady state EMG, classification: linear support vector machines	Discrimination of eight hand movements	EMG control of robotic Prostheses for amputees and partially paralyzed	Discrimination rate
[143]	One subject	EEG and sEMG	Dynamic Bayesian network	Discrimination between left and right hand movement	BCI	Discrimination rate
[17]	Eleven healthy subjects	sEMG	Pattern recognition: HMM	Identification of six distinct limb motions (wrist flexion, wrist extension, supination, pronation, hand open, and hand close).	Control of upper extremity prostheses	Discrimination rate
[54]	12 healthy subjects	sEMG	Gaussian mixture models	Multiple limb motion classification (wrist flexion, wrist extension, forearm supination, forearm pronation, hand open, and hand close)	EMG control of powered upper limb prostheses	Classification rate
[56]		sEMG	Features: Wavelet transform (variance, maximum and mean absolute value of the wavelet coefficients), classification: ANN	Identification of finger movements (middle finger flexion/ extension, index finger extension/ flexion, thumb extension/ flexion)	Control of prosthetic hand	Classification rate
[64]	Four healthy subjects	sEMG	Features: Local extrema and zero crossing of a new multi-wavelet function	Identification of different hand movements (hand opening and closing, pinch, thumb flexion, wrist radial flexion and extension and wrist flexion and extension)	Identification of hand movements	Classification rate
[53]	One C4 tetraplegic subject	Inspiration and expiration	Selection of predefined movements, trapezoidal approximation method of EMG activities	Multichannel stimulation patterns	Multichannel FES system for C4 tetraplegics	Capability of a C4 tetraplegic to conduct the predefined movements
[67]	One subject	sEMG	Linear multiple regression model	Joint angles (multi-finger angles corresponding to the different motions) to discriminate between different motions (grip, open, and chuck of a hand)	EMG control of upper limb prosthesis	Tracking error
[139]	One healthy subject	Hand kinematics	Probabilistic approach based on Bayes' theorem	Muscle activity and frequency modulated activity	Restoration of movement using FES	RMS error between desired movements and movements generated by electrical stimulation
[165]	Ten normally limbed subjects	sEMG	optimal wavelet packet method	Classification of different hand movements	Classification of sEMG	Classification rate
[164]	Five healthy subjects	sEMG	Features: square mean value, standard deviation and kurtosis index, Classification: Nearest neighbor statistical algorithm	Classification of different EMG patterns	Non-invasive HMI	Percentage error between classification and the desired command
[166]	Four healthy subjects	sEMG	Three layer ANN based on a modified Backpropagation algorithm and AR modeling	Identification of different finger motions	EMG based teleoperating a dexterous robot hand	classification rate
[31]	12 normally limbed subjects	sEMG	Features: time domain statistics, i.e. the number of zero crossings, the waveform length, the number of slope sign changes, and the mean absolute value in each analysis window, classification: linear discriminant analysis	Continuous classification of different hand movements	EMG control of powered prostheses.	Classification rate
[115]	Four healthy subjects	sEMG	K-nearest neighbor classifier, genetic algorithm for feature selection	Classification of different movements	Operating a robotic hand prosthesis	Classification error rate
[130]	Four subjects with limb deficiencies	sEMG	Classification: approximate maximum likelihood classifier, hierarchical classifier, fuzzy classifier, multilayer perceptron	Classification of different switch signals	EMG control of hand prostheses	Misclassification rates
[107]	Three normal subjects	sEMG	Feature extraction: Gabor transform and mean absolute value to extract information from the EMG, classification: Feed- forward neural network	Classification between different type of movements (wrist motions: supination, pronation, flexion and extension, hand motions: 2-5th fingers flexion, 2-5th fingers extension, thumb flexion, thumb extension, 4-5th fingers flexion, and 2-3rd fingers extension)	EMG control of prosthetic hand	Classification rate (Discriminating rates are calculated by the comparison between the instructions and the control commands in an ability test)
[32]	16 healthy subjects	sEMG	Features: time-frequency representations based on Hudgins' time domain features, the short-time Fourier transform, the wavelet transform, and the wavelet packet transform, dimensionality reduction: PCA and CS, classification: Linear discriminant analysis, Multilayer perceptron	Classification of different states	Classification of sEMG	Classification error
[41]	4 complete C5/C6 SCI subjects	sEMG	Two layer Neural network	Stimulation level	FES to employ triceps extension	Amount of stimulation and size of the produced force

**Table 2.1:** Literature overview of studies using sEMG for decoding.

The control methods in upper extremity neuroprosthesis have relied primarily on external joint angle sensors, accelerometers, or switches. One important area in the applicability of FES/FET systems is control algorithms that enable intuitive operation of the neuroprosthesis with little conscious attention [47]. As it appears from the outline above, only few have used the EMG for control in a FES and FET systems and most researchers have focused on discriminating among classes of movements, hence not providing continuous control.

The use of the sEMG for continuous control of a FES/FET system requires identification of appropriate control muscles contemporary with determining how to optimally extract useful information from the sEMG that can serve as indicator for the users intent [114, 120].

Moreover, extracting sufficient information from the available sources of the paralyzed individuals, e.g. sources of which voluntary control is maintained, is likewise essential. Especially in cases of the sEMG being somewhat inadequate in expressing the intent of the user, additional sources can presumably improve decoding.

## 2.5 Decoding based on a multi-modal approach

The user intent is in most FES systems decoded from single sources of information [112], e.g. the EMG as outlined above. Using only a single modality as a source of information may not be sufficient to adequately reduce the uncertainty in the decoded intention of the user [141].

On the contrary natural control of movement is based on multimodal approach, and to achieve accurate reach and grasp objects, gaze-direction is important since it reflects our attention, intention, and desire [16]. Some of the most common motor behaviors in everyday life are pointing and tracking objects under visual guidance, hence eye-hand coordination plays an important role in goal-directed reaches in everyday living, since the arm movements are typically associated with eye-movements toward the same goal in this kind of movement. It has been shown that performance (such as time and accuracy) improves when eyes and hand moves together [123, 85, 163, 103, 160, 30].

Thus detection of gaze-direction is believed to contribute with important (and additional) information compared to the sEMG. During reaching tasks, gaze anticipates the movements of the arms, thus giving useful information about the intention of the movement [158], and in fact the eye movements usually precede the motor acts they mediate by a fraction of a second [74]. [151] used gaze direction in combination with detection of the human body motion, which allowed them to disambiguate possible interpretation of the actions of the human. [116] showed a tight coupling of the gaze and hand movement during a natural task and furthermore [134] showed that the human brain utilizes continuous visual feedback from the hand to guide reaching movements throughout their extent.

[7] investigated the improving effect of neural prosthesis when gaze direction was accounted for in the operation of the prosthesis. It was shown that eye-position did improve the performance, defined as the accuracy of the desired endpoint. The decoding accuracy of the reaching endpoint was improved when eyetracking was included (by the use of eyetracking glasses) (66%, 59% on average across four start configurations) and when gaze-direction instead estimated from the neural activity (61%, 56%) was compared to a configuration without any eyetracking (45%, 44%).

The effect of including multiple modalities was investigated in a preliminary analysis based on recorded data from [95].

### 2.5.1 Analysis of the effect of including gaze-direction

The aim of the following analysis was to achieve insight into the effect of adding information from an extra modality to predict the users intent in form of muscle activations and kinematics during single and multi joint reaching movements.

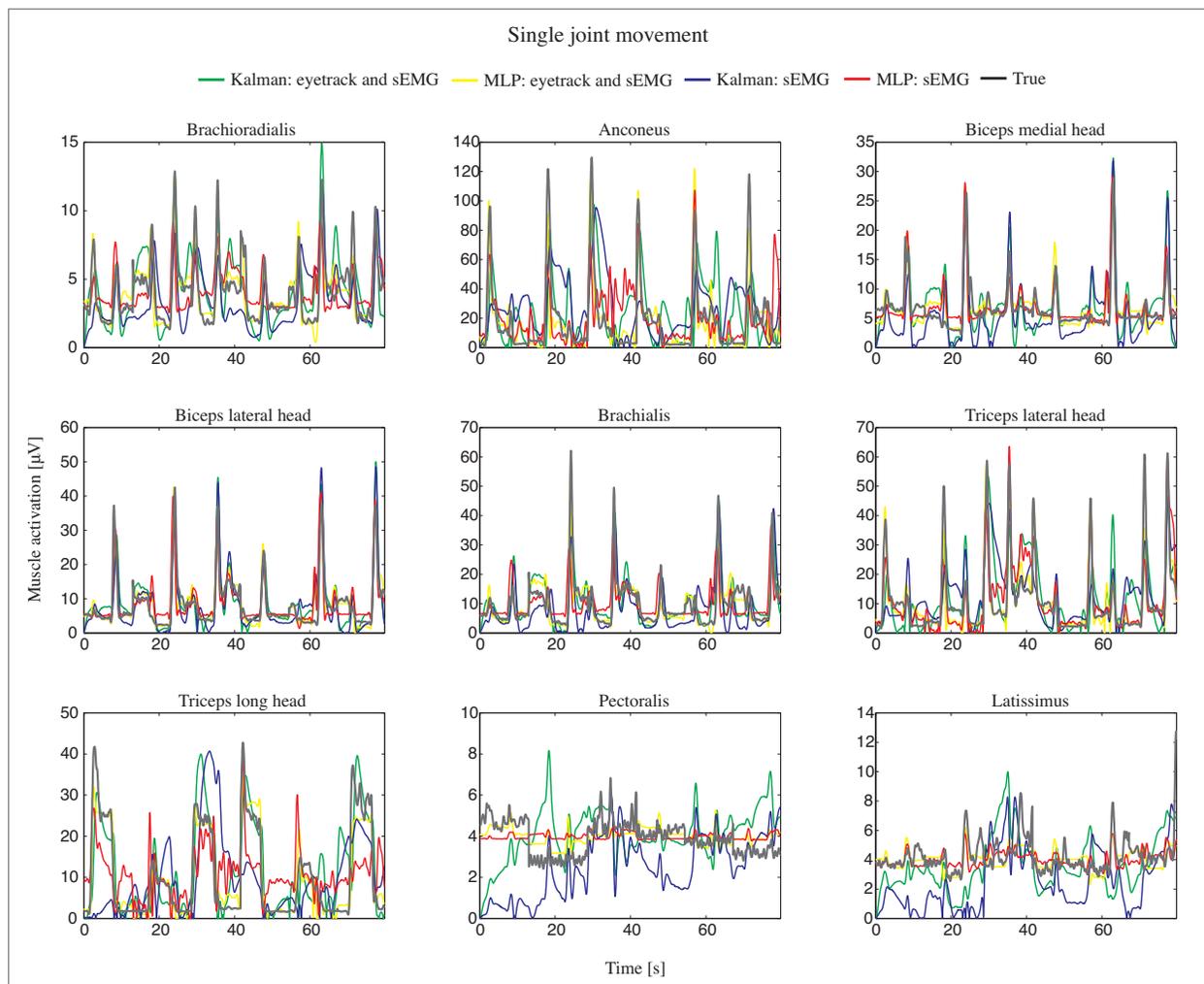
Based on recordings from [95], who recorded sEMG from 12 extensor and flexor muscles of the upper arm and concurrently the kinematics from single and multi joint movements, the muscle activity was predicted in the 9 distal muscles (Brachioradialis, Anconeus, Biceps medial head, Biceps lateral head, Brachialis, Triceps lateral head, Triceps long

head, Pectoralis, Latissimus) and the position in  $x$  and  $y$  of the endpoint of the hand, from the three deltoid muscles (Deltoid anterior, Deltoid medial, Deltoid posterior) and simulated gaze-direction. The recordings were based on the protocol outlined in section 5.1.

Since no gaze-direction was recorded during these experiments, this was simulated from the kinematic data representing the endpoint position of the marker placed most distal on the arm (corresponding to the endpoint position of the hand), with noise added.

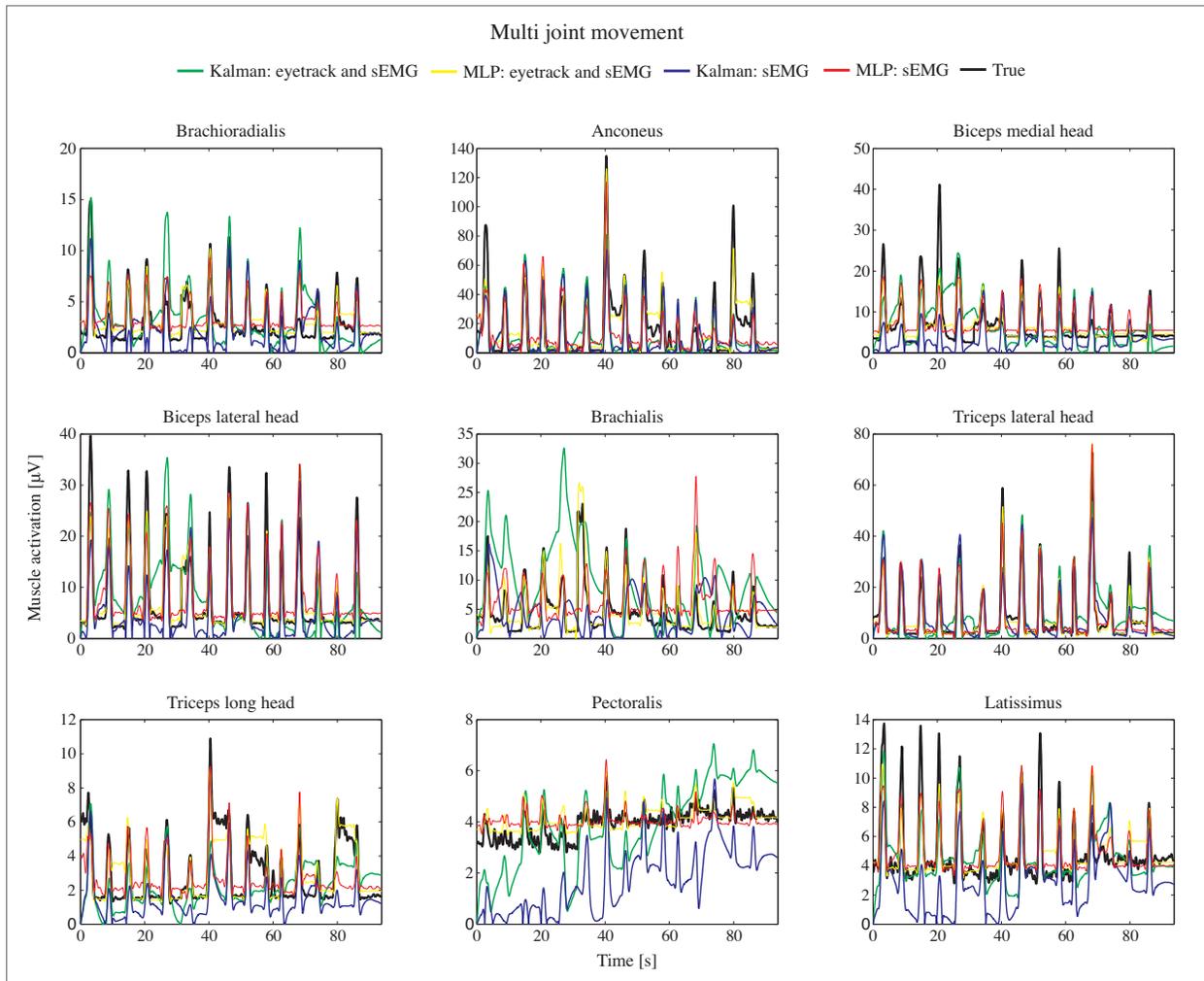
The predictions were performed utilizing a MLP NN and the Kalman filter (for comparison) and the predictions were evaluated based on the coefficient of determination,  $R^2$ . The MLP NN and the Kalman filter is introduced in chapter 4 on page 29, and the practical implementation in section 5.2 on page 43 and  $R^2$  is introduced in chapter 4 on page 29. Further details of the results in this analysis are found in chapter 6 on page 57.

Figure 2.3 and 2.4 shows the predictions of the muscle activity during single and multi joint movement based on MLP NN and the Kalman filter with (yellow and green respectively) and without (red and blue respectively) the inclusion of gaze-direction.



**Figure 2.3:** Predictions of the muscle activity during single joint movement based on MLP NN and the Kalman filter with (yellow and green respectively) and without (red and blue respectively) the inclusion of gaze-direction.

Figure 2.3 revealed well predicted muscle activity for all muscles with single exceptions, i.e. for the Kalman filter (both cases) in Pectoralis and Latissimus. Likewise the Kalman filter based on sEMG and gaze-direction showed only minor accuracy in Brachialis. Also the baseline in e.g. Triceps long head did not capture the baseline well, but was in the contrary underestimated compared to the other methods and the true signal.



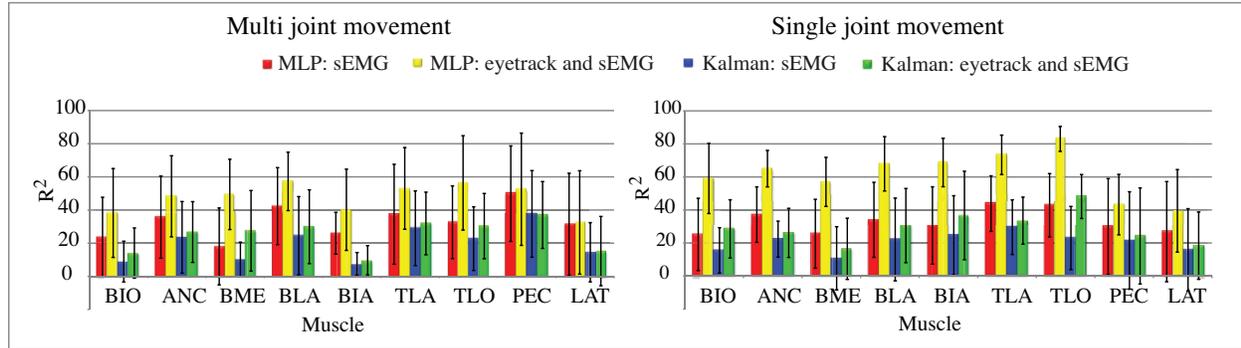
**Figure 2.4:** Predictions of the muscle activity during multi joint movement based on MLP NN and the Kalman filter with (yellow and green respectively) and without (red and blue respectively) inclusion of gaze-direction.

From 2.4, it was seen that MLP NN based on sEMG and gaze-direction gave predictions, which very well reconstructed the true muscle activity, capturing most characteristics. MLP NN based on the sEMG only showed well predicted results in most cases, though e.g. in Pectoralis the baseline was often over estimated. The configurations based on the Kalman filter predicted well the activity in Biceps lateral head, Brachialis, and Brachioradialis, whereas excessive under- and overestimations were seen in Pectoralis and Latissimus.

Figure 2.5 shows the performance of the predictions of the muscle activity during single and multi joint movement based on MLP NN and the Kalman filter with (yellow and green respectively) and without (red and blue respectively) the inclusion of gaze-direction.

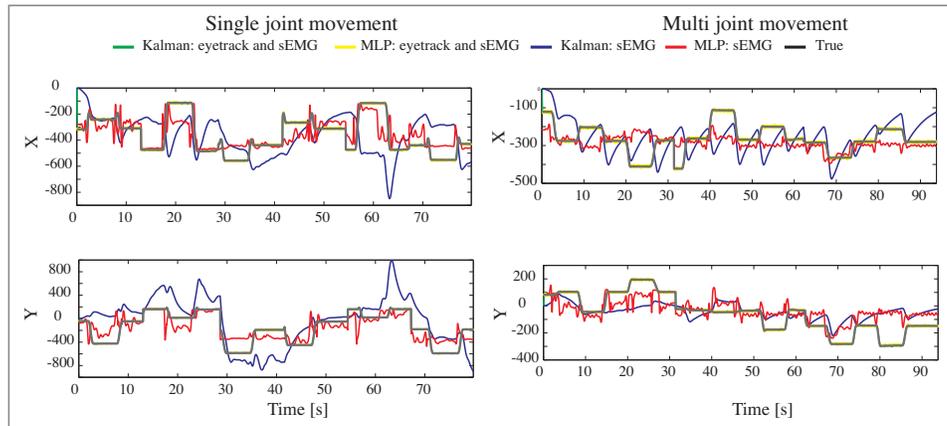
The predictions of the muscle activity revealed a clear improvement for both the single and multi joint movements by including the gaze-direction and utilizing the MLP NN for the predictions. In all cases gaze-direction improved the predictions and furthermore the MLP NN outperformed the Kalman filter in all cases.

The muscle activity predictions were improved on average across all muscles for the single and multi joint movements



**Figure 2.5:** Performance of predictions of the muscle activity during single (left) and multi joint (right) movement based on MLP NN and the Kalman filter with (yellow and green respectively) and without (red and blue respectively) the inclusion of gaze-direction.

with approx. 24% based on MLP NN, and 5% using the Kalman filter. For the single joint movement the predictions of the muscle activity were improved by approx. 29% and 9% for the MLP NN and the Kalman filter respectively. Figure 2.6 shows the predictions of the kinematics during single and multi joint movement. Predictions based on both sEMG and gaze-direction is very close to the true signal, which make them hard to identify on the figures.

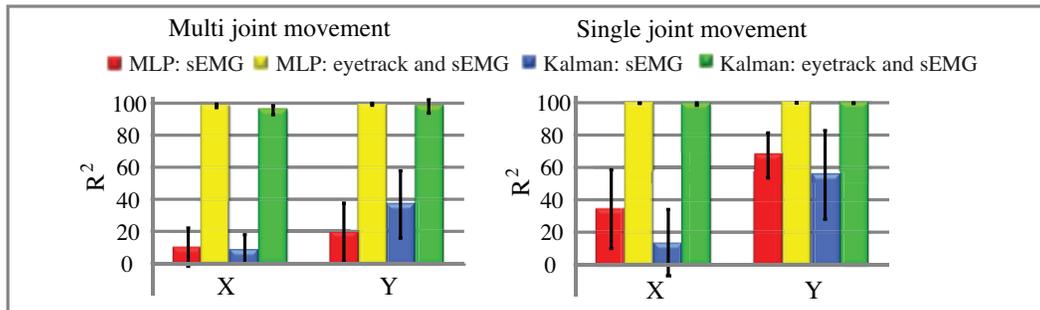


**Figure 2.6:** Predictions of the kinematics during single joint (left) and multi joint (right) movement based on MLP NN and the Kalman filter with (yellow and green respectively) and without (red and blue respectively) the inclusion of gaze-direction.

From figure 2.6 it was clear that the inclusion of the predicted gaze-direction substantially improved the predictions, seen as prediction very close to the true signal what makes them hard to identify on the figure. This was the case for both the single and multi joint movements. Without the gaze-direction included, the MLP NN outperformed the Kalman filter by capturing more of the edges in the true signal.

Though, for the single joint movement, the prediction based on this configuration still showed a lack of capturing the extremities in the true position. The Kalman filter only followed the trend over large periods, thus not capturing the peaks seen in the true signal. For the multi joint movement, none of the two configurations captured well the characteristics of the true position. In  $Y$  the predictions only hardly followed the variations in the true position. In  $X$ , the Kalman filter showed a very varying prediction substantially different from the true position.

Figure 2.7 shows the performance of the predictions of the kinematics during single and multi joint movement based on MLP NN and the Kalman filter with (yellow and green respectively) and without (red and blue respectively) the inclusion of gaze-direction.



**Figure 2.7:** Performance predictions of the kinematics during multi (right) and single joint (left) movement based on MLP NN and the Kalman filter with (yellow and green respectively) and without (red and blue respectively) the inclusion of eyetracking.

For the Kinematics the improvements were substantially higher. For the multi joint movement, the predictions were improved on average by approx. 84% and 74% using the MLP NN and Kalman filter respectively. For the single joint movement, the predictions were improved by approx. 49% and 65% for the MLP NN and the Kalman filter respectively.

The preliminary results clearly indicated that adding an extra modality in form of gaze-direction had the potential of increasing the prediction performance during reaching movements. The outcome of the kinematics predictions were as expected, since the simulated gaze-direction is almost similar to the true kinematics.

## 2.5.2 Controller

The controller of a FES or FET system provides the required muscle stimulation level (i.e. specification of temporal patterns of muscle stimulation to provide the desired movement), based on the commands from the user, i.e. the decoded intent of the user. For upper limb neuroprosthesis, the appropriate shoulder and arm muscles needs to be stimulated to enable coordinated arm movements [20]. Not only should the controller provide muscle stimulation but also regulation of these patterns to correct for unanticipated changes in either the stimulated muscles (e.g. fatigue) or the environment (e.g. external perturbations).

The complexity of the musculoskeletal systems complicates the temporal specification of the control, for the upper limb neuroprosthesis. The stimulation patterns must account for the nonlinear nature of the muscles, between muscle output and limb output, contemporary with the varying load encountered during interaction with the environment [20]. Different approaches can be undertaken to access the muscle activation needed for the stimulation and the relationship between muscle activity and various dynamics and kinematics of movement have been investigated widely in the literature.

One approach is an inverse musculoskeletal model, a mathematical description of the musculoskeletal system, with the kinematics, i.e. motions, as input to compute the required forces or muscle activations [109, 60, 51, 11].

The inverse dynamic problem involves the transformation from a desired motion to the forces or muscle activations that are needed to drive the forearm and hand during reaching movement, which is a hard computational challenge because several limb segments must be combined, and the continuous changes in the mechanical properties of the limbs and the environment must be taken into account [10].

Another approach is by predicting the muscle activation levels directly from the signals recorded from the modalities that are remained under voluntary control. Continuously prediction of the muscle activity pattern during execution of a desired movement trajectory, can serve as control signal directly, e.g. by conversion into frequency-modulated trains of constant current pulses. Previously stimulus frequency has been linearly related to the amplitude of the average muscle activity in some specified time period [139]. [41] used the activity from the proximal muscles under voluntary control (shoulder muscles and elbow flexor muscles) in C5-C6 tetraplegics to estimate the stimulation level in Triceps to produce elbow extension. The muscle activation level in the shoulder and elbow flexors were associated to a prede-

finest set of endpoint force load during training, which would also occur during validation.

Proportional control consists of commands which yields a system response with a magnitude proportional with the command provided [137]. Examples are observation of wrist joint angle, and EMG of the wrist extensor muscles [47]. [137] Static control in the contrary is based on on/off control, such as button switches [98], wrist mounted switches [125], and sip-puff switches[45].

## 2.6 Reduction of the complexity for movement control

Human movements involves activation of a large number of motor units (MUs) and the movements can span multiple muscles. Coordination of a large number of D.O.F in the musculoskeletal system is necessary to produce movement. The number of available D.O.F of the body is typically greater than that required to reach a behavioral goal and the number of muscles per one D.O.F is much greater than two. This redundancy is exploited by the nervous system to control actions in a flexible way, i.e. the same behavioral goal can be accomplished in differently depending on intention, external environment, or intrinsic constraints. [6] Thus the many D.O.F of the musculoskeletal system provide great flexibility, but at the same time it increases the complexity of the control problem.

Even simple tasks, such as moving the hand to a target location, implies an infinite number of possible paths that the hand can move along. For each of these paths the hand can follow an infinite number of trajectories, i.e. velocity profiles, and each location of the hand along the path could be achieved by multiple combinations of joint angles [173]. Each arm configuration can be achieved by many different muscle activations, due to the overlapping actions of the muscles and their ability to co-contract [173].

This redundancy is characteristic in most everyday tasks [157] and introduces the redundancy problem in control, where the number of available independently controlled variables available for a controller, is greater than the number of independent parameters describing a motor task. Of all the possible solutions, one must be chosen. The essence of motor control has been formulated to be the elimination of the redundant D.O.F.[76]

Any ordinary human activity requires the cooperation among huge number of structurally diverse elements (e.g. joints and muscles). It has been hypothesized for such complex living systems that the elements are organized into synergies defined as functional groupings of structural elements (e.g. neurons, muscles, or joints), temporarily constrained to act as a single unit [72]. The synergy hypothesis considered in this work is related to muscle synergies [79, 10, 159, 25, 27, 1].

Thus, the neural control of movement has been studied on the assumption that the CNS simplify the complexity in the control of movements by producing movements in a simplified manner. One strategy of the production of movement is believed to be spinally organized muscle groupings, in which the activation level is specified together [159]. Thus synchronous muscle synergies is a set of relative non-negative levels of muscle activation that recruits a group of muscles in a coordinated manner, i.e. coherent activations of a group of muscles[68, 26, 25, 27]. This simplifies the control by the CNS since groups of muscles can be controlled as units, which has been indicated in previous studies including leg movements in frogs [27, 26], forearm movements of monkeys [108],and humans [55].

Further it has been suggested that the CNS constructs complex muscle activation patterns for a wide range of motor behaviors based on flexible combinations of only a few muscle synergies [154, 95]. Hence the produced movements are based on a coordinated activation of muscle synergies.[154] The set of relative non-negative levels of muscle activation (the muscle synergy) is scaled by a non-negative motor command (neural command signal) and the global muscle activation level (muscle activation pattern) of each muscle is constructed by the summation of the contributions from all synergies, weighted by the motor commands, i.e. summing the muscle activation patterns generated by different synergies [154, 95].

Hence the movement tasks are executed by translating neural task-level commands into muscle activation patterns by synchronous muscle synergies[154, 95]. Thus muscle synergies are basic building blocks from which many different movements can be derived. [34]

Movement disability is one of the most invalidating injuries that can follow an SCI, which imply loss of independence and good health. Paralyzed individuals lose their abilities to interact with their surroundings and cannot complete many daily activities critical for independent living. High-level SCI is devastating causing a near total loss of independence. This independence has high economical cost for the society in form of medical cost for the paralyzed individuals who have a long life expectancy.

Damage to nerves are in the far most cases permanent and therefore rehabilitation technology for the SCI population for performing daily activities or for restoring even a modest level of independence has a great potential to generate an enormous impact on quality of life and to lower the related costs to society. Especially active control of elbow extension has shown to be very important for ensuring independence in individuals who are subject to tetraplegia.

FES is a promising technology for restoring motor function for improving the overall health of SCI individuals through enhanced use of the cardiovascular and musculoskeletal systems. FES systems have been in focus in much research, particularly for the restoration of arm reaching. This includes the multi joint system consisting of the shoulder and elbow during planar movements, which is especially relevant in a wide range of functional tasks. Regain useful arm movement is regarded as one of the most important factors for regaining functional independence.

Individuals suffering from C4 tetraplegia have only very few muscles under voluntary control limiting the type of movements they can generate and furthermore they are subject to extensive denervation of shoulder and elbow muscles important for controlling the proximal limb. These factors complicates the restoration of arm reaching in this population and consequently only few have used FES for this purpose.

Many approaches in development of FES systems have utilized pre-programmed set of muscle stimulation patterns and in some systems assistance is required to place the object in the user's hand or at a specific location in the workspace, which limits the functional use of these devices. Also the classification of different classes of movement have been the aim in many studies. More flexible control of a FES system is therefore believed to expand the repertoire of motor functions available to paralyzed individuals. Restoration of flexible arm control through FES presents challenges to both decoding of the users intent and controlling the stimulation of the respective muscles to realize that intent through movements of the paralyzed limb.

Several studies have focused on decoding the users intent through intracortical recordings of neural activity, while only a few have utilized noninvasive methods based on sEMG signals [41]. Such a noninvasive approach would besides FES also be clinical applicable for FET.

Only few have estimated the muscle stimulation level directly [3, 119, 139], in the contrary to an inverse modeling approach based predictions of Kinematics. Especially the use of sEMG as control signal in such an approach has been subject for limited investigation, i.e. that by [41] was limited to a predefined set of associations between an endpoint force and stimulation level.

These facts motivated the focus in this study to be aimed at improving the decoding of the users intent and create the basis for extending the possible muscle stimulation patterns in a FES and FET system. The preliminary results presented and the results outlined from [7] underlined the rationality in pursuing the approach of decoding the users intent from the muscle activity in the distal muscles in the upper limb or the kinematics based on the muscle activity in the proximal Deltoid muscle in combination with gaze-direction.

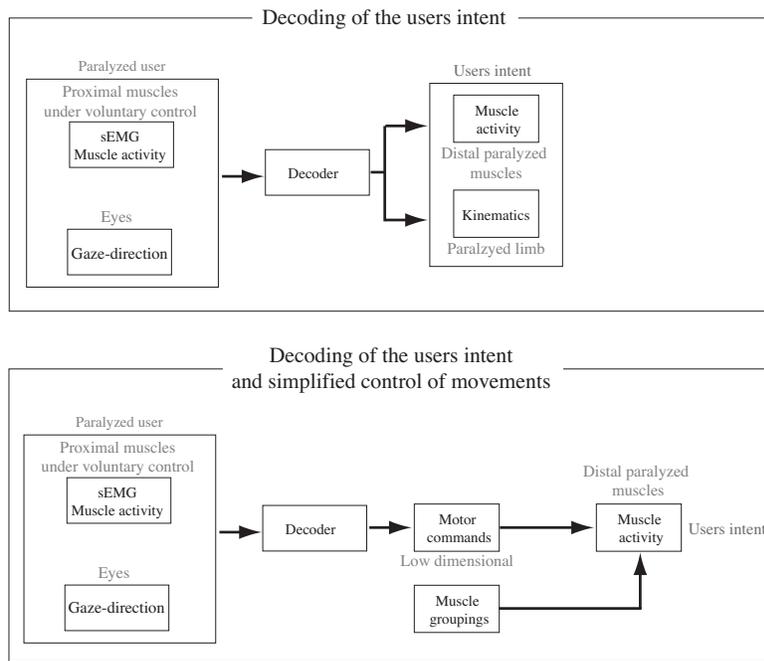
The believe in muscle synergies appeared as a sound foundation for the simplified control of movements by the CNS in which spinally organized muscle groupings can be controlled as units, and seemed as an engaging strategy for application for electrical stimulation of paralyzed limbs. This approach would allow construction of complex muscle activation patterns for a range of motor behaviors based on flexible combinations of a few muscle synergies. A direct

implication of this could be the simplification of stimulation control.

It was hypothesized that non-invasively decoding of the users intent during planar arm reaching could be obtained from a multimodal approach based on sEMG and gaze-direction.

In addition a simplified representation of the users intent was investigated to serve as control in electrical stimulation.

The project aim is illustrated in figure 3.1



**Figure 3.1:** Top: The first aim was to decode the users intent from proximal recorded sEMG combined with gaze-direction of the user recorded with eyetracking to determine whether enough information could be extracted from the selected modalities to predict this intention (muscle activity or kinematics). Bottom: Second aim was to investigate whether the decoded intent of the user in form of muscle activity could be simplified for the control of stimulation.

The overall objective of this work was to develop the foundation to a clinical applicable system using electrical stimulation both for functional and therapeutic use. In a closed loop FES, the approach taking targets individuals suffering from high tetraplegia and should enable them to perform goal directed planar movement. In FET control, the foundation is for therapeutic control for e.g. hemiplegic individuals and for rehabilitation of individuals who have suffered from stroke.

The project was a proof of concept of the suggested methods.

---

# Methods

---

**II**



Decoding of the users intent is founded on signals generated from modalities that are remained under the users voluntary control. Two relevant types of signals during a reaching task is believed to be the myoelectrical signal and gaze-direction, which will be apparent from the succeeding sections.

The myoelectrical signal provides access to the physiological processes that cause the muscles to generate force, produce movement and accomplish the countless functions, which allow us to interact with the surrounding world. It represents electrical currents generated in the muscles during contraction, depicting the neuromuscular activities [64]. Voluntary limb movement is a result of the brain generating a spike train of action potentials that is transmitted through the efferent nerve to a presynaptic membrane, causing the neurotransmitter in this membrane to release and diffuse to the postsynaptic muscle membrane. This causes a depolarization and an initiation of an action potential, which propagates across the length of the muscle's surface via a ion transfer, causing the muscle to contract. The waveform traveling along the muscle fibers is known as a motor unit action potential (MUAP). The sEMG records the composite of the voltage changes generated by these individual active MUAPs, i.e. the sum of the electrical contributions made by the active motor units as detected by electrodes placed on the skin overlying the muscle[171, 88, 33]. The gaze-direction recorded from eyetracking, is tight coupled to the hand movement during natural reaching movements [134]. Evidence has suggested that humans plan upper limb activity in three-dimensional Cartesian coordinates [9], and that this target activity is transformed into the muscle activity required to produce the desired limb dynamics [149].

The aim was to decode the users intent, which was expressed in terms of the following

1. Arm kinematics
2. The muscle activation pattern in the distal primary extensors and flexors of the arm
3. Low dimensional motor commands

The muscle activity can be used directly in the stimulation process of FES and FET systems, e.g. by converting the activity into frequency-modulated trains of constant current pulses [139]. Hence, expressing the intent of the user in terms of muscle activity, can at the same time serve as control signal for the electrical stimulation in the construction of specific movements.

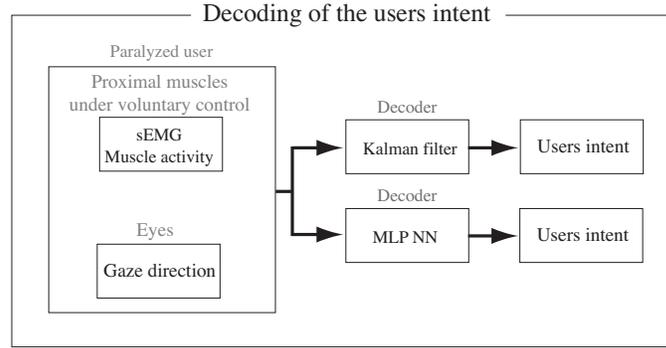
The predictions should be based on the muscle activity in the proximal flexion/extension muscles of the shoulder and from the gaze-direction of the user as depicted in figure 4.1

## 4.1 Extraction of muscle activation level from the EMG

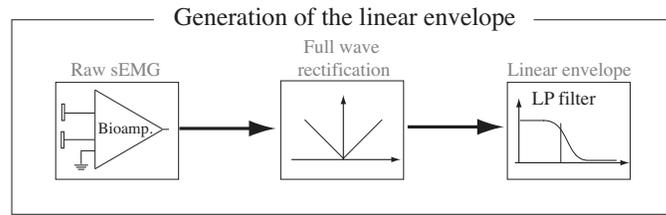
The raw EMG signal is a complex spiky signal that can be difficult to interpret. The linear envelope is a rectified and smoothed version of the EMG, which result in a slowly varying trend, representing the muscle activity.

The raw EMG signal is transformed into a linear envelope by full-wave rectification followed by a low-pass filtering. The characteristics of the low-pass filter determines the smoothness and duration of the bursts of activity [4, 50, 89]. A cut-off frequency of 1 Hz was chosen [95, 50, 15, 144, 119]. Figure 4.2 illustrates the process of extracting the linear envelope.

Two methods were investigated for the predictions, namely the Kalman filter and the Multi layer perceptron neural network (MLP) NN, which are introduced in the following together with the extraction and reconstruction method for the low dimensional control. Finally the evaluation measure used for quantification of the accuracy of the predictions is presented.



**Figure 4.1:** Decoding the users intent in terms of kinematics, muscle activity in the primary extensors and flexors, and low dimensional motor commands, based on the combination of the recorded muscle activity in the proximal flexion/extension muscles of the shoulder and gaze direction. Two decoders were compared on their prediction performances.



**Figure 4.2:** Generation of the linear envelope of the sEMG.

## 4.2 Decoding based on the Kalman filter

The Kalman filter can estimate the state of a linear dynamic system from a series of noisy measurements together with a model of the dynamics of the system. Thus although the state of a system is not directly available, it is possible to estimate the state indirectly. The Kalman filter has been used for prediction of the kinematics (e.g. hand endpoint position and velocity) [173], the hand trajectory (position, velocity, and acceleration) based on intracortical recordings representing the neural firings in the motor cortex [178, 177, 175].

The Kalman filter uses knowledge about the dynamics of the system to estimate the system state, i.e. the muscle activity or the endpoint position of the hand. The state from a previous time instance ( $\mathbf{x}(k-1)$ ) is converted into a state in a future point in time ( $\mathbf{x}(k)$ ) [167]. The generative model underlying the Kalman filter is illustrated in figure 4.3

$\mathbf{z}$  representing the observations, i.e. the eyetracking and muscle activity in the proximal muscles,  $x$  representing the muscle activity, kinematics, or the motor commands depending on the representation of the intent of the user.

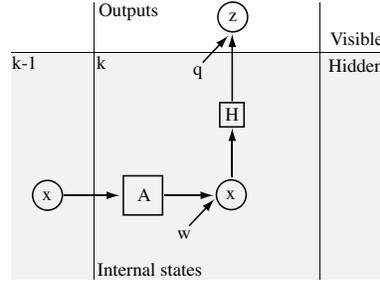
The Kalman filter is an estimator, that minimizes the estimated error covariance and estimates the state  $\mathbf{x} \in \mathfrak{R}^n$  of a discrete-time controlled process,  $n$  denoting the number of states as expressed in equation 4.1 [167]

$$\mathbf{x}_k = \mathbf{A}\mathbf{x}_{k-1} + \mathbf{w}_k \quad (4.1)$$

The observations  $\mathbf{z}_k \in \mathfrak{R}^n$  can be expressed as in equation 4.2

$$\mathbf{z}_k = \mathbf{H}\mathbf{x}_k + \mathbf{q}_k \quad (4.2)$$

$\mathbf{w}_k$  and  $\mathbf{q}_k$  are random variables representing the process and measurement noise respectively [167]. These are independent of each other, have white noise properties, and normal probability distributions as expressed in equation 4.3.



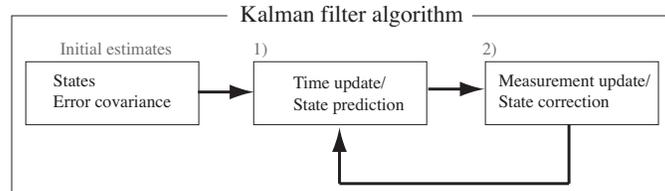
**Figure 4.3:** Underlying model of the Kalman filter. Only the output  $\mathbf{z}$ , is observable, while the underlying parameters  $\mathbf{w}$ ,  $\mathbf{q}$  represents process and measurement noise respectively, and the underlying true state are hidden.  $\mathbf{A}$  is a linear coefficient matrix that relates the system state  $\mathbf{x}$ , at a previous time instant ( $\mathbf{x}(k-1)$ ) to the state at the current time step ( $\mathbf{x}(k)$ ).  $\mathbf{H}$  relates the states to the measurements  $\mathbf{z}_k$ .

$$\begin{aligned} p(\mathbf{w}) &\sim \mathcal{N}(0, \mathbf{W}) \\ p(\mathbf{q}) &\sim \mathcal{N}(0, \mathbf{Q}) \end{aligned} \quad (4.3)$$

In practice, the process noise covariance  $\mathbf{W}$  and measurement noise covariance  $\mathbf{Q}$  might change with each time step, i.e. each measurement, but are assumed constant, and can therefore be determined prior from training [167].

The matrix  $\mathbf{A} \in \mathcal{R}^{n \times n}$  relates the state at the previous time step,  $k-1$ , to the state at the current step,  $k$ . This is a linear coefficient matrix, which perform state prediction without considering any measurements. In practice  $\mathbf{A}$  might change with each time step, but is often assumed constant [167]. The matrix  $\mathbf{H} \in \mathcal{R}^{m \times n}$  relates the states to the measurements  $\mathbf{z}_k$  with  $m$  denoting the number of input channels.  $\mathbf{H}$  might also change with each time step, but is often assumed constant [167].

The Kalman filter algorithm is depicted in figure 4.4



**Figure 4.4:** Kalman filter algorithm. Initially estimates are provided for the states and the error covariance. The state and error covariance is predicted in (1) and updated in (2). Modified from [167]

The Kalman filter estimates the states by the use of feedback control. The first main step estimates the process state at one particular time, and updates the process state with obtained feedback in form of a noisy measurement, within the second main step.

The equations for the Kalman filter can be divided into the following [167]

### 1. Time update/ state predictor equations

Responsible for predicting the current states based on previous states, and error covariance estimates to obtain an *a priori* estimate for the current time step

### 2. Measurement update equations/ state corrector equations

Responsible for the feedback and incorporation of new a measurement into the a priori estimate, with the aim of obtaining an improved *a posteriori* estimate

In the Kalman filter cycle, the time prediction projects the current state estimate ahead in time, before the measurement update corrects the projected estimate with an actual measurement at that specific time.

The a priori and a posteriori estimate errors ( $\mathbf{e}$ ) can be defined as in equation 4.4 with  $\hat{\mathbf{x}}_k^- \in \mathfrak{R}^n$  being defined as the a priori state estimate at time  $k$  giving knowledge about the process prior to  $k$  and  $\hat{\mathbf{x}}_k \in \mathfrak{R}^n$  being defined as the a posteriori state estimate at time  $k$  given measurement  $\mathbf{z}_k$ .

$$\begin{aligned} \text{a priori estimate error} &= \text{true state} - \text{a priori estimate} &\Leftrightarrow \mathbf{e}_k^- &\equiv \mathbf{x}_k - \hat{\mathbf{x}}_k^- & (4.4) \\ \text{a posterior estimate error} &= \text{true state} - \text{a posteriori estimate} &\Leftrightarrow \mathbf{e}_k &\equiv \mathbf{x}_k - \hat{\mathbf{x}}_k \end{aligned}$$

The a priori estimate error covariance is expressed in equation 4.5, since the errors are zero mean.

$$\mathbf{P}_k^- = E [\mathbf{e}_k^- \mathbf{e}_k^{-T}] \quad (4.5)$$

and the a posteriori estimate error covariance is described in equation 4.6

$$\mathbf{P}_k = E [\mathbf{e}_k \mathbf{e}_k^T] \quad (4.6)$$

The a posteriori state estimate  $\hat{\mathbf{x}}_k$  can be expressed as a linear combination of the a priori estimate  $\hat{\mathbf{x}}_k^-$ , and the weighted difference between an actual measurement  $\mathbf{z}_k$  and the measurement prediction  $\mathbf{H}\hat{\mathbf{x}}_k^-$  as expressed in equation 4.7

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + \mathbf{K} (\mathbf{z}_k - \mathbf{H}\hat{\mathbf{x}}_k^-) \quad (4.7)$$

When the measurement *residual*,  $(\mathbf{z}_k - \mathbf{H}\hat{\mathbf{x}}_k^-)$  in equation 4.7 approximates zero, the predicted and actual measurement are in close agreement.

The matrix  $\mathbf{K} \in \mathfrak{R}^{n \times m}$  in equation 4.7 is the Kalman gain, which minimizes the a posterior error covariance  $P_k$  from equation 4.6. [167]

Implementation of the Kalman filter was done in Matlab<sup>®</sup> based on [178]. For a more detailed description of the Kalman filter, please refer to [167, 59].

### 4.3 Decoding based on a multi layer perceptron network

Neural networks have been applied in a wide variety of engineering problems involving pattern recognition, pattern classification, adaptive filtering, and control [20, 162, 106, 71, 41]. The successes presented in [20, 162, 106, 71, 41], indicates the usefulness of neural networks in different components of a variety of FES control systems.

An artificial neural network (ANN) is an array of processors that is linked by connections that can be weakened or strengthened and the concept is based on inspiration from the interconnected neurons of the brain[152]. Thus the functional model of the biological neuron contains three basic components, namely the synapses of the neuron (modeled as weights), and the components representing the activity within the neuron cell, which consists of an adder (sums up all inputs modified by their weights), and an activation function. In this fashion the neuron receives an input and produces a response as output [152].

The ANN is an adaptive, and most often a nonlinear system, which estimate a function (or the relationship between input and output vectors) without requiring a mathematical description of how the output functionally depends on the input, i.e. the network learn from input/output data samples [152]. The capability of learning complex nonlinear input-output mappings is one of the characteristics that make neural networks an attractive option in many biomedical problems. The adaptive capabilities of neural networks makes them particularly attractive for rehabilitation applications in which the system is often customized for particular individuals.[20, 106]

The network is adaptive in the sense that the system parameters are changed during operation, i.e. during training. After the training phase the ANN parameters are fixed and the system can be deployed, i.e. during validation.

The input/output training data are fundamental in ANN since it conveys the necessary information to find the optimal parameters of the network. According to the success of estimating a specific function based on the series of examples

of correct responses, the connections between the processors are strengthened or weakened. Hence instead of working based on an explicit set of rules to follow, the network operates based on the weights of the links within it. The non-linear nature of the neurons (the processing elements) provides the system with a high level of flexibility to achieve the desired input/output relation. [152]

One of the most widely used types of neural networks is the MLP NN (also denoted a multilayer feed-forward neural network). An elaborated description of the neural network is found in appendix chapter F on page 109.

### Introduction to the MLP network

An MLP NN consists of a network of perceptrons (neurons), in which the data flow strictly feedforward from input to output units, and the data processing can extend over multiple layers of units, without feedback connections. The MLP NN consists of the following elements [132]

**Input layer** is a layer of neurons that receives information from external sources (in a biological neuron the input corresponds to sensory nerves), and passes this information to the network for processing

**Hidden layer** is a layer of neurons that receives information from the input layer and processes them in a hidden way, i.e. it has no direct connections to the outside world. All connections from the hidden layer are to other layers within the system

**Output Layer** is a layer of neurons that receives processed information and sends output signals out of the neural network (corresponding to motor nerves in a biological neuron)

**Bias** is an offset or threshold value in the neurons. The function of the bias is to provide a threshold for the activation of neurons. The bias input is connected to each of the hidden and output neurons in a network

A typical MLP NN consists of a set of source nodes forming the input layer, one or more hidden layers of computation nodes, and an output layer of nodes. A single layer neural network has severe restrictions limiting the type of tasks such a network can solve. A two layered feed-forward neural network with biases, a non-linear sigmoid layer, and a linear output layer is capable of approximating any function with a finite number of discontinuities to arbitrary precision [36, 48, 24].

If linear output neurons are used, the network outputs can take on any value instead of being limited to a small range as is the case with the sigmoid functions.[91]

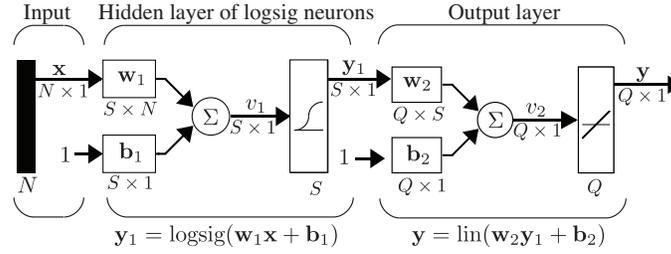
Each perceptron computes a single output from multiple real-valued inputs by forming a linear combination according to its input weights, the synaptic efficacy in a biological neuron, the bias, forming the post-synaptic potential of a neuron, and transforming the output through the nonlinear activation function. The strength of the connection between an input and a perceptron is determined by the value of the weight, i.e. negative weight values reflect inhibitory connections, while positive values designate excitatory connections.

The actual activity within the neuron cell is determined by linear combining the input and the weights together with the bias. Lastly the activation function ( $\phi$ ) maps the output into the range 0 to 1 in the case of the Sigmoid activation function, such as the simple logistic Sigmoid function. This function allows MLP networks to model both mildly and strongly nonlinear mappings, since the function is linear near the origin and saturating rather quickly away from the origin. [152] This transfer function is often used in back-propagation networks, e.g. [69, 63, 70]. The capability of MLP NN stems from the non-linearities used within the nodes.

A two layer network with one hidden layer can be represented as shown in figure 4.5

$\mathbf{x}$  represents the input units ( $N \times 1$ ),  $\mathbf{W}_1$  ( $S \times N$ ) the weights between each input unit and each of the units in the hidden layer ( $S \times 1$ ),  $\mathbf{y}_1$  ( $S \times 1$ ) is the output vector of the hidden layer,  $\mathbf{W}_2$  ( $Q \times S$ ) being the weights between the hidden layer and the output layer ( $Q \times 1$ ).  $\mathbf{b}_1$  and  $\mathbf{b}_2$  are the biases for each of the units in the hidden and the output layer respectively. The input signal propagates through the network layer-by-layer and all neurons from one layer are connected to the neurons in the next layer.

The network shown is a full-connected (output from each input and hidden neuron is distributed to all of the neurons



**Figure 4.5:** Two layer MLP neural network with one hidden layer consisting of  $S$  units with the logistic Sigmoid as transfer function, and one output layer consisting of  $Q$  units with linear transfer functions.  $\mathbf{x}$  ( $N \times 1$ ) representing the inputs,  $\mathbf{W}_1$  ( $S \times N$ ) being the weights between each input and each of the units in the hidden layer ( $S \times 1$ ),  $\mathbf{y}_1$  ( $S \times 1$ ) being the output vector of the hidden layer,  $\mathbf{W}_2$  ( $Q \times S$ ) being the weights between the hidden layer and the output layer ( $Q \times 1$ ).  $\mathbf{b}_1$  and  $\mathbf{b}_2$  are the biases for each of the units in the hidden and the output layer respectively.

in the following layer), two layer, feed-forward (the values only move from input to hidden to output layers, no values are fed back to earlier layers), perceptron neural network. The number of input neurons corresponds to the number of input variables (e.g. the number muscle activations), and the number of output neurons is the same as the number of desired output variables (e.g. muscle activation, low dimensional motor commands, kinematics).

The process of the neural network can be expressed in equation 4.8

$$y_m = \underbrace{\sum_{q=1}^Q w_{mq}}_{\text{Output layer}} \cdot \underbrace{\phi \left( \sum_{n=1}^N w_{kn} \cdot x_n \right)}_{\text{1st hidden layer}} \quad (4.8)$$

$x_n$  being the  $n^{\text{th}}$  input,  $w_{kn}$  being the weight from input  $n$  to node  $k$ ,  $w_{mq}$  being the weight from node  $q$  to output  $m$ , and  $y_m$  being output  $m$ .

The network is an implementation of a composite function from input to output space, which is called the network function [132].

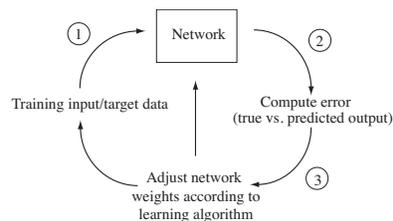
### Learning of the neural network

A neural network has to be trained, such that the network produces the desired output given a set of inputs. I.e. the strength of the connection weights have to be adjusted adequate to achieve the desired behavior, which allow the characteristics of the input training data to be inferred. This requires a learning algorithm, to which a set of desired input-output mappings are presented, and in response executes correction steps iteratively until the network learns to produce the desired response. As seen in figure 4.6, the learning algorithm is a closed-loop presentation of input-output mappings (training data) and of corrections to the network parameters to minimize the prediction error.[132] During the training process, the outputs can likewise be specified as targets.

The error of a particular configuration of a network, can be determined by applying the network on training data set and compare the predictions with the true outputs, thus forming the error function. The error surface is a  $N + 1$  dimensional surface, each weight and threshold being a dimension in the space. Hence the goal of the training process is to find the optimal combination of weights, i.e. the lowest point in the many dimensional error surface, so that the network function approximates a given function described implicitly through some training samples [132, 152].

From an initially configuration of weights and thresholds, the objective of the training algorithm is to seek for the global minimum. The gradient of the error surface is typically calculated in each point and the search continues along the steepest descent.

The back propagation algorithm is a popular learning method, which looks for the minimum of the error function in weight/bias space using the method of gradient descent. This method requires the computation of the gradient vector of the error surface at each iteration step.



**Figure 4.6:** Closed loop learning algorithm. The learning algorithm adapts the network parameters according to previous experience until a solution is found. For each input, the network produces an output (1) and the error between the predicted and desired output is calculated (2). The weights are adjusted to reduce the error according to some learning algorithm. (1) - (3) are repeated for every sample in the case of incremental training until the errors are minimized. Modified from [132]

The sigmoid function always have a positive derivative, so the slope of the error function provides a greater or lesser descent direction, which can be followed. The gradient vector points along the line of steepest descent from the current point, thus by moving along by some step size the vector the error will decrease.

The learning algorithm progresses iteratively through a number of epochs. For each epoch the training data set is applied to the network, from which the prediction error can be computed and the weights can be adjusted according to the error surface gradient and process can be repeated. The process continues until some criteria have been reach, e.g. a certain number of epochs have evolved, the error has reached an appropriate level or when it stops improving.

In the back-propagation learning algorithm the forward pass predicts the outputs from the given inputs evaluated and in the backward pass, the partial derivatives of the cost function with respect to the different weight parameters are propagated back through the network where the weights then can be adapted. Using the back propagation algorithm, the output errors are propagated back to the previous layer using the output element weights [132]. The following summarizes the steps and equations involved in the backpropagation algorithm [132].

1. Set initial weights
2. Feed forward computation
3. Backpropagation to output layer
4. Backpropagation to hidden layer
5. Weight update

The algorithm is stopped according to a defined stop criterion. A feed forward computation stores both the evaluation of the primitive function and the derivatives. In the back propagation the constant 1 is fed into the output layer and the network is run backwards. The incoming information to a node is added and the result is multiplied by the derivative stored in that unit. The result is transmitted to the left in the unit and the result collected at the input unit is the derivative of the network function with respect to the input to the network.

Please refer to appendix section F.3 on page 112 for a more detailed description of the back propagation algorithm.

The backpropagation method suffers from various convergence problems [129]. These problems can be solved by using second order derivatives, which is applied in the Levenberg-Marquardt (LM) algorithm, one of the most widely used optimization algorithms and is suggested for moderate-sized feedforward neural networks (up to several hundred weights) [43, 84, 28]. In the following, the LM algorithm [80, 133] is introduced to an extent which explains the parameters, which are configured in practice. For a detailed description of the LM algorithm please refer to [62, 124], since this is beyond the scope of this report.

### The Levenberg-Marquardt algorithm

The LM algorithm relies on both the first and second order derivative of the error in the search for the optimal weights. The LM algorithm provides a faster solution due to the incorporation of the second derivative of the error information

and automatic incorporates adjustments of the learning parameters. [80]

The idea of the LM algorithm is to minimize the squared distance  $\epsilon^T \epsilon$ ,  $\epsilon = \mathbf{x} - \hat{\mathbf{x}}$ , i.e.  $\epsilon$  is the vector of network errors. Letting  $f$  being an assumed functional relation which maps a parameter vector  $\mathbf{p} \in \mathbf{R}^m$  to an estimated measurement vector  $\hat{\mathbf{x}} = f(\mathbf{p})$ ,  $\hat{\mathbf{x}} \in \mathbf{R}^n$ . The basis of the LM algorithm is a linear approximation to  $f$  in the neighborhood of  $\mathbf{p}$ . [133, 80]

For a small  $\|\Delta_p\|$ , a Taylor series expansion lead to the approximation in equation 4.9

$$f(\mathbf{p} + \Delta_p) \approx f(\mathbf{p}) + \mathbf{J}\Delta_p \quad (4.9)$$

$\mathbf{J}$  being the Jacobian matrix  $\frac{\partial f(\mathbf{p})}{\partial \mathbf{p}}$ , containing the first order derivatives of the network errors with respect to the weights and biases. At each step in the iterative algorithm is to find the  $\Delta_p$  that minimizes the quantity given in equation 4.10

$$\|\mathbf{x} - f(\mathbf{p} + \Delta_p)\| \approx \|\mathbf{x} - f(\mathbf{p}) - \mathbf{J}\Delta_p\| = \|\epsilon - \mathbf{J}\Delta_p\| \quad (4.10)$$

The minimum is attained when  $\mathbf{J}\Delta_p - \epsilon$  is orthogonal to the column space of  $\mathbf{J}$ , yielding  $\mathbf{J}^T(\mathbf{J}\Delta_p - \epsilon) = 0$ ,  $\Delta_p$  being the solution to the normal equations in 4.11

$$\mathbf{J}^T \mathbf{J} \Delta_p = \mathbf{J}^T \epsilon \quad (4.11)$$

The matrix  $\mathbf{J}^T \mathbf{J}$  in the left side in equation 4.11 is the approximate Hessian, i.e. an approximation to the matrix of second order derivatives. The LM algorithm solves a slight variation of equation 4.11 as seen in equation 4.12, called the augmented normal equations

$$\mathbf{N} \Delta_p = \mathbf{J}^T \epsilon \quad (4.12)$$

The off-diagonal elements of  $\mathbf{N}$  are identical to the corresponding elements of  $\mathbf{J}^T \mathbf{J}$  and the diagonal elements are given in equation 4.13

$$\mathbf{N}_{ii} = \mu + [\mathbf{J}^T \mathbf{J}]_{ii}, \mu > 0 \quad (4.13)$$

The diagonal elements of  $\mathbf{J}^T \mathbf{J}$  is altered by damping, where  $\mu$  is the damping term. If the updated parameter vector  $\mathbf{p} + \Delta_p$ ,  $\Delta_p$  computed from equation 4.12, leads to an reduction in the error  $\epsilon$ , the update is accepted and the process repeats with a decreased damping term. Otherwise equation 4.12 is solved again, and the process iterates until a value of  $\Delta_p$  is found (corresponding to one iteration of the LM algorithm). The damping term is adjusted at each iteration to assure a reduction in the error  $\epsilon$ , thus the damping is raised if a step fails to reduce  $\epsilon$  and otherwise the damping is reduced. [73]

The steps of the LM algorithm is the following [150]

1. Inputs are presented to the network and the corresponding outputs and errors are computed. The mean square error over all inputs is computed
2. The Jacobian matrix  $\mathbf{J}$  with respect to  $\mathbf{p}$  is computed,  $\mathbf{p}$  representing the weights and biases of the network
3. The Levenberg-Marquardt weight update equation 4.12 is solved to obtain  $\Delta_p$
4. The error is recomputed using  $\mathbf{p} + \Delta_p$ . In case the new error is smaller than the one computed in step 1, the training parameter  $\mu$  is reduced by  $\mu^-$ , and  $\mathbf{p} = \mathbf{p} + \Delta_p$  is computed and the algorithm returns back to step 1. If the error is not reduced, then  $\mu$  is increased by  $\mu^+$  and returns to step 3
5. The algorithm is terminated when at least one of the following conditions is met
  - Magnitude of gradient ( $\epsilon \epsilon^T$ , i.e.  $\mathbf{J}^T \epsilon$ ) drops below some threshold  $\epsilon_1$
  - The relative change in the magnitude of  $\delta_p$  drops below a threshold  $\epsilon_2$
  - The error  $\epsilon^T \epsilon$  drops below a threshold  $\epsilon_3$
  - A maximum number of iterations  $k_{max}$  is completed

$\mu^+$  and  $\mu^-$  are predefined values, typically initiated to 10 and 0.1 respectively.[150]

## 4.4 Extraction of muscle synergies and motor commands

The muscle activation pattern can be expressed as in equation 4.14 as the relationship between muscle synergies and motor commands [95, 1]

$$X = S \cdot P(k) \quad (4.14)$$

$S$  and  $P$  being

$$S = \begin{matrix} \overbrace{\begin{bmatrix} s_{11} & s_{12} & \dots & s_{1N} \\ \vdots & \vdots & \vdots & \vdots \\ s_{M1} & s_{M2} & \dots & s_{MN} \end{bmatrix}}^{N \text{ synergies}} \\ \left. \vphantom{\begin{bmatrix} s_{11} & s_{12} & \dots & s_{1N} \\ \vdots & \vdots & \vdots & \vdots \\ s_{M1} & s_{M2} & \dots & s_{MN} \end{bmatrix}} \right\} M \text{ muscles} \end{matrix}$$

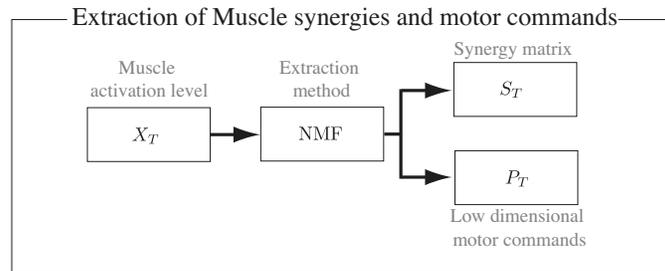
$$P = \begin{matrix} \overbrace{\begin{bmatrix} p_1(1) & p_1(2) & \dots & p_1(K) \\ \vdots & \vdots & \vdots & \vdots \\ p_N(1) & p_N(2) & \dots & p_N(K) \end{bmatrix}}^{K \text{ samples}} \\ \left. \vphantom{\begin{bmatrix} p_1(1) & p_1(2) & \dots & p_1(K) \\ \vdots & \vdots & \vdots & \vdots \\ p_N(1) & p_N(2) & \dots & p_N(K) \end{bmatrix}} \right\} N \text{ commands} \end{matrix}$$

for  $N < M$ .  $P(k) = [p_1(k), p_2(k), \dots, p_N(k)]$  being a vector of motor commands,  $p_n(k)$  being the  $n$ 'th motor command at time  $k$ , i.e.  $P(N \times K)$ . The activity levels of the  $M$  involved muscles are described by the muscle activation vector  $X(k) = [x_1(k), x_2(k), \dots, x_M(k)]$ ,  $x_m(k)$  being the level of activation for the  $m$ 'th muscle.  $S (M \times N)$  is the muscle synergies (each column representing a synergy) representing the gain by which the  $n$ 'th motor command is transferred to the  $m$ 'th muscle activation signal. Thus, the muscle activity  $X$  can be represented by a linear combination of the  $p$  motor commands.

Non-negative Matrix Factorization (NMF) method, is among the most widely used in the literature, e.g. [95, 1], to estimate  $S$  and  $P$ , as seen in equation 4.14 knowing only  $X(k)$  and constraining  $S$  and  $P(k)$  to be non-negative.

A positive driving signal for generating the activity is in correspondence with a positive muscle activity.

The procedure for extraction and reconstruction is illustrated in figure 4.7.



**Figure 4.7:** Extraction of synergies ( $S_T$ ) and motor commands ( $P_T$ ), extracted from training data ( $X_T$ ), denoted by subscript  $T$  using the NMF method.

### 4.4.1 Extraction of synergies and motor commands

Based on [78] the NMF implies finding a set of non-negative matrix factors  $S (M \times N)$  and  $P (N \times K)$ , such that  $X = SP$ , as expressed in equation 4.15

$$\text{NMF}(X) \rightarrow SP \quad (4.15)$$

$X$  ( $M \times K$ ) being a non-negative matrix.

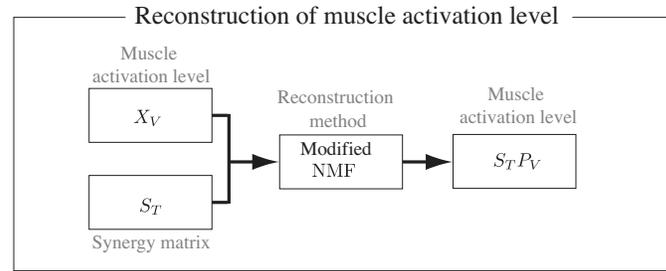
Equation 4.16 expresses the NMF algorithm based on the multiplicative update rules [78] ensuring the Euclidean distance between the true matrix  $X$  and  $SP$ ,  $\|X - SP\|$  (cost function used for quantification of the quality of the approximation) is non-increasing and invariant if  $S$  and  $P$  are at a stationary point of the distance.

$$P_{nk} \leftarrow P_{nk} \frac{(S^T X)_{nk}}{(S^T SP)_{nk}} \quad S_{mn} \leftarrow S_{mn} \frac{(XP^T)_{mn}}{(SPP^T)_{mn}} \quad (4.16)$$

Thus, the synergy matrix  $S$  and motor commands  $P(k)$  are updated iteratively during a number of iterations, from which the muscle activation to some degree can be reconstructed depending on the dimensionality of the factorization. For further details of the NMF algorithm, please refer to [78].

#### 4.4.2 Reconstruction of the muscle activation from synergies and motor commands

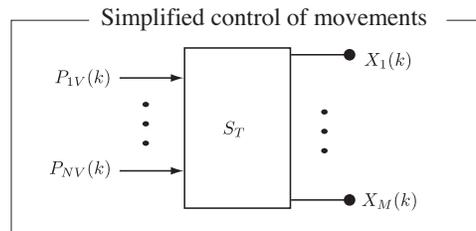
[95] showed that the muscle activation can be reconstructed from  $S_T$  and  $X_V$  using a modified version NMF by fixing the set of synergies and hence only updating the motor commands in the reconstruction of the muscle activation as depicted in figure 4.8



**Figure 4.8:** Reconstruction of muscle activity from a fixed set of muscle synergies ( $S_T$ ) extracted from the training and the muscle activation ( $P_V$ ) from the validation data using the modified NMF.

#### 4.4.3 A simplified approach for stimulation control

In the context of a FES/FET system the reduction of the complexity in the control of movement implies that the number of control signals from the controller, can be reduced as illustrated in figure 4.9

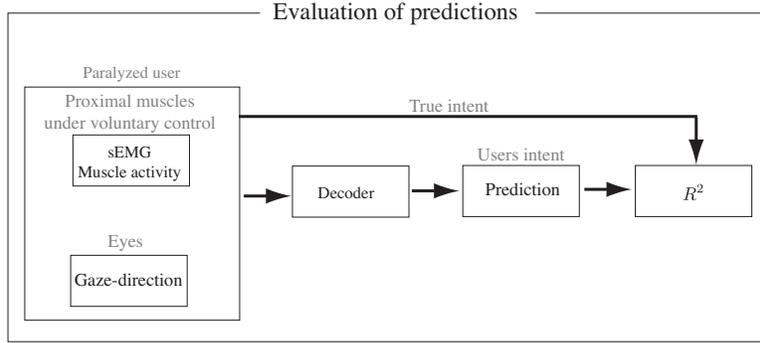


**Figure 4.9:** Simplified approach for stimulation control.  $M$  muscles can be controlled by the synergy matrix  $S_T$  extracted once from training data and the  $N < M$  decoded motor commands  $P_V(k)$  at each time instance  $k$  from online recordings.

In practice this can be achieved by estimating the muscle synergy matrix once from training data, and decode the motor commands at each time instance online.

## 4.5 Evaluation of predictions

The prediction of the kinematics and muscle activity was evaluated by the coefficient of determination ( $R^2$ ) as illustrated in figure 4.10



**Figure 4.10:** Evaluation of predictions by  $R^2$ .

$R^2$  is presented in equation 5.2.4, which expresses the percent explained variability in the true value explained by the predicted value [104].  $R^2$  is in linear regression equal to the square of the correlation coefficient between two variables [104].

$$R^2 = r^2 = \left( \frac{cov(x,y)}{\sigma_x \sigma_y} \right)^2 = \frac{\sum_{n=1}^N (x_n - \mu_x)(y_n - \mu_y)}{N - 1} \quad (4.17)$$

$x_n$  and  $y_n$  being the observed values,  $\mu$  their samples mean, and  $\sigma$  being the standard deviations (see equation 4.18) of the two variables.

$$\sigma = \sqrt{\frac{\sum_{n=1}^N (x_n - \mu_x)^2}{N - 1}} \quad (4.18)$$

$R^2$  has been used by several other authors [119, 95, 170, 26, 140, 169] for evaluating the performance of the reconstruction and prediction of signals.

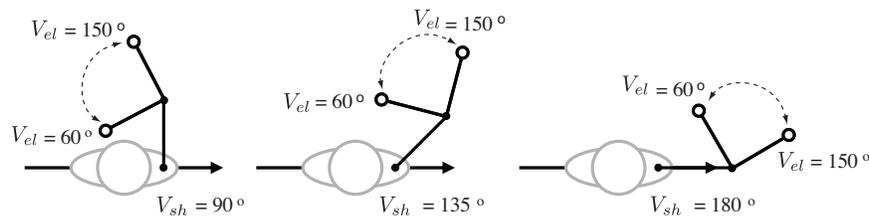


## 5.1 Preliminary data

This section contains an outline of the protocol described in detail in [95]. The aim of the data collected based on this protocol was to achieve some preliminary insight of the effect of including multiple modalities in the prediction of the users intent in form of arm kinematics or muscle activity.

Data was collected from 7 subjects during multi joint and single joint (elbow joint) movements in the horizontal plane with the subjects sitting in front of a table with the dominant right arm supported by a custom designed mechanical apparatus allowing flexion and extension of shoulder and elbow. Audible cues were provided to indicate the start and end of a movement to ensure correct speed of the movements. Each movement was of 1 s duration and between each movement, there was a pause of 5 s.

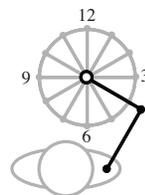
The starting position for the single joint task was with the elbow flexed, i.e.  $V_{el} = 60^\circ$  as illustrated on figure 5.1, and the shoulder angle at  $90^\circ$  (EL1),  $135^\circ$  (EL2) or  $180^\circ$  (EL3) depending on the variation.



**Figure 5.1:** Elbow movements (extension of the elbow from  $60^\circ$  to  $150^\circ$ ) at three different fixed shoulder angles.  $V_{sh}$  being the shoulder angle, and  $V_{el}$  being the elbow angle. Modified from [95]

The subject was asked to extend the elbow to the angle  $150^\circ$ . After a pause the subject was asked to do the reverse movement, returning to the starting position. This was repeated additional 4 times, so that a total of 5 extensions and 5 flexions were obtained in 60 seconds. Two repetitions of each variation were made.

The multi joint reaching task is shown in figure 5.2 where the subject was asked to reach targets in 12 directions, using both the shoulder and elbow. The start position of the elbow was  $90^\circ$  and start position the shoulder was varied in three variations from  $V_{sh} = 100^\circ$ ,  $110^\circ$ , and  $120^\circ$ .

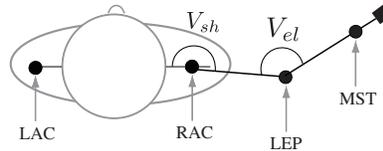


**Figure 5.2:** Multi joint movement. The subject was asked to reach targets in 12 directions marked on a target board, using both the shoulder and elbow. The start position of the elbow was  $90^\circ$  and start position the shoulder was varied in three variations from  $100^\circ$ ,  $110^\circ$ , and  $120^\circ$ . Modified from [95]

Surface EMG signals were recorded synchronously in a bipolar configuration from 12 muscles of the right arm and upper trunk (Brachioradialis, Anconeus, Biceps medial head, Biceps lateral head, Brachialis, Triceps lateral head,

Triceps long head, Deltoid medial, Pectoralis, Deltoid anterior, Deltoid posterior, Latissimus).

A motion tracking with 8 infrared digital video cameras system recorded the position of reflective markers placed at four superficial locations of the shoulder and arm of the subject, specifically left acromion (LAC), right acromion (RAC), lateral epicondyle of humerus (LEP) and posterior midpoint between styloid processes of radius and ulna (MST) as depicted in figure 5.3.



**Figure 5.3:** Position of reflective markers at left acromion (LAC), right acromion (RAC), lateral epicondyle of humerus (LEP) and posterior midpoint between styloid processes of radius and ulna (MST). Modified from [95]

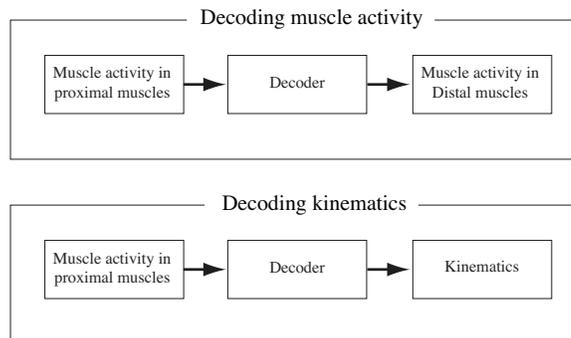
The EMG envelope was obtained by full wave rectification and low pass filtering and the 3D positions of the reflective markers were projected to the horizontal plane.

The analysis based on this preliminary data is described in the following.

### Post processing

The position of the endpoint of the hand and the muscle activity in the 9 distal muscles (Brachioradialis, Anconeus, Biceps medial head, Biceps lateral head, Brachialis, Triceps lateral head, Triceps long head, Pectoralis) were predicted based on the muscle activity in the three parts of the proximal deltoid muscle as outlined in figure 5.4.

The predictions were based on two different decoders, namely the Kalman filter and a MLP NN as introduced in section 4.2 on page 30 and 4.3 on page 32 respectively.



**Figure 5.4:** Outline of the approach for predicting the muscle activity and kinematics based on either Kalman filtering or MLP NN as decoding method.

Eyetracking was simulated as the recorded position of marker 4 (corresponding to the endpoint of the hand) with noise added (zero-mean and SD of 4).

The evaluation of the predictions was based on the  $R^2$  introduced in section 5.2.4 on page 53. All predictions were based on 80% training data and 20% validation data.

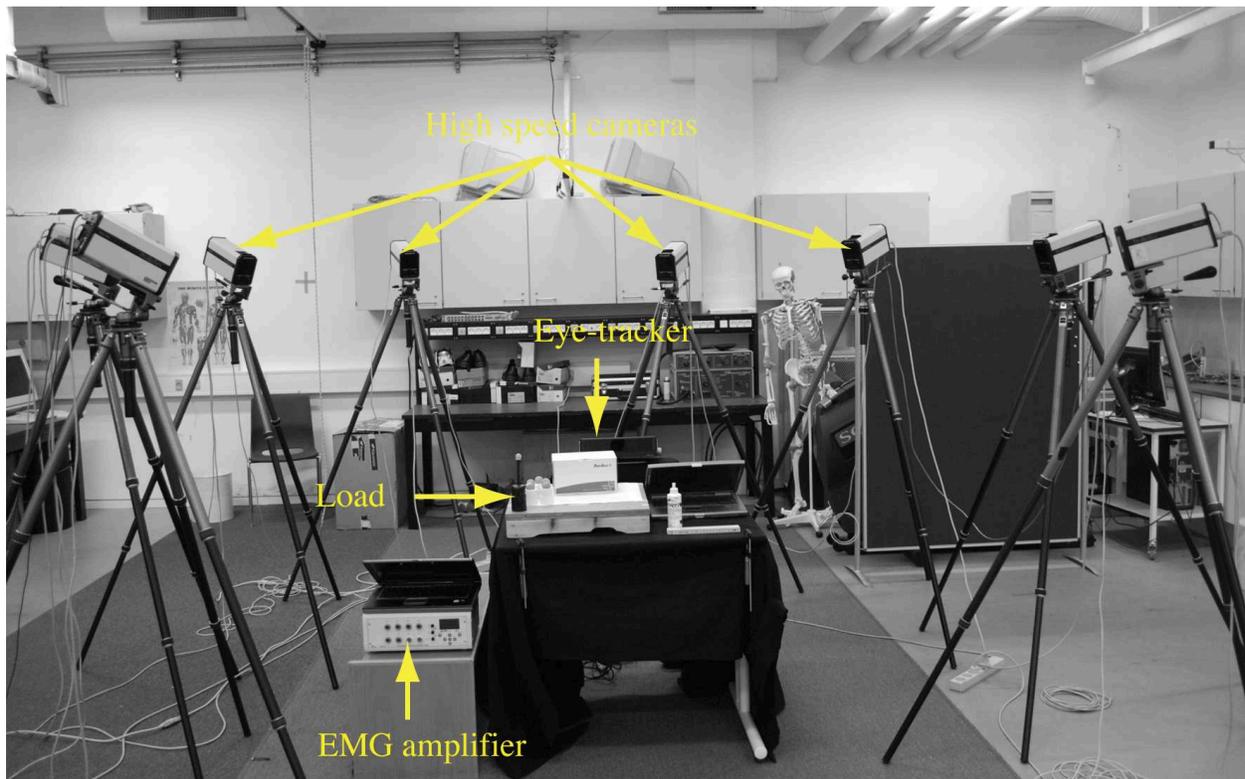
## 5.2 Conducted experiments

The aim of conducting these experiments was to collect data to investigate whether the muscle activity in primary extensors and flexors of the arm, as described in appendix chapter B.1 on page 93, in combination with gaze-direction contained sufficient information for prediction of the users intent. The protocol was modified from [95].

### 5.2.1 Experimental setup

The experiment consisted of planar multijoint arm movement with the dominant arm of 1 subject (age 26, weight 78 kg, height 1.79 m). The subject provided written informed consent before participation and the procedures were approved by the local ethic committee.

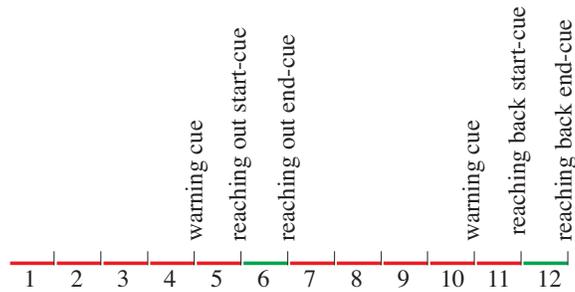
During movement in the horizontal plane the subject sat in front of a table with the lower arm supported against a horizontal surface and the subject was instructed to relax the unrestrained hand during recordings. The experimental setup is illustrated in figure 5.5. Pictures of the entire setup are located in appendix chapter C on page 95.



**Figure 5.5:** Experimental setup illustrated with the motion capture system, eyetracking, EMG amplifier, and hand load.

Audible cues were given to indicate the start and end of a movement as illustrated on figure 5.6 with a warning cue before each 'start movement' signal to indicated start of movement, and an 'end movement' signal to indicate when target should be reached. Thus each movement had a duration of 1 s and the interval between each movement was 5 s. During the reaching movements, the subject moved a load (200 g, 19 cm high, radius 2 cm) in the dominant hand to improve the signal to noise ratio of the EMG and likewise, the load served as a focus point for the gaze during the movement, ensuring that the head was in an angle making the eyes detectable for the eyetracker. Previously it has been shown that synergies are invariant for speed and load in terms of muscle activation [37].

Before the reaching task, the subject was asked to train the reaching movement, to learn the position of the targets and to maintain the correct speed of the movements. The position of the targets corresponding to that of a clock had to be



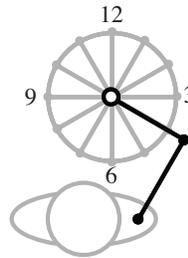
**Figure 5.6:** Audible movement cues indicating when the subject should start and stop the movement, and hence the speed of the movement. One warning cue was provided before each 'start-movement' cue.

memorized, since the subject was asked to keep the gaze at the focus point on the moving load. This dissociated from the natural focus point during movement, i.e. the endpoint target, due to a limitation of the specific eyetracker, which was not attached to the head of the subject.

Between each trial (3 in total), consisting of one repetition of reaching to each of the 12 randomized targets, the subject rested for 2 minutes. The subject did not report any fatigue during the experiment.

### 5.2.2 Task description

The subject was asked to perform planar goal directed movements of 1 second duration. The goal directed movements are illustrated on figure 5.7



**Figure 5.7:** Goal directed movements to 12 targets marked with a target board positioned with respect to the eyetracker, such that target 6 still allowed the eyetracker to track the gaze.

The reaching movements were performed at a target board with 12 evenly spaced targets placed on a circle. In the starting position the target board marking the 12 targets was positioned in front of the eyetracking, such that target 6 still allowed the eyetracker to track the gaze and the hand was placed in the midpoint of the target board.

Targets were reached in randomized order (each target occurred 1 time in each trial) and every other movement was reaching back to the midpoint. This yielded a total of 36 reaching-out and 36 reaching-back movements during a period of 2 seconds (1 in each direction). During the tasks, the subject was instructed to carefully follow the reaching focus point on the load hand with the eyes.

### 5.2.3 Data recording

Three types of signals were recorded during the experiment

1. Surface EMG
2. Motion capture
3. Gaze-direction

#### Surface EMG

Surface EMG signals were recorded synchronously in a bipolar configuration from 12 muscles of the right arm and upper trunk. Self-adhesive gel-filled Ag/AgCl electrodes (Ambu®Neuroline 720 01-K/12, Ambu A/S, Ballerup, Denmark) placed with an inter-electrode distance of 22 mm (centre to centre).

Signals were recorded with a sample frequency of 2048 Hz and amplified with a gain of 2000 (EMG-USB, LISiN - OT Bioelettronica, Rivarolo, Torino, Italy), band pass filtered (8<sup>th</sup> order Bessel filter, bandwidth 10-750 Hz) and A/D-converted with a 12-bit resolution (Acquisition v. 1.70 for EMG128 USB2, Sirio Automazione S.R.L, Rivoli, Italy).

Prior to electrode placement hair was removed and the skin surface was prepared by gentle abrasion (Every conductive abrasive paste, Meditec, Parma, Italy). The skin was then cleaned with water and finally dried with paper.

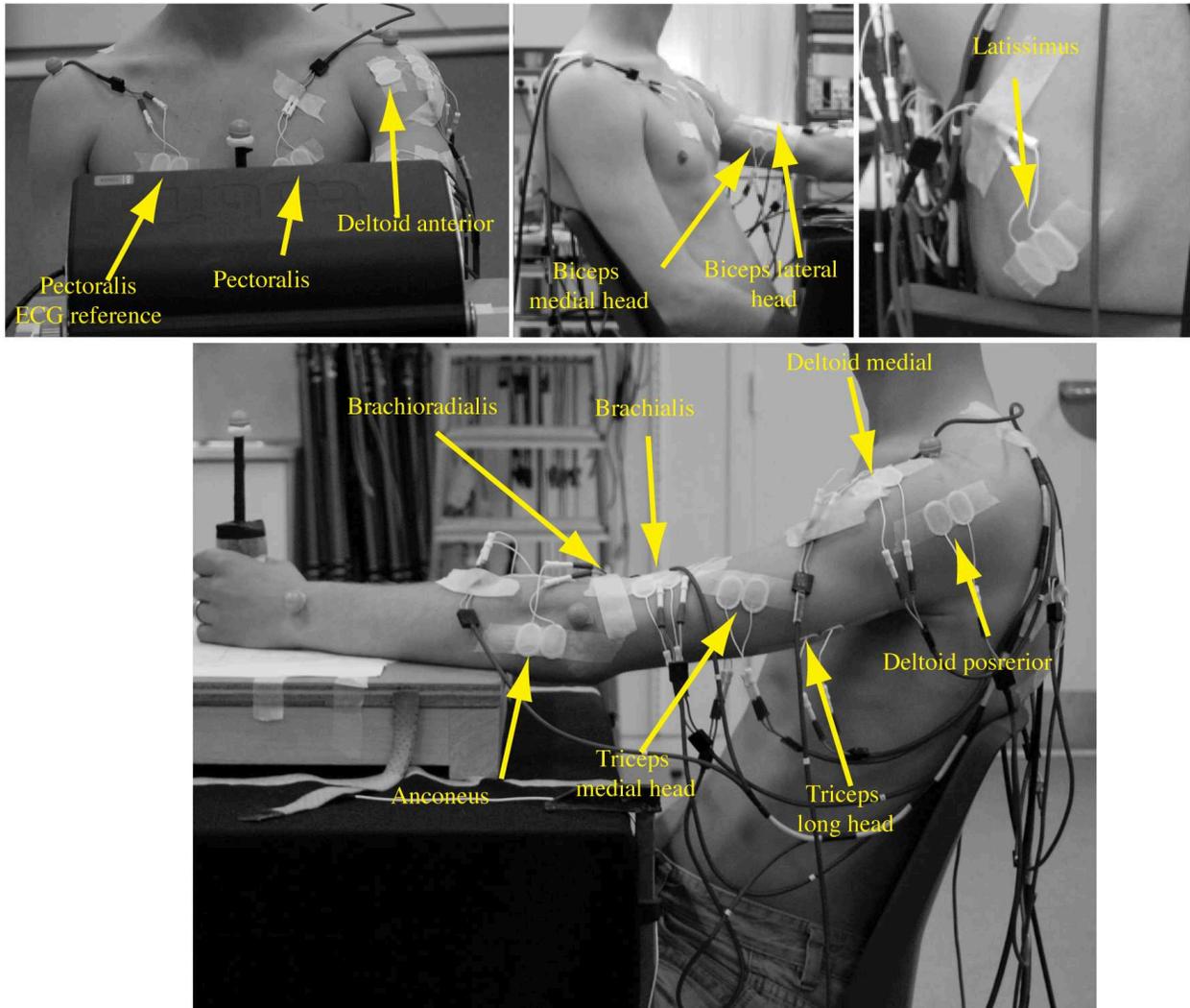
The reference electrode was a conductive wrist band soaked in water prior to the experiment to increase conductivity. Bipolar electrode pairs were placed in the following 12 muscle sites

1. Brachioradialis (BIO, 1/6 of the distance ranging from the midpoint between the cubit fossa and the lateral epicondyle to the styloid process of ulna)
2. Anconeus (ANC, 2 cm distal to the midpoint between the lateral epicondyle and the olecranon process)
3. Biceps brachii medial head (BME)
4. Biceps brachii lateral head (BLA)
5. Brachialis (BIA, 4 cm in the direction towards the acromion, from the midpoint between the fossa cubit and the lateral epicondyle)
6. Triceps lateral head (TLA)
7. Triceps long head (TLO)
8. Deltoid medial part (DME)
9. Pectoralis major (PEC, 1/3 of the distance ranging from the glenohumeral joint to the lowest point of the sternum)
10. Deltoid anterior part (DAN)
11. Deltoid posterior part (DPO)
12. Latissimus dorsi (LAT, 4 cm below the inferior border of the scapula, half the distance between the spine and the lateral edge of the body)

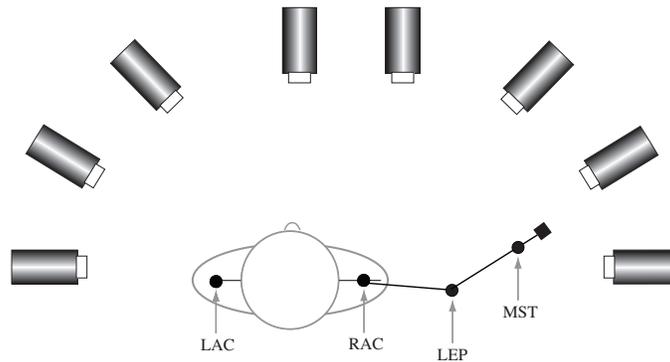
For muscles where no specific electrode placement is mentioned, the electrodes were placed according to SENIAM recommendations [50]. A pair of electrodes was placed over the left Pectoralis major and served as a ECG reference. The electrode placements are illustrated in figure 5.8.

#### Motion capture

High speed cameras tracked the changing positions and orientations of the upper limb segments via reflective ball shaped markers (diameter 18 mm) placed superficial at the shoulder and arm of the subject, i.e. at the left acromion



**Figure 5.8:** Electrode placement. BIO (brachioradialis), ANC (anconeus), BME (biceps brachii medial head), BLA (biceps brachii lateral head), BIA (brachialis), TLA (triceps lateral head), TLO (triceps long head), DME (deltoid medial part), PEC (pectoralis major), DAN (deltoid anterior part), DPO (deltoid posterior part), LAT (latissimus dorsi).



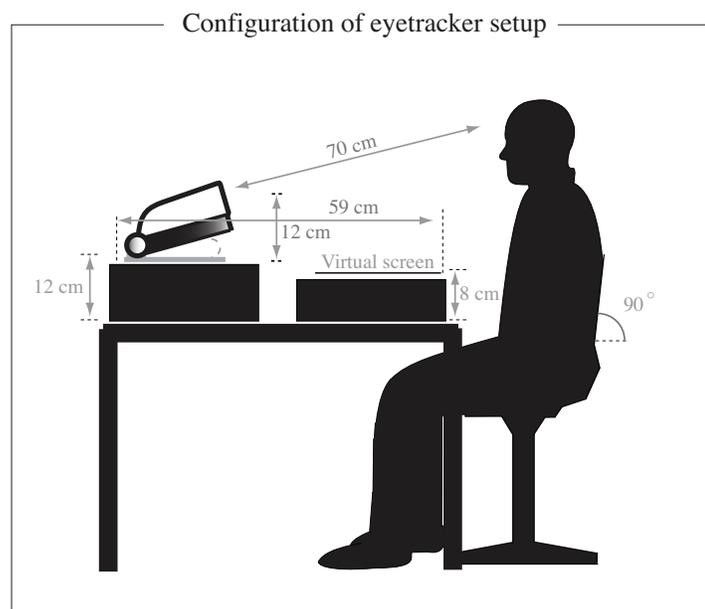
**Figure 5.9:** Placement of the four reflective markers at left acromion (LAC), right acromion (RAC), lateral epicondyle of humerus (LEP) and at the posterior midpoint between styloid processes of radius and ulna (MST). The position was recorded using the eight shown high-speed infrared digital cameras.

(LAC), right acromion (RAC), lateral epicondyle of humerus (LEP) and posterior midpoint between styloid processes of radius and ulna (MST) as illustrated in figure 5.9

The position of the markers were recorded using a motion tracking system (Qualisys Track Manager, Qualisys AB, Gothenburg, Sweden) with 8 infrared digital video cameras (ProReflex MCU, Qualisys AB, Gothenburg, Sweden). Data was recorded with a sampling frequency of 240 Hz.

### Gaze-direction

Gaze-direction was recorded using an eyetracker (Tobii X120 Eye-Tracker, firmware 1.1.14) sampled at 120 Hz, and acquired using Tobii Studio v.1.3.23 running on Windows XP. The eye tracker communicated with the computer via a standard network cable [155]. Figure 5.10 shows the eyetracking setup.



**Figure 5.10:** Relevant measures for the eyetracking setup. The subject was placed approximately 70 cm. from the eye-tracker in an upright position ( $90^\circ$  with the horizontal plane).

The distance from the eye-tracker to the subject was approximately 70 cm, adjusted to achieve the optimal configuration. The placement of the eye-tracker ensured that the gaze angle did not exceed  $35^\circ$  to any point on the virtual screen.

**Distance to calibration grid** 59 cm, measured from the back edge of the eye-tracker to the front end of the virtual screen

**Horizontal eyetracker angle** Adjusted with respect to each subject, such that the eyes are placed in the center of the status track window when the subject is placed 70 cm from the eye-tracker. Approx.  $15^\circ$

**Height from table to virtual screen** 8 cm

**Height from table to eyetracker foot** Approx. 12 cm, adjusted to the individual subject, so ensure optimal tracking of the eyes

Initially the eyetracker was calibrated, to learn the characteristics of the subject's eye movement to achieve accurate estimation of the gaze points, and to associate a position of the eye to a specific location in the working space. During calibration, the subject concentrated on focusing the eyes on five calibration points, depicting the extremity points and the center of the workspace.

An elaborated description of the configuration of the specific eyetracker is located in appendix chapter G.

## 5.2.4 Post processing

The post processing methods used for prediction and evaluation are based on the methods presented in chapter 4 on page 29.

### Extracting the muscle activation

The EMG signals were band pass filtered (4<sup>th</sup> order zero-lag Butterworth digital filter, pass band 20-400 Hz) to attenuate DC offset, motion artifacts and high frequency noise [50].

The filtered signal was full wave rectified and low pass filtered (4<sup>th</sup> order zero-lag (the data was processed in both forward and reverse direction) Butterworth digital filter, cut-off frequency 1 Hz) to obtain the muscle activation patterns. See also section 4.1.

### Projection of markers to the horizontal plane

The 3D positions of the reflective markers were projected to the horizontal plane by discarding the Z-coordinate since the subject performed planar arm movement, i.e. the positions were projected to the horizontal plane.

### Synchronization of signals

The EMG, motion capture, and eyetracking were synchronized as illustrated on figure 5.11.

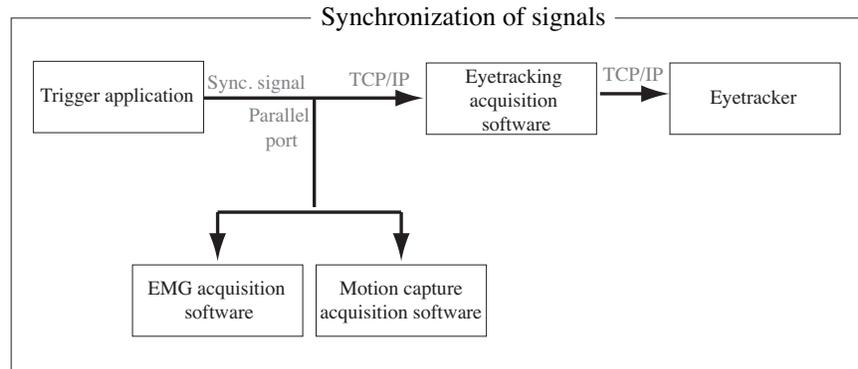
An external application was developed to start the Eyetracker, log events and at an event send a signal to the other recording system via the PCs parallel port. The logged event and signal sent to the other recording systems were used as synchronization signal.

The signal from the trigger-application was initiated when the recordings were already started on the EMG, the motion capture, and eyetracking systems. All samples recorded from the three recording systems before and after the synchronization signal indicating start and stop of recordings were discarded.

The development of the synchronization configuration is explained in detail in appendix H on page 123.

### Resampling to a common sampling frequency

To reduce the computation time, the EMG envelopes (sample freq. 2048 Hz), motion capture recordings (sample freq. 240 Hz), and the eyetracking (sample freq. 120 Hz) were resampled to a common sampling frequency of 40 Hz



**Figure 5.11:** Synchronization of the sEMG, motion capture and eyetracking recordings.

by interpolation. The resampling induced no significant loss of information in the EMG envelopes since frequency content above 20 Hz was highly attenuated. A 40 Hz sampling frequency for the motion capture and eyetracking was also sufficient since no human movements are expected to have components faster than 10 Hz.

### Reduction of ECG artifact in the EMG recordings

Due to EMG recordings near the heart, such as electrode locations at the Pectoralis major, and Latissimus as shown by [95], a ECG reference channel was recorded simultaneously with the sEMG and used for removal of the ECG contamination by a Least Mean Squares (LMS) adaptive transversal (50 taps) filter with a step size of  $5 \cdot 10^{-8}$  [95].

### Prediction of the users intent

The predictions were based on a MLP NN as presented in section 4.3.

The network consisted of 1 hidden layer with 10 neurons, for the prediction of both the muscle activity, motor commands, and the kinematics, using the sigmoid transfer function in all units. The transfer function for the output layer was a linear function returning the input passed to it. Learning of the neural network was based on the Levenberg-Marquardt back-propagation.

The number of neurons was based on the analysis of the  $R^2$  as a function of number of neurons in the hidden layer as outlined in figure 5.12, and elaborated in appendix chapter 6 on page 57.

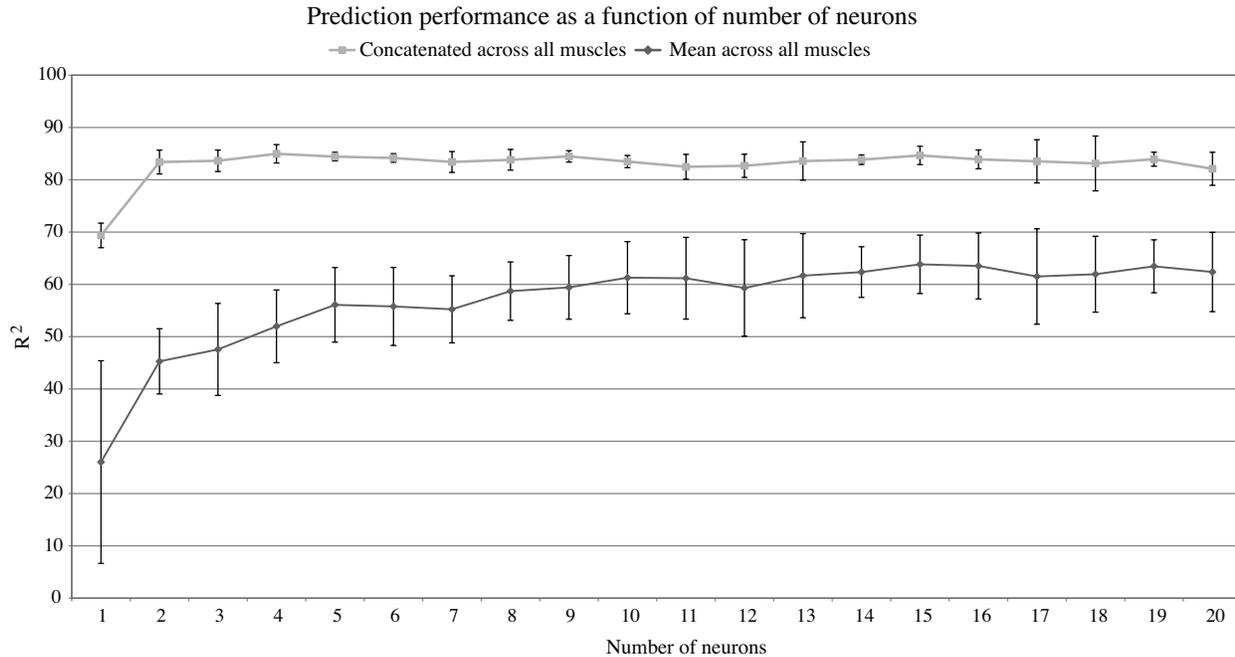
Initially the relation between input and output patterns was learned by the network in a training session. The training stopped if the maximum number of repetitions were reached, or the validation performance increased more than 6 times since the last time it decreased [28].

Following the training session was the validation session, which reconstructed the output signal. The inputs for this session were not part of the training data.

The feedforward neural network was implemented in Matlab<sup>®</sup> using the Neural Network Toolbox Matlab<sup>®</sup> [28] using `newff[training input, training target, hidden layer size(s), transfer function(s) from hidden and output layer]` (creates the feed-forward network object), `train[network object, training input, training target]` (trains the neural network), and `sim[input, network object]` (simulates the network). The `newff` automatically initializes the weights with the Nguyen-Widrow layer initialization function, which initializes a layer's weights and biases according to the Nguyen-Widrow initialization algorithm [105, 111]. The algorithm chooses values in order to distribute the active region of each neuron (determined by the transfer function) in the layer approximately evenly across the layer's input space. Due to a degree of randomness in the values, they are not initialized to the same value each time the function is called.

The initialization of the weights can influence the performance of the neural network, thus each predictions was performed 3 times to ensure that the network solution was not trapped in any local minimum.

Levenberg-Marquardt algorithm was used for training, which also was part of the neural network toolbox for Matlab. In Matlab it is implemented in Matlab<sup>®</sup> via a standard backpropagation technique [43]. The weight and the biases of



**Figure 5.12:** Prediction performance of the muscle activity as a function of number of neurons depicted from the concatenated data (light grey) and for mean across all muscles.

the network were iteratively adjusted to minimize the network performance function chosen as the mean square error between the estimated and true outputs.

To make the training more efficient, the inputs and outputs to the NN was preprocessed, per default in the implementation in Matlab. Network-input processing functions transform the input into a better form for the network, i.e. the inputs and outputs were scaled to fall within a specified range  $[-1, 1]$  and processing functions associated with the network output transformed targets into a better form for network training, and reverse transformed outputs back to the characteristics of the original target data. [28]

**Summary of parameters in neural network** The feed-forward neural network was configured with the following settings [28]

**Number of hidden layers** 1

**Number of neurons in hidden layer** 10 for both the prediction of kinematics and muscle activity

**Transfer functions** 'log-sig' and 'purelin' function for the hidden and output layer respectively

**Learning algorithm** Levenberg-Marquardt back-propagation using the standard configuration in Matlab (for  $\mu$ ,  $\mu^+$ ,  $\mu^-$ ,  $\mu_{max}$ ) [28]

**Maximum number of repetitions** 150

**Initialization algorithm** Nguyen-Widrow

**Performance function** Mean square error

**Maximum validation failures** 6

The following subsections describe the predictions depending on the type of signal to be predicted, and on which data basis. The first 80% of the data was used for training, and the last 20% which was not included in the training, was used for validation.

### Prediction configurations

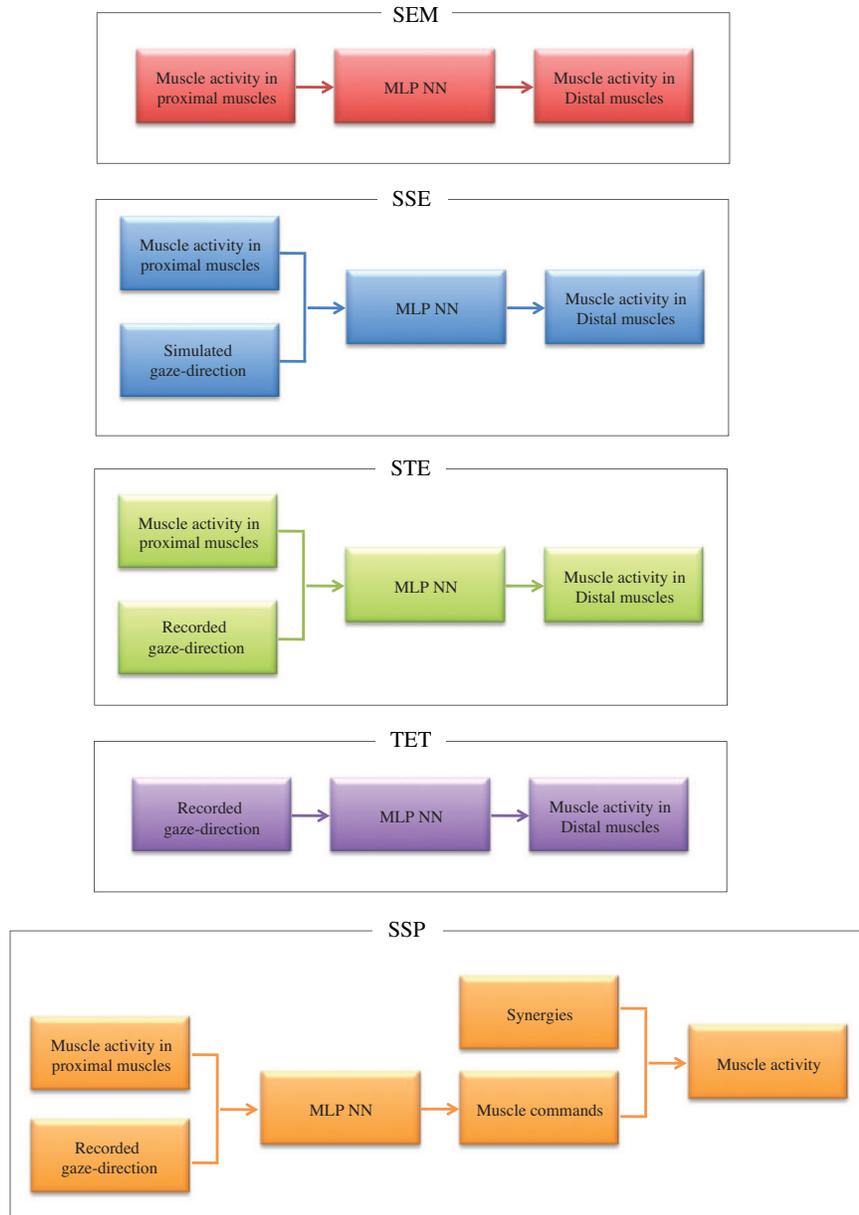
The different variations of the predictions are summarized in table 5.1 and are named according to the prediction variation, and colored. For each configuration, the inputs is displayed and so is each of the outputs that is predicted by the use of the individual configurations.

Conf.	Input			Output		
	sEMG <sub>p</sub>	True gaze	Sim. gaze	sEMG <sub>d</sub>	Motor commands	Kinematics
<b>SEM</b>	X			X		X
<b>SSE</b>	X		X	X		X
<b>STE</b>	X	X		X		X
<b>TET</b>		X			X	X
<b>SSP</b>	X	X			X	

**Table 5.1:** Variations of the input and output for the different prediction configurations. sEMG<sub>p</sub> being the muscle activity in the proximal Deltoid muscle, true gaze being the recorded gaze-direction, sEMG<sub>d</sub> being the muscle activity in the distal muscles, motor commands being the low dimensional motor commands, kinematics being the position of the endpoint of the hand.

The colors and abbreviation of the variations follows throughout the rest of the report. Figure 5.13 depicts an overview of the different configurations

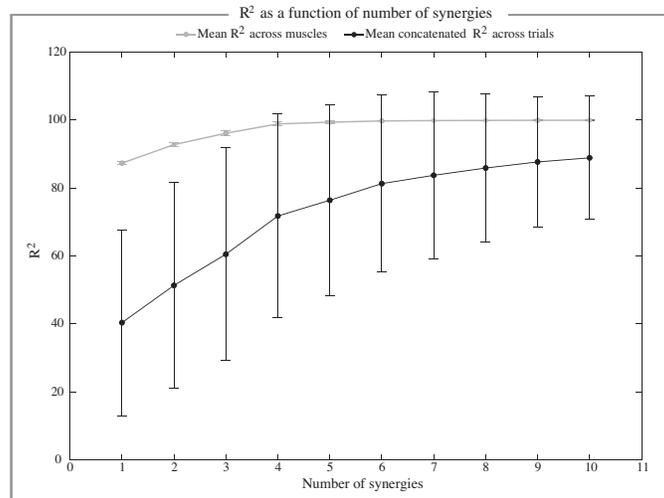
The muscle activity in the SSP configuration was reconstructed from the predicted motor commands and the four synergies, c.f. appendix section E on page 103 extracted from the training data.



**Figure 5.13:** Overview of the prediction configurations. Shown only for the muscle activity as output. SEM (red) is based on the muscle activity in the Deltoid muscle as input, and are used for prediction of muscle activity in the distal muscles and kinematics. SSE (blue) is based on the muscle activity in the Deltoid muscle, and simulated gaze-direction for the prediction of muscle activity in the distal muscles and kinematics. STE (green) is based on the muscle activity in the Deltoid muscle and recorded gaze-direction, and used for the prediction of the muscle activity in the distal muscles and kinematics. TET (purple) is based on the recorded gaze-direction, and used for the prediction of the muscle activity in the distal muscles and also the kinematics. Finally, SSP (orange) is based on the muscle activity in the Deltoid muscle and recorded gaze-direction, and used for the prediction of the low dimensional motor commands. The muscle activation level is reconstructed by use of the synergy matrix extracted from the training data and the predicted motor commands.

### Extraction of synergies and motor commands

The motor commands and muscle synergies were extracted from the muscle activation level from all muscles, both proximal and distal muscles, from the training data. The extracted motor commands (training target), the muscle activity (from the proximal muscles), and recorded gaze-direction served as basis for predicting the motor commands. The muscle synergies and motor commands were extracted based on NMF as described in section 4.4.3 on page 38. Figure 5.14 depicts the reconstruction performance as a function of number of synergies.



**Figure 5.14:** Reconstruction performance of predicting the motor commands in the individual muscles (black, mean  $\pm$  SD across the number of muscles) and for the concatenated data for all muscles (grey, mean  $\pm$  SD across the three trials) as a function of number of synergies.

Since the aim was to predict the muscle activity in the individual muscles, 4 synergies were chosen to be extracted based. Further details of this analysis is found in appendix, chapter E.

### Prediction performance

The predictions of the kinematics and muscle activations, and the reconstruction of the muscle activations from the synergies and motor commands, were evaluated by  $R^2$ , introduced in equation 4.17 on page 39. For both the prediction of the muscle activation and the kinematics,  $R^2$  was both computed for the individual muscles and across all muscles, i.e. for the muscle activations these were concatenated for all muscles and for the kinematics the position in  $x$  and  $y$  was concatenated.



---

# Results

---

**III**



The analysis in this chapter is the basis for the results outlined in the introduction in section 2.5 on page 19, and served as basis for indicating whether the muscle activity from the proximal shoulder muscles in the Deltoid contained sufficient information, to allow decoding of the muscle activity in the distal muscles of the forearm and the kinematics of the hand (hand endpoint position).

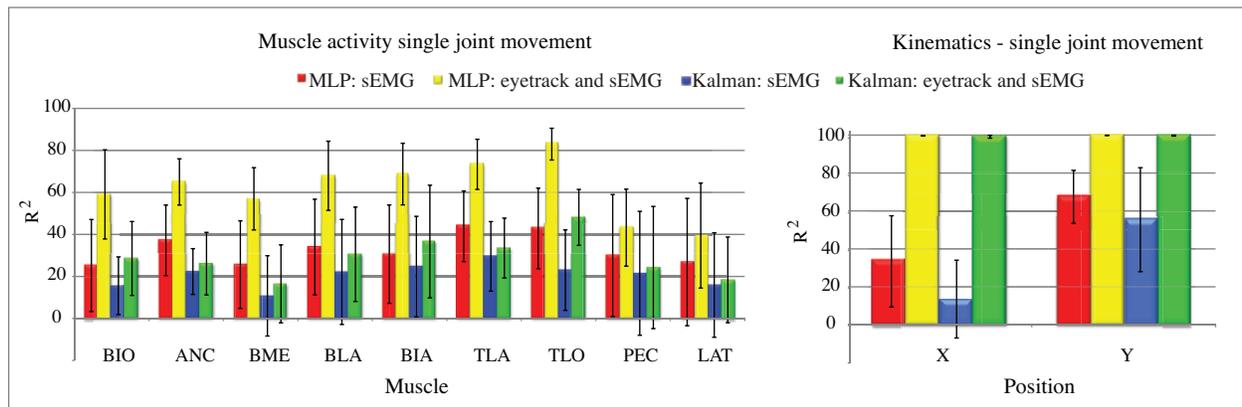
In addition it was the aim to investigate the effect of adding eyetracking to the prediction basis. Hence these results was the foundation for the evolvement of the main focus of this project.

Finally two methods of predictions were compared, namely one based on the Kalman filter and one based on the MLP neural network.

The results obtained was based on data obtained by [95] as outlined in chapter 5.1 on page 41.

## 6.1 Prediction of kinematics and muscle activity

Figure 6.1 and 6.2 shows the evaluation (in means of  $R^2$ ) of the predictions of the kinematics and the muscle activity respectively utilizing both the Kalman filter and the MLP NN with and without simulated eyetracking. The results are represented as mean across all seven subjects  $\pm$  SD. The MLP NN was configured according to the protocol in chapter 5.2 and was optimized with respect to both the kinematic and the muscle activity predictions.

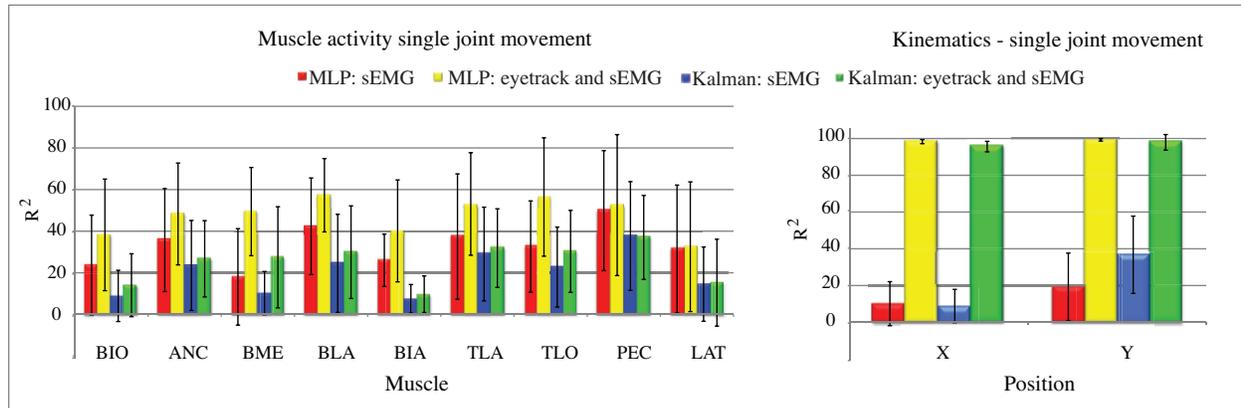


**Figure 6.1:** Comparison of predictions using the Kalman filter and the MLP NN of the kinematics in the single joint movement based on the muscle activity in the three proximal deltoid muscles with (green and yellow respectively) and without simulated eyetracking (blue and red respectively).

Prediction of the muscle activity seen on figure 6.1 (left) based on the single joint movement revealed that the MLP NN using sEMG combined with eyetracking yielded the best prediction performance with an  $R^2$  up to 83% for Triceps long head. This combination produced on average across all muscles an prediction performance of approx. 62% compared to 33% for the prediction based on MLP using only sEMG. The Kalman filter yielded on average a prediction of the muscle activity based on the sEMG only of approx. 21%, and with eyetracking included approx. 29%.

The prediction of the kinematics seen on figure 6.1 (right) was at approx. 100% on average across the two directions for both of the prediction methods and with eyetracking included. Based on the sEMG only, the prediction using the MLP NN gave a prediction performance of approx. 51% and the Kalman filter a prediction of 34%. The outcome of the kinematics predictions were as expected, since the simulated eyetracking is almost similar to the true kinematics.

Predictions of the muscle activity and kinematics for the multi joint movements seen on figure 6.2 (left and right respectively) showed similar characteristics as the predictions during the single joint movement.



**Figure 6.2:** Comparison of predictions using the Kalman filter and the MLP NN of the kinematics in the multi joint movement based on the muscle activity in the three proximal deltoid muscles with (green and yellow respectively) and without simulated eyetracking (blue and red respectively).

The muscle activity was on average across all muscles using the MLP NN based on sEMG only predicted with a performance of approx. 33%, and combining the sEMG with the eyetracking yielded a performance of approx. 48%. Predictions of the muscle activity based on the Kalman filter based on the sEMG and sEMG and eyetracking yielded on average approx. 20% and 25% respectively. The eyetracking also improved the prediction of the kinematics to the multi joint movement yielding a performance of approx. 99% and 97% for the MLP NN and the Kalman filter respectively.

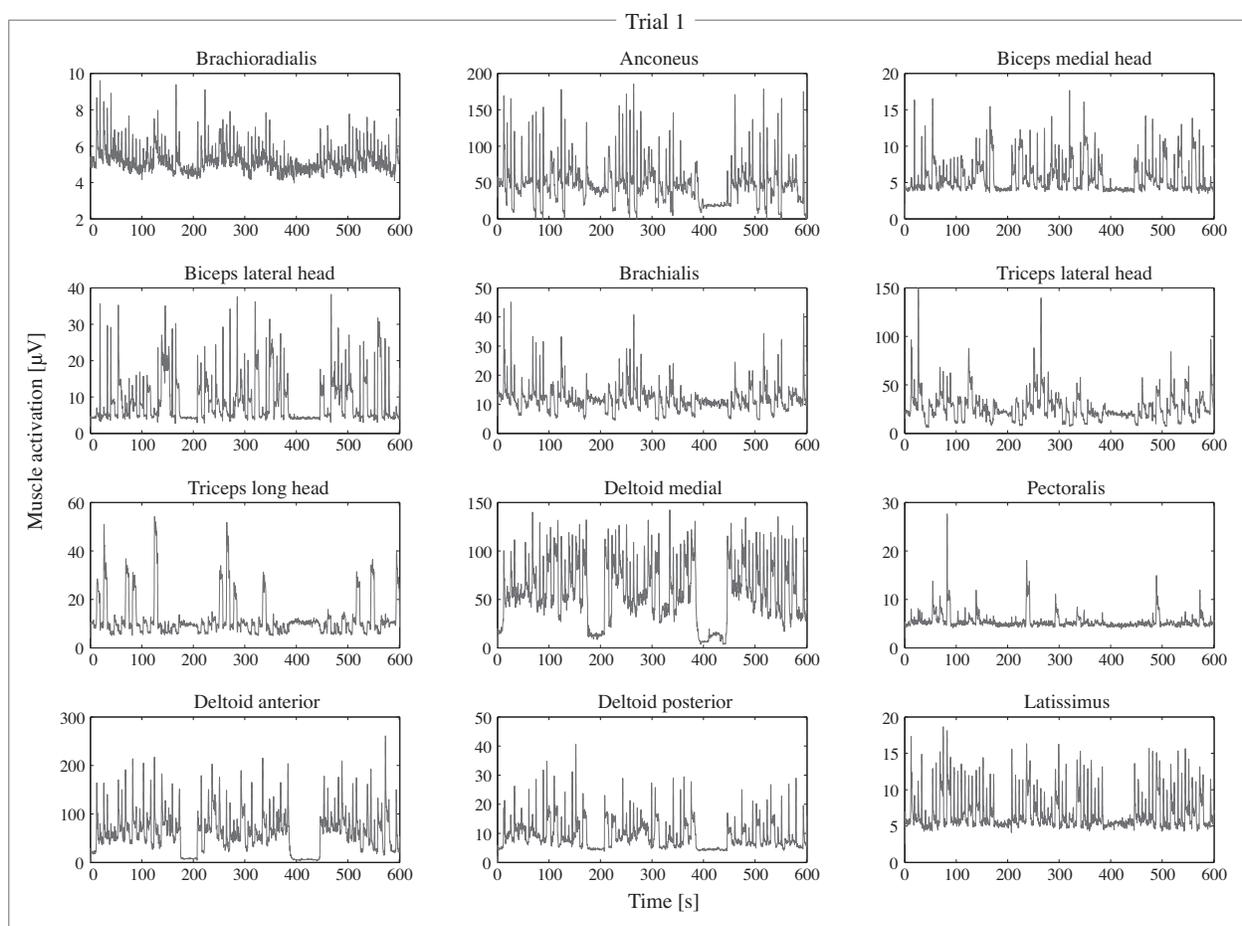
From the predictions seen in figure 6.1 and 6.2 it was concluded that the MLP NN yielded the best predictions of both the kinematics and the muscle activity. Furthermore it appeared clearly that the simulated eyetracking improved all predictions compared to using only the sEMG.

This chapter contains the results achieved from the conducted experiments.

## 7.1 Recorded data

The trials performed by subject 1 followed the experimental protocol in section 5.2.

Figure 7.1 depicts the recorded muscle activity for subject 1 for all 12 muscles shown from in the time interval 200-400 s.

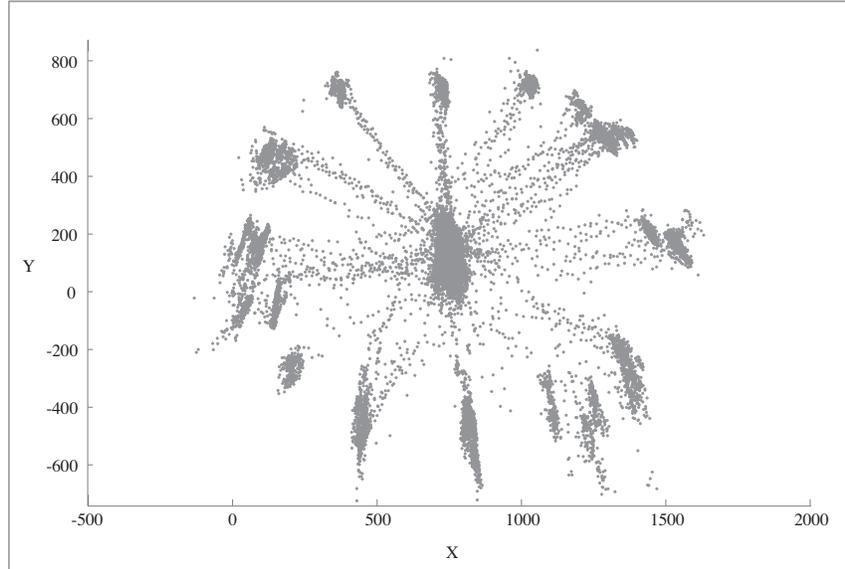


**Figure 7.1:** Recorded muscle activity from trial 1.

From figure 7.1 the movements was seen as clear peaks in the muscle activity. Not all 12 muscles participated in the different movements, e.g. as seen for Pectoralis. It was besides noted that the muscle activity did not reach the baseline before a new movement was started seen in all cases, e.g. as was seen in the peaks for Triceps long head and in Biceps medial head after approx. 225 s.

Figure 7.2 shows the recorded eyetrack from trial 1.

From figure 7.2 it was seen that the subject followed the hand endpoint very precise, since the 12 targets were clearly identified from the eyetracking. The dense areas in the figure were due to the 5 s pauses between each reaching movement, and thus more fixation points were recorded during these periods.



**Figure 7.2:** Recorded gaze-direction from trial 1.

## 7.2 Prediction of muscle activity

This subsection presents the predictions of the muscle activity in the nine distal muscles based on the different configurations introduced in the protocol in chapter 5.2 on page 43.

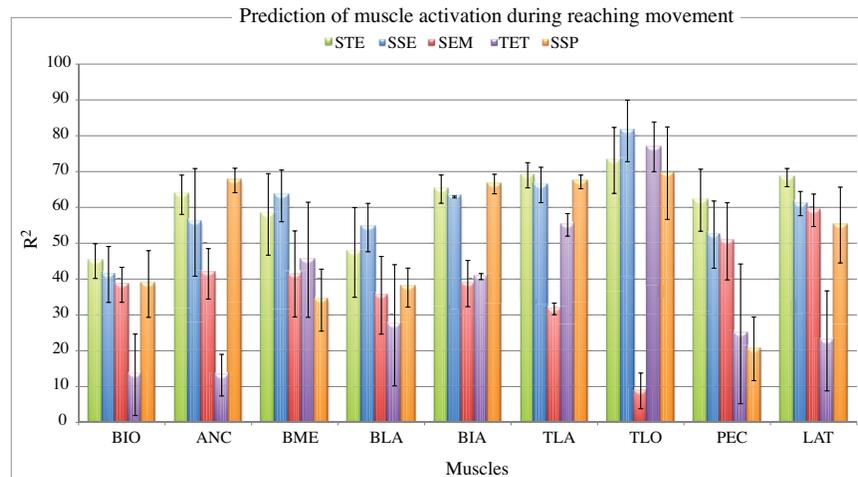
The evaluation of the predictions is depicted in figure 7.3 on the facing page based on the different configurations summarized in table 7.1. The evaluations were provided as mean  $\pm$  SD across the three trials.

Conf.	Input			Output		
	sEMG <sub>p</sub>	True gaze	Sim. gaze	sEMG <sub>d</sub>	Motor commands	Kinematics
<b>SEM</b>	X			X		X
<b>SSE</b>	X		X	X		X
<b>STE</b>	X	X		X		X
<b>TET</b>		X			X	X
<b>SSP</b>	X	X			X	

**Table 7.1:** Variations of the input and output for the different prediction configurations. sEMG<sub>p</sub> being the muscle activity in the proximal Deltoid muscle, true gaze being the recorded gaze-direction, sEMG<sub>d</sub> being the muscle activity in the distal muscles, motor commands being the low dimensional motor commands, kinematics being the position of the endpoint of the hand.

STE based on the sEMG and the recorded gaze-direction varies from approximate  $45.0\% \pm 4.8\%$  in Brachioradialis up to approx.  $73.1\% \pm 9.2\%$  in Triceps long head with an overall prediction average of  $61.3\% \pm 7.0\%$  (average of evaluations shown across all muscles). Concatenation of the predictions across all muscles yielded an  $R^2$  of  $83.5\% \pm 1.2$ .

SSE based on the simulated gaze-direction and the sEMG followed trend of the predictions based on the sEMG and recorded gaze-direction. Predictions from  $41.3\% \pm 7.8\%$  (Brachioradialis) up to  $81.3\% \pm 8.6\%$  (Triceps long head) was achieved. On average the muscle activity was predicted with an  $R^2$  of approx.  $59.9\% \pm 7.0\%$ . Concatenation



**Figure 7.3:** Evaluation of the predictions of the muscle activation in the 9 distal muscles. The  $R^2$  is shown for all the three trials (mean  $\pm$  SD) for each individual muscle for the predictions using the different configurations. SEM (red) was based on the muscle activity in the Deltoid muscle as input. SSE (blue) was based on the muscle activity in the Deltoid muscle, and simulated gaze-direction. STE (green) was based on the muscle activity in the Deltoid muscle and recorded gaze-direction. TET (purple) was based on the recorded gaze-direction. Finally, SSP (orange) was based on the muscle activity in the Deltoid muscle and recorded gaze-direction, and used for the prediction of the low dimensional motor commands. The muscle activation level was reconstructed by use of the synergy matrix extracted from the training data (from all 12 muscles) and the predicted motor commands.

resulted in an evaluation on  $80.6\% \pm 3.6\%$ .

SEM based on the muscle activation yielded predictions from approx.  $8.8\% \pm 5.0\%$  (Triceps long head) up to  $59.2\%$  (Latissimus) and on average across all the muscles the muscle activity was predicted with an  $R^2$  of  $38.4\% \pm 7.0$ . Predictions on the concatenated muscle activity yielded an  $R^2$  on  $71.5\% \pm 2.2\%$ .

TET based on gaze-direction revealed predictions from  $13.2\% \pm 5.8\%$  (Anconeus) up to  $76.9\% \pm 6.9\%$  (Triceps long head). The average predictions gave an  $R^2$  of  $35.5\% \pm 10.5\%$  and predictions on the concatenated data resulted in an  $R^2$  on  $66.0\% \pm 3.6\%$ .

SSP predicting the motor commands based on sEMG and eyetracking. Prediction results yielded  $R^2$  values of  $20.6\% \pm 8.9\%$  (Pectoralis) and up to  $69.6\% \pm 12.9\%$  (Triceps long head) and average across all muscles of  $50.7\% \pm 7.1\%$ . The prediction of the concatenated data yielded an  $R^2$  of  $84.7\% \pm 2.0\%$ .

The evaluations revealed that the predictions using STE yielded the highest predictions in all muscles. SSP showed a good performance in many cases, similar to the performance of the STE. In Pectoralis, the SEM showed a better performance than the SSP.

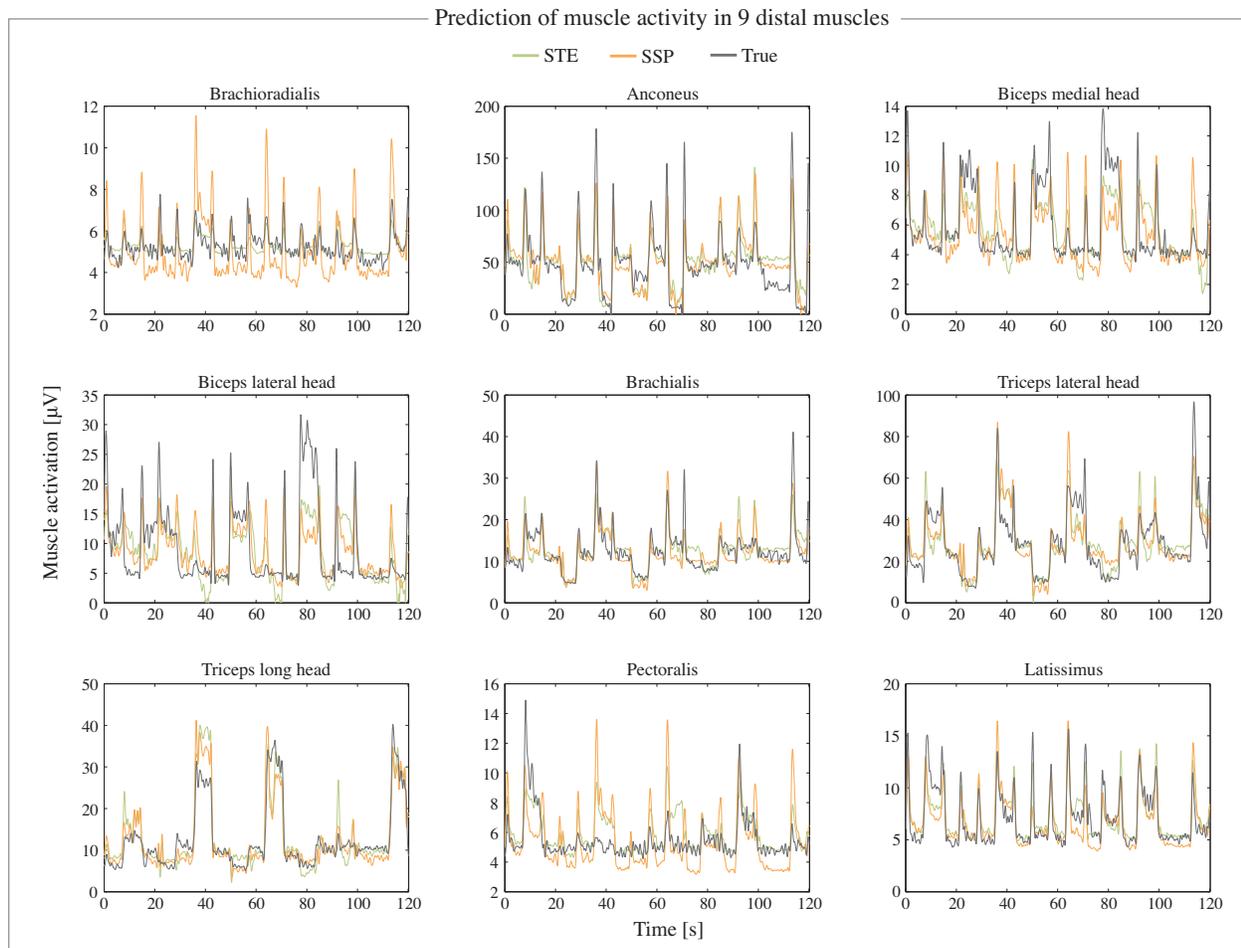
Only SSE showed close to a similar prediction performance in the predictions, but since this was based on simulated gaze-direction, this served as a comparison with the recorded gaze-direction. It was indicated from the simulated gaze-direction, which was almost similar to the true endpoint position of the hand, that it was of crucial influence on the predictions, in correspondence with the recorded gaze-direction.

Especially the predictions of the muscle activity in Triceps long head and Triceps lateral head based on STE, SSP, and TET showed a high  $R^2$  and in the contrary showed a low performance for SEM indicating that the predictions of the muscle activity in these muscles were highly dependent on the gaze-direction and concurrently, the sEMG did not contain much information about the reaching tasks, i.e. the muscles were active for all targets but contained not enough information about the movements to be able to predict the muscle activation. STE and SSP improved the predictions with almost 36% for Triceps lateral head, and approx. 61% for Triceps long head.

In all cases it was seen that the sEMG and the gaze-direction yielded higher predictions when combined (STE and SSP) than individually (SEM and TET).

From these results it was indicated that the optimal configuration for predicting the muscle activity was based on the sEMG and gaze-direction, and even suggested that the muscle activity could be reconstructed satisfactory in many cases based on the predictions of the motor commands.

Figure 7.4 depicts the predictions of all 9 distal muscles



**Figure 7.4:** Prediction of muscle activity in all 9 distal muscles using STE (green) based on the muscle activity in the Deltoid muscle and recorded gaze-direction. SSP (orange) was based on the muscle activity in the Deltoid muscle and recorded gaze-direction, and used for the prediction of the low dimensional motor commands. The muscle activation level was reconstructed by use of the synergy matrix extracted from the training data (based on all 12 muscles) and the predicted motor commands. The predictions were compared to the true muscle activity (grey).

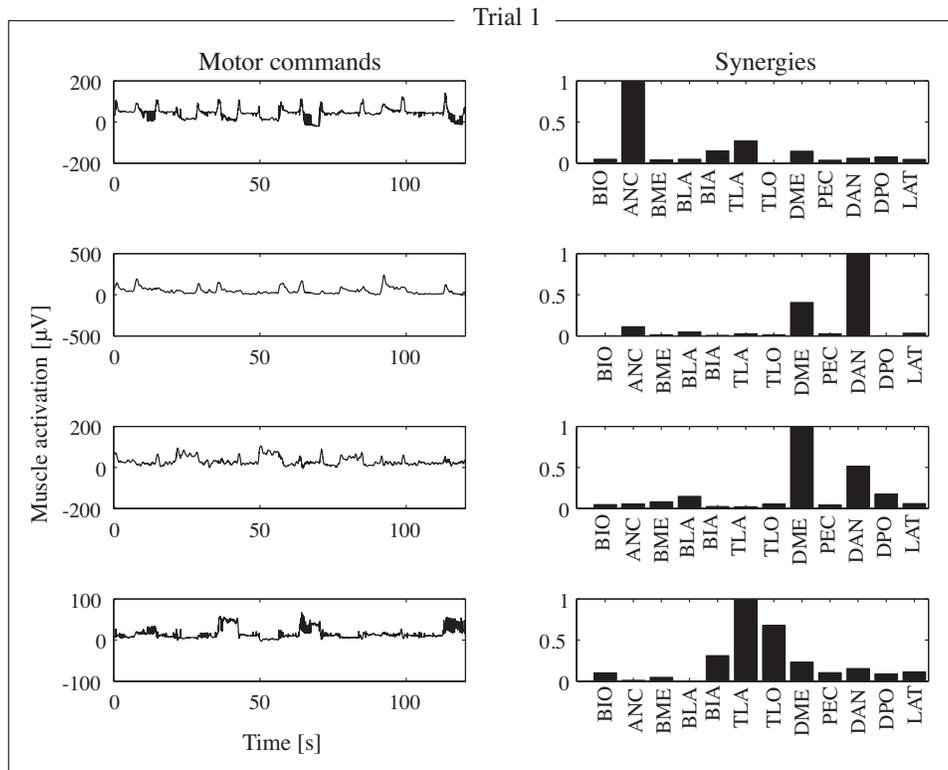
From figure 7.4 it was seen that the muscle activity in all muscles were predicted quite well throughout the entire period. The predictions captured most aspects of the muscle activity. Both configurations tended to have some characteristic prediction errors, including not capturing the extreme peaks of the muscle activity, e.g. in the Biceps lateral head, Biceps medial head and Latissimus there are cases of underestimation and for Brachioradialis, and Pectoralis overestimation was seen in some cases for the SSP configuration especially. This was noticeable for all cases where the muscle activity did not reach the baseline before the onset of a new movement. During the quiescent periods, the SSP configuration seemed to underestimate the activity, e.g. in Brachioradialis and Pectoralis.

### 7.3 Prediction of motor commands

This subsection presents the results from the predictions based configuration *SSP*, implying the prediction of the motor commands and reconstruction of the muscle activity. The extracted synergies from the training data and the predicted motor commands are shown in figure 7.5-7.7 for trial 1-3 respectively.

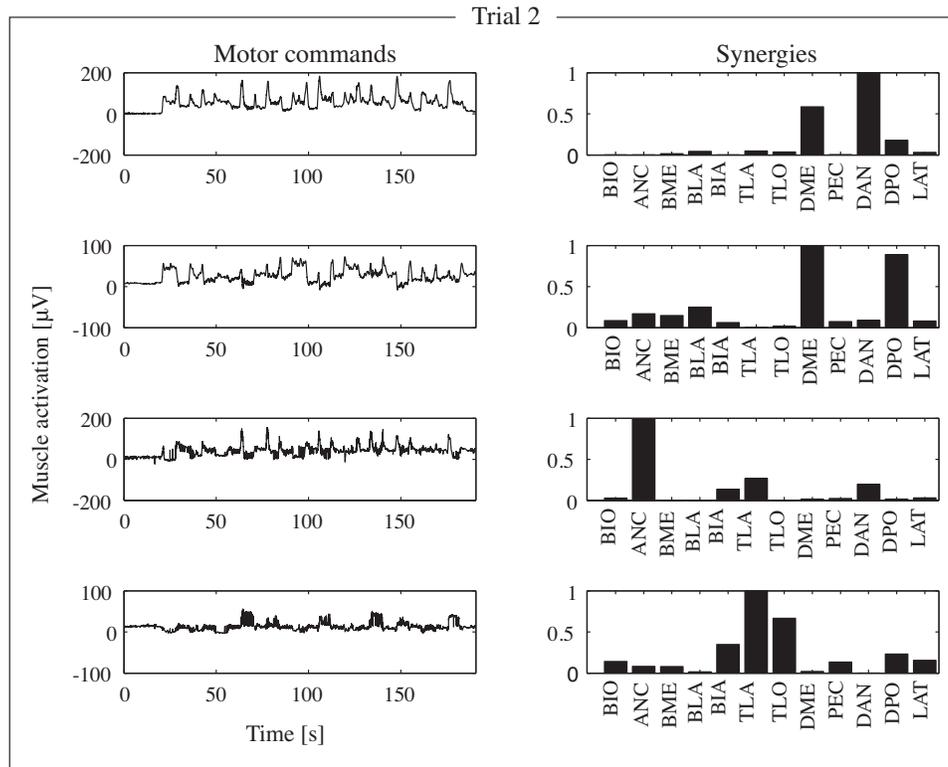
7.5 showed the synergies and motor commands from trial 1. It was seen that the muscles were maximally active in only one synergy at the time. E.g. Anconeus was active only in the first synergy, responsible for the extension of the forearm. Deltoid anterior was primarily active in the second synergy responsible for flexion of the shoulder. Deltoid medial was active in the third synergy responsible for abduction of the shoulder.

Synergy 4 consisted of the primary extensor muscles Triceps Lateral head, and Triceps long head.



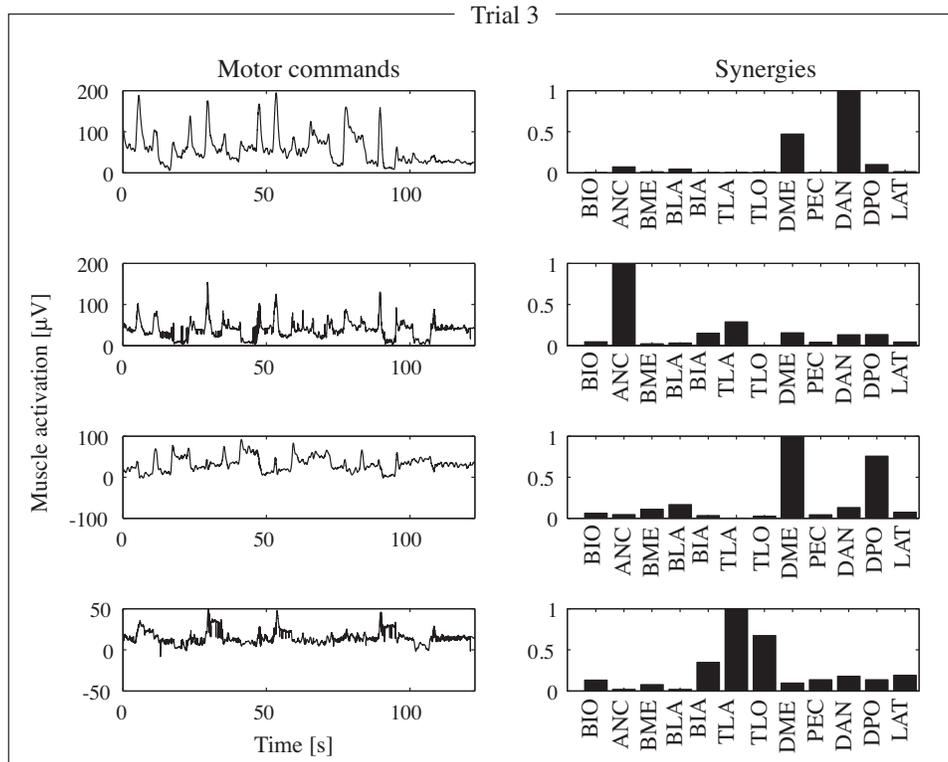
**Figure 7.5:** Extracted synergies and predicted motor commands normalized, corresponding to predictions using configuration *SSP* based on the muscle activity in the Deltoid muscle and recorded gaze-direction, and used for the prediction of the low dimensional motor commands. The muscle activation level was reconstructed by use of the synergy matrix extracted from the training data (all 12 muscles) and the predicted motor commands. Shown for trial 1.

Trial two, depicted in figure 7.6, showed a similar pattern for the synergies as in trial 1. Synergy one with the Deltoid anterior being mainly active. Synergy 2 with Deltoid posterior and medial being most active. Anconeus was mainly active in the third synergy and synergy 4 consisted primarily of the active extensor muscles Triceps lateral and long head.



**Figure 7.6:** Extracted synergies and predicted motor commands normalized, corresponding to predictions using configuration SSP based on the muscle activity in the Deltoid muscle and recorded gaze-direction, and used for the prediction of the low dimensional motor commands. The muscle activation level was reconstructed by use of the synergy matrix extracted from the training data (all 12 muscles) and the predicted motor commands. Shown for trial 2.

Trial three, shown in figure 7.7 showed similar characteristics to the first two trials, showing synergy 1 with Deltoid medial and anterior being mainly active, synergy 2 with Anconeus being mainly active, and synergy 3 with the Deltoid medial and anterior being most active, and synergy 4 with Triceps lateral head and long head being mainly active.



**Figure 7.7:** Extracted synergies and predicted motor commands normalized, corresponding to predictions using configuration SSP based on the muscle activity in the Deltoid muscle and recorded gaze-direction, and used for the prediction of the low dimensional motor commands. The muscle activation level was reconstructed by use of the synergy matrix extracted from the training data (all 12 muscles) and the predicted motor commands. Shown for trial 3.

### 7.3.1 Summary

The synergies from the three trials revealed very similar patterns and it appeared as if the synergies could be attributed to separation of movements, e.g. flexion, extension, or abduction as suggested by e.g. [95].

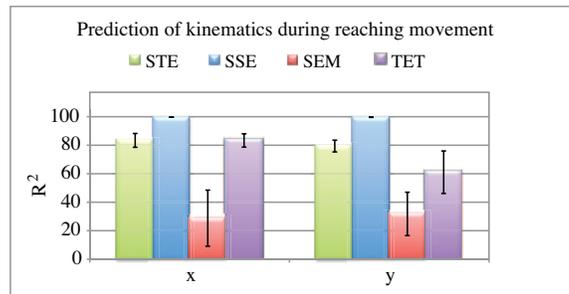
Furthermore it was indicated that one synergy was active during quiescent segments, e.g. synergy 2 in trial 2 with the Deltoid posterior and medial being most active, since these are primarily responsible for abduction of the shoulder.

From the synergies it was revealed that the primarily active muscles were Anconeus, medial, anterior and posterior part of the Deltoid, Triceps long head and Triceps lateral head, and the Pectoralis.

## 7.4 Prediction of kinematics

This subsection presents the prediction of kinematics corresponding to the endpoint of the hand ( $x$ - and  $y$ -coordinate for marker 4).

The evaluation of the predictions is depicted in figure 7.8 based on STE (green), SSE (blue), SEM (red), TET (purple) as mean  $\pm$  SD across the three trials.



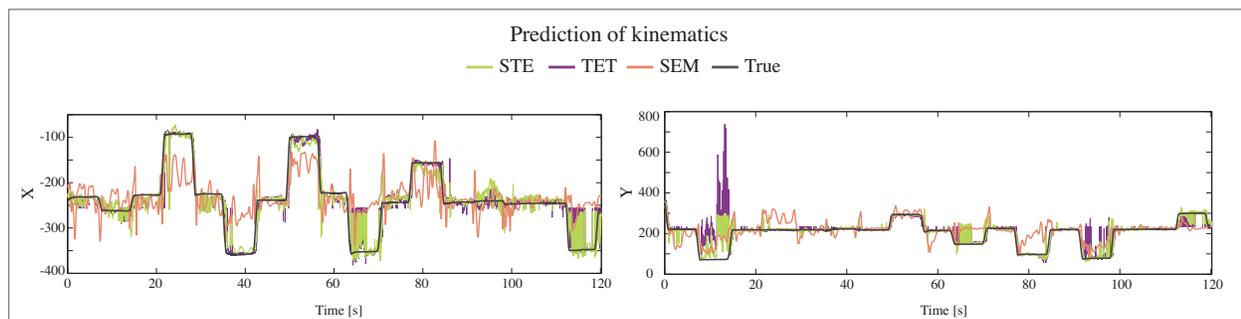
**Figure 7.8:** Evaluation of kinematics predictions. The  $R^2$  is shown for all the three trials (mean  $\pm$  SD) for each movement direction. SEM (red) was based on the muscle activity in the Deltoid muscle as input. SSE (blue) was based on the muscle activity in the Deltoid muscle, and simulated gaze-direction. STE (green) was based on the muscle activity in the Deltoid muscle and recorded gaze-direction. TET (purple) was based on the recorded gaze-direction.

SEM showed an  $R^2$  of  $83.4\% \pm 4.9$  in  $x$  and  $79.4\% \pm 4.1$  in  $y$ , and TET showed a prediction performance of  $83.4\% \pm 4.7$  in  $x$  and  $61.0\% \pm 14.9$  in  $y$ , whereas STE yielded an prediction of  $83.4\% \pm 4.9$  in  $x$  and  $79.4\% \pm 4.1$  in  $y$ . As expected, SSE yielded a very good prediction ( $99.7\% \pm 0.0$  in  $x$  and  $99.6\% \pm 0.1$  in  $y$ ), since this basically equals the true kinematics, though with noise added.

gaze-direction alone did show a prediction of 55% better in  $x$  and 30% better in  $y$  compared to the predictions achieved with the sEMG alone.

It was seen that the combination of gaze-direction and sEMG (STE) outperformed the predictions based on the signals individually. Additionally the standard deviation was decreased compared to the SEM in both  $x$  and  $y$  and than TET in  $y$ .

Figure 7.9 shows the predicted position (in  $x$  and  $y$  respectively) for trial 1 utilizing SEM, STE, and the true position.



**Figure 7.9:** Prediction of position in  $x$  (left), and  $y$ -direction (right). SEM (red) was based on the muscle activity in the Deltoid muscle as input. STE (green) was based on the muscle activity in the Deltoid muscle and recorded gaze-direction. TET (purple) was based on the recorded gaze-direction. The configurations was compared to the true position (grey).

Left shows for SEM that the prediction follows the trend of the true position, but throughout the period, the estimate appeared noisy.

STE in the contrary showed a better performance, e.g. all peaks were identified, though in some cases the position was predicting excessive noisy in the negative decreasing peaks, which also characterized some periods with the TET configuration. Both STE and TET follows the overall trend of the true signal, and especially the rising peaks and the

first falling peak were well predicted. From figure 7.9 (right), the position based on SEM was followed somewhat the true position, not reaching the peaks of the positions and showed sporadically peaks in the prediction. The predictions based on STE showed a trend close to the true position, but in some cases the position was predicting excessive noisy.

## 7.5 Summary

From the above observations it was clear that the muscle activity could be predicted very well based on the muscle activity from the deltoid muscle in combination with eyetracking. Clearly it was revealed that the inclusion of an extra modality, i.e. using both sEMG and eyetracking, indeed improved the prediction performance of the muscle activity, compared to using only one modality, i.e. sEMG or eyetracking separately. Likewise, predictions of the motor commands (SSP) and reconstruction of the muscle activity yielded in many cases a similar performance as achieved using both modalities (STE).

In general the predictions showed that the overall trend of the true muscle activity was followed well for all muscles. It was seen that the EMG did not reach the baseline in multiple muscles before a new movement is started and in these intervals the muscle activity was often under estimated.

Similar for the prediction of the kinematics, adding an additional modality, clearly improved the prediction performance compared to predictions based on sEMG only and minor compared to predictions based on the eyetracking only.

The predictions of the kinematics revealed somewhat noisy estimates both based on the sEMG or eyetracking individually.



---

# **Recapitulation and future work**

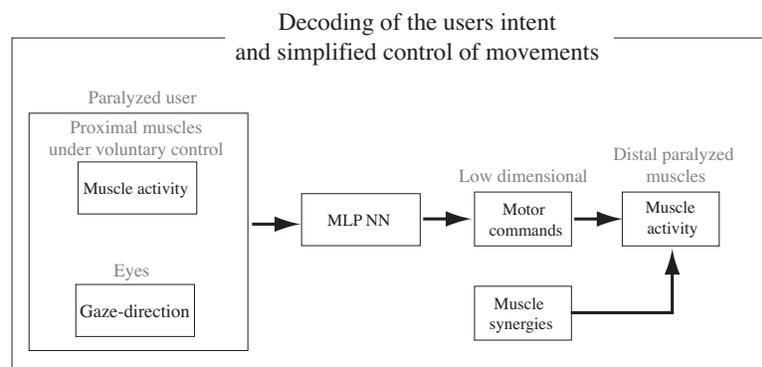
---

**IV**



The overall objective of this work was to contribute to the development of an upper-limb FES and a FET system applicable for clinical use. The systems should be able to restore abilities regarded critical for independent living, aiming to provide the possibility to improve living for high-level tetraplegic in case of FES, and for individuals with reduced voluntary control of muscles, e.g. due to a prolonged fixation of the joint, in case of FET.

Specifically, a configuration was proposed to predict the users intent based on signals recorded non-invasively. In addition the possibility of simplifying the control of electrical stimulation was investigated, applicable for stimulation in FES and FET as depicted in figure 8.1



**Figure 8.1:** The users intent expressed as the muscle activity in the distal muscles was decoded based on the muscle activity in the proximal muscles, and the control of movement was simplified by muscle synergies.

The users intent was from preliminary data in 7 subjects predicted based on surface EMG and simulated gaze-direction, which indicated a clear improvement compared to predictions based on the muscle activity only. A Multi Layer Perceptron neural network using the Levenberg-Marquardt algorithm for training outperformed the predictions achieved with a Kalman filter.

Based on non-invasively recorded EMG from the Deltoid muscle in combination with gaze-direction, it was shown, that the muscle activation level in 9 distal muscles in the arm, could be predicted continuously from one subject. Further it was shown, that the muscle activity in the 9 muscles could be reconstructed based on a low dimensional muscle activation pattern, indicating a simplified approach for the stimulation control in FES or FET. This low dimensional approach was based on muscle synergies, extracted by non-negative matrix factorization. Muscle synergies have been defined as being sets of muscles whose relative activations are believed to be neurally predetermined.

The accuracy of the predictions in the main part of the muscles was comparable to what others previously have shown for the prediction of the muscle activity based on intracortical recordings [119], and for prediction of kinematic signals [140, 169]. Though it should be noted that in [119] the EMG envelopes had a bandwidth of 10 Hz compared to 1 Hz in this work, which was approx. similar to that of the kinematic studies.

The  $R^2$  reported for the concatenated data across all muscles yielded substantially higher performance, letting the most active muscles influence the performance measure mostly.

The hand endpoint position was likewise predicted in one subject yielding a clear improvement by combining the muscle activity and gaze-direction compared to the predictions based on the individual signals. The predictions showed a performance in the range of what previous has been reported based on intracortical recordings.

Selecting the appropriate muscle activation levels to achieve a given task is extremely complex. [41] have previously selected the set of muscles based on the MSE that should serve as controller input for stimulation. The approach taken in this work was related to a physiological strategy for the control of movements of the arm, which is believed to be the natural approach of the CNS to simplify the complexity in the control of movements, namely by spinally

organized muscle groupings, which can be controlled as units [159, 25, 27].

The approach of specifying a low-dimensional approach has the advantage of being able to control individual D.O.F compared to e.g. predetermined trajectories based on limb kinematics, in which fixed trajectories are based on kinematic a coupling between D.O.F, i.e. where at least one joint follows another joint in some predetermined manner [89].

In addition, direct prediction of muscle activity allows a more extensively range of movements, due to the substantial challenge associated with the identification of the muscle stimulation pattern to elicit specific movements [3] from arm kinematics, e.g. [109, 60, 51, 11, 29]. In endpoint control strategy, the kinematic control relies on the assignment of resolving joint positions corresponding to the endpoint trajectory. Dynamical control relies on finding the joint torques that will cause the arm to follow a specified trajectory [89].

Several studies in the area of muscle synergies have reported, that a set of basis synergies was capable of describing the observed data sets. Only a few have reported, that the extracted synergies formed a predictive framework, i.e. being able to describe movement tasks others than from which the synergies were extracted [95, 1]. This work indicates that the extracted synergies were capable of reconstructing the muscle activity during a variety of multi joint reaching movements, similar to those from which the synergies were extracted.

An attractive approach would be to investigate to what extent muscle synergies extracted from healthy subjects could be applied for control in paralyzed subjects, and even to what extent synergies extracted from one side of a subject could reconstruct the muscle activity in the opposite site. This would first and foremost shed light on the generality and robustness of the synergies for the predictions of muscle activation patterns of new movements, and secondly it is believed to find its usefulness in FES and FET applications.

The current work investigated the prediction of the muscle activity, which was accomplished by computing the mapping between multiple muscle activation patterns in the proximal shoulder muscle and those in the distal arm muscles. This mapping was utilized in a decoder applied on offline data, which was not part of the mapping construction, and validation was evaluated by the coefficient of determination. A similar prediction in real time could serve as basis for electrical stimulation.

Such a system would thus utilize what is believed to be natural muscle synergies and integrate remaining voluntary control with FES. Hence the user should simply attempt to move their hand and let the gaze-direction follow a given trajectory to a location, to which the controller should respond with the appropriate stimulation levels to the paralyzed muscles, grouped as muscle synergies.

The results presented in this work did not state exactly how the prediction and low-dimensional control of stimulation are implemented in a FES and FET system. However, the potential of the methods has been indicated.

In general the results should be considered with a note of caution, due to the fact that this was a proof of concept, based on 1 subject only. The robustness of the presented methods should be tested, i.e. validation should be based on a more substantial data foundation both for inter- and intra-subject signal variability.

Also, in this work the performance of the methods was evaluated based on data from one healthy subject, with an implicit assumption that the characteristics will carry over to a fundamental different population. It is expected that differences exists in the muscle activation patterns between healthy and movement-impaired subjects. Furthermore, the predictions were based on the muscle activity recorded from the proximal Deltoid muscles, which could be affected in high-level tetraplegics yielding a poorer range of motion [57]. Hence decoding based on the specific target group is a necessity.

Additionally the performance of the predictions should be applied as control signal for the stimulation in a FES and FET and resulting kinematics should be evaluated. Initially this could be performed on a musculoskeletal model of the upper extremity. A model as developed by e.g. [11] could facilitate testing the electrical stimulation based on the predicted intend of the user. This model can simulate different degrees of SCI and can take muscle denervation, disuse atrophy, and limited muscle activation due to electrode placement into account. This testing could provide insight into the effect on the kinematics caused by the electrical stimulation based on the decoded intent.

Future work should additionally demonstrate the capability of individuals to accurately control the muscle synergies in a real-time FES/FET application.

Any increase in the reconstruction accuracy can be of importance when the signal has to be used for control. Even modest improvement can show up to have disproportionate importance for a control application. Hence it is suggested that further improvements of the decoding should be pursued both for the decoding method and recordings.



# Bibliography

- [1] A. B. Ajiboye and R. F. Weir. Muscle synergies as a predictive framework for the emg patterns of new hand postures. *Journal of neural engineering*, 6, 2009.
- [2] A. J. Andersen, J. Horsewell, B. Ovesen, and V. Rasmussen. *Para- og tetraplegi - haandbog for rygmarsvsskadede*. RYK - Rykmarsvsskadede i Danmark, first edition, 2002.
- [3] C. V. Anderson and A. J. Fuglevand. Probability-based prediction of activity in multiple arm muscles: Implications for functional electrical stimulation. *J Neurophysiol*, 100(1):482–494, 2008.
- [4] A. B. Arsenault, D. A. Winter, and R. G. Marteniuk. Is there a normal profile of emg activity in gait? *Medical and Biological Engineering and Computing*, 24(4):337–343, 1985.
- [5] A. T. C. Au and R. F. Kirsch. Emg-based prediction of shoulder and elbow kinematics in able-bodied and spinal cord injured individuals. *Rehabilitation Engineering, IEEE Transactions on*, 8(4):471–480, 2000.
- [6] R. Balasubramaniam and A. G. Feldman. Guiding movements without redundancy problems. *Coordination Dynamics*, 2004.
- [7] A. P. Batista, B. M. Yu, G. Santhanam, S. I. Ryu, A. Afshar, and K. V. Shenoy. Cortical neural prosthesis performance improves when eye position is monitored. *IEEE Trans Neural Syst Rehabil Eng*, 16(1):24–31, 2008.
- [8] F. Biering-Soerensen. Rygmarsvsskade - den moderne behandling. *Ugeskrift for laeger*, 162(20), 2001.
- [9] E. Bizzi, N. Hogan, F. A. Mussa-Ivaldi, and S. Giszter. Does the nervous system use equilibrium-point control to guide single and multiple joint movements? *Behavioral and Brain Sciences*, 15:603–613, 1992.
- [10] E. Bizzi, M. C. Tresch, P. Saltiel, and A. D’avella. New perspectives on spinal motor systems. *Nat Rev Neurosci.*, 1(2):101–108, 2000.
- [11] D. Blana, J. G. Hincapie, E. K. Chadwick, and R. F. Kirsch. A musculoskeletal model of the upper extremity for use in the development of neuroprosthetic systems. *J Biomech*, 41(8):1714–1721, 2008.
- [12] B. Blankertz, G. Curio, and K.-R. MÅijller. Classifying single trial eeg: Towards brain computer interfacing. In *Advances in Neural Information Processing Systems 14 - Fifteenth Neural Information Processing Systems*, 2001.
- [13] A. M. Bryden, W. D. Memberg, and P. E. Crago. Electrically stimulated elbow extension in persons with c5/c6 tetraplegia: A functional and physiological evaluation. *Arch Phys Med Rehabil*, 81(1):80–88, 2000.
- [14] M. Bundgaard. President for society for spinal cord injuries in denmark, February 2009 2009.
- [15] G. Cappellini, Y. P. Ivanenko, R. E. Poppele, and F. Lacquaniti. Motor patterns in human walking and running. *J Neurophysiol*, 95(6):3426–3437, 2006.
- [16] F. Carpi and D. D. Rossi. Non invasive brain-machine interfaces. Technical report, University of Pisa, 2006.

- [17] A. D. C. Chan and K. B. Englehart. Continuous myoelectric control for powered prostheses using hidden markov models. *Biomedical Engineering, IEEE Transactions on*, 52(1):121–124, 2005.
- [18] G. Cheron, J. P. Draye, M. Bourgeois, and G. Libert. A dynamic neural network identification of electromyography and arm trajectory relationship during complex movements. *IEEE Trans Biomed Eng*, 43(5):552–558, 1996.
- [19] C. D. Clemente. *Anatomy - a regional atlas of the human body*. Lippincott Williams & Wilkins, 5 edition, 2006.
- [20] P. E. Crago, N. Lan, P. H. Veltink, J. J. Abbas, and C. Kantor. New control strategies for neuroprosthetic systems. *Journal of Rehabilitation Research and Development*, 33(2):158–172, 1996.
- [21] P. E. Crago, W. D. Memberg, M. K. Usey, M. W. Keith, R. F. Kirsch, G. J. Chapman, M. A. Katorgi, and E. J. Perreault. An elbow extension neuroprosthesis for individuals with tetraplegia. *IEEE Trans Rehabil Eng*, 6(1):1–6, 1998.
- [22] G. H. Creasey, C. H. Ho, R. J. Triolo, D. R. Gater, A. F. Dimarco, K. M. Bogie, and M. W. Keith. Clinical applications of electrical stimulation after spinal cord injury. *J Spinal Cord Med*, 27(4):365–375, 2004.
- [23] G. H. Creasy, K. L. Kilgore, D. L. Brown-Triolo, J. E. Dahlberg, P. H. Peckham, and M. W. Keith. Reduction of costs of disability using neuroprostheses. *Assist Technol*, 12(1):67–75, 2000.
- [24] G. Cybenko. Approximations by superpositions of sigmoidal functions. *Math. Control, Signals, Systems*, 2:303–314, 1989.
- [25] A. D’avella. Decomposition of emg patterns as combinations of time-varying muscle synergies. In *Proceedings of the 1st international IEEE EMBS, conference on Neural Engineering*, pages 55–58, 2003.
- [26] A. D’avella and E. Bizzi. Shared and specific muscle synergies in natural motor behaviors. *Proceedings of the National Academy of Sciences of the United States of America*, 102(8):3076–3081, 2005.
- [27] A. D’avella, P. Saltiel, and E. Bizzi. Combinations of muscle synergies in the construction of a natural behavior. *Nature Neuroscience*, 6:300–308, 2003.
- [28] H. Demuth, M. Beale, and M. Hagan. *Neural network toolbox 6 users guide*. The MathWorks, 2009.
- [29] L. Dipietro, A. Sabatini, and P. Dario. Artificial neural network model of the mapping between electromyographic activation and trajectory patterns in free-arm movements. *Medical and Biological Engineering and Computing*, 41(2):124–132, 2003.
- [30] K. C. Engel and J. F. Soechting. Interactions between ocular motor and manual responses during two-dimensional tracking. *Prog Brain Res*, 142:141–153, 2003.
- [31] K. Englehart and B. Hudgins. A robust, real-time control scheme for multifunction myoelectric control. *Biomedical Engineering, IEEE Transactions on*, 50(7):848–854, 2003.
- [32] K. Englehart, B. Hudgins, P. A. Parker, and M. Stevenson. Classification of the myoelectric signal using time-frequency based representations. *Medical engineering and physics*, 21(6-7):431–438, 1999.
- [33] D. Farina, R. Merletti, and R. M. Enoka. The extraction of neural strategies from the surface emg. *Journal of Applied Physiology*, 96:1486 – 1495, 2004.
- [34] T. Flash and B. Hochner. Motor primitives in vertebrates and invertebrates. *Current Opinion in Neurobiology*, 15(6):660–666, 2005.
- [35] O. Fukuda, J. Arita, and T. Tsuji. An emg-controlled omnidirectional pointing device. *Systems and Computers in Japan*, 37(4):1996–2003, 2006.

- [36] K. Funahashi. On the approximate realization of continuous mappings by neural networks. *Neural Networks*, 2(3), 1989.
- [37] D. A. Gabriel. Shoulder and elbow muscle activity in goal-directed arm movements. *Exp Brain Res*, 116(2):359–366, 1997.
- [38] A. P. Georgopoulos and J. Ashe. One motor cortex, two different views. *Nature Neuroscience*, 3:963, 2000.
- [39] A. P. Georgopoulos, J. F. Kalaska, R. Caminiti, and J. T. Massey. On the relations between the direction of two-dimensional arm movements and cell discharge in primate motor cortex. *J. Neurosci.*, 2(11):1527–1537, 1982.
- [40] A. P. Georgopoulos, R. E. Kettner, and A. B. Schwartz. Primate motor cortex and free arm movements to visual targets in three-dimensional space. ii. coding of the direction of movement by a neuronal population. *J Neurosci*, 8(8):2928–2937, 1988.
- [41] J. P. Giuffrida and P. E. Crago. Utilizing remaining voluntary muscle synergies to control fes elbow extension after spinal cord injury, 2004.
- [42] J. H. Grill and P. H. Peckham. Functional neuromuscular stimulation for combined control of elbow extension and hand grasp in c5 and c6 quadriplegics. *IEEE Trans Rehabil Eng*, 6(2):190–199, 1998.
- [43] M. T. Hagan and M. Menhaj. Training feed-forward networks with the marquardt algorithm. *IEEE Transactions on Neural Networks*, 5(6):989–993, 1994.
- [44] P. S. Hammon, S. Makeig, H. Poizner, E. Todorov, and V. R. De Sa. Predicting reaching targets from human eeg. *Signal Processing Magazine, IEEE*, 25(1):69–77, 2008.
- [45] Y. Handa. Current topics in clinical fes in japan. *Journal of Electromyography and Kinesiology*, 7(4):269–274, 1997.
- [46] C. E. Hansen, H. Rishave, and P. Tangh. *Haandbog om para- og tetraplegi*. Paraplegikerkredsen, second edition, 1986.
- [47] R. L. Hart, K. L. Kilgore, and P. H. Peckham. A comparison between control methods for implanted fes hand-grasp systems. *Rehabilitation Engineering, IEEE Transactions on*, 6(2):208–218, 1998.
- [48] E. Hartman, J. D. Keeler, and J. M. Kowalski. Layered neural networks with gaussian hidden units as universal approximations. *Neural Computation*, 2(2):210–215, 1990.
- [49] R. Herman, J. He, S. D’luzansky, W. Willis, and S. Dilli. Spinal cord stimulation facilitates functional walking in a chronic, incomplete spinal cord injured. *Spinal Cord*, 40(2):65–68, 2002.
- [50] H. J. Hermens, B. Freriks, R. Merletti, D. Stegeman, J. Blok, G. Rau, C. Disselhorst-Klug, and G. HÅd’gg. *European recommendations for surface electromyography, results of the seniam project*. Roessingh Research and Development b.v., 1999.
- [51] C. Hogfors, G. Sigholm, and P. Herberts. Biomechanical model of the human shoulder - 1. elements. *J Biomech*, 20(2):157–166, 1987.
- [52] W. P. Hope and H. Rassoulain. Modular neural networks applied to surface emg signals for limb function identification. In *Systems, Man, and Cybernetics, 1999. IEEE SMC '99 Conference Proceedings. 1999 IEEE International Conference on*, volume 2, pages 418–423 vol.412, 1999.
- [53] N. Hoshimiya, A. Naito, M. Yajima, and Y. Handa. A multichannel fes system for the restoration of motor functions in high spinal cord injury patients: A respiration-controlled system for multijoint upper extremity. *IEEE Trans Biomed Eng*, 36(7):754–760, 1989.

- [54] Y. Huang, K. B. Englehart, B. Hudgins, and A. D. Chan. A gaussian mixture model based classification scheme for myoelectric control of powered upper limb prostheses. *IEEE Trans Biomed Eng*, 52(11):1801–1811, 2005.
- [55] F. Ishida, K. Karatsu, and Y. Sakaguchi. Muscle synergies extracted from human grasping movements. *International Congress Series*, 1301:110–113, 2007.
- [56] M. W. Jiang, R. C. Wang, J. Z. Wang, and D. W. Jin. A method of recognizing finger motion using wavelet transform of surface emg signal. In *Engineering in Medicine and Biology Society, 2005. IEEE-EMBS 2005. 27th Annual International Conference of the*, pages 2672–2674, 2005.
- [57] M. W. Johnson and P. H. Peckham. Evaluation of shoulder movement as a command control source. *Biomedical Engineering, IEEE Transactions on*, 37(9):876–885, 1990.
- [58] B. A. Kakulas. Neuropathology: The foundation for new treatments in spinal cord injury. *Spinal Cord*, 42(10):549–563, 2004.
- [59] R. E. Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME—Journal of Basic Engineering*, 82(Series D):35–45, 1960.
- [60] D. Karlsson and B. Peterson. Towards a model for force predictions in the human shoulder. *J Biomech*, 25(2):189–199, 1992.
- [61] M. W. Keith, P. H. Peckham, G. B. Thrope, K. C. Stroh, B. Smith, J. R. Buckett, K. L. Kilgore, and J. W. Jatich. Implantable functional neuromuscular stimulation in the tetraplegic hand. *J Hand Surg [Am]*, 14(3):524–530, 1989.
- [62] C. T. Kelley. *Iterative methods for optimization*. Society for Industrial and Applied Mathematics - SIAM, 1999.
- [63] M. F. Kelly, P. A. Parker, and R. N. Scott. The application of neural networks to myoelectric signal analysis: A preliminary study. *IEEE Trans Biomed Eng*, 37(3):221–230, 1990.
- [64] M. Khezri and M. Jahed. Introducing a new multi-wavelet function suitable for semg signal to identify hand motion commands. In *Engineering in Medicine and Biology Society, 2007. EMBS 2007. 29th Annual International Conference of the IEEE*, pages 1924–1927, 2007.
- [65] M. Khezri and M. Jahed. A novel approach to recognize hand movements via semg patterns. In *Engineering in Medicine and Biology Society, 2007. EMBS 2007. 29th Annual International Conference of the IEEE*, pages 4907–4910, 2007.
- [66] K. L. Kilgore, P. H. Peckham, M. W. Keith, G. B. Thrope, K. S. Wuolle, A. M. Bryden, and R. L. Hart. An implanted upper-extremity neuroprosthesis. follow-up of five patients. *J Bone Joint Surg Am*, 79(4):533–541, 1997.
- [67] T. Kitamura, N. Tsujiuchi, and T. Koizumi. Hand motion estimation by emg signals using linear multiple regression models. In *Engineering in Medicine and Biology Society, 2006. EMBS '06. 28th Annual International Conference of the IEEE*, pages 1339–1342, 2006.
- [68] M. D. Klein Breteler, K. J. Simura, and M. Flanders. Timing of muscle activation in a hand movement sequence. *Cereb. Cortex*, 17(4):803–815, 2007.
- [69] Y. Koike, H. Hirose, Y. Sakurai, and T. Iijima. Prediction of arm trajectory from a small number of neuron activities in the primary motor cortex. *Neuroscience Research*, 55(2):146–153, 2006.
- [70] Y. Koike and M. Kawato. Estimation of dynamic joint torques and trajectory formation from surface electromyography signals using a neural network model. *Biological Cybernetics*, 73(4):291–300, 1995.

- [71] A. Kostov, B. J. Andrews, D. B. Popovic, R. B. Stein, and W. W. Armstrong. Machine learning in control of functional electrical stimulation systems for locomotion. *Biomedical Engineering, IEEE Transactions on*, 42(6):541–551, 1995.
- [72] P. N. Kugler, J. A. S. Kelso, and M. T. Turvey. *The development of movement control and co-ordination*. John Willet & Sons. Ltd., 1982.
- [73] M. Lampton. Damping-undamping strategies for the levenberg-marquardt nonlinear least-squares method. *Computers in Physics*, 11(1):110–115, 1997.
- [74] M. F. Land and M. Hayhoe. In what ways do eye movements contribute to everyday activities? *Vision Research*, 41(25-26):3559–3565, 2001.
- [75] M. L. Latash. *Neurophysiological basis of movement*. Human Kinetics, 2nd edition, 2008.
- [76] M. L. Latash, M. T. Turvey, and N. A. Bernstein. The bernstein problem: How does the nervous system make its choices? In *Dexterity and its development*, pages 277–303. Lawrence Erlbaum Associates, 1996.
- [77] P. A. Lathem, T. L. Gregorio, and S. L. Garber. High-level quadriplegia: An occupational therapy challenge. *Am J Occup Ther*, 39(11):705–714, 1985.
- [78] D. D. Lee and H. S. Seung. Algorithms for non-negative matrix factorization. *Advances in neural information processing systems*, 13:556–562, 2001.
- [79] W. A. Lee. Neuromotor synergies as a basis for coordinated intentional action. *J Mot Behav*, 16(2):135–170, 1984.
- [80] M. I. A. Lourakis. A brief description of the levenberg-marquardt algorithm implemented, 2005.
- [81] C. L. Mackenzie and T. Iberall. *The grasping hand*. Elsevier, 1994.
- [82] S. Mangold, T. Keller, A. Curt, and V. Dietz. Transcutaneous functional electrical stimulation for grasping in subjects with cervical spinal cord injury. *Spinal Cord*, 43(1):1–13, 2005.
- [83] E. N. Marieb and K. Hoehn. *Human anatomy & physiology*. Pearson Education, seventh edition, 2007.
- [84] T. H. Martin, B. D. Howard, and B. Mark. *Neural network design*. PWS Publishing Co., 1996.
- [85] J. A. Mather and J. R. Lackner. Multiple sensory and motor cues enhance the accuracy of pursuit eye movements. *Aviat Space Environ Med*, 51(9 Pt 1):856–859, 1980.
- [86] J. W. McDonald, D. Becker, C. L. Sadowsky, Sr. Jane, J. A., T. E. Conturo, and L. M. Schultz. Late recovery following spinal cord injury. case report and review of the literature. *J Neurosurg*, 97(2 Suppl):252–265, 2002.
- [87] W. D. Memberg, P. E. Crago, and M. W. Keith. Restoration of elbow extension via functional electrical stimulation in individuals with tetraplegia. *J Rehabil Res Dev*, 40(6):477–486, 2003.
- [88] R. Merletti and D. Farina. Surface emg processing: Introduction to the special issue. *Biomedical Signal Processing and Control*, 3(2):115–117, 2008.
- [89] R. Merletti and P. A. Parker. *Electromyography - physiology, engineering, and noninvasive applications*. Ieee press series in biomedical engineering. John Wiley and Sons, Inc., Hoboken, New Jersey, 2004.
- [90] G. J. Merli, S. Crabbe, R. G. Paluzzi, and D. Fritz. Etiology, incidence, and prevention of deep vein thrombosis in acute spinal cord injury. *Arch Phys Med Rehabil*, 74(11):1199–1205, 1993.
- [91] M. Minsky and S. Papert. *Perceptrons*. MIT Press, Cambridge, 1969.

- [92] T. Mohr, J. Podenphant, F. Biering-Sorensen, H. Galbo, G. Thamsborg, and M. Kjaer. Increased bone mineral density after prolonged electrically induced cycle training of paralyzed limbs in spinal cord injured man. *Calcif Tissue Int*, 61(1):22–25, 1997.
- [93] D. W. Moran and A. B. Schwartz. One motor cortex, two different views. *Nature Neuroscience*, 3:963, 2000.
- [94] S. Morita, T. Kondo, and K. Ito. Estimation of forearm movement from emg signal and application to prosthetic hand control. In *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on*, volume 4, pages 3692–3697 vol.3694, 2001.
- [95] S. Muceli, A. T. Boye, A. D’avella, and D. Farina. Multi-joint reaching tasks described by a single synergy matrix and implications for functional electrical stimulation. *In press*, 2009.
- [96] M. Munih and M. Ichie. Current status and future prospects for upper and lower extremity motor system neuroprostheses. *Neuromodulation*, 4(4):176–186, 2001.
- [97] B. Nan, F. Osamu, and T. Toshio. Emg-based motion discrimination using a novel recurrent neural network. *J. Intell. Inf. Syst.*, 21(2):113–126, 2003.
- [98] R. H. Nathan. Control strategies in fns systems for the upper extremities. *Crit Rev Biomed Eng*, 21(6):485–568, 1993.
- [99] R. H. Nathan and A. Ohry. Upper limb functions regained in quadriplegia: A hybrid computerized neuromuscular stimulation system. *Arch Phys Med Rehabil*, 71(6):415–421, 1990.
- [100] National Institute of Neurological Disorders and Stroke. Cerebral palsy information page: National institute of neurological disorders and stroke (ninds), September 2008 2009.
- [101] National Spinal Cord Injury Statistical Center. Understanding sci and functional goals, 2000.
- [102] National Spinal Cord Injury Statistical Center. Facts and figures at a glance, 2008.
- [103] S. F. Neggers and H. Bekkering. Integration of visual and somatosensory target information in goal-directed eye and arm movements. *Exp Brain Res*, 125(1):97–107, 1999.
- [104] P. Newbold, W. L. Carlson, and B. Thorne. *Statistics for business and economics*. Prentice Hall, sixth edition, 2006.
- [105] D. Nguyen and B. Widrow. Improving the learning speed of 2-layer neural networks by choosing initial values of the adaptive weights. In *Joint Conference on Neural Networks*, volume 3, pages 21–26, 1990.
- [106] L. Ning, F. Hua-Quan, and P. E. Crago. Neural network generation of muscle stimulation patterns for control of arm movements. *Rehabilitation Engineering, IEEE Transactions on*, 2(4):213–224, 1994.
- [107] D. Nishikawa, Y. Wenwei, H. Yokoi, and Y. Kakazu. Emg prosthetic hand controller using real-time learning method. In *Systems, Man, and Cybernetics, 1999. IEEE SMC ’99 Conference Proceedings. 1999 IEEE International Conference on*, volume 1, pages 153–158 vol.151, 1999.
- [108] S. Overduin, A. D’avella, and E. Bizzi. Embedded synergy combination in primate reaching. In *34th Annual Meeting of the Society for Neuroscience*, 2004.
- [109] M. G. Pandy. Computer modeling and simulation of human movement. *Biomedical engineering*, 3:245–273, 2001.
- [110] P. Parker, K. Englehart, and B. Hudgins. Myoelectric signal processing for control of powered limb prostheses. *Journal of Electromyography and Kinesiology*, 16(6):541–548, 2006.

- [111] A. Pavelka and A. Prochazka. Algorithms for initialization of neural network weights. In *Konference MATLAB*, pages 1–7. Institute of Chemical Technology, Department of Computing and Control Engineering, 2004.
- [112] V. I. Pavlovic. *Dynamic bayesian networks for information fusion with applications to human computer interfaces*. PhD thesis, University of Illinois, 1999.
- [113] P. H. Peckham, M. W. Keith, K. L. Kilgore, J. H. Grill, K. S. Wuolle, G. B. Thrope, P. Gorman, J. Hobby, M. J. Mulcahey, S. Carroll, V. R. Hentz, and A. Wiegner. Efficacy of an implanted neuroprosthesis for restoring hand grasp in tetraplegia: A multicenter study. *Arch Phys Med Rehabil*, 82(10):1380–1388, 2001.
- [114] P. H. Peckham and J. S. Knutson. Functional electrical stimulation for neuromuscular applications. *Annu. Rev. Biomed. Eng.*, 7:327–360, 2005.
- [115] D. Peleg, E. Braiman, E. Yom-Tov, and G. F. Inbar. Classification of finger activation for use in a robotic prosthesis arm. *IEEE Trans Neural Syst Rehabil Eng*, 10(4):290–293, 2002.
- [116] J. B. Pelz. *Visual representations in a natural visuo-motor task*. PhD thesis, University of Rochester, 1995.
- [117] E. J. Perreault, P. E. Crago, and R. F. Kirsch. Postural arm control following cervical spinal cord injury. *IEEE Trans Neural Syst Rehabil Eng*, 9(4):369–377, 2001.
- [118] A. B. A. C. H. N. Peter Boord. Brain-computer interface-fes integration: Towards a hands-free neuroprosthesis command system. *Neuromodulation*, 7(4):267–276, 2004.
- [119] E. A. Pohlmeier, S. A. Solla, E. J. Perreault, and L. E. Miller. Prediction of upper limb muscle activity from motor cortical discharge during reaching. *J. Neural Eng.*, 4:369–379, 2007.
- [120] D. Popovic and M. Popovic. Upper-extremity fes systems - control methods, August 23-27 1999.
- [121] D. Popovic, A. Stojanovic, A. Pjanovic, S. Radosavljevic, M. Popovic, S. Jovic, and D. Vulovic. Clinical evaluation of the bionic glove. *Archives of Physical Medicine and Rehabilitation*, 80(3):299–304, 1999.
- [122] D. B. Popovic and T. Sinkjaer. *Control of movement for the physically disabled*. Center for Sensory-Motor Interaction, second edition, 2003.
- [123] C. Prablanc, J. F. Echallier, E. Komilis, and M. Jeannerod. Optimal response of eye and hand motor systems in pointing at a visual target. *Biological Cybernetics*, 35(2):113–124, 1979.
- [124] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical recipes in c: The art of scientific computing*. Cambridge University Press, New York, 1993.
- [125] A. Prochazka, M. Gauthier, M. Wieler, and Z. Kenwell. The bionic glove: An electrical stimulator garment that provides controlled grasp and hand opening in quadriplegia. *Archives of Physical Medicine and Rehabilitation*, 78(6):608–614, 1997.
- [126] K. T. Ragnarsson. Physiologic effects of functional electrical stimulation-induced exercises in spinal cord-injured individuals. *Clin Orthop Relat Res*, (233):53–63, 1988.
- [127] K. T. Ragnarsson. Functional electrical stimulation after spinal cord injury: Current use, therapeutic effects and future directions. *Spinal Cord*, 46(4):255–274, 2008.
- [128] M. Rako, B. Freudenschus, W. Girsch, C. Hofer, J. Kaus, T. Meiners, T. Paternostro, and W. Mayr. Electromyogram-controlled functional electrical stimulation for treatment of the paralyzed upper extremity. *Artificial Organs*, 23(5):466–469, 1999.
- [129] A. Ranganathan. The levenberg-marquardt algorithm, 2004.

- [130] M. Reischl, L. Groll, and R. Mikut. Optimized classification of multiclass problems applied to emg-control of hand prostheses. In *Neural Networks, 2004. Proceedings. 2004 IEEE International Joint Conference on*, volume 3, pages 1941–1946 vol.1943, 2004.
- [131] M. M. Rodgers, R. M. Glaser, S. F. Figoni, S. P. Hooker, B. N. Ezenwa, S. R. Collins, T. Mathews, A. G. Suryaprasad, and S. C. Gupta. Musculoskeletal responses of spinal cord injured individuals to functional neuromuscular stimulation-induced knee extension exercise training. *J Rehabil Res Dev*, 28(4):19–26, 1991.
- [132] R. Rojas. *Neural networks - a systematic introduction*. Springer, 1996.
- [133] S. Samarasinghe. *Neural networks for applied sciences and engineering: From fundamentals to complex pattern recognition*. CRC Press, 2007.
- [134] J. A. Saunders and D. C. Knill. Humans use continuous visual feedback from the hand to control fast reaching movements. *Experimental Brain Research*, 152(3):341–352, 2003.
- [135] G. Schalk, J. Kubanek, K. J. Miller, N. R. Anderson, E. C. Leuthardt, J. G. Ojemann, D. Limbrick, D. Moran, L. A. Gerhardt, and J. R. Wolpaw. Decoding two-dimensional movement trajectories using electrocorticographic signals in humans. *Journal of Neural Engineering*, (4):264–275, 2007.
- [136] T. V. Schroeder, S. Schulze, J. Hilsted, and J. Aldershvile. Column traumaer. In *Basisbog i medicin og kirurgi*. Munksgaard Danmark, 2003.
- [137] T. R. D. Scott and M. Haugland. Command and control interfaces for advanced neuroprosthetic applications. *Neuromodulation*, 4(4):165–175, 2001.
- [138] A. M. Scremin, L. Kurta, A. Gentili, B. Wiseman, K. Perell, C. Kunkel, and O. U. Scremin. Increasing muscle mass in spinal cord injured persons with a functional electrical stimulation exercise program. *Arch Phys Med Rehabil*, 80(12):1531–1536, 1999.
- [139] H. M. Seifert and A. J. Fuglevand. Restoration of movement using functional electrical stimulation and bayes' theorem. *J Neurosci*, 22(21):9465–9474, 2002.
- [140] M. D. Serruya, N. G. Hatsopoulos, L. Paninski, M. R. Fellows, and J. P. Donoghue. Brain-machine interface: Instant neural control of a movement signal. *Nature*, 416(6877):141–142, 2002.
- [141] R. Sharma, V. I. Pavlovic, and T. S. Huang. Toward multimodal human-computer interface. *Proceedings of the IEEE*, 86(5):853–869, 1998.
- [142] P. Shenoy, K. J. Miller, B. Crawford, and R. N. Rao. Online electromyographic control of a robotic prosthesis. *IEEE Trans Biomed Eng*, 55(3):1128–1135, 2008.
- [143] P. Shenoy and R. P. N. Rao. Dynamic bayesian networks for braincomputer interfaces. *Advances NIPS*, 17:1265–1272, 2005.
- [144] R. Shiavi, C. Frigo, and A. Pedotti. Electromyographic signals during gait: Criteria for envelope filtering and number of strides. *Med Biol Eng Comput*, 36(2):171–178, 1998.
- [145] M. L. Sipski and J. S. Richards. Spinal cord injury rehabilitation, state of the science. *Am J Phys Med Rehabil*, 95:310–342, 2006.
- [146] B. T. Smith, M. J. Mulcahey, and R. R. Betz. Development of an upper extremity fes system for individuals with c4 tetraplegia. *IEEE Trans Rehabil Eng*, 4(4):264–270, 1996.
- [147] G. J. Snoek, M. J. Ijzerman, F. A. C. G. I. T. Groen, T. S. Stoffers, and G. Zilvold. Use of the ness handmaster to restore handfunction in tetraplegia: clinical experiences in ten patients. *Nature Spinal Cord*, 38:244–249, 2000.

- [148] L. Srinivasan and E. N. Brown. Dynamic-goal state equations for tracking reaching movements using neural signals. In *Biomedical Robotics and Biomechanics, 2006. BioRob 2006. The First IEEE/RAS-EMBS International Conference on*, pages 758–762, 2006.
- [149] S. Srinivasan. *A rhythmic movement control scheme using artificial neural networks*. University of Saskatchewan, 1992.
- [150] N. N. R. R. Suri, D. Deodhare, and P. Nagabhushan. Parallel levenberg-marquardt-based neural network training on linux clusters - a case study, 2002.
- [151] K. Takakiy, D. Aritay, S. Yonemoto, and R. Taniguchi. Using gaze for 3-d direct manipulation interface.
- [152] J. G. Taylor. *Neural networks and their applications*. John Wiley and sons, 1996.
- [153] F. Tenore, A. Ramos, A. Fahmy, S. Acharya, R. Etienne-Cummings, and N. V. Thakor. Towards the control of individual fingers of a prosthetic hand using surface emg signals. In *Engineering in Medicine and Biology Society, 2007. EMBS 2007. 29th Annual International Conference of the IEEE*, pages 6145–6148, 2007.
- [154] L. H. Ting and J. L. Mckay. Neuromechanics of muscle synergies for posture and movement. *Current Opinion in Neurobiology*, 17(6):622–628, 2007.
- [155] Tobii. Tobii x120 eye tracker manual, 2009.
- [156] E. Todorov. Direct cortical control of muscle activation in voluntary arm movements: A model. *Nature Neuroscience*, 3:391–398, 2000.
- [157] E. Todorov. Optimality principles in sensorimotor control. *Nat Neurosci*, 7(9):907–915, 2004.
- [158] D. Torricelli, I. Bernabucci, M. Goffredo, and T. D’alessio. Eye-reach: A multimodal interface based on user intention prediction, 2003.
- [159] M. C. Tresch, P. Saltiel, and E. Bizzi. The construction of movement by the spinal cord. *Nat Neurosci*, 2(2):162–167, 1999.
- [160] P. Van Donkelaar and J. Staub. Eye-hand coordination to visual versus remembered targets. *Exp Brain Res*, 133(3):414–418, 2000.
- [161] S. Vaynman and F. Gomez-Pinilla. License to run: Exercise impacts functional plasticity in the intact and injured central nervous system by using neurotrophins. *Neurorehabil Neural Repair*, 19(4):283–295, 2005.
- [162] P. H. Veltink, N. J. M. Rijkhoff, and W. L. C. Rutten. Neural networks for reconstructing muscle activation from external sensor signals during human walking. In *Intelligent Motion Control, 1990. Proceedings of the IEEE International Workshop on*, volume 2, pages 469–473, 1990.
- [163] J. L. Vercher, G. Magenes, C. Prablanc, and G. M. Gauthier. Eye-head-hand coordination in pointing at visual targets: Spatial and temporal analysis. *Exp Brain Res*, 99(3):507–523, 1994.
- [164] N. Vitiello, U. Olcese, C. M. Oddo, J. Carpaneto, S. Micera, M. C. Carrozza, and P. Dario. A simple highly efficient non invasive emg-based hmi. In *Engineering in Medicine and Biology Society, 2006. EMBS '06. 28th Annual International Conference of the IEEE*, pages 3403–3406, 2006.
- [165] G. Wang, Z. Wang, W. Chen, and J. Zhuang. Classification of surface emg signals using optimal wavelet packet method based on davies-bouldin criterion. *Medical and Biological Engineering and Computing*, 44(10):865–872, 2006.
- [166] J. Z. Wang, R. C. Wang, F. Li, M. W. Jiang, and D. W. Jin. Emg signal classification for myoelectric teleoperating a dexterous robot hand. In *Engineering in Medicine and Biology Society, 2005. IEEE-EMBS 2005. 27th Annual International Conference of the*, pages 5931–5933, 2005.

- [167] G. Welch and G. Bishop. An introduction to the kalman filter. 2001.
- [168] R. D. Welch, S. J. Lobley, S. B. O'sullivan, and M. M. Freed. Functional independence in quadriplegia: Critical levels. *Arch Phys Med Rehabil*, 67(4):235–240, 1986.
- [169] J. Wessberg, C. R. Stambaugh, J. D. Kralik, P. D. Beck, M. Laubach, J. K. Chapin, J. Kim, S. J. Biggs, M. A. Srinivasan, and M. A. Nicolelis. Real-time prediction of hand trajectory by ensembles of cortical neurons in primates. *Nature*, 408(6810):361–365, 2000.
- [170] D. T. Westwick, E. A. Pohlmeier, S. A. Solla, L. E. Miller, and E. J. Perreault. Identification of multiple-input systems with highly coupled inputs: Application to emg prediction from multiple intracortical electrodes. *Neural Comput*, 18(2):329–355, 2006.
- [171] K. R. Wheeler, M. H. Chang, and K. H. Knuth. Gesture-based control and emg decomposition. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 36(4):503–514, 2006.
- [172] J. R. Wolpaw, N. Birbaumer, D. J. Mcfarland, G. Pfurtscheller, and T. M. Vaughan. Brain-computer interfaces for communication and control. *Clin Neurophysiol*, 113(6):767–791, 2002.
- [173] D. M. Wolpert. Computational approaches to motor control. *Trends in Cognitive Sciences*, 1(6):209–216, 1997.
- [174] W. Wu, M. J. Black, D. Mumford, Y. Gao, E. Bienenstock, M. Serruya, and J. P. Donoghue. Inferring hand motion from multi-cell recordings in motor cortex using a kalman filter, 2002.
- [175] W. Wu, Y. Gao, E. Bienenstock, J. P. Donoghue, and M. J. Black. Bayesian population decoding of motor cortical activity using a kalman filter. *Neural Comput*, 18:80–118, 2006.
- [176] W. Wu and N. G. Hatsopoulos. Real-time decoding of nonstationary neural activity in motor cortex. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 16(3):213–222, 2008.
- [177] W. Wu, A. Shaikhouni, J. P. Donoghue, and M. J. Black. Closed-loop neural control of cursor motion using a kalman filter. *Conf Proc IEEE Eng Med Biol Soc*, 6:4126–4129, 2004.
- [178] B. M. Yu, C. Kemere, G. Santhanam, A. Afshar, S. I. Ryu, T. H. Meng, M. Sahani, and K. V. Shenoy. Mixture of trajectory models for neural decoding of goal-directed movements. *J Neurophysiol*, 97(5):3763–3780, 2007.

---

# Appendix

---

**V**



---

# Literature overview

---



This chapter contains an overview of the literature, which deals with different decoding methods based on single and multiple disparate sources for non-invasively estimation of the human motor intention. This chapter constituted the point of departure for selecting the methods and signals for decoding of the users intent.

This is not a complete overview of the literature within this area, since decoding is used in a variety of fields, e.g. prediction of discrete classes of movement, continuous movement, level of muscle activity, finger movements, arm movements, joint torque etc.

English-language articles were identified in a search of Google Scholar and PubMed for the following key words

- Inference of hand movements
- Decoding arm movement EMG
- Decoding arm trajectory
- Decoding arm kinematics
- Decoding hand trajectory
- Decoding hand kinematics
- Decoding hand trajectory EMG
- Classification of movements from EMG
- Multimodal decoding kinematics
- Multimodal decoding kinematics EMG
- Decoding arm kinematics eye-tracking
- FES eye tracking
- Hand kinematics eye tracking
- Hand kinematics eye tracking FES
- Motor intention Eye tracking FES
- Decoding motor intention Eye tracking FES
- Eye-tracking hand trajectory

Articles based on sEMG were selected.

[119] have predicted the EMG signals recorded from four arm and hand muscles (Medial deltoid, biceps, triceps, and above the flexor musculature for the hand) from 32 channel intra-cortical recordings from individual neurons (from M1) in two male rhesus monkeys during button pressing (required the monkey to reach with its left arm from a hold position at its side to one of four buttons) and prehension movements (relatively unrestricted reaching movements to grasp small food rewards, prehension tasks). The reaches were made within a workspace in front of the monkey subtending 80-90° and limited by the length of the monkey's reach. The surface EMG were predicted with an  $R^2$  of 75-80% based on linear filtering.

[5] have based predictions of shoulder and elbow kinematics (angle, angular velocity, and angular acceleration) of all four joint angles (elbow flexion-extension and shoulder horizontal flexion-extension, elevation-depression, and

internal-external rotation) on the sEMG (six shoulder and elbow muscles) from anterior deltoid, middle deltoid, posterior deltoid, biceps, triceps, and the clavicular portion of the pectoralis from six able-bodied, and in the C5 paralyzed the paralyzed triceps and clavicular pectoralis were replaced by the upper trapezius and middle trapezius muscles, with an RMS error of less than  $20^\circ$  during drawing, reaching and serial single joint movements. The decoding was based on a developed time-delayed artificial neural network (TDANN).

[69] have reconstructed EMG signals from six muscles corresponding to four muscles performing flexion/extension, abduction/adduction of the shoulder joint (deltoid-clavicular part, pectoralis major, trapezius, latissimus dorsi) and from two muscles responsible for flexion/extension of the elbow joint (biceps long head and triceps long head) from neuron activity recorded from a stainless steel recording chamber (42 points of microelectrode entry resulting in applying recordings from 18 neurons) M1 by the use of linear summation of the neuron activities during button pressing (reach its arm to correct target buttons on a front panel. The animal always performed the task with its left arm. From the reconstructed EMG they reconstructed the joint angles based on an three layer ANN model (The units in the first and third layer were linear functions, and the second layer units were nonlinear sigmoid functions) trained with trained with a back-propagation algorithm. The correlation coefficients for the elbow (TRL and BIL) and shoulder (PMJ, TPZ, DLC and LTD) were about 0.934 and 0.986, respectively. The reconstructed trajectories of the hand and elbow position showed a correlation with the actual arm movement on average about 84% (correlation coefficient).

[3] have estimated the levels of muscle activity (EMG) during a wide range of free movements based on kinematic information of the upper limb (EMG). The method is based on conditional distributions based on hand kinematics from free arm movement and associated sEMG from 12 muscles used for the control of arm movement (serratus anterior, anterior deltoid, posterior deltoid, pectoralis major, latissimus dorsi, teres major, biceps brachii, brachialis, brachioradialis, triceps brachii, extensor carpi radialis longus, and flexor carpi radialis) from one subject. From these conditional probability distributions the patterns of muscle activity was estimated during eight different movement tasks in five subjects. On average approx. 40% of the variance in the actual sEMG signals could be accounted for in the predicted EMG signals.

[139] predicted patterns of muscle activity (EMG) needed to produce various types of desired finger movements (repeated tapping motions similar to key presses, pushing movements involving simultaneous extension and flexion of finger joints). sEMG was recorded from from main muscles that control flexion and extension of the third digit, i.e. the digit 3 compartments of the flexor digitorum profundus, flexor digitorum superficialis, and extensor digitorum for training. Profiles of the predicted activity were used to drive frequency modulated muscle stimulators to evoke multi-joint finger movements. A probabilistic approach was taken based on Bayes' theorem. Movements generated by electrical stimulation and desired movements yielded RMS errors between 18-26%.

[63] have extracted four different classes of patterns from sEMG signals based on a discrete Hopfield network used to extract features (the first time-series parameter and the signal power) from the EMG and a multilayer perceptron network was used to classify the feature set. The movements were elbow extension, elbow flexion, wrist pronation, and wrist supination. A two-layer perceptron neural network was capable of classifying all the sets of features correctly.

[94] estimated the joint torque from sEMG based on an ANN (learning system based on feedback error learning schema) and used this as an input to a dynamic hand model to reconstruct the joint angles based on four channels of sEMG recorded from the forearm (radial side, dorsal side, ulnar side, and palmar side). The joint angles and the sEMG signal was applied in a prosthetic hand to enable grasping/opening, wrist flexion/extension, pronation/supination. The accuracy is not quantified but from the position figure comparing the estimated position [rad] to the real measure position it is seen that the estimate follows the original movement.

[35] have developed a EMG-controlled omnidirectional device that is based on a Recurrent Log-Linearized Gaussian Mixture Network using EMG signals. The neural net is based on a hidden Markov model. The idea is to express infinitely many movement directions of a pointer as combinations of probabilities of movement in preset reference directions. They apply an impedance model (inertia and viscosity is included) to make the pointer operate in accordance with the physical laws that humans experience on a daily basis. sEMG was recorded from 7 channels (6 channels from the right forearm at extensor carpi radialis muscle, extensor carpi radialis muscle, extensor carpi ulnaris muscle, extensor carpi ulnaris muscle, flexor digitorum superficialis muscle, and proper extensor indicis muscle and 1 channel from the left arm at (extensor carpi radialis muscle. The neural net was trained by associating the movement direction of the pointer with the bending direction of the right wrist.

The error (in rad.) between the movement direction of the pointer estimated from the EMG signals measured and the

direction instructed to the operator were calculated as the evaluation index (the effect of the length of the time-series during training and validation was evaluated as well). The developed method showed the best accuracy of the estimated direction (average error as low as 0.072 rad.).

[52] discriminated between elbow flexion and extension based on one channel of sEMG signal and the application of multilayer perceptron neural network (two-layer Time-Delay neural network for predictor modules in network). The identification rate for flexion and extension signals varied from 82%-100%. It was concluded that the modular neural network could identify limb functions when presented with a single channel surface EMG signal. Moreover the identification rate exceeded 96% when the length of signal was at least 150 sampling points.

[53] developed a respiration-controlled multichannel FES system for controlling upper extremities in high cervical cord injury, C4 tetraplegics. Based on inspiration and expiration using two respiratory sensors predefined movements were elicited and the pattern of muscle activity associated with a particular movement were obtained from normal subjects to identify patterns of muscle activity associated with a particular movement. The multichannel stimulation patterns for control of the multi-joint hand-wrist-elbow system were created by trapezoidal approximation method of the EMG activities. It was shown that a C4 patient could achieve versatile control of the upper extremities.

[97] presented a pattern discrimination method based on six channels of six channels sEMG recorded from electrodes attached at the forearm and upper arm (Flexor Carpi Radialis (FCR), Extensor Carpi Ulnaris (ECU), Flexor Carpi Ulnaris (FCU), Biceps Brachii (BB), Triceps Brachii (TB)) in five subjects (three healthy and two amputees). They used a recurrent Neural network based on a hidden Markov model. Six different motions of the wrist (flexion, extension, supination, pronation) and hand grasping and opening) were discriminated continuously. A discrimination rate of up to 99.06% were shown.

[70] reconstructed human arm movement from sEMG (deltoid-clavicular part, deltoid-acromial part, deltoid-scapular part, pectoralis major, and teres major, for double-joint muscles: biceps-long head and triceps-long head, for flexion/extension of elbow joint: brachialis, triceps-medial head, and triceps-lateral head) using a forward dynamics model acquired by an artificial neural network within a modular architecture. Specifically, the dynamic joint torques at the elbow and shoulder were estimated for movements in the horizontal plane. The aim was to construct a complete forward dynamics model of the human arm, which affords accurate estimation of movement trajectories from the input of physiological signals (sEMG). The method was based on artificial neural network that learned the non-linear functions relating physiological recordings of the EMG signals to the simultaneous measurement of two-joint planar movement trajectories. The EMG signal was input and end-point trajectories was the output. They treated the EMG signals as a record of the motor commands to the muscles, since the motor neuron activity could not be directly measured. Thus, the EMG activity was considered a reasonable reflection of the firing rate of a motor neuron. The trajectory was computed at each time step by predicting the dynamic torque by the neural network model from the position and velocity values at the current time step and the past 500 ms of EMG data. The predicted torque was used as control input to the forward dynamics equation. Coefficient of determination for position data was 0.948.

[29] investigated the mapping between arm kinematic variables and spatiotemporal patterns of EMG activity recorded from pectoralis major, anterior deltoid, posterior deltoid, biceps brachii and triceps brachii while the subject performed a three-dimensional movement. More specifically they looked at 'reach-to-grasp-with-return', at the wrist trajectories in the horizontal plane. They showed that the Back Propagation Through Time-ANN was able to learn the non-linear, complex mapping between EMG signals from selected muscles at the shoulder and elbow joints and the corresponding trajectory patterns. The BPTT-ANN was able to reproduce movement trajectories well based on the corresponding EMG patterns. Mean RMS was  $0.092 \pm 0.016$ . This meant that EMG signals from the shoulder and arm contain a significant, maybe redundant, amount of information about wrist movement kinematics.

[18] have proposed an approach based on dynamic recurrent neural networks to identify the relationship between sEMG recorded from seven muscles (posterior deltoid external and internal, anterior deltoid, median deltoid, pectoralis major superior and inferior, and latissimus dorsi) and the arm kinematics ( $x$  and  $z$  coordinates) during the drawing of the figure eight with the right arm extended. The method was based on artificial dynamic recurrent neural networks to map the raw sEMG data of the figure eight movement onto the corresponding kinematics of the arm. It is concluded that the method performance is good and the simulated curve reproduced all the particularities of the human complex movements, whatever the initial direction of the movement. In addition the robustness of the method was tested by applying perturbations in the range of 20% to the EMG signals. This resulted in a difference in the position on less than 6%. The results showed that DRN's are successful in identifying the complex mapping between full-wave

rectified EMG signals and upper-limb trajectory.

[153] presents preliminary work where 12 individuated flexion and extension movements were decoded with an accuracy higher than 98% based on sEMG signals (collected from 32 bipolar electrodes placed mainly on an able-bodied individual's forearm performing) from natural hand and finger movements. The method was based on ANN (Non-linear decoding filters were designed using multilayer, feed-forward Artificial Neural Networks). Feature extraction was based on four time-domain EMG features; Mean of the absolute value, Willison Amplitude, Variance, and Waveform length.

[65] has developed an approach to differentiate between eight different hand movements (hand opening and closing, pinch, thumb flexion, wrist radial flexion and extension and wrist flexion and extension) from recorded sEMG signals (extensor digitorum, extensor carpi radialis, the palmaris longus and the flexor carpi ulnaris). Feature extraction is based on three types of feature representations; time domain (mean absolute value (MAV), zero crossing (ZC), Wilson amplitude (WAMP), slope sign changes (SSC) and coefficients of autoregressive model (AR)), time-frequency (short time Fourier transform (STFT), wavelets transform (WT), and wavelet packets transform (WPT)) domain and their combination. Dimensionality reduction technique was applied to simplify the task of the classifier (class separability (CS) and principle component analysis (PCA)). Classifiers were based on ANN and fuzzy inference system. Combined features with PCA dimensionality reduction and FIS as classifier provided the best results (83% correct classification on average).

[142] presents results from a pilot study. Initially they showed offline classification accuracy of 92-98% for a eight-class classification problem from static sEMG signals recorded from the forearm muscles (brachioradialis, extensor carpi ulnaris, pronator teres, extensor communis digitorum, flexor carpi radialis, anconeus, pronator quadratus). They used simple features (rms values over windows) and continuously classify windows of data while the subject maintain a static hand gesture. In addition they used linear support vector machines (SVM) for classification.

[143] show how a dynamic Bayesian network can be used to infer probability distributions over brain- and body-states during planning and execution of actions. This approach allowed continuous tracking and prediction of internal states over time, and to generate control signals based on entire probability distributions over states. They modeled the dynamics of hidden brain- and body-states using a Dynamic Bayesian Network that was learned directly from EEG and EMG data. [143] analyzed left versus right hand movement task and presents preliminary results showing supervised learning and Bayesian inference of hidden state for a dataset containing simultaneous EEG and EMG recordings. 8 EEG channels were recorded around the motor area of cortex (C3, Cz, C4, FC1, FC2, CP1, CP2, Pz). An error of 15% was achieved by the use of the EEG to discriminate between left and right hand movement.

[17] presents an ongoing investigation of dexterous and natural control of upper extremity prostheses using the four channels of EMG during six distinct limb motions (wrist flexion, wrist extension, supination, pronation, hand open, and hand close). The sEMG was recorded with four channels placed equidistant from the wrist and elbow, with an equal spacing around the forearm. They used a hidden Markov model to process four channels of EMG, with the task of discriminating six classes of limb movement. The method allow a continuous stream of class decisions to be delivered to a prosthetic device. The method based on HMM provided greater accuracy than a multilayer perceptron neural network. The classification of continuous sEMG signals resulted in an average accuracy of 94.63% using the HMM method compared to the MLP method which had an accuracy of 93.27%, both on average.

[54] introduced and evaluated the use of Gaussian mixture models (GMMs) for multiple limb motion classification using continuous myoelectric signals (recorded over the wrist flexors and extensors equidistant from the elbow and wrist). The GMM was compared to three commonly used classifiers; a linear discriminant analysis, a linear perceptron network, and a multilayer perceptron neural network. Six limb motions (wrist flexion, wrist extension, forearm supination, forearm pronation, hand open, and hand close) were performed and the sEMG were recorded by electrodes placed on the forearm above the wrist flexors and extensors, and on each side of the forearm. The GMM showed an average (across the 12 subjects) classification error of 3.72%.

[56] presented an identification method of finger motions (middlefinger flexion/ extension, indexfinger extension/ flexion, thumb extension/ flexion) using the wavelet transform of multi-channel EMG signals. The sEMG (recorded from 16 channels from surface electrodes attached at the upper arm and fore arm, corresponding to the flexor muscles including flexor carpi ulnari and flexor digitorum superficialis, the extension muscles including extensor digitorum and extensor pollicis brevis) was analyzed based on the wavelet transform with features computed as the variance, maximum and mean absolute value of the wavelet coefficients. Classification was based on ANN to identify finger motion.

Correct classification of finger movement above 80% was achieved.

[64] introduced a new multiwavelet function for sEMG signal (2 channels, electrodes positioned at the forearm under the elbow) intended for tasks that involve hand movement recognition. Eight unique classes of hand motions (hand opening and closing, pinch, thumb flexion, wrist radial flexion and extension and wrist flexion and extension) were classified based on 2 channels of sEMG recordings from electrodes positioned at the forearm under the elbow. Local extrema and zero crossing was used as DWT features. Average classification accuracy of up to 87% was achieved by the proposed multi-wavelet function.

[165] presented an optimal wavelet packet method based on Davies-Bouldin criterion and relative energy representation of wavelet packet, which is applied to sEMG signals classification. Two channels of sEMG were recorded (above flexor carpi radialis and over the extensor carpi radialis longus) from ten normally limbed subjects. Different movements were classified (forearm pronation, forearm supination, hand close and hand open) by the used of the optimal wavelet packet for feature extraction and a neural network based on error backpropagation rule for classification. PCA was used for dimensionality reduction. A classification of 93.75% was achieved.

[164] developed a finite state algorithm used to operate a mobile robot and applied to EMG-based control. Two channels of sEMG were recorded, one from each biceps from five healthy subjects. Nearest Neighbor statistical algorithm were used to classify The four states, "stop", "forward", "left turn" and "right turn" based on the features square mean value, standard deviation and kurtosis index. An average classification error of 3.5% was achieved.

[166] presented a sEMG based classification system based on an three layer ANN based on a modified Backpropagation algorithm and AR modeling for teleoperating a dexterous robot hand. Six different finger motions (thumb extension, thumb flexion, index finger extension, index finger flexion, middle finger extension, and middle finger flexion) were classified from the sEMG. Four channels of sEMG were recorded from extensor digitorum, extensor pollicis brevis, flexor carpi ulnaris, flexor digitorum superficialis from four healthy subjects. A classification rate above 77% were achieved by all subjects.

[31] presented an ongoing investigation of dexterous and natural control of upper extremity prostheses based on sEMG. A continuous classifier was constructed and applied in a four-class problem in hand and wrist control (wrist flexion, wrist extension, radial deviation and ulnar deviation) based on recordings from four channels of sEMG from the forearm above the wrist flexors and extensors, and on each side of the forearm, roughly equidistant from the elbow and wrist from 12 normally limbed subjects. The different classes were classified from a feature set consisting of time domain statistics, i.e. the number of zero crossings, the waveform length, the number of slope sign changes, and the mean absolute value in each analysis window. Classification was based on linear discriminant analysis. Classification error around 7% was achieved.

[115] used a combination of a K-nearest neighbor classifier and a genetic algorithm for feature selection from sEMG signals recorded from two channels at the flexor muscles of the right forearm in four healthy subjects. Average classification error rate of approximately 2% was achieved.

[130] proposed a control scheme for a multifunctional EMG control of hand prostheses. A transformation matrix was used to reduce the feature dimension by linear aggregation. Two channels of sEMG were recorded from two subjects with lower limb deficiencies and two channels were recorded from two subjects with upper limb deficiency (recorded from forearm flexors in the first group and from biceps and triceps in second group) and two subjects with upper limb deficiency (recorded from biceps and triceps). Different classifiers (approximate maximum likelihood classifier, hierarchical classifier, fuzzy classifier, multilayer perceptron, ) were evaluated based on misclassification rates.

[107] proposed a real-time learning method for a EMG prosthetic hand controller for discrimination between 10 different hand motions (wrist motions: supination, pronation, flexion and extension, hand motions: 2-5th fingers flexion, 2-5th fingers extension, thumb flexion, thumb extension, 4-5th fingers flexion, and 2-3rd fingers extension). Two channels of sEMG were recorded, one from each of the forearms. They use the Gabor transform and mean absolute value to extract information from the EMG. Feed- forward neural network was used as the nonlinear function for the adaption unit (to learn the characteristics of the operators), i.e. the classifier. Two channels of sEMG were recorded from three normal subjects. An average (for all subjects for the last three trials) classification rate of 91.5% was achieved.

[32] explored the efficacy of feature sets derived from time-frequency representations based on Hudgins' time domain features, and those derived from short-time Fourier transform, the wavelet transform, and the wavelet packet transform. Four classes of myoelectric signal patterns were collected from the biceps and triceps (flexion and extension of the elbow, and pronation and supination of the forearm) from 16 healthy subjects. The sEMG was recorded from

four channels from the forearm above the wrist flexors and extensors, and on each side of the forearm. Dimensionality reduction was performed by PCA and CS. An average classification error of 6.25% was achieved.

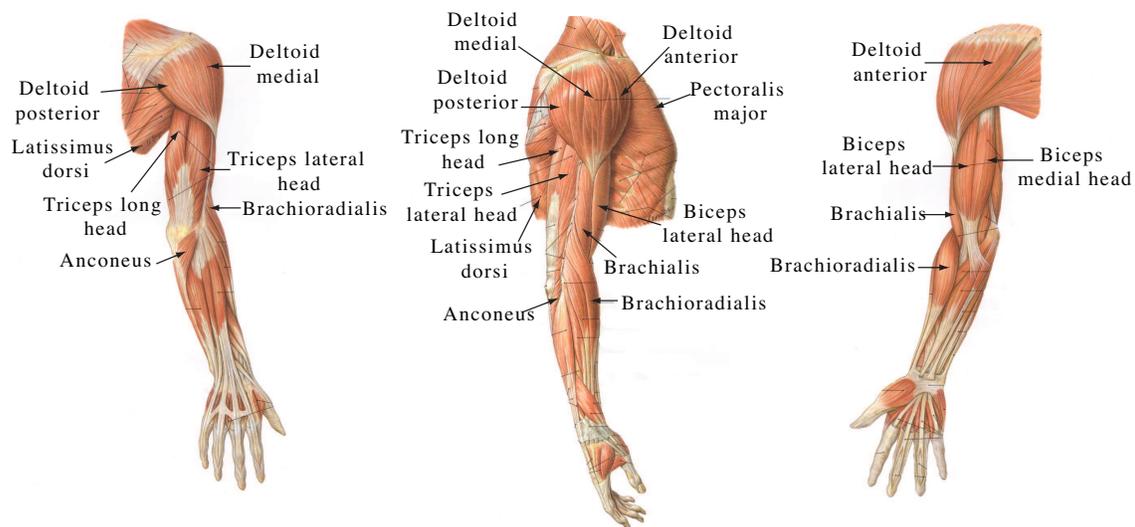
[67] performed signal processing of four channels sEMG signals (four channels placed on the forearm without following any strict pattern) using a linear multiple regression model to predict the joint angles (multi-finger angles corresponding to the different motions) to discriminate between different motions (grip, open, and chuck of a hand). It was shown that the discrimination of grip, open, and chuck motions was possible. The average tracking error between estimated and measured joint angles was within less than  $2^\circ$  for a dynamic target angle and within less than  $5^\circ$  with a dynamic target angle.

[41] trained an NN controller to output an appropriate level of triceps stimulation based on the remaining voluntary controlled upper extremity muscles, i.e. shoulder muscles and elbow flexor, in C5/C6 SCI subjects. The aim was to allow subjects to generate and control endpoint force vectors unachievable without stimulation of the triceps. Elbow extension moments were computed based on a biomechanical model (position of humerus and girdle and external load on humerus was input, muscle forces were output) for a specific subject and goal isometric endpoint force vector that the subject encountered during data collection. The elbow moments were converted to stimulation levels by experimentally measuring elbow moment as a function of stimulus level using an elbow moment transducer.

The NN was trained with the EMG signals as input and the triceps stimulation levels used to obtain the goal force vector as output. The NN consisted of one hidden layer (5 neurons with tan-sig transfer function) and one output layer. It was concluded that the synergistic controller produced larger forces compared to no stimulation for endpoint force directions predicted to require triceps stimulation. The subject tracked isometric force vectors equally well with each control method (synergistic and constant stimulation), but used 35, 20, and 36% less stimulation over the x, y, and z-axes respectively with synergistic control compared to constant stimulation.

## B.1 Muscles responsible for arm reaching

This section describes the shoulder and elbow muscles (large superficial flexor and extensor muscles of the upper limb) responsible for reaching movements in the horizontal plane. The muscles are listed in table B.1 with indication of name and function. Figure B.1 shows the relevant anatomy for these muscles.



**Figure B.1:** Flexor and extensor muscles of the right arm responsible for reaching movements. Left: posterior view, center: lateral, right: anterior view. Modified from [19]

Name	Function
Shoulder	
Deltoid medial	Abduction of shoulder
Deltoid posterior	Abduction, extension (and transverse extension), and lateral rotation at shoulder (rotary movement around the longitudinal axis of the bone away from the center of the body; turning the upper arm outward)
Deltoid anterior	Abduction, flexion (and transverse flexion), and medial rotation at shoulder (rotary movement around the longitudinal axis of the bone toward the center of the body; turning the upper arm inward)
Chest	
Pectoralis	flexion and extension at shoulder, transverse adduction and flexion, adduction and extension
Back	
Latissimus dorsi	shoulder adduction, extension, transverse extension, medial rotation

Upper arm	
Biceps brachii, long head (lateral)	Flexion at elbow and shoulder, supination
Biceps brachii, short head (medial)	Flexion at elbow and shoulder, supination
Brachialis	Flexion at elbow
Triceps brachii, long head	Extension at elbow, extension and adduction at shoulder
Triceps brachii, medial head	Extension at elbow
Triceps brachii, lateral head	Extension at elbow
Forearm	
Anconeus	extension of forearm at elbow
Brachioradialis	flexion of the forearm at elbow (when forearm is semi-pronated)

**Table B.1:** Primary muscles for reaching movements in the horizontal plane.

## B.2 Remaining movement abilities under different levels of injury

Table B.2 provides an overview of the abilities that remains under different levels of injury (included from C3-C6) .

Level of injury	Abilities
C1-C3	C3-limited movement of head and neck
C3-C4	Usually has head and neck control. Individuals at C4 level may shrug their shoulders
C5	Typically has head and neck control, can shrug shoulder and has shoulder control. Can bend his/her elbows and turn palms face up
C6	Has movement in head, neck, shoulders, arms and wrists. Can shrug shoulders, bend elbows, turn palms up and down and extend wrists

**Table B.2:** Remaining movement abilities under different levels of injury

---

# Photos of experimental setup

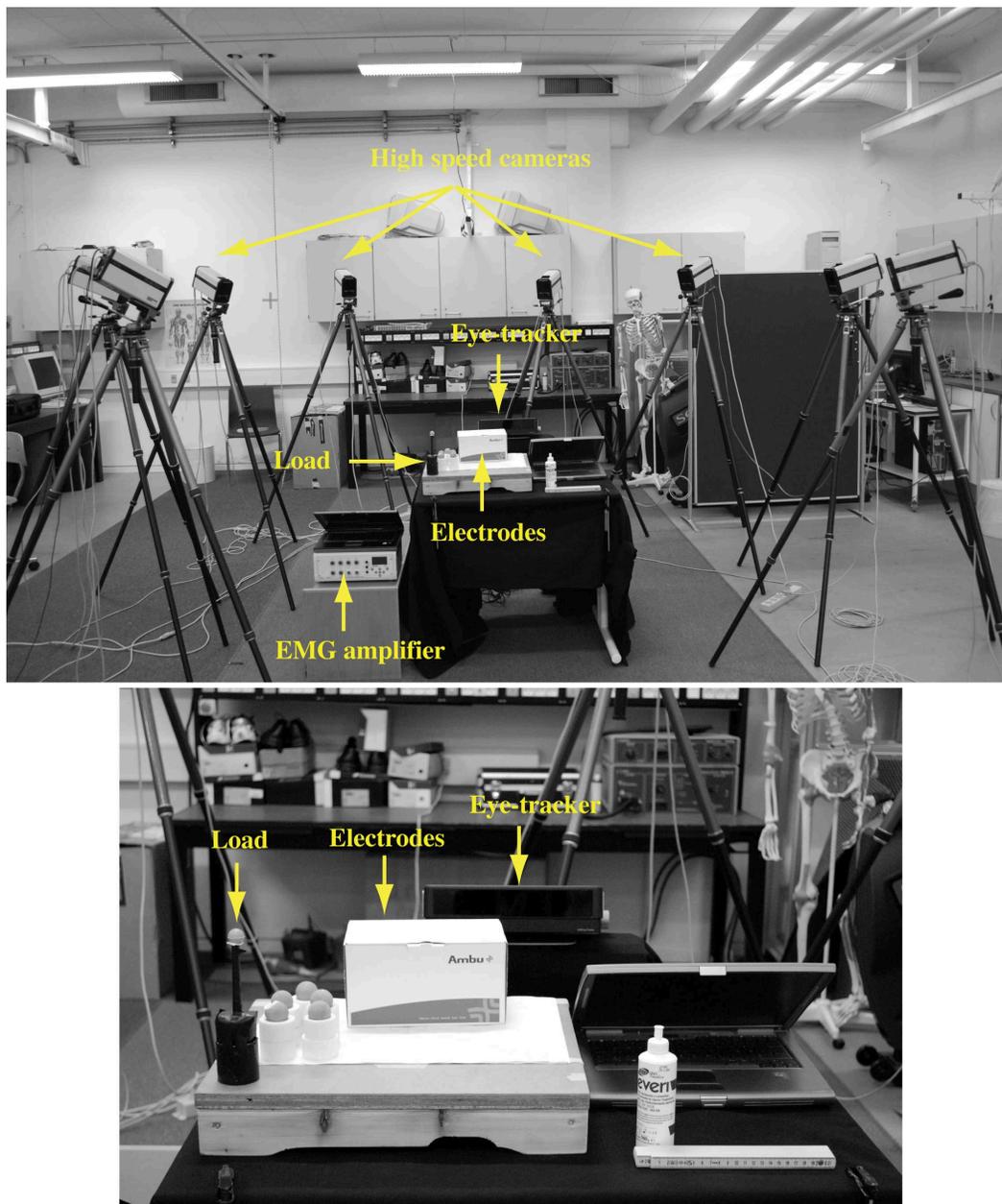
---

# C

This chapters contains pictures of the conducted experiment.

## C.1 Equipment

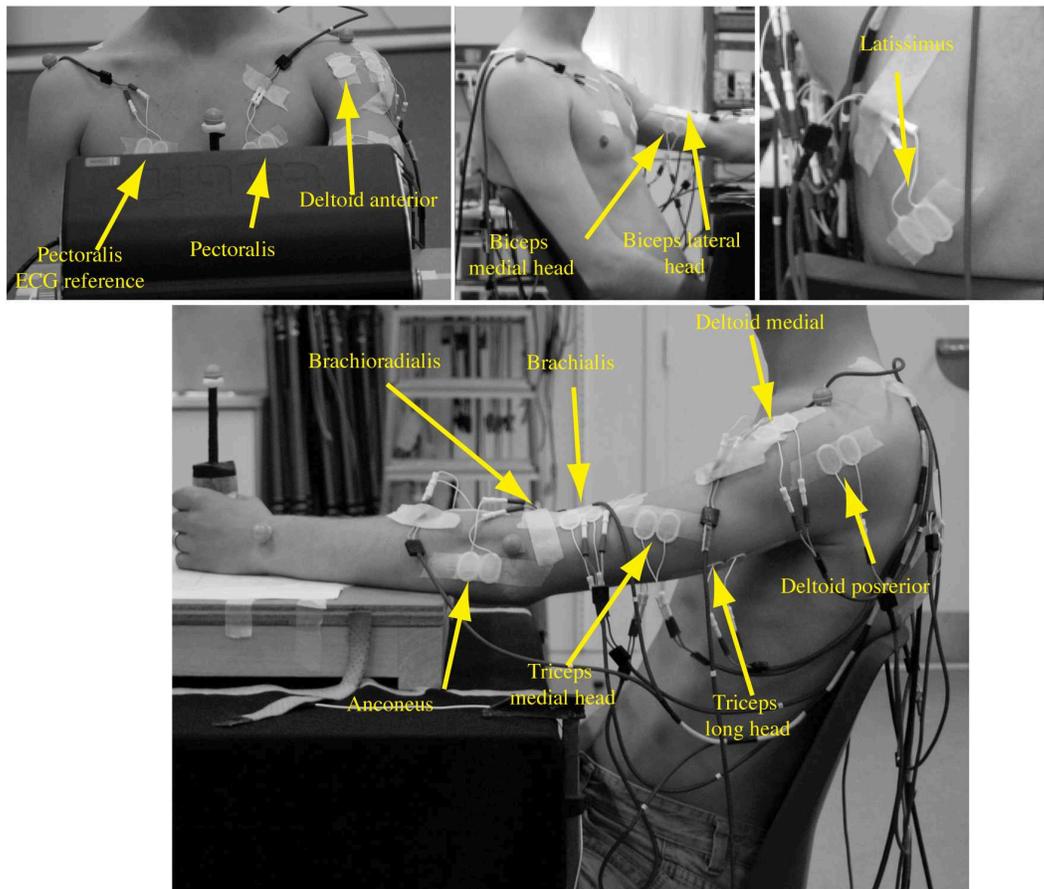
The equipment used in the experimental setup is shown on figure C.1



**Figure C.1:** Experimental equipment. Hand load, EMG amplifier, high speed cameras and eye-tracker

## C.2 Montage of sEMG electrodes

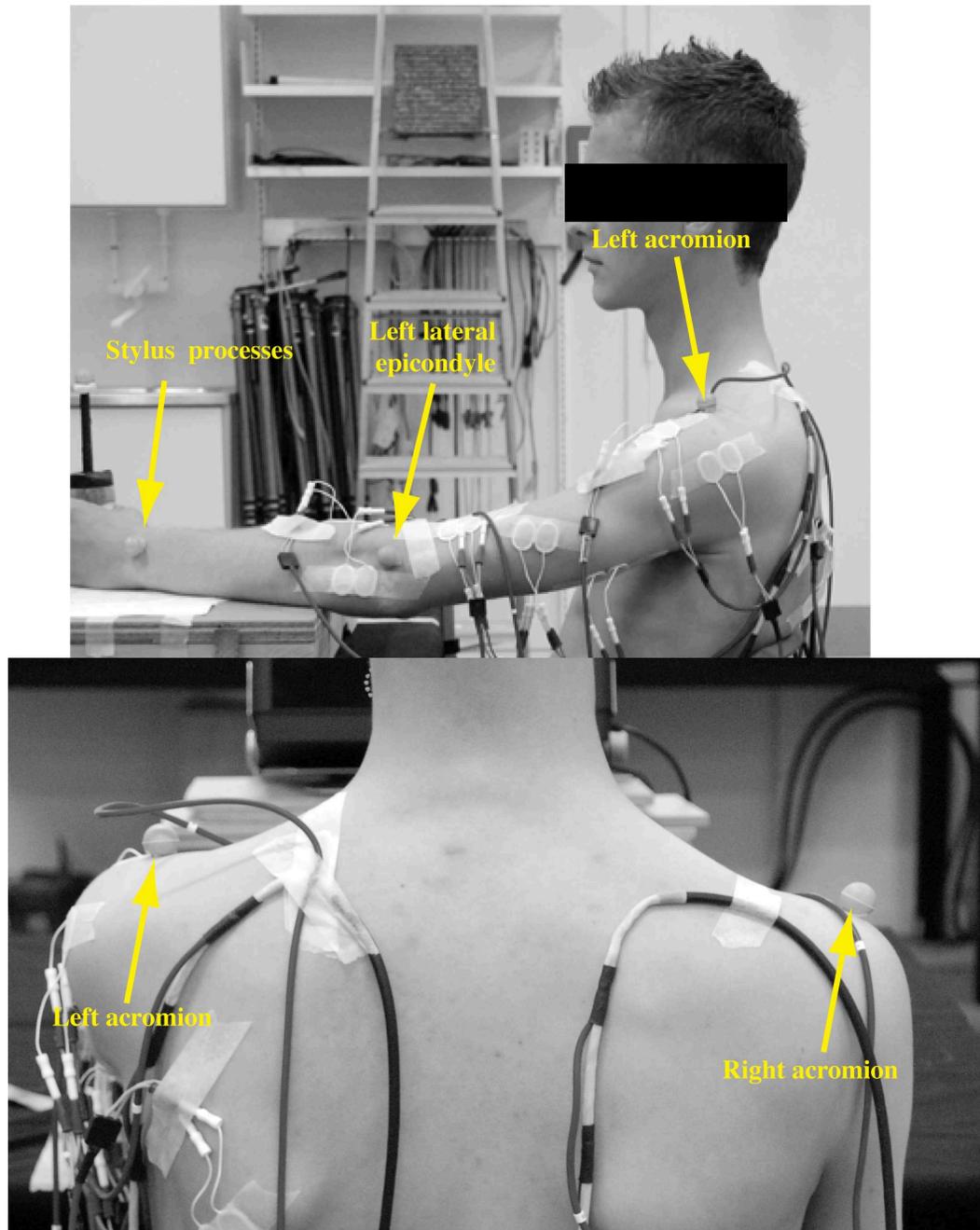
The montage of the sEMG electrodes is shown in figure C.2



**Figure C.2:** Montage of the sEMG electrodes.

## C.3 Placement of reflective markers for motion capture

The placement of the reflective ball markers is shown in figure C.3



**Figure C.3:** Placement of the reflective markers for motion capture for a left handed subject. Markers placed between the stylus processes at Radius and Ulna, at the left lateral epicondyle, and at the right and left acromion.



---

# Optimization of the neural network

---

# D

This chapter contains the optimization results of the neural network with respect to the data recorded from subject 1 as described in the protocol in section 5.2 on page 43. The optimization is based on eyetracking and muscle activity as input.

The neural network were configured with one hidden layer with a nonlinear transfer function and one output layer with a linear transfer function together with the training algorithm as described in subsection 5.2.4 on page 49. The parameters that are determined based on the following results, are the number of neurons in the hidden layer. The number of neurons were varied from 1-20 neurons with a maximum number of epochs sat to 150.

All evaluations are based on the average across the three trials recorded. Depictions of the muscle activity and kinematics are from trial one in all cases.

## D.1 Prediction of muscle activity

This sections contains the basis for the selection of the number of neurons in the hidden layer for the prediction of muscle activity.

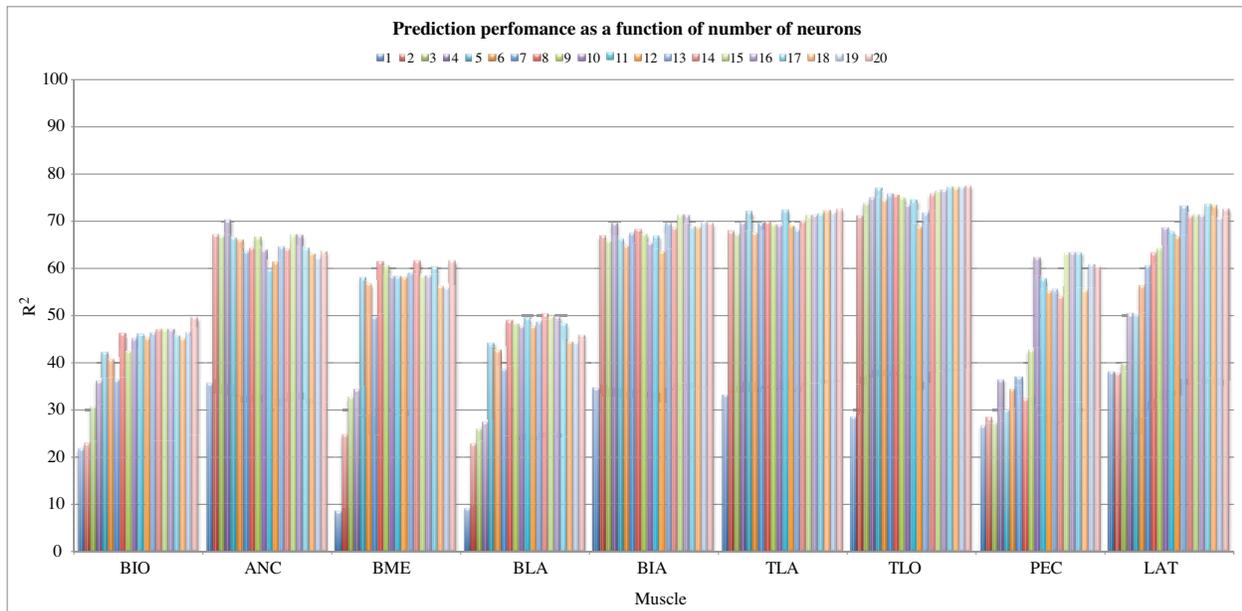
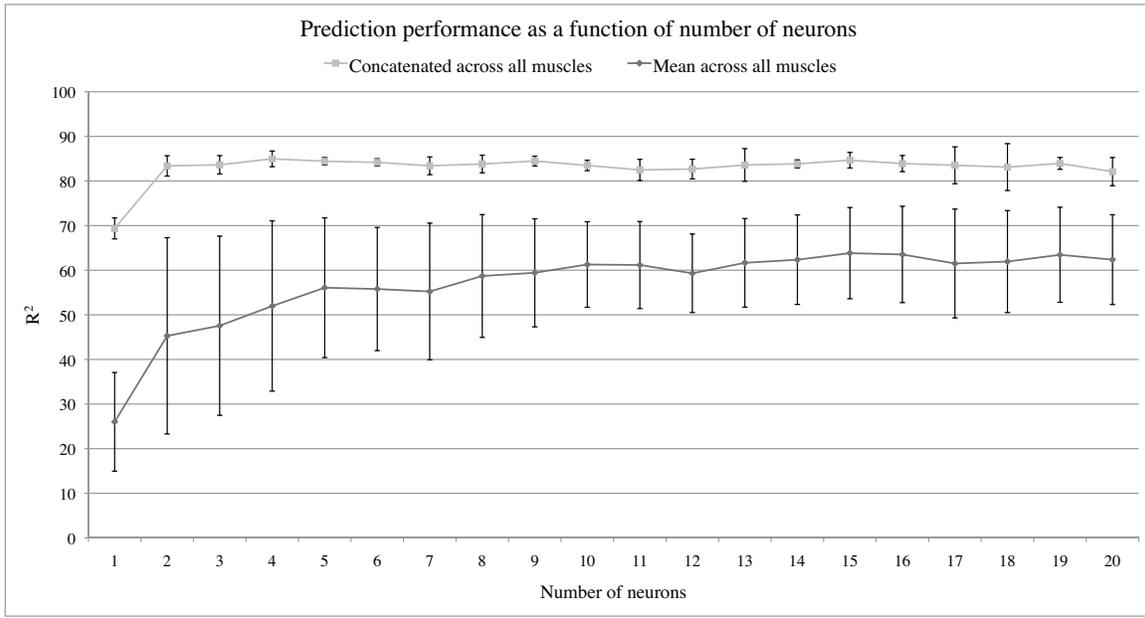
### D.1.1 Prediction performance as a function of neurons

The prediction performance as a number of neurons in the hidden layer is illustrated for the muscle activity in figure D.1

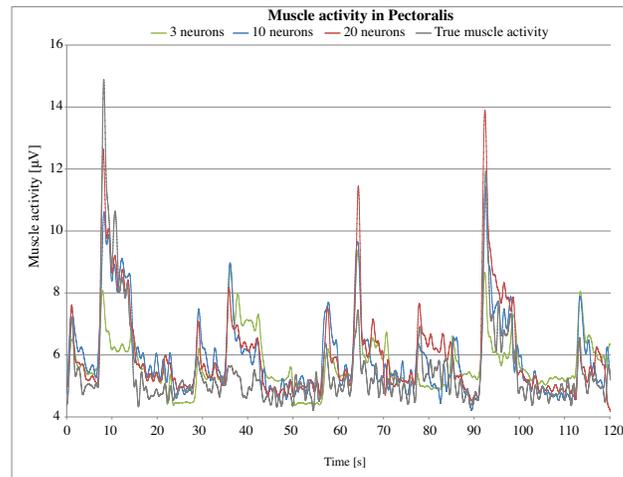
From figure D.1 (top), it was seen that the predictions of the concatenated data yielded an  $R^2$  value above 80% based on one hidden layer with at least 2 neurons. Figure D.1 (bottom) showed a clear difference between 1 hidden neuron compared to 2 and up to 20 neurons in the hidden layer.

For the prediction of the muscle activity in Brachioradialis, 1-3 neurons in the hidden layer showed a clear lower prediction compared to 4-20 neurons. For Biceps medial and lateral head, the predictions were clearly improved for 5 and more neurons. In Pectoralis the predictions were improved from 9-10 neurons in the hidden layer. 10 neurons in the hidden layer were the least number of neurons that yielded the high prediction performance for all muscles.

The predictions of the muscle activity for the Pectoralis muscle is shown on figure D.2 using 3 (green), 10 (blue), and 20 (red) neurons in the hidden layer, compared to the true (bold grey) muscle activity. This muscle was selected, since the muscle activity clearly indicated the effect of increasing the number of neurons.



**Figure D.1:** Prediction performance of the muscle activity as a function of number of neurons. Top:  $R^2$  from the predictions of the concatenated data across all muscles (dark grey) and mean across all muscles (light grey). Bottom:  $R^2$  from the predictions of the muscle activity based on the individual muscles (number of neurons appears consecutive for each muscle)



**Figure D.2:** Prediction of the muscle activity in Pectoralis using 3 (green), 10 (blue), and 20 (red) neurons in the hidden layer, compared to the true activity (bold grey).

From figure D.2 it was clearly revealed that 3 neurons performed worst compared to 10 and 20 neurons. It was seen that the prediction did not follow the characteristics of the true signal especially in the extreme peaks, and is in general smoother.

The predictions based on 10, and 20 neurons had many similarities, and both followed in general the overall trend of the true muscle activity. The difference between the two was seen in the high peaks during movement, where the configuration based on 10 neurons achieved higher values, where 20 neurons in the hidden layer tended to overestimate the activity. Overall 10 neurons in the hidden layer achieved a very accurate prediction in the peaks indicating a movement.

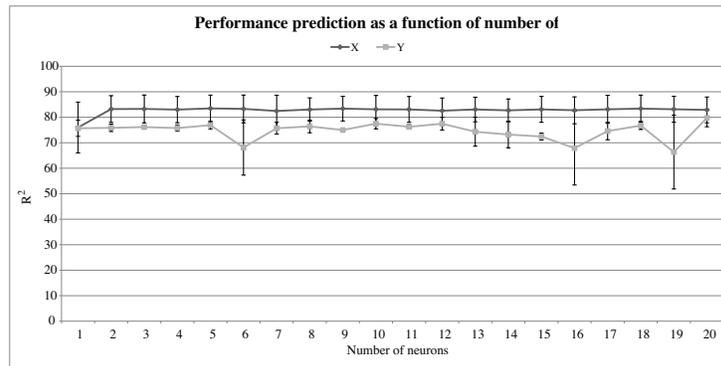
Based on the above observations 10 neurons were selected for the hidden layer.

## D.2 Prediction of kinematics

This sections contains the basis for the selection of the number of neurons in the hidden layer for the prediction of the kinematics.

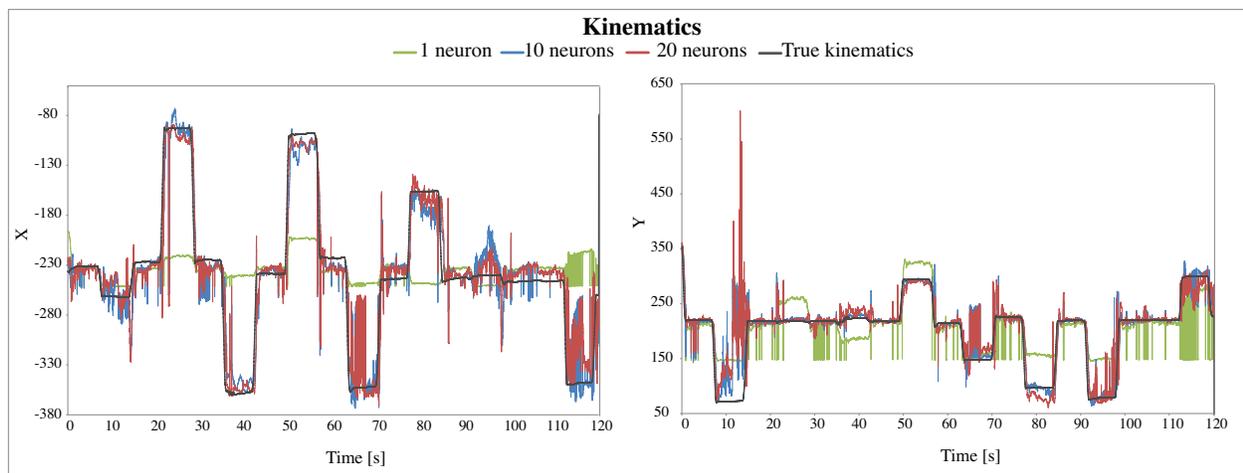
### D.2.1 Prediction performance as a function of neurons

The prediction performance as a number of neurons in the hidden layer is illustrated for the kinematics in figure D.3 From figure D.3 it was seen that the predictions were very stable in both  $x$  and  $y$ -direction using more than 2 neurons in the hidden layer, with only minor variations in the predictions for the varying number



**Figure D.3:** Prediction performance of the kinematic as a function of number of neurons in  $x$ - (dark grey) and  $y$ -direction (light grey).

Figure D.4 shows the prediction of the position in  $x$  (left) and  $y$  (right) using 1 (green), 10 (blue), and 20 (red) neurons in the hidden layer, compared to the true position (bold dark grey).



**Figure D.4:** Prediction of the kinematics using 1 (green), 10 (blue), and 20 (red) neurons in the hidden layer, compared to the true kinematics (bold grey).

The predictions in the  $x$ -direction revealed a poorer prediction based on 1 neurons in the hidden layer, where the position was flat during the entire period compared to the true position. All movements tended to be underestimated.

The predicted position using 10 and 20 neurons had similar characteristics and in some parts of the period, the trend of the true position was followed well by both configurations. Though, the predictions were noisy in some transient segments.

Similar for the predictions in the  $y$ -direction yielded 1 neuron the worst results, with a quite noisy signal throughout the entire period and with both under- and overestimation of the true position in some cases. The predictions based on 10 and 20 neurons followed the true signal quite well in large parts of the period.

Based on these observations, 10 neurons were selected for the hidden layer.

# Selection of number of synergies

# E

This chapter contains the results from which the optimal number of synergies were selected for the extraction of synergies and motor commands. The number of synergies and motor commands to extract were determined by evaluating the reconstruction performance with a varying number extracted synergies and motor commands.

The results were based on the muscle activity from three trials recorded from subject 1 as depicted in figure E.1 - E.3.

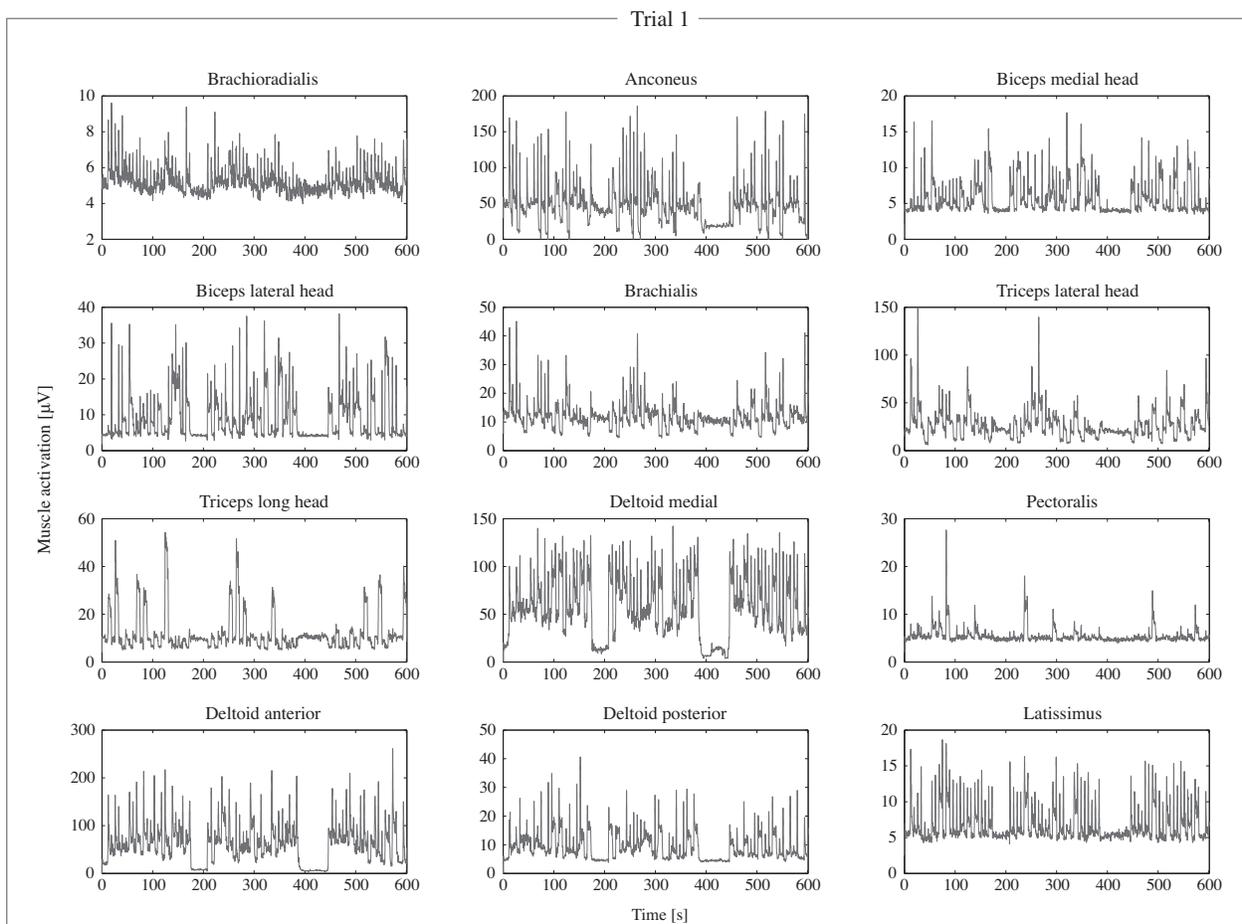
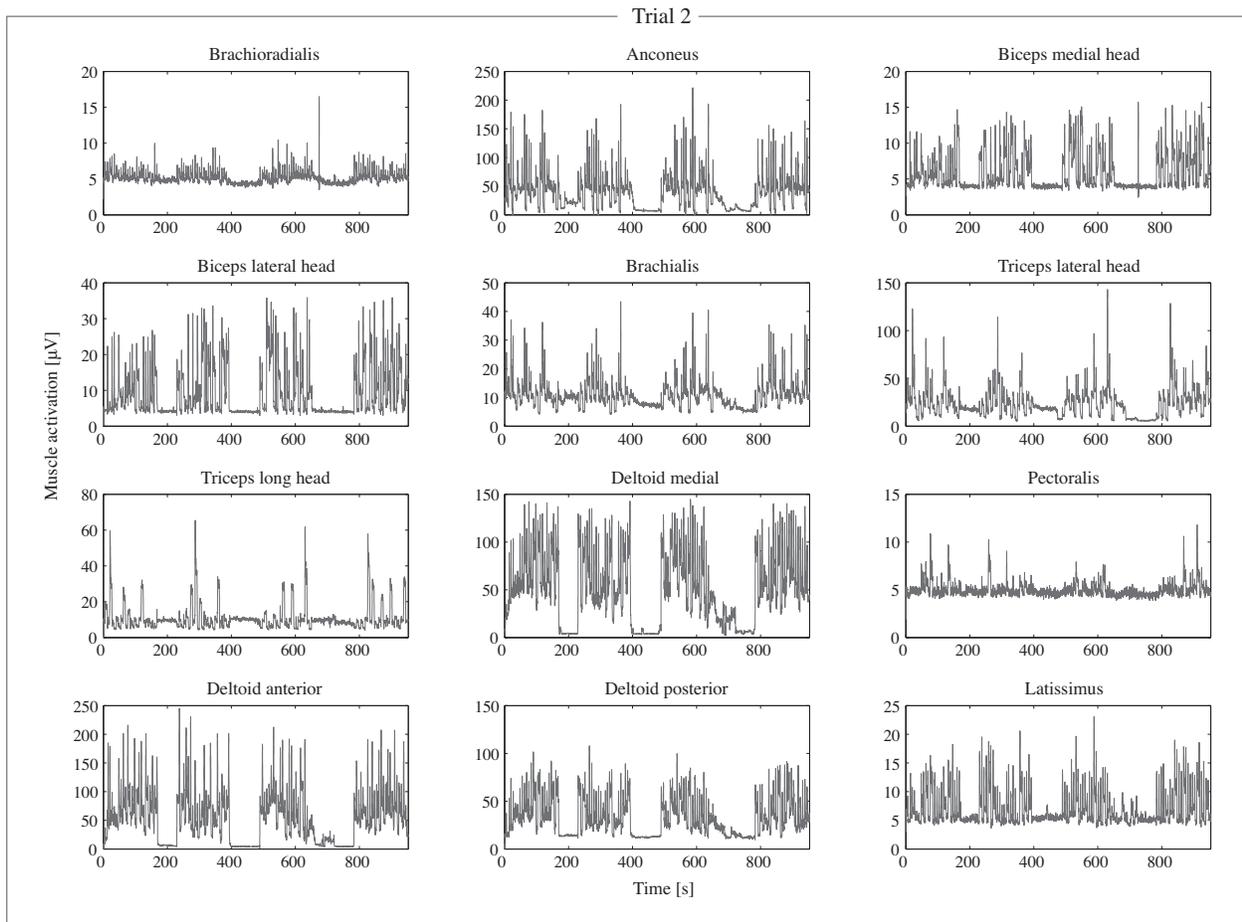


Figure E.1: Recorded muscle activity from trial 1.



**Figure E.2:** Recorded muscle activity from trial 2.

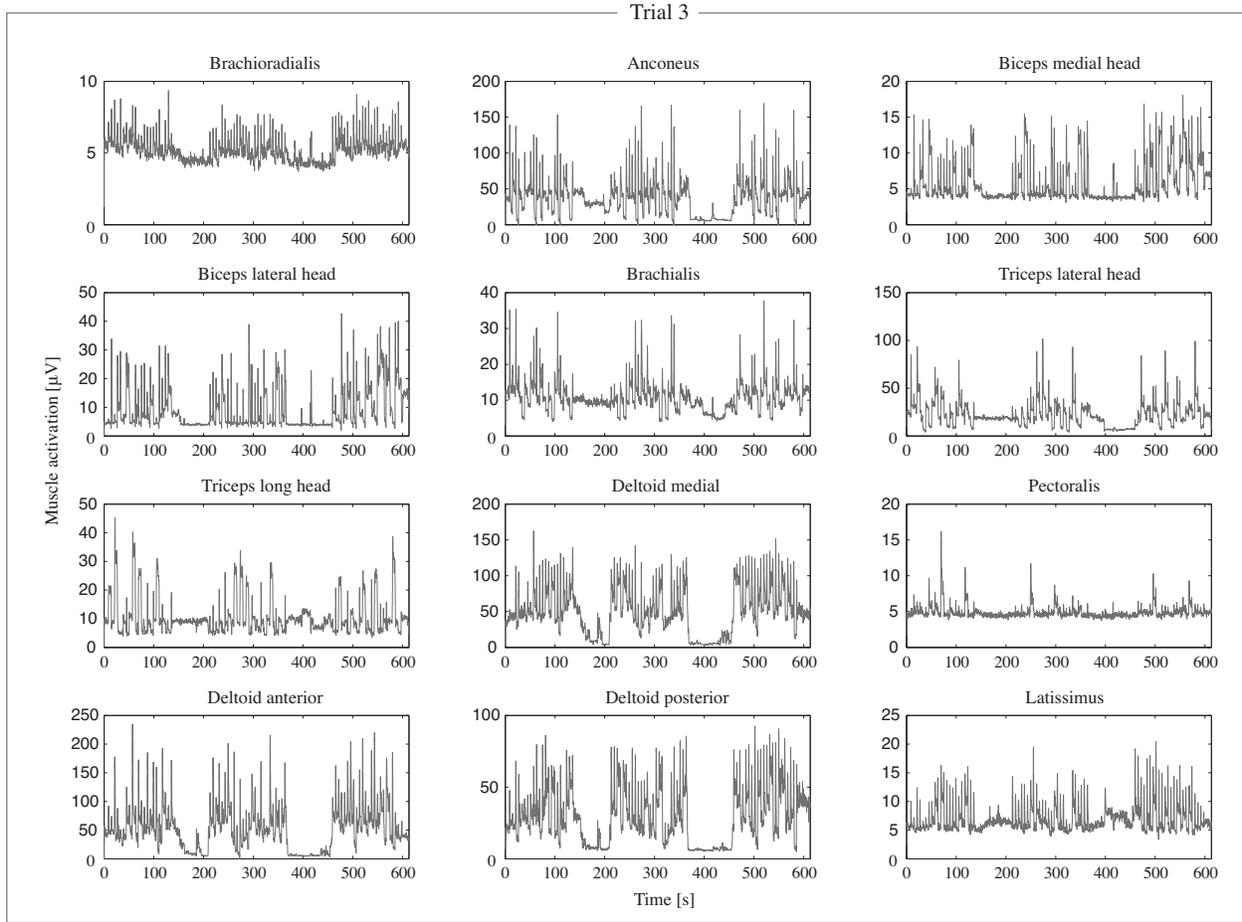


Figure E.3: Recorded muscle activity from trial 3.

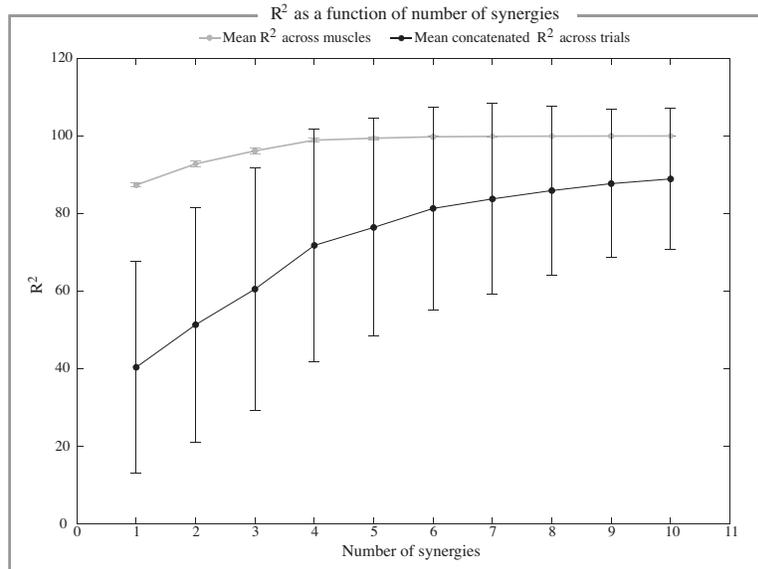
## E.1 Reconstruction performance as a function of number of synergies

The muscle activity was reconstructed from a varying number ( $N$ ) of synergies ( $S$ ) and motor commands ( $P$ ). The sizes of  $S$  and  $P$  are illustrated below

$$S = \left. \begin{array}{cccc} \overbrace{s_{11} & s_{12} & \dots & s_{1N}}^{N \text{ synergies}} \\ \vdots & \vdots & \vdots & \vdots \\ s_{M1} & s_{M2} & \dots & s_{MN} \end{array} \right\} M \text{ muscles}$$

$$P = \left. \begin{array}{cccc} \overbrace{p_1(1) & p_1(2) & \dots & p_1(K)}^{K \text{ samples}} \\ \vdots & \vdots & \vdots & \vdots \\ p_N(1) & p_N(2) & \dots & p_N(K) \end{array} \right\} N \text{ commands}$$

Figure E.4 and E.5 depicts the performance of the reconstruction of the muscle activity with a varying number of extracted synergies and motor commands shown for the individual muscles and for the concatenated muscles.

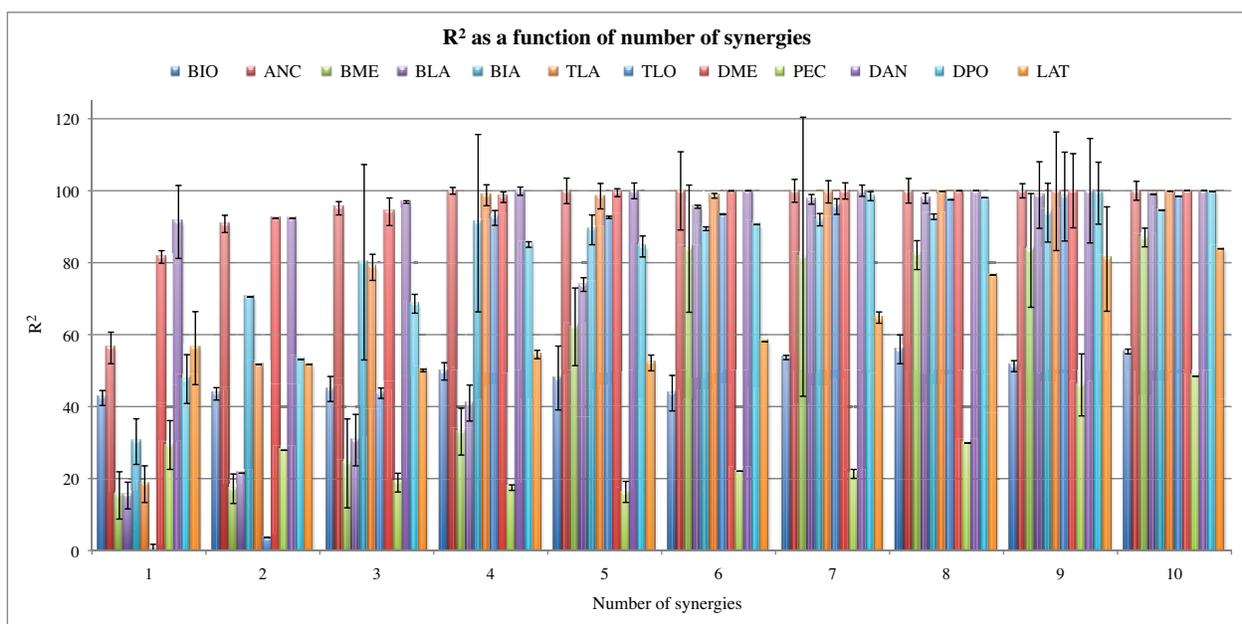


**Figure E.4:** Reconstruction performance of predicting the motor commands in the individual muscles (black, mean  $\pm$  SD across the number of muscles (in all three trials)) and for the concatenated data for all muscles (grey, mean  $\pm$  SD across the three trials) as a function of number of synergies.

From figure E.5 it appeared that one synergy was capable of reconstructing approx. 87% of the variability in the original muscle activity based in the Deltoid anterior. Especially the muscle activity in the three parts of the Deltoid were well estimated and so was the Anconeus using only one synergy. At four synergies, the activity in the Triceps long head was substantially improved compared to three synergies. Also Triceps lateral head, Brachialis, and Biceps lateral head. Pectoralis yielded in general a poor reconstruction.

It was seen that the prediction performance was improved for more than half the muscles up to 4 synergies, whereafter the performance flattened. Triceps long head and Lateral head especially showed an improvement from 3 to 4 synergies.

Based on the above observations, 4 was chosen as the number of synergies and motor commands for the predictions during reaching using configuration SSP (based on the muscle activity in the Deltoid muscle and gaze-direction for the extraction of the motor commands) as described in section 5.2 on page 43.



**Figure E.5:** Reconstruction performance of predicting the motor commands in the individual muscles (mean  $\pm$  SD across all trials), as a function of number of synergies.



---

# Introduction to neural networks

---

# F

Neural networks have been applied in a wide variety of engineering problems involving pattern recognition, pattern classification, adaptive filtering, and control [20, 162, 106, 71, 41]. The successes presented in [20, 162, 106, 71, 41], indicates the usefulness of neural networks in different components of a variety of FES control systems.

An artificial neural network (ANN) is an array of processors that is linked by connections that can be weakened or strengthened and the concept is based on inspiration from the interconnected neurons of the brain[152]. The ANN is based on the operation of biological neural networks, i.e. each neuron generates an output based on the inputs it receives from other neurons. Thus the functional model of the biological neuron contains three basic components, namely the synapses of the neuron (modeled as weights), and the components representing the activity within the neuron cell, which consists of an adder (sums up all inputs modified by their weights), and an activation function. In this fashion the neuron receives an input and produces a response as output [152].

The ANN is an adaptive, and most often a nonlinear system, which estimate a function (or the relationship between input and output vectors) without requiring a mathematical description of how the output functionally depends on the input, i.e. the network learn from input/output data samples [152]. The capability of learning complex nonlinear input-output mappings is one of the characteristics that make neural networks an attractive option in many biomedical problems. The adaptive capabilities of neural networks makes them particularly attractive for rehabilitation applications in which the system is often customized for particular individuals.[20, 106]

The network is adaptive in the sense that the system parameters are changed during operation, i.e. during training. After the training phase the ANN parameters are fixed and the system can be deployed, i.e. during validation.

The input/output training data are fundamental in ANN since it conveys the necessary information to find the optimal parameters of the network. According to the success of estimating a specific function based on the series of examples of correct responses, the connections between the processors are strengthened or weakened. Hence instead of working based on an explicit set of rules to follow, the network operates based on the weights of the links within it. The nonlinear nature of the neurons (the processing elements) provides the system with a high level of flexibility to achieve the desired input/output relation. [152]

One of the most widely used types of neural networks is the MLP NN (also denoted a multilayer feed-forward neural network).

## F.1 Multilayer perceptron neural network

An MLP NN consists of a network of perceptrons (neurons), in which the data flow strictly feedforward from input to output units, and the data processing can extend over multiple layers of units, without feedback connections. The MLP NN consists of the following elements [132]

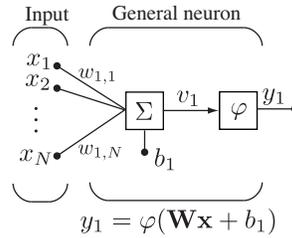
**Input layer** is a layer of neurons that receives information from external sources (in a biological neuron the input corresponds to sensory nerves and output to motor nerves), and passes this information to the network for processing

**Hidden layer** is a layer of neurons that receives information from the input layer and processes them in a hidden way, i.e. it has no direct connections to the outside world. All connections from the hidden layer are to other layers within the system

**Output Layer** is a layer of neurons that receives processed information and sends output signals out of the neural network

**Bias** is an offset or threshold value in the neurons. The function of the bias is to provide a threshold for the activation of neurons. The bias input is connected to each of the hidden and output neurons in a network

A typical neuron from a MLP network is illustrated in figure F.1. A larger neural network is constructed from several interconnecting neurons.



**Figure F.1:** Typical neuron of a MLP network consists of an input  $\mathbf{x}$  (with  $N$  elements), the weights  $\mathbf{W}$ , the bias  $b$ , and an activation function  $\varphi$ . Modified from [152]

The output  $y_k$  of the network is obtained by successively forming the linear combination of the input  $x_k$ , the weight  $w_k$ , and the bias  $b_k$  (i.e.  $v_k = \sum_{j=1}^N w_{kj} \cdot x_{kj} + b_k$ ,  $N$  being the number of inputs), and nonlinearly transform this by a nonlinear activation function, e.g. the  $S$ -shaped sigmoidal function, i.e. the output becomes  $\varphi(v_k)$ . [152] Thus the activation function relates the weighted sum plus a threshold of the synaptic inputs to the output.

Thus, each perceptron computes a single output from multiple real-valued inputs by forming a linear combination according to its input weights (the synaptic efficacy in a biological neuron), the bias (forming the post-synaptic potential of a neuron), and transforming the output through the nonlinear activation function. The strength of the connection between an input and a perceptron is determined by the value of the weight, i.e. negative weight values reflect inhibitory connections, while positive values designate excitatory connections.

Next the actual activity within the neuron cell is modeled by linear combining the input and the weights together with the bias. Lastly the activation function ( $\varphi$ ) maps the output into the range 0 to 1 in the case of the Sigmoid activation function, such as the simple logistic Sigmoid function defined in equation F.1. The function is bounded and monotonically increasing tending to 0 as the linearly combined input tends to  $-\infty$  and approaches 1 as the input tends to  $\infty$ . This function allows MLP networks to model both mildly and strongly nonlinear mappings, since the function is linear near the origin and saturating rather quickly away from the origin. [152] The capability of MLP NN stems from the non-linearities used within the nodes.

$$\varphi(x) = \frac{1}{1 + \exp^{-v_k}} \in (0, 1) \quad (\text{F.1})$$

This transfer function is often used in backpropagation networks, e.g. [69, 63, 70]. The derivative of equation F.1 can be expressed in terms of the function itself.

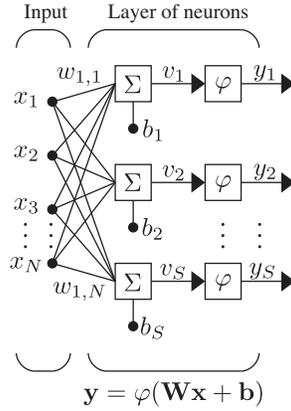
## F.2 Structure of MLP NN

Typical MLP NN consists of a set of source nodes forming the input layer, one or more hidden layers of computation nodes, and an output layer of nodes. A single layer neural network has severe restrictions limiting the type of tasks such a network can solve. A two layered feed-forward neural network with biases, a non-linear sigmoid layer, and a linear output layer is capable of approximating any function with a finite number of discontinuities to arbitrary precision [36, 48, 24].

If linear output neurons are used, the network outputs can take on any value instead of being limited to a small range as is the case with the sigmoid functions [91].

Figure F.2 shows a single layer network with  $N$  number of input elements and  $S$  number of neurons.

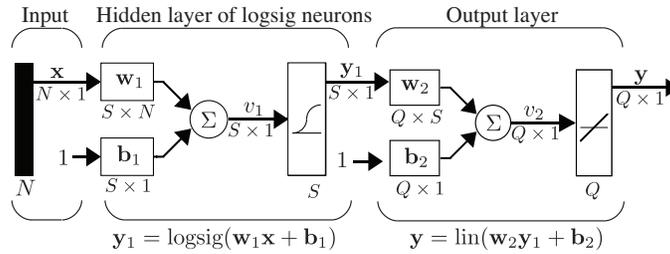
In figure F.2  $\mathbf{x}$  ( $N \times 1$ ) represents the input units,  $\mathbf{W}$  ( $S \times N$ ) the weights between each input and the unit in the following layer, and  $\mathbf{b}$  ( $S \times 1$ ) the bias added to each of the weighted sums from each of the neurons in the current layer,  $\mathbf{y}$  ( $S \times 1$ )



**Figure F.2:** Single layer MLP NN with  $N$  input units,  $S$  units in the hidden layer, and  $S$  units in the output layer. Modified from [152]

the output of the current layer, and finally the activation function  $\phi$  for each of the units.

Thus, a two layer network with one hidden layer can be represented as shown in figure F.3 where the input signal propagates through the network layer-by-layer and all neurons from one layer are connected to the neurons in the next layer.



**Figure F.3:** Two layer MLP neural network with one hidden layer consisting of  $S$  units with the logistic Sigmoid as transfer function, and one output layer consisting of  $Q$  units with linear transfer functions.

In figure F.3  $\mathbf{x}$  represents the input units ( $N \times 1$ ),  $\mathbf{W}_1$  ( $S \times N$ ) the weights between each input unit and each of the units in the hidden layer ( $S \times 1$ ),  $y_1$  ( $S \times 1$ ) is the output vector of the hidden layer,  $\mathbf{W}_2$  ( $Q \times S$ ) being the weights between the hidden layer and the output layer ( $Q \times 1$ ).  $\mathbf{b}_1$  and  $\mathbf{b}_2$  are the biases for each of the units in the hidden and the output layer respectively.

The network shown is a full-connected (output from each input and hidden neuron is distributed to all of the neurons in the following layer), two layer, feed-forward (the values only move from input to hidden to output layers, no values are fed back to earlier layers), perceptron neural network. The number of input neurons corresponds to the number of input variables (e.g. the number muscle activations), and the number of output neurons is the same as the number of desired output variables (e.g. muscle activation, low dimensional motor commands, kinematics).

The process of the neural network can be expressed in equation F.2

$$y_m = \underbrace{\sum_{q=1}^Q w_{mq}}_{\text{Output layer}} \cdot \underbrace{\phi \left( \sum_{n=1}^N w_{kn} \cdot x_n \right)}_{\text{1st hidden layer}} \quad (\text{F.2})$$

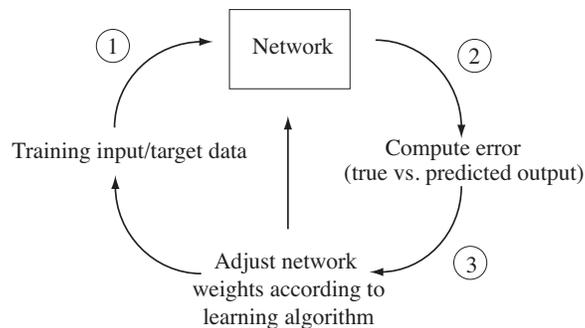
$x_n$  being the  $n^{\text{th}}$  input,  $w_{kn}$  being the weight from input  $n$  to node  $k$ ,  $w_{mq}$  being the weight from node  $q$  to output  $m$ , and  $y_m$  being output  $m$ .

The feed-forward neural network is also defined as a computational graph, whose nodes are computing units with directed edges that transmit numerical information from node to node. Each computing unit is capable of evaluating a single primitive function of its input. Hence, the network is an implementation of a composite function from input to output space, which is called the network function.[132]

### F.3 Learning of a neural network

A neural network has to be trained, such that the network produces the desired output given a set of inputs, i.e. the strength of the connection weights have to be adjusted adequately to achieve the desired behavior, which allow the characteristics of the input training data to be inferred. This requires a learning algorithm, to which a set of desired input-output mappings are presented, and in response executes correction steps iteratively until the network learns to produce the desired response. [132]

As seen in figure F.4, the learning algorithm is a closed-loop presentation of input-output mappings (training data) and corrections to the network parameters to minimize the prediction error [132]. In the training process the outputs is often specified as targets.



**Figure F.4:** Closed loop learning algorithm. The learning algorithm adapts the network parameters according to previous experience until a solution is found. For each input, the network produces an output (1) and the error between the predicted and desired output is calculated (2). The weights are adjusted to reduce the error according to some learning algorithm. (1) - (3) are repeated for every sample until the errors are minimized in the case of incremental training. Modified from [132]

Applying the network on the training data and comparing the predictions with the true outputs (the targets), forming the error function, can determine the error of a particular configuration of the network. Hence the goal of the training process is to find the optimal combination of weights, i.e. the lowest point in the many dimensional error surface each weight and threshold being a dimension in space, so that the network function approximates a given function described implicitly through the training samples [132, 152].

From an initial configuration of weights and thresholds, the objective of the training algorithm is to seek for the global minimum. The gradient of the error surface is typically calculated in each point and the search continues along the steepest descent[132].

The backpropagation algorithm is a popular learning method, which looks for the minimum of the error function in the weight/bias space using the method of gradient descent. This method requires the computation of the gradient vector of the error surface at each iteration step; hence differentiability of the error function is guaranteed by the choice of activation function, e.g. the sigmoid function. A differentiable activation function makes the function computed by a neural network differentiable (assuming that the integration function at each node is just the sum of the inputs), since the network itself only computes function compositions; hence the error function also becomes differentiable. [132]

The sigmoid function always have a positive derivative, so the slope of the error function (mean square error) provides a greater or lesser descent direction, which can be followed. The gradient vector points along the line of steepest

descent from the current point, thus by moving along by some step size the error will decrease.

The learning algorithm progresses iteratively through a number of epochs. For each epoch the training data set is applied to the network, from which the prediction error can be computed and the weights can be adjusted according to the error surface gradient and process can be repeated. The process continues until some a criterion has been reach, e.g. a certain number of epochs have evolved, the error has reached an appropriate level or when the network stops improving.

In the backpropagation learning algorithm the forward pass predicts the outputs from the given inputs and in the backward pass, the partial derivatives of the cost function with respect to the different weight parameters are propagated back through the network where the weights then can be adapted. Thus only the errors in the output are known and the output errors are propagated back to the previous layer using the output element weights [132]. The following summarizes the steps and equations involved in the backpropagation algorithm [132], here in a two layer network with a single hidden layer with a sigmoid transfer function as illustrated in figure F.3

1. Feed forward computation
2. Backpropagation to output layer
3. Backpropagation to hidden layer
4. Weight update

The algorithm is stopped according to a defined stop criteria.

In the description of the backpropagation algorithm, the the weights between the input and the hidden layer are denoted  $w_{ij}^{(1)}$ , for the  $i$ 'th input unit and the  $j$ 'th hidden layer unit, the weights between the hidden layer and the output layer are denoted  $w_{ij}^{(2)}$ , for the  $i$ 'th hidden layer unit and the  $j$ 'th output layer unit.  $\mathbf{Y}^{(1)}$  is the output from the hidden layer, and  $\mathbf{Y}^{(2)}$  is the output from the output layer.

The bias of each of the computing units is implemented as the weight of an additional edge by extending the input vector and output vector of the hidden unit with a 1-component (not shown on the figure). Thus the weights between the constant 1 and the hidden unit is denoted  $w_{N+1,S}^{(1)}$  and between the hidden unit and the output unit  $j$  is denoted  $w_{S+1,Q}^{(2)}$ . This leads to  $(N+1) \times S$  weights between the input sites and the hidden units and  $(S+1) \times Q (=1)$  between the hidden and output units. The weight matrices with the last row corresponding to the bias of the computing units are denoted  $\overline{\mathbf{W}}_1$  and  $\overline{\mathbf{W}}_2$ . The input vector  $\mathbf{x}$  is likewise extended to  $\overline{\mathbf{x}} = (x_1, \dots, x_N, 1)$ . The output  $y_j$  of the  $j$ 'th hidden unit is given by equation F.3

$$y_j^{(1)} = \varphi \left( \sum_{i=1}^{N+1} w_{ij}^{(1)} \overline{x}_i \right) \quad (\text{F.3})$$

since the excitation of the  $j$ 'th hidden layer is  $w_{ij}^{(1)} \overline{x}_i$  and  $\varphi$  being the sigmoid activation function. Hence the output vector  $\mathbf{y}^{(1)}$  with its components being the outputs of the hidden layer units can be written as the vector-matrix multiplication in equation F.4

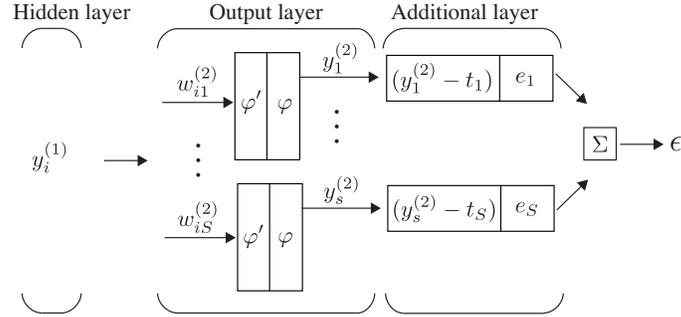
$$\mathbf{y}^{(1)} = \varphi \left( \overline{\mathbf{x}} \overline{\mathbf{W}}_1 \right) \quad (\text{F.4})$$

and by applying each sigmoid  $\varphi$  to each component of the argument vector.

The excitation of the units in the output layer can be computed by using the extended vector  $\overline{\mathbf{y}}^{(1)}$  as seen in equation F.5

$$\mathbf{y}^{(2)} = \varphi \left( \overline{\mathbf{y}}^{(1)} \overline{\mathbf{W}}_2 \right) \quad (\text{F.5})$$

The total error of a training set can be computed by addition of the errors computed for each of the training input-output sample. This error computation is based on an extended network as the one shown in figure F.5



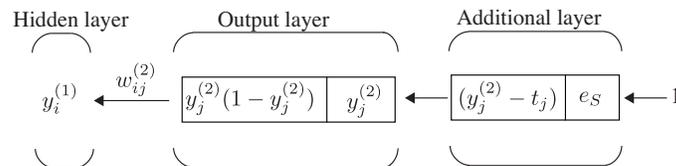
**Figure F.5:** Extended multilayer network for the computation of the network error.  $y_i^{(1)}$  being the  $i$ 'th output of the hidden layer units,  $w_{ij}^{(2)}$  being the weight between this unit ( $i$ ) and the output unit ( $j$ ),  $\phi$  being the activation function, and  $\phi'$  the derivative of the activation function, and  $\mathbf{y}^{(2)}$  being the outputs of the output layer. Modified from [132]

In figure F.5 The network has been extended with an additional layer of units, in which the error function  $e$  is computed for the  $i$ 'th component of the output vector. Additionally the error  $(y_i^{(2)} - t_i)$ ,  $t$  being the true output, is computed and store. Still, each output unit  $i$  in the original network computes the sigmoid  $\phi$  and produces the output  $y_i^{(2)}$ . It is seen how the addition of the error functions for each output unit yields the network error  $\epsilon$ .

For  $p$  input-output samples in the training set, the total error function can be computed by creating  $p$  networks like the one shown in figure F.5 and adding the individual error functions.

After setting the initial weights, the algorithm follows the steps listed above. Initially the input vector  $\mathbf{x}$  is presented to the network, and the vectors  $\mathbf{y}^{(1)}$  and  $\mathbf{y}^{(2)}$  are computed and stored in each unit, and likewise is the derivative of the activation function.

After the feed-forward step follows the backpropagation step, in which the partial derivative  $\frac{\partial E}{\partial w_{ij}^{(2)}}$  is computed. The backpropagation path from the network output to the output unit  $j$  is illustrated in figure F.6



**Figure F.6:** Backpropagation path to output unit  $j$ . The constant 1 is fed into the output unit and the network is run backwards. Incoming information to a node is added and the result is multiplied by the value stored in the left part of the unit.  $y_j^{(2)}$  being the output of output neuron  $j$ ,  $t_j$  being the true output of unit  $j$ ,  $e_S$  being the error function for unit  $j$ . Modified from [132]

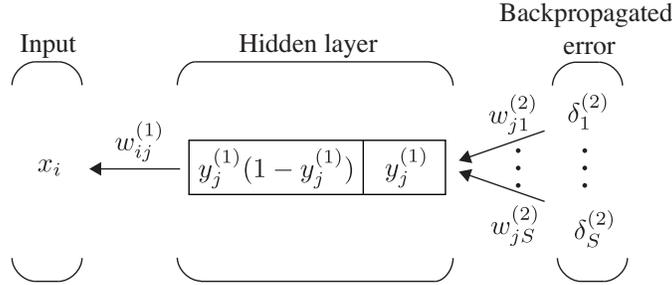
Thus, the backpropagated error  $\delta_j$  can be written in equation F.6

$$\delta_j^{(2)} = y_j^{(2)}(1 - y_j^{(2)})(y_j^{(2)} - t_j) \quad (\text{F.6})$$

Hence the partial derivative can be written in equation F.7

$$\frac{\partial E}{\partial w_{ij}^{(2)}} = \delta_j^{(2)} x_i^{(1)} \quad (\text{F.7})$$

by considering the weight  $w_{ij}^{(2)}$  to be a variable and the input  $x_i^{(1)}$  to be a constant (originating from the previous layer). The backpropagation to the hidden layer can now be computed as illustrated in figure F.7.



**Figure F.7:** Backpropagation path from the output to the hidden layer.  $w_{ij}^{(1)}$  being the weight from hidden unit  $j$  to input  $i$ ,  $y_j^{(1)}$  being the output of the hidden unit  $j$ ,  $w_{jk}^{(2)}$  being the weight from output unit  $k$  to hidden unit  $j$ ,  $\delta_k^{(2)}$  being the backpropagated error from output unit  $k$ . Modified from [132]

The partial derivative of interest is now  $\frac{\partial E}{\partial w_{ij}^{(1)}}$ . Each unit  $j$  in the hidden layer is connected to each unit  $q = 1, \dots, Q$  in the output layer with the weight  $w_{jq}^{(2)}$ . The error is computed by taking into account all possible backwards paths as expressed in equation F.8

$$\delta_j^{(1)} = y_j^{(1)}(1 - y_j^{(1)}) \sum_{q=1}^Q w_{jq}^{(2)} \delta_q^{(2)} \quad (\text{F.8})$$

And the partial derivative becomes as expressed in equation F.9

$$\frac{\partial E}{\partial w_{ij}^{(1)}} = \delta_j^{(1)} x_i \quad (\text{F.9})$$

$x_i$  being the input vector.

Finally, after computing all the partial derivatives, the last step in the algorithm is to update the weights in the negative gradient direction. A learning step  $\gamma$  defines the step length of the correction. The corrections are expressed in equation F.10

$$\Delta w_{ij}^{(2)} = -\gamma x_i^{(1)} \delta_j^{(2)}, \quad \text{for } i = 1, \dots, S+1; j = 1, \dots, Q \quad (\text{F.10})$$

$$\Delta w_{ij}^{(1)} = -\gamma x_i \delta_j^{(1)}, \quad \text{for } i = 1, \dots, N+1; j = 1, \dots, S \quad (\text{F.11})$$

$$(\text{F.12})$$

Remembering that  $x_{N+1} = y_{S+1}^1 = 1$ . Thus the corrections to the weights are made after the backpropagated error has been computed for all units in the network. The case with number of training samples  $p > 1$ , the weight corrections are computed for each pattern  $\Delta_1 w_{ij}^{(1)}, \Delta_2 w_{ij}^{(1)}, \dots, \Delta_p w_{ij}^{(1)}$  and the update in the gradient direction becomes  $\Delta w_{ij}^{(1)} = \Delta_1 w_{ij}^{(1)} + \Delta_2 w_{ij}^{(1)} + \dots + \Delta_p w_{ij}^{(1)}$  for learning in batch mode, in which weights and biases are only updated after all the inputs and targets are presented.

In summary, the feed forward computation stores both the evaluation of the primitive function and the derivative. In the backpropagation the constant 1 is fed into the output layer and the network is run backwards. The incoming information to a node is added and the result is multiplied by the derivative stored in that unit. The result is transmitted to the left in the of the unit and the result collected at the input unit is the derivative of the network function with respect to the input to the network.

This method suffers from various convergence problems [129]. These problems can be solved by using second order derivatives, which is applied in the Levenberg-Marquardt (LM) algorithm one the most widely used optimization algorithm and is suggested for moderate-sized feedforward neural networks (up to several hundred weights) [43, 84, 28].

In the following, the LM algorithm [80, 133] is introduced to an extent which explains the parameters, which are configured in practice. For a detailed description of the LM algorithm please refer to [62, 124], since this is beyond the scope of this report.

### The Levenberg-Marquardt algorithm

The LM algorithm relies on both the first and second order derivative of the error in the search for the optimal weights. The LM algorithm provides a faster solution due to the incorporation of the second derivative of the error information and automatically incorporates adjustments of the learning parameters. [80]

The idea of the LM algorithm is to minimize the squared distance  $\epsilon^T \epsilon$ ,  $\epsilon = \mathbf{x} - \hat{\mathbf{x}}$ . Letting  $f$  being an assumed functional relation which maps a parameter vector  $\mathbf{p} \in \mathbf{R}^m$  to an estimated measurement vector  $\hat{\mathbf{x}} = f(\mathbf{p})$ ,  $\hat{\mathbf{x}} \in \mathbf{R}^n$ . The basis of the LM algorithm is a linear approximation to  $f$  in the neighborhood of  $\mathbf{p}$ . [133, 80]

For a small  $\|\Delta_p\|$ , a Taylor series expansion leads to the approximation in equation F.13

$$f(\mathbf{p} + \Delta_p) \approx f(\mathbf{p}) + \mathbf{J}\Delta_p \quad (\text{F.13})$$

$\mathbf{J}$  being the Jacobian matrix  $\frac{\partial f(\mathbf{p})}{\partial \mathbf{p}}$ . At each step in the iterative algorithm is to find the  $\Delta_p$  that minimizes the quantity given in equation F.14

$$\|\mathbf{x} - f(\mathbf{p} + \Delta_p)\| \approx \|\mathbf{x} - f(\mathbf{p}) - \mathbf{J}\Delta_p\| = \|\epsilon - \mathbf{J}\Delta_p\| \quad (\text{F.14})$$

The minimum is attained when  $\mathbf{J}\Delta_p - \epsilon$  is orthogonal to the column space of  $\mathbf{J}$ , yielding  $\mathbf{J}^T(\mathbf{J}\Delta_p - \epsilon) = 0$ ,  $\Delta_p$  being the solution to the normal equations in F.15

$$\mathbf{J}^T \mathbf{J} \Delta_p = \mathbf{J}^T \epsilon \quad (\text{F.15})$$

The matrix  $\mathbf{J}^T \mathbf{J}$  in the left side in equation F.15 is the approximate Hessian, i.e. an approximation to the matrix of second order derivatives. The LM algorithm solves a slight variation of equation F.15 as seen in equation F.16, called the augmented normal equations

$$\mathbf{N} \Delta_p = \mathbf{J}^T \epsilon \quad (\text{F.16})$$

The off-diagonal elements of  $\mathbf{N}$  are identical to the corresponding elements of  $\mathbf{J}^T \mathbf{J}$  and the diagonal elements are given in equation F.17

$$\mathbf{N}_{ii} = \mu + [\mathbf{J}^T \mathbf{J}]_{ii}, \mu > 0 \quad (\text{F.17})$$

The diagonal elements of  $\mathbf{J}^T \mathbf{J}$  is altered by damping, where  $\mu$  is the damping term. If the updated parameter vector  $\mathbf{p} + \Delta_p$ ,  $\Delta_p$  computed from equation F.16, leads to a reduction in the error  $\epsilon$ , the update is accepted and the process repeats with a decreased damping term. Otherwise equation F.16 is solved again, and the process iterates until a value of  $\Delta_p$  is found (corresponding to one iteration of the LM algorithm). The damping term is adjusted at each iteration to assure a reduction in the error  $\epsilon$ , thus the damping is raised if a step fails to reduce  $\epsilon$  and otherwise the damping is reduced. [73]

The steps of the LM algorithm is the following [150]

1. Inputs are presented to the network and the corresponding outputs and errors are computed. The mean square error over all inputs is computed
2. The Jacobian matrix  $\mathbf{J}$  with respect to  $\mathbf{p}$  is computed,  $\mathbf{p}$  representing the weights and biases of the network
3. The Levenberg-Marquardt weight update equation F.16 is solved to obtain  $\Delta_p$
4. The error is recomputed using  $\mathbf{p} + \Delta_p$ . In case the new error is smaller than the one computed in step 1, the training parameter  $\mu$  is reduced by  $\mu^-$ , and  $\mathbf{p} = \mathbf{p} + \Delta_p$  is computed and the algorithm returns back to step 1. If the error is not reduced, then  $\mu$  is increased by  $\mu^+$  and returns to step 3

5. The algorithm is terminated when at least one of the following conditions is met

Magnitude of gradient ( $\epsilon \epsilon'$ , i.e.  $\mathbf{J}^T \epsilon$ ) drops below some threshold  $\epsilon_1$

The relative change in the magnitude of  $\delta_p$  drops below a threshold  $\epsilon_2$

The error  $\epsilon^T \epsilon$  drops below a threshold  $\epsilon_3$

A maximum number of iterations  $k_{max}$  is completed

$\mu^+$  and  $\mu^-$  are predefined values, typically initiated to 10 and 0.1 respectively.[150]



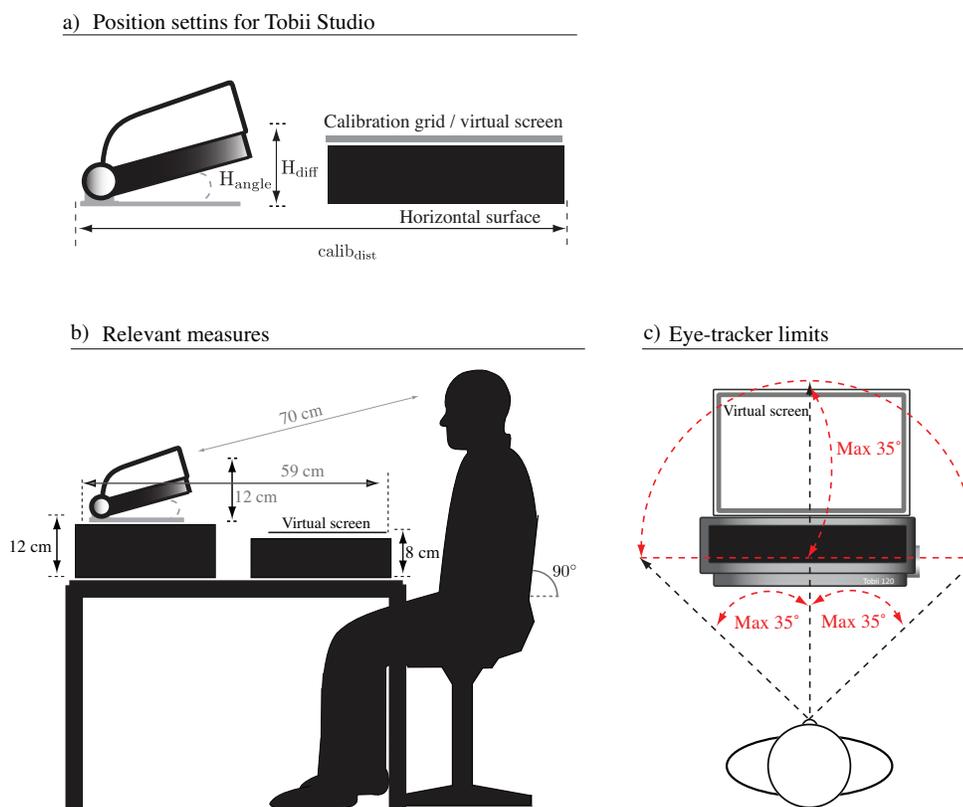
# Configuration of applied eyetracker

# G

This chapter contains an elaborated description of the configuration of the specific eyetracker.

## G.1 Eyetracker Tobii X120

Eye-tracking was recorded using Tobii X120 Eye-Tracker (firmware 1.1.14) (sampling frequency 120 Hz), and Tobii Studio v.1.3.23 eye-tracker software running on Windows XP. The eye tracker communicated with the computer via a standard network cable.[155] Figure G.1 shows the eyetracking setup.



**Figure G.1:** Eye-tracking setup. a) Position parameters set in Tobii Studio b) Relevant measures for the eyetracking setup. The subject was placed approximately 70 cm. from the eye-tracker in an upright position ( $90^\circ$  with the horizontal plane). c) Limits of the eye-tracker. It was ensured that gaze angle did not exceed  $35^\circ$  to any point on the virtual screen.

**Distance to calibration grid,  $calib_{dist}$**  -59 cm, the minus indicates that the eye-tracker is located behind the virtual screen measured from the back edge of the eye-tracker to the front end of the virtual screen

**Horizontal eye-tracker angle,  $H_{angle}$**  Adjusted with respect to each subject, such that the eyes are placed in the center of the status track window when the subject is placed 70 cm from the eye-tracker. Approx.  $15^\circ$

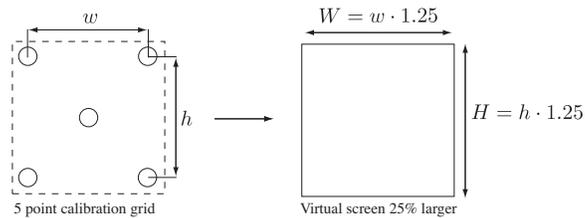
**Vertical eye-tracker angle**  $90^\circ$  ( $0^\circ$  indicates completely vertical)

**Active display area** 40 cm  $\times$  30 cm set as illustrated in figure G.2

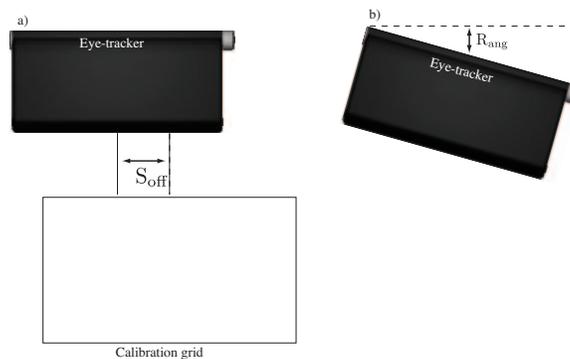
**Height difference between display and eye-tracker,  $H_{diff}$**  15 cm. The height from the bottom of the eye-tracker foot and the bottom of the active display area.

**Side offset,  $S_{\text{off}}$**  0 cm, as illustrated in figure G.3

**Rotation,  $R_{\text{ang}}$**   $0^\circ$  as illustrated in figure G.3



**Figure G.2:** The height and width of the virtual screen set 25% Larger than the calibration area



**Figure G.3:** a) Side offset measured from the center of the eye-tracker to the center of the active area b) Rotation measured between the back of the eye-tracker and the active display

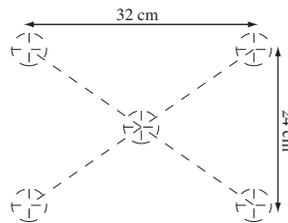
The distance from the eye-tracker to the subject was approximately 70 cm, adjusted to achieve the optimal configuration according to the track status window, i.e. two dots representing the eyes should be seen in the middle of track status window and the status color on the bottom of the window should show green as seen in figure G.4



**Figure G.4:** Track status meter of eye-tracker. An optimal configuration with two white dots representing the eyes and the status color being green, the black box represents the field of view of the eyetracking.

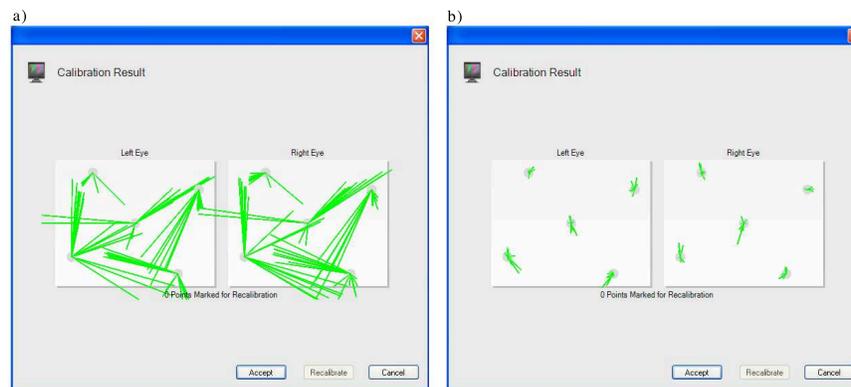
The placement of the eye-tracker ensured that the gaze angle did not exceed  $35^\circ$  to any point on the virtual screen. Initially the eye-tracker was calibrated, to learn the characteristics of the subject's eye movement to achieve accurate estimation of the gaze points. Thus when the eyes were detected the eye tracker tracker was trained to associate a position of the eye to a specific location on the object. During calibration, the subject concentrated on focusing the eyes on the calibration points. Relaxed movement improved the calibration quality.

The setup was configured as a 'Screen recording' configured in Tobii Studio and performed with a perspective calibration. This was performed by placing a calibration grid (white paperboard with 5 calibration points marked, one in each corner and one in the middle as seen on figure G.5) on the surface to be tracked and with the subject placed in the center of the calibration grid. The calibration grid was attached to the surface below the virtual screen.



**Figure G.5:** Perspective 5 point calibration grid

Manual calibration mode was selected in Tobii Studio and was evaluated according to calibration plot showing error vectors as seen on figure G.6, thus in case of great errors (long lines indicates insufficient calibration) (a), recalibration was performed until sufficient calibration was achieved (b).



**Figure G.6:** Calibration plot from the eye-tracker. a) Calibration plot of the eye-tracker shows great error, implying that recalibration is needed. b) Calibration of the eye-tracker shows only small error vectors, thus no recalibration is needed.

The following settings were configured in Tobii Studio

**Fixation filter** ClearView fixation filter

**Fixation radius** 100

**Min duration** 50

The following parameters was exported from Tobii Studio after the recordings

**DateTimeStamp** Recalculate d and shown in minutes, seconds and milliseconds (mm:ss:ms)

**Event** Automatic events (start and stop, key presses), logged events (TriggerData)

**Descriptor** Description of the event ('Starts', 'Stop'), the synchronization signals

**GazePointX** The horizontal grid position based on the average of both eyes

**GazePointY** The vertical grid position based on the average of both eyes

Separated in three files, on containing **DateTimeStamp**, **Event**, **Descriptor**, another containing **GazePointX** and **GazePointY**, and the last file containing **Event**. The above configuration was saved in expSetup (Export/Text Export/) and selected for all recordings. The data points between the logged events 'Start' and 'Stop' were used for the data analysis.



---

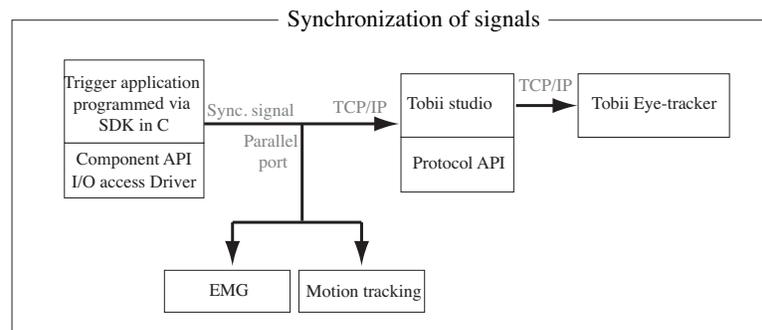
# Trigger application

---

# H

This chapter describes how the synchronization was initiated by the eyetracking software. Additionally this chapter documents in detail how the trigger application was programmed via the API.

The synchronization setup is shown in figure H.1



**Figure H.1:** Synchronization of the recording systems. Via an application programmed in C++ utilizing the Tobii SDK, a signal was sent to via the parallel port to initiate the other recording systems and at the same time, the eyetracking was initiated.

## H.1 Required software

The following software was used for the development of the application to trigger Tobii Studio.

- Windows XP (operating system)
- Tobii Studio v. 1.3.23 (Eye-tracking software)
- Tobii Studio SDK v. 2.4.3 (Software development kit for Tobii Studio)
- cv.dll, cv.lib, and cv.h (Makes the Tobii Trigger Low Level API available to the compiler)
- Microsoft Visual Studio 2008 (compiler)
- UserPort.sys (kernel driver for Windows NT/2000, which enables usermode programs to I/O ports, hence it becomes possible to access hardware directly from a normal executable as in previous Windows versions and DOS)

It appears from the description below how the different elements listed above was used.

## H.2 Interfacing the parallel port

This section describes in detail how the interfacing with the parallel port was conducted to be able to use an external trigger signal for synchronization of the eye-tracker with the EMG and motion capture system. Initially the parallel port is introduced.

### H.2.1 The parallel port

Initially it is needed to interface with the parallel port to acquire the analog input signal which serves as the external trigger. Initially the parallel port is introduced followed by a description on how to interface it. The parallel port is a 25 pin female connector as seen in figure H.2. The primary use of the parallel port in the computer is to connect printers and is specifically designed for this purpose.



**Figure H.2:** Parallel port.

The lines in DB25 connector are divided in to three groups

- Data lines (data bus)
- Control lines
- Status lines

Thus data is transferred via the data lines (8 lines), control lines (4 lines) controls the peripheral devices and they returns status signals back to the computer through the status lines (5 lines). These three categories of lines are connected to Data, Control And Status registers internally in the computer, i.e. by manipulating these registers it is possible to read and write to the parallel port. The registers are virtually connected to the corresponding lines, hence what is written to the registers appears in corresponding lines as voltages. E.g. if '0x01' is written to pin 0 in the Data register, the line Data0 will be driven to a high voltage, '0xFF' will drive all pins in the Data register to a high voltage. The parallel port is TTL-compatible, implying 'high' is between +2.4 and +5v and a 'low' is between 0 and 0.8v.

### H.2.2 The address of the parallel port

In a typical PC the base address of LPT1 is 0x378. The data register then resides at this base address, the status register at base address +1 (0x379) and the control register is at base address +2 (0x37a).

### H.2.3 Reading from and writing to the parallel port

UserPort.sys was installed by copying it to Windows/system32 and run UserPort.exe. select the addresses that should be open and click start. The parallel port was tested by a simple analog circuit consisting of a LED and a resistor and lpt.exe, which allows you to write and read from the parallel port via a GUI, i.e. turning the LED on and off.

The kernel driver UserPort.sys was started by using UserPort.exe, which allowed access to the parallel port directly from a Win32 application written in C. The driver made it possible to read from and write to the parallel port by pre-configured functions.

## H.3 Tobii Studio

Triggered events defined what parts of the data that was to be analyzed. The application developed to trigger the Tobii Studio interfaced also with the parallel port . This was achieved by using the Tobii SDK, and specifically the Tobii Studio Trigger Low Level API, which is an API accessible from Windows 32-bit environments including Windows XP. The API is implemented as a traditional Windows DLL by cv.dll and can be used by any language that supports DLL calls such as C. This API is a set of function calls that handles the connection to and the communication with Tobii Studio. Two steps were initially performed as described below

- Tobii SDK was installed (moved CVTrigComp.dll and cv.dll to a directory accessible to the computer and registering the components)
- cv.dll, cv.lib, and cv.h was moved from the folders in C:/Program Files/Tobii/SDK to the directory in which the project was located

The following functions is accessible by using the API declared in the header file cv.h

**CV\_Init(ipAddress, CLEARVIEW\_SERVER\_PORT\_NUMBER, NULL)** Connect to ClearView and performs internal initializations, creates a connection to ClearView and sets up necessary internal states

**CV\_StartWithName("My first recording")** Start a named recording

**CV\_LogEvent("MyEventText")** Add a text data record to an ongoing recording, which is logged

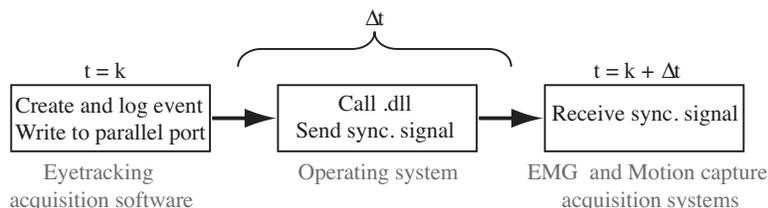
**CV\_GetLastError** Get error details if any function call returned error. Whenever a CV\_ERROR is returned from any CV\_XXX function a call to this function will return the specific last error

**CV\_Stop()** Stop an ongoing recording, Disconnects ClearView and performs necessary cleanups

**CV\_Close()** Disconnect from ClearView and perform internal cleanups

## H.4 Synchronization via software

The data recording systems, i.e. EMG, motion capture, and eyetracking were synchronized via a signal send from the developed application to the eyetracking acquisition software, at the time of an event, created at the start of the recordings of the eyetracker. This synchronization signal was sent to the EMG and motion capture system. Figure H.3 depicts the synchronization process and the possible delay this imposes between the signals.



**Figure H.3:** Possible synchronization delay of the data recording systems. The synchronization signal sent from the application to the EMG and motion capture system are delayed with  $\Delta t$  with respect to the logged event.

As depicted on figure H.1 the developed application communicates with eyetracker acquisition software via an API. At the time of the event being logged in the eyetracking acquisition software, a .dll was called before the synchronization signal was sent. This implied a delay between the recording of the eyetracking and the other recording systems. The delay was depended on the operating system, i.e. the number of processes running at the time. Having only the eyetracking acquisition software launched minimized this delay but theoretically, it could have an effect upon the predictions.

