Cortical Plasticity

A Model of Somatosensory Cortex

Magnus Johnsson
Master's Thesis in Cognitive Science
Spring 2004

LUNDS
UNIVERSITET

Lund University
Cognitive Science

Table of Contents

Abstract. After nerve injuries, e.g. following a re-implantation of an accidentally cut hand, nerve fibres grows together again but will be misconnected. This implies an incorrectly mapping of the hand in somatosensory cortex. However, the brain is plastic and thus a natural question is if there is a way to stimulate the hand of such a patient so that the recovery will be optimal. To come up with suggestions on this, a computational model of the dorsal column-medial leminiscal system was implemented. Different versions of this model were used to simulate the reorganization of somatosensory cortex due to misconnected nerve fibres. A number of tactile training methods were simulated and their capability to reorganize cortex was measured. The simulation results brought about the following recommendation on how to stimulate a patient's hand: Randomly select a small set of points on the palm and stimulate them shortly, and then select some new points. This recommendation is due to that such training methods worked well in all versions of the model. When the nerve injury is only partial, perhaps so-called cortical induction would also be a good choice.

1. Introduction

When an object is grasped by the hand, a number of touch sensitive receptors in the skin are stimulated, and signals are sent through the nerve fibres in the spinal column, via the medulla and the thalamus to the somatosensory cortex. The object is deconstructed into small segments, because every nerve fibre only conveys information from a small part of the skin surface, but the pattern is recreated in the brain by the cortical analysis. If there is an injury to the nerves from the hand to the medulla, the pattern of signals may be severely disturbed and the system will not work properly.

Therefore, there is an interest in knowing how different forms of sensory stimulation influences the reorganization of somatosensory cortex following nerve injuries, e.g. where the nerves between the hand and the brain have been cut partly or altogether and then have grown back together. One example of such a situation is when a hand has been cut during an accident and has then been sewn together with the arm again.

After such an injury, the nerves certainly grow together again, but unfortunately the individual nerve fibres will be misconnected. The consequence of this is that the sensory signals to the brain will be incorrectly mapped in the cortex.

However, the brain is plastic, and it has an ability for reorganization. Therefore the somato-sensory maps in cortex begin to reorganize to some extent, and begin to code in a more correct way again.

It is interesting to know how different forms of sensory stimulation, or training, influences how fast and to what qualitative level this reorganization in somatosensory cortex occurs, because this could be used to design better rehabilitation programs for patients suffering from misconnected nerve fibres.

One way to get an idea of somatosensory reorganization is to create a computational model of the relevant neurophysiological systems, and then use that model to simulate the reorganization that takes place. By accomplishing simulations for different forms of sensory training and compare these to each other, one can hope to get a hint on what form of sensory stimulation will be appropriate for a real patient, so that his or her recovery from the injury will be as good as possible.

One can wonder, why the use of simulations with computational models should be so good, and to what extent they tell us anything about reality. The answer is that a good computational model is founded in theoretical and empirical knowledge that has been formulated as executable code. Due to the computational power of modern computers it becomes possible to investigate consequences of scientific theories to an extent that would otherwise not be possible, since the chains of casual relations can exhibit a complexity that is hardly penetrable. The results from a simulation with a computational model can then be compared to empirical observations and if they agree, they constitute a support for the theory that the computational model was built on. The results can also point out the direction for further empirical investigations by providing new predictions. In this way simulations can also contribute to extend the existing theory. This is exactly the aim of this project, i.e. to aid in the extension of the existing knowledge by generating new ideas with clinical application.

More precisely the aims of this project have been to construct a computational model of cortical plasticity in somatosensory cortex following nerve regeneration, and to use it to investigate how different forms of sensory training influence the reorganization of the maps in somatosensory cortex.

2. Theoretical Background

In this chapter, a theoretical background will be given. It contains a discussion, based on the literature, of the neurophysiological systems that have been modelled. The purpose of this is to show

what empirical knowledge the model is based on. Discussions on neural modelling in general, and a review of research on, and modelling of, cortical plasticity are given. A description of Kohonen's self-organizing map algorithm is given, since the model of the somatosensory cortex is based on it. In addition a suggested method for sensory re-education when the nerve injury is only partial, which has been simulated in this project, is discussed.

2.1 Neurophysiological Systems

The system that conveys information of touch from the skin to the somatosensory cortex is called the dorsal column-medial leminiscal system (Gardner & Kandel, 2000; Gardner, Martin & Jessell, 2000), see Fig. 2.1. It is organized so that the sensory information is processed in a series of relay regions, and this is a common organizational principle for sensory systems. For touch, there are only two relays on the path between the receptors in the skin and the somatosensory cortex. The receptor neurons are located in ganglions on the dorsal root of a spinal nerve and they have axons that are divided into two branches. One branch terminates in the skin where its terminals are sensitive to tactile stimuli. The other branch of the axon enters the spinal column, ascends through it and terminates in the gracile or the cuneate nuclei in caudal medulla in the brain stem. The afferent axons from medulla cross on their way through the medulla and pons and terminate in the ventral posterior lateral nucleus of the contralateral thalamus. Consequently, the right hemisphere of the brain receives sensory input from the left part of the body and vice versa. From the thalamus, the neurons send their axons straight to the primary somatosensory cortex, Brodmann's areas 1 and 3b, which are located in the postcentral gyrus in the parietal lobe.

A relay nucleus consists of projection (relay) neurons, with afferent axons that terminate in the next nucleus in the pathway, and inhibitory interneurons (Gardner & Kandel, 2000; Gardner, Martin & Jessell, 2000). The afferent axons branches to innervate several neurons in the consecutive relay nucleus or cortex. One consequence of this is that the receptive field of a neuron enlarges for each relay site, and the cortical neurons have the largest receptive fields. A receptive field is the area on the skin innervated by a neuron (Gardner & Kandel, 2000; Gardner, Martin & Jessell, 2000), and for the neurons in the dorsal root ganglion the receptive fields delimits the resolution of tactile stimulation, i.e. smaller and more receptive fields in a skin area gives a better resolution. In the dorsal root ganglion, the receptive field of a neuron represents the input from only one receptor, while in the somatosensory cortex the receptive field of a neuron represents the composite input of 300 – 400 receptors. However, the size of a receptive field in cortex or in a relay nucleus is not fixed, but can be modified by experience or for example, injury to a sensory nerve. In spite of the size of the receptive field of cortical neurons, fine details can be discriminated. This is because a cortical neuron responds most intensely in the middle of its receptive field.

As mentioned above there are inhibitory interneurons in each relay nucleus, i.e. in the cuneate nucleus and gracile nucleus in medulla and ventral posterior lateral nucleus in thalamus, and they are also present in somatosensory cortex. One function of the inhibitory interneurons is to improve the resolution of the sensory input, and this is done because the inhibition restricts the spread of excitation, which is caused by the diverging of the axons coming in from the previous relay nucleus. Another function is to provide a reinforcement of contrasts. A consequence of the inhibition is that the activity in those neurons that are not most intensely activated is reduced, so that only the strongest of several competing responses are expressed.
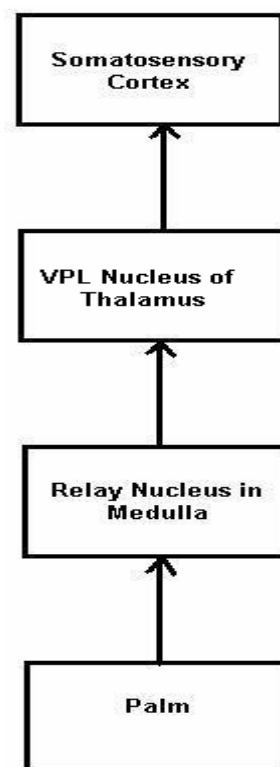

Fig. 2.1. The parts of the dorsal column-medial leminiscal system that conveys touch information from the palm.

4

The receptors in the skin are of different types, but for the submodality touch, the peripheral terminals of the dorsal root ganglion neurons, which as mentioned above, are sensitive to tactile stimulation, are encapsulated in non-neural structures called corpuscles. These structures alter the touch sensitive properties, so that the receptor responses are more functionally well fitted. Touch sensitive receptors are called mechanoreceptors. The axons of the dorsal root ganglion neurons are different depending on receptor type (Gardner, Martin & Jessell, 2000), e.g. for touch sensitive receptors the axons are more myelinated than the axons belonging to receptors for pain. Because of this difference, the signals from touch sensitive receptors propagate faster than the signals from receptors for pain. All nerve fibres from a dorsal root are connected to a continuous area of skin, and such an area is called a dermatome.

In the sensory system, from the receptors in the skin to the somatosensory cortex, a somatotopic organization is maintained (Gardner & Kandel, 2000). This means that the representation of the skin surface in the relay nuclei and in the somatosensory cortex preserves the topography of the skin surface. So there is a sensory map at each level of the sensory pathway.

Although there is no general theory on the operation of cortex, there are some general principles of the cortical architecture and processing (Buonomano & Merzenich, 1998). Sensory cortex, as well as most other cortical areas, is divided into six layers, L-I-VI. The sensory input arrives first and foremost to L-IV and to some extent to L-VI and L-III and are, as mentioned above, arising from thalamus via thalamocortical axons. Output to other cortical and subcortical regions proceeds primarily via pyramidal cells in L-V and L-VI. There is probably a general flow of information between the cortical layers, but also a horizontal flow within the same layer. Horizontal connectivity is probably important for cortical map reorganization. Cortical map reorganization will be discussed further in the next section below.

Besides the organization into layers, somatosensory cortex is also organized into columns. The neurons in each column, innervates a tiny continuous area on the skin, in the case of touch, and columns are organized in somatosensory cortex so that they preserve the topography of the skin. In somatosensory cortex, there is a complete map of the body. In fact, there are complete topographical representations of the body in each of the four areas BA1, BA2, BA3a and BA3b. Such a map is not an undistorted representation of the body, instead certain parts of the body are exaggerated, while others are diminished. This is because the

innervation density varies for different body parts. For example the hand, foot, and mouth have large representations. The representations in cortex can be altered by experience, so they are not fixed.

It is a common feature of sensory areas, that they represent their sensory epithelial surfaces in a topographical way (Buonomano & Merzenich, 1998). When it comes to the somatosensory cortex, the mapping is somatotopic, which, as mentioned above, means neighbouring cortical regions responds to neighbouring regions on the skin surface. However, a cortical map does not need to be a representation of spatially separated features. It can also, for example, be a representation of temporally separated features. For example, Kilgard and Merzenich (2001) carried out an experiment with adult rats in which they concluded that after repeated exposure to three-element sequences (high tone – low tone – noise burst), 25% of the neurons in auditory cortex exhibited a responding facilitation to a low tone if it was preceded by a high tone.

A lot of research has been done on the somatosensory cortex. An example of how this research is done is a way to detect the representation of the contralateral body surface in somatosensory cortex. Microelectrodes can be used for this purpose (Florence, Jain & Kaas, 1997). These microelectrodes can be made of glass and filled with concentrated NaCl and are used to make extracellular recordings of action potentials at a depth of 700-750 μm in cortex (Reinecke, Dinse, Reinke & Witte, 2003). The research is often done with the aid of animals and while studying cortical plasticity (discussed in the next section) it is unfortunately very often necessary to harm the experimental animal by destroying some of its channels for sensory input or create lesions in the brain. However, in the last decade another method called TMS (transcranial magnetic stimulation) has been increasingly used to create virtual lesions, by disrupting the neuronal activity in cortical areas (Siebner & Rothwell, 2002).

2.2 Cortical Plasticity

The brain is continually reorganizing to adapt to the environment of its owner. This is confirmed in studies carried out by Kilgard, Pandya, Engineer and Moucha (2002) that suggest that sensory activation alter the functional organization of cortex to generate the most useful representation of the sensory environment.

In the developing brain, the plasticity is very extensive. This was shown, for example, in a study made by Kahn and Krubitzer (2002). In their study,

the visual input to the brain was destroyed in some pups, very early in life, before the establishment of the retino-geniculo-cortical pathway. This resulted in a complete capture of the visually processing regions by the somatosensory and the auditory systems. One plausible conclusion following from this is that the peripheral innervation plays a very large role in the organization of cortex.

According to Florence, Jain and Kaas (1997) the degree of plasticity is less in the adult brain than in the developing brain. Further evidence for this view is provided by Sawaki, Yaseen, Kopylev and Cohen (2003). They found that the ability of the motor cortex to reorganize decreased with age.

However, the cortical maps in the adult brain are not static. Instead, according to Buonomano and Merzenich (1998), they undergo plastic changes due to peripheral manipulations and experiences throughout the whole life.

The reorganization of cortex is probably not a phenomenon isolated to cortex, but depends on subcortical structures as well. Cortex seems to reorganize its representations constantly. The reorganization is dependent on the importance of the stimuli, i.e. important stimuli facilitate plasticity while unimportant are ignored. Little is known about the mechanism behind this, but it has been suggested by Kilgard and Merzenich (1998) that there are neuromodulatory systems that mark important stimuli. They stimulated nucleus basalis in adult rats electrically, and at the same time exposed them to auditory stimuli. This resulted in a massive reorganization of primary auditory cortex and thus they suggest that nucleus basalis may have a neuromodulatory function.

Currently it is believed that the neural mechanism behind cortical representational remodelling is based on synaptic plasticity mainly depending on what is called LTP (long-term potentiation) of excitatory synapses following a Hebbian learning rule (Buonomano & Merzenich, 1998). At a synaptic level this means an increasing strength between synapses which fires together. At a higher neuronal level, this causes the formation of topographic maps.

Florence, Jain and Kaas (1997) carried out experiments with monkeys, in which they studied the cortical reorganization following different forms of modifications of the input to the brain. In one study they made a transection of the median nerve in the arm of an owl monkey. Such a transection causes permanent denervation of the radial part (thumb side) of the palm. In the normal monkey, area 3b in somatosensory cortex represents the hand. The representations of the palm

and the fingers are large, while the representation of the dorsal surface of the hand is small. Immediately after the transection, the deprived zone in area 3b was activated by input from the dorsal surface of the hand. Over time, even more of the deprived zone becomes activated by the dorsal surface of the hand, the palm and adjacent fingers.

A possible explanation for the immediate changes of representations may be that the afferent axons from thalamus to L-IV in area 3b are believed to have a wider projection zone than what is reflected in the topographic map in cortex during normal conditions (Florence, Jain & Kaas, 1997), i.e. the axons fringes into the regions of neighbouring representations. If the level of inhibition is reduced due to trauma, the representations will expand into the deprived zone.

An irritating and sometimes painful phenomenon that probably is caused by unwanted subcortical reorganization is phantom limbs (Ranadive, 1997). An experiment described by Florence, Jain and Kaas (1997) explains why. In the experiment one hand of a macaque monkey was amputated. The arm and the hand are represented in a subcortical structure called cuneate nucleus with the hand occupying all but the most dorsal extreme. After the amputation of the hand the arm representation expands into the deprived zone. The axons from this area are connected with the somatosensory cortical area that represents the hand and inputs from these axons are interpreted as hand sensations, so a tactile stimulation of the arm will now be experienced as both a touch on the arm and on the amputated hand.

2.3 Cortical Induction

Patients with nerve injury in their arm might end up with the nerve fibres completely misconnected after the healing process. To help these patients an urgent question is how to optimize the rehabilitation, to make the cortical maps reorganize as well as possible. One question then, is if there are some kinds of re-education programs that can be used for this purpose. And there are, because studies have shown, for example, that an early use of tactile stimulation after nerve repair will give a significantly better discriminative sensibility (Rosen, Balkenius & Lundborg, 2003).

If the arm nerve is only partially injured the patient might end up in a situation with one part of the hand correctly mapped and the other part incorrectly mapped in somatosensory cortex. To rehabilitate such patients a method for sensory re-education that tries to extend the map for the intact part of the hand has been suggested. This method is

called cortical induction (Rosen, Balkenius & Lundborg, 2003).

As mentioned, cortical induction presumes that not all nerve fibres have been injured, but that a part of the skin area on the palm is correctly mapped in the cortex. The training method is based on the ability of sensory cortex to expand the part of the map corresponding to a frequent stimulus, and its ability to differentiate between two different stimuli that are alternately presented.

By simultaneously stimulating two points along the border between the skin area with a correct mapping and the skin area without a correct mapping in cortex, where one point is located on one side of the border and the other point on the other side, until the patient cannot any longer discriminate between stimulation of the two points, the intact part of the cortical map will expand to include the point in the incorrectly mapped skin area. When the points cannot be discriminated any longer, the two points are stimulated alternately until the patient can discriminate them again. Then a local expansion of the correctly mapped cortical map has been obtained. By repeating these two phases along the border of the intact part of the cortical map, the border will gradually move forward to include more and more of the incorrect part, until the whole cortical map is correct again.

One can say that the training method cortical induction uses the intact topographical mapping as a seed to grow a new topographical mapping in the incorrectly mapped region.

2.4 Modelling

In neural modelling, there are two distinctive motifs (Kohonen, 2001). One is to on the basis of biological inspiration develop new useful technical devices with intelligent properties. The other motif is to describe and get an understanding of biological neural systems. Often the circuits designed for both of these purposes are artificial neural networks. It is the second of the motifs above that lies behind this project.

In modern neurophysiology, each neuron is described as a complex dynamical system. It is controlled by neural signals, chemical transmitters, messenger molecules etc. So if one would try to take into account every known fact about the biological neural system in a model it would demand more computer power than is available to a researcher today. For example, to describe the global meteorological system would be trivial compared with a complete description of the brain.

Fortunately, it is not be necessary to model every detail of the biological neural system to learn anything useful from the model. It is only necessary to model relevant parts of the system. So the only problem is to know what parts are relevant, but this is the same problem that a theoretical scientist will meet while deciding upon what to include in a theory, or an experimental scientist while deciding on the design of an experiment. It is a matter of judgement and experience.

There are many types of artificial neural networks, but in principle, according to the descriptions given in Russel and Norvig (1995), they work as follows: An artificial neural network consists of a set of nodes, or units, that are connected to each other by links. For each link there is a weight, i.e. a number. Some of the nodes are input nodes and some are output nodes. Every node carries an activation value and when one wants to present the network for input, the input has to be translated into a form that is suited for the network. When that is done the input is presented to the network by activating its input nodes, i.e. for each node assign a certain value to its activation. The activation for the other nodes in the network is calculated. This is done by taking the activation for each node that has an incoming connection, multiply it by the weight associated with the connection and then sum all these products.
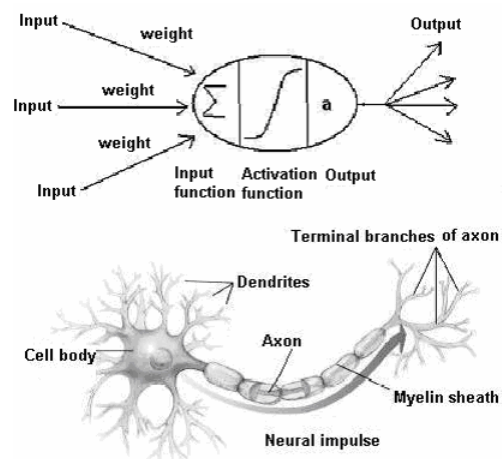

Fig 2.2. Uppermost a node in an artificial neural network. At the bottom, a neuron (http://www.arts.uwaterloo.ca/~sspencer/psych 101/lecture3/lecture3.pdf, 2004-05-16). In a biological neural network the neurons takes its input signals from its dendrites, processes the input in a way corresponding to the activation function in an artificial neural network and then conveys the resulting signal through its axon. The synapses in a biological neural network correspond to the weights in an artificial neural network.

This sum of products is then transformed by a non-linear activation function that gives the activation for the node. Finally the activation in the output nodes is calculated and this constitutes the output. Perhaps the output has to be translated to a proper form afterwards. An artificial neural network learns by adjusting its weights in accordance with some learning algorithm. If the artificial neural network needs the correct output corresponding to the input during learning, it is called a supervised artificial neural network. If it does not need this, it is called an unsupervised artificial neural network.

2.6 Modelling Cortical Plasticity

A growing amount of cortical plasticity modelling has been done. For example the Slovak researcher Lubica Benuskova and colleges have accomplished five studies on the plasticity of the somatic sensory ("barrel") cortex in rats (Benuskova, Diamond & Ebner, 1994; Benuskova, 1995; Benuskova, Diamond & Ebner, 1996; Benuskova, Diamond, Ebner & Armstrong-James, 1999; Benuskova, Rema, Armstrong-James & Ebner, 2001). These studies have been done by creating computer models of parts of a barrel column in the cortex of rats and then these models have been compared to electrophysiological experimental observations. The electrophysiological measurements were made on a single barrel column corresponding to one of the whiskers after normal experience and after experience with all but two adjacent whiskers trimmed. The trimming of all but two whiskers causes a change in the sensory input, which in turn is responsible for the reorganization of the barrel cortex. All models in these studies were based on the Bienenstock, Cooper and Munro (BCM) theory of synaptic plasticity. In short, the hallmark of the BCM theory is the dynamic synaptic modification threshold. Whether a neuron's activity will lead to weakening or strengthening of its impinging synapses is dictated by this threshold. The threshold is proportional to the averaged (over some recent past) activity of the neuron. In the first study, only one neuron in a barrel column was modelled. In the following studies, the model was gradually extended by including the interaction of inhibitory and excitatory interneurons in one barrel column, and in the last few a large part of one barrel column was included. The last study also included modelling of prenatal exposure to alcohol and it was confirmed in accordance with experimental results that such exposure impaired plasticity in barrel cortex. Generally, the behaviour of the models was consistent with the electrophysiological observations.

One model called LISSOM (Laterally Inter-connected Synergetically Self-Organizing Map),

described by Sirosh and Miikkulainen (1994), has been used to model cortical lesion plasticity. This model is a variant of the basic self-organizing map described below in the next section. The basic difference is that another rule is used to update the weights. From the results of the simulation with the LISSOM model, two suggestions on techniques for enhancement of recovery from surgery in the sensory cortices emanated. Namely, that blocking of lateral inhibition locally in cortex after surgery, and a forced cortical reorganization before the surgery, will speed up the recovery.

Goodall, Reggia, Chen, Ruppin, and Whitney (1996) studied the effects of sudden focal lesions of varying size to the cortex. In their study they used a model of primary sensorymotor cortex controlling the position of a simulated arm.

Wiemer, Spengler, Joublin, Stagge and Wacquant (1998) designed a model of early sensory cortex in which not only a topological, but also a temporal, representation of the input from the external world was taken into account. This was done because according to neurobiological experiments on early sensory cortex, the temporal distance between stimuli appears to be transformed into spatial distances in the cortical representation. Their simulations agreed with the experimental results.

Kalarickal and Marshall (2002) presented a study in which they used a computational model of somatosensory cortex and thalamus to model changes in receptive fields after intracortical microstimulation (ICMS) and peripheral stimulation. The model consists of a two level architecture with lateral inhibition and afferent excitation. In the simulations topographical changes of the receptive fields similar to those experimentally observed occurred. The simulations with the model also generated new and testable predictions.

Some computational models that includes the hand and the somatosensory cortex have been made. For example a model made by Mazza and Roque-da-Silva (1999). Their model includes tactile receptors for a hand connected to a cortical layer, consisting of inhibitory and excitatory neurons, corresponding to L-IV in area 3b. The purpose of the model was to study reorganization of somatotopic maps. They first ran a simulation so that a cortical map corresponding to an intact hand was formed. They then simulated a fusion of two digits and the cortical map was reorganized. After that they simulated a separation of the digits again and the cortical map was again reorganized. Even simulations corresponding to removal and re-implantation of a digit were run. The results showed that the model was capable to transform the

somatotopic map of the hand in cortex in a way corresponding to empirical observations in primates.

In another study, they investigated how cortical representations are affected by cortical lesions (Mazza & Roque-da-Silva, 2000). Cortical lesions may namely imply unbalances in the organization of the somatotopic maps. This study was made by the aid of a computational model of the somatosensory system. The model consisted of 512 tactile receptors that represented a palm connected to 1024 excitatory and 512 inhibitory neurons in a cortical layer corresponding to L-IV in area 3b. In accordance with experimental data the simulations reproduced the formation and maintenance of somatotopic maps and the reorganization of the maps after destruction of neurons.

A study made by Lin, Liljeholm, Ozdzynski och Beatty (2002) investigates what is called the neuronal empiricism hypothesis. This hypothesis is the idea that all cortical regions uses the same fundamental computational algorithm, and that cortical neurons learns the characteristic properties of their receptive fields by extracting statistical regularities from the input-streams they are processing. In their study, they used, Kohonen's self-organizing map algorithm together with simulated input from the hand. They model the organization of the neurons in area 3b as a response to stimulation of the ventral side of the hand. Simulations of a normal hand, followed by simulations of a hand with an amputated digit were carried out. In the simulation with an amputated digit, the cortical map changed dramatically so that all cortical representations of the missing digit disappeared. The result from the study suggests that cortical neurons learn the properties and organization of their receptive field by statistical regularities in the stream of information they receive.

2.7 Kohonen's Self-Organizing Map

To create a proper model of certain cortical regions we need neural networks that are able to learn without the aid of a supervisor, i.e. an unsupervised neural network. The kind of learning in this category of neural networks is usually called competitive, or self-organizing. A self-organizing map (Kohonen, 1990; Kohonen, 2001), abbreviated SOM, is an instance of this category.

In a competitive learning network each node receives the same input. The nodes influence each other by lateral interactions that are both inhibiting and exciting. The winner, i.e. the node which is most activated is allowed to learn, i.e. update its weights, and in addition the nodes in a vicinity of the winner node are also allowed to update its weights, but to a lesser degree. To what degree the nodes are allowed to update its weights depends on the distance to the winner node. In the long run this will lead to self-organization of the nodes in the network so that every node will be sensitized to a certain domain of the input space, and the network will also conserve the order of the input space.

Formally one can describe the SOM as a non-linear, ordered and smooth mapping from a manifold of a perhaps high-dimensional input space onto a low-dimensional array. In fact, it is a non-linear projection of the probability density function $p(x)$, $x \in \mathbf{R}^n$ onto the array of nodes, which means that if the input from certain parts of the input space are more frequent or there is a greater probability that the input samples belongs to these parts, then this will be reflected in the map in such a way that the corresponding parts in the map will allocate a larger area. In most cases, the SOM are arranged as a two-dimensional array of nodes and the input is connected to every node in the network (Wedel & Polani, 1996). A model (reference) vector $m_i = [\mu_{i1}, \mu_{i2}, \ldots, \mu_{in}]^\mathrm{T} \in \mathbf{R}^n$, later called a weight vector, is associated with every node i in the network, where it is assumed that the input vector $x \in \mathbf{R}^n$.

If the SOM is implemented in software, there is no need to find the magnitude of the node's activation. It is enough to check which node has the weight vector that is closest to the input vector in some metric and then map the input onto that node.

It is common to use the Euclidian distance to find the weight vector with the closest match to the input vector. Another common and more "biological" method is based on the dot product of the input vector $x \in \mathbf{R}^n$ and $m_i \in \mathbf{R}^n$. These two variants of the SOM algorithm will be described below.

The SOM algorithm using the Euclidian distance: For every node i there is a weight vector $m_i \in \mathbf{R}^n$. At time step t an input signal $x \in \mathbf{R}^n$ activates a winner node c for which the following is valid:

$$\| m_c - x \| \leq \| m_i - x \| \ \forall i,$$

i.e. c is the node which weight vector best approximates the input signal. Every node updates its weights according to Kohonen's learning rule:

$$m_i(t + 1) = m_i(t) + h_{ci}(t) \cdot [x(t) - m_i(t)],$$

where h is a neighbourhood function, which is used in the following way: the learning is supposed to depend on where a node is located in relation to the

winner node c. For the neighbourhood function h the following must be valid: $h_{ci}(t) \to 0$ when $t \to \infty$. Otherwise the process will not converge. Often $h_{ci}(t) = h(\|r_c - r_i\|, t)$, where $r_c \in \mathbf{R}^2$ and $r_i \in \mathbf{R}^2$ are location vectors of node c and node i in the array, i.e. the location on the cortical surface. When $\|r_c - r_i\|$ increases $h_{ci} \to 0$. h might be defined in terms of the Gaussian function:

$$h_{ci}(t) = \alpha(t) \cdot \exp(- \|r_c - r_i\|^2 / 2\sigma^2(t)),$$

where $\alpha(t)$ is a scalar-valued learning-rate factor, and $\sigma(t)$ corresponds to the radius of the neighbourhood. $\alpha(t)$ and $\sigma(t)$ are monotonic and decreasing functions of t.

The dot product SOM algorithm: Instead of checking the Euclidian distance between the input vector and the weight vectors, the dot product of the vectors can be used to get the best match. The difference between the variant described above and this variant is that to the winner node, one calculates the dot product between the input vector and each weight vector. For the winner node c the following is valid:

$$X^T(t)m_c(t) \geq X^T(t)m_i(t) \ \forall i,$$

The weights are updated according to:

$$m_i(t + 1) = [m_i(t) + \alpha'(t)x(t)] / \| m_i(t) + \alpha'(t)x(t) \|$$
if $i \in N_c(t)$

and

$$m_i(t + 1) = m_i(t) \text{ if } i \notin N_c(t)$$

where $N_c(t)$ is the neighbourhood of the node c at time t and $0 < \alpha'(t) < \infty$.

Before processing, the weights have to be initialized. This can be done by assigning them randomly, but if they are initialized with some kind of order the process will converge much more rapidly.

In a general way, the adaptive SOM processes can explain the organization found in varying structures in the brain and artificial self-organizing maps share many features with brain maps, i.e. the experimentally deduced topographical organization of the brain.

3. Methods

To investigate the influence from different forms of sensory training on the reorganization of the maps in somatosensory cortex I have built a series of models that consist of the relevant neural systems.

These systems are: a part of a dorsal root ganglion that takes input from a palm, a part of medulla, a part of thalamus, and a part of primary somatosensory cortex. I have been working in steps in this project. For each step I have extended the current model to a more advanced one. Simulations have been done with the resulting models in each step. In the models, the number of neurons in each relay nucleus has been kept constant. This is not because it is realistic, but because I could not find any information about the relations between the numbers of neurons in each site. Anyway, these relations are not important for the simulation results. The number of neurons in each relay site, and the number of receptors in the palm, has been 400, and in the cortex module the number of neurons has been 900. This is true for all versions of the model. In the models, it is possible to simulate the misconnection of nerve fibres between the hand and the dorsal root ganglion that follows nerve regeneration.

When creating computational models, it would be a great idea to stick to some kind of standard to enable others to benefit from it. This is exactly what has been done in this project since it is a subproject of Ikaros (Balkenius, 2003). Ikaros is a project that aims to develop an infrastructure that can be used for system level modelling of the brain. In addition, it keeps databases with computational models and experimental and functional brain data. Ikaros provides the programmer with an environment which aims to be a standard method for system level modelling of the brain. In Ikaros, there are a collection of models corresponding to different brain regions. There are also modules working as drivers for I/O. Ikaros has a plug-in architecture and that enables the programmer to add new models in an easy way.

During the simulations the quality of the map formed in the cortex module was measured. The method for map quality measurement used in these simulations is based on a method described in Teuvo Kohonen's book Self-Organizing Maps (Kohonen, 2001). My slightly modified version of this method works as follows: Randomly select n input samples from the input space and activate the model with them. For each of them, check which neuron in the cortex module has the best matching weight vector and which neuron has the second best matching weight vector. Count the number of cases in which the neuron with the best matching weight vector and the neuron with the second best matching weight vector are neighbours in the map. Divide the resulting sum with n to make the measure independent of the size of the input space. The resulting number is a measure of the quality of the map, i.e. how well organized the cortical map is. In the simulations with all models n = 800. The

choice of n is based on a method for selection dimensioning found in the literature (Klefsjö & Hellmer, 1991).

This method of quality measurement has the benefit of providing a numerical value signifying the quality of the cortical map for each training method before, after and during the learning phase. A training method with a higher value on this quality measure is considered better than a training method with a lower value, provided that both are trained with an equal amount of iterations.

A number of training methods, i.e. different methods for input generation, where created, and simulations were done with the models with a set, or a subset, of these training methods. There were 30 simulations with each training method used for each model. This was necessary because some of the training methods depended on a random function and in the quality measurements only randomly selected subsets of the input space were used. Due to this, the simulations were not deterministic and to get significant results, simulations had to be repeated several times. With 30 simulations or more nothing has to be assumed about distribution. With all versions of the model besides the model of partial nerve injury, see chapter 6, simulations with all training methods described in Fig. 3.1 were done. The training methods used with the model of partial nerve injury are described in chapter 6.

In the simulations, quality measurements were done a number of times during the reorganization of the cortex map after the mixing of the nerve fibres. The simulations started with a well organized map, i.e. well organized according to the state of the model before nerve fibre mixing. This was accomplished by reading in a set of weights to the cortex module in the beginning of each simulation. These weights were generated by a simulation with training method 1, see Fig. 3.1, and the reason to choose this particular training method for this purpose was because the organization of the relevant cortical areas for touch in reality should be formed by all kinds of constellations of stimuli, at least approximately, and what training method 1 provides is an approximation of that. By reading in the weights, it was possible to turn on mixing at the very beginning of the simulations, which saved a lot of computer time.

However, the main reason for starting the simulation by reading in weights corresponding to a well organized map was to allow every training method to start with the model in an identical state. This is important since the differences in map quality should depend on the differences of the training methods only.

Training Method	Description
1	In each iteration, a random number of points on the palm are stimulated with a random intensity.
2	In each iteration, one point on the palm is stimulated, in such an order that every point is stimulated from left to right, row by row in the same manner as one reads a text in English. When the last point on the last row has been stimulated, the same process starts over.
3	In each iteration, one point on the palm is stimulated, and which point is selected is chosen at random.
4	In each iteration, two points on the palm are stimulated, one on the edge of the palm, in a clockwise order, and starting with the first element in the first row and one in the same order as in training method 2, but with the edge left out.
5	In each iteration, two points on the palm are stimulated, and which points are selected is chosen at random.

Fig. 3.1. The table describes all training methods used in the simulations, besides cortical induction.

The simulations results were analyzed statistically. Confidence intervals for the map quality were calculated with a confidence level of 99% (p = 0.01), and presented as diagrams.

The simulations started with the nerve fibres from the palm mixed, i.e. a randomly set map between the output of the palm and the input of the medulla was turned on. In the simulations with the model of partial nerve injury, the nerve fibres were only partly mixed. Besides the model of partial nerve injury, every simulation consisted of a training period with 2000 iterations with one of the training methods. The map quality was measured, with the method described above, before the training period and then every 500 iterations. The specifics of the simulations with the model of partial nerve injury are described in chapter 6.

4. A Basic SOM Model

For a basic SOM model four Ikaros modules were implemented: NeuralField2D, SOM, Mixer and StimGen. These modules are described below.

11

NeuralField2D: This module describes a neural field, i.e. a collection of neurons that interacts with each other by inhibition and excitation. The neurons are represented by a matrix. This matrix, which is also the modules output, describes the activation of each neuron. As input the module also takes a matrix. Depending on the input matrix, the elements in the output matrix are updated. This is done in such a way that the input matrix is projected onto the output matrix and each element in the output matrix is influenced by the activations of the projected elements in the input matrix. The level of influence of the elements in the projected input matrix on an element in the output matrix is determined by their distance to the element in question and their activation level. The elements of the projected input matrix have both inhibitory and excitatory influence on the elements of the output matrix. In practice, this is done by using a Gaussian function for both inhibition and excitation, but with one sigma value for inhibition and another for excitation. So, when the activation level for an element in the output matrix is calculated, it is done by projecting the elements of the input matrix onto the output matrix, and for each of them calculate the distance to the element in the output matrix. This distance is then inserted into two Gaussian functions, one for inhibition and one for excitation. The values of the Gaussian functions are multiplied with the activation level of the input matrix element which projection is considered. These products are calculated for each element in the input matrix and summed. So, in the end, there are to sums, one for inhibition and one for excitation and these are normalized. Inhibition is then subtracted from excitation and if the result is positive, the activation of the output matrix element is set to this value, otherwise it is set to zero. The operation of the module is equivalent to a neural field with inhibitory and excitatory interactions, but without learning.

In the model, there were three instances of this module. The palm and the dorsal root ganglion were modelled with an instance of it, but with inhibition turned of. This is not strictly authentic when compared with the neurophysiological system which it intends to model, but in the dorsal root ganglion the neurons do not inhibit or excite each other, but their corresponding receptors in the palm have a receptive field with some spatial extension and these overlaps. Therefore a stimulus on a point on the palm will excite a collection of neurons in the dorsal root ganglion and the excitation level of a dorsal root ganglion neuron is determined by where in its receptive field the stimulus is applied, with the excitation largest if the stimulus is applied in the centre of the receptive field and with a decline when it deviates to its borders. So, when a point is stimulated on the palm, a group of adjacent neurons

in the dorsal root ganglion will be excited, since the stimulus is applied inside their receptive fields. For neurons close to the centre of the group, the excitation are large and the excitation for neurons declines with the distance from the centre of the group. The group consists of adjacent neurons, because the axons from the palm conserve the somatotopic organization in the dorsal root ganglion. To model this part of the system with an instance of NeuralField2D is not to far from the truth since it should behave in a similar way.

The relay nuclei in the medulla and the thalamus were also modelled with instances of Neural-Field2D. Both inhibition and excitation were turned on.

Somatosensory cortex was modelled by another module, SOM. This module is an implementation of the self-organizing map described in section 2.6 above. In the simulations, the dot-product algorithm was used and the shape of the neighbourhood was Gaussian. This module also contains functionality for measurements of the map quality.

In addition to NeuralField2D and SOM, there were two other modules used in the simulations. One of them was Mixer, which creates a random map between input and output. It takes a matrix as input an also outputs a matrix. This mapping can be turned on after a specified number of iterations. The other module was StimGen, which generates input matrices to the palm according to different training methods. In the simulations, the module instances were lined up as shown in Fig. 4.1.
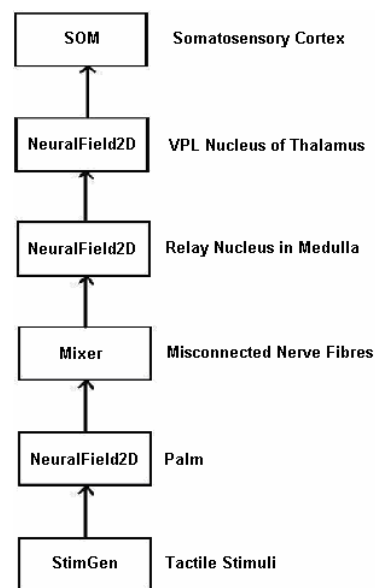

Fig. 4.1. The figure shows the modules in the basic SOM model and their biological counterparts.

12

This model was the first one and it was also the simplest. The main purpose of building it was to get a working model up and running.

In Fig. 4.2 the results from the simulations with the basic SOM model are presented as a diagram, and in Fig. 4.3 momentary depictions from a couple of simulations with the basic SOM model are shown. This is to illustrate the activation at different sites along the pathway.

The results from these simulations shows that the training methods used did not differ from each other very much. The basic SOM seems to organize well with any method and it seems to be a far too simplistic model to be really interesting. However, the purpose of this version of the model was first and foremost to get a working model of the relevant neurophysiological systems up and running.
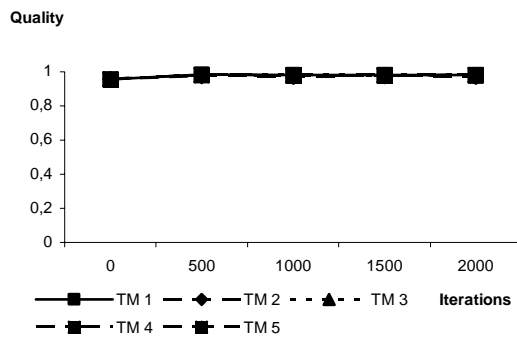

Fig. 4.2. The results of the simulations with the basic SOM model presented as a diagram. The confidence interval for every training method at every point of measurement is smaller than $\pm$ 0.004 (p = 0.01). As we can see in the diagram, the training methods are in practice equally good.

5. A Multiple Peak Model

It would be more natural if the learning in the map when several points on the palm are stimulated would have an influence on its ability to map the same points separately too. To take this into account, the SOM module in the basic SOM model was modified.

The SOM module in this modified model differs from the basic SOM in that the learning, i.e. the updating of the weights belonging to the neurons in the map does not only depend on the winner neuron. In the basic SOM, how much the weights belonging to a certain node will be updated, depends on the distance to the winner node. In this version of the model, how much a nodes weights will be updated, depends on all nodes. For every node in the network, an influence from each node is calculated in a similar way as from the winner node in the case of the basic SOM. The results of the calculations are multiplied by the level of activation of the influencing nodes, so that the level of influence on a node by another node depends on the level of activation, i.e. the degree of matching of the nodes weights to the input vector, for the other node. To calculate the influence on a certain node in the network, the influence from all the nodes in the network are normalized by dividing with the largest influence and then the normalized influences are summarized. The resulting sum is taken as the neighbourhood function h described in section 2.4 above. The difference is that in this modified SOM, the neighbourhood function h describes not only how to update a nodes weight depending on its distance to the winner node, but how to update the nodes weights depends on its distance to all the other nodes and their levels of activation.
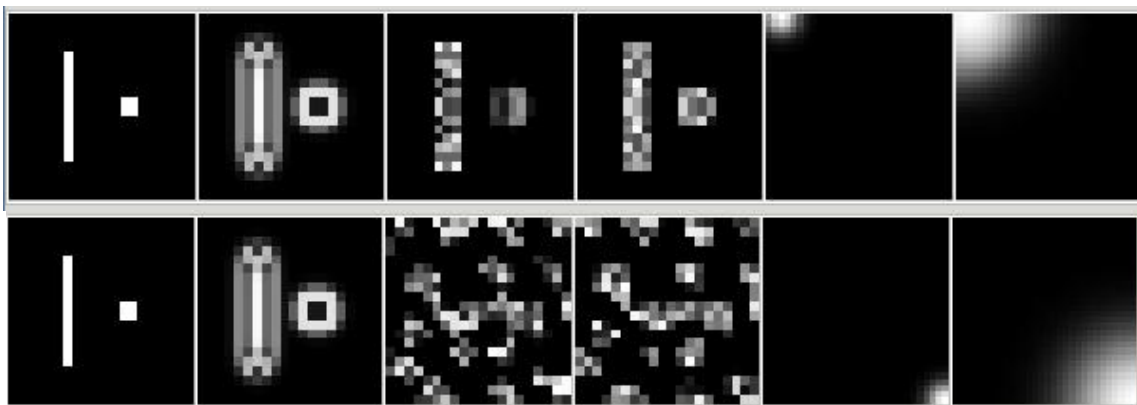

Fig. 4.3. The figure shows the activation at different levels at a moment in a simulation with the basic SOM model. In the uppermost sequence of frames there is no nerve fibre mixing, while in the sequence at the bottom there is. From left to right the frames show: the applied stimuli, the palm, the relay nucleus in medulla, the relay nucleus in thalamus, the winner neuron (and a small vicinity) in cortex, and the activation in cortex.

The simulations with this model, which is somewhat more realistic than the basic SOM model, generated results, see Fig. 5.1, that were more interesting.
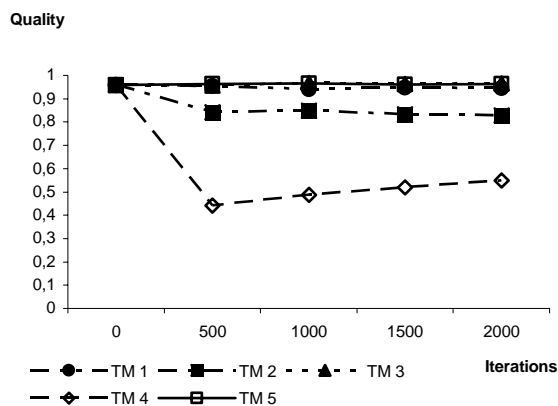

Fig. 5.1. The results of the simulations with the multiple peak model. The confidence interval for every training method at every point of measurement is at most ± 0.06 (p = 0.01). Training methods 1, 3, and 5 are good, but training method 2 is worse and training method 4 is far worse than the others.

Training methods 1, 3, and 5 reorganized the cortical map fast and with a high quality, and the quality remained high at a constant level during the simulations. Training methods 2 and 4 were worse with training method 2 stabilizing at a lower level than 1, 3, and 5. Training method 4 decreased the map quality very much during the first 500 iterations, but then the quality started to improve slowly but steadily.

Training methods 2 and 4 have in common that they are stimulating the palm in an orderly manner, i.e. they are not based on any random function. However the training method 4 is much worse than training method 2, at least during the first 2000 iterations.

The common feature of training methods 1, 3 and 5 is that they are all based on a random selection of stimuli points. These results may be taken as at least a weak possibility that the training methods used for optimization of the recovery after nerve regeneration should be based on random stimuli. But as we will se in later models, this is not always the case.

6. A Model of Partial Nerve Injury

To enable a simulation of those cases where only a part of the nerve in the arm was injured, the multiple peak model was modified. One thing that was modified was the module Mixer. This was done to allow mixing of only a part of the nerve fibres. When the nerve injury is only partial there are reasons to believe that a certain training method called cortical induction, described above in section 2.3, would be of great value. Therefore, in addition to the changes in the module Mixer, a new training method was implemented in the module StimGen.

Actually, first an accurate version of this training method was implemented, but there were problems to get it to work with a model with a very limited number of neurons. It was necessary to keep the number of neurons limited, because otherwise the model would have been too computationally expensive. Another problem was that it would have been necessary to run quite a lot of iterations in a simulation, because of the nature of cortical induction, i.e. it repeats the stimuli of a couple of points many times before proceeding. I implemented and tested an accurate implementation of cortical induction and a rough estimation is that it would take approximately 12000 iterations in each simulation with a model of the same size as the previous models, so it would take a lot of computer time to run the simulations. However, this was not the main reason to abandon it. The main reason, instead, was that I run into strange troubles with the implementation where the simulation unexpectedly hanged and the stimulation of the edge didn't proceed anymore after several thousands of iterations, for some reason. Since it would take about 12000 iterations, only to test any changes to the implementation, which means about 8 hours waiting for the computer, it would not be reasonable to continue.

Therefore, to get some idea of the usefulness of cortical induction, I created a simplified version of it. This version of cortical induction is presented below:

Training Method Cortical Induction: This function implements a simplified version of cortical induction. It works by randomly select one point on the edge of the part of the palm corresponding to the well organized part of cortex and one point in the same column but one row further down into the part of the palm that corresponds to the part of cortex that is not well organized yet. This is done as many times as there are points along the edge, then for each iteration one point on the edge is selected randomly. This is done as many times as there are points along the edge, then for each iteration one point on the row one step below the edge is selected randomly. This is also done as many times as there are points along the edge, then the edge is moved one row further down and the process above starts

over. This continues until the edge is moved to the last row.

The simulations started with the nerve fibres for half the palm mixed, and as in the previous models weights corresponding to a well organized cortical map before mixing were read in. The map quality was measured before and after the learning phase which lasted for 600 iterations. The reason to run precisely 600 iterations was that that was exactly the number of iterations needed to let all of the part of the palm with mixed nerve fibres be stimulated with the training method cortical induction, and to be comparable each of the other training methods should be run with equally many iterations.

The training methods used in the simulations with this model, were training methods 1, 3 and 5 used in the previous model, and in addition the new training method "cortical induction". For each of these training methods only the nerve fibres for half the palm were mixed. The reason to choose training methods 1, 3, and 5 was because they were good in the previous model.

In Fig. 6.1 the results from simulations with the model of partial nerve injury are shown in a diagram, and in Fig. 6.2 momentary depictions from a simulation with the model is shown to illustrate the activation at different sites along the pathway.

In this simplified version of cortical induction, there was no feedback from the SOM module providing information about to what degree the correct part of the cortical map had expanded locally around the edge. Therefore it is possible that a different amount of stimulation around the edge before putting it forward, would give a better result for the training method cortical induction. However, the ability to reorganize the cortical map is already very
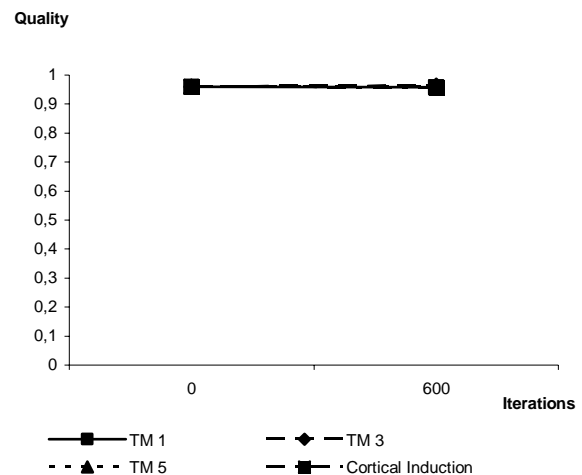

Fig. 6.1. The results of the simulations with the model of partial nerve injury, presented as a diagram. There was no significant difference (p = 0.01) in map quality, for any training method, after 600 iterations compared with the map quality after 0 iterations. Neither was there any significant difference (p = 0.01) in map quality after 600 iterations, when the training methods were compared to each other.

good for the training method cortical induction as is the case for the other training methods used in the simulations with this model. Therefore, I suspect that this model is far too simple to indicate the usefulness of cortical induction. Maybe simulations of partial nerve injury, using a later version of the model (described below), for example with the leaky integrator and with disturbances in the timing of signals would give more interesting results enabling me to provide a better statement on this.
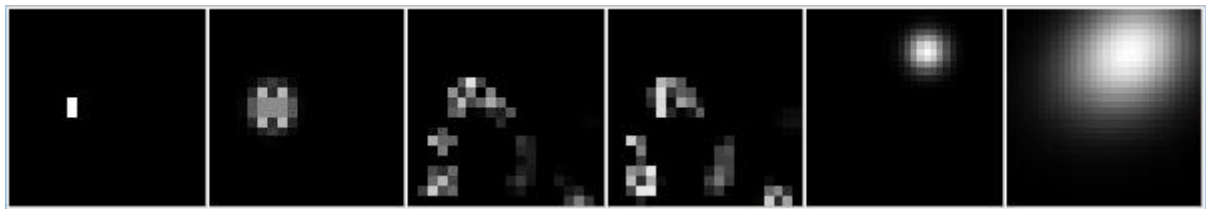

Fig. 6.2. The figure shows the activation at different levels at a moment in a simulation with cortical induction and the nerve fibres corresponding to the lower part of the palm mixed. From left to right the frames show: the applied stimuli, the palm, the relay nucleus in medulla, the relay nucleus in thalamus, the winner neuron (and a small vicinity) in cortex, and the activation in cortex.

7. A Leaky Integrator Model

One way to extend the multiple peak model to be more consistent with, and more accurately model, the operation of the neurons in the real cortex is to let the activation in the cortex module linger for some time. This could be done by partly add the activation from the previous iteration to the current activation. This type of neuron models is called a leaky integrator (Kohonen, 2001).

To implement a leaky integrator some functionality was added to the SOM module in the multiple peak model. In the SOM module the activations from the previous iteration are added to the current iteration's activations, but first the previous activations are multiplied with a factor which expresses the activation decay. As a result the activations in an iteration will not disappear immediately after this iteration, but decline smoothly during a number of future iterations. For example with an activation decay set to 0.5, as in my simulations with this model, the activations from the previous iteration will still be present but with the strength reduced to 50% of its initial value. Activations 7 iterations behind the current one will still be present, but their strength will now be reduced to less than 1% of their initial values. The simulation results with this version of the model are presented below in Fig. 7.1.

In the simulations with this model, training methods 3 and 5 still seems to be good as in the multiple peak model, but training method 5 has deteriorated a bit.
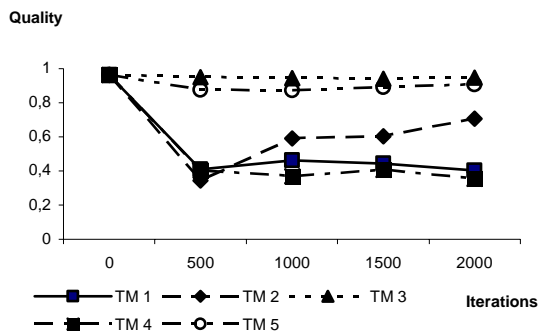
Quality

Fig. 7.1. The results of the simulations with the leaky integrator model, presented as a diagram. The confidence interval for every training method at every point of measurement is at most ± 0.10 (p = 0.01). Training method 3 is good, and training method 5 is probably a bit worse but rather good. The others are worse but training method 2 seems to improve steadily after 500 iterations.

But training method 1 is not good at all anymore. This effect is probably due to that the lingering activation has a smoothening effect when this particular training method is used, so that in every iteration cortex tends to be more homogeneously activated no matter what stimuli of the kind generated with training method 1 is put into the model. This smoothening effect might arise because in every iteration the main part of the palm is stimulated more or less, and this means that in the next iteration there will be lingering activation corresponding to an input vector of the same type as the current one, but with another activation pattern. Since the degree of stimulation at every point in training method 1 is randomly chosen to a value between 0 and 5, the effect should be that the activation of cortex in every iteration corresponds to a subset of the input space that are in average more homogeneously activated in every point than the input space itself. Therefore cortex is probably very homogeneously activated and this changes the cortical map a lot, in a bad direction, since the map is a projection of the probability density function of the input space.

Training method 2 has deteriorated too after 500 iterations, compared with the multiple peak model, but then it improves with the number of iterations, and I think it is likely that it would reach the same level of quality in the cortical organization if more iterations were run.

Training method 4 is as bad as training method 1 in this model, but it is a bit more difficult to explain why it doesn't improves again after the dip at 500 iterations as it did in the multiple peak model. Instead it seems to remain at a low level of quality.

8. Sensitivity to Timing of the Signals

I had the idea that maybe after a severe nerve injury there might be disturbances in the timing of signals. This idea was founded in two reasons. First, as mentioned in section 2.1, the signals from pain receptors are conveyed at a slower pace than the signals from touch receptors. Second, when the nerve fibres grows together again after a nerve injury, there are fibres belonging to pain receptors that grows together with fibres belonging to touch receptors and vice versa. Together this means that some receptors that were previously conveying pain signals will now convey touch signals and it is quite plausible that the transfer of the signal from the palm to the medulla will then take a little longer, since the part of the nerve fibre previously conveying pain signals is less myelinated than is normal for a nerve fibre that conveys touch signals. To test this, the leaky integrator model was extended so that the conveying of the signals from

every other receptor took a little longer to reach medulla. This has been implemented by the use of three more modules than in the leaky integrator model. After the module StimGen there is a new module Separate which separates the incoming matrix into two output matrices of equal size as the input matrix. This is done by letting every other element from the input matrix go into one of the output matrices and the rest of the elements go into the other output matrix. One of these output matrices is then inserted into an instance of the module NeuralField2D called palm1 with inhibition turned of as it was for palm in the previous models. The other output matrix from the Separate module goes into an instance of a new module called DelayOne2D and from there into another instance of NeuralField2D with inhibition turned of called palm2. The only function of DelayOne2D is to output the same matrix as it took as input. The purpose of this is simply to delay this part of the information flow one iteration. The output from palm1 and palm2 goes into an instance of a new module called Merge which merges its two input matrices into one. From Merge the information proceeds to the previously described module Mixer, which simulates nerve fibre mixing. The output matrix from Mixer goes into the part of the model that represents medulla, as before, and from here and on, the model is similar to the leaky integrator model. Fig. 8.1 shows the modules of the model and how they are connected.

In Fig. 8.2 the results from the simulations with the leaky integrator model, extended with two different rates of signal transfer, are presented.
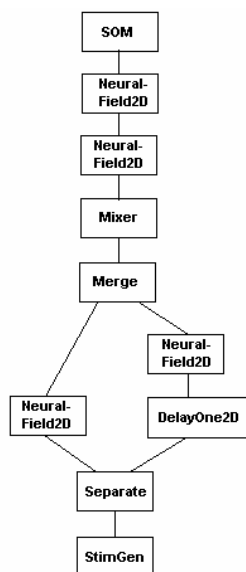

Fig. 8.1. The figure shows the modules in the simulations with the leaky integrator model, extended with two different rates of signal transfer.
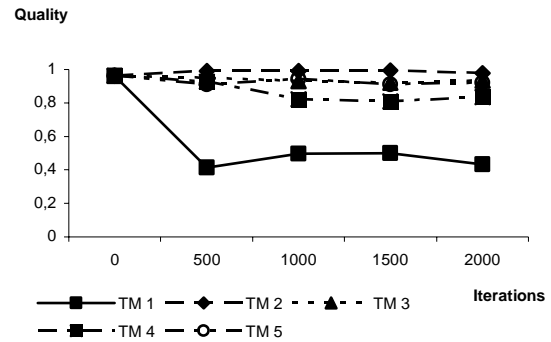

Fig. 8.2. The results of the simulations with the leaky integrator model, extended with two different rates of signal transfer, presented as a diagram. The confidence interval for every training method at every point of measurement is at most $\pm$ 0.11 (p = 0.01). Training method 1 gives a very bad organization of the cortical map in this model, while the others seem quite good except possibly training method 4.

The results from the simulations with this version of the model are very interesting, since the addition of two different rates of signal transfer to the leaky integrator model seems to take away the deterioration in the leaky integrator model compared to the multiple peak model. An exception is the training method 1 which is still bad, probably because of the reasons discussed in section 7.3. On the other hand, the reorganization ability of training method 4 seems to be improved compared to the multiple peak model and so do training method 2 which is now extremely good. That the cortical reorganization is better in this model compared to the leaky integrator model is quite interesting, since one should expect the opposite with the motivation for this model presented above.

9. A Two Palm Model

As mentioned in chapter two the bundles of afferent axons from the nuclei in medulla cross over two the contralateral side of thalamus, but in reality there are also some ipsilateral connections, so there are also an ipsilateral activation. Therefore, stimulation of the healthy arm will lead to some activation of the cortex areas corresponding to the injured arm.

To get an idea of how an equal and simultaneous stimulation of the hand with healthy nerve fibres would influence the recovery of the hand with injured nerve fibres, the previous model, including multiple peak activation, a leaky integrator, and two different rates of signal transfer, was extended to model the neurophysiological systems corresponding to both hands up and till medulla. The system

17

for one of the hands corresponds to an injured arm and the system for the other hand corresponds to a healthy arm. So, the system corresponding to the injured hand has mixing of its nerve fibres turned on, while the system corresponding to the other hand has not. For the system corresponding to the injured arm, there are modules for thalamus and cortex, as in the previous model, but for the system corresponding to the other arm there are not. In the model the relay nuclei in medulla corresponding to the healthy side was connected to the relay nuclei in thalamus corresponding to the injured side.

The NeuralField2D module used for medulla at the healthy side was slightly modified, so that every other element in its output matrix is changed to zeroes. This was because there are less ipsilaterally connected axons, and changing every other element to a zero should correspond to neurons with no ipsilateral axons.

The StimGen module for the healthy side was also slightly modified. This was done to disable it to interfere with the quality measurements, i.e. while measuring quality, only the injured side were stimulated.

Fig. 9.1 shows a schematic depiction of the modules in the two palm model and how they are connected. In Fig. 9.2, the results from the simulations with the two palm model are presented.
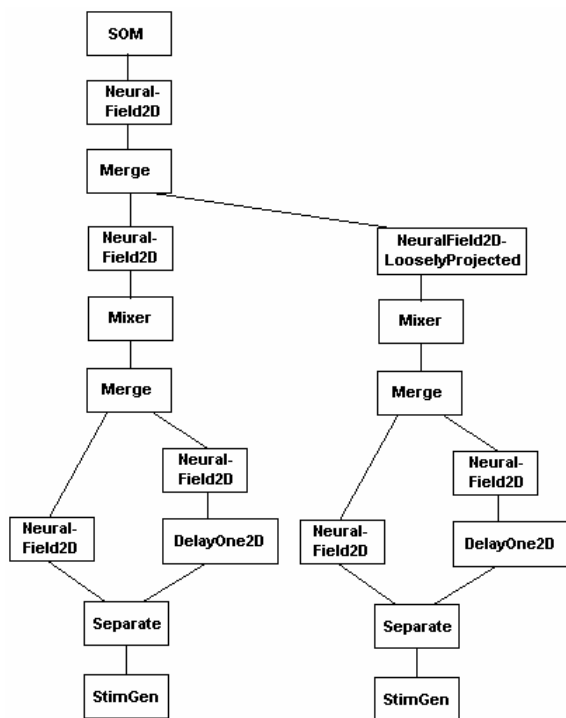

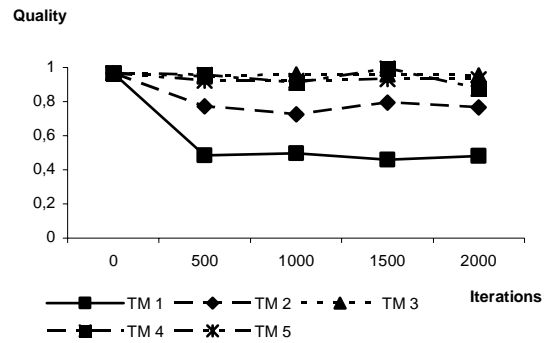**Fig. 9.1.** The figure shows the modules in the two palm model.

Fig. 9.2. The results of the simulations with the two palm model, presented as a diagram. The confidence interval for every training method at every point of measurement is at most ± 0.10 (p = 0.01). Training method 1 gives a very bad organization of the cortical map in this model, training method 2 is better but not very good as in the previous model. The other training methods seem quite good.

The simultaneous stimulation of the palm of the healthy arm does not seem to influence the reorganization very much. This was a bit surprising, since a deterioration of the reorganization due to the ipsilateral activation was expected. This was expected since due the stimulation of the hand of the healthy arm, should tend to push the cortical map into an organization similar to the one prevailing before the nerve injury. Therefore the impact on the cortical organization due to the new situation following the mixing of nerve fibres would be resisted to some extent. But as seen, this was not the case.

10. Discussion

I have simulated the reorganization of somato-sensory cortex to get an idea on what constitutes a proper sensory re-education after injuries to the nerves between the hand and the brain. This work has proceeded by implementing gradually more advanced versions of the model. With each version of the model, simulations were run. The simulations with all versions of the model, gave that a continuous random selection of a small set of points for stimulation on the palm, is a good re-educational strategy.

In general training methods 3 and 5 have worked well in all models. Training method 2 worked fairly well, but not as good as training methods 3 and 5. Training method 1 is not a good choice in those models that include a leaky integrator. A discussion on the reasons for this was included in chapter 7 above. When it comes to training method 4, it

seems in summary, that it would not be a particularly good choice, but even this method becomes fairly good in the model with two different rates of signal transfer. This seems to be a property of that model. I got the impression that the addition of different rates of signal transfer to the leaky integrator model has a stabilizing influence, in that it becomes less sensitive to the choice of training method.

Generally, training methods 3 and 5 have been good in all models, so to hazard a guess and come up with a recommendation; it would be that a training method after nerve generation should randomly choose a small set of points on the hand and stimulate them shortly and then select some new points.

To randomly choose a large part of the receptors of the hand in every iteration of the loop, and stimulate them with a randomly chosen intensity may on the contrary be a less useful method.

When it comes to partial nerve injuries, i.e. there is a part of the hand that is correctly mapped in the cortex, the recommendation would be the same. Perhaps cortical induction would be as good or better, but to enable a statement on this, it would be necessary to run simulations of partial nerve injuries with my more advanced models and in addition with the more realistic implementation of cortical induction. In this project, simulations of partial nerve injuries were only done with the multiple peak model and a simplified version of cortical induction.

The reader should observe that the recommendations above are very loosely founded, since all models are tremendous simplifications of the real neurophysiological systems.

An interesting observation is that training method 2 showed to be extremely good in the model with different rates of signal transfer. This indicates that in reality there might be a possibility that the best design of a training method is not to use sets of randomly selected stimuli points, but a design with some kind of order in the selection, or a combination of random selection and ordered selection.

All versions of the model made in this project are considerable simplifications of their neurophysiological counterparts. One reason for this was to keep the burden of work at a reasonable level. Therefore, I have restricted the model to comprise only the palm, instead of the whole hand, and only touch sensitive receptors have been included. Another reason was that the accessibility to computer power is always limited and therefore

the different neurophysiological systems included in the computational model have been modelled with a considerably less amount of neural cells than in the biological counterparts. For the same reason, the level of detail in the model has been kept at a reasonable level.

With a more well thought-out set of training methods, it would probably be possible to make a stronger statement on the best design of training methods, than I have been able to do, but I did not put a large amount of effort on this. This was because one of the most interesting questions would be to confirm or refute the usefulness of cortical induction. The other training methods were selected to have something to compare it with. So, of course it would have been better if I had pondered on what kind of training methods I would construct to get evenly distributed samples from the set of all possible training methods.

It is not too easy to compare my model with those mentioned in section 2.5, since my simulations are specifically designed to shed some light on the problem with cortical reorganization induced by misconnected nerve fibres, and non of the models mentioned in section 2.5 are aimed at that problem. However, there were mentioned some models that, like mine, includes the hand and the somatosensory cortex. So, it could be worthwhile to mention something on what I have done differently in comparison to them.

Mazza and Roque-da-Silva (1999; 2000) made a couple of models which included the hand and the somatosensory cortex. With the aid of one of these, they simulated the reorganization of somatosensory cortex after changes to the sensory input. One advantage of my model compared to theirs, is that I have included the relay nuclei in the medulla and the thalamus. This makes my model more biologically plausible in that respect.

This is also true for the model made by Lin, Liljeholm, Ozdzdynski and Beatty (2002), also mentioned in section 2.5. Their model also included the hand and the somatosensory cortex. Their model of somatosensory cortex, like mine, used Kohonen's self-organizing map algorithm. But an extension in my model, compared to theirs, is that in my model, short of the first version, I have modified the original self-organizing map algorithm to include influences from all activity in the cortex module on the updating of the weights. In the original algorithm, the updating of the weights of a node depends on the winner node, and the distance to the winner node, while in my modified version of the algorithm, the updating of the weights of a node depends on all the other nodes, but the influence from another node is proportional to how well it

matches the input vector and the distance. The idea is that this should make the model more biologically plausible, in that stimulating several points on the hand will also lead to learning corresponding to separate stimulation of these points.

As a conclusion, I will discuss some suggestions for further research. One suggestion is to build a more realistic and authentic model of the dorsal column-medial leminiscal system. Then one should take into account that there are not only receptors for touch sensations in the hand. There are also receptors for pain, heat and proprioception, and for example there are different types of receptors for touch, as indicated above. Receptors for different submodalities do not project onto the same maps in the somatosensory cortex, so when the nerve fibres have grown together in a random way there are severe problems that have not been taken into account in the simple model in this project. In reality, the problem with misconnections of the nerve fibres are not limited to a reorganization of a map in cortex, because receptors that are expected to project onto one area in the somatosensory cortex are suddenly projected onto another area that is a completely different map. For example a receptor that previously mediated touch, will suddenly convey its signals to a map that expects information from pain receptors. Therefore the reorganization in reality will never be complete, because the nerve fibre mixing is not bounded within a single cortical map.

Another suggestion for further research, is a project that could take its start in the current work, and extend it to also include parts of motor cortex, influences from higher cortical areas, and a more advanced model of a hand. What I have in mind is an attempt to build a larger model of somatosensory cortex, that does not only consists of primary areas, but also secondary. I think of a model that is able, to some extent, of stereognosis, i.e. a model that is able to identify objects with the aid of somatosensory information. This model should consist of modules that took in information of touch and proprioception from, for example, a robot hand, or its simulated equivalent, and integrated this in secondary cortical area modules, enabling an identification of different kinds of object. This would be interesting from a pure scientific viewpoint, and it would also have practical values in for example robotics.

References

Balkenius, C., (2003). Ikaros (2004-01-23). http://www.lucs.lu.se/IKAROS/.

Benuskova, L. (1995). On the role of inhibition in cortical plasticity: a computational study, *Proc. ICANN'95*, 2, 521-526.

Benuskova, L., Diamond, M.E., & Ebner, F.F. (1994). Dynamic synaptic modification threshold: Computational model of experience-dependent plasticity in adult rat barrel cortex, *PNAS*, 91, 4791-4795.

Benuskova, L., Diamond, M.E., & Ebner F.F. (1996). Computational study of plasticity in barrel cortex, In Sincak, P. (Ed), *ELFA: Proc. 1st Slovak neural net. symp.*, 129-136.

Benuskova, L., Diamond, M.E., Ebner, F.F., & Armstrong-James, M. (1999). Computational study of experience-dependent plasticity in adult rat cortical barrel-column, *Network: Computation in neural systems*, 10, 303-323.

Benuskova, L., Rema, V., Armstrong-James, M., & Ebner, F.F. (2001). Theory for normal and impaired experience-dependent plasticity in neocortex of adult rats, to appear in *PNAS* 2001.

Buonomano, D.V., & Merzenich, M.M. (1998). Cortical plasticity: From synapses to maps, *Annu. rev. neurosci.*, 21, 149-186.

Florence, S.L., Jain, N., & Kaas, J.H. (1997). Plasticity of somatosensory cortex in primates, *Seminars in neuroscience*, 9, SN970101, 3-12.

Gardner, E.P., & Kandel, E.R. (2000). Touch. In Kandel, E.R., Schwartz, J.H., & Jessell, T.M., (ed.). *Principles of neural science*, 451-471, McGraw-Hill.

Gardner, E.P., Martin, J.H., & Jessell, T.M. (2000). The bodily senses. In Kandel, E.R., Schwartz, J.H., & Jessell, T.M., (ed.). *Principles of neural science*, 430-450, McGraw-Hill.

Goodall, S., Reggia, J.A., Chen, Y., Ruppin, E., & Whitney, C. (1997). A computational model of acute focal cortical lesions (2003-11-10). http://citeseer.nj.nec.com/ goodall97computational.html

Kahn, D.M., & Krubitzer, L. (2002). Massive cross-modal cortical plasticity and the emergence of a new cortical area in developmentally blind mammals, *PNAS*, 99, 17, 11429-11434.

Kalarickal, G.J., & Marshall, J.A. (2002). Rearrangement of receptive field topography after intracortical and peripheral stimulation: the role of plasticity in inhibitory pathways, *Network: computation in neural systems*, 13, 1, 1-40.

Kilgard, M.P., & Merzenich, M.M. (1998). Cortical map reorganization enabled by nucleus basalis activity, *Science*, 279, 1714-1718.

Kilgard, M.P., & Merzenich, M.M. (2001). Order-sensitive plasticity in adult primary auditory cortex, *PNAS*, 99, 5, 3205-3209.

Kilgard, M.P., Pandya, P.K., Engineer, N.D., & Moucha, R. (2002). Cortical network reorganization guided by sensory input features, *Biol. cybern.*, 87, 333-343.

Klefsjö, B., & Hellmer, S. (1991). Räkna med slumpen, Lund, Studentlitteratur.

Kohonen, T. (1990). The self-organizing map, *Proceedings of the IEEE*, 78, 9, 1464-1480.

Kohonen, T. (2001). Self-organizing maps, Berlin, Springer-verlag.

Lin, A., Liljeholm, M., Ozdzynski, P., & Beatty, J. (2002). Visualizing plastic change in a large model of somatosensory cortex using an adaptive coordinates algorithm, *Neurocomputing*, 44-46, 521-526.

Mazza, M.B., & Roque-da-Silva, A.C. (1999). Computational model of topographic reorganization in somatosensory cortex in response to digit lesions, *Neurocomputing*, 26-27, 435-441.

Mazza, M.B., & Roque-da-Silva, A.C. (2000). Realistic computer simulation of cortical lesion induced imbalances in properties of somatotopic maps, *Neurocomputing*, 32-33, 453-459.

Ranadive, U. (1997). Phantom limbs and rewired brains, *Technology review*, 100, 1, 17-18.

Reinecke, S., Dinse, H.R., Reinke, H., & Witte, O.W. (2003). Induction of bilateral plasticity in sensory cortical maps by small unilateral cortical infarcts in rats, *European journal of neuroscience*, 17, 623-627.

Rosen, B., Balkenius, C., & Lundborg, G. (2003). Sensory re-education today and tomorrow: a review of evolving concepts, *British journal of hand therapy*, 8, 2, 48-56.

Russel, S., & Norvig, P. (1995). *Artificial intelligence – a modern approach*, New Jersey, Prentice- Hall.

Sawaki, L., Yaseen, Z., Kopylev, L., & Cohen, L.G. (2003). Age-dependent changes in the ability to encode a novel elementary motor memory, *Ann neurol*, 53, 521-524.

Siebner, H.R., & Rothwell, J. (2002). Transcranial magnetic stimulation: new insights into represent-ational cortical plasticity, *Exp brain res*, 148, 1-16.

Sirosh, J., & Miikkulainen, R. (1994). Modeling cortical plasticity based on adapting lateral interaction, *The neurobiology of computation: Proceedings of the annual computational neuroscience meeting*, Kluwer academic publishers (1994).

Wedel, J., & Polani, D. (1996). Critic-based learning of actions with self-organizing feature maps (2003-11-10). http://citeseer.nj.nec.com/wedel96criticbased.html

Wiemer, J., Spengler, F., Joublin, F., Stagge, P., & Wacquant, S. (1998). A model of cortical plasticity: Integration and segregation based on temporal input patterns, In Niklasson, L., Boden, M., & Ziemke, T. (Eds), *Proceedings of the 8th international conference on artificial neural networks,* 367-372, Springer-verlag, London (1998).