Institut für Parallele und Verteilte Systeme
Universität Stuttgart
Universitätsstraße 38
D–70569 Stuttgart

Diplomarbeit Nr. 3294

Neuronale Wissensrepräsentation und antizipierende hierarchische Speicher

Ralf Wittmann

Studiengang:	Informatik
Prüfer:	Prof. Dr. rer. nat. habil. Paul Levi
Betreuer:	Dipl.-Inf. Kai Häussermann
begonnen am:	24. Januar 2012
beendet am:	13. Dezember 2013
CR-Klassifikation:	C.1.3, F.1.1, I.1.2.4, I.1.2.6, I.5

Inhalt

Diese Arbeit ist all denen gewidmet, die in widrigen Zeit nicht den Glauben an mich verloren haben. Insbesondere danke ich Maria, Georg, Michael, meinen Eltern, Elvira Zais, Dr. Thomas Meyer, meinem Betreuer Kai Häussermann, Prof. Paul Levi und meinem väterlichen Freund Prof. Bodo Volkmann.

„Mithin, sagte ich ein wenig zerstreut, müssten wir wieder von dem Baum der Erkenntnis essen, um in den Stand der Unschuld zurückzufallen? Allerdings, antwortete er, das ist das letzte Kapitel von der Geschichte der Welt.“

(Heinrich von Kleist, *Über das Marionettentheater*)

Abstract

Neurowissenschaftliche Erkenntnisse über Struktur und Funktion des Neokortex lassen eine Neuorientierung auf dem Gebiet der künstlichen neuronalen Netze ratsam erscheinen. Es gilt, seine tiefe und hierarchische Architektur, seine universelle Fähigkeit, nicht nur räumliche sondern simultan auch zeitliche Muster zu erkennen und zu antizipieren, technisch zu realisieren. Dabei spielt auch ein aktuelles Modell der neuronalen Wissensrepräsentation, Sparse Coding, eine wichtige Rolle. Neben Deep Learning-Netzen, sind auch hierarchisch-temporale Speicher (HTM) ein vielversprechender Ansatz auf diesem Gebiet. Lernalgorithmen für HTM werden vorgestellt und mit Hilfe der openHTM-Plattform, die eine frei verfügbare Version des kommerziellen Grok-Systems ist, erste Experimente durchgeführt.

Neuroscientific insights into neocortical structure and function motivate reorientation in the field of artificial neural networks. It is desirable to seek technical solutions which exhibit its deep and hierarchical architecture and its universal ability to detect and anticipate spatio-temporal patterns. A current model of neural knowledge representation, sparse coding, is instrumental in achieving this goal. Among other concepts like deep learning, hierarchical-temporal memory (HTM) is a promising approach. Learning algorithms for HTM are discussed, and simple experiments with the openHTM-platform, a free and open source version of the commercial Grok system, are presented.

1 Einleitung

Erkenntnisse der Neurowissenschaften über Struktur und Funktionsweise des Neokortex inspirieren neuartige Lernalgorithmen und Datenstrukturen.

Aufgrund der erheblichen Fortschritte der Neurowissenschaften in den letzten Jahrzehnten und der Stagnation in der klassischen, symbolischen KI, ist es erforderlich, das Konzept der künstlichen neuronalen Netze zu überdenken. Zwar haben diese nach wie vor einen festen Platz im Repertoire des maschinellen Lernens[1], doch sind sie von den Fähigkeiten ihres biologischen Vorbilds zu weit entfernt, als dass man eine weitere Entwicklung erwarten könnte, wenn man nur an Verfeinerungen der gegenwärtigen Modelle arbeitete. Die Neurowissenschaften sind - wie wenige andere wissenschaftliche Gebiete - von der Zusammenarbeit verschiedenartigster Disziplinen geprägt. Hierzu gehören die Biologie, die Physiologie, die Biochemie, die kognitive Psychologie, die Physik, die statistische Mathematik und schließlich auch die Informatik, die die Möglichkeit bietet, durch numerische Simulationen Modellvorstellungen zu überprüfen. Ferner bringt die Informatik – quasi als Rüstzeug – eine lange Tradition der konnektionistischen, subsymbolischen Informationsverarbeitung mit. Aus diesem Konglomerat erwächst das zurzeit sehr aktive Fach Computational Neuroscience.

[1] Klassische Backpropagation-Netze wurden seit Mitte der 90er Jahre zunehmend von kernelbasierten Support Vector Machines (SVM) verdrängt. Eine SVM findet eine optimale Trennung der Trainingsbeispiele (im Sinne des PAC-Lernens), wogegen neuronale Netze aufgrund des Gradientenabstiegverfahrens anfällig für lokale Minima sind. Letztere brauchen mehr Trainingsbeispiele (zur Anpassung der zahlreichen Gewichte), dagegen brauchen SVM beim Training mehr Rechenzeit, da sie nicht nur ein Optimierungsproblem lösen müssen, sondern durch den „Kernel-Trick“ ein zweites im dualen Merkmalsraum. In der Regel liegt die Fehlerrate einer gut eingestellten SVM deutlich unter der eines neuronalen Netzes. Das zeigt aber auch, dass SVM einige manuelle Vorgaben (die Auswahl des Kernels, dessen Parameter etc.) brauchen, um ihr Potenzial auszuschöpfen. Der Trainingsvorgang klassischer neuronaler Netze scheint darüber hinaus leichter parallelisierbar zu sein (mittels der Map-Reduce-Technik) als der einer SVM. Auch wenn die Vorzüge der SVM insgesamt wohl überwiegen, ist immer noch Raum für Backpropagation-Netze [Bishop2006] [Schölkopf2002].

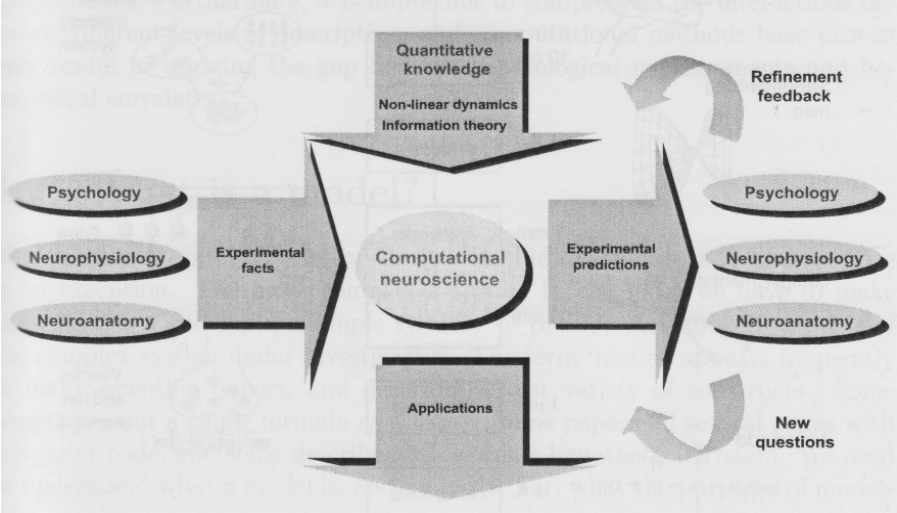

Abbildung 1: Computational Neuroscience ist eine Disziplin zwischen den Welten (Trappenberg, 2004).

Sucht man in der Natur nach Inspirationen für technische Anwendungen, so hat man zwei grundlegend verschiedene Möglichkeiten. Man kann mit höchster Präzision die Natur bis ins Detail nachbilden. Oder aber man versucht, die Essenz der zu Grunde liegenden Strukturen und Prozesse zu erkennen und sie auf einer abstrakteren Ebene technisch zu implementieren. Vergegenwärtigen wir uns zum Beispiel den Flugmechanismus der Libelle. Die schwingende Bewegung ihrer Flügel wird in der Natur durch die Kontraktion von Muskeln leicht realisiert. Damit hat die Technik ihre Schwierigkeiten, nicht jedoch mit Drehbewegungen. Daher übernimmt man nur das abstrakte Prinzip des natürlichen Vorbilds, und übersetzt es so, dass es mit bereits vorhandenen Möglichkeiten verwirklicht werden kann. In unserem Beispiel hat dies bekanntlich zur Entwicklung von Helikoptern geführt. Der entscheidende Punkt ist, dass ein zu starres Beharren auf einer direkten Übertragung der biologischen Lösung unweigerlich zum Scheitern geführt hätte[2].

In der Neurowissenschaft ist das Human Brain Project ein aktuelles Beispiel für die Strategie der detailgetreuen Imitation der Natur. Bei diesem internationalen Großpro-

[2] Dank fortschreitender Miniaturisierung kann man heute dem Libellenflug sehr nahekommen (vgl. BionicOpter der Firma Festo). Sikorsky und die anderen Vorväter des Hubschraubers taten gut daran, sich an den technologischen Gegebenheiten ihrer Zeit zu orientieren.

jekt werden Nervenzellen und ihre Verknüpfungen bis auf die Ebene einzelner biochemischer, genetischer und elektrischer Prozesse simuliert. Ziel ist es, eine funktionierende Simulation von Teilen des menschlichen Gehirns zu erhalten, wobei der Aufwand an Rechenleistung und Speicher enorm ist. Hunderte von Prozessoren werden benötigt, um die Dynamik eines einzelnen Neurons zu verwirklichen. Für praktische Anwendungen, wie etwa die Steuerung eines Roboters wäre eine solche Vorgehensweise nicht zielführend.

Die Lektüre von Jeff Hawkins' Buch *On Intelligence: How a New Understanding of the Brain will Lead to the Creation of Truly Intelligent Machines* war für mich die entscheidende Anregung zur vorliegenden Arbeit. Wenn auch nicht durchweg neu, so ermöglichten mir die darin vorgestellten Konzepte – darunter das Memory-Prediction-Framework, der hierarchisch temporale Speicher, die Rolle der Zeit für den Lernvorgang – einen frischen Blick auf die Probleme des maschinellen Lernens. Hawkins, durch die Erfindung des Palm (ein früher PDA) finanziell unabhängig geworden, gründete die Start-Up-Firma Numenta, deren Ziel es ist, auf diesen Ideen aufbauende Software zu entwickeln. Auch wenn bis heute kein Durchbruch im Sinne eines Quantensprungs gelungen zu sein scheint, ist ihre Werthaltigkeit gegeben.

Bei all der Faszination, die von diesen gewiss vielversprechenden Ansätzen ausgeht, ist auch ein Maß an Demut angebracht. Allzu oft schon haben neue Ideen die Forschergemeinde zu den kühnsten Prognosen[3] veranlasst, nur um dann wieder in der Schublade zu verschwinden, da sie zwar synthetische Probleme im Miniaturformat zu bewältigen vermochten, die Skalierung auf realistische Fälle aber schuldig blieben. So bleibt zu hoffen, dass die im Folgenden vorgestellten Konzepte wenigstens einen weiteren Mosaikstein im ausgedehnten Feld der KI ergeben werden.

Überhaupt ist die KI vermutlich ein zu breit verästeltes Gebiet, als dass eine einzelne Idee den Gordischen Knoten durchschlagen könnte. Wenn es etwas gibt, das man den

[3] Man denke nur an Kurzweil, der schon wiederholt die „Singularität“ vorhersagte, den Zeitpunkt, an dem Maschinen die geistigen Leistungen des Menschen überträfen und dann die Geschicke der Welt in ihre Hand nähmen.

Heiligen Gral der KI nennen könnte, bestünde der in einer Synthese aus konnektionistischen und symbolverarbeitenden Methoden in hybriden Systemen, um das Beste aus zwei Welten zu verknüpfen. Man denke da zum Beispiel an einen Roboter, der sich im Hörsaal bewegt, eine Gleichung auf der Tafel sieht, diese mit Hilfe eines Computeralgebrasystems löst, und sich darüber mit einer Person unterhält.

Fast alle Hypothesen, die bisher über das Gehirn aufgestellt wurden, machen implizit zwei Annahmen. Erstens, dass Neuronen ihre DNA nicht als Datenspeicher oder für Rechenvorgänge benutzen[4]. Zweitens, dass sie sich wie ein klassischer Computer verhalten und Quanteneffekte keine Rolle spielen. Beide Postulate sind weder zwingend noch trivial. Vielleicht liegt gerade unter dem Schleier der surreal erscheinenden Quantenmechanik der Schlüssel zur Lösung des größten Rätsels der Hirnforschung, das kein reduktionistisches Modell plausibel erklären kann: Die Existenz unseres Bewusstseins. Wie gelangt der Geist in die Maschine?

[4] Das menschliche Genom besteht aus etwa $6 \cdot 10^9$ Basenpaaren, was einem Informationsgehalt von 1.5 Gigabyte entspricht, der Speicherkapazität von zwei CDs. Bei ca. $20 \cdot 10^9$ Nervenzellen erhält man insgesamt $3.2 \cdot 10^{19}$ Byte. Auch wenn der größte Teil der DNA fest mit Erbinformation belegt ist, liegt dieser Wert weit jenseits der Kapazität moderner Supercomputer.

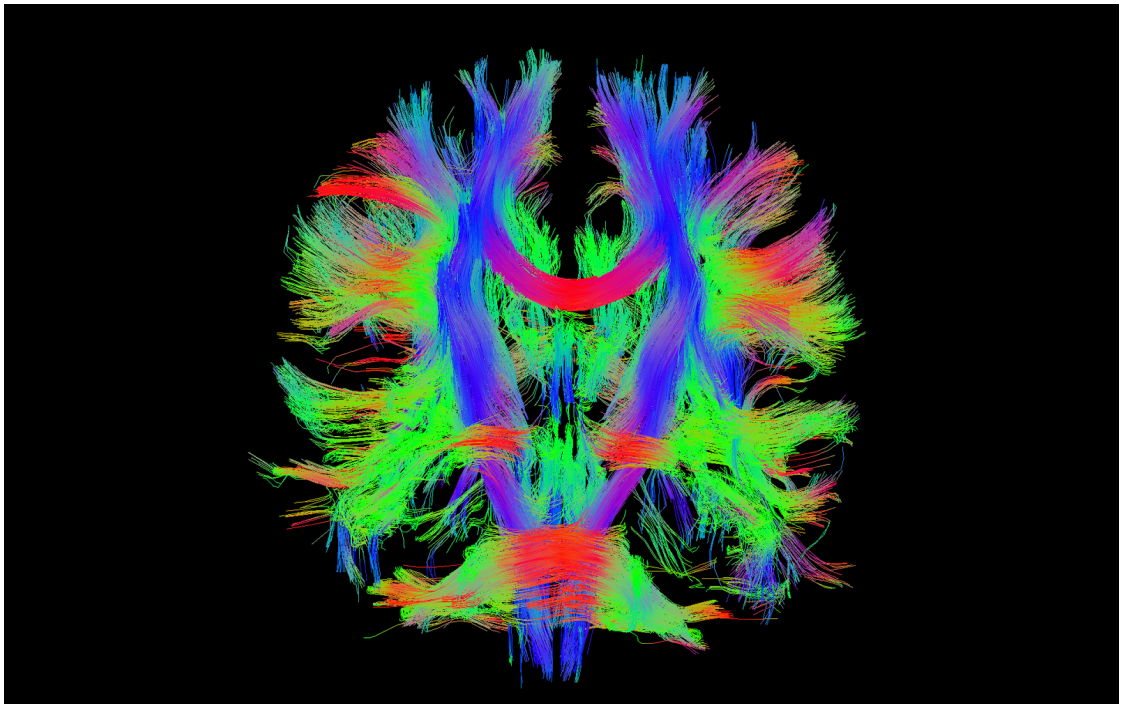

Abbildung 2: Unter dem Begriff Konnektom versteht man die Gesamtheit der Nervenleitungen im Gehirn. Nervenleitungen sind im Wesentlichen Bündel aus Axonen ("Weiße Substanz"), in denen in Wasser gelöste Nährstoffe fließen. Die Diffusionsbewegung dieser Wassermoleküle kann man durch Diffusion Tensor Imaging, eine Variante des MRI, sichtbar machen. Das Bild zeigt Verbindungen in großem Maßstab zwischen verschiedenen Gehirnteilen (A. Lahti, University of Alabama, 2013).

2 Das Neuron und seine Verbindungen

Nervenzellen sind zusammen mit den Gliazellen, die hauptsächlich eine Stützfunktion haben, die Grundbausteine des Gehirns.[5] Sie sind auf Informationsverarbeitung und -weiterleitung spezialisiert. Wie andere Körperzellen besitzen sie einen Zellkörper (Soma), einen Zellkern (Nukleus) sowie die typischen Organellen eukaryiotischer Zellen (Mitochondrien, Ribosomen, Endoplasmatisches Retikulum, Golgi-Apparat). Ihr besonderes Merkmal ist ihre Form. Sie besitzen zweierlei Arten von Zellfortsätzen: einen oder mehrere Dendriten (griechisch: Baum) und ein Axon (griechisch: Achse), das beim Menschen eine Länge von bis zu einem Meter erreichen kann. Das Rückenmark besteht zu einem wesentlichen Teil aus Axonbündeln. Vereinfacht gesagt, sind die Dendriten die Eingänge des Neurons, das Axon sein Ausgang. Neuronen werden strukturell (nach Anzahl und Art ihrer Ein- und Ausgänge) und funktional (nach ihrer Aufgabe) klassifiziert.

- Unipolare sensorische Neuronen besitzen ein Axon und keinen Dendriten. Beim Menschen findet man sie als Stäbchen und Zapfen in der Retina oder als Tastrezeptoren in der Haut.
- Bipolare sensorische Neuronen besitzen ein Axon und einen Dendriten. Man findet sie vor allem in der Netzhaut, wo sie die Informationen der lichtempfindlichen Stäbchen und Zapfen gewichten, sammeln und an Ganglienzellen weiterleiten.
- Multipolare Neuronen verfügen über zahlreiche Dendriten und ein Axon. Sie machen einen Großteil des Neokortex aus, aber man findet sie auch als motorische Neurone, die Signale zu Muskeln und Organen leiten.

[5] Tatsächlich ist die Funktion der Gliazellen (vom griechischen Wort für Leim) weitaus komplexer. Sie spielen eine wichtige Rolle in der Entwicklung des Nervensystems, in der Reparatur von beschädigten Nervenzellen, in der synaptischen Plastizität und der Ausformung der Synapsen, und sie dienen als elektrischer Isolator des Axons. Somit sind sie Partner der Neuronen im Prozess der Informationsverarbeitung [Koob2009].

- Das Axon kann von einer Myelinhülle (einer Membran aus Proteinen und Lipiden) umgeben sein. Sie dient als elektrischer Isolator, der die Signalübertragung längs des Axons erheblich beschleunigt. Derartige Nervenfasern haben eine weiße Farbe, weshalb man von „weißer Substanz“ spricht. Nicht umhüllte Axone erscheinen grau, die sprichwörtlichen „grauen Zellen“ des Kortex.
- Interneurone übertragen Impulse zwischen sensorischen und motorischen Neuronen.

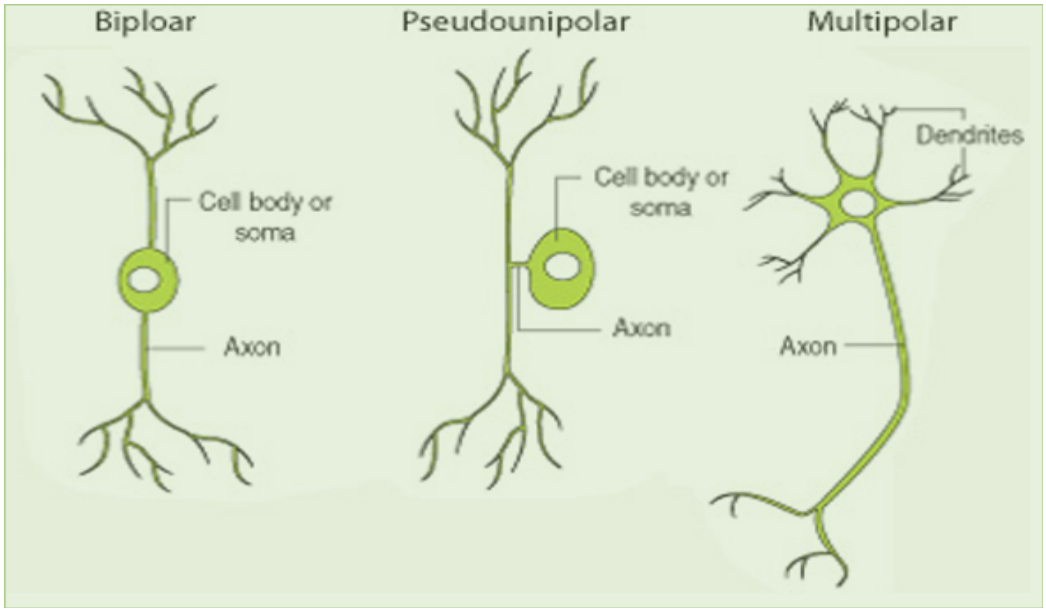

Abbildung 3: Haupttypen der Nervenzellen, ohne Gliazelle (MedCell, Yale University).

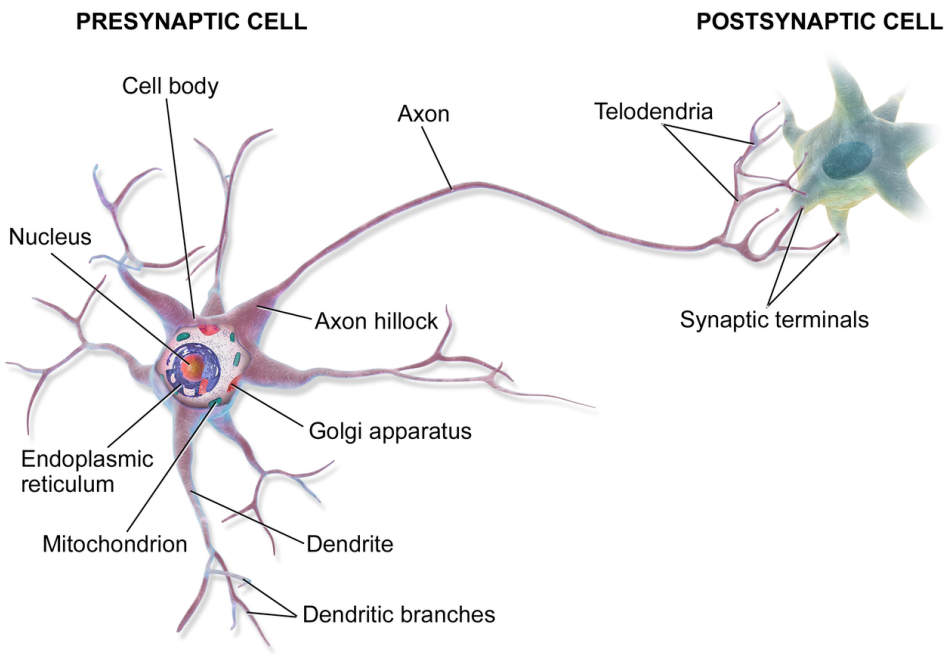

Abbildung 4: Grundstruktur einer Nervenzelle. Das Axon ist hier nicht von einer Myelinhülle umgeben (Wikipedia).

Nervenzellen haben im Vergleich zu anderen Körperzellen eine besonders lange Lebensdauer. Lange Zeit galt es als Dogma, dass im Gehirn eines Erwachsenen keine neuen Nervenzellen mehr gebildet werden, was jedoch inzwischen widerlegt wurde. Zwar teilen sich Neuronen dort nicht mehr, sie können aber aus Stammzellen generiert werden. Bei entsprechender geistiger und körperlicher Aktivität kann dies bis ins hohe Alter geschehen. Die Neubildung von Neuronen ist ein integraler Teil des Lernprozesses, der bisher bei der Konstruktion künstlicher neuronaler Netze nur unzureichend berücksichtigt wurde.

Ähnlich wie Muskelzellen haben Neuronen einen besonders hohen Energiebedarf, weshalb das Gehirn erhebliche Ressourcen beansprucht. Außerhalb des zentralen Nervensystems gelegene Neuronencluster bezeichnet man als Ganglien. Sie sind der evolutionäre Vorläufer des Gehirns.

2.1 Ein kurzer Abstecher in die Elektrizitätslehre

Körperzellen sind insgesamt elektrisch neutral, doch können an manchen Stellen positive oder negative Ladungen dominieren. Folgendes ist für das Verständnis der Signalübertragung in Nervenzellen grundlegend.

- Gegensätzliche Ladungen ziehen sich an. Um sie zu trennen, muss Energie aufgewendet werden. Sie besitzen dann eine potenzielle Energie. Die elektrische Spannung ist die Potenzialdifferenz zwischen zwei Punkten.
- Der Fluss elektrischer Ladungen zwischen zwei Punkten ist der elektrische Strom. Je höher die Spannung, desto größer die Stromstärke; je höher der Widerstand, desto geringer die Stromstärke (Ohmsches Gesetz).
- Ströme im Körper entsprechen dem Fluss von Ionen durch Membranen, die einen elektrischen Widerstand besitzen.
- Der Transport der Ionen durch die Membran erfolgt durch Ionenkanäle. Diese können
    - immer geöffnet sein,
    - abhängig vom Membranpotenzial selektiv geöffnet werden,
    - oder chemisch durch die Anwesenheit von Neurotransmittern gesteuert werden.
- Durch Diffusion fließen Ionen entlang elektrochemischer Gradienten (Konzentrationsunterschiede).

Im Ruhezustand sorgen Ionenpumpen (Kalium und Natrium) dafür, dass zwischen der Innen- und der Außenseite der Membran eine Spannung von – 70 mV herrscht (polarisierter Zustand). Diese Potenzialdifferenz kann sich nur verändern, wenn a) die Membran selektiv ihre Durchlässigkeit ändert oder b) wenn sich die Ionenkonzentration auf einer Seite ändert.

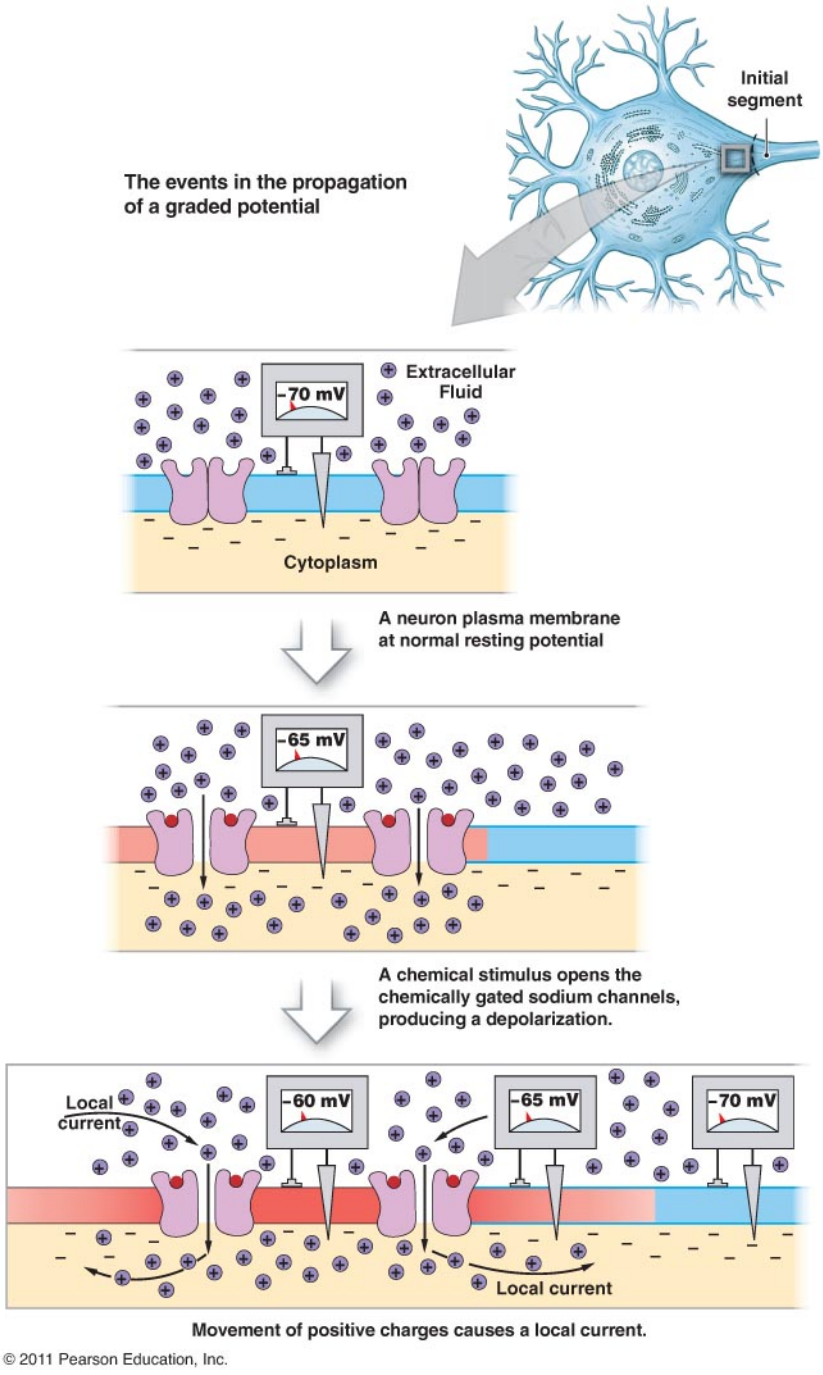

Abbildung 5: Fortpflanzung einer Potenzialänderung im Dendriten (Pearson, 2011).

Stimulation im Dendriten führt zu einer Änderung der Polarisierung. Diese pflanzt sich entlang der Membran zum Zellkörper hin fort. Dabei schwächt sich der elektrische Strom ab. Die ursprüngliche Stromstärke am Ort der Stimulation hängt von der Stärke des Reizes ab. Im Dendriten und im Zellkörper verhalten sich Neuronen also wie analoge Schaltkreise.

2.2 Aktionspotenziale

Die im Zellkörper angelangten Potenzialänderungen summieren sich im Axonhügel, an dem das Axon den Zellkörper verlässt, und aktivieren dort Ionenkanäle. Dadurch fließt ein Strom entlang der Membran, der wiederum benachbarte Ionenkanäle öffnet. Nach etwa einer Millisekunde schließen sich die Kanäle wieder. Da sie eine gewisse Zeit brauchen, um erneut angeregt werden zu können, pflanzt sich das Aktionspotenzial nur in einer Richtung fort. In den fünfziger Jahren haben Hodgkin und Huxley eine nichtlineare Differenzialgleichung aufgestellt, die die Ionenströme durch und entlang der Membran beschreibt (Nobelpreis 1962) [Hodgkin52]. Ihre numerische Lösung liefert die zeitliche Potenzialänderung an einer festgelegten Stelle des Axon und kommt den empirischen Messungen sehr nahe. Die Gleichung von Hodgkin und Huxley dient noch heute als Grundlage der Simulation von Neuronen.

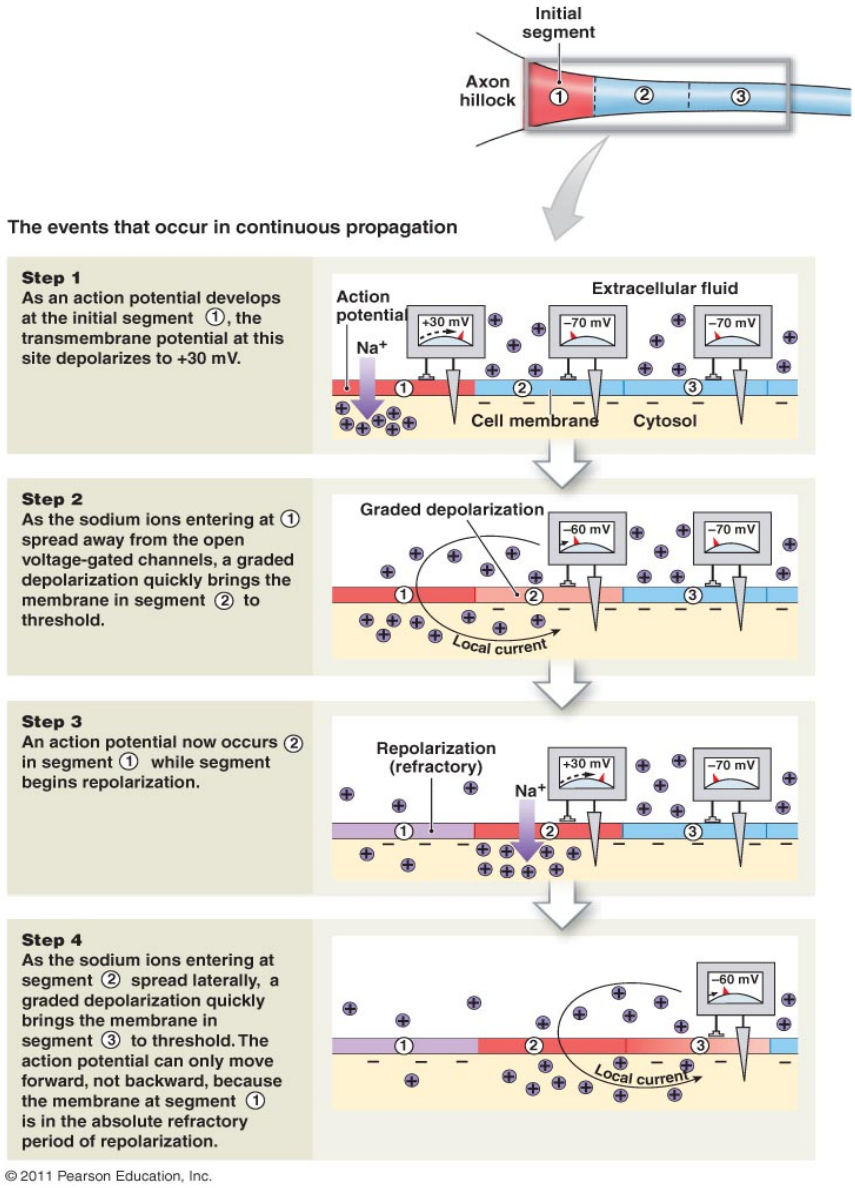

Abbildung 6: Der Weg des Aktionspotenzials entlang des Axons (Pearson, 2011).

Wichtig für das Verständnis der Signalweiterleitung ist, dass das Aktionspotenzial überall die gleiche Amplitude und Form besitzt. Ein einzelner derartiger Impuls kann somit fast keine Information tragen, außer durch die spezifische Leitung (Axon), auf der er gesendet wird. Die Information über die Stärke des Ausgangsreizes wird durch

die Frequenz einer Folge von Aktionspotentialen und die Dauer des Feuerns der Nervenzelle kodiert.

Ein etwas anderes Bild ergibt sich, wenn das Axon von einer isolierenden Myelinhülle umgeben ist. Sie umschließt die Nervenfaser nicht durchgängig, sondern ist im Abstand von etwa 1 mm durch sogenannte Ranviersche Schnürringe unterbrochen. An den umhüllten Stellen kann das elektrische Feld die Membran nicht durchdringen. Eine Potenzialänderung kann erst wieder an der nächsten nicht isolierten Stelle erfolgen. Dadurch springt das Aktionspotenzial von einem Schnürringen zum nächsten, was die Leitungsgeschwindigkeit erheblich erhöht und außerdem Energie für die Ionenpumpen einspart .

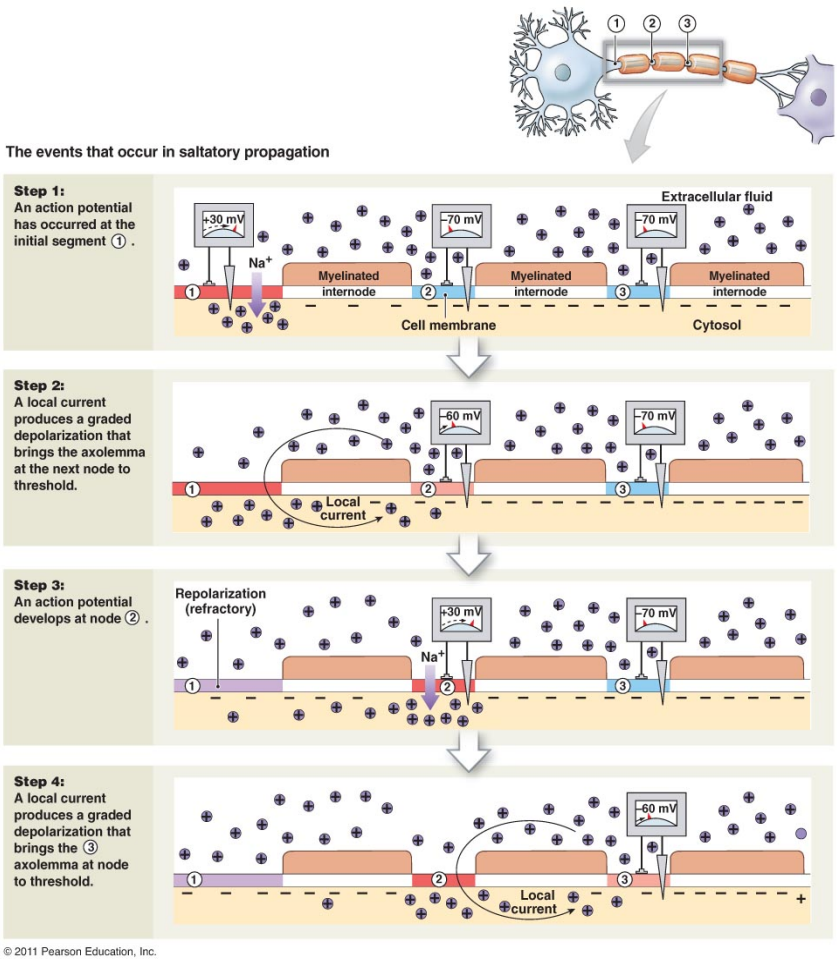

Abbildung 7: Beschleunigte Signalübertragung an einem myelinumhüllten Axon (Pearson, 2011).

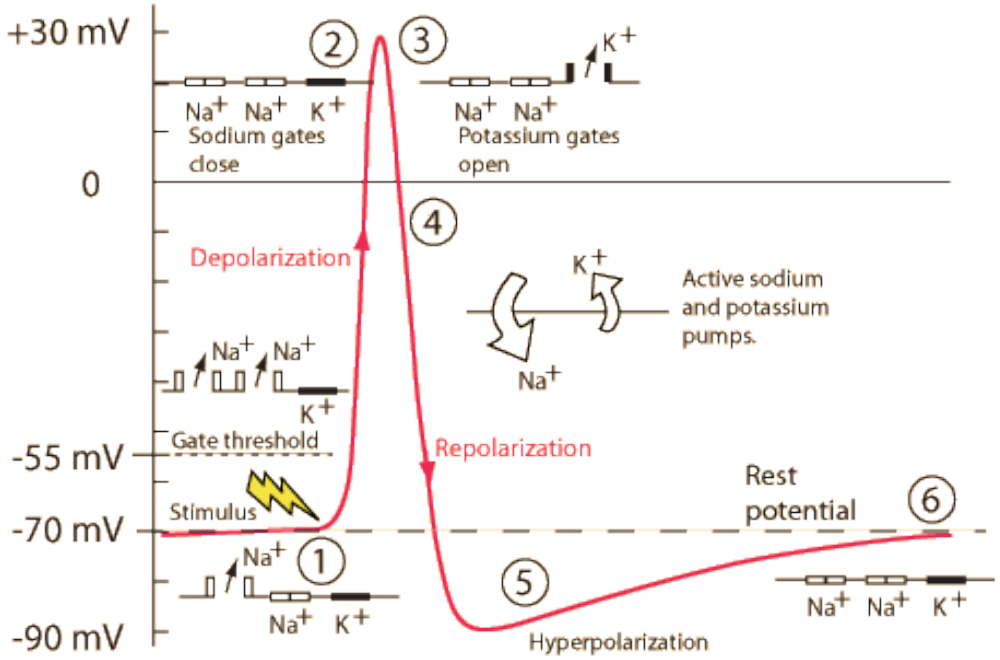

Abbildung 8: Die charakteristische Form des Aktionspotenzials (K. X. Charand, Georgia State University).

2.3 Die Synapse: chemische Schaltstelle für elektrische Signale

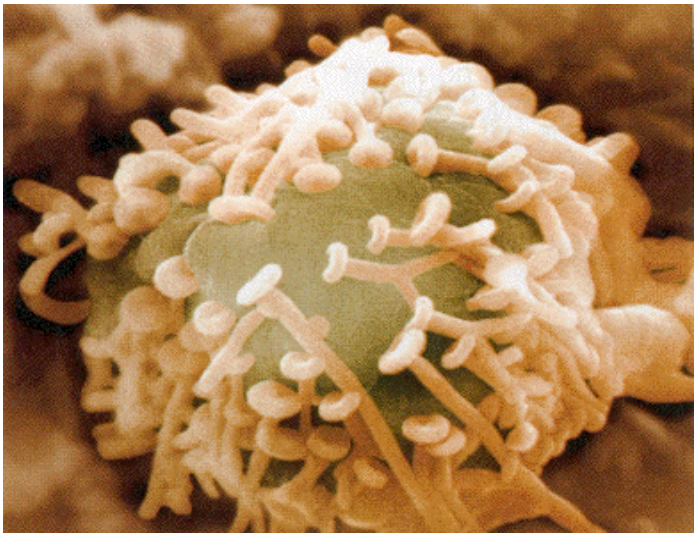

Abbildung 9: Elektronenmikroskopische Aufnahme von Axonenden, die am Zellkörper eines Neurons andocken (G. Boeree).

Synapsen (griechisch: „ineinander greifende Teile“) sind die Schnittstellen zwischen Neuronen und anderen Körperzellen, meist andere Neuronen. Ihre große Mehrheit vermittelt ankommende Signale nicht direkt auf elektrischem Wege, sondern bedient sich in Gestalt der sogenannten Neurotransmitter eines chemischen Umweges. Neurotransmitter werden vom endoplasmatischen Retikulum in der Nähe des Zellkernes gebildet, in Bläschen verpackt und an die Enden des Axons transportiert. Später werden ihre chemischen Zerfallsprodukte zurück geleitet und für die Synthese wiederverwendet.

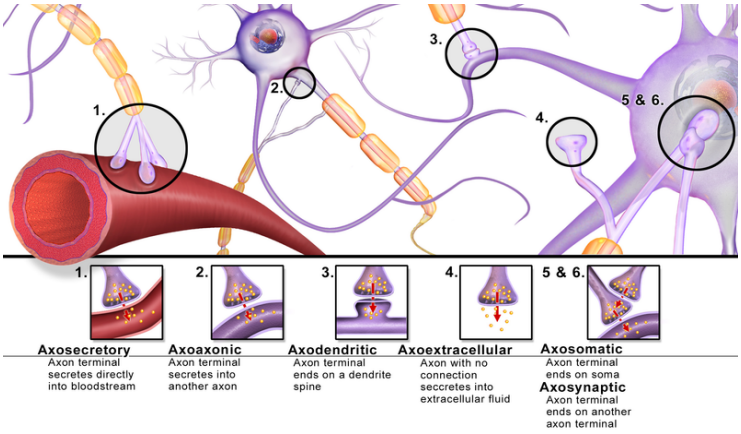

Abbildung 10: Nicht alle Synapsen verbinden Nervenzellen. Manche aktivieren Muskelfasern, andere geben Neurotransmitter direkt in die Blutbahn ab (Wikipedia).

Erreicht ein Aktionspotenzial die Synapse, so öffnen sich dort spannungsgesteuerte Ionenkanäle und es kommt zum raschen Einströmen von Calciumionen in die Synapse. Diese bewirken, dass die in Bläschen gespeicherten Neurotransmitter in den synaptischen Spalt freigesetzt werden. Der Neurotransmitter diffundiert auf die andere Seite des Spalts und bindet sich dort an die Rezeptoren von chemisch gesteuerten Ionenkanälen, die sich daraufhin öffnen und Ionen durch die Membran des Dendriten der nachfolgenden Zelle einströmen lassen. Das Ruhepotenzial im Dendriten ändert sich sprunghaft, und der Impuls nimmt seinen Lauf wie oben beschrieben. Für die Reizübertragung benötigt eine Synapse etwa 0.5 ms.

Die chemische Zwischenstation in der Signalübertragung hat einen entscheidenden Vorteil: Unterschiedliche Neurotransmitter mit unterschiedlichen Eigenschaften erlauben es, das Signal wesentlich flexibler zu modulieren, als es durch eine rein elektrische Kopplung möglich wäre. Ein präsynaptisches Aktionspotenzial kann postsynaptisch aktivierend oder hemmend wirken. Heute ist eine Vielzahl an Neurotransmittern bekannt. Am besten erforscht sind Serotonin, Noradrenalin, Dopamin und Acetylcholin. Interessanterweise können Neurotransmitter in verschiedenen Hirnregionen verschiedene Modulationseigenschaften haben. Der Wirkungsmechanismus der meisten Psychopharmaka besteht in der Verstärkung oder Hemmung der Ausschüttung oder Wie-

deraufnahme von Neurotransmittern an der Synapse. Das gleiche gilt für die meisten Drogen und auch für Alkohol.

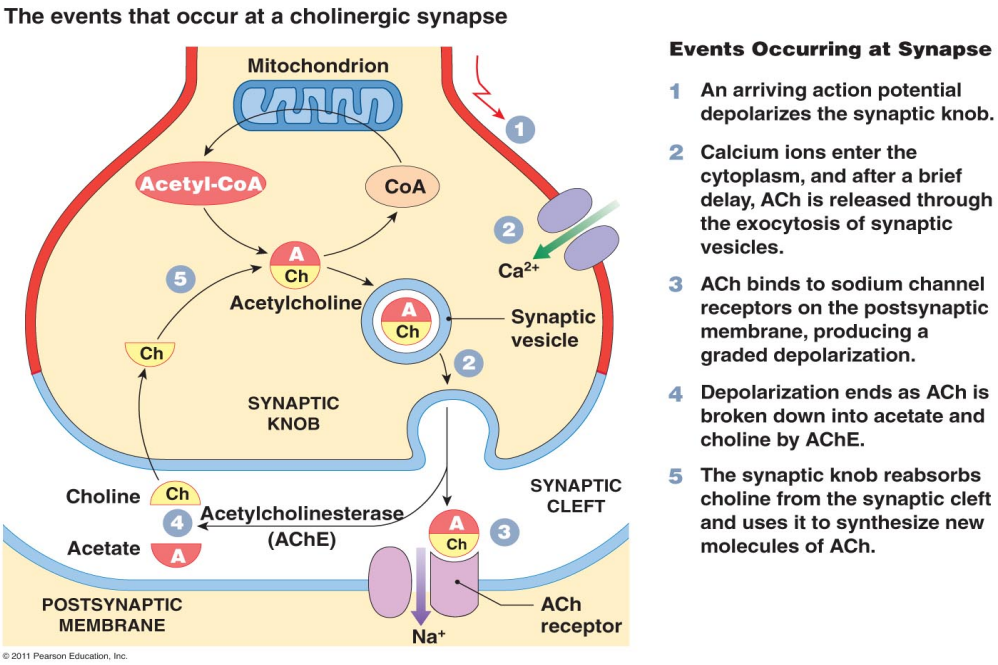

Abbildung 11: Funktionsweise einer Synapse am Beispiel des Neurotransmitters Acetylcholin (Pearson, 2011).

2.4 Synaptische Plastizität: der Ursprung der Hebbschen Lernregel

Die Kopplungsstärke einer Synapse ist nicht konstant. Häufig aktive Synapsen erhöhen ihre Effektivität. Vermutlich beruht hierauf die Fähigkeit des Lernens. Donald Hebb formulierte 1949 seine berühmte Lernregel [Hebb49]:

„Wenn ein Axon der Zelle A […] Zelle B erregt und wiederholt und dauerhaft zur Erzeugung von Aktionspotenzialen in Zelle B beiträgt, so resultiert dies in Wachstumsprozessen oder metabolischen Veränderungen in einer oder in beiden Zellen, die

bewirken, dass die Effizienz von Zelle A in Bezug auf die Erzeugung eines Aktionspotenzials in B größer wird.“

Leichter zu merken ist die etwas flapsige Zusammenfassung der Neurowissenschaftlerin Carla Shatz:

„What fires together, wires together.“

Ganz wörtlich zu nehmen ist dies aber nicht, da es beim Lernen um Kausalität geht. Und damit um zeitversetzte Feuern der beteiligten Neuronen. Außerdem betrachten wir hier nicht die Neuverdrahtung zwischen Neuronen, sondern ausschließlich die Modifikation bestehender Verknüpfungen.

Mathematisch formuliert lautet die Hebbsche Regel wie folgt:

$$\Delta w_{ij} = \eta o_i a_j \quad .$$

Dabei ist Δw_{ij} die Änderung des Gewichts w_{ij}, η eine konstante Lernrate, o_i die Ausgabe der Vorgängerzelle i und a_j die Aktivierung der Nachfolgezelle j.

Die bekannte Delta-Regel, die in verallgemeinerter Form im Backpropagation-Verfahren für mehrstufige künstliche neuronale Netze angewendet wird, ist eine Umsetzung der Hebbschen Regel.[6]

Die Lebensdauer der synaptischen Veränderung ist äußerst variabel. Die Zeitskala reicht dabei von Millisekunden bis hin zu Stunden (man spricht dann von Long-Term Potentiation bzw. -Depression), möglicherweise auch erheblich länger.

[6] Die naheliegende direkte Verstärkung der Verbindungsgewichte lässt diese exponentiell wachsen und führt zu instabilen Netzen. Die Delta-Regel berücksichtigt dieses Problem.

3 Der Neokortex: Die Krone der neuronalen Evolution

Unter dem Begriff Neokortex (lateinisch Cortex = Rinde) versteht man den stammesgeschichtlich jüngsten Teil der Großhirnrinde der Säugetiere. Er bildet die Oberfläche der beiden Gehirnhälften, ist 2 bis 4 mm dick und besteht aus sechs Schichten (bezeichnet mit I bis VI), wobei VI die innerste ist[7]

Es ist bemerkenswert, dass bei allen Säugetieren, den Menschen eingeschlossen, die Anzahl der Schichten gleich ist, lediglich die Größe des Neokortex variiert. Als Fläche misst er beim Menschen etwa 0.25 qm, was ungefähr der Größe eines Taschentuchs entspricht, wobei er aber hochgradig gefaltet ist, um das Verhältnis von Oberfläche zu Volumen zu optimieren. Beim Menschen macht er 80% des Gehirnvolumens aus und enthält ca. 20 Milliarden Nervenzellen. Er ist Sitz der höheren Hirnfunktionen (Sinneswahrnehmung, motorische Steuerung, Sprache, räumliches und bewusstes Denken). Der Neokortex ist nicht der einzige Teil des Gehirns, der für kognitive Prozesse zuständig ist. Neben dem Thalamus[8] ist auch der Hippocampu[9]s hierfür von Bedeutung. Wir beschränken uns aber auf die Analyse des Kortex.

Etwa 80% der kortikalen Neuronen sind Pyramidenzellen, die ihren Namen aufgrund ihres im Schnittbild dreieckigen Zellkörpers tragen. Sie besitzen ein einzelnes, oft sehr langes Axon und haben eine exzitatorische Funktion: in ihrem Dendriten findet man wesentlich mehr verstärkende (exzitatorische) als hemmende (inhibitorische) Synapsen. Diese exzitatorische Wirkung ist nicht lokal beschränkt. Informationen können in weit entfernt liegende Regionen weitergeleitet werden. Die restlichen Neuronen des Neokortex sind sogenannte Inter- oder Zwischenneurone. Sie schwächen eingehende Aktionspotenziale ab (Inhibition) und wirken nur auf benachbarte Zellen [Lund88].

[7] In manchen Regionen auch weniger als sechs Schichten.

[8] Der Thalamus agiert als Schnittstelle zwischen Sinnesorganen und Kortex. Zusätzlich gibt es Rückkopplungsschleifen mit den kortikalen Schichten I und V, was zur Erkennung und Speicherung zeitlicher Muster beiträgt.

[9] Der Hippocampus spielt eine wichtige Rolle im hierarchischen, bidirektionalen Kognitionsprozess. Er wird aktiv, wenn weiter unten liegende Ebenen auf „unerwartete“ Ereignisse treffen und formt daraus das Langzeitgedächtnis.

Abbildung 12: Pyramidenzellen aus dem Neokortex einer Katze, Zeichnung des Neurophysiologen Santiago Ramon y Cajal (1852-1934).

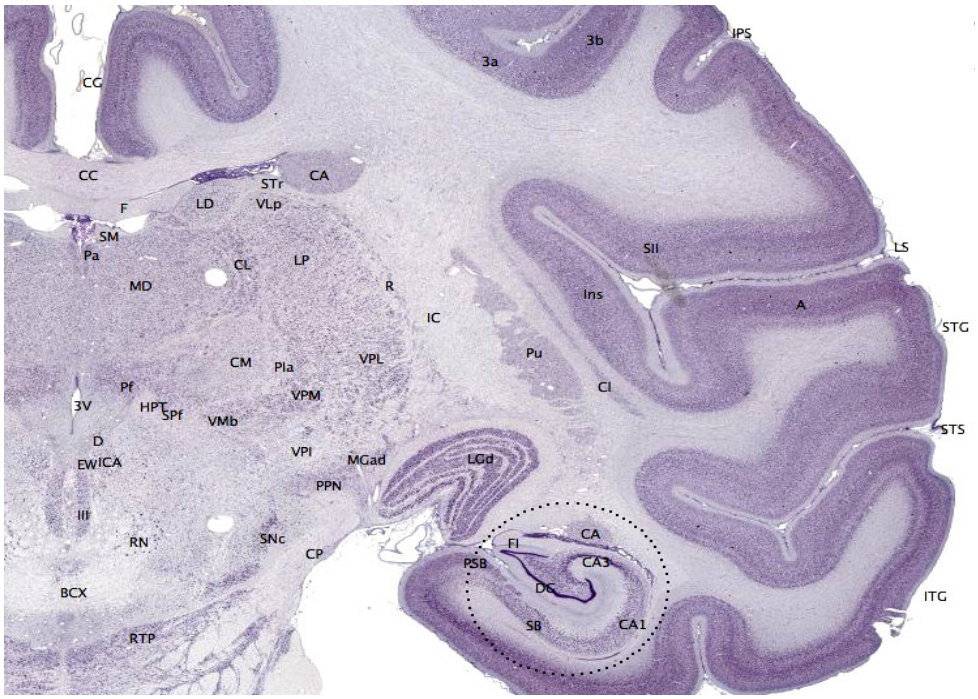

Abbildung 13: Großhirnrinde eines Makaken (violett eingefärbt). Diese sogenannte graue Substanz (überwiegend aus Zellkörpern bestehend) liegt außen, die weiße Substanz (überwiegend Axone) innen (www.brainmaps.org).

3.1 Laminare Struktur des Neokortex, erläutert am Beispiel des primären visuellen Kortex

Lichtreize werden auf den Fotorezeptoren der Retina (Stäbchen und Zapfen) registriert. Über zwei Stationen zur Zwischenverarbeitung (Bipolarzellen und Ganglien), die als Grundstufen der Bildverarbeitung noch im Auge liegen, laufen sie dann als Aktionspotenziale den Sehnerv entlang, der über Kreuz verläuft, wodurch rechtes und linkes Bild vertauscht werden. Der Sehnerv endet in einem auf visuelle Wahrnehmung spezialisierten Teil des Thalamus, dem corpus geniculatum laterale (engl. LGN). Dort findet eine weitere Grundstufe der Bildverarbeitung statt, bevor die Signale schließlich in den primären visuellen Kortex V1 gelangen. Der hier geschilderte hierarchische Prozess

der Informationsverarbeitung ist nicht ausschließlich feed forward, sondern man findet auf mehreren Ebenen Rückkopplungen. (z.B. von V1 zurück zum Thalamus, was vermutlich der Fokussierung dient). Auch liegt hier nicht die Endstufe der optischen Wahrnehmung, denn Informationen werden in Kortexareale mit noch höherem Abstraktionsgrad weitergeleitet.

Jedes Neuron des V1 besitzt ein Wahrnehmungsfeld, das einem diskreten Gebiet auf der Retina entspricht. Diese Abbildung ist retinotop, d.h. die Topologie des Bildes auf der Retina bleibt erhalten, allerdings nicht seine Geometrie. Bei der Abbildung des Gesichtsfelds auf die Retina finden zahlreiche räumliche Transformationen statt. So ist etwa die Hälfte der Fläche von V1 einem nur zwei Prozent großen Gebiet im Zentrum des Sehfeldes zugeordnet.

Die Haupteinspeisung in den V1 erfolgt in Schicht IV (ihrerseits in Unterschichten unterteilt). Die genaue Art der Verdrahtung ist noch immer Gegenstand aktueller Forschung. Die vierte Auflage von Kandels einflussreichem Lehrbuch *Principles of Neural Science* aus dem Jahr 2000 zeichnet ein Bild, das in Abbildung 3 dargestellt ist [Kandel2000].

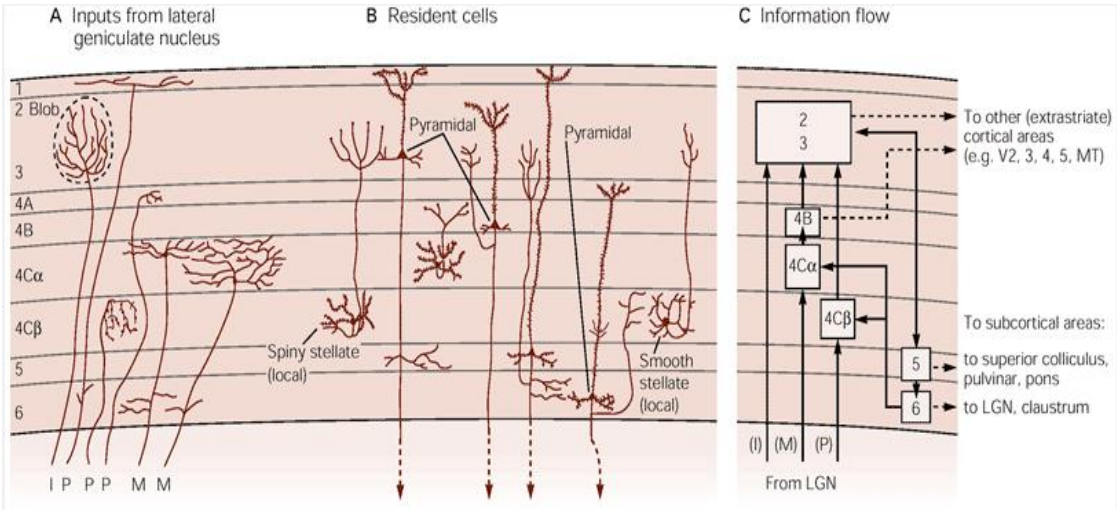

Abbildung 14: A) Enden der Nervenbahnen aus dem LGN. B) Zelltypen im V1. C) Informationsfluss. (Kandel et al., 4. Auflage, 2000).

Vier Jahre später findet man in einer statistischen Auswertung der V1-Synapsen bereits andere Angaben [Binzegger2004].

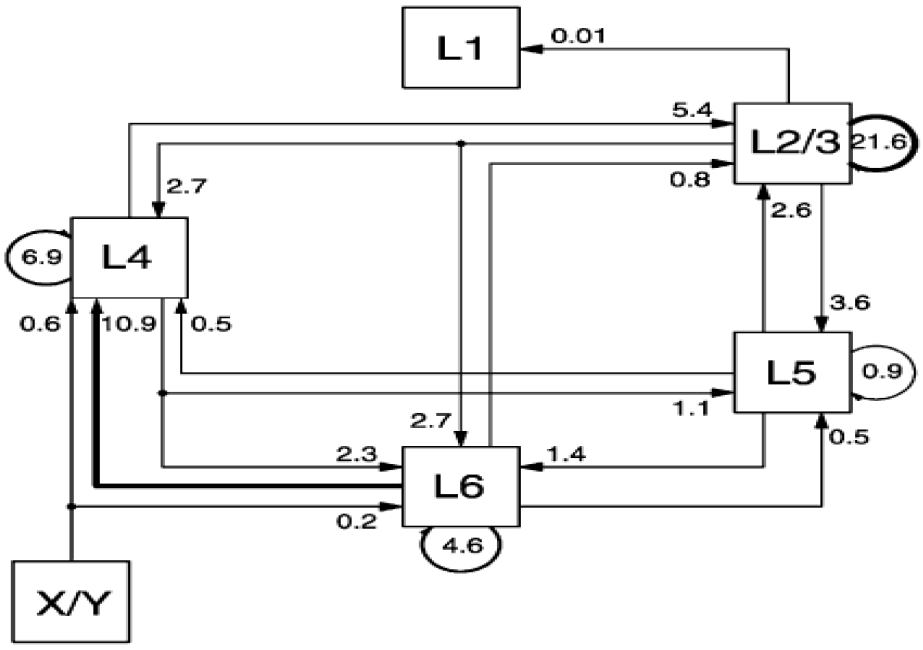

Abbildung 15: Exzitatorische Leitungsbahnen im V1. X/Y steht hier für den LGN. Die angegebenen Werte entsprechen dem Anteil an der Gesamtzahl der Verbindungen. Inhibitorische Verbindungen sind nicht aufgeführt (Binzegger, 2004).

Die Zellen des V1 werden hinsichtlich ihrer selektiven Reaktion auf elementare Stimuli unterschieden [Hubel62], [Hubel95], [DeAngelis95], [Carandini2006] [Carandini99] [Daugman85]:

- **Richtung**:

  1. Einfache Zellen haben ein längliches Wahrnehmungsfeld und sprechen auf dementsprechend geformte Bildelemente an. Die Wirkung einfacher Zellen entspricht in etwa den Gabor-Filtern in der digitalen Bildverarbeitung.

  2. Komplexe Zellen fassen die Ausgabe mehrerer einfacher Zellen zusammen. Sie repräsentieren gerichtete Elemente höherer Invarianz.

- **Räumliche Frequenz:** Verschiedene Zellen sind der unterschiedlichen räumlichen Dichte von Bildelementen zugeordnet (z.B. Abstände in einem Balkenmuster).

  - **Gerichtete Bewegung**: Erweiterung des Wahrnehmungsfeldes um die Dimension der Zeit.
  - **Zeitliche Frequenz:** Zeit zwischen Hell/Dunkel-Wechseln.
  - **Okulare Dominanz:** Sehen mit beiden Augen ermöglicht dreidimensionale Wahrnehmung (bei Tieren mit nach vorne gerichteten Augen).
  - **Farbe:** Drei Untertypen spannen einen dreidimensionalen Farbraum auf (ähnlich dem RGB-Raum).

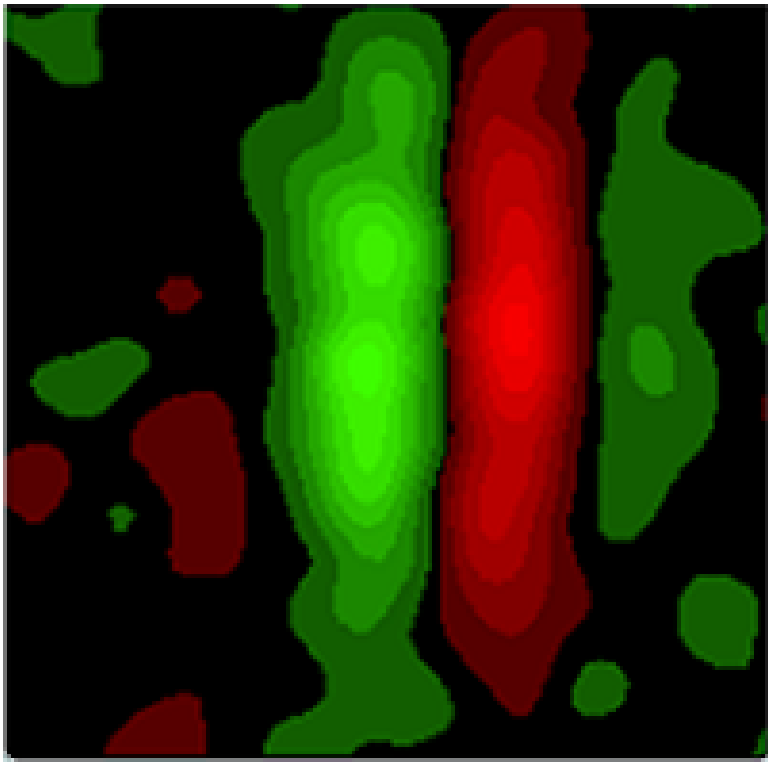

Abbildung 16: Längliches Wahrnehmungsfeld einer "einfachen" Zelle. Rot bedeutet Hemmung (DeAngelis, 1995).

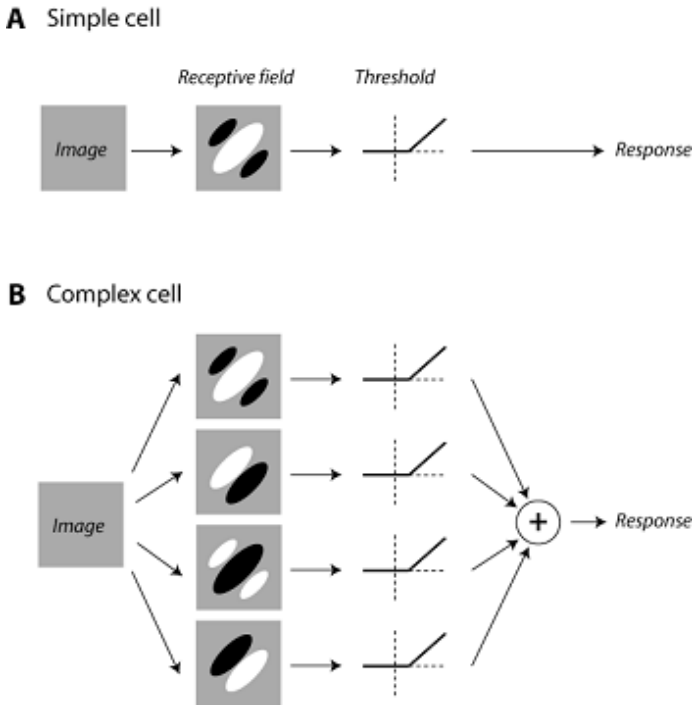

Abbildung 17: Filterfunktion einfacher und komplexer Zellen (Carandini, 2006).

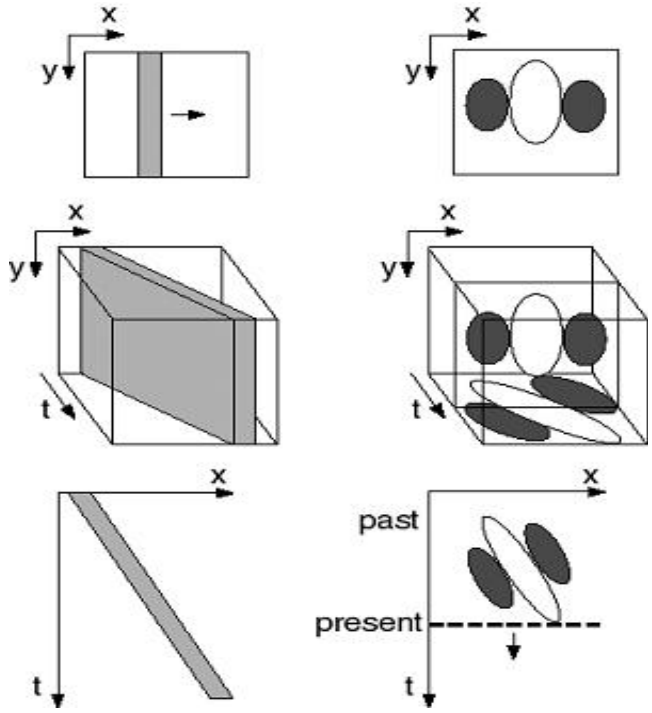

Abbildung 18: Wahrnehmungsfelder in Raum und Zeit ermöglichen die Erkennung von Bewegungen (Carandini, 1999).

Nach der Entdeckung der einfachen Zellen vermutete man zunächst in der Kantenerkennung die Hauptaufgabe von V1. Dies hat die digitale Bildverarbeitung erheblich beeinflusst, wo man aber einsehen musste, dass diese nur ein kleiner Schritt auf dem Weg zum automatisierten Bildverstehen ist. Auch wenn man Teile des V1 unter dem Aspekt der Fourier-Analyse betrachtet, kommt man nicht weit.

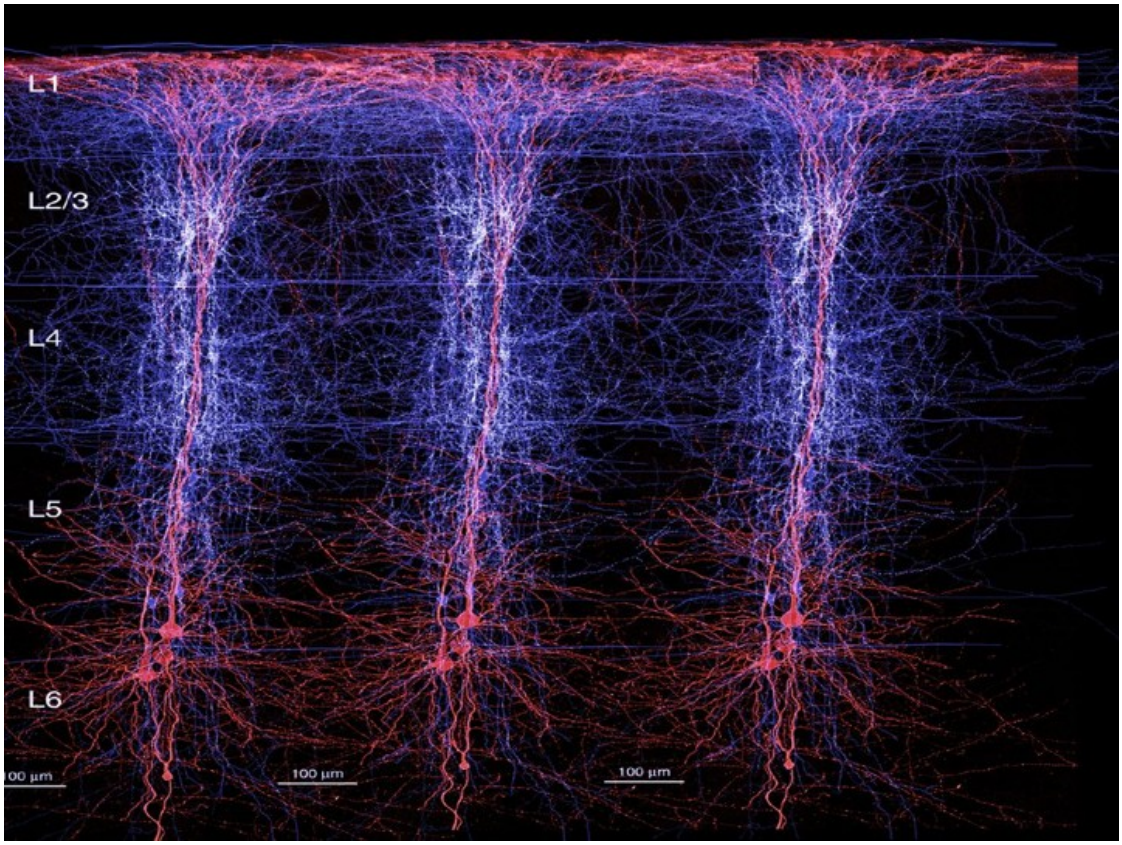

Abbildung 19: Dieser Querschnitt zeigt die sechs kortikalen Schichten. Pyramidenzellen sind rot eingefärbt, hemmende Verbindungen blau. Gut zu erkennen sind drei kortikale Säulen (Blue Brain Project).

3.2 Die kortikale Säule: Ein elementares Berechnungsmodul?

Im Jahre 1955 machte der Neurowissenschaftler Vernon Mountcastle bei der Untersuchung des sensorischen Kortex einer Katze eine bahnbrechende Entdeckung [Mountcastle57], die ansatzweise bereits 1934 von Lorente de No postuliert worden war [No34]. Er versenkte eine Mikroelektrode senkrecht zur kortikalen Oberfläche und zeichnete die gemessenen elektrischen Ströme im Verhältnis zur Eindringtiefe auf, während die Sinnesorgane des Tieres von außen stimuliert wurden. Dabei zeigte sich, dass alle Neuronen innerhalb eines vertikalen Bereichs synchron feuerten. Mountcastle konnte diese Neuronengruppen den Wahrnehmungsfeldern der stimulierten Sinnesorgane zuordnen. Er stellte die Hypothese auf, dass die sich vertikal durch alle Schichten erstreckenden Strukturen eine elementare, diskrete Organisationseinheit des Kortex bilden. Die Zellen einer kortikalen Säule erhalten offenbar dieselben Eingaben, liefern dieselben Ausgaben und sind lateral untereinander verschaltet. Ihr Aufbau ist selbst über Speziesgrenzen hinweg ähnlich. Die Unterschiede liegen vor allem in ihrer Größe (im Durchmesser etwa 0.3 bis 0.8 mm) und der Anzahl der enthaltenen Neurone (80 bis 150).[10]

Mit der Größe des Neokortex variiert die Gesamtzahl der vorhanden Säulen jedoch stark. Je höher entwickelt ein Lebewesen ist, je höher sein kognitives Leistungsvermögen, desto mehr Säulen. Ausgehend von der typischen Größe einer Säule und der Größe des Kortex, vermutet man beim Menschen etwa zwei Millionen kortikale Säulen.

[10] Für kleine Exemplare findet man in der Literatur auch den Begriff Minisäule. Gelegentlich versteht man darunter auch eine Menge aus 50-100 Säulen, die sich zu einer übergeordneten Struktur (Hypersäule) verbinden [Rinkus2010].

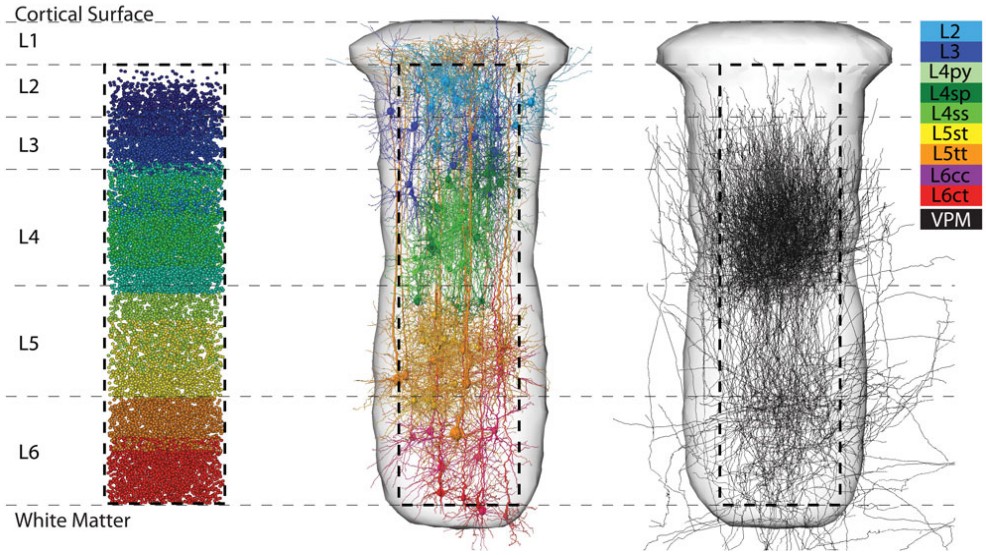

Abbildung 20: Kortikale Säulen gelten als elementare Verschaltungseinheiten der Großhirnrinde. Links: 3D-Rekonstruktion der Nervenzellen in einer kortikalen Säule im sensorischen Kortex einer Ratte. Die verschiedenen Farben geben den jeweiligen Zelltyp wieder. Mitte: Dendriten dieser Nervenzellen. Rechts: Axone (MPI for Neuroscience, Florida).

In den sechziger Jahren konnten Hubel und Wiesel, die für ihre Arbeit 1981 den Nobelpreis erhielten, die Vermutung Mountcastles erhärten. Gegenstand ihrer Untersuchung war der visuelle Kortex von Katzen, der durch im Sehfeld vorbei wandernde Lichtpunkte stimuliert wurde. Gesucht war ein Zusammenhang zwischen der Richtung dieser Bewegung und den dabei aktiven Neuronen.

Wie bei Mountcastle ergab sich auch hier, dass bei gleichen Reizen benachbarte Neurone feuerten, bis am Rande dieser räumlichen Anordnung ein sprunghafter Wechsel der korrespondierenden Reize auftritt. Es zeigte sich, dass die nächste Neuronengruppe auf Bewegungen ansprach, deren Richtung um etwa zehn Grad versetzt war. Neben diesen diskreten, richtungsselektiven Strukturen fand man auch eine streifenförmige räumliche Unterteilung in Neuronen, die stärker auf Reize aus dem linken bzw. rechten Auge reagierten (okulare Dominanz). Dies scheint eine Rolle für das räumlichen Sehen zu spielen. Allerdings keine ausschließliche, denn nicht alle Säugetiere verfügen über okular dominante Säulen. Insgesamt ergibt sich aus richtungsselektiven

und okular dominanten Gebieten somit ein zweidimensionales Gitter, das auch als Eiswürfelmodell des visuellen Kortex bezeichnet wird.

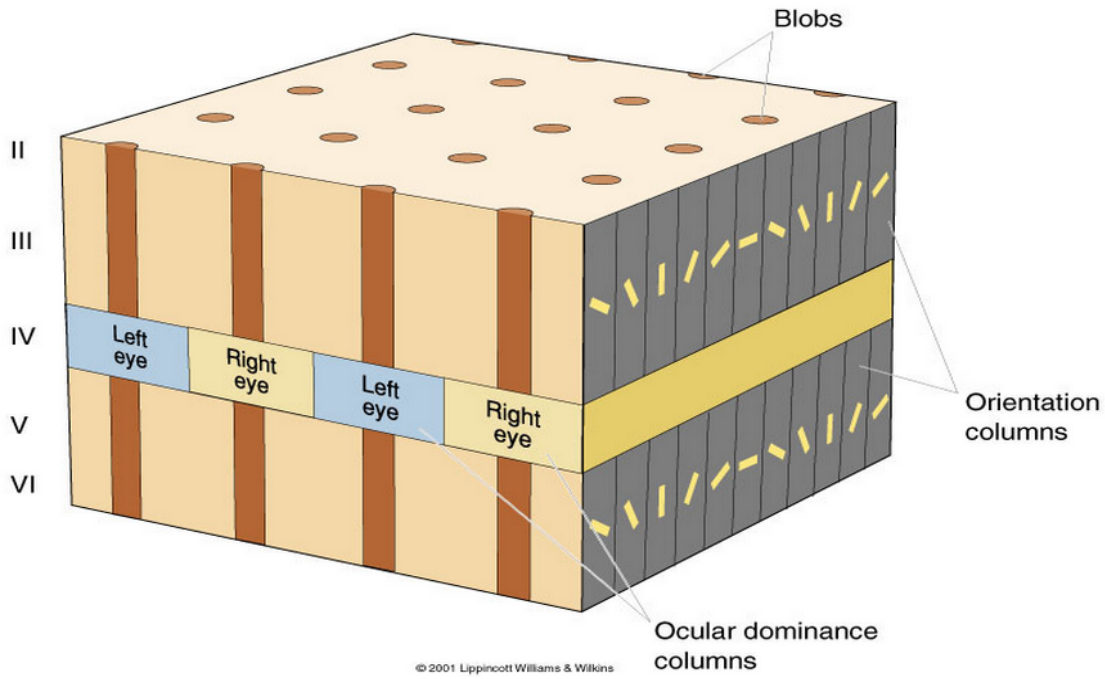

Abbildung 21: Das Eiswürfelmodell des visuellen Kortex. Dieser etwa 1 mm² große Ausschnitt enthält sämtliche neuronalen Schaltkreise, die nötig sind, um einen Teil der visuell wahrnehmbaren Welt zu verarbeiten. L und R beziehen sich auf rechtes und linkes Auge. Die Schrägstriche sollen die Richtung der Lichteindrücke andeuten (Lippincott, Williams & Wilkins., 2001).

3.3 Kritik an der Säulenhypothese

„*Leider scheint sich die Natur unseres intellektuellen Bedürfnisses nach Einfachheit und Allgemeingültigkeit nicht bewusst zu sein, und sehr oft ergötzt sie sich an Komplikation und Vielfalt.*“ [Cajal06]

Das Säulenmodell ist äußerst attraktiv für Neurowissenschaftler, da es ermöglicht, die scheinbar unüberwindliche Komplexität der kortikalen Verdrahtung auf eine Menge parallel arbeitender, weitgehend identischer Module zu reduzieren. Jedes dieser Module berechnet denselben Algorithmus für unterschiedliche, benachbarte Eingaben. Versteht man die Funktionsweise eines dieser Module, so versteht man alle und damit

die des gesamten Neokortex. Und natürlich beruht auch die Grundannahme dieser Arbeit, die Existenz eines universellen kortikalen Lernalgorithmus, auf dieser Vermutung. Allein deshalb muss sie aber nicht richtig sein [Horton2005]. Vielleicht ist all dies nur Wunschdenken und die Natur doch komplexer, die Details wichtiger, die Möglichkeit zur Abstraktion geringer, als wir uns das erhoffen. Folgende Kritikpunkte können nicht von der Hand gewiesen werden :

• Es ist denkbar, dass die anatomisch zweifelsfrei vorhandenen Säulen Artefakte des Wachstumsprozesses sind, somit Strukturen ohne Funktion.
• Bei vielen Spezies (unter anderem bei Nagetieren) fehlt die typische Eiswürfelanordnung, die Hubel und Wiesel im visuellen Kortex von Katzen fanden.
• Betrachtet man die Ränder der Säulen genauer, findet man Überlappungen, d.h. zu mehreren Säulen gehörende Zellen [Tsunoda2011].
• In der DNA ließ sich bisher keine genetische Blaupause zur Konstruktion der Säulen finden.
• Während der geschilderten Experimente waren die Tiere betäubt. Dies kann die Ergebnisse beeinflussen. Z.B konnten die Sinnesorgane nicht explorativ eingesetzt werden.

Insbesondere ein aktuelles Forschungsergebnis relativiert die uneingeschränkte Gültigkeit der Säulenhypothese: 2013 untersuchten Constantinople und Bruno den Informationsfluss im sogenannten Fasskortex von Ratten (so bezeichnet aufgrund der gut erkennbaren Säulen). Von den Sinnesorganen (hier: Schnurrhaare) über den Thalamus gelangt die Information in Schicht IV, dann über die Schichten II und III zu den Ausgabeneuronen in Schicht V. Wenn man Schicht IV mittels Lidocain betäubt, müsste das System funktionsunfähig werden. Das Gegenteil war aber der Fall, und bei weiteren Untersuchungen zeigte sich, dass auch die Schicht V über direkte Eingabeleitungen verfügt [Constantinople2013]. Somit muss die Säulenhypothese modifiziert werden.

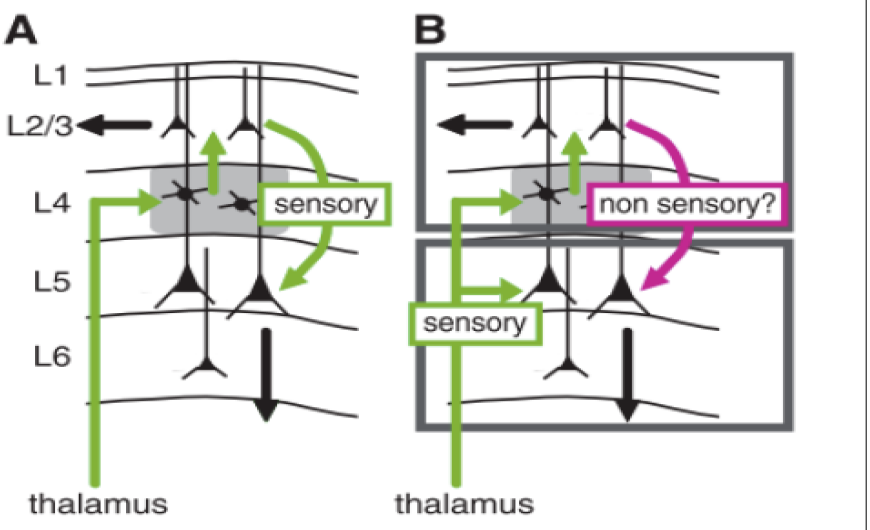

Abbildung 22: A) Bisheriges Modell des Informationsflusses. B) Modifiziertes Modell (Constantinople et al., 2013)

Aber kann man eine Säule überhaupt prinzipiell als ein isoliertes Element beschreiben, wenn man sie - *in natura* wie *in vitro* – nur verbunden mit anderen Säulen beobachten kann? Schließlich sind Säulen untereinander stark gekoppelt, und es ist nicht klar, ob diese Tatsache vernachlässigbar ist. Betrachtet man den Neokortex als nichtlineares System, muss man damit rechnen, dass das Ganze sich anders verhält als die Gesamtheit seiner Teile. Selbst wenn man diese Teile verstünde, stieße man dann an die Grenzen des Reduktionismus. Immerhin reduziert die Modellierung kortikaler Säulen die Anzahl der beteiligten Knotenpunkte etwa um den Faktor 10.000 gegenüber der Modellierung einzelner Neuronen. Natürlich ist es fraglich, biologische Systeme unter der Kategorie der nichtlinearen dynamischen Systeme einzuordnen, da sie sich ja gerade dadurch auszeichnen, dass sie robust gegenüber äußeren Störungen sind.

Alle diese Einwände sind stichhaltig – und wir wissen nicht, welche Erkenntnisse die Zukunft bringen wird. Dennoch scheint die Säulenhypothese Substanz zu haben. Die große Mehrheit der Forscher sieht in ihr mehr als nur das menschlichen Bestreben, Ordnung ins Chaos zu bringen.

3.4 Quasi-realistische Simulation kortikaler Säulen

Jenseits der stark vereinfachten künstlichen neuronalen Netze, wie sie im maschinellen Lernen seit Langem bekannt sind, ist es dank der heute vorhandenen Computerleistung möglich, alles über Nervenzellen vorhandene Wissen in Simulationsmodelle einfließen zu lassen. Dabei wird die große Zahl verschiedener Neuronentypen abgebildet, ebenso wie deren Charakteristika (Aktionspotenziale, elektrische Leitungsprozesse, Neurotransmitter, Ionenkanäle etc.). Für die nahe Zukunft ist geplant, auch die Genexpression auf molekularer Ebene einzubeziehen. Die Vorteile einer derart detailgetreuen Simulation liegen auf der Hand. Mit ihr ist es möglich, das vorhandene Wissen über Neuronen und ihre Netzwerke zu überprüfen. Sollten sich die konstruierten Modelle tatsächlich realistisch verhalten, kann man Experimente durchführen, die im Labor nicht machbar sind. Die Medizin könnte viel über neurologische (Demenz, Alzheimer, Parkinson etc.) und psychiatrische (Depression, Schizophrenie etc.) Krankheiten lernen und pharmazeutische Wirkstoffe erproben. Für das Ziel der Konstruktion intelligenter Maschinen ist folgende Fragestellung von Bedeutung: Inwieweit lässt sich das quasi-realistische Modell auf algorithmischer Ebene vereinfachen, ohne dass es seine Funktion verliert?

Das bekannteste Großprojekt auf diesem Gebiet ist das Blue Brain Project, eine Zusammenarbeit der École polytechnique fédérale de Lausanne und IBM unter Federführung von Henry Markram. Im Jahr 2006 startete das Projekt mit der Simulation einer kortikalen Säule einer Ratte (etwa 10.000 Neuronen und 100 Millionen Synapsen). Das Projekt verwendet das NEURON-Softwarepaket und läuft auf einem Blue Gene Supercomputer. Bis 2014 soll ein komplettes Rattenhirn *in silico* nachgebaut werden, bis 2023 ein menschliches. Eine Erweiterung des Blue Brain Project - unter Einbeziehung einer Vielzahl europäischer Forschungseinrichtungen – ist das Human Brain Project. Es wurde Über einen Zeitraum von zehn Jahren (ab Januar 2013) mit einem Budget von 1 Mrd. € ausgestattet und ist so eines der teuersten und ehrgeizigsten Forschungsprojekte der EU [Markram2012].

Vor Projektbeginn erstellte Markram ein Karte der Vernetzung der kortikalen Neuronen, vor allem mit Hilfe der Patch-Clamp-Technik, für deren Entdeckung Sakmann und Neher 1991 den Nobelpreis erhielten. Dabei werden die Nervenzellen in einer dünnen Schicht aus Kortexgewebe mit einer miniaturisierten, flüssigkeitsgefüllten Pipette berührt, durch die Ströme im Picoamperebereich gemessen werden, die dann einzelnen Ionenkanälen in der Zellmembran zugeordnet werden können. Diese Ionenkanäle sind mit der elektrischen Signalleitung in Dendriten und Axonen verknüpft. So lassen sich ausgehende Signale exakt verfolgen. Beobachtet man das Verhalten der Ionenkanäle mehrerer Zellen, kann man daraus schließen, welche miteinander verbunden sind.

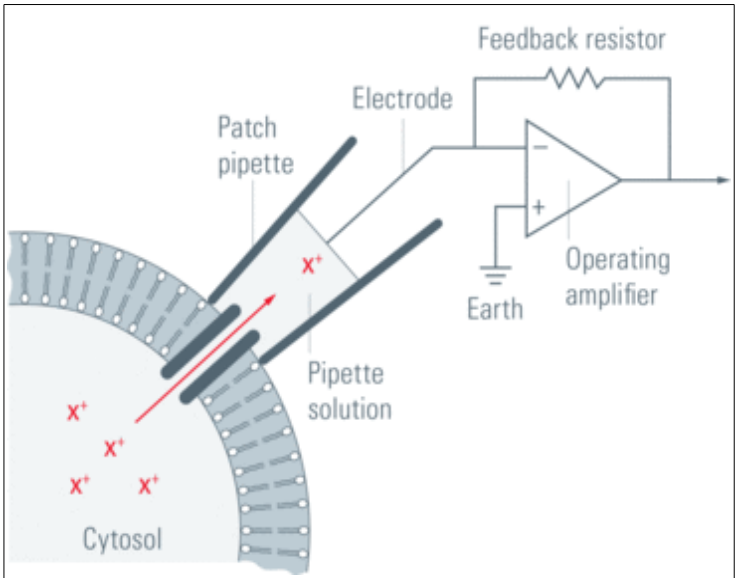

Abbildung 23: Messung des durch einen Ionenkanal fließenden Stromes mit der Patch-Clamp-Technik (Leica Microsystems).

Bei alledem sind einige Worte der Kritik angebracht. Kann man etwas simulieren, das man nicht hinreichend genau kennt? Die Kunst der Simulation besteht darin, ein Modell zu verwenden, das auf elementaren, erprobten und gesicherten Wirkprinzipien beruht und von der Detailebene weg abstrahiert. Kurz: Man simuliert Wichtiges und lässt Unwichtiges weg. Von einer befriedigenden, d.h. mathematisch formulierbaren, Theorie des Neokortex ist man aber noch weit entfernt. Stattdessen füttert man die Rechner mit gewaltigen Mengen an Rohdaten, deren Semantik im Wesentlichen noch

unbekannt ist. Dies bricht mit der u.a. von Karl Popper formulierten deduktiven wissenschaftlichen Methode, aufgrund von Beobachtungen eine Hypothese aufzustellen und diese anschließend durch Experimente zu testen. Die von Markram verwendeten Modelle haben vermutlich viel zu viele offene Parameter, als dass man diese gezielt einstellen könnte. Einen Versuch ist es allemal wert, doch bleibt die Frage, ob die erheblichen Forschungsgelder nicht in der Grundlagenforschung besser angelegt wären.

Für diese Einwände spricht, dass seit Projektbeginn noch keine Ergebnisse veröffentlicht wurden, die Fortschritte nachweisen. Sollten sich die Modelle tatsächlich so verhalten wie ihre biologischen Vorbilder, hätte man damit noch lange keine Theorie des Neokortex. Es wäre wie so oft im maschinellen Lernen: Auch wenn ein System funktioniert, weiß man nicht warum.

Im Report des Human Brain Project an die EU-Kommission von 2012 lesen sich die Zielsetzungen des Projekts dann auch wesentlich bescheidener und realistischer als die marktschreierisch verfassten Pressemeldungen, die durch die Medien gingen [HBP2012]. Unabhängig von seinem weiteren Fortgang ist das Projekt eine enorme Anstrengung, deren Stärke darin besteht, eine große Zahl an Forschern verschiedenster Disziplinen zusammenzubringen.

4 Neuronale Wissensrepräsentation

Wenn wir uns in der Welt bewegen, beobachten, lernen und agieren, führt dies zu einer Änderung unseres neuronalen Zustands. Die äußere Welt wird auf die innere abgebildet. Alle realen Objekte – abstrakte ebenso wie konkrete – besitzen ein neuronales Korrelat. Auch bereits internalisierte Objekte können Gegenstand dieser Abbildung sein und werden dann zu einer Erinnerung höherer Ordnung[11]. Wir alle haben schon die Erfahrung gemacht, dass uns etwas real erscheint, das wir nie erlebt haben, z.B. in einem besonders intensiven Traum. In diesem Kapitel geht es darum, zu untersuchen, wie das Gehirn vorgeht, wenn es interne Darstellungen anlegt (Kodierung). Im umgekehrten Fall versuchen wir zu ermitteln, welches äußere Objekt gegeben war, wenn wir den Zustand mindestens einer Nervenzelle experimentell beobachten können (Dekodierung).

Der neuronale Kode repräsentiert somit Wissen über die Außenwelt oder den Zustand vorgeschalteter Neuronen. Im Sinne Shannons ist er die kleinstmögliche Symbolmenge, die in der Lage ist, alle relevanten Informationen darzustellen.

Wir haben gesehen, dass die Ausgabe von Neuronen (Aktionspotenziale) digitaler Natur ist. Eine Nervenzelle feuert, oder sie feuert nicht. Dazwischen gibt es nichts. Information kann daher nur in der zeitlichen Abfolge der Aktionspotenziale oder im Zustand einer Gruppe von Neuronen zu einem bestimmten Zeitpunkt liegen. Letzteren kann man als Bitvektor betrachten.

[11] Der berühmte Satz Wittgensteins, „Die Welt ist alles, was der Fall ist“, trifft diese moderne erkenntnistheoretische Einsicht wohl am Besten.

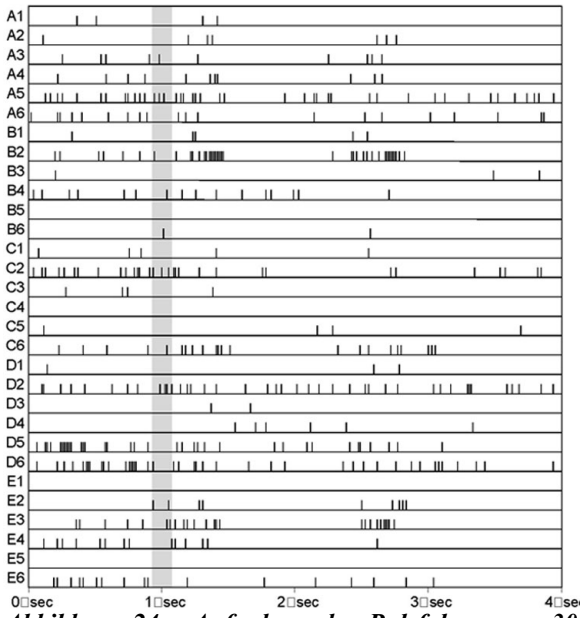

Abbildung 24: Aufnahme der Pulsfolgen von 30 zufällig ausgewählten Neuronen aus dem visuellen Kortex. Man kann dieses Bild wie eine Orchesterpartitur lesen. Ein einzelner Impuls dauert etwa 1 ms. Der graue Balken entspricht einer Zeit von 150 ms, in der beispielsweise ein Gesicht erkannt werden kann (Krüger und Aiple, 1988).

4.1 Frequenzmodulation und temporale Kodierung

1926 beobachteten Adrian und Zotterman, dass die durch einen Muskel ausgeübte Kraft direkt (aber nicht linear) von der Frequenz der an ihm ankommenden Aktionspotenziale abhängt. Sie mutmaßten, dass *allein* diese Frequenz der Informationsträger in der Kommunikation zwischen Neuronen sei, und Variationen in den Impulsabständen zufälliger Natur seien [Adrian26]. Allerdings gab schon Laplace zu bedenken:

"Zufälligkeit ist nur ein Maß unserer Unkenntnis der verschiedenen Vorgänge, die an der Erzeugung von Ereignissen beteiligt sind." (Laplace, 1825)

Hat man es mit einem konstanten oder sich nur langsam verändernden Stimulus zu tun, funktioniert der Ansatz der Frequenzmodulation gut. In der Realität ist eine weitgehend stationäre Eingabe aber selten der Fall [Krüger88]. Das von den Fotorezeptoren der Netzhaut empfangene Bild beispielsweise verändert sich rasch, wenn wir uns bewegen. Und selbst bei der Betrachtung eines statischen Bildes führen die Augen ständig sogenannte Sakkaden aus, kleine Änderungen des Betrachtungswinkels, die die Aufmerksamkeit auf verschiedene interessante Gebiete des Bildes lenken, was auch für das räumliche Sehen hilfreich ist. Da sich das Bild auf der Netzhaut dabei mehrmals in der Sekunde deutlich ändert, bleibt keine Zeit, die erforderlichen Informationen durch die Frequenz zu kodieren. Ein Mechanismus mit einer höheren zeitlichen Auflösung ist nötig, der Fluktuationen in der Zeit zwischen einzelnen Impulsen Bedeutung zumisst. Man spricht dann von temporaler Kodierung [Stein2005].

Ein gutes Beispiel hierfür ist auch das Echolot der Fledermäuse, das eine höchst präzise 3D-Reproduktion ihrer Umgebung erstellt. Im Radius einiger Meter kann es zwischen Objekten differenzieren, deren Entfernung sich nur um wenige Millimeter unterscheidet, was erheblich genauer als die besten technischen Lösungen ist. Offensichtlich ist hierfür nicht nur eine massive Parallelisierung erforderlich, sondern auch asynchrone Signalverarbeitung. Einen zentralen Taktgeber, wie ihn digitale Rechner besitzen, findet man im Gehirn nicht.

Die Erforschung der temporalen Kodierung steht noch am Anfang, da sie voraussetzt, mehrere kommunizierende Neuronen gleichzeitig präzise steuern und belauschen zu können. Herkömmliche elektrische Mikrosonden sind dafür ein zu grobes Instrument. Einen möglichen Ausweg bietet das noch junge Gebiet der Optogenetik. Dabei werden die Ionenkanäle von Nervenzellen durch das Einbringen spezifischer Gene fremder Organismen so modifiziert, dass sie durch Lichtimpulse von festgelegter Wellenlänge aktiviert werden können [Petreanu2007]. Dadurch können einzelne Aktionspotenziale ausgelöst werden, die dann postsynaptisch mit der Patch-Clamp-Technik gemessen werden.

4.2 Populationskodierung

Bei der Kodierung eines Reizes braucht sich das Gehirn nicht auf ein einzelnes Neuron, dessen Signal ja mit Rauschen behaftet ist, zu beschränken, sondern es kann auch eine Gruppe (Population) von Neuronen verwenden, die parallel arbeiten. Da bei diesem Vorgang die Dimensionalität sehr groß wird (Anzahl der Neuronen x Anzahl der Stimuli x Zeit), kommt für seine Beschreibung nur das vereinfachende Modell der Frequenzmodulation in Frage, das jedem Neuron der Gruppe zu einem festen Zeitpunkt (und über ein möglichst kurzes Intervall gemessen) eine Impulsfrequenz zuordnet. Der Populationskode ist somit einfach ein Vektor, dessen Komponenten angeben, wie viele Impulse im Zeitintervall gezählt wurden. Für die Populationskodierung erhält man damit ein mathematisch gut beschriebenes Modell, das es erlaubt für die Kodierung und Dekodierung die Werkzeuge der multivariaten Statistik zu verwenden [Deneve99].

Unter der Tuning Curve eines Neurons versteht man sein Antwortverhalten, d.h. die Frequenz der Aktionspotenziale aufgrund eines parametrisierten Reizes, beispielsweise des Beugungswinkels eines Gelenks, des Temperaturempfindens auf der Haut oder auch der Richtung optischer Reize, wie wir sie bereits kennengelernt haben (vgl. Eiswürfelmodell des primären visuellen Kortex). Typischerweise legt man für statistische Berechnungen als Tuning Curve die Normalverteilung zugrunde, deren Maximum (Erwartungswert) sich linear zum Parameter des Stimulus verhält. Dies kommt dem empirisch ermittelten Verhalten von Neuronen recht nahe.

Für praktische Anwendungen relevant ist die Frage der Dekodierung: Welcher Stimulus hat die aufgezeichneten Impulsfrequenzen einer Gruppe von Neuronen verursacht? Da alle Neuronen unzuverlässige Signale senden, kann man diese Frage nur probabilistisch beantworten. Am einfachsten ist es, hierfür alle Messwerte als Vektoren (mit Länge = Impulsfrequenz und Richtung = Richtung des optischen Stimulus) zu addieren. Besser zur Dekodierung geeignet ist die Maximum-Likelihood-Methode, bei der man den Parameter der Wahscheinlichkeitsverteilung so variiert, dass die beobach-

teten Daten die wahrscheinlichste Erklärung für den entsprechenden Stimulus darstellen.

Die Likelihood-Funktion ist in diesem Fall die bedingte Wahrscheinlichkeit $P(\langle \boldsymbol{r} | s \rangle)$ für das Auftreten der Impulsfrequenzen r_i in der Population unter der Bedingung, dass der Stimulus s gegeben sei. Als Schätzung für den tatsächlichen, unbekannten Stimulus s wird dann derjenige Wert verwendet, der diese Wahrscheinlichkeit maximiert:

$$\widehat{s_{ML}} = \underset{s}{argmax}\, P(\boldsymbol{r} | s)$$

Für die Impulsfrequenzen kann man beispielsweise eine Poisson-Verteilung ansetzen. Wenn man ferner voraussetzt, dass diese voneinander unabhängig sind, erhält man für eine Population von N Neuronen

$$p(\mathbf{r} \mid s) = \prod_{i=1}^{N} \frac{e^{-f_i(s)} f_i(s)^{r_i}}{r_i!}$$

wobei die r_i hier ganzzahlig sind (Anzahl der Impulse). Die $f_i(s)$ sind die Tuning Curves der Neuronen, für die man Messwerte (bei einer numerischen Berechnung) oder wie oben beschrieben eine Gaußsche Glockenkurve als Näherung einsetzen kann.

Ein grundsätzliches Problem ist, dass man mit heutigen Methoden maximal etwa 100 Neuronen simultan beobachten kann, während tatsächlich vermutlich wesentlich mehr an der internen Repräsentation von Reizen beteiligt sind.

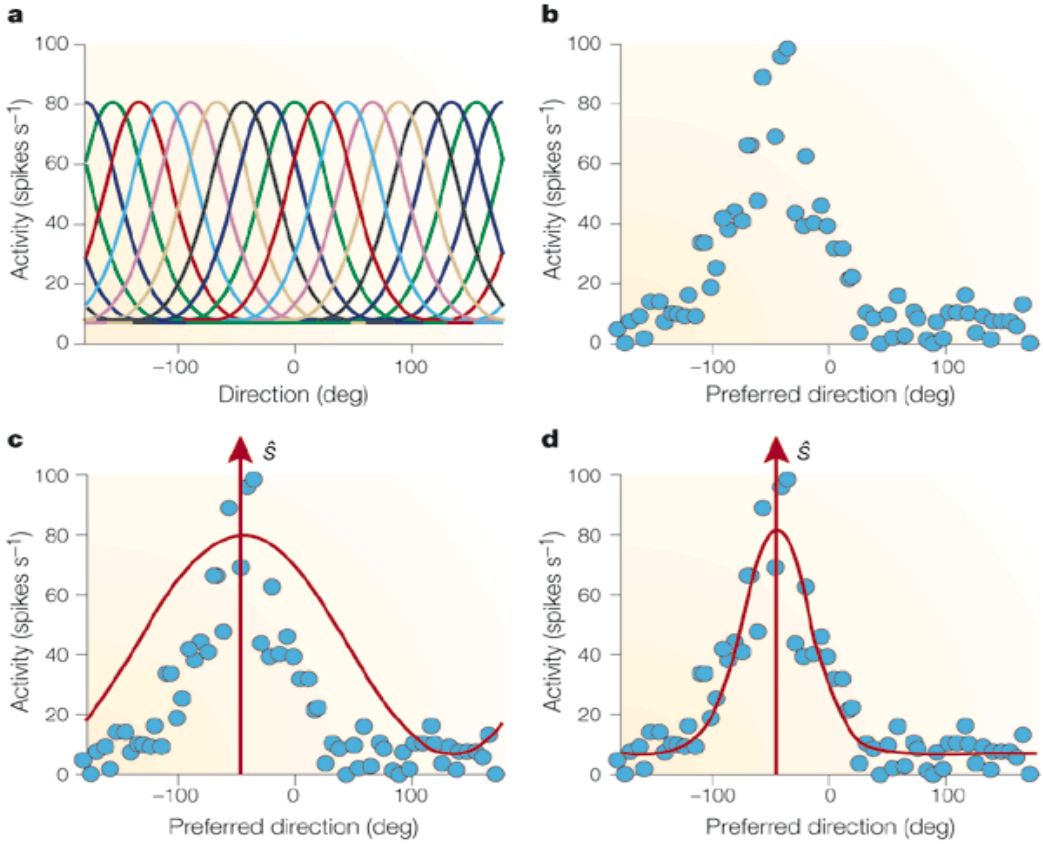

Abbildung 25: Methoden zur Dekodierung des Populationskodes am Beispiel der richtunsselektiven Zellen des V1 a) Richtungsabhängige Tuning Curves diverser Neuronen (als Normalverteilung) b) Signal mit Rauschen von 64 experimentell beobachteten Neuronen bei einem Stimulus im Winkel von -90° c) Dekodierung durch Vektoraddition (äquivalent zum Einpassen einer Cosinus-Kurve) und d) mittels Maximum-Likelihood-Methode (Deneve et al., 1999).

Dieses und ähnliche statistische Verfahren werden für Gehirn-Computer-Schnittstellen und zur Steuerung biomechanischer Prothesen verwendet. Eine moderne Prothese misst fortlaufend die Aktivität mehrerer Nervenleitungen (die sonst an Muskelfasern endeten), interpretiert sie und setzt sie in Steuerbefehle für die eingebauten Aktuatoren um. Im gewissermaßen umgekehrten Fall versucht man, die Datenströme künstlicher Sinnesorgane so zu kodieren, dass sie über eine Schnittstelle im Nervensystem möglichst authentische Sinneseindrücke hervorrufen (vgl. bereits verfügbare Cochlea- und erste experimentelle Retina-Implantate).

5 Sparse Coding und Deep Learning

Im folgenden abstrahieren wir von im zeitlichen Impulsverlauf enthaltener Information und betrachten eine Population von Neuronen als Einheiten, die entweder aktiv oder nicht aktiv sind, was einem Bitvektor entspricht. Welche Kodes sind dann geeignet, Wissen über die Umgebung zu repräsentieren? Welche Vor- und Nachteile haben sie? Außerdem suchen wir nach Verfahren des unüberwachten Lernens[12], um diese Kodes durch Trainieren zu generieren. Beim unüberwachten Lernen geht es darum, selbstständig Muster in den Trainingsbeispielen zu finden. Alternativ kann dies als Erlernen einer Funktion charakterisiert werden, die sich einer A-Priori-Wahrscheinlichkeitsverteilung annähert, deren korrespondierender unbekannter Zufallsprozess die Trainingsbeispiele erzeugt.

5.1 Lokale Kodes und die „Großmutterzelle“

Auf einer Computertastatur ist immer nur eine Taste gedrückt (*Shift*, *Alt* und *Control* ausgenommen). Dies entspricht einem Bitvektor, der aus genau einer Eins und sonst nur Nullen besteht. Man spricht dann von einem lokalen Kode. Er repräsentiert nur sehr wenige Symbole, nämlich eines pro Bit, und hat somit einen sehr hohen Speicherbedarf. Sein Vorteil besteht darin, dass die Zuordnung von Ein- und Ausgabe äußerst einfach ist. Außerdem können mehrere Zeichen gleichzeitig dargestellt werden (maximal: alle). Auf der Computertastatur entspräche das mehreren simultan gedrückten Tasten, was natürlich nicht üblich ist, wohl aber im Falle eines Klaviers.

Die „Großmutterzelle“ (Lettvin, 1969) ist ein hypothetisches Neuron, das genau einem komplexen (aber spezifischen) Gedächtnisinhalt entspricht und aktiv wird, wenn man die Großmutter erblickt, hört oder ertastet. Analog gäbe es dann Neuronen für Vater und Mutter, aber auch für rote Autos, unser Lieblingsessen, etc. Dieser Ansatz versucht das sogenannte Bindungsproblem zu lösen: Wie integriert das Gehirn eine Viel-

[12] Das maschinelle Lernen hat sich überwiegend auf überwachte Verfahren konzentriert, die aber in realistischen Szenarien selten anzutreffen sind.

zahl sensorischer Eindrücke zu einer einheitlichen Wahrnehmung [Bowers2009]? Nicht nur der große Speicherbedarf ist hier ein Einwand. Schwerer wiegt die Tatsache, dass bei einem Ausfall des Großmutterneurons die komplette Erinnerung an die Großmutter verloren wäre. Im Laufe unseres Lebens sterben in der Tat etliche Nervenzellen ab, und offenbar ist unser Gedächtnis in dieser Hinsicht robuster konstruiert. Es hat die Eigenschaft der „graceful degradation“: Einzelne Fehler verringern die Fähigkeiten des Gesamtsystems nur schrittweise und nicht abrupt. Nicht nur biologische Systeme verfügen über diese Eigenschaft, sondern sie wird auch in der Technik angestrebt. Beispielsweise lassen sich CDs auch dann noch abspielen, wenn sie Kratzer aufweisen. Dem liegt ein fehlerkorrigierender Reed-Solomon-Kode zugrunde.

Ein lokales Lernverfahren beruht auf der Forderung, dass für benachbarte Trainingsbeispiele x_i und x_j auch die entsprechenden Ausgabewerte $f(x_i)$ und $f(x_j)$ nahe beieinander liegen sollten. Ein solcher Algorithmus konvergiert in der Regel schnell und eignet sich gut zur Interpolation, aber er bringt auch das Problem des Overfitting mit sich: Er kann nicht über die Trainingsdaten hinaus extrapolieren. Bei mit Rauschen behafteten Daten wird er das Rauschen als Teil des Signals interpretieren. Ein Beispiel hierfür ist der unüberwachte k-Means-Algorithmus zur Clusterbildung, der den Eingaberaum in k Regionen aufteilt, in deren Zentrum Cluster der Trainingsbeispiele liegen. Vor allem wenn der Parameter k zu groß gewählt ist, kann der Algorithmus neue Datenpunkte nicht sinnvoll den erzeugten Clustern zuordnen (Generalisierung).

5.2 Dicht verteilte Kodes

Gewissermaßen das gegenteilige Extrem ist ein Kode, der Information dicht auf mehrere gleichzeitig aktive Einheiten verteilt. Er geht sparsam mit dem zur Verfügung stehenden Speicher um, da er einen hohen Informationsgehalt hat. Der übliche Binärkode beispielsweise kann mit *n* Bits 2^n Symbole repräsentieren. Er nutzt die kombinatorischen Möglichkeiten eines Bitvektors maximal aus, d.h. im Mittel sind n/2 Bits

gesetzt. Auch der bekannte ASCII-Kode fällt in diese Kategorie. Natürlich können mit einem derartigen Kode nicht mehrere Symbole gleichzeitig dargestellt werden.

Die Abbildung zwischen einer verteilten Repräsentation und ihrer Ausgabe, d.h. die Menge der repräsentierten Objekte, kann sehr komplex sein und ein vielschichtiges neuronales Netz erfordern. Entsprechende Lernalgorithmen (z.B. Backpropagation) sind langsam, selbst für überwachtes Lernen. Allerdings haben diese Lernverfahren die Fähigkeit zur Generalisierung: Sie können brauchbare Vermutungen über Teile des Eingaberaumes anstellen, die sie anhand der Trainingsbeispiele nicht kennen.

Wenn n groß ist (und die Zahl der darstellbaren Zustände damit exponentiell) wird die Kapazität eines verteilten Kodes nicht genutzt, da die Zahl der vom System tatsächlich beobachteten Zustande sich niemals dieser Kapazität annähern wird. Dieser Umstand ist ein Hinweis darauf, dass Raum für Verbesserungen existiert.

5.3 Sparse Coding: Energieeffizienz im Kortex

Sparse Coding ist der Versuch, die Vorteile lokaler und dicht verteilter Kodes zu kombinieren, ohne deren Nachteile hinnehmen zu müssen, und ist somit ein Kompromiss zwischen zwei Extremen. Während bei einer lokalen Repräsentation nur eines aus n Bits gesetzt ist und bei einer dicht verteilten im Mittel n/2 Bits, wählt ein Sparse Code eine in der Regel feste Anzahl an Bits aus, die deutlich kleiner als n/2 ist.

	Kapazität	***Speicherbedarf***	***Lerngeschwindigkeit***	***Generalisierung***	***Fehlertoleranz***	***mehrere Objekte pro Bitvektor***
lokal	sehr niedrig: O(n)	hoch: O(n)	sehr schnell	keine	keine	maximal n
sparse	hoch	mittel	schnell	mittel	hoch	einige
dicht verteilt	sehr hoch: O(2^n)	niedrig: O(log n)	langsam	hoch	sehr hoch	nur eines

Im Kortex garantieren hemmende Neuronen, dass immer nur ein kleiner Prozentsatz der Gesamtpopulation aktiv ist. Derselbe Mechanismus wird auch im weiter unten beschriebenen HTM-Algorithmus verwendet. Die Eingabe einer HTM-Region ist immer ein dicht verteilter Bitvektor, der in einen Sparse Code umgewandelt wird.

Sei die Eingabe beispielsweise 20.000 Bit lang. Die Anzahl der aktiven Einheiten variiert im zeitlichen Verlauf im Allgemeinen stark. Bei der dann erzeugten internen Repräsentation sind typischerweise 2% von 10.000 Bit aktiv[13]. Die Eingabe ändert sich mit der Zeit, aber im erzeugten Kode werden immer etwa 200 Bit gleichzeitig aktiv sein.

Nun hat es den Anschein, dass dieser Prozess, der in unserem Beispiel die Eingabelänge ja halbiert, einen großen Verlust an Information mit sich bringt. Tatsächlich aber ist die Kapazität der komprimierten Darstellung immer noch ausreichend groß. Die Eingaben, denen das System tatsächlich begegnet, stellen nur einen winzigen Bruchteil des Raumes aller möglichen Eingaben dar. Somit hat der Informationsverlust keine negativen Konsequenzen.

Historisch gesehen kann man einen weiten Bogen spannen, an dessen Beginn die Entdeckung richtungsselektiver Zellen im primären visuellen Kortex durch Hubel und Wiesel steht. Zu Beginn der Neunzigerjahre stellten sich Ohlshausen und Field an der Cornell University die Frage, ob es einen tieferen Sinn hat, dass diese Zellen ausgerechnet Richtungen (Linien in verschiedenen Winkeln) erkennen können [Olshausen97] [Olshausen96]. Sie stellten einen Satz Grauwertbilder alltäglicher Szenen zusammen (Bäume, Blätter, Gesichter, Tiere, Berge etc.) und extrahierten daraus tausende 16 × 16 Pixel große Teilbilder. Nun stelle man sich vor, man habe 400 Dias zur Verfügung, die ebenfalls 16 × 16 Pixel groß sind. Ziel ist es, eine Teilmenge der 400 Dias übereinander zu legen und dadurch jedes der Teilbilder rekonstruieren zu können. Ma-

[13] Der Anteil aktiver Einheiten beruht auf Erfahrungswerten. Es gibt Schätzungen, wonach im Kortex immer etwa 1% der Nervenzellen aktiv sind.

thematisch gesprochen entspricht dies einer Linearkombination aus Basisvektoren. Welche Dias, d.h. Basisvektoren, würde man wählen?

Zu dieser Aufgabe gibt es eine triviale Lösung: Da jedes Dia einem 256 Bit langen Bitvektor entspricht, nummeriere man die Dias von 1-256, und setze bei jedem genau ein Bit. Die Dias 257-400 bestehen nur aus Nullen. Zweifellos kann man mit diesen 400 Dias jedes beliebige 16 × 16 Pixel große Bild erzeugen. Man kann damit nicht nur „realistische“ Bilder erzeugen, sondern auch solche, die nur aus Zufallsbits bestehen. Und genau hierin liegt ein Hinweis darauf, dass die Natur anders vorgeht. Worin liegt der Nachteil dieser simplen Lösung? Wenn Neuronen feuern, verbrauchen sie Energie. Wünschenswert wäre es also, dass möglichst wenige Neuronen gleichzeitig aktiv sind [Rehn2007]. Für unser Beispiel bedeutet das, möglichst wenige der 400 Dias für die Rekonstruktion der Bilder zu verwenden. Für manche Bilder benötigt man sehr wenige Dias, für andere etwas mehr. Formal gesprochen suchen wir eine Basis, die a) möglichst genaue Rekonstruktionen ermöglicht und b) die Sparseness-Eigenschaft hat. Diese besteht darin, dass für realistische Bilder möglichst viele der Koeffizienten der Linearkombination den Wert Null haben. Darüber hinaus liefern Sparse Codes besonders kurze Koeffizientenvektoren für häufig auftretende Bilder, was man mit der Huffman-Kodierung vergleichen kann.

Sei **x** das zu erzeugende Bild und $\tilde{\boldsymbol{x}}$ eine Annäherung daran, dann ist folgendes lineare Erzeugendensystem gesucht:

$$\boldsymbol{x} \sim \hat{\boldsymbol{x}} = D\,\boldsymbol{b}$$

Die Matrix D besteht aus den Basisvektoren der Abbildung. Sie ist ein Analogon zu den Wahrnehmungsfeldern kortikaler Neuronen. Der Vektor **b** enthält die Gewichte der Linearkombination, d.h. die Koordinaten bezüglich der Basis D. Man kann ihn als neuronale Repräsentation eines Sinneseindruckes interpretieren. Im Bayesschen Sinne ist die A-Priori-Wahrscheinlichkeit für die neuronale Aktivität der Population gegeben

durch $p(\boldsymbol{b})=\prod_i p(b_i)$, wenn man annimmt, dass die einzelnen Einheiten voneinander unabhängig sind. Der Index i läuft hier und im Folgenden immer über die Population. Die multivariate Wahrscheinlichkeitsverteilung zwischen Eingaben und neuronalen Repräsentationen ist dann nach der Produktregel der Wahrscheinlichkeitsrechnung

$$p(\boldsymbol{x},\boldsymbol{b})=p(\boldsymbol{x}|\boldsymbol{b})\prod_i p(b_i) \quad ,$$

wobei $p(\boldsymbol{x}|\boldsymbol{b})$ die Likelihood-Funktion ist. Nun kann man das Kodierungsproblem als Suche nach der wahrscheinlichsten neuronalen Repräsentation einer gegebenen Eingabe auffassen. Mit anderen Worten, es soll die A-Posteriori-Wahrscheinlichkeit

$$p(\boldsymbol{b}|\boldsymbol{x})=\frac{p(\boldsymbol{x}|\boldsymbol{b})\,p(\boldsymbol{b})}{p(\boldsymbol{x})}$$

maximiert werden (Bayessche Regel). Dabei spielt p(**x**) keine Rolle, da es für eine konkrete Eingabe konstant ist. Die A-Posteriori-Wahrscheinlichkeit ist somit zur oben angegebenen multivariaten Wahrscheinlichkeitsverteilung proportional. Da alle auftretenden Terme multiplikativ sind, genügt es, die Energiefunktion

$$E(\boldsymbol{b})=-\log(p(\boldsymbol{x},\boldsymbol{b}))$$

zu minimieren. Die Minimierung läuft über D, d.h. die gesuchten Basisvektoren. Wir verwenden eine Gaußsche Likelihood-Funktion und machen den Ansatz einer verallgemeinerten Laplace-Verteilung $p(b_i)\propto\exp(-\theta f(b_i))$, wobei θ ein freier Parameter ist. Insgesamt erhalten wir

$$E(\boldsymbol{b})=\frac{1}{2}\sum_i (x_i-\hat{x}_i)^2+\theta\sum_i f(b_i) \quad .$$

Hierbei ist der erste Term auf der rechten die Log-Likelihood-Funktion der Normalverteilung und steht für den quadratischen Fehler in der Güte der Reproduktion. Er entspricht dem Ausdruck $\|\boldsymbol{x} - D\boldsymbol{b}\|_2^2$, den man aus der Hauptkomponentenanalyse (engl. PCA) kennt.

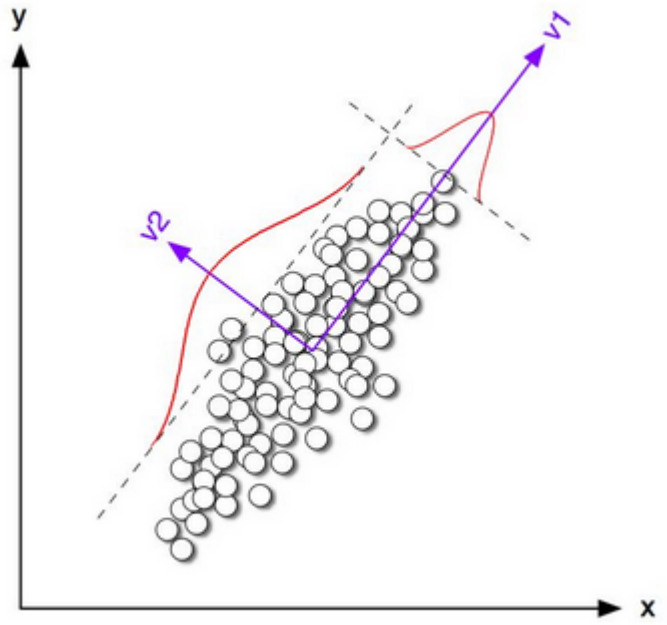

Abbildung 26: Die Hauptkomponentenanalyse (hier in zwei Dimensionen) sucht in meist hoch dimensionalen Punktmengen nach linear unabhängigen Unterräumen mit möglichst großer Varianz. Man erreicht dies durch eine Hauptachsentransformation mittels der Eigenvektoren der Kovarianzmatrix (Wikipedia).

In der Tat ist Sparse Coding eine Verallgemeinerung der PCA, die einen zweiten Term in das Minimierungsproblem aufnimmt, der - bildlich gesprochen - große Koeffizienten bestraft und dadurch die angestrebte Sparseness-Eigenschaft garantiert. Der Parameter θ wird als Sparseness-Parameter bezeichnet. Er bestimmt die Balance zwischen den beiden Minimierungsdirektiven [Rehn2007].

Wir haben hiermit Sparse Coding auf ein Optimierungsproblem zurückgeführt, das im Allgemeinen aber schwer zu lösen ist. Wie schwer, hängt davon ab, welche Annah-

men man über die Funktion $f(b_i)$ macht. Wenn sie differenzierbar ist, lässt sich z.B. das Gradientenabstiegsverfahren einsetzen. Eine genauere Lösung erhält man, wenn man für f die L_1-Norm verwendet. Zwar existieren diverse exakte Lösungen, doch reicht es für die meisten praktischen Anwendungen aus, Näherungsalgorithmen zu verwenden, wie auch im weiter unten beschriebenen HTM-Algorithmus. Die Vorstellung, dass der Kortex Minima der Energiefunktion $E(\boldsymbol{b})$ präzise berechnet, ist abwegig. Sparseness ist ein emergentes Phänomen eines komplexen dynamischen Systems. Evolutionär betrachtet, hatten Organismen, die für dieselbe kognitive Leistung weniger Energie brauchten, einen Vorteil.

Sparse Coding erzeugt Sparse Distributed Representations (SDR). Eine SDR kann mehrere Merkmale simultan darstellen, die sich nicht gegenseitig ausschließen, ja sogar statistisch voneinander abhängig sein können. Hier zeigen SDR-basierte Verfahren ihre Verwandtschaft zur Hauptkomponentenanalyse und zu Algorithmen, die mehrere, global verteilte Cluster erzeugen (z.B. Vektorquantisierung) [Bengio2009]. Wegen ihrer Fähigkeit zur Dimensionsreduktion, können SDR-Verfahren deshalb ein Mittel sein, um dem sogenannten "Fluch der hohen Dimension" zu begegnen. Bei hoch dimensionalen Eingabedaten wächst das Volumen, in dem sich Datenpunkte bewegen können exponentiell, und es wird immer schwieriger, sie zu gruppieren. Glücklicherweise liegen auch in Zustandsräumen hoher Dimension die Datenpunkte oft auf einer niedrig dimensionalen Mannigfaltigkeit, d.h. einem Unterraum der lokal euklidische Charakteristika aufweist [Saul2003].

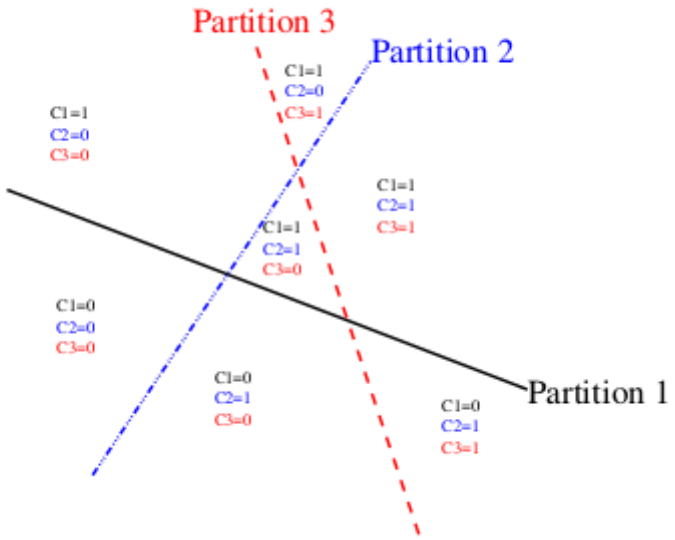

Abbildung 27: Mit nur drei Merkmalen erhält man in diesem Beispiel sieben unterscheidbare Gebiete im Zustandsraum (Bengio, 2009).

Eine Basis, die mehr Elemente hat als der zugehörige Vektorraum Dimensionen, nennt man Englischen „overcomplete“. In der Kodierung eingesetzt bietet diese Redundanz größere Robustheit gegenüber Fehlern und Rauschen, ermöglicht Sparseness und flexibles Pattern Matching. Da ein Objekt durch einen Vektor repräsentiert wird, dessen Komponenten mehrheitlich gleich Null sind, kann man ihn effizient speichern, indem man nur die von Null verschiedenen Eintrage zusammen mit ihrem Index speichert. Jeder Eintrag hat eine semantische Bedeutung, d.h. er steht für eine Eigenschaft des Objekts. Überlagert man mehrere Vektoren (bei Bitvektoren durch eine UND-Verknüpfung), so kann man gemeinsame Eigenschaften direkt ablesen. Verknüpft man mehrere Bitvektoren mittels ODER, dann kann man sofort erkennen, ob ein Kandidat zur Vereinigungsmenge der korrespondierenden Objekte gehört. Nicht nur in der Biologie, auch in der Informatik sind dies bemerkenswerte Eigenschaften.

Es gibt Schätzungen, wonach neuronale Repräsentationen im V1 um den Faktor 500 redundant sind [Földiak2008]. Das hieße, dass an der Darstellung eines der oben

erwähnten 16x16 Pixel großen Bildausschnitte etwa 128.000 Nervenzellen beteiligt wären. Sparse Coding beschränkt sich nicht auf Bildverarbeitung und -erkennung. In der Audiokodierung bietet es die Möglichkeit intelligenterer, musikalischerer Repräsentationen, die eher dem menschlichen Hören entsprechen [Cornuz2007].

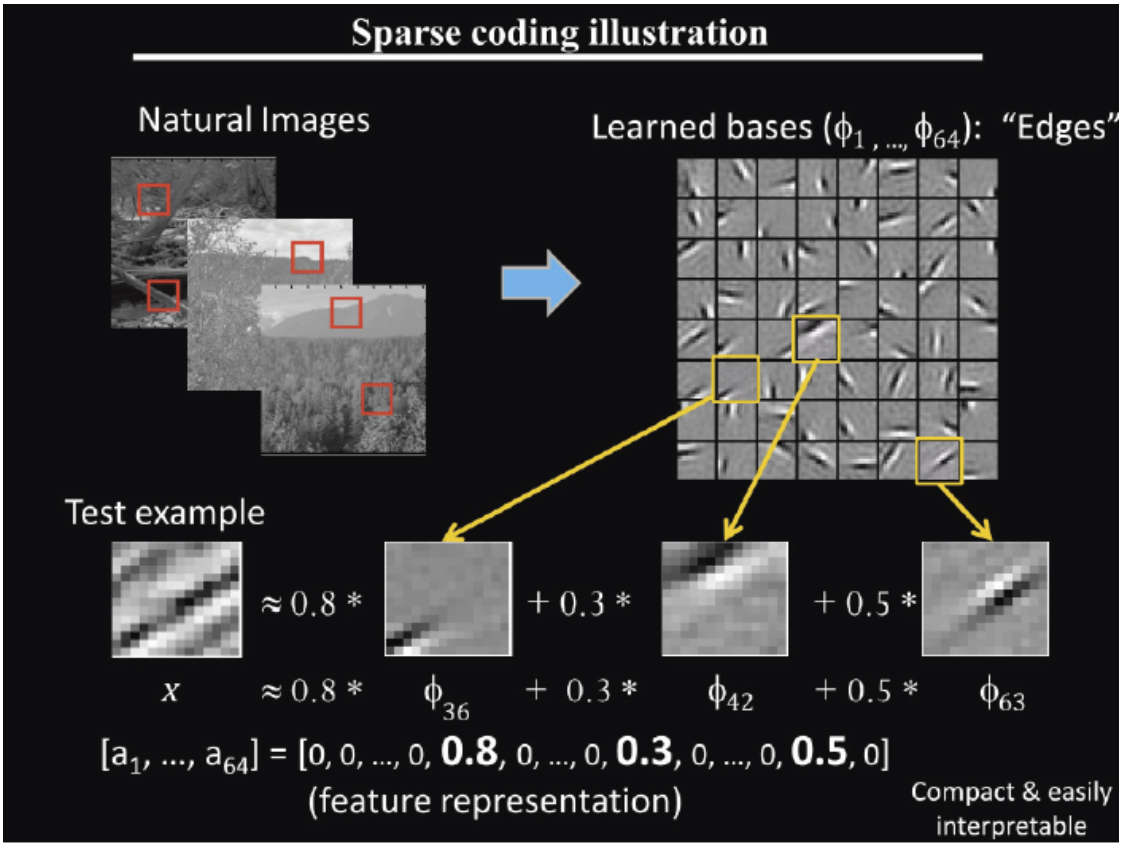

Abbildung 28: Olshausen und Field konnten zeigen, dass Sparse Coding (auf realistische Bildfragmente angewendet) tatsächlich zu Strukturen führt, die gerichteten Linien ähneln. Intuitiv erinnert dieser Vorgang an eine Wavelet-Transformation (Laserson, 2007).

5.4 Deep Learning: Sparse Coding in mehrstufigen neuronalen Netzen

Die Geschichte der künstlichen neuronalen Netze ist von großen Hoffnungen und herben Rückschlägen geprägt. Bereits 1969 bewiesen Minsky und Papert, dass Perzeptronen-Netze (ein Konzept aus den 50er Jahren) linear nicht separierbare Funktionen nicht oder nur mit großem Aufwand berechnen konnten [Minsky69][14]. Obwohl dieser Makel relativ leicht behoben werden konnte, wurden dringend benötigte Forschungsgelder gestrichen. Es folgte der berühmte „KI-Winter".

In den 80er Jahren flammte das Interesse an neuronalen Netzen durch das Bekanntwerden des eleganten Backpropagation-Algorithmus wieder auf, der Minima der Fehlerfunktion mittels Gradientenabstieg sucht. Doch dann zeigte sich, dass dieser Gradient beim Durchlaufen mehrstufiger Netze so klein wird, dass eine Minimierung aus numerischer Sicht kaum noch möglich ist [Hochreiter91]. Diese Erkenntnis stellte das gesamte Backpropagation-Verfahren in Frage. Neuronale Netze erwiesen sich bezüglich ihrer Tiefe als schlecht skalierbar.

Erst durch die gewaltige Steigerung der Rechenleistung mittels massiver Parallelisierung, wie sie insbesondere kostengünstig durch Standard-GPUs möglich geworden ist, erlebt das klassisches Backpropagation-Verfahren zur Zeit wieder eine Renaissance [Ciresan2010]. Ein großes Manko von Backpropagation und verwandten Methoden bleibt aber bestehen: Sie eignen sich nur zum überwachten Lernen. Jedoch sind Daten, die bereits in Kategorien eingeteilt wurden, die Ausnahme. Wogegen nicht klassifizierte Rohdaten jeder Art in großer Menge zur Verfügung stehen.

Heute ist bekannt, dass ein simples Feed-Forward-Netz mit einer einzigen verborgenen Schicht jede stetige Funktion auf einer kompakten Teilmenge des R^n approximieren kann (Universal Approximation Theorem) [Hornik91]. Wozu besteht dann überhaupt ein Bedarf für tiefere Netze?

[14] Was Minsky und Papert im Detail bewiesen haben, war weder überraschend noch wirklich neu. Ihr vielzitiertes (und vermutlich selten gelesenes) Buch war in erster Linie für Dritte ein Vorwand, um Forschungspolitik zu betreiben.

Für ein praktikables Verfahren ist nicht nur seine Berechenbarkeit von Bedeutung, sondern auch seine Komplexität. Es gibt unter diesem Aspekt gute Gründe für tiefe Netze:

- Zweistufige Netze aus Logik-Gattern brauchen im Allgemeinen exponentiell viele Knoten (bezüglich der Eingabelänge), um Boolesche Funktionen zu berechnen [Wegener87].
- Manche Funktionen (z.B. die Paritätsfunktion), die von d-stufigen Netzen mit polynomialer Gatteranzahl berechnet werden können, benötigen eine exponentielle Zahl an Gattern, wenn man die Tiefe auf d-1 beschränkt [Hastad86].

Bevor man überhaupt beginnt, sich mit maschinellem Bildverstehen zu beschäftigen, muss man sich fragen, was man zu sehen erwartet. Welche Struktur hat die Welt? Nach welchen Prinzipien sind Objekte aufgebaut? Man kann Bildverstehen als die Umkehrung der Computergrafik definieren, oder um es poetischer zu formulieren:

> *„Du gleichst dem Geist, den Du begreifst!“* (Goethe, *Faust I*)

Mit Hilfe dieses Gedankens kommt man schnell zu der Erkenntnis, dass die Welt hierarchisch organisiert ist[15]. Große Objekte sind aus Teilen zusammengesetzt, die ihrerseits Objekte sind und aus Teilen bestehen. Ein Gesicht hat Augen, Mund, Nase und Ohren. Bäume bestehen aus Blättern, Stamm und Ästen. Ingenieure und Softwareentwickler zerlegen komplexe Maschinen und Programme in Teilmodule. Ein komplexer Klang besteht aus Akkorden, die von verschiedenen Instrumenten gespielt werden, deren Töne man in Sinusschwingungen zerlegen kann.

Wenn wir uns neue Fertigkeiten aneignen, erlernen wir zuerst einfache Konzepte und setzen diese dann zu komplexeren zusammen. Vorwissen ist von großer Wichtig-

[15] Und übrigens auch unser Wahrnehmungsapparat. Darin besteht der Bezug zum Zitat von Goethe: Unser kognitives System und die sinnlich erfassbare Welt sind zueinander isomorph.

keit, denn es erlaubt, anhand relativ weniger Beispiele Neues zu erlernen. Möglicherweise werden wir mit elementarem Wissen über die Struktur der Welt geboren, quasi mit einer axiomatischen A-Priori-Wahrscheinlichkeitsverteilung. Selbst moderne maschinelle Lernverfahren benötigen dagegen sehr viele Trainingsbeispiele, da es schwer ist, Vorwissen zu integrieren. Eine erwähnenswerte Ausnahme bildet die induktive Logikprogrammierung, die Sätze der Prädikatenlogik nicht nur durch Beispielaussagen, sondern auch mittels Hintergrundwissen erlernen kann.

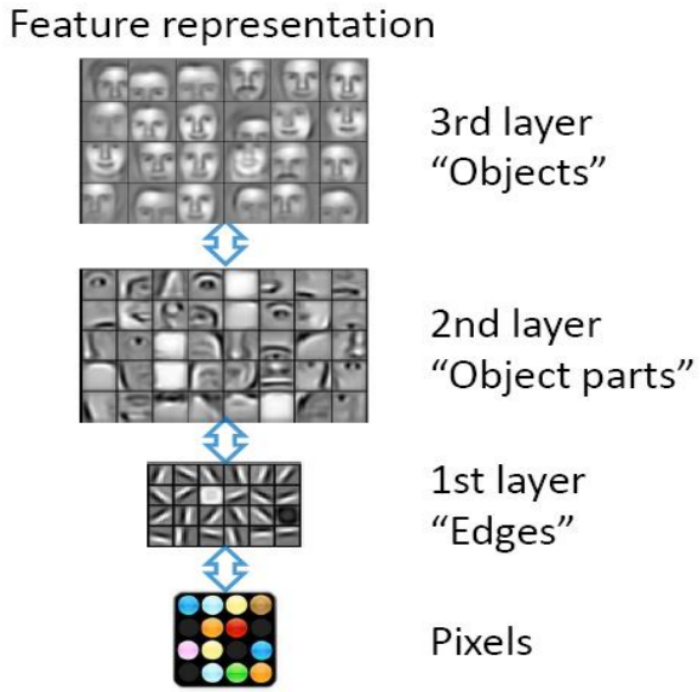

Abbildung 29: Hierarchisch strukturiertes Sehen (Lee, 2010).

Der Urvater tiefer Architekturen dürfte das Neocognitron-Netz sein [Fukushima80]. Angelehnt an die Entdeckungen von Hubel und Wiesel im V1 besteht aus es zwei Knotentypen: S-Zellen (von engl. „simple“) sprechen auf Merkmale in ihrem Wahrnehmungsfeld an (z.B. Linien), nachgeschaltete C-Zellen („complex“) sorgen für Invarianz unter bestimmten geometrischen Transformationen. Mehrere dieser Doppelschichten können aufeinander gestapelt werden, um komplexe Merkmale durch einfa-

chere zu kodieren. Unter anderem hat sich dieser Aufbau zur Handschrifterkennung bewährt.

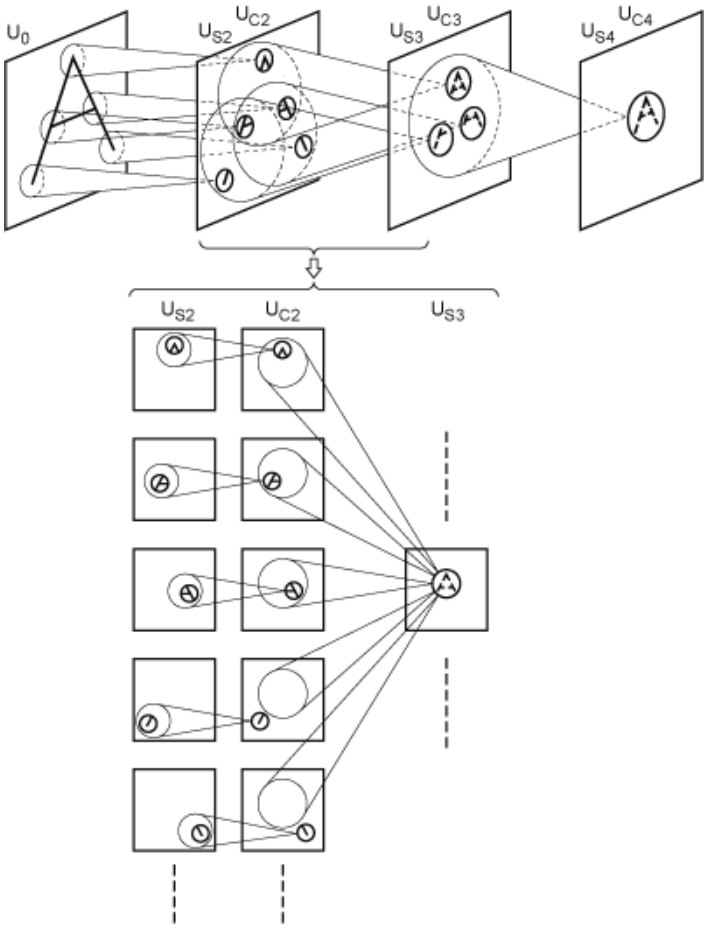

Abbildung 30: Schema eines Neocognitron-Netzes (Scholarpedia).

Einen Schritt weiter in dieser Richtung geht das HMAX-Modell [Riesenhuber99], das am hierarchischen und laminaren Aufbau des visuellen Kortex orientiert ist. Wie das Neocognitron verfügt es über S- und C-Zellen (die hier aber anders konstruiert sind) und als oberste Schicht sogenannte VT-Zellen (View Tuned), die auf 2D-Ansichten dreidimensionaler Objekte ansprechen.

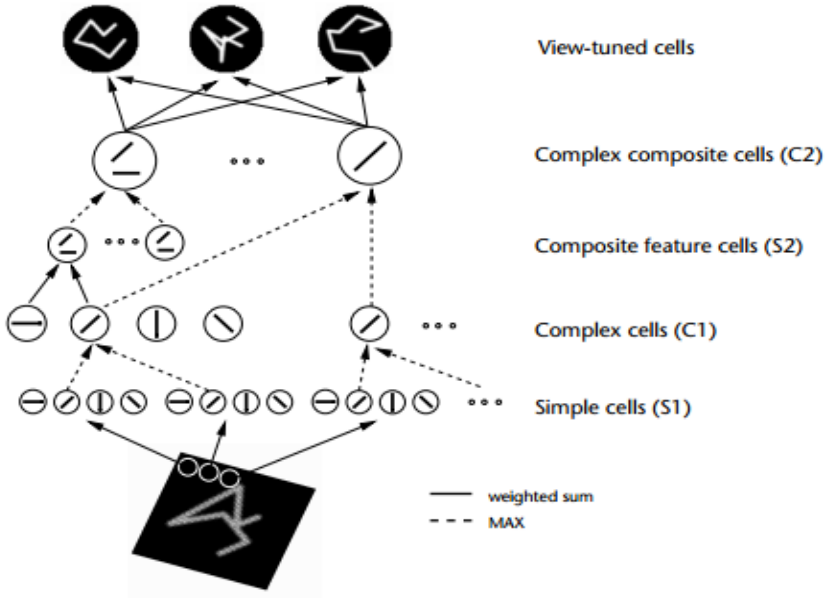

Abbildung 31: HMAX-Netz (Riesenhuber, 1999).

Der entscheidende Nachteil solcher Systeme, besteht darin, dass die Merkmale (engl. „features“) von Hand programmiert werden müssen, sodass sie nicht flexibel genug sind, um für andere Aufgaben (mit anderen Merkmalen) eingesetzt werden zu können. Wie wir gesehen haben, kann dem durch Sparse Coding in Verbindung mit unüberwachtem Lernen möglicherweise abgeholfen werden.

Geoffrey Hinton und einigen anderen Forschern gelang im Jahr 2006 ein großer Schritt hin zu tiefen Netzarchitekturen, die sich unüberwacht trainieren lassen [Hinton2006] [LeCun2006] [Bengio2007]. Ihre Neuerungen bestanden im Kern darin, nicht das gesamte Netz in einem Durchgang zu trainieren, sondern Schicht für Schicht:

- Trainiere die unterste Schicht unüberwacht.
- Erzeuge Sparse Distributed Representations für die unterste Schicht.
- Trainiere dann die darüber liegende Schicht mit den bereits erlernten Repräsentationen der Vorgängerschicht als Eingabe. Wiederhole dies, bis das Netz durchlaufen ist.

- Feinjustierungen (vor allem der obersten Schicht zum Zweck der Klassifikation) können von Hand vorgenommen werden.

Dieser Algorithmus ist schnell, da er im Wesentlichen „greedy“ abläuft. Das Substrat dieser Konstruktion ist ein spezieller Typ rekurrenter, stochastischer neuronaler Netze, die Boltzmann-Maschine.

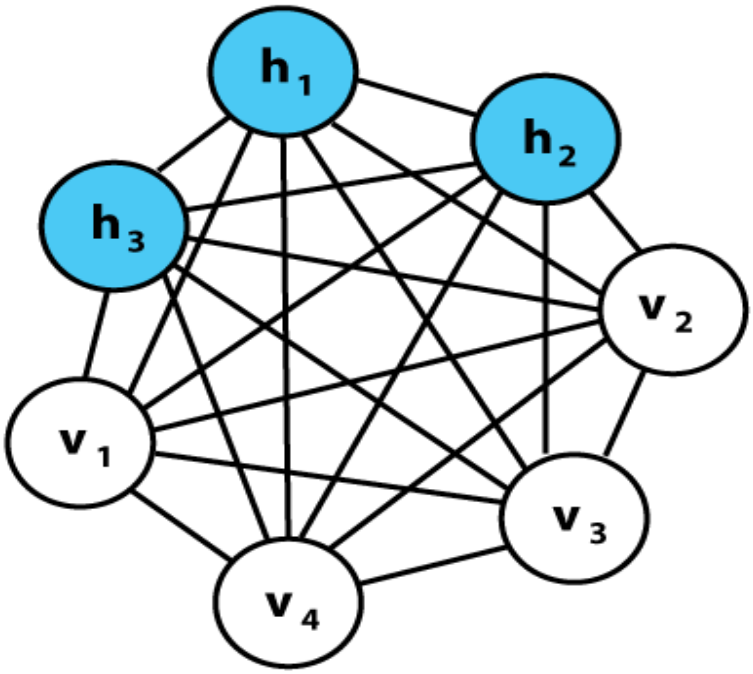

Abbildung 32: Eine Boltzmann-Maschine ist ein ungerichteter, vollständig verbundener Graph, dessen Knoten üblicherweise auf binäre Werte beschränkt sind. "Sichtbare" Knoten (blau) entsprechen der Eingabe (z.B. ein Knoten pro Pixel), "unsichtbare" (weiß) latenten Variablen, die abstrakte Merkmale darstellen sollen. Alle Kanten haben Gewichte (Wikipedia).

Für eine Boltzmann-Maschine bedeutet Inferenz, die latenten Variablen aus der Konfiguration des Graphen abzuleiten. Umgekehrt werden beim Trainieren die Gewichte der Kanten so angepasst, dass das Netz die beobachtete Eingabe mit größter Wahrscheinlichkeit generiert. Die einzelnen Knoten entsprechen Bernoulli-verteilten stochastischen Variablen. Die Wahrscheinlichkeit, dass Knoten s_i aktiv ist, ist gegeben durch:

$$p(s_i=1)=sig(\sum_j s_j w_{ji})$$

Hier sind die w_{ij} die Gewichte der Kanten. Als Aktivierungsfunktion dient die differenzierbare logistische Funktion. Ein Lernalgorithmus für Boltzmann-Maschinen ist erst dann praktikabel, wenn man nur bipartite Graphen zulässt, d.h. Verbindungen innerhalb der Mengen der sichtbaren und versteckten Knoten verbietet. Man spricht dann von einer Restricted Boltzmann Machine (RBM) [Lee2010].

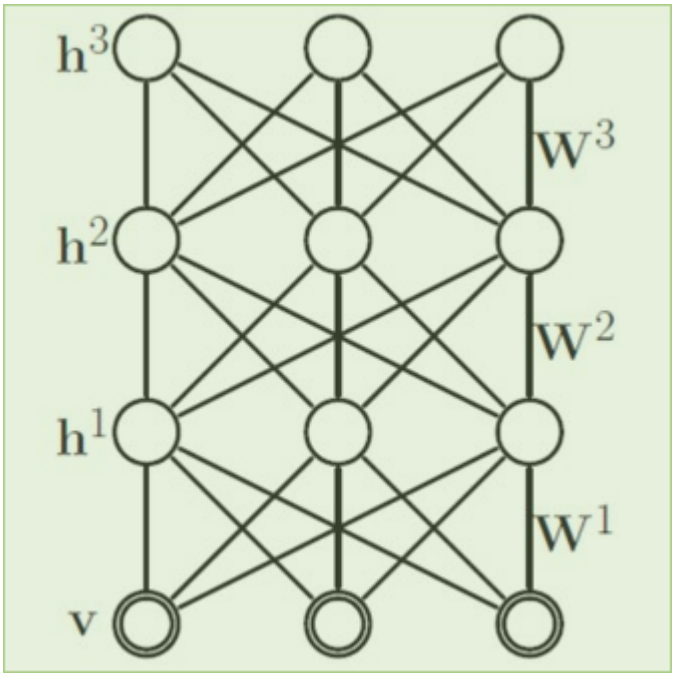

Abbildung 33: Mehrstufige RBM mit drei Eingabe- und neun versteckten Knoten (Hinton, 2006).

Einer Konfiguration (**v,h**) einer RBM ist dann eine Energie zugeordnet[16] :

$$E(\boldsymbol{v},\boldsymbol{h})=-\sum_{i,j} v_i h_j w_{ij}$$

[16] Der Energiebegriff stammt aus der Thermodynamik, bzw. der statistischen Mechanik. Ludwig Boltzmann war ein Pionier dieses Gebietes. Die folgenden Gleichungen sind vereinfacht, da sie auf die Einbeziehung der thermodynamischen Partitionsfunktion verzichten.

Eine multivariate Wahrscheinlichkeitsverteilung über dem Konfigurationsraum ist gegeben durch

$$p(\boldsymbol{v},\boldsymbol{h}) \propto \exp(-E(\boldsymbol{v},\boldsymbol{h})) \quad .$$

Nun ergibt sich die Marginalverteilung der Eingabeknoten v_i als Summe über alle möglichen Konfigurationen der versteckten Knoten h_i zu

$$p(\boldsymbol{v}) \propto \sum_{h_i} \exp(-E(\boldsymbol{v},\boldsymbol{h})) \quad .$$

Diese betrachten wir als Likelihood-Funktion, die zu maximieren ist. Wenn man die Trainingsbeispiele v_i zu einer Matrix V zusammensetzt, die Unabhängigkeit der beiden Knotenmengen der RBM ausnutzt und logarithmiert, ergibt sich insgesamt

$$\underset{W}{argmax}\, E[\sum_{\boldsymbol{v}\in V} \log p(\boldsymbol{v})] \quad .$$

Für dieses Optimierungsproblem existieren hinreichend schnelle Algorithmen. Ein Deep-Learning-Netz setzt sich aus mehreren übereinander gestapelten RBMs zusammen. Dabei werden die versteckten Knoten einer Schicht zu Eingabeknoten der darüber liegenden. Eventuell schaltet man nach der Ausgabeschicht noch einen Klassifikator, z.B. eine Support Vector Machine.

Deep Learning gehört heute zu den erfolgreichsten Lernverfahren auf vielen Gebieten (darunter die Klassifikation großer Bilddatenbanken [Krizhevsky2012] und Phonemerkennung) und erweist sich stellenweise sogar manuell optimierten Systemen überlegen.

6 Die Rolle der Zeit: antizipierende hierarchische Speicher

Welche essenzielle Zutat fehlt in unserer Annäherung an einen künstlichen Kortex? Wir leben in einer sich unablässig und auf allen Zeitskalen verändernden Welt und tragen durch unser Handeln zu ihrem Wandel bei. Wenn wir von Eingabedaten sprechen, meinen wir eigentlich Datenströme. Die besprochenen Sparse Distributed Repesentations sind jedoch statisch. Auch Deep-Learning-Netze sind letztlich Einbahnstraßen, die nichts weiter tun, als zu jeder Eingabe eine Funktion zu berechnen und den Funktionswert auszugeben.

In Kapitel 4.1 haben wir gesehen, dass unser Gehirn zu kognitiven Leistungen in kürzester Zeit fähig ist, z.B. dem Erkennen eines Gesichts in Sekundenbruchteilen. Die häufig angeführte Erklärung, dies werde durch massive Parallelverarbeitung erreicht, greift aber zu kurz. Wenn die Aktivierung eines Neurons (einschließlich der Refraktärphase) etwa 10 ms benötigt, und die Gesichtserkennung in etwa 200 ms erfolgt, bleiben unserem Gehirn für diese anspruchsvolle Aufgabe nur 20 Rechenschritte. Kein Algorithmus (gleich welchen Grades an Parallelisierung) vermag das zu leisten[17].

Es gibt nur einen logischen Ausweg aus dieser Zwickmühle: Zumindest Teile des eingehenden Datenstroms müssen in der Vergangenheit schon vorverarbeitet worden sein. Unser Gehirn hat aus vergangenen spatio-temporalen Mustern gelernt, wenn möglich über die Details hinweg generalisiert und schließlich geeignete interne Repräsentationen angelegt. Von nun an befindet es sich in einem Zustand permanenter Erwartung. Ohne dass dies in unser Bewusstsein vordringt, projiziert es einen verzweigten Baum möglicher Szenarien in die Zukunft, spielt diese durch (besonders die erfahrungsgemäß wahrscheinlichsten) und bereitet sich auf sie vor. Wenn dann ein konkre-

[17] Parallelisierung ist ein mächtiges Werkzeug, doch ist nicht alles parallelisierbar. Geht es z.B. darum, einen Stein hundert Schritte weit zu tragen, vollbringen tausend Menschen das nicht schneller als einer.

ter Fall eintritt, hat es die entscheidende Rechenarbeit schon geleistet und kann augenblicklich reagieren. Es ist wie bei jenem einfachen, aber für den nicht Eingeweihten verblüffenden Kartentrick: Man zeigt uns ein Kartendeck und fordert uns auf, uns eine Karte zu merken, die der Zauberkünstler nicht kennt. Dann sollen wir ihm die Karte nennen - sagen wir Pik-Ass -, worauf er selbige triumphierend aus seiner rechten Hosentasche zieht. Der „Trick" besteht darin, dass er bereits vorher alle nötigen Karten an verschiedenen Orten versteckt hat. Er war umfassend auf alle Szenarien vorbereitet und brauchte dann nur noch ein Minimum an Information, um das richtige Szenario auszuwählen.

Haben wir mit dieser Technik nun die Quadratur des Kreises erreicht, d.h. ein Problem in weniger Rechenschritten gelöst, als es theoretisch möglich ist? Natürlich nicht, denn es gibt nur eine begrenzte Zahl an wahrscheinlichen Szenarien, für die Vorarbeit zu erbringen ist. Auch der Kartentrick funktioniert nur, da es sehr wenige Auswahlmöglichkeiten gibt. Unser Gehirn vermag zu antizipieren, da es bewährte Annahmen über die räumliche und zeitliche Struktur der Welt macht. Geschieht etwas völlig unerwartetes, beginnt der kognitive Prozess wieder bei Null. Vor dem No-Free-Lunch-Theorem gibt es leider kein Entkommen. Wir können nur eine sehr kleine Klasse von Aufgaben besonders effizient lösen, zum Glück eine relevante.

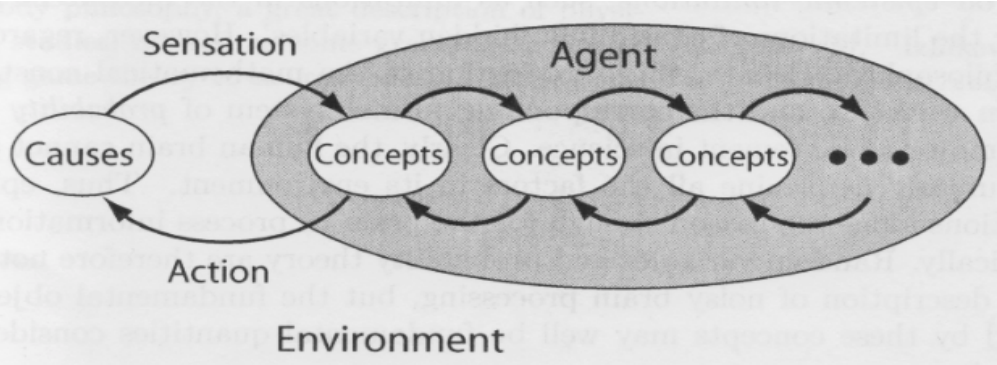

Abbildung 34: Von Beobachtungen ausgehend, versucht ein Agent ursächliche Zusammenhänge in seiner Umgebung zu erkennen. Er wechselt zwischen verschiedenen internen Zuständen, aktualisiert diese und wählt geeignete Handlungen aus (Trappenberg, 2010).

Ein antizipierender hierarchischer Speicher, den wir im Folgenden nach Hawkins als HTM (Hierarchical Temporal Memory) bezeichnen[18], hat Ähnlichkeit zum Konzept des Hidden Markov Model (HMM), das seinerseits ein Spezialfall eines Bayesschen Netzes ist. Man kann ihn am ehesten mit einem hierarchischen HMM[19] variabler Ordnung vergleichen, denn die alleinige Kenntnis des Vorgängerzustandes (HMM erster Ordnung) genügt nicht, um nützliche probabilistische Vorhersagen über die Zukunft zu machen. Auch liegt bei einem HMM die Betonung auf der zeitlichen Dimension (z.B. in der Abfolge von Lauten bei der Spracherkennung). Räumliche und zeitliche Muster können nicht simultan verarbeitet werden. Dies ist aber eine der herausragenden Fähigkeiten des Kortex.

[18] Ein von Hawkins weitgehend synonym verwendeter Begriff ist der des Memory Prediction Framework.

[19] Ein hierarchisches HMM lässt sich immer in ein äquivalentes flaches HMM transformieren, kommt aber der Struktur seiner Aufgabe unter Umständen näher.

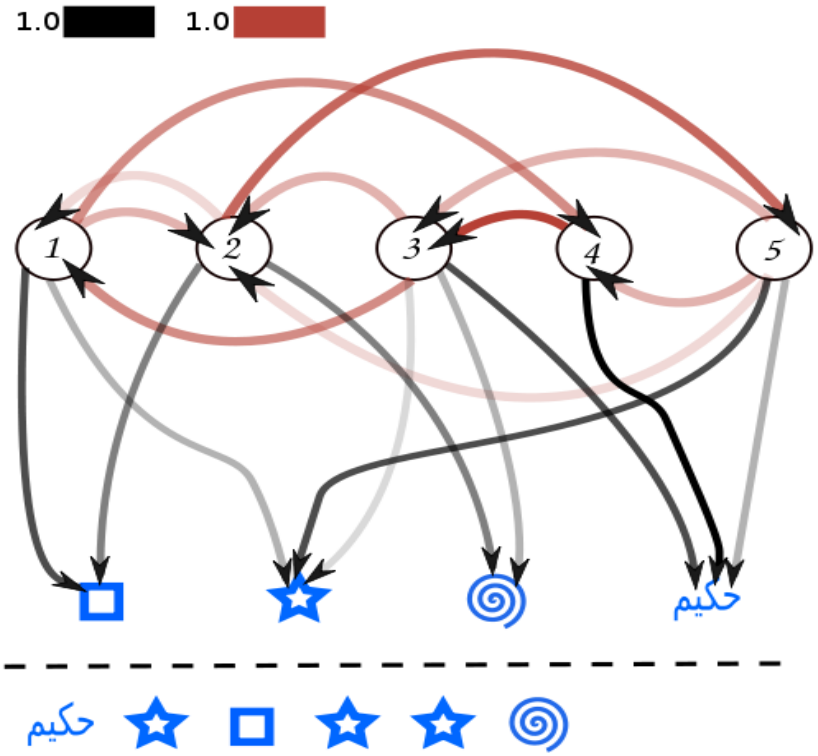

Abbildung 35: Hidden Markov Graph, bei dem die Wahrscheinlichkeit für Zustandsänderungen (rot) und Ausgabesymbole (grau) durch die Farbintensität der Kanten angegeben ist. Wenn die unten dargestellte Symbolsequenz beobachtet wurde, interessieren wir für uns die wahrscheinlichste Abfolge der inneren Zustände, die wir als Ursache der Beobachtung betrachten. Diese Fragestellung kann effizient durch den Viterbi-Algorithmus beantwortet werden (Wikipedia).

Der Faktor Zeit spielt auch bei der Objekterkennung eine große Rolle. Wenn wir z.B. einen Ball durch unser Blickfeld rollen sehen, wird sich der auf die Retina projizierte Ball von Einzelbild zu Einzelbild weniger stark verändern als der Hintergrund. Wir können aus der zeitlichen Nähe also schließen, dass wir immer dasselbe Objekt sehen. Prinzipiell ist Objekterkennung im dynamischen Fall einfacher als im statischen.

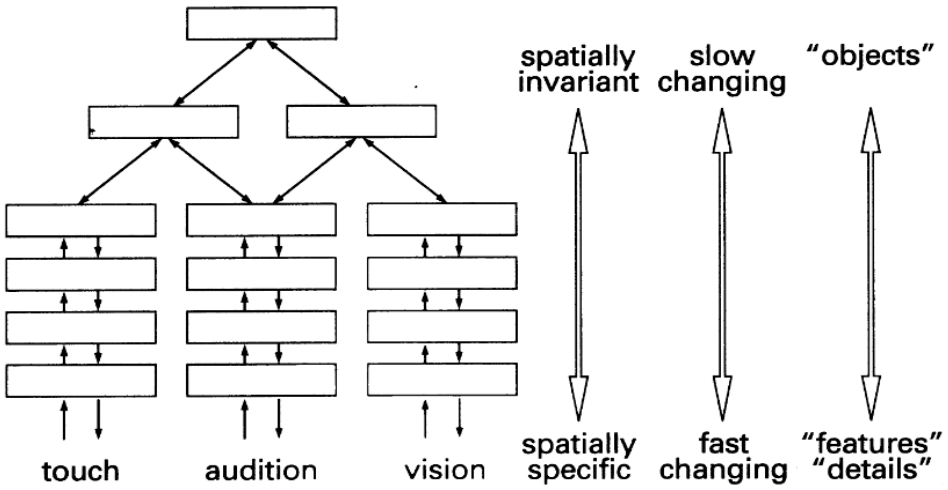

Abbildung 36: Grundsätzlicher Aufbau eines HTM, der mehrere Sensoren verwendet, um spatio-temporal invariante Repräsentationen zu erzeugen. Die rechteckigen Untereinheiten werden als Regionen bezeichnet (Hawkins, 2004).

Ein HTM ist nichts Geringeres als der Versuch, einen universellen Lernalgorithmus zu entwickeln, inspiriert von der Tatsache, dass sich die für verschiedene Sinnesorgane im Kortex zuständigen Bereiche nur marginal unterscheiden. In einem berühmten Tierexperiment wurde das von den Augen kommende visuelle Signal zum auditorischen Kortex umgeleitet, der dann zu Sehen lernte [Roe92].

Ein HTM besteht aus mehreren Regionen, die hierarchisch angeordnet sind, meist pyramidenförmig. Der Informationsfluss erfolgt von unten nach oben, wobei die Ausgabe der obersten Region direkt zur Klassifikation von Objekten verwendet werden kann, und gleichzeitig auch in der Gegenrichtung. Eine Region kann nach unten hin ausgeben, was sie als nächste Eingabe erwartet. Diese Daten können z.B. zur motorischen Steuerung verwendet werden. Ein angenehmer Nebeneffekt ist, dass ein HTM die Eingabedaten komprimiert, indem er wiederholt vorkommende Muster nur einmal speichert und sie dann referenzieren kann.

Die in den Regionen gespeicherten Repräsentationen werden nach oben hin zunehmend abstrakter, d.h. sie

- decken einen größeren Teil des Wahrnehmungsfeldes ab,
- werden zeitlich stabiler (ändern sich seltener),
- werden zunehmend räumlich invariant (gegenüber Drehung, Translation und Skalierung).

Jede Region erzeugt Sequenzen aus Sparse Distributed Representaions der Eingabedaten, die sie von einer Vorgängerregion bzw. direkt von den Sensoren erhält, und gibt diese nach oben weiter. Die maximale Länge der Sequenzen ist fest vorgegeben und bestimmt, an wie viele zurückliegende Schritte sich die Region erinnern kann.

Wenn eine Region einer Eingabe begegnet, die eine unvollständige Version einer ihr bereits bekannten SDR ist (z.B. eine teilweise verdecktes Objekt), fungiert sie als autoassoziativer Speicher und vervollständigt diese. Die in der SDR angelegte Redundanz macht dies möglich [Hawkins2009].

Was geschieht, wenn eine Region eine Eingabe erhält, die keinerlei Ähnlichkeit mit den gespeicherten Mustern hat? Sie passt sich dann schrittweise an, d.h. legt eine neue SDR an, wenn diese Eingabe mehrfach vorkommt. Der Mechanismus hierfür ist eine graduelle Anpassung der Gewichte zwischen den Netzwerknoten, analog zum Hebbschen Lernen (vgl. Kapitel 7).

Bei der Beschreibung des HTM war bisher weder von Lern- noch von Inferenzphasen die Rede, sondern nur von spatio-temporaler Abstraktion und Sequenzvorhersage. Dies ist kein Zufall, sondern eine seiner bemerkenswertesten Eigenschaften. Trainieren (Lernen) und Inferenz (Erinnern und Prognostizieren) erfolgen simultan, was man als Online Learning bezeichnet[George2008].

7 Der kortikale Lernalgorithmus

Ziel der von Hawkins im Jahr 2005 gegründete Firma Numenta[20] ist die Umsetzung der bisher geschilderten Konzepte in eine marktreife Software. Deren erste Version namens NuPIC (Numenta Platform for Intelligent Computing) wurde 2007 veröffentlicht, eine zweite namens Grok im Jahr 2013. Ursprünglich war geplant, für diese Arbeit den Algorithmus selbst zu implementieren, was sich aber im Rahmen der zur Verfügung stehenden Zeit als nicht durchführbar erwies. Schließlich waren bei Numenta mehrere professionelle Softwareentwickler damit über Jahre beschäftigt. Im Folgenden wird der aktuelle Grok-Algorithmus skizziert [Numenta2011]. Außerdem werden einige Experimente diskutiert, die mit der frei verfügbaren Open-Source-Version openHTM, in C++ von einer Gruppe von Entwicklern[21] implementiert, durchgeführt wurden. All dies ist als Proof of Concept zu verstehen.

Die wesentliche Neuerung von Grok gegenüber NuPIC besteht darin, dass dieselben Elemente (bezeichnet als Zellen), die den Eingabedatenstrom in Sparse Distributed Representations transformieren, auch Speicherung und Vorhersage zeitlicher Sequenzen übernehmen. Ganz wie dies auch bei ihrem biologischen Vorbild der Fall ist. Zwar verfügen die künstlichen Neuronen über Dendriten und Axone, doch war eine quasi-realistische Simulation nie vorgesehen, sondern eher eine Interpretation der Natur, die technisch leicht realisierbar ist.

7.1 Beschreibung des Algorithmus

Ein vollständiger HTM besteht aus mehreren, meist pyramidenartig angeordneten Regionen. Wir beschränken uns auf eine einzelne Region, die bereits die interessantesten Eigenschaften sichtbar macht. Ihr Wahrnehmungsfeld (WF) ist quadratisch, kann intern aber als eindimensionaler Vektor gespeichert werden. Die Eingabe ist binär. An-

[20] Von lateinisch *mentis,* „den Geist betreffend“. Später in Grok umbenannt (ein Wort aus R. Heinleins Roman *„Stranger in a Strange Land“*).

[21] D. Ragazzi, B. Matt, D. King, U. Kirschenmann, M. Ferrier et al. Verfügbar unter http://sourceforge.net/projects/openhtm/ bzw. https://github.com/MichaelFerrier/HTMCLA.

dere Eingabeformate müssen vorher kodiert werden. Die Zellen einer Region sind in Säulen angeordnet. Bei n Zellen pro Säule können temporale Sequenzen der Länge n erkannt werden. Eine Zelle kennt drei Zustand: inaktiv, aktiv und prädiktiv[22].

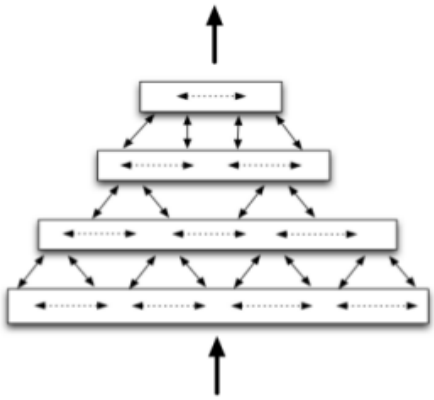

Abbildung 37: Aus vier Regionen bestehende HTM. Information wird vertikal zwischen den Ebenen und horizontal innerhalb der Ebenen ausgetauscht (Numenta, 2011).

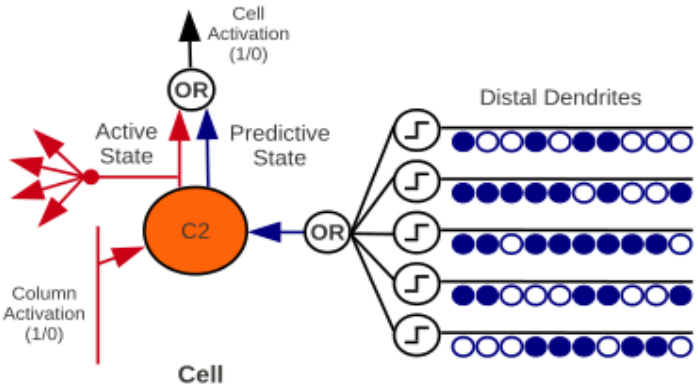

Abbildung 38: HTM-Zelle (Price, 2011)

[22] Sie ist dann im Vorhersagemodus und erwartet, im nächsten Schritt aktiviert zu werden.

HTM-Zellen sind über ihren proximalen Dendriten lateral mit anderen Zellen der Region verbunden. Die Kreise rechts (in Abb. 38) stellen die Menge der potentiellen Synapsen dar, die nur innerhalb eines festgelegten Radius um die Zelle liegen können. Die Verbindung kommt dann zustande, wenn die Synapse aktiv ist (blau eingefärbt). Dazu verfügt jede potentielle Synapse über ein Datenfeld, das ihre aktuelle Verbindungsstärke speichert. Nur wenn diese über einem globalen Permanenz-Parameter liegt, leitet die Synapse Impulse weiter. Der proximale Dendrit (unten links) empfängt binäre Feed-Forward-Daten von unten[23], die für alle Zellen einer Säule gelten. Das Axon einer Zelle leitet Signale auf zweierlei Arten weiter:

1. Wenn die Zelle im aktiven Zustand ist, erregt sie ihre horizontalen Nachbarn.
2. Ist sie im prädiktiven *oder* aktiven Zustand, leitet sie dies an die weiter oben liegende Region (oder an die direkte Ausgabe) weiter.

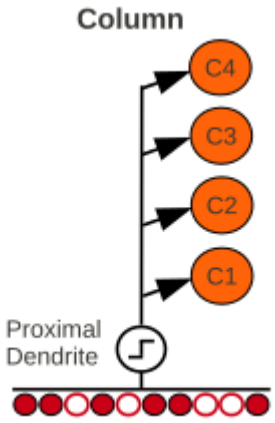

Abbildung 39: HTM-Säule mit vier Zellen, die als Ganzes nur von unten her aktiviert werden kann, wenn in ihrem Wahrnehmungsfeld ein fester Anteil an Bits gesetzt ist (Price, 2011).

[23] Vom WF oder der Vorgängerregion.

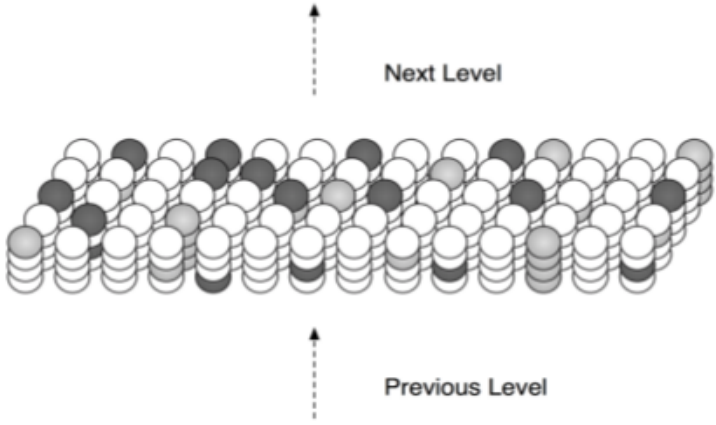

Abbildung 40: Region mit 4-stufigen Säulen. Aktive Zellen (grau) wurden von unten her angeregt. Zellen im prädiktiven Zustand (schwarz) wurden lateral von benachbarten aktiven Zellen erregt (Numenta, 2011).

Alle Zellen der Region sind im Unterschied zu ihren biologischen Vorbildern synchron getaktet. Lernen und Inferenz laufen parallel und in zwei Phasen ab, dem räumlichen und dem zeitlichen Pooling [Price2011] [Numenta2011].

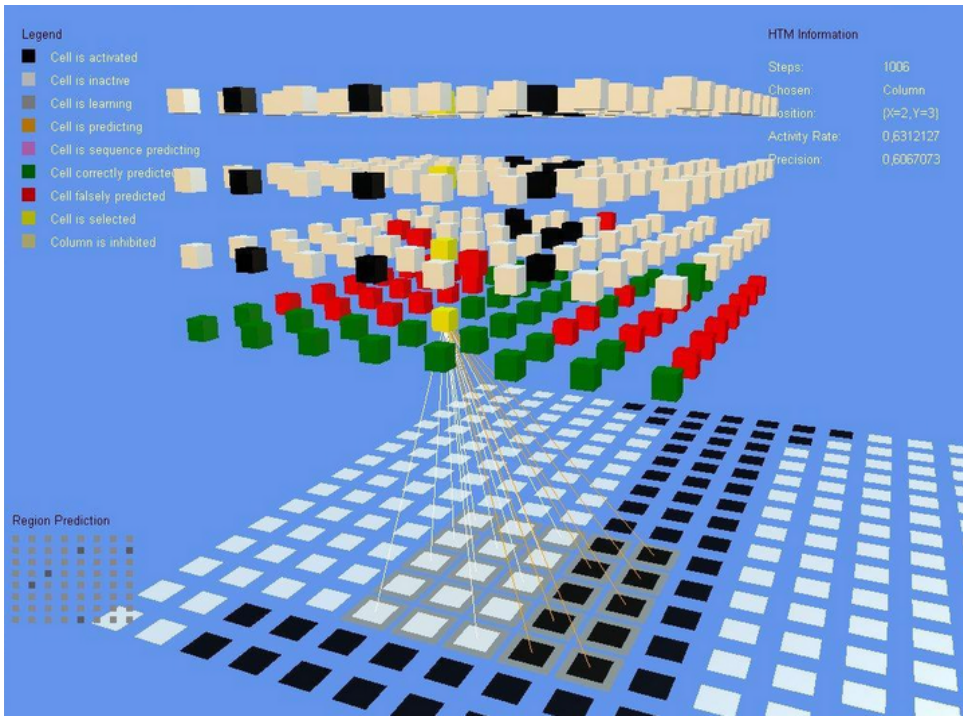

Abbildung 41: Bildschirmfoto von openHTM. Man kann erkennen, welche Pixel der binären Eingabe zum Wahrnehmungsfeld der gelb eingefärbten Säule gehören (openHTM, 2011).

7.1.1 Räumliches Pooling

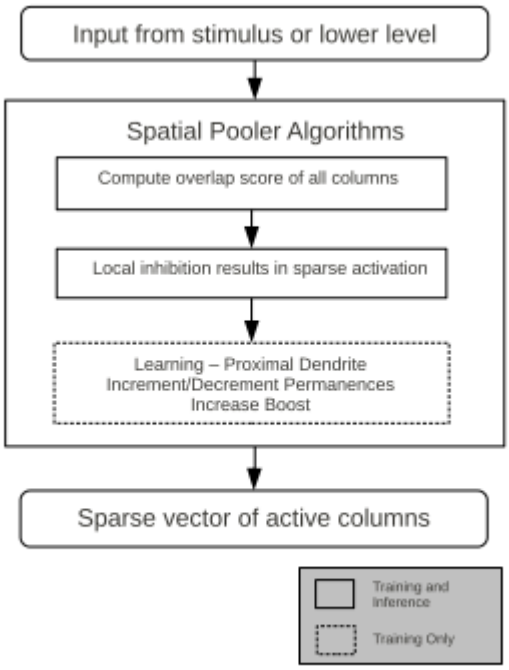

Abbildung 42: Ablaufdiagramm des räumlichen Poolers. Wenn die Region nur zur Inferenz verwendet wird, kann die Lernphase (gestrichelte Umrandung) entfallen (Price, 2011).

Ziel des Spatial Pooling ist es, den Eingabestrom in eine SDR zu transformieren, die sogar mehrere Merkmale der Eingaben gleichzeitig beinhalten kann. Der Algorithmus läuft folgendermaßen ab:

1. Zu Beginn werden zufällige Pixel den Wahrnehmungsfeldern der einzelnen Säulen zugeordnet.
2. Jede Säule zählt die gesetzten Pixel in ihrem WF, die über leitende Synapsen (Verbindungsgewicht > Permanenz) verbunden sind. Dieser Vorgang ähnelt Faltung und Subsampling in der Bildverarbeitung.

3. Das Ergebnis wird mit einem Boost-Wert multipliziert. Dieser Wert ist umso größer, je seltener die Säulen in der Vergangenheit angeregt wurden. So soll eine allzu einseitige Aktivität unter den Säulen vermieden werden.

4. Säulen mit hoher Aktivierung hemmen ihre Nachbarn innerhalb eines festen Radius (k-winner-take-all). Diese einfache, vom Kortex inspirierte Heuristik erzeugt näherungsweise eine SDR. Eine Näherung, die sich in der Praxis bewährt hat, und die viel Rechenzeit (für exakte Lösungsalgorithmen) spart.

5. Hebbsches Lernen: Im Wahrnehmungsfeld werden Synapsen, die häufig/selten zur Aktivierung der Säule beitragen, verstärkt/geschwächt. Ihr Gewicht wird mit dem Permanenz-Parameter verglichen, und sie werden leitend bzw. nicht leitend geschaltet.

6. Über die durchschnittliche Aktivierung der Säulen wird Buch geführt und dementsprechend ihre Boost-Wert angepasst.

7.1.2 Zeitliches Pooling

Bisher wurden die Säulen der Region als Einheit betrachtet. Nun spielt jede ihrer Zellen eine gesonderte Rolle. Wie im SP besitzen sie eine Menge potentieller Synapsen (distaler Dendrit), allerdings nicht zum Wahrnehmungsfeld hin, sondern zu ihren Nachbarn. Entsprechend dem Verhältnis ihres Gewichts zum Permanenz-Parameter, sind sie leitend bzw. nicht leitend. Die Idee des Temporal Pooling besteht darin, jede SDR im Kontext verschiedener vergangener Eingaben darzustellen. Dies ist möglich, da jede Säule eine SDR durch verschiedene Untermengen ihrer Zellen kodieren kann.

- Jede neue vom SP kommende Eingabe ist eine SDR der zur Zeit aktiven Säulen.

- Wenn eine Säule nach dem SP aktiv ist und über prädiktive (im vorhergehenden Schritt aktive) Zellen verfügt, dann werden *diese* Zellen auf aktiv gesetzt, da ihre Vorhersage eingetroffen ist. Inaktive Zellen behalten ihren Status. Ist eine Säule aktiv, aber keine ihrer Zellen prädiktiv, dann werden alle Zellen auf aktiv gesetzt. Die Menge der nun aktiven Zellen der Region stellt die gegenwärtige Eingabe im Kontext der vorherigen dar.

- Eine inaktive Säule, die über prädiktive Zellen verfügt, wird insgesamt auf inaktiv gesetzt, da ihre Vorhersage sich als falsch erwiesen hat.

- Jede Zelle in jeder Säule berechnet den aktuellen Wert ihres proximalen Dendriten, wobei nur die aktiven Zellen in ihrer Umgebung einfließen.

- Hebbsches Lernen findet auch hier statt: Diejenigen Synapsen im distalen Dendriten, die an einer korrekten Vorhersage teilgenommen haben, werden verstärkt.

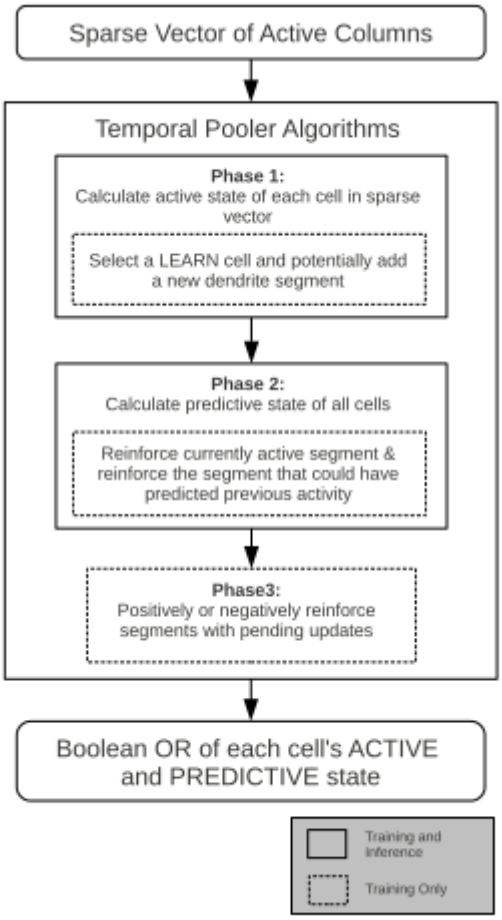

Abbildung 43: Der Temporal-Pooling-Algorithmus im Überblick (Price, 2011).

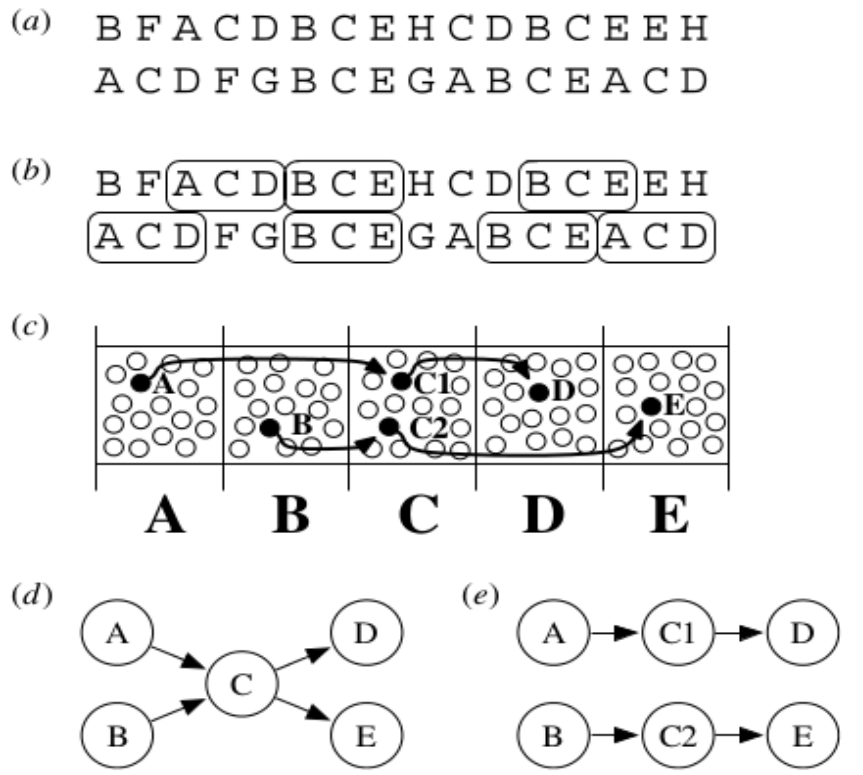

Abbildung 44: Vereinfachte Veranschaulichung des Temporal Pooling. a) Eingabesequenz [jeweils ein Buchstabe A-H] einer Region mit fünf Säulen [A-E]. b) Sich wiederholende Untersequenzen. "C" kommt in zwei Untersequenzen vor. c) Verbindungen im distalen Dendriten der Zellen, die die zeitliche Abfolge darstellen. d) HMM erster Ordnung, das keine Vorhersage im Zustand C machen kann. e) Zwei Markov Ketten mit „kontextsensitiven" C1/C2 (Hawkins, 2009).

Die Einzelheiten der Implementierung des Temporal Poolers sind komplex. Hier sei auf [Numenta2011] und [Price2011] verwiesen. Je nach Größe der Säulen können auch Vorhersagen höherer Ordnung gemacht werden.

7.2 Vorverarbeitung der Eingabe

Da es ein elementares Designprinzip ist, nur binäre Eingabeströme zu verwenden, ist im Allgemeinen eine Kodierung anderer Datenformate nötig. Sei die Eingabe ein Bitstring, der aus k Substrings der Länge n besteht. Je nach Anforderung können diese Substrings beispielsweise für Symbole, Zahlenwerte, farbige Pixel oder Tonhöhen stehen. Deren genaue Kodierung ist ein eigenständiges Problem, bei dem viele Parameter optimiert werden müssen. Numenta führt dazu zahlreiche parallel laufende Experimente auf Hochleistungsrechnern durch. Bisher haben sich folgende Richtlinien herauskristallisiert:

- Für Kategorien setzt man eine feste Anzahl m der n Bits des Substrings. Dabei sollten bei verschiedenen Kategorien keine Überschneidungen vorkommen.

- Bei Skalaren diskretisiert man nach folgendem Muster, wenn m=3 Bits gesetzt werden sollen:

Wert	Darstellung
1	*111000 … 0000*
2	*011100 … 0000*
n-m	*000000 … 1110*
n-m+1	*000000 … 0111*

7.3 Ergebnisse der Arbeit mit openHTM

Die Software wurde mit *Visual Studio 2012* und der Bibliothek *Qt 5.0.2* auf einem Rechner unter *Windows 7* (64-Bit) erfolgreich kompiliert. Das GUI der Anwendung ist ergonomisch gestaltet, und Ergebnisse werden ansprechend visualisiert.

Die Eingabedaten, die Topologie der HTM-Region sowie diverse Parameter werden in XML-Dateien gespeichert und dann von openHTM eingelesen. Nur wenige Experimente konnten durchgeführt werden, da alle Eingabebilder von Hand angelegt werden mussten. Fertige binäre Datensammlungen solch niedriger Auflösung waren mir nicht bekannt. Neben dem unten geschilderten Erlernen und Vorhersagen von Buchstabensequenzen, wurde auch mit einem springenden Ball in einer Bildsequenz experimentiert, dessen Verhalten das System einigermaßen prognostizieren konnte.

Die größte Schwierigkeit scheint darin zu bestehen, eine Vielzahl an Parametern[24] so einzustellen, dass man ein funktionierendes Modell erhält. Im gegebenen Setting war dies schlicht unmöglich. Bei Numenta lässt man dazu bis zu hundert Modelle an einer Swarm Particle Optimization (SPO) teilnehmen. Die Modelle durchwandern als virtuelle Teilchen, mit Zufallswerten für Position und Geschwindigkeit initialisiert den Parameterraum. Dabei sucht nicht nur jeder einzelne Partikel lokal nach Minima (wie z.B. beim Simulated Annealing), sondern er orientiert sich in seiner Bewegung auch am Wissen der Gesamtpopulation über die Struktur des Parameterraums. Ursprünglich entstammt dieser Algorithmus der Beobachtung von Vogel- und Fischschwärmen und findet oft gute Näherungslösungen selbst in ungünstigen Suchräumen, die zum Steckenbleiben in lokalen Minima verleiten [Eberhart95].

[24] Permanenz, Lernrate, Lernradius, Hemmungsradius, Boost-Wert, etc.

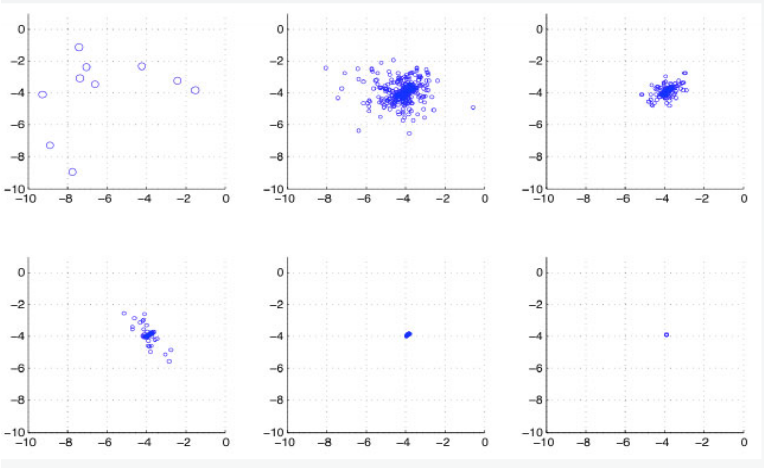

Abbildung 45: Ein Partikelschwarm auf der Suche nach einer (möglichst) optimalen Parameterkonfiguration für ein HTM-Modell. Aus Gründen der Anschaulichkeit sind hier nur zwei Dimensionen gezeigt (Numenta).

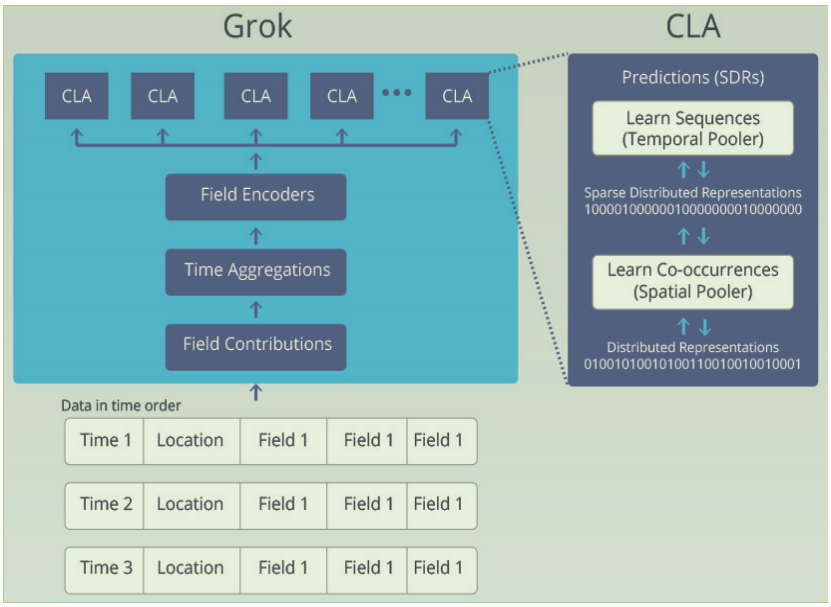

Abbildung 46: Gesamtübersicht des Grok-Systems: Der Eingabedatenstrom durchläuft mehrere Filter und Kodierer zur Vorverarbeitung (vgl. 7.2), gelangt dann in eine große Anzahl parallel arbeitender HTM-Modelle (hier als CLA bezeichnet, für Cortical Learning Algorithm), von denen das Beste per PSO ausgewählt wird (Numenta).

Beim folgenden Testlauf wurde der HTM-Region wiederholt die Bildfolge „AAAX“ präsentiert. Zunächst wird nur der Spatial Pooler trainiert. Erst wenn dieser nach etwa

2000 Iterationen brauchbare SDR erzeugt hat[25], je eine für „A“ und „X“, wird der Temporal Pooler zugeschaltet. Nach etwa 1000 weiteren Durchgängen hat dieser die temporalen Zusammenhänge soweit erkannt, dass er nach dreimal „A“ ein „X“ vorhersagt. Die hohe Zahl von 3000 benötigten Trainingsdurchgängen erklärt sich dadurch, dass ja keine abstrakten Zeichen erlernt wurden, sondern 12 x 12 Pixel große Schwarzweißbilder.

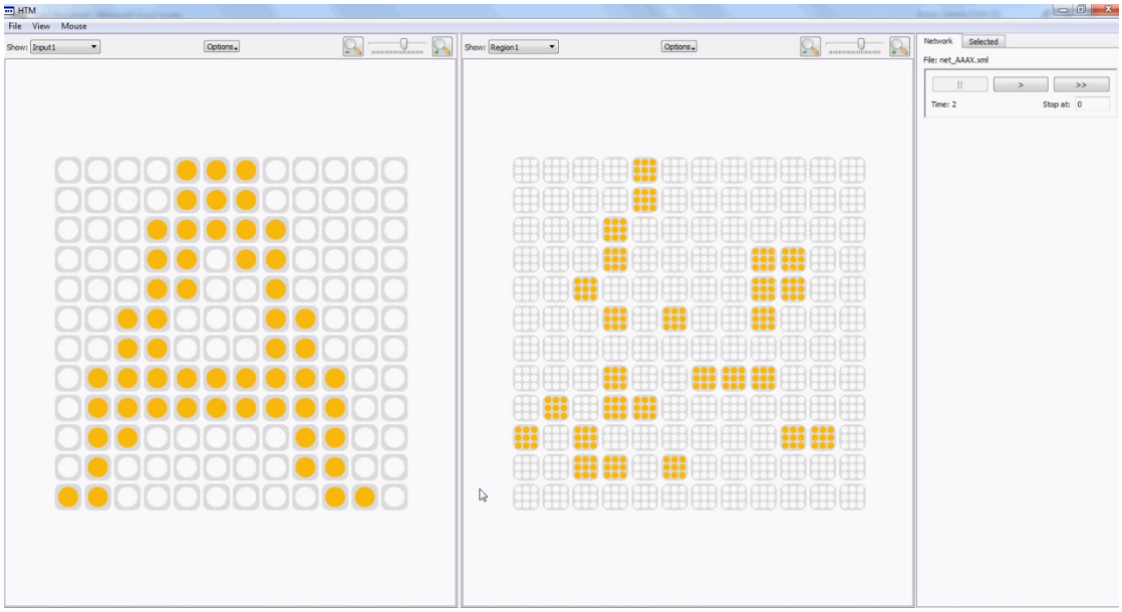

Abbildung 47: Das GUI der openHTM-Software. Links sieht man das Wahrnehmungsfeld mit dem Eingabebild (12 x 12 Pixel), rechts die HTM-Region derselben Größe mit neun Zellen pro Säule. Durch die Eingabe aktivierte Zellen sind orange gefärbt.

[25] In einem zweiten Experiment wurde der Boosting-Mechanismus über eine Änderung in der XML-Konfigurationsdatei deaktiviert. Es zeigte sich, dass viel zu wenige der verfügbaren Säulen am Trainingsprozess teilnahmen, als dass eine SDR hätte entstehen können.

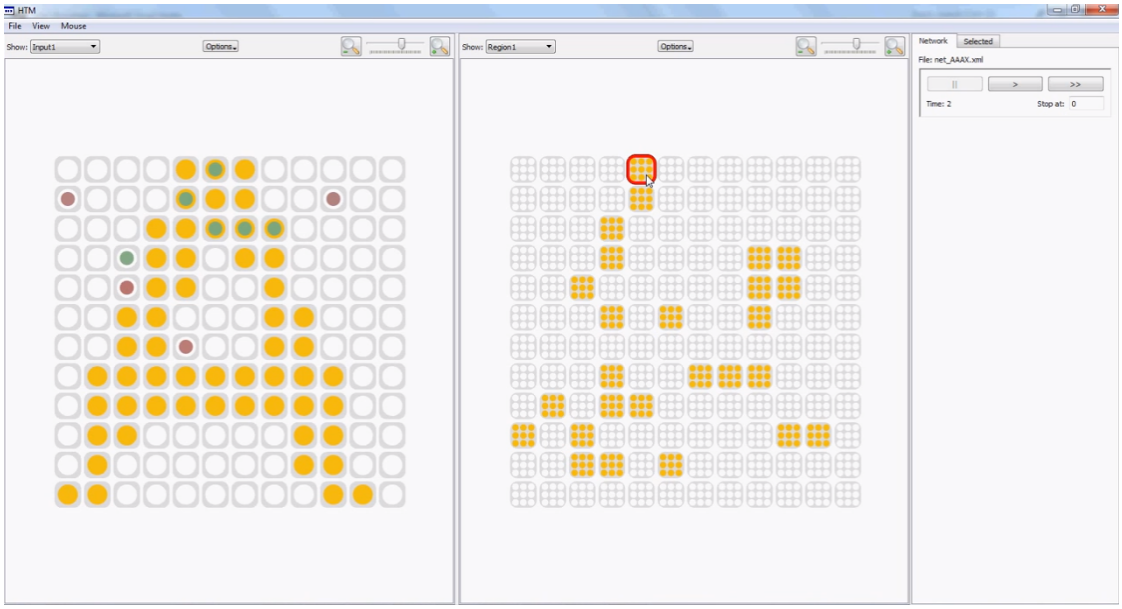

Abbildung 48: Links ist der proximale Dendrit der rot umrandeten Säule markiert. Zur Aktivierung beitragende Synapsen sind grün, nicht beitragende lila gefärbt.

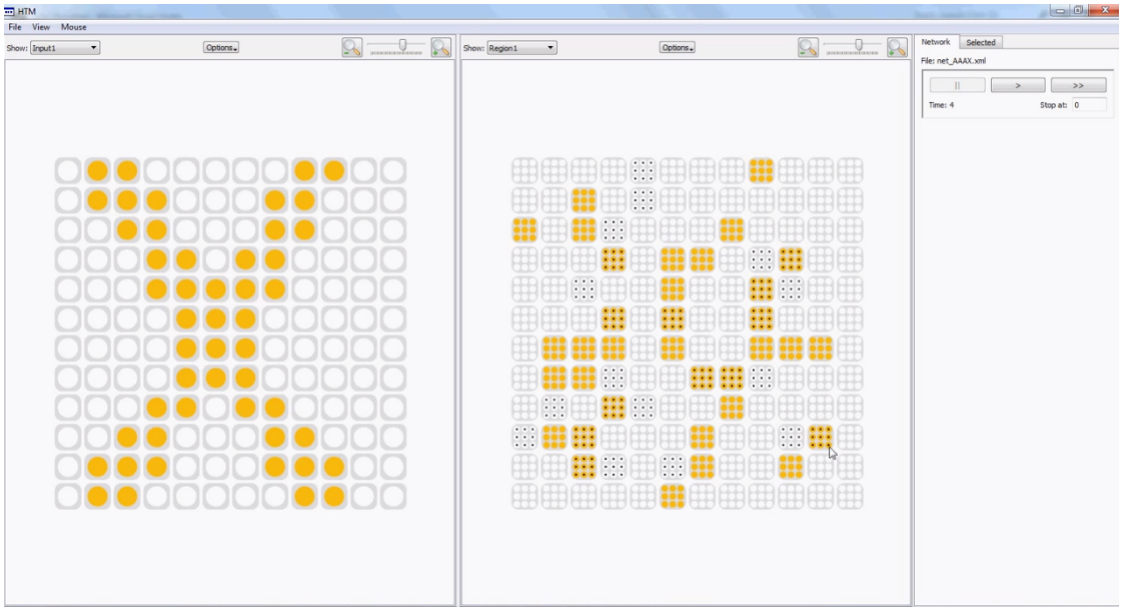

Abbildung 49: Hier sind die bei "A" aktiven Säulen (orange) denen für "X" gegenübergestellt (schwarz). Es gibt Überlappungen, was nicht wünschenswert ist.

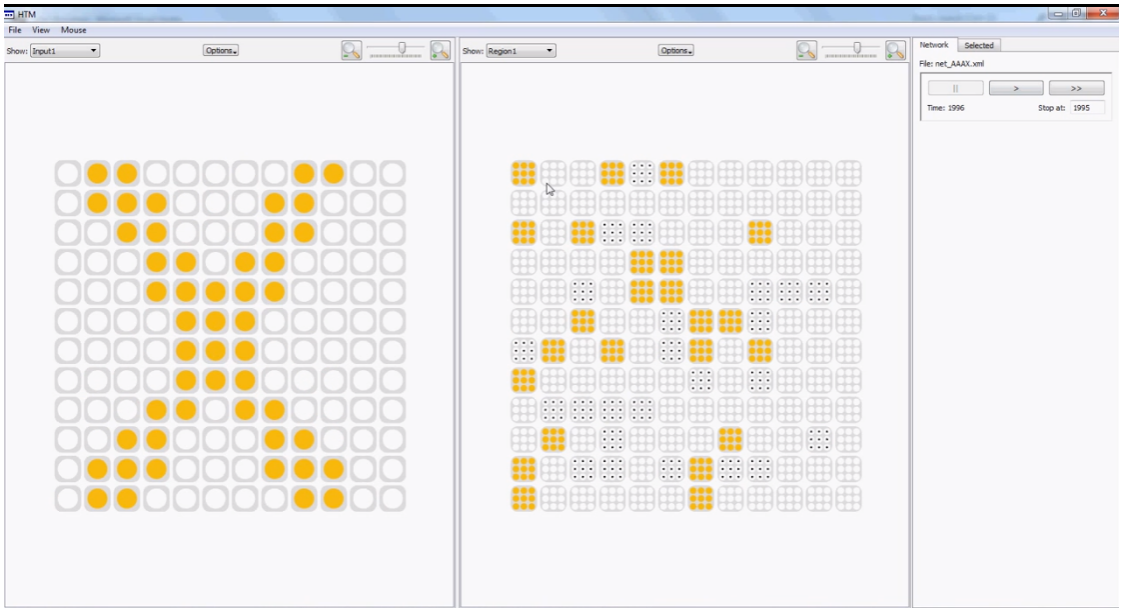

Abbildung 50: Etwa 2000 Schritte später, hat der Boosting-Mechanismus dafür gesorgt, dass beide Säulenmengen disjunkt sind. Eine SDR ist entstanden.

Abbildung 51: Nun setzt der temporale Pooler ein. Orange gefärbte Zellen sind im aktiven Zustand, gelbe im prädiktiven. Links: Das Wahrnehmungsfeld der Säule, die die im rechten Bild rot umrandete Zelle beinhaltet.

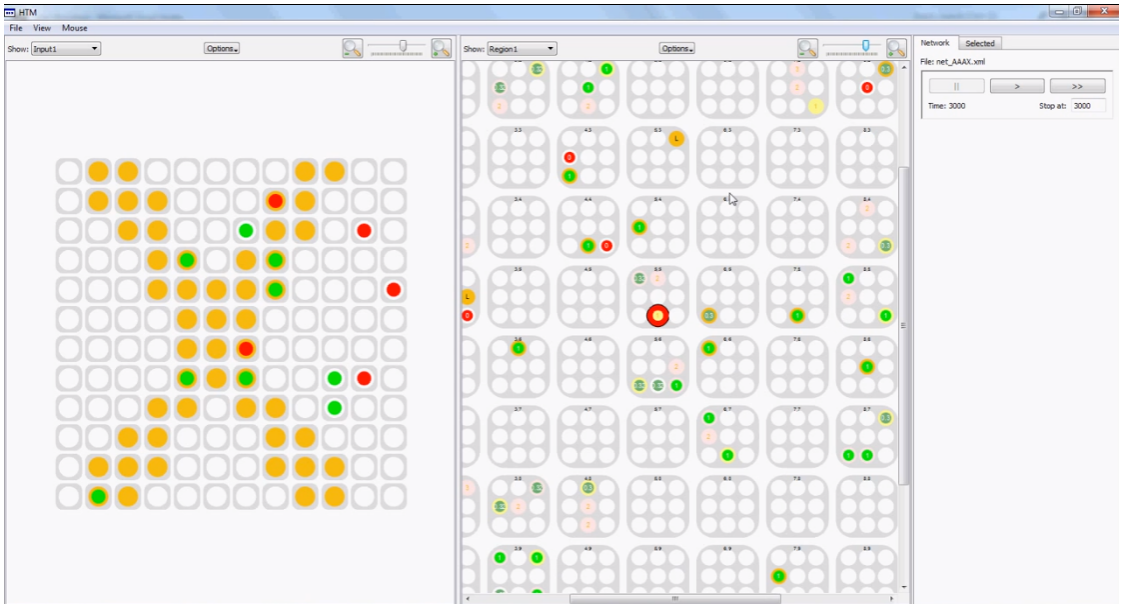

Abbildung 52: Hier sieht man prädiktive Zustände höherer Ordnung.

Abbildung 53: Rechts ist nun die Vorhersage der Region zu sehen, nachdem sie drei aufeinanderfolgende "A" beobachtet hat. Sie hat das zeitliche Muster erlernt.

7.4 Abschließende Betrachtungen

Grundsätzlich scheint ein Proof of Concept für simple Beispiele erbracht. Doch die Frage der Eignung der Algorithmen für komplexere Datenströme (wie Audio oder Video) ist offen. Numenta hat hierzu noch nichts Überprüfbares veröffentlicht. Man wirbt aber damit, dass eine Betaversion von Grok, die in der firmeneigenen Cloud läuft, erfolgreich im industriellen Bereich eingesetzt wird [Technology Review (MIT), Feb. 2013]:

1. Die Firma EnerNoc verwendet Grok, um Verbrauchsschwankungen in Elektrizitätsnetzen in Echtzeit zu prognostizieren.

2. Grok überwacht in einer europäischen Windfarm 800 Windräder über je 34 Temperatursensoren, um Anomalien aufzuspüren und gegebenenfalls Wartungspersonal zu entsenden, noch bevor ein Defekt aufgetreten ist. Besonders in Offshore-Anlagen könnte man dadurch erhebliche Kosten einsparen.

3. Im Finanzbereich sucht Grok nach Abweichungen von typischen Handelsmustern, um in Echtzeit betrügerische Transaktionen zu entdecken.

Hawkins ist ein fächerübergreifend denkender und unkonventionellen Lösungen gegenüber offener Entrepreneur, kein Wissenschaftler in einem akademischen Umfeld. Damit ist er nicht an Peer Review gebunden, was aber kein Nachteil sein muss. Wenn sich seine Software am Markt bewährt, steht das einem guten akademischen Leumund in nichts nach. Man sollte nicht vergessen, dass weder die Glühbirne noch das Automobil an Universitäten entwickelt wurden.

Für den Numenta-Algorithmus spricht Einiges. Seine hierarchische Architektur, die gewissermaßen den Aufbau vieler Objekte der wahrnehmbaren Welt widerspiegelt. Seine Fähigkeit zum unüberwachten Online Learning in Echtzeit (bei entsprechend leistungsfähiger Hardware). Die effiziente Erkennung temporaler Muster in Datenströ-

men, besonders dann, wenn diese nicht statistisch unabhängig sind, was in realistischen Fällen typisch ist[26], beim Maschinellen Lernen aber oft nicht berücksichtigt wird.

Ein Vorschlag, das HTM-Modell zu modifizieren, wäre es, die Rolle der Synapsen zu erweitern. Diese sind in der Natur keineswegs binär. Sie können graduell hemmend oder verstärkend wirken und spielen somit eine wesentlich gewichtigere Rolle als ihr künstlicher Gegenpart.

Selbst ein HTM mit großer Speicherkapazität wird unausweichlich dazu gezwungen, sein gespeichertes Wissen zu überschreiben, wenn er fortlaufend mit neuen Daten und fremden Mustern konfrontiert wird. Nötig wäre in diesem Fall ein übergeordneter Automatismus, der den HTM überwacht, relevante SDR extrahiert und an anderer Stelle speichert. Mit anderen Worten: ein Langzeitgedächtnis; eine Garbage Collection unter umgekehrtem Vorzeichen, die den Kurzzeitspeicher nicht nur von überflüssigem Datenballast befreit, sondern wertvolles Wissen zur späteren Wiederverwendung auslagert. Dazu gehörte auch ein Mechanismus, der die SDR im Langzeitspeicher wieder in den HTM kopiert, wenn sie zu einem späteren Zeitpunkt vermehrt nachgefragt werden. Ein solcher Mechanismus könnte ausgelöst werden, wenn im Kurzzeit-HTM besonders häufig Anomalien durch unbekannte Muster festgestellt werden. Dann bestünde immerhin die Hoffnung, im Langzeit-HTM fündig zu werden. Man könnte sich hier an modernen Rechnerarchitekturen orientieren, die verschiedenartige Speichertechniken mit gestaffelten Zugriffszeiten verwenden (Prozessorcache, RAM, Festplatte, Netzwerk).

Vermutlich werden die geschilderten Verfahren nicht der Weisheit letzter Schluss sein. Doch ist es meine subjektive Einschätzung, dass sie plausibel und elegant genug sind, um der Wahrheit zumindest ein Stück näherzukommen. Von ihrem biologischen Vorbild, dem Neokortex, übernehmen sie soviel Funktionalität wie möglich und so wenig Komplexität wie nötig. Sie stellen – ganz gemäß Ockhams Rasiermesser – eine

[26] Man denke z.B. an das Wetter von heute im Vergleich zum gestrigen.

Konzentration auf das Wesentliche dar. Die Verheißung eines biologisch inspirierten, universellen Lernverfahrens besteht deshalb weiter.

8 Literaturverzeichnis

Adrian26
Adrian, Edgar Douglas, and Yngve Zotterman. "The impulses produced by sensory nerve-endings Part II. The response of a Single End-Organ." *The Journal of physiology* 61.2 (1926): 151-171.

Bell97
Bell, Anthony J., and Terrence J. Sejnowski. "The “independent components” of natural scenes are edge filters." *Vision research* 37.23 (1997): 3327-3338.

Bengio2007
Bengio, Yoshua, et al. "Greedy layer-wise training of deep networks."*Advances in neural information processing systems* 19 (2007): 153.

Bengio2009
Bengio, Yoshua. "Learning deep architectures for AI." *Foundations and trends® in Machine Learning* 2.1 (2009): 1-127.

Binzegger2004
Binzegger, Tom, Rodney J. Douglas, and Kevan AC Martin. "A quantitative map of the circuit of cat primary visual cortex." *The Journal of Neuroscience*24.39 (2004): 8441-8453.

Bishop2006
Bishop, Christopher M. „*Pattern Recognition and Machine Learning“, S*pringer(2006).

Bowers2009
Bowers, Jeffrey S. "On the biological plausibility of grandmother cells: implications for neural network theories in psychology and neuroscience."*Psychological review* 116.1 (2009): 220.

Cajal06
Ramón y Cajal, S „The structure and connexions of neurons“(1906)

Carandini2006
Carandini, Matteo. "What simple and complex cells compute." *The Journal of physiology* 577.2 (2006): 463-466.

Carandini99
Carandini, Matteo, David J. Heeger, and J. Anthony Movshon. "Linearity and gain control in V1 simple cells." *Models of cortical circuits*. Springer US, 1999. 401-443.

Ciresan2010
Cireşan, Dan Claudiu, et al. "Deep, big, simple neural nets for handwritten digit recognition." *Neural computation* 22.12 (2010): 3207-3220.

Constantinople2013
Constantinople, Christine M., and Randy M. Bruno. "Deep cortical layers are activated directly by thalamus." *Science* 340.6140 (2013): 1591-1594.

Cornuz2007
Cornuz, Grégory, et al. "Object coding of harmonic sounds using sparse and structured representations." *Proceedings of the 10th International Conference on Digital Audio Effects*. 2007.

Daugman85
Daugman, John G. "Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters."*Optical Society of America, Journal, A: Optics and Image Science* 2.7 (1985): 1160-1169.

DeAngelis95
DeAngelis, Gregory C., Izumi Ohzawa, and Ralph D. Freeman. "Receptive-field dynamics in the central visual pathways." *Trends in neurosciences* 18.10 (1995): 451-458.

Deneve99
Deneve, Sophie, Peter E. Latham, and Alexandre Pouget. "Reading population codes: a neural implementation of ideal observers." *Nature neuroscience* 2.8 (1999): 740-745.

Eberhart95
Eberhart, Russell, and James Kennedy. "A new optimizer using particle swarm theory." *Micro Machine and Human Science, 1995. MHS'95., Proceedings of the Sixth International Symposium on*. IEEE, 1995.

Földiak2008
Földiák, P. and Endres, D. “Sparse Coding.“ Scholarpedia(2008)

Fukushima80
Fukushima, Kunihiko. "Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position."*Biological cybernetics* 36.4 (1980): 193-202.

George2008
George, Dileep. „*How the brain might work: A hierarchical and temporal model for learning and recognition*.“ Diss. Stanford University, 2008.

Hastad86
Hastad, Johan. "Almost optimal lower bounds for small depth circuits."*Proceedings of the eighteenth annual ACM symposium on Theory of computing*. ACM, 1986.

Hawkins2004
Hawkins, Jeff, and Sandra Blakeslee. "On intelligence: How a new understanding of the brain will lead to the creation of truly intelligent machines."*Henry Holt & Company, New York, NY* (2004).

Hawkins2005
George, Dileep, and Jeff Hawkins. "A hierarchical Bayesian model of invariant pattern recognition in the visual cortex." *Neural Networks, 2005. IJCNN'05. Proceedings. 2005 IEEE International Joint Conference on*. Vol. 3. IEEE, 2005.

Hawkins2009
Hawkins, Jeff, Dileep George, and Jamie Niemasik. "Sequence memory for prediction, inference and behaviour." *Philosophical Transactions of the Royal Society B: Biological Sciences* 364.1521 (2009): 1203-1209.

HBP2012
The Human Brain Project. „A Report to the European Commission“, 2012.

Hebb49
Hebb, Donald Olding. „*The organization of behavior: A neuropsychological theory.“ Reprint*. Psychology Press, 2002.

Hinton2006
Hinton, Geoffrey E., Simon Osindero, and Yee-Whye Teh. "A fast learning algorithm for deep belief nets." *Neural computation* 18.7 (2006): 1527-1554.

Hochreiter91
Hochreiter, Sepp. "Untersuchungen zu dynamischen neuronalen Netzen." Diplomarbeit*, Institut fur Informatik, Technische Universitat, Munchen*(1991).

Hodgkin52
Hodgkin, Alan L., and Andrew F. Huxley. "A quantitative description of membrane current and its application to conduction and excitation in nerve."*The Journal of physiology* 117.4 (1952): 500.

Hornik91
Hornik, Kurt. "Approximation capabilities of multilayer feedforward networks." Neural networks 4.2 (1991): 251-257.

Horton2005
Horton, Jonathan C., and Daniel L. Adams. "The cortical column: a structure without a function." *Philosophical Transactions of the Royal Society B: Biological Sciences* 360.1456 (2005): 837-862.

Hubel62
Hubel, David H., and Torsten N. Wiesel. "Receptive fields, binocular interaction and functional architecture in the cat's visual cortex." *The Journal of physiology*160.1 (1962): 106.

Hubel95
Hubel, David H. „*Eye, brain, and vision*.“ Scientific American Books, 1995.

Kandel2000
Kandel, Eric R., James H. Schwartz, and Thomas M. Jessell, eds. „*Principles of neural science*. Vol. 4.“ New York: McGraw-Hill, 2000.

Koob2009
Koob, Andrew. „*The Root of Thought: Unlocking Glia: the Brain Cell That Will Help Us Sharpen Our Wits, Heal Injury, and Treat Brain Disease.“* FT Press, 2009.

Krizhevsky2012
Krizhevsky, Alex, Ilya Sutskever, and Geoff Hinton. "Imagenet classification with deep convolutional neural networks." *Advances in Neural Information Processing Systems 25*. 2012.

Krüger88
Krüger, J., and F. Aiple. "Multimicroelectrode investigation of monkey striate cortex: spike train correlations in the infragranular layers." *Journal of neurophysiology* 60.2 (1988): 798-828.

Laserson2011
Laserson, Jonathan. "From Neural Networks to Deep Learning: zeroing in on the human brain." *XRDS: Crossroads, The ACM Magazine for Students* 18.1 (2011): 29-34.

LeCun2006
Poultney, Christopher, Sumit Chopra, and Yann LeCun. "Efficient learning of sparse representations with an energy-based model." *Advances in neural information processing systems*. 2006.

Lee2010
Lee, H. „NIPS 2010 Workshop onDeep Learning and Unsupervised Feature Learning: Tutorial on Deep Learning and Applications“, U. Of Michigan, 2010

Lund88
Lund, Jennifer S. "Anatomical organization of macaque monkey striate visual cortex." *Annual review of neuroscience* 11.1 (1988): 253-288.

Markram2012
Oroquieta, Felipe, Henry Markram, and Kathleen S. Rockland. "The neocortical column." *Frontiers in Neuroanatomy* 6.22 (2012): 1-2.

Minsky69
Minsky, Marvin, and Papert Seymour. "Perceptrons." MIT Press (1969).

Mountcastle57
Mountcastle, Vernon B. "Modality and topographic properties of single neurons of cat’s somatic sensory cortex." *J. Neurophysiol* 20.4 (1957): 408-434.

Mountcastle97
Mountcastle, Vernon B. "The columnar organization of the neocortex." *Brain*120.4 (1997): 701-722.

No34
Lorente de Nó, Rafael. "Studies on the structure of the cerebral cortex. II. Continuation of the study of the ammonic system." *Journal für Psychologie und Neurologie* (1934).

Numenta2011
„Hierarchisch Temporaler Speicher und HTM-basierte kortikale Lernalgorithmen“, Deutsche Übersetzung von Ingmar Baetge

Ohlshausen96
Olshausen, Bruno A. "Emergence of simple-cell receptive field properties by learning a sparse code for natural images." *Nature* 381.6583 (1996): 607-609.

Ohlshausen97
Olshausen, Bruno A., and David J. Field. "Sparse coding with an overcomplete basis set: A strategy employed by V1?." *Vision research* 37.23 (1997): 3311-3325.

Petreanu2007
Petreanu, Leopoldo, et al. "Channelrhodopsin-2–assisted circuit mapping of long-range callosal projections." *Nature neuroscience* 10.5 (2007): 663-668.

Price2011
Price, Ryan William. „*Hierarchical temporal memory cortical learning algorithm for pattern recognition on multi-core architectures*.“ Diss. Portland State University, 2011.

Ragazzi2013
D. Ragazzi, B. Matt, D. King, U. Kirschenmann, M. Ferrier et al., „openHTM“-Software, 2013

Rehn2007
Rehn, Martin, and Friedrich T. Sommer. "A network that uses few active neurones to code visual input predicts the diverse shapes of cortical receptive fields." *Journal of computational neuroscience* 22.2 (2007): 135-146.

Riesenhuber99
Riesenhuber, Maximilian, and Tomaso Poggio. "Hierarchical models of object recognition in cortex." *Nature neuroscience* 2.11 (1999): 1019-1025.

Rinkus2010
Rinkus, Gerard J. "A cortical sparse distributed coding model linking mini-and macrocolumn-scale functionality." *Frontiers in neuroanatomy* 4 (2010).

Roe92
Roe, Anna W., et al. "Visual projections routed to the auditory pathway in ferrets: receptive fields of visual neurons in primary auditory cortex." *The Journal of neuroscience* 12.9 (1992): 3651-3664.

Saul2003
Saul, Lawrence K., and Sam T. Roweis. "Think globally, fit locally: unsupervised learning of low dimensional manifolds." The Journal of Machine Learning Research 4 (2003): 119-155.

Schölkopf2002
Schölkopf, B.and Smola, A.J. „Learning with Kernels“, MIT Press(2002).

Stangor2012
Stangor, Charles. "Introduction to psychology." Flat World (2012).

Stein2005
Stein, Richard B., E. Roderich Gossen, and Kelvin E. Jones. "Neuronal variability: noise or part of the signal?." *Nature Reviews Neuroscience* 6.5 (2005): 389-397.

Tsunoda2011
Tsunoda, Kazushige, et al. "Complex objects are represented in macaque inferotemporal cortex by the combination of feature columns." *Nature neuroscience* 4.8 (2001): 832-838.

Wegener87
Wegener, Ingo. "The complexity of Boolean functions." John Wiley & Sons (1987).

9 Abbildungsverzeichnis

Sofern nicht bereits im Literaturverzeichnis aufgeführt. Einige Abbildungen sind - wie mittlerweile Vieles aus dem kollektiven Gedächtnis - Fundstücke einer Google-Bildsuche. Sie wurden verwendet, da sie ihren Gegenstand besonders gut illustrieren, auch wenn sich ihre Originalquelle nicht mehr ohne Weiteres ermitteln lässt.

Abb. 2
Medizinische Fakultät der University of Alabama at Birmingham,
http://themixuab.blogspot.de/2013/04/image-post-1-brain-message-superhighways.html.

Abb. 3
MedCell-Archiv der Yale University,
http://medcell.med.yale.edu/systems_cell_biology/nervous_system.php.

Abb. 4
http://commons.wikimedia.org/wiki/File:Blausen_0657_MultipolarNeuron.png.

Abb. 5-7, 11
Ursprünglich aus einem Lehrbuch (Pearson Educatio4n),
http://www.highlands.edu/academics/divisions/scipe/biology/faculty/harnden/2121/notes

Abb. 8
http://hyperphysics.phy-astr.gsu.edu/hbase/biology/actpot.html

Abb. 9
http://webspace.ship.edu/cgboer/theneuron.html

Abb. 10
http://commons.wikimedia.org/wiki/File:Blausen_0843_SynapseTypes.png

Abb. 12
http://www.nobelprize.org/nobel_prizes/medicine/laureates/1906/cajal-photo.html

Abb. 20
http://www.mpg.de/4688312/realistic_3D_brain_circuit

Abb. 21

http://mcb.berkeley.edu/courses/mcb64/cortex.html

Abb. 23

http://www.leica-microsystems.com/science-lab/the-patch-clamp-technique/

Abb. 30

http://www.scholarpedia.org/article/Neocognitron

Abb. 32

http://commons.wikimedia.org/wiki/File:Boltzmannexamplev1.png

Abb. 35

http://commons.wikimedia.org/wiki/File:HMMsequence.svg

Abb. 45-46

https://www.groksolutions.com/technology.html

Erklärung

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbstständig und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe. Alle Stellen, die wörtlich oder sinngemäß aus veröffentlichten und nicht veröffentlichten Schriften entnommen wurden, sind als solche kenntlich gemacht. Die Arbeit ist in gleicher oder ähnlicher Form oder auszugsweise im Rahmen einer anderen Prüfung noch nicht vorgelegt worden.

Stuttgart, den 13.12.2013

_____

Ralf Wittmann