Aus der Klinik für Neurologie

der Medizinischen Fakultät Charité - Universitätsmedizin Berlin

DISSERTATION

Optimale Stimuli in einem hierarchischen SFA-Netzwerk

zur Erlangung des akademischen Grades

Doctor medicinae (Dr. med.)

vorgelegt der Medizinischen Fakultät

Charité - Universitätsmedizin Berlin

von

Christian Hinze

aus Cottbus

Datum der Promotion:

Eidesstattliche Versicherung

Ich, Christian Hinze, versichere an Eides statt durch meine eigenhändige Unterschrift, dass ich die vorgelegte Dissertation mit dem Thema: Optimale Stimuli in einem hierarchischen SFA-Netzwerk selbstständig und ohne nicht offengelegte Hilfe Dritter verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel genutzt habe. Alle Stellen, die wörtlich oder dem Sinne nach auf Publikationen oder Vorträgen anderer Autoren beruhen, sind als solche in korrekter Zitierung (siehe „Uniform Requirements for Manuscripts (URM)“ des ICMJE -www.icmje.org) kenntlich gemacht. Die Abschnitte zu Methodik (insbesondere praktische Arbeiten, Laborbestimmungen, statistische Aufarbeitung) und Resultaten (insbesondere Abbildungen, Graphiken und Tabellen) entsprechen den URM (s.o) und werden von mir verantwortet.

Die Bedeutung dieser eidesstattlichen Versicherung und die strafrechtlichen Folgen einer unwahren eidesstattlichen Versicherung (§156,161 des Strafgesetzbuches) sind mir bekannt und bewusst.

Datum Unterschrift

Danksagung

Meinen ganz besonderen Dank möchte ich Herrn Prof. Dr. Laurenz Wiskott für seine Unterstützung, seine sehr intensive Betreuung und vor allem seine Geduld bei dieser Doktorarbeit aussprechen. Weiterhin danke ich Herrn Niko Wilbert für die konstruktiven Diskussionen und das Bereitstellen der erforderlichen SFA-Netzwerke. Herrn Prof. Dr. Curio danke ich für den Spielraum, diese Doktorarbeit in einem Grenzbereich zwischen Mathematik und Neurologie im Fach Medizin angefertigt haben zu dürfen.

Diese Arbeit wurde betreut durch

Prof. Dr. L. Wiskott

Theory of Neural Systems, Institut für Neuroinformatik, Ruhr-Universität Bochum

Prof. Dr. G. Curio

Klinik für Neurologie, Campus Benjamin Franklin, Charité Universitätsmedizin Berlin

Inhaltsverzeichnis

1 Einleitung **6**

2 Slow Feature Analysis **11**

3 Netzwerkarchitektur, Trainingsdaten und Methoden **16**

3.1 Netzwerk und Trainingsdaten 16

3.2 Das Optimierungsproblem 21

4 Ergebnisse und Interpretation **26**

4.1 Lokalisierungsverhalten bei Kugelnebenbedingung 27

4.1.1 Konkavität und Konvexität 27

4.1.2 Konkavität und Konvexität im SFA-Netzwerk 30

4.1.2.1 Die Maxima von Schicht 1 33

4.1.2.2 Die Maxima von Schicht 2 34

4.2 Andere Nebenbedingungen 38

5 Diskussion und Zusammenfassung **41**

A Bildergalerie **46**

Literaturverzeichnis **52**

Kapitel 1

Einleitung

Die Bildverarbeitung im menschlichen Gehirn durchläuft von der Retina an bis hin zu höheren Zentren wie beispielsweise dem inferioren Temporallappen (IT) viele verschiedene Verarbeitungsschichten. Dabei werden die rezeptiven Felder mit höherer Ebene immer größer und die verarbeiteten Informationen zur höheren Schicht hin immer komplexer. Dies geht bis hin zu hoch spezialisierten Zellen im IT, in welchem einzelne Zellen zur Erkennung eines bestimmten Gesichtes einer bestimmten Person gefunden wurden (Desimone 1991, Quiroga et al. 2005, Gross 2008).

Das visuelle System betreffend wurden vor allem rezeptive Felder von Zellen des primären visuellen Kortex V1 seit der grundlegenden Arbeit von Hubel und Wiesel aus dem Jahr 1962 (Hubel and Wiesel 1962) hinsichtlich ihrer Eigenschaften wie Invarianz und optimale Stimuli sehr gut untersucht. Durch die beiden genannten Autoren fand auch die klassische Unterscheidung von Zellen in V1 in komplexe und einfache Zellen statt. Beide Zelltypen können als Kanten- oder Balkendetektoren beschrieben werden, einfache Zellen reagieren am besten auf Balken einer bestimmten Orientierung und Position, komplexe Zellen hingegen reagieren bei bevorzugter Ausrichtung hinreichend invariant auf die genaue Position eines solchen Balkens. Mathematisch können sowohl einfache als auch komplexe Zellen mittels Gabor-Wavelets beschrieben werden (Pollen and Ronner 1981, Adelson and Bergen 1985, Jones and Palmer 1987).

Aus physiologischen Experimenten ist über die Struktur rezeptiver Felder höherer Schichten hingegen weit weniger bekannt, da Zellen höherer Schichten bevorzugt auf komplexere Stimuli reagieren, größere Invarianzen aufzeigen und sich somit ihre Untersuchung schwieriger gestaltet (Richmond et al. 1987). Es sind viele Ansätze unternommen worden, mit Hilfe verschiedener Stimuli-Sets und Paradigmen Zellen auf höheren Schich-

ten zu charakterisieren und zu klassifizieren (siehe z.B. Cadieu et al. 2007, Plebe 2012, Gegenfurtner et al. 1997, Baizer 1982, David et al. 2006, Richmond et al. 1987). Die Testung und Klassifizierung von Neuronen im IT mit herkömmlichen Stimuli wie unterschiedlichen Gabor-Wavelets ist nicht mehr zufriedenstellend möglich (Pollen et al. 1984). Andere Ansätze mit komplexeren Teststimuli zeigen hingegen ein z.T. deutlich differenzierteres Ansprechen (Richmond et al. 1987).

Mithilfe von mit quasi-natürlichen Bildsequenzen trainierten SFA-Netzwerken (SFA=*Slow Feature Analysis*, siehe Kap. 2) konnten bereits viele Eigenschaften wie optimale Stimuli und Invarianzen von Zellen in V1 reproduziert werden. So zeigen trainierte Zellen bereits nach einem SFA-Schritt viele Eigenschaften von Zellen in V1 wie Selektivität hinsichtlich Orientierung, Frequenz und z.T. Länge und Breite präsentierter Stimuli (Berkes and Wiskott 2005). Es finden sich zudem Zellen wieder, die entsprechend den Kriterien eines physiologischen Experimentes komplexen bzw. einfachen Zellen im biologischen Sinn entsprechen.

Der Vorteil eines SFA-Netzwerkes gegenüber dem biologischen Experiment ist die Zugänglichkeit für mathematisch-analytische Methoden. Wenn man davon ausgeht, dass die Verarbeitung im visuellen Kortex prinzipiell hierarchisch verläuft und sich die guten Übereinstimmungen von Zellen in V1 vor Augen hält, scheint es naheliegend zu fragen, ob nicht Eigenschaften von Zellen höherer Schichten eines hierarchischen SFA-Netzwerkes (also nach mehreren SFA-Schritten) Informationen über die rezeptiven Felder in höheren Schichten im visuellen Kortex liefern könnten. In dieser Arbeit geht es dabei um die Frage der optimalen Stimuli.

Gearbeitet wurde mit einem hierarchischen, vierschichtigen SFA-Netzwerk, welches mit quasi-natürlichen Bildsequenzen trainiert wurde. Die Architektur des Netzwerkes sowie die verwendeten Trainingsdaten werden ausführlich in Abschnitt 3.1 erläutert. Mittels eines Gradientenabstiegsverfahrens wurden numerisch die optimalen Stimuli für SFA-Zellen auf allen Schichten unter verschiedenen Nebenbedingungen ermittelt (Abschn. 3.2, Anhang A). Ein Ergebnis dieser Arbeit ist es, dass sich die optimalen Stimuli von Zellen ab Schicht 2 deutlich anders verhalten als dies von der ersten Schicht bekannt ist. Die optimalen Stimuli zeigen starke Lokalisierungen, wenn man -wie dies für einen SFA-Schritt in Berkes and Wiskott (2005) getan wurde- eine Kugelnebenbedingung wählt (siehe Abb. 1.1). Das dargestellte Beispiel zeigt ein charakteristisches Optimum einer Zelle in Schicht 2 bei Kugelnebenbedingung. Die Kugelnebenbedingung verlangt, dass der optimale Stimulus auf einer Kugeloberfläche einer bestimmten Norm gesucht werden muss. Es fällt bei dem so gewonnenen Stimulus in Abbildung 1.1 auf, dass fast die

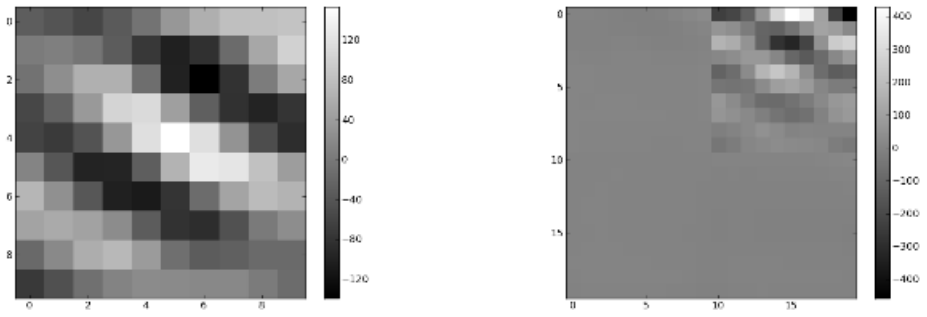

Abbildung 1.1: *Typische Maxima aus Schicht 1 und 2 unter Kugelnebenbedingung eines mit natürlichen Bildern trainierten SFA-Netzwerkes.* ***Links:*** *Dargestellt ist ein Maximum aus Schicht 1.* ***Rechts:*** *Ein repräsentatives Maximum einer Zelle aus Schicht 2. Fast die gesamte Bildnorm ist im rechten oberen Bildausschnitt konzentriert. Dieser Ausschnitt entspricht der Größe eines rezeptiven Feldes von Schicht 1.*

gesamte Energie in einer Fläche in der rechten oberen Bildhälfte konzentriert liegt. Diese Fläche entspricht dabei dem rezeptiven Feld einer Zelle aus Schicht 1. Es wird also im dargestellten Beispiel die gesamte Energie auf das rezeptive Feld einer Zelle aus Schicht 1 verwendet. Dieses Lokalisierungsphänomen zieht sich durch alle Schichten ab Schicht 2.

In Kapitel 4 wird dieses Lokalisierungsverhalten mathematisch begründet. Es wird sich zeigen, dass hierfür die für die Zellen bzw. den SFA-Algorithmus verwendete Funktionenmenge in Kombination mit der Kugelnebenbedingung verantwortlich ist. In unserem Fall waren die verwendeten Funktionen Polynome 2. Grades (siehe Kap. 2). In einem nächsten Schritt werden in Abschnitt 4.2 andere Nebenbedingungen hergeleitet und diskutiert, die dieses Lokalisierungsverhalten weniger oder überhaupt nicht zeigen (Abb. 1.2). Anzumerken ist auch, dass Zellen aus Schicht 1 unter diesen veränderten Nebenbedingungen ähnliche Ergebnisse zeigen wie für die Kugelnebenbedingung. Eine abschließende Diskussion findet in Kapitel 5 statt.

Fernab der Frage, in wie weit das Konzept optimaler Stimuli für höhere Schichten sinnvoll ist, zeigt diese Arbeit, dass die sonst gewählte Kugelnebenbedingung für höhere Schichten in SFA-Netzwerken ungünstig ist[1] und nicht gewählt werden sollte. Die Resultate der Optimierung unter geeigneteren Bedingungen wie z. B. der Würfelnebenbedingung

[1] Unter der Maßgabe, dass die zugrundeliegende Funktionenmenge Polynomen 2. Grades entspricht.

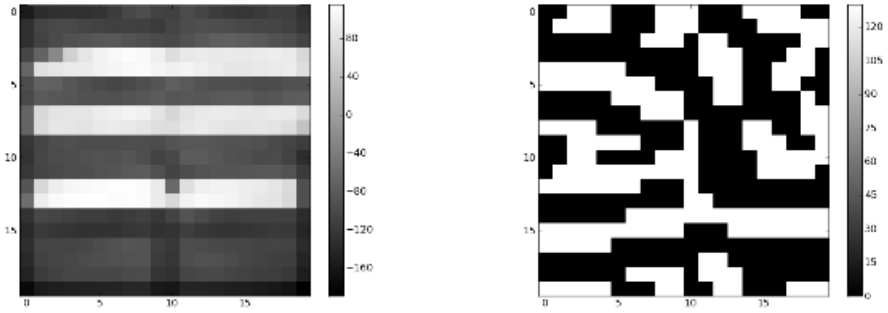

Abbildung 1.2: *Dargestellt sind repräsentative Maxima von Zellen aus Schicht 2 unter verschiedenen Nebenbedingungen.* ***Links:*** *Gezeigt wird ein Maximum aus Schicht 2 unter der Nebenbedingung* $\sum_{i=1}^{n} x_i^4 \leq const.$ ***Rechts:*** *Dargestellt ist ein Maximum aus Schicht 2 unter einer Würfelnebenbedingung, d.h.,* $max\{|x_1|, \ldots, |x_n|\} \leq const.$ *Die Konstanten wurden wie der Kugelradius als Mittelwert aus den Trainingsdaten gewonnen.*

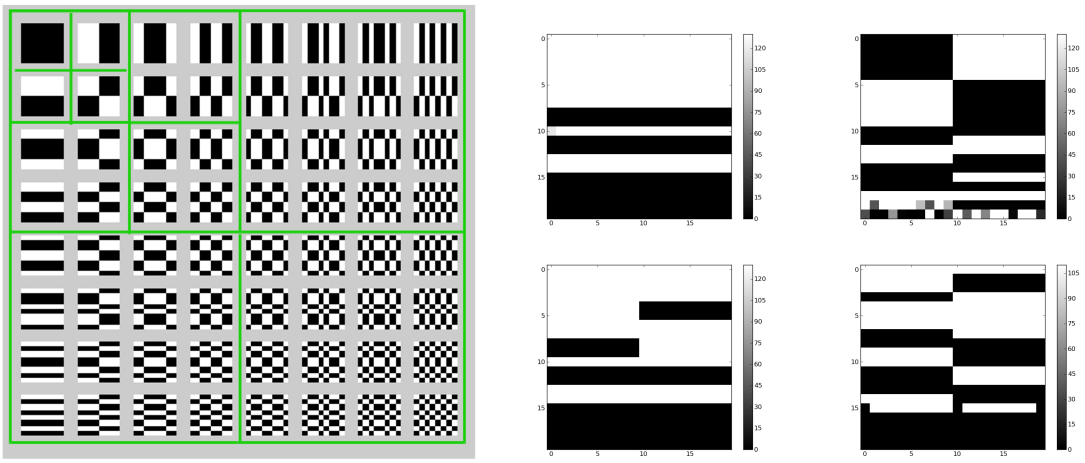

Abbildung 1.3: *Auf der* ***linken*** *Seite sind die Walsh-Hadamard-Basen für* 8×8 *Pixel messende Bilder dargestellt. Auf der* ***rechten*** *Seite exemplarische Maxima unter Würfelnebenbedingung von Schicht 2.*

zeigen, dass die Optima höherer Schichten rein qualitativ komplexer und die erkennbaren geometrischen Formen komplizierter werden. Dennoch bleiben die Optima schwer interpretierbar. Durch die bislang fehlende systematische Charakterisierung von Zellen höherer Schichten des visuellen Kortex fehlt ebenso eine Referenz zum Vergleich. Dennoch finden sich Optima unter Würfelnebenbedingung auch in bereits verwendeten Paradig-

men der Literatur. Hier sei vor allem auf eine Arbeit aus dem Jahr 1987 von Richmond et al. (1987) hingewiesen, in der zweidimensionale Walsh-Hadamard-Basen zur Testung des neuronalen Antwortverhaltens von Zellen im IT verwendet wurden. Ähnliche optimale Stimuli finden sich unter Würfelnebenbedigung auch im verwendeten SFA-Netzwerk wieder (Abb. 1.3). Eine ausführlichere Diskussion hiervon findet sich ebenfalls in Kapitel 5.

Kapitel 2

Slow Feature Analysis

Dieses Kapitel soll kurz das Prinzip der SFA (*Slow Feature Analysis*) erläutern. Für eine ausführlichere Einführung in die SFA sei auf Wiskott and Sejnowski (2002) verwiesen. Die Einführung in die SFA dieses Kapitels hält sich eng an die Publikationen von Wiskott et al. (insbesondere an Wiskott and Sejnowski 2002 und Berkes and Wiskott 2005) und übernimmt auch zum größten Teil die Notation. Durch eben diesen grundlegenden Charakter werden diese Arbeiten nur noch an vereinzelten Stellen im folgenden Text dieses Kapitels zitiert.

Der Gedanke hinter dem Ansatz von SFA ist es, aus schnell variierenden Eingangsdaten langsam variierende Informationen zu extrahieren. Beim visuellen System hieße dies bspw. aus vielen zeitlich sich schnell ändernden Rezeptorsignalen langsame Informationen wie Objektidentität oder Ausrichtung im Raum zu ermitteln (siehe hierzu Abbildung 2.1). Bei SFA werden unmittelbar über gelernte, nicht-lineare Transformationen langsam variierende Informationen aus den zur Verfügung stehenden Eingangsdaten gewonnen.

Möchte man dieses Prinzip mathematisch formulieren, so könnte man sagen, dass man in einer Menge (nicht-linearer) Transformationen F diejenigen herausfinden will, die über den Eingangsdaten $\mathbf{x}(t)$ (z.B. Bildinformation über die Zeit) am langsamsten variieren und dennoch Informationen transportieren. Desweiteren könnte man fordern, dass verschiedene aus den Eingangsdaten gewonnene Ausgabesignale voneinander unabhängig sein sollen. Formal geht es also darum, aus den Eingangsdaten $\mathbf{x}(t)$, $\mathbf{x}(t) \in \mathbf{R}^n$ Ausgangssignale $y_j(t) = g_j(\mathbf{x}(t))$, $j = 1, \ldots, N$ zu gewinnen, die (in einem noch zu definierenden Sinn) so langsam wie möglich über die Zeit variieren, aber gleichzeitig Informationen transportieren, also nicht-trivial sind. Die Schreibweise $y_j(t) = g_j(\mathbf{x}(t))$ soll suggerieren, dass die y_j als Funktionen in Abhängigkeit vom Eingangsvektor $\mathbf{x}(t)$ zu verstehen sind.

Die Funktionen g_j sind beispielsweise Elemente eines bestimmten Funktionenraumes. Als Transformationsmenge F wird meist der Vektorraum der Polynome 2. Grades in n Variablen benutzt. Die Zahl n entspricht der Dimensionalität der Eingangsdaten, also z.B. eines Bildes. Legt man sich auf den Vektorraum der Polynome 2. Grades in n Variablen fest, so hat jedes $g \in F$ eine Darstellung, $g(x) = \frac{1}{2}\mathbf{x}^T\mathbf{H}\mathbf{x} + \mathbf{f}^T\mathbf{x} + c$, $\mathbf{H} \in \mathbf{R}^{n\times n}$, $\mathbf{f} \in \mathbf{R}^n$, $c \in \mathbf{R}$. Es sei aber darauf hingewiesen, dass die SFA nicht an diesen speziellen Raum gebunden ist, sondern auf prinzipiell jedem Vektorraum vollzogen werden kann.

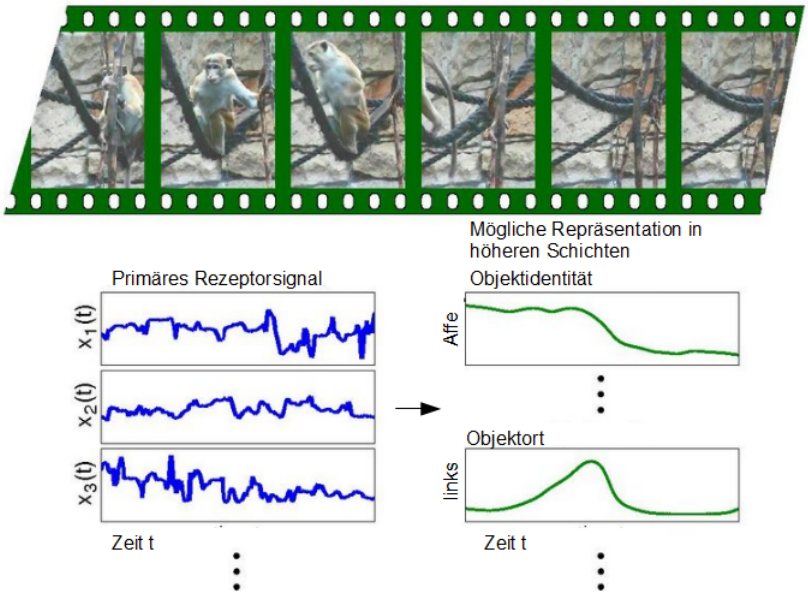

Abbildung 2.1: *Die Abbildung soll das Langsamkeitsprinzip von SFA motivieren. Der Filmstreifen* ***oben*** *zeigt einen Affen, der im Laufe der Zeit den Bildausschnitt nach links verlässt. Im* ***linken unteren Feld*** *sind mögliche primäre Rezeptorsignale z.B. der Retina dargestellt, die im Grunde nur die Helligkeit/Farbe sehr kleiner Bildauschnitte messen und üblicherweise schnell mit der Zeit variieren. Sie sind die einzige optische Datenquelle, aus der die Information der oben dargestellten Bildszene rekonstruiert wird. Der Mensch ist in der Lage, daraus instantan langsam variierende Informationen wie z.B. das dargestellte Objekt und seine Position im Bild zu bestimmen, die im* ***rechten unteren Bildbereich*** *über die Zeit dargestellt sind. Man könnte also davon ausgehen, dass das Gehirn gelernt hat, diese langsam variierenden Signale aus den vielen sich schnell ändernden Rezeptorsignalen zu extrahieren. Ein Prinzip, das man versucht hat, mittels SFA zu formalisieren.* Quelle: http://www.scholarpedia.org/article/Slow_feature_analysis

Das Optimierungsproblem (das Finden der am langsamsten variierenden Funktionen), das es zu lösen gilt, lässt sich aus Obigem wie folgt formulieren:

Sei mit $\mathbf{x}(t)$ ein beliebiges $n-$dimensionales Eingangssignal in Abhängigkeit von der

Zeit gegeben. Finde auf einer noch zu definierenden Menge von Funktionen in $\mathbf{x}$ diejenigen Funktionen $\mathbf{g} := (g_1, \ldots, g_N)$, die folgendes Optimierungsproblem lösen ($y_j(t) := g_j(\mathbf{x}(t)), \forall j$),

$$\langle \dot{y}_j^2(t) \rangle_t \quad = \quad \min, \tag{2.1}$$

unter den Nebenbedingungen, dass

$$\begin{aligned} \langle y_j(t) \rangle_t &= 0 \quad \text{(Erwartungswert 0)}, \\ \langle y_j^2(t) \rangle_t &= 1 \quad \text{(Varianz 1)}, \\ \langle y_i(t) y_j(t) \rangle_t &= 0 \; \forall i < j \quad \text{(Dekorreliertheit und Ordnung)}. \end{aligned} \tag{2.2}$$

Mit $\langle . \rangle_t$ ist die zeitliche Mittelung und mit $\dot{y}_i(t)$ ist die zeitliche Ableitung gemeint. Wird dieses Optimierungsproblem gelöst, dann repräsentiert der Vektor $\mathbf{y} = (y_1, \ldots, y_N)$ die (nach dieser Definition) am langsamsten variierenden Funktionen auf dem Eingangssignal $\mathbf{x}(t)$ beginnend mit dem langsamsten Ausgangssignal $y_1(t)$. Die Signale variieren mit steigendem Index immer schneller.

Wird das Problem auf einem endlich-dimensionalen Vektorraum gestellt, so kann man das Lösen der Optimierungsaufgabe auf das Lösen eines verallgemeinerten Eigenwertproblems reduzieren. Hat der Vektorraum Dimension k, dann kann eine Basis $\{h_1, \ldots, h_k\}$ gewählt werden. Jedes $y_j(t) = g_j(\mathbf{x}(t))$ lässt sich dann offensichtlich wiefolgt schreiben,

$$y_j(t) = g_j(\mathbf{x}(t)) = \sum_{i=1}^{k} w_i^{(j)} h_i(\mathbf{x}(t)). \tag{2.3}$$

Das Problem reduziert sich also darauf, die Wichtungsvektoren $\mathbf{w}^{(j)}$ bzw. Koeffizienten $w_i^{(j)}$ zu finden. Sei hierzu $\mathbf{h}(\mathbf{x}(t)) := (h_1(\mathbf{x}(t)), \ldots, h_k(\mathbf{x}(t)))^T - \mathbf{h_0}$, wobei $\mathbf{h_0} := \langle \mathbf{h} \rangle_t$ dem zeitlichen Mittel entspricht, so dass $\mathbf{h}$ Erwartung 0 hat. Dies kann man ohne Einschränkung der Allgemeinheit annehmen. Dann lösen die geeignet normierten Eigenvektoren des verallgemeinerten Eigenwertproblems (vEWP),

$$\mathbf{AW} = \mathbf{BW\Lambda}, \tag{2.4}$$

mit

$$\mathbf{A} : \quad = \quad \langle \dot{\mathbf{h}} \dot{\mathbf{h}}^T \rangle_t \quad \text{(Kovarianzmatrix der Zeitableitungen)},$$

$$\mathbf{B} \;:=\; \langle \mathbf{h}\mathbf{h}^T \rangle_t \quad \text{(Kovarianzmatrix)}, \tag{2.5}$$

die Optimierungsaufgabe der SFA. Die Matrix $\mathbf{\Lambda}$ hat Diagonalform und als Einträge die verallgemeinerten Eigenwerte von (2.4).

Dies ist wie folgt zu sehen: Wie oben schon erwähnt, kann man davon ausgehen, dass $\langle \mathbf{h} \rangle_t = 0$. Das zieht automatisch nach sich, dass auch jede lineare Kombination von Komponenten von $\mathbf{h}$ Erwartungswert 0 hat, denn

$$\langle g_j(\mathbf{x}(t)) \rangle_t = \sum_{i=1}^{k} w_{i,j} \langle h_i(\mathbf{x}(t)) \rangle_t = 0. \tag{2.6}$$

Damit wäre also die erste Nebenbedingung von (2.2) erfüllt. Das SFA-Optimierungsproblem fordert desweiteren, dass

$$\frac{\langle \dot{y}_j^2 \rangle_t}{\langle y_j^2 \rangle_t} = \frac{\mathbf{w_j}^T \mathbf{A} \mathbf{w_j}}{\mathbf{w_j}^T \mathbf{B} \mathbf{w_j}} \tag{2.7}$$

minimal wird. Man sucht also nach dem Minimum der Funktion $f(\mathbf{w}) := \frac{\mathbf{w^T A w}}{\mathbf{w^T B w}}$. Für diese Funktion gilt offenbar $f(\lambda \mathbf{w}) = f(\mathbf{w})$ für alle $\lambda \neq 0$ und alle $\mathbf{w}$. Der Gradient von f ist gegeben durch

$$\nabla f(\mathbf{w}) = \frac{2 \left[\mathbf{w}^T \mathbf{B} \mathbf{w}\right] \mathbf{A}\mathbf{w} - 2 \left[\mathbf{w}^T \mathbf{A} \mathbf{w}\right] \mathbf{B}\mathbf{w}}{\left[\mathbf{w}^T \mathbf{B} \mathbf{w}\right]^2}. \tag{2.8}$$

Durch Nullsetzen des Zählers erhält man daraus

$$2 \left[\mathbf{w}^T \mathbf{B} \mathbf{w}\right] \mathbf{A}\mathbf{w} - 2 \left[\mathbf{w}^T \mathbf{A} \mathbf{w}\right] \mathbf{B}\mathbf{w} = 0. \tag{2.9}$$

Vektoren $\mathbf{w_j}$, die das vEWP $\mathbf{AW} = \mathbf{BW\Lambda}$ lösen, erfüllen auch die notwendige Bedingung (2.9) für ein Minimum, denn mit $\mathbf{\Lambda} = diag(\lambda_1, \ldots, \lambda_k)$ ergibt sich

$$\begin{aligned} 2 \left[\mathbf{w_j}^T \mathbf{B} \mathbf{w_j}\right] \mathbf{A}\mathbf{w_j} - 2 \left[\mathbf{w_j}^T \mathbf{A} \mathbf{w_j}\right] \mathbf{B}\mathbf{w_j} &= 2\lambda_j \left[\mathbf{w_j}^T \mathbf{A} \mathbf{w_j}\right] \mathbf{A}\mathbf{w_j} \\ &\quad -2\lambda_j \left[\mathbf{w_j}^T \mathbf{A} \mathbf{w_j}\right] \mathbf{A}\mathbf{w_j} \\ &= 0. \end{aligned} \tag{2.10}$$

Alle anderen Nicht-Eigenvektoren $\mathbf{v}$ lassen sich eineindeutig wie folgt schreiben

$$\mathbf{A}\mathbf{v} = \mu \mathbf{B}\mathbf{v} + r_\perp, \tag{2.11}$$

mit $\mu \in \mathbf{R}$ und $[\mathbf{Bv}]^T \mathbf{r}_\perp = 0$. Daraus ergibt sich

$$\begin{aligned} 2\left[\mathbf{v}^T\mathbf{Bv}\right]\mathbf{Av} - 2\left[\mathbf{v}^T\mathbf{Av}\right]\mathbf{Bv} &= 2\left[\mathbf{v}^T\mathbf{Bv}\right] r_\perp - 2\left[\mathbf{v}^T r_\perp\right]\mathbf{Bv} \\ &\neq 0. \end{aligned} \tag{2.12}$$

Minimierende Vektoren sind also notwendigerweise Eigenvektoren des vEWP in (2.4). Da der Wert des Terms in (2.7) nicht von der Norm des Eigenvektors abhängt, kann man diesen entspechend der Bedingungen des SFA-Optimierungsproblems wählen. Es ist weiterhin anzumerken, dass die Funktion f offensichtlich keine lokalen Maxima besitzt. Außerdem sind die so gewonnenen Funktionen unkorrelliert im Sinne von (2.2), denn es gilt $\mathbf{w_i}^T\mathbf{Bw_j} = 0$, $i \neq j$ und somit

$$\begin{aligned} \langle y_i y_j \rangle_t &= \langle \mathbf{w_i}^T\mathbf{h}\mathbf{w_j}^T\mathbf{h} \rangle_t \\ &= \langle \mathbf{w_i}^T\mathbf{h}\mathbf{h}^T\mathbf{w_j} \rangle_t \\ &= \langle \mathbf{w_i}^T\mathbf{Bw_j} \rangle_t = 0. \end{aligned} \tag{2.13}$$

Offensichtlich wird die langsamste Funktion durch den Eigenvektor zum kleinsten Eigenwert gegeben usw. Damit ist durch Lösen des vEWP auch das Optimierungsproblem der SFA gelöst. Sollten z.B. mehrere Eigenvektoren den gleichen (kleinsten) Eigenwert haben, so wäre die Lösung des SFA-Algorithmus' nicht mehr eindeutig, da jede Linearkombination langsamste Funktion wäre. Das gilt analog für jeden anderen Eigenwert. Dazu ist zu sagen, dass es bisher in den Simulationen nicht zu diesem Fall gekommen ist, sondern die Eigenwerte stets einen 1−dimensionalen Eigenraum hatten und die Lösung der langsamsten, zweitlangsamsten usw. Funktion stets eindeutig war.

Wie schon weiter oben im Text erwähnt, verwendeten wir in unserem Fall Polynome zweiten Grades in $\mathbf{x}$. Es wurde dieser Raum gewählt, weil er relativ niedrig-dimensional ist und dennoch nicht-lineare Transformationen erlaubt. Eine Basis kann sofort mittels der entsprechenden Monome $\mathbf{h} = (x_1, \ldots, x_n, x_1^2, \ldots, x_n^2, x_1x_2, \ldots)$ angegeben werden.

Natürlich kann man das Lernproblem der SFA auch für den linearen Fall lösen. Eine Basis $\{h_1, \ldots, h_n\}$ kann leicht angegeben werden, mit $h_i(\mathbf{x}(t)) := x_i(t)$. Die Matrizen $\mathbf{B}$ und $\mathbf{A}$ entprechen dann der Kovarianzmatrix und der Kovarianzmatrix der Zeitableitungen des Eingangssignals $\mathbf{x} \in \mathbf{R}^n$.

Erwähnt sei, dass das Optimierungsproblem auch auf unendlich dimensionalen Vektorräumen mit Hilfe der Variationsrechnung formuliert und gelöst werden kann (siehe hierzu Franzius et al. 2007).

Kapitel 3

Netzwerkarchitektur, Trainingsdaten und Methoden

3.1 Netzwerk und Trainingsdaten

Den in Kapitel 2 beschriebenen Algorithmus kann man selbstverständlich über beliebig viele Schritte fortführen. Dabei dienen die Outputs nach einem SFA-Schritt als neue Eingangs- und Trainingsdaten für den nächsten SFA-Schritt. Das für diese Arbeit verwendete, vierschichtige Netzwerk ist in Abbildung 3.4 dargestellt und wird im Folgenden genauer erläutert.

Kernstück jeder Schicht bildet das in Abbildung 3.1 gezeigte Element. Das dargestellte Element beinhaltet den eigentlichen SFA-Schritt. Auf Schicht 1 wird es mit Bildsequenzen einer Größe von 10×10 Pixeln trainiert. Die eigentlichen Trainingsbilder hatten eine Größe von 92×92 Pixeln mit Grauwerten zwischen 0 und 255, von denen das Netzwerk aber der Einfachheit halber aus technischen Gründen nur einen 90×90 Pixel großen Ausschnitt sieht. Ein Problem mit dem in Kapitel 2 beschriebenen Algorithmus ergibt sich aus der Dimensionalität des (quadratisch) expandierten Signals. In unserem Fall würde sich diese bei einer Eingangsdimensionalität von 8100 bereits auf $90 \times 90 + \frac{90 \times 90 \times [90 \times 90 + 1]}{2} = 32,817,150$ belaufen, was den Rechenaufwand schnell unbeherrschbar macht. Eben aus diesem Grund reduziert man die Eingangsdimensionalität eines SFA-Elements wie in Abbildung 3.1 und verwendet dann Kopien dieses Elements, um das gesamte Bild abzudecken. Dabei gehen natürlich mögliche geometrisch weit voneinander entfernte Korrelationen verloren. Doch auch im visuellen Kortex des Menschen

sehen bspw. Neurone in V1 nur Ausschnitte des Bildes, Zellen in höheren Schichten wiederum größere Bereiche (siehe z. B. Smith et al. 2001). Eine Zelle in Schicht 4 unseres Netzwerks sieht das gesamte Bild von 90×90 Pixeln.

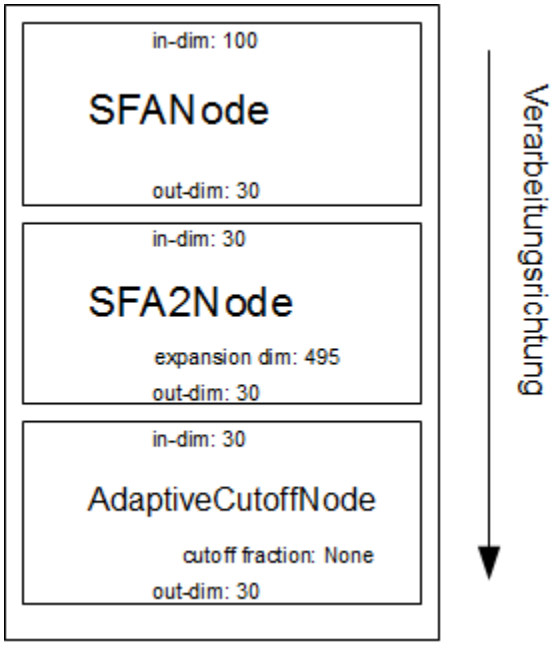

Abbildung 3.1: *Ausschnitt aus Schicht 1 (Siehe auch Abb. 3.4). Solche Elemente finden sich in jeder Schicht des Netzwerkes wieder. In ihnen ist der SFA-Schritt implementiert. Mit* in-dim *und* out-dim *sind jeweils Dimension der Eingangs- und Ausgangsdaten bezeichnet.* ***Der erste Knoten*** *ist ein linearer SFA-Knoten (*SFANode*) zur Dimensionsreduktion (in diesem Fall von* 100 *auf* 30*). In diesem Knoten findet keine quadratische Expansion des Signals statt, sondern es erfolgt ein linearer SFA-Schritt (siehe Ende Kapitel 2). Als* ***nächster Knoten*** *und Verarbeitungsschritt folgt der SFA2-Knoten (*SFA2Node*) mit quadratischer Expansion wie in Kapitel 2 beschrieben. Dieser beinhaltet also die auf den Ausgängen des linearen Knotens am langsamsten variierenden Polynome zweiten Grades in -in diesem Fall-* 30 *Variablen. Die Ausgänge des Knotens repräsentieren die* 30 *langsamsten Funktion, beginnend mit der langsamsten.* ***Die Cut-off-Knoten*** *(*AdaptiveCutoffNode*) sind optional und haben in unserem Netzwerk keinen Einfluss auf den Ausgabewert. Wenn sie aktiviert sind, besteht die Möglichkeit, Outputs über und/oder unter einem gewissen Wert abzuschneiden.*

Es werden also die SFA-Elemente trainiert und dann geklont, so dass die benötigte Eingangsdimensionalität erreicht werden kann. Am Beispiel von Schicht 1 ist dies in Abbildung 3.2 weiter ausgeführt. In Schicht 1 werden 81 Kopien des SFA-Elements benötigt, welches selbst nur einen Bildausschnitt von 10×10 Pixeln sieht.[1] Der Begriff

[1] Die Trainingsdaten des SFA-Elements von Schicht 1 sind dann alle 81 Bildsequenzen der Größe 10×10 der ursprünglichen Bildsequenz.

des rezeptiven Feldes wird hier aufgrund der Netzwerknomenklatur mehrdeutig benutzt. Zum einen bezeichnet er die Größe und Beschaffenheit des Eingangsdatenfeldes bezogen auf die nächst untere Schicht und zum anderen den gesehenen Ausschnitt des Ursprungsbildes. An den entsprechenden Stellen wird explizit darauf hingewiesen werden, welche Bedeutung gemeint ist.

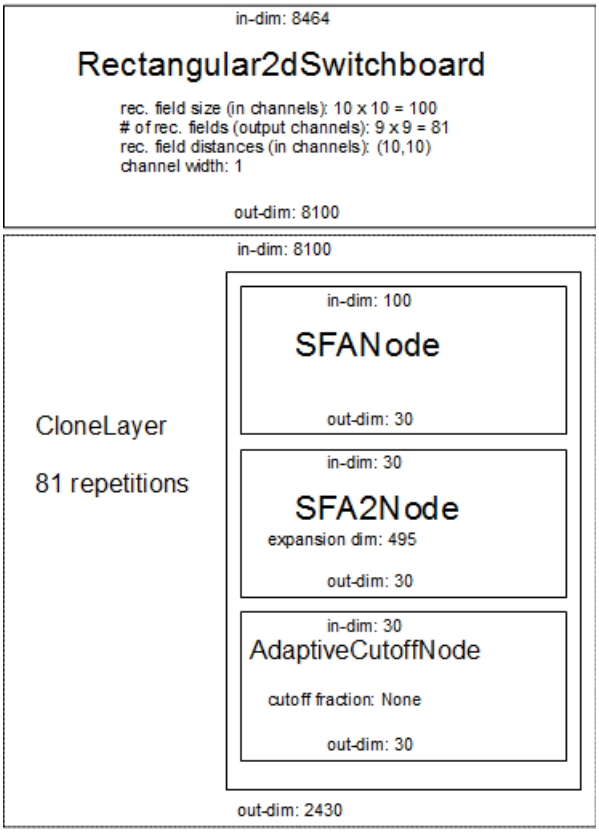

Abbildung 3.2: *Dargestellt ist die vollständige Schicht 1 des verwendeten Netzwerkes. Das SFA-Element ist wie in Abbildung 3.1. Neu hinzugekommen sind das Switchboard (*Rectangular2dSwitchboard*) und der Clone layer-Knoten (*CloneLayer*). Die* ***Funktion des Switchboards*** *ist die Bereitstellung der richtigen Eingangsvektoren für jedes einzelne Klon des SFA-Elements. Es hat keinen Einfluss auf die Daten selbst, sondern lediglich sortierende Funktion. Die Einträge im Kästchen des Switchboards geben Auskunft über die Struktur der rezeptiven Felder der jeweiligen Schicht (hier Schicht 1). Alle Größen werden in* channels *angegeben. Ein* channel *fasst eine bestimmte Anzahl von Ein- oder Ausgangsdaten zusammen. Bspw. entspricht ein Bildpixel des Eingangsbildes, also des Eingangssignals von Schicht 1, einem* channel (*demzufolge* channel width: 1). *Entsprechend der dargestellten Struktur hat Schicht 1* $81 \times 30 = 2430$ *Ausgänge. Jedes SFA-Element hat* 30 *Ausgänge und wird in Schicht 2 zu einem* channel *zusammengefasst. Die Einträge* rec. field size, # of rec. fields, rec. field distances *entsprechen Größe, Anzahl und Abstand der rezeptiven Felder. Die Begrifflichkeiten werden in Abbildung 3.3 noch einmal anhand von Schicht 2 erläutert.*

Die Struktur eines rezeptiven Feldes für Schicht 2 auf den Ausgängen von Schicht 1 ist in Abbildung 3.3 erläutert. In Schicht 2 überlappen sich zum ersten Mal rezeptive Felder bezogen auf die darunter liegende Schicht und das Ursprungsbild. Die Struktur der anderen Schichten ist in gleicher Art in Abbildung 3.4 dargestellt. Auf Schicht 4 gibt es nur noch ein rezeptives Feld und somit nur ein SFA-Element, welches den gesamten Ausgang von Schicht 3 und somit das gesamte Ursprungsbild sieht.

Die grundlegende Architektur des Netzwerkes ist übernommen von schon zuvor erfolgreich verwendeten Netzwerken, mit denen bspw. die Unterscheidung von Buchstaben unabhängig von Größe, Position und Orientierung sowie die Unterscheidung anderer natürlicher Objekte wie Bildern von Fischen in unterschiedlicher Perpektive möglich war.

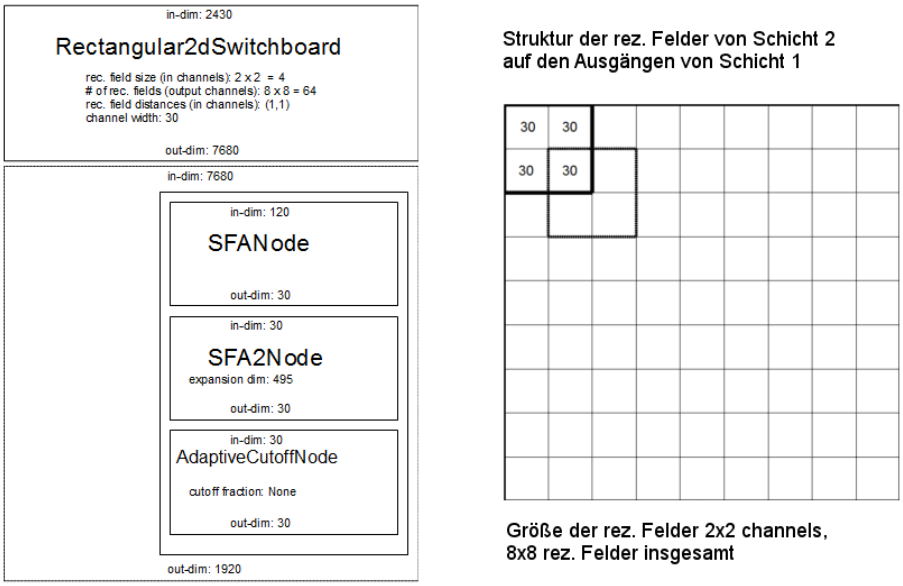

Abbildung 3.3: *Auf der* ***linken*** *Seite dargestellt, ist ein Ausschnitt aus Abbildung 3.4, der die gesamte Schicht 2 zeigt. Die Eingangsdimensionalität ist* 2430*, was genau den* 81×30 *Ausgängen von Schicht 1 entspricht. Die* 30 *Ausgänge eines SFA-Elements von Schicht 1 werden nun zu einem* channel *zusammengefasst (*channel width: 30*). Ein rez. Feld von Schicht 2 auf den Ausgängen von Schicht 1 hat eine Größe von* 2×2 *channels. Dies ist schematisch auf der* ***rechten*** *Seite abgebildet, hier nicht zu verwechseln mit dem rezeptiven Feld eines SFA-Elements von Schicht 2 auf dem Ursprungsbild. Die Eingangsdaten für ein SFA-Element auf Schicht 2 sind also die Ausgänge von* 2×2 *SFA-Elementen von Schicht 1, die Eingangsdimensionalität ist somit* $2 \times 2 \times 30 = 120$*. Die rez. Felder überlappen sich zudem um je einen* channel *(*rec. field distances: 1,1*), so dass es insgesamt* 8×8 *rez. Felder gibt und entsprechend* 64 *Kopien des SFA-Elements auf Schicht 2. Da ein SFA-Element die Ausgänge von* 4 *SFA-Elementen von Schicht 1 sieht, entspricht dies einem gesehenen Bereich von* 20×20 *Pixeln im Ursprungsbild.*

Abbildung 3.4: *Dargestellt ist der Aufbau des für die Experimente verwendeten hierarchischen SFA-Netzwerks als Übersicht. Es besteht aus vier Schichten, die von oben links bis unten rechts dargestellt sind. Der Ausdruck „CloneLayer ... repetitions" gibt an, wieviele Einheiten der daneben dargestellten SFA-Elemente in einer Schicht vorhanden sind. Eine detaillierte Beschreibung der Netzwerkmerkmale findet sich im Text und in den Abbildungen.*

Als Trainingsdaten wurden auf eine geeignete Größe skalierte Naturfilme von *National Geographic* verwendet, um möglichst nah an der physiologischen Situation zu sein. Die Filme wurden mit *pyglet* (www.pyglet.org) auf die passende Auflösung gebracht. Die Trainingssequenzen hatten eine Länge von ca. 10000 Bildern à 90×90 (92×92) Pixeln.

3.2 Das Optimierungsproblem

Ohne den Quellcode im Detail aufzuführen und zu besprechen, sollen im folgenden Kapitel die grundlegenden Prinzipien der verwendeten Verfahren erläutert werden. Um an die Optima der Zellen zu gelangen, müssen entsprechende Maximierungs- bzw. Minimierungsproblem für die einzelne Polynome (Zellen) unter bestimmten Nebenbedingungen gelöst werden. Eine analytische Lösung wie im quadratischen Fall (nach einem SFA-Schritt, siehe Berkes and Wiskott 2005) ist im Regelfall nicht mehr möglich. Die Literatur bietet hier jedoch eine Fülle von Methoden, um solche Probleme numerisch anzugehen. Die für diese Arbeit genutzten Algorithmen finden sich im Buch von R. Fletcher (Fletcher 2000). Für unsere Simulationen wurden zwei Optimierungsmethoden dieses Buches implementiert, zum einen ein Quasi-Newton-Ansatz mittels BFGS-Algorithmus (Fletcher 2000: S. 49ff) und ein herkömmlicher Gradientenabstieg. Der Quasi-Newton-Algorithmus gibt über verschiedene Auswertungen des Gradienten eine Näherung für die Hesse-Matrix vor. Aufgrund der hohen Dimensionalität des Problems erwies sich dieser Ansatz jedoch als zu langsam und somit nicht geeignet. Der normale Gradientenabstieg benötigte zwar weit mehr Iterationen, war jedoch letztlich deutlich schneller und wurde für alle weiteren Simulationen genutzt. Das gesamte Programm dieser Arbeit wurde in *python* unter Nutzung des MDP (*Modular toolkit for Data Processing, http://mdp-toolkit.sourceforge.net/*) geschrieben. Zur weiteren Erläuterung des Optimierungsalgorithmus wird im Folgenden z. T. die englische Terminologie von Fletcher (Fletcher 2000) benutzt.

Das Prinzip der Optimierung unter allen Nebenbedingungen war stets so, dass per Vorauswahl zwischen verschiedenen Startpunkten gewählt werden konnte, dann wurde die Abstiegsrichtung (aus adaptiertem (neg.) Gradienten) berechnet und per Liniensuche (*line search*, Fletcher 2000: S. 33ff) der nächste akzeptable Punkt (*acceptable point*) bestimmt. Beendet wurde das Verfahren, wenn die Norm der Projektion des Gradienten auf den Tangentialraum der jeweiligen Nebenbedingungsmannigfaltigkeit einen bestimmten Wert unterschritt. Dieser Wert ergab sich aus verschiedenen Testdurchläufen und wurde

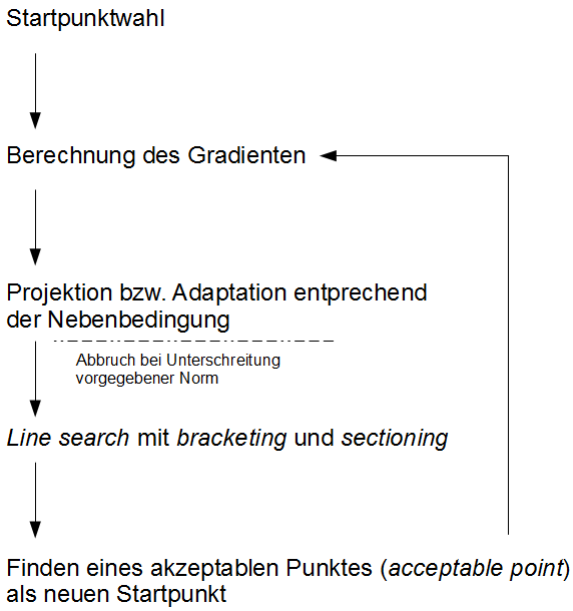

Abbildung 3.5: *Schematische Darstellung des Optimierungsalgorithmus'.*

zu höheren Schichten hin größer. Die Abbruchsgröße der Norm bewegte sich zwischen 10^{-9} für Schicht 1 und ca. 10^{-6} für Schicht 4. Das Prinzip ist trotz seiner Einfachheit noch einmal in Abbildung 3.5 graphisch dargestellt.

Nun sollen die einzelnen Verfahrensabschnitte noch detaillierter besprochen werden. Die wählbaren Startpunktoptionen waren ein homogener Startpunkt (entsprechende Norm, gleicher Wert in allen Pixeln), ein gegebener Startpunkt (z.B. aus einer externen Datei), eine Zufallsstartpunkt geeigneter Norm oder ein zufälliger Startpunkt, der mittels eines natürlichen Bildes generiert wurde. Der homogene Startpunkt spielte natürlich auf höheren Schichten eine untergeordnete Rolle, da es sich zumeist um Zellen mit mehreren lokalen Maxima und Minima handelte. Die anderen Startpunktvarianten waren zur Konsistenztestung implementiert worden. Die Maxima und Minima ließen sich unter allen Bedingungen stets gut reproduzieren, so dass letztlich die Option des reinen Zufallsstartpunktes für die meisten Simulationen genutzt wurde. Der Radius der Kugelnebenbedingung und die Kantenlänge der Würfelnebenbedingung ergaben sich als Durchschnitt der Trainingsdaten.

Die Abstiegsrichtung ergab sich wie dies aus der Literatur bekannt ist, aus dem (neg.) Gradienten, welcher aus den SFA-Netzwerk berechnet werden kann, und der entsprechenden Nebenbedingungsmannigfaltigkeit (Kugel,Würfel usw.). Die Suchrichtung der *line search* war die Projektion des (neg.) Gradienten auf den Tangentialraum der ent-

sprechenden Mannigfaltigkeit im Startpunkt. Für die Kugel wurde dann das Problem per *line search* auf der Kreislinie mit eben dieser Suchrichtung als Tangentialvektor betrachtet. Die Herangehensweise für den Würfel und andere Nebenbedingungen wird weiter unten gesondert erläutert. Resultat dieses Schrittes war also eine Trajektorie bestimmter Länge auf der Kugel (in den meisten Fällen ein Halbkreis), auf der nun geeignete Punkte für den nächsten Iterationsschritt gesucht werden konnten.

a.
```
for i in range(0,sp.dim):
    if searchDir[0][i]<0:
        sp.startPoint[0][i] = 0.
    if searchDir[0][i]>0:
        sp.startPoint[0][i] = cubeLength
    if searchDir[0][i] == 0.:
        zeroCount = zeroCount + 1
```

b.
```
for i in sp.boundCompsMax:
    if searchDir[0][i]>=0:
        searchDir[0][i] = 0
for i in sp.boundCompsMin:
    if searchDir[0][i]<=0:
        searchDir[0][i] = 0
```

Abbildung 3.6: ***a.*** *Dargestellt ist die* ***vorgeschaltene Schleife****, die vor der eigentlichen Optimierung unter Würfelnebenbedingung durchgeführt wurde. Mit* sp *ist ein Startpunkt-Objekt bezeichnet, mit den entsprechenden Attributabfragen* sp.dim *und* sp.startPoint*, ersteres gibt die Dimension des Startpunktes, letzteres den aktuellen Startpunkt als array zurück. In der Variable* searchDir *ist die aktuelle Suchrichtung hinterlegt (bereits an den Würfel angepasst). Ist eine Komponente der Suchrichtung (*searchDir*) negativ, so wird die entsprechende Startpunktkomponente auf* 0 *gesetzt, ist sie positiv, wird die Startpunktkomponente auf die Würfelkantenlänge gesetzt, d.h., auf den Rand des Würfels in dieser Komponente. Gezählt werden am Ende die Nullkomponenten der Suchrichtung (*zeroCount*) in den so gewonnenen neuen Startpunkten.* ***b.*** *Dargestellt ist die Anpassung des (neg.) Gradienten an den Würfel. Die Atrribute* sp.boundCompsMax *bzw.* sp.boundCompsMin *beinhalten alle Komponenten des aktuellen Startpunktes, die bereits der Würfelkantenlänge entsprechen bzw.* 0 *sind. Eine Komponente der Suchrichtung wird dann auf* 0 *gesetzt, wenn sie auf dem Rand des Würfels aus dem Würfel heraus zeigt. D.h., wenn sie positiv für eine Startpunkt-Komponente ist, die bereits auf Würfelkantenlänge ist oder negativ für eine Startpunkt-Komponente, die* 0 *ist, gezählt wird dann, wie schon erwähnt, die Anzahl der Nullkomponenten der Suchrichtung im neuen Startpunkt.*

Ein Kernstück des implementierten Verfahrens bildete die nun folgende *line search*, die vollständig aus dem Buch von Fletcher übernommen wurde (Fletcher 2000: S. 33ff). Die festzulegenden Konstanten wurden z. T. entsprechend der Empfehlung des Autors wie folgt gewählt: $\rho = 0.01$, $\sigma = 0.4$, $\tau_1 = 7$, $\tau_2 = 0.1$, $\tau_3 = 0.5$. Die *bracketing phase* und die *sectioning phase* erfolgten wie im Buch beschrieben. Einige Parameter der *line search*, die z. B. die Länge der betrachteten Trajektorie (im Fall der Kugel πR, also ein Halbkreis) für *bracketing* und *sectioning* bestimmt, mussten eigenständig gewählt werden und sind problemabhängig. Es hat sich als praktikabel erwiesen, dass, falls auf der gegebenen Trajektorie keine geeigneten Punkte im *bracketing* gefunden werden konnten,

der Endpunkt der entsprechenden Trajektorie als neuer Startpunkt gewählt wurde.

Auf den $n-$Kugeln ($\sum x_i^n = const,\ n > 2$) konnte das soeben beschriebene Prinzip analog fortgesetzt werden. Etwas andere Probleme brachte die Würfelnebenbedigung sowohl für Startpunkt, Abstiegsrichtung als auch *line search* mit sich. In den Simulationen bestätigte sich die intuitive Vermutung, dass sich die Extrema in Würfelecken befinden. Startet man mit einem Zufallsbild oder einem anderen Startpunkt, der selbst kein Würfeleckpunkt ist, in die Optimierung, so findet der Algorithmus oft lange keine geeigneten Intervalle in der *bracketing phase*, so dass der Startpunkt immer wieder neu (wie oben beschrieben) am Ende der Trajektorie der *line search* festgelegt wird. Es bedarf bei z. B. einer Dimensionalität von 8100 für Schicht 4 sehr vieler Iterationsschritte, bis überhaupt ein Würfeleckpunkt (oder ein Punkt in der Nähe) erreicht ist. Wir entschieden uns deswegen in diesem Fall zur Durchführung einer vorgeschaltenen Schleife vor der eigentlichen Optimierung, die einen geeigneten Punkt in der Nähe eines Würfeleckpunktes abhängig vom initialen Startpunkt als neuen Startpunkt bestimmt. Das Kernstück dieser vorgeschaltenen Schleife ist in Abbildung 3.6 dargestellt. Die genannte vorgeschaltene Schleife setzt eine Startpunktkomponente auf Würfelkantenlänge (also auf den Würfelrand), wenn die Suchrichtung in dieser Komponente positiv ist, also prinzipiell auf den Würfelrand zu zeigt und die entsprechende Startpunktkomponente noch nicht selbst der Würfelkantenlänge entspricht. Dies geschieht analog für den Fall, dass die Suchrichtung in einer Startpunktkomponente negativ ist, dann wird die entsprechende Startpunktkomponente auf 0 gesetzt. In einem ideal auskonvergierten Algorithmus würde sich ein Extremum in einer Würfelecke befinden[2] und alle Komponenten der Suchrichtung aus dem Würfel an diesem Punkt herauszeigen. Dies kann die vorgeschaltene Schleife nur in Ausnahmefällen bewerkstelligen. In der vorgeschaltenen Schleife werden aber diese Startpunktkomponenten auf dem Würfelrand (Punktkomponente 0 und negative Suchrichtungskomponente bzw. Punktkomponente in Höhe der Würfelkantenlänge und positive Suchrichtungskomponente) durch die Variable zeroCount gezählt (siehe Abb. 3.6). Die vorgeschaltene Schleife wird beendet, wenn die Variable zeroCount über eine bestimmte Anzahl von Durchläufen (in unseren Simulationen wurde die Anzahl auf 20 gesetzt) nicht wächst. Mit dem dann gewonnenen Startpunkt wird die bekannte Optimierung durchgeführt.

Die Länge der Trajektorie für die *line search* unter Würfelnebenbedingung muss natürlich jedesmal abhängig vom aktuellen Startpunkt neu bestimmt werden. Die Trajektorien

[2]Das bedeutet nicht, dass es keine Extrema außerhalb der Würfelecken geben kann. Unter unseren Rahmenbedingungen war dies jedoch nie der Fall.

selbst sind Geraden auf der Würfeloberfläche bis zum nächsten Rand. Die Anpassung des (neg.) Gradienten an die Würfeloberfläche ist Abbildung 3.6 b. zu entnehmen. Zur Visualisierung der genannten vorgeschaltenen Schleife ist eine Bildsequenz Startpunkt, neuer Startpunkt nach vorgeschaltener Schleife und in der Optimierung gewonnenes Extremum in Abbildung 3.7 dargestellt. Z. T. konnten große Teile der Optimierung in der vorgeschaltenen Schleife absolviert werden.

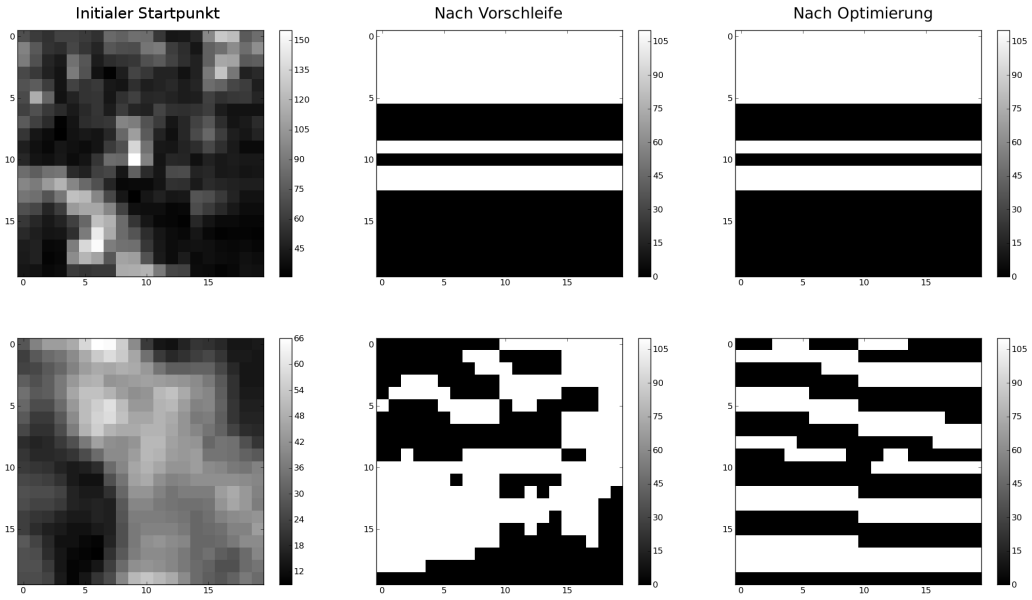

Abbildung 3.7: *Dargestellt sind Sequenzen initialer Startpunkt (**links**, hier zufällig aus einem natürlichen Bild) - neuer Startpunkt nach vorgeschaltener Schleife (**mittig**) - Extremum nach Optimierung (**rechts**) anhand zweier Zellen aus Schicht 2.*

Kapitel 4

Ergebnisse und Interpretation

Die Ergebnisse der Simulationen sind in Anhang A dargestellt. Es finden sich dort die Minima und Maxima für Zellen aller vier Schichten des verwendeten Netzwerkes unter verschiedenen Nebenbedingungen. Die nicht aufgeführten Stimuli (z.B. weitere lokale Extrema oder Extrema höherer Zellen) unterscheiden sich qualitativ nicht von den dargestellten.

Bei einem ersten Betrachten fällt ein Merkmal der Bilder unmittelbar auf, wenn man sich zunächst die gefundenen Optima (und damit sind in diesem Kontext sowohl Minima als auch Maxima gemeint) unter Kugelnebenbedingung ansieht (siehe Abb. A.1, A.2, A.3 und A.4). Wo unter Kugelnebenbedingung die Stimuli für Schicht 1 noch das gesamte rezeptive Feld (auf das Ursprungsbild bezogen) ausfüllen, kommt es ab Schicht 2 zu einer Lokalisierung des Optimums auf ein, maximal zwei rezeptive Felder der Größe eines rezeptiven Feldes von Schicht 1 (in unserer Netzwerk-Architektur also ein Bild der Größe 10×10 Pixel). Dieses Phänomen wird in Schicht 3 und 4 noch ausgeprägter.[1]Dies ist offensichtlich nicht der Fall für die Würfelnebenbedingung, bei der die optimalen Stimuli in jeder Schicht das gesamte rezeptive Feld ausfüllen (Abb. A.5). Im Folgenden soll diskutiert werden, warum es im Fall der Kugelnebenbedingung zu eben einer solchen Lokalisierung bei den optimalen Stimuli kommt und warum dies bei der Würfelnebenbedingung nicht auftritt.

[1]Es sei angemerkt, dass die Optima von Schicht 1 der Vollständigkeit halber angeführt wurden. Diese werden, wie weiter oben im Text beschrieben, ausführlich u.a. in Berkes and Wiskott (2005) diskutiert.

4.1 Lokalisierungsverhalten bei Kugelnebenbedingung

4.1.1 Konkavität und Konvexität

Der Grund für das beschriebene Phänomen der Lokalisierung in den optimalen Stimuli liegt im unterschiedlichen Verhalten konkurrierender konkaver und konvexer Funktionen im Rahmen bestimmter Optimierungsprobleme mit Randbedingungen begründet. Dies soll im Folgenden motiviert werden.

Hierzu nehme man eine skalare, nicht negative Funktion m und betrachte folgendes Optimierungsproblem,

$$\begin{aligned} m(h_1) + m(h_2) &= max, \\ h_1 + h_2 &= h_0, \\ h_i &\geq 0. \end{aligned} \tag{4.1}$$

Anschaulich könnte das heißen, dass man eine Gesamtenergie h_0 zur Verfügung hat, die man auf in diesem Fall zwei gleich geartete Kanäle verteilen kann. Die Frage wäre jetzt, wie man die Energie so verteilt, dass die Summe maximal (minimal) wird. Offensichtlich hängt das sehr stark von den Eigenschaften der Funktion m ab. Zwei prinzipielle Fälle sollen betrachtet werden, die uns auch bei der Optimierung der SFA-Zellen begegnen.

In den beiden Fällen ist die Funktion m einmal streng konkav ($m''(h) < 0,\ \forall h$) und einmal streng konvex ($m''(h) > 0,\ \forall h$). In diesen einfachen Fälle muss man nicht mit Gradienten und Lagrange-Multiplikatoren arbeiten, wenn man h_2 durch h_1 substituiert. Es folgt ein eindimensionales Maximierungsproblem mit folgender notwendigen Bedingung,

$$[m(h_1) + m(h_0 - h_1)]' = 0. \tag{4.2}$$

Rechnet man dies mit der Kettenregel aus, so ergibt sich,

$$m'(h_1) - m'(h_0 - h_1) = 0. \tag{4.3}$$

Ein erster Kandidat für ein Extremum ergibt sich also im Punkt $h_1 = h_2 = \frac{h_0}{2}$. Außer auf den Rändern ($h_1 = 0$ und $h_2 = 0$) kann aufgrund der Bedingung $m''(h) < 0,\ \forall h$ bzw. $m''(h) < 0,\ \forall h$ kein weiterer Kandidat hinzukommen. Im konkaven Fall ist die zweite Ableitung im Punkt $h_1 = h_2 = \frac{h_0}{2}$ negativ, es handelt sich also um ein Maximum,

umgekehrt im konvexen Fall. Die Randpunkte ($h_1 = 0$ oder $h_2 = 0$) sind im konkaven Fall Minima, im konvexen die Maxima. Die Fälle sind graphisch in Abbildung 4.1 dargestellt.

Zur weiteren Motivation soll nun an einem stark vereinfachten, an das SFA-Optimierungsproblem angelehnten Beispiel diese Situation weiter verdeutlicht werden. Das folgende Beispiel veranschaulicht den konvexen Fall und die prinzipielle Situation für Zellen in Schicht 2. Man nehme hierzu die Funktionen $f(x) = x^2$ und $g(y_1, y_2) = y_1^2 + y_2^2$. Gegeben sei nun das folgende Optimierungsproblem,

$$\begin{aligned} g(f(x_1), f(x_2)) &= max, \\ x_1^2 + x_2^2 &= h_0. \end{aligned} \tag{4.4}$$

Es gilt,

$$g(f(x_1), f(x_2)) = x_1^4 + x_2^4. \tag{4.5}$$

Im Beispiel repräsentiert die Funktion g eine Zelle auf Schicht 2, die Eingangsdaten von zwei Zellen von Schicht 1 ($f(x_1)$ und $f(x_2)$) empfängt. Ein rezeptives Feld von Schicht 1 hat in diesem Fall die Größe eines Pixels und eine Zelle in Schicht 2 sieht zwei rezeptive Felder von Schicht 1. Gleichung 4.5 soll unter der entsprechenden Nebenbedingung $x_1^2 + x_2^2 = h_0$ maximiert werden. Betrachtet man die rechte Seite von Gleichung 4.5, so ist die Situation auch als konkurrierendes Problem zweier Unterprobleme betrachtbar, beide Terme, x_1^4 und x_2^4, sind prinzipiell bestrebt die gesamte Energie auf sich zu vereinen. Stellt man einem Kanal oder Pixel (x_1 oder x_2) die Energie h zur Verfügung, so beträgt der erreichbare Maximalwert für die Terme x_1^4 oder x_2^4 mit dieser Energie offensichtlich h^2. Damit kann man die Maximalwertfunktion in Abhängigkeit von der Energie h definieren und sie lautet für beide Terme, $m(h) = h^2$. Umformuliert kann man das Problem in 4.4 (mit h_1 und h_2 werden die Energien, die x_1 und x_2 zur Verfügung gestellt werden, bezeichnet) also auch wie folgt schreiben,

$$\begin{aligned} m(h_1) + m(h_2) &= max, \\ h_1 + h_2 &= h_0. \end{aligned} \tag{4.6}$$

Dies entspricht aber genau der Situation der Ausgangsbetrachtung wie in Gleichung 4.1. Im gerade betrachteten Fall ist die Funktion m natürlich streng konvex, sie entspricht der quadratischen Parabel. Durch die hohe Potenz vom Grad 4 wird also unter quadratischer Norm oder Kugelnebenbedingung die Maximalwertfunktion konvex und es ist zum

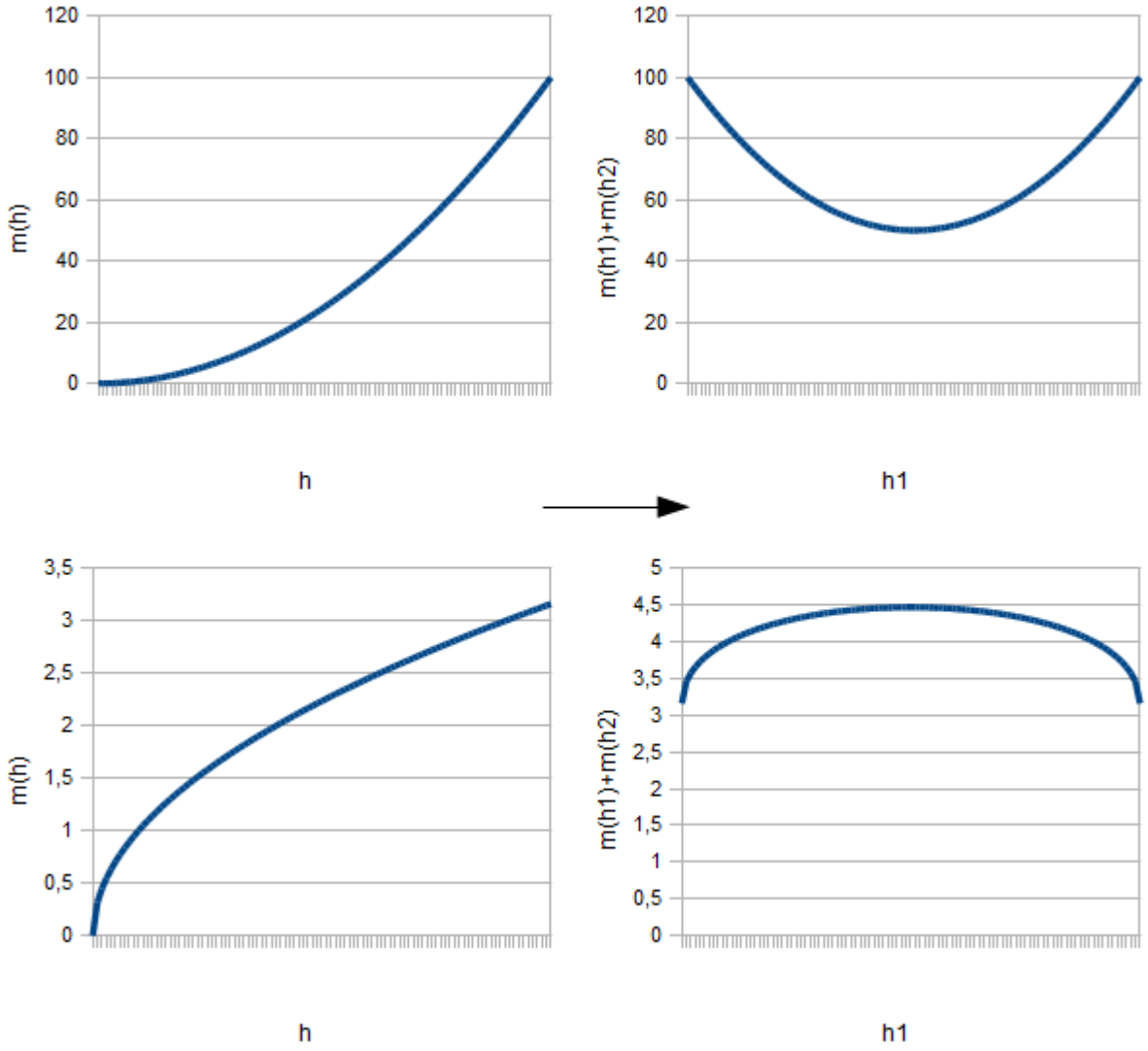

Abbildung 4.1: *Dargestellt sind die beiden im Text diskutierten Fälle einer konkaven bzw. konvexen Funktion m.* ***Oben links*** *dargestellt ist ein mögliches Beispiel einer streng konvexen Funktion m.* ***Oben rechts*** *wird die Summe in diesem Fall gezeigt, ein Maximum ist nur bei vollständiger Verwendung der Energie auf eine der Variablen (x_1, x_2 bzw. h_1, h_2) erreichbar. Auf der x-Achse ist h_1 aufgetragen, welches von 0 bis h_0 alle Werte durchläuft. Analoge Kurven finden sich* ***unten*** *für den konkaven Fall.*

Erreichen eines Maximalwertes sinnvoller, die gesamte Energie in nur einen Bildpixel oder Kanal (x_1 oder x_2) zu stecken, in diesem Fall wäre der Maximalwert h_0^2. Im Fall einer Gleichverteilung der Energie wäre der erreichbare Funktionswert nur $\frac{h_0^2}{4} + \frac{h_0^2}{4} = \frac{h_0^2}{2}$.

Die Situation ist völlig anders, wenn das Maximierungsproblem bspw. wie folgt aussieht,

$$\begin{aligned} x_1 + x_2 &= max, \\ x_1^2 + x_2^2 &= h_0. \end{aligned} \tag{4.7}$$

In diesem Fall kann der Term x_i in Abhägngigkeit von der zur Verfügung gestellten Energie h (also bspw. $x_1^2 = h$) maximal den Wert $\sqrt{h}$ erreichen, die Funktion m hat also die Form $m(h) = \sqrt{h}$ und ist somit streng konkav. Würde man nun die gesamte Energie auf einen Pixel verwenden, so wäre der erreichbare Funktionswert $\sqrt{h_0}$. Wenn man die Energie gleichmäßig auf beide Kanäle verteilt, so läge der erreichbare Funktionswert der Summe bei $2\sqrt{\frac{h_0}{2}} = \sqrt{4\frac{h_0}{2}} = \sqrt{2h_0} > \sqrt{h_0}$.

Es mag verkomplizierend scheinen, dass mit einem Konstrukt wie der Funktion m gearbeitet wird, denn die prinzipielle Situation bleibt für jedes SFA-Netzwerk wie im eben dargelegten Beispiel, durch die zweite Schicht treten Potenzen vierten Grades auf, die umgangssprachlich die quadratische Norm übersteigen und somit zum Lokalisierungsphänomen führen. Nur besitzt die Funktion m im höherdimensionalen Fall eine kompliziertere Geometrie, die in Abhängigkeit von der Norm h_0 differenziertere Betrachtungen erfordert (siehe Abb. 4.6). Die Lokalisierung ab einem bestimmten Radius ist aber prinzipiell wie im genannten Beispiel.

4.1.2 Konkavität und Konvexität im SFA-Netzwerk

Ähnliche Prinzipien finden sich nun auch bei den Optima unter Kugelnebenbedingung ab Schicht 2 im SFA-Netzwerk. Es sollen hier nur die Maxima von Schicht 1 und 2 diskutiert werden. Man wird sehen, dass die Lokalisierung mit höherer Schicht weiter zunehmen muss. Es finden sich zudem leicht analoge Argumente und Abschätzungen für die entsprechenden Minima.

Der Aufbau eines rezeptiven Feldes von Schicht 2 ist in Abbildung 4.2 dargestellt. Wie bereits erwähnt, ist eine Zelle in jeder Schicht, so auch in Schicht 2, eine quadratische Funktion in den Ausgängen der nächst unteren Schicht. Für eine Zelle in Schicht 2 soll

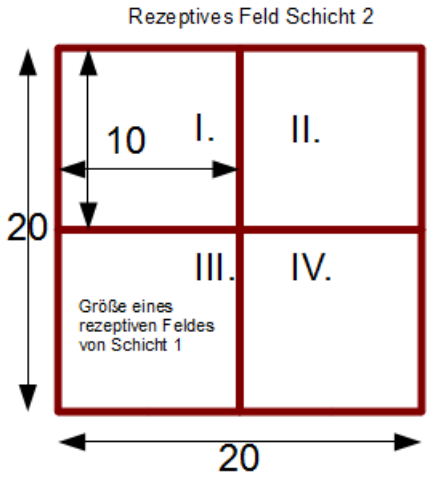

Abbildung 4.2: *Rezeptives Feld von Schicht 2 bezogen auf das Ursprungsbild. Längen in Pixel. Jede Zelle in Schicht 2 sieht entsprechend der Architektur des Netzwerkes vier rezeptive Felder von Schicht 1 (nummeriert mit I.-IV.). Jede einzelne SFA-Zelle von Schicht 1 hat* 10×10 *Eingänge und* 30 *Ausgänge (siehe Kapitel 3.1).*

sie schematisch in folgender Form geschrieben werden, $\frac{1}{2}\mathbf{y}^T\mathbf{H}\mathbf{y}+\mathbf{f}^T\mathbf{y}+c,\ \mathbf{y} \in \mathbf{R}^{120},\ \mathbf{H} \in \mathbf{R}^{120\times 120},\ \mathbf{f} \in \mathbf{R}^{120},\ c \in \mathbf{R}$. Das in Abbildung 4.2 dargestellte rezeptive Feld zeigt, dass die vier von einer Zelle in Schicht 2 "gesehenen" rezeptiven Felder von Schicht 1 völlig unabhängig voneinander reguliert werden können. Alles, was eingehalten werden muss, ist die Randbedingung $r^2 = const$ bzw. $r_I^2 + r_{II}^2 + r_{III}^2 + r_{IV}^2 = const$. Die Variablen r_I etc. stehen für die Norm des entsprechenden rezeptiven Feldes von Schicht 1.
Vorab sei gesagt, dass die quadratischen Formen ab Schicht 2 und höher von ihrer Beschaffenheit her nicht ein bestimmtes rezeptives Feld in Schicht 1 präferieren, was sonst die Diskussion erheblich vereinfachen würde. Im Gegenteil, die vier rezeptiven Felder sind rein energetisch (von der Matrixnorm her gesehen) völlig gleichgestellt.
Man kann sich das Optimierungsproblem in Schicht 2 nun auch als konkurrierendes Optimierungsproblem vieler quadratischer (auf das Input-Bild bezogen dann sogar Ordnung 4) Subprobleme vorstellen. Dies ist in Abbildung 4.3 für den quadratischen Term dargestellt. Analog kann man sich die entsprechenden linearen Subterme bilden. Schreibt man sich die Matrix der betrachteten Zelle in Schicht 2, $\mathbf{H} \in \mathbf{R}^{120\times 120}$, und den entsprechen-

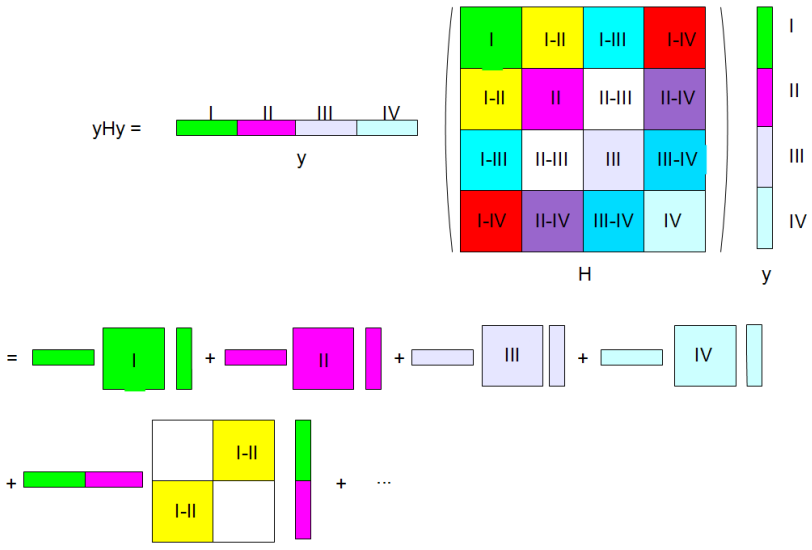

Abbildung 4.3: *Darstellung einer quadratischen Form $y^t Hy$ von Schicht 2 als Summe quadratischer Unterformen.*

den Linearterm $f \in \mathbf{R}^{120}$ wie folgt, $\mathbf{H_{ij}} \in \mathbf{R}^{30\times 30}$, $\mathbf{f_i} \in \mathbf{R}^{30}$,

$$\mathbf{H} = \begin{pmatrix} \mathbf{H_{11}} & \mathbf{H_{12}} & \mathbf{H_{13}} & \mathbf{H_{14}} \\ \mathbf{H_{21}} & \mathbf{H_{22}} & \mathbf{H_{23}} & \mathbf{H_{24}} \\ \mathbf{H_{31}} & \mathbf{H_{32}} & \mathbf{H_{33}} & \mathbf{H_{34}} \\ \mathbf{H_{41}} & \mathbf{H_{42}} & \mathbf{H_{43}} & \mathbf{H_{44}} \end{pmatrix}, \quad \mathbf{f} = \begin{pmatrix} \mathbf{f_1} \\ \mathbf{f_2} \\ \mathbf{f_3} \\ \mathbf{f_4} \end{pmatrix}, \tag{4.8}$$

dann schreibt sich das Optimierungsproblem einer Zelle in Schicht 2,

$$\underbrace{\sum_{i=1}^{4} \underbrace{\left[\frac{1}{2} \mathbf{y_i}^T \mathbf{H_{ii}}\, \mathbf{y_i} + \mathbf{f_i}^T \mathbf{y_i} + \overbrace{\frac{c}{4}}^{=:c_i} \right]}_{=:QT_i} + mQT_{ij}}_{} = max, \tag{4.9}$$

$$r^2 = r_I^2 + r_{II}^2 + r_{III}^2 + r_{IV}^2 = const. \tag{4.10}$$

Das Kürzel mQT_{ij} steht hier für 'mixed quadratic terms', also die verbleibenden gemischten Terme der Form,

$$mQT_{ij} = \frac{1}{2} \left(\mathbf{y_i}^T, \mathbf{y_j}^T \right) \begin{pmatrix} & \mathbf{H_{ij}} \\ \mathbf{H_{ji}} & \end{pmatrix} \begin{pmatrix} \mathbf{y_i} \\ \mathbf{y_j} \end{pmatrix}, \; i \neq j, \; i < j. \tag{4.11}$$

Die gemischten Terme sind bestrebt, die Energie auf die beiden eingehenden Vektoren gleich zu verteilen.

Im folgenden kurzen Abschnitt wird erläutert, warum es nicht zur Lokalisierung auf Schicht 1 kommt.

4.1.2.1 Die Maxima von Schicht 1

Sei $F(\mathbf{x}) := \frac{1}{2}\mathbf{x}^T\mathbf{G}\mathbf{x} + \mathbf{d}^T\mathbf{x} + r$ irgendeine quadratische Form einer SFA-Zelle aus Schicht 1. Es gelte ferner die Nebenbedingung $g(\mathbf{x}) := \sum_{i=1}^{n} x_i^2 = h > 0$. Die Funktion m sei die Maximalwertfunktion, d.h., $m(h)$ ist das Maximum der Funktion F für Norm h. Dann gilt

$$m(h) \sim qh + p\sqrt{h} + r,\ p,\ q,\ r \in R \setminus \{0\},\ \forall h > 0. \tag{4.12}$$

Wobei mit dem Zeichen $\sim$ ein qualitatives 'verhält sich wie' gemeint ist. Die Funktion $m(h) := qh + p\sqrt{h} + r$ ist stetig und streng konkav.
Zur Begründung sei mit $\mathbf{e}$ der normierte Eigenvektor (EV) zum größten Eigenwert (EW) von H bezeichnet. Sei $\mathbf{z}$ ein weiterer normierter Vektor, so dass $\mathbf{z}^T\mathbf{G}\mathbf{z}$, $\mathbf{d}^T\mathbf{z} > 0$[2]. Dann gilt,

$$\frac{1}{2}h\left(\mathbf{z}^T\mathbf{G}\mathbf{z}\right) + \sqrt{h}\mathbf{d}^T\mathbf{z} + r \leq m(h) \leq \frac{1}{2}h\left(\mathbf{e}^T\mathbf{G}\mathbf{e}\right) + \sqrt{h}\|\mathbf{d}\|^2 + r. \tag{4.13}$$

Mit $q_{max} := \frac{1}{2}\mathbf{e}^T\mathbf{G}\mathbf{e}$, $q_z := \frac{1}{2}\left(\mathbf{z}^T\mathbf{G}\mathbf{z}\right)$und $\kappa = \mathbf{d}^T\mathbf{z}$ ergibt sich,

$$q_z h + \kappa\sqrt{h} + r \leq m(h) \leq q_{max}h + \|\mathbf{d}\|^2\sqrt{h} + r. \tag{4.14}$$

Anschaulich ist $m(h)$ zwischen Funktionen eingeschlossen, die die ganze Zeit streng konkav sind, aber zunehmend immer linearer werden. Dies ist für eine Zelle aus Schicht 1 in Abbildung 4.4 dargestellt.
Natürlich kann man das Optimierungsproblem auf Schicht 1 genauso als Optimierungsproblem mit konkurrienden quadratischen Unterformen schreiben (siehe dazu Gleichung (4.10)). Die Unterformen selbst verhalten sich aber unter Optimierung bei Kugelnebebedingung ebenso wie die zuvor beschriebene Funktion F, was dem konkaven Fall in Abschnitt 4.1.1 entspricht. Die Optima werden also das gesamte rezeptive Feld ausfüllen und die Bildenergie wird gleichmäßig verteilt.

[2] $\mathbf{H}$ hat eine ONB von EV und ein gleichmäßig im Positiven und Negativen verteiltes EW-Spektrum, zudem ist $\mathbf{d}$ in unserem Netzwerk nie selbst EV von $\mathbf{H}$.

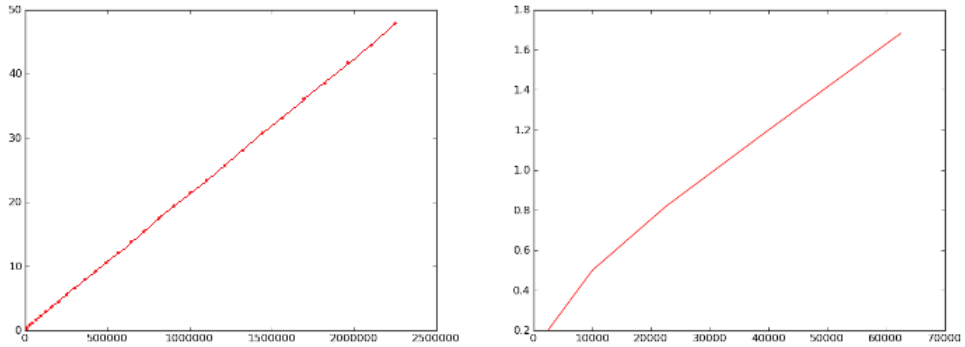

Abbildung 4.4: *$m(h)$-Simulationsergebnis für eine Zelle aus Schicht 1. x-Achse: $h = r^2$, y-Achse: der bei diesem Radius erreichte Maximalwert. Es wurde eine beliebige Einheit aus Schicht 1 gewählt. Rechts kleinerer Ausschnitt, der die Konkavität besonders von kleinen Radien besser zeigt.*

4.1.2.2 Die Maxima von Schicht 2

Das Verhalten der Zellen ändert sich ab Schicht 2 im SFA-Netzwerk deutlich. Für steigende h wird der führende Term der quadratischen Unterformen QT_i und auch der Mischformen mQT_{ij} konvex (Def. siehe (4.9)). Betrachten wir hierzu irgendeine Zelle in Schicht 2, die für den Rest des Kapitels fixiert bleiben soll. Es wird für diese Zelle kein gesonderter Index mitgeführt. Das von der Zelle gesehene Bild $\mathbf{x}$ hat nun die Größe 20×20, was genau 4 rezeptiven Feldern von Schicht 1 entspricht. Man schreibe $\mathbf{x}$ als $\mathbf{x}^T = (\mathbf{x_1}^T, \mathbf{x_2}^T, \mathbf{x_3}^T, \mathbf{x_4}^T)$, die $\mathbf{x_i}$ entsprechen hierbei den Inputvektoren der entsprechenden 4 rezeptiven Felder, also $\mathbf{x_i} \in \mathbf{R}^{100}$. Die gleiche Schreibweise soll für den Output-Vektor von Schicht 1 genutzt werden, $\mathbf{y}^T = (\mathbf{y_1}^T, \mathbf{y_2}^T, \mathbf{y_3}^T, \mathbf{y_4}^T)$, $\mathbf{y_i} \in \mathbf{R}^{30}$. Mit $m_i(h)$ sei das Maximum von QT_i der betrachteten Zelle von Schicht 2 auf der Menge $\sum_{k=1}^{100} x_{i,k}^2 = h$ bezeichnet. Angemerkt sei, dass $\mathbf{y_i}$ natürlich von $\mathbf{x_i}$ abhängt, also eigentlich $\mathbf{y_i}(\mathbf{x_i})$ geschrieben werden müsste.

Ausgeschrieben ist der Funktionswert der $i-$ten quadratischen Unterform QT_i der betrachteten Zelle von Schicht 2,

$$\mathbf{z_i}(\mathbf{x}, h) = \mathbf{y_i}(\sqrt{h}\mathbf{x_i})^T \mathbf{H_{ii}} \mathbf{y_i}(\sqrt{h}\mathbf{x_i}) + \mathbf{f_i}^T \mathbf{y_i}(\sqrt{h}\mathbf{x_i}) + c_i. \tag{4.15}$$

In jeder der 30 Komponenten von $\mathbf{y_i}(\sqrt{h}\mathbf{x_i})$ steht eine quadratische Form in $\mathbf{x_i}$. Für die

vektorielle Funktion $\mathbf{y_i}$ gilt aufgrund der Ergebnisse für Schicht 1 im vorigen Abschnitt,

$$\mathbf{y_i} = h \underbrace{\begin{pmatrix} q_{1,x} \\ \vdots \\ q_{30,x} \end{pmatrix}}_{=:\mathbf{q_x}} + \sqrt{h} \underbrace{\begin{pmatrix} p_{1,x} \\ \vdots \\ p_{30,x} \end{pmatrix}}_{=:\mathbf{p_x}} + \mathbf{r_x}, \tag{4.16}$$

$q_{i,x},\ p_{i,x} \in \mathbf{R},\ \mathbf{r_x} \in \mathbf{R}^{30}$. Die höchste Potenz von h, die somit nach Einsetzen entsteht, stammt aus dem Term,

$$h^2 \left(q_{1,x}, \ldots, q_{30,x}\right) \mathbf{H_{ii}} \begin{pmatrix} q_{1,x} \\ \vdots \\ q_{30,x} \end{pmatrix}. \tag{4.17}$$

Fasst man alle Terme zusammen, ergibt sich für (4.15) bei festem $\mathbf{x}$,

$$\mathbf{z_{i,x}}(h) := \mathbf{z_i}(\mathbf{x}, h) = \alpha_{1,x} h^2 + \alpha_{2,x} h\sqrt{h} + \alpha_{3,x} h + \alpha_{4,x}\sqrt{h} + \alpha_{5,x}, \tag{4.18}$$

für $\alpha_{i,x} \in \mathbf{R}$. Der Exponent bzw. Index $\mathbf{x}$ soll die Abhängigkeit von $\mathbf{x}$ selbst andeuten. Offensichtlich gilt für ein normiertes $\mathbf{x}$,

$$\alpha_{1,x} h^2 + \alpha_{2,x} h\sqrt{h} + \alpha_{3,x} h + \alpha_{4,x}\sqrt{h} + \alpha_{5,x} \leq m_i(h). \tag{4.19}$$

Sei $\mathbf{x}$ nun ein beliebiger normierter Inputvektor. Dann existiert eine positive Zahl κ, so dass $||\mathbf{q_x}||, ||\mathbf{p_x}||, ||\mathbf{r_x}|| \leq \kappa$. Sei $\mathbf{e}$ der EV zum größten EW von $\mathbf{H_{ii}}$. Dann gilt,

$$\begin{aligned} m_i(h) &\leq \left[h\kappa\mathbf{e} + \sqrt{h}\kappa\mathbf{e} + \kappa\mathbf{e}\right]^T \mathbf{H_{ii}} \left[h\kappa\mathbf{e} + \sqrt{h}\kappa\mathbf{e} + \kappa\mathbf{e}\right] + \sqrt{h}\kappa||\mathbf{f_i}||^2 + c_i \\ &= \alpha_1 h^2 + \alpha_2 h\sqrt{h} + \alpha_3 h + \alpha_4\sqrt{h} + \alpha_5, \end{aligned} \tag{4.20}$$

für geeignete $\alpha_i > 0$. Aus (4.19) und (4.20) ergibt sich somit für jedes normierte $\mathbf{x} \in \mathbf{R}^{100}$,

$$\alpha_{1,x} h^2 + \alpha_{2,x} h\sqrt{h} + \alpha_{3,x} h + \alpha_{4,x}\sqrt{h} + \alpha_{5,x} \leq m_i(h) \leq \alpha_1 h^2 + \alpha_2 h\sqrt{h} + \alpha_3 h + \alpha_4\sqrt{h} + \alpha_5. \tag{4.21}$$

Aufgrund der Unkorreliertheit der Ausgänge von Schicht 1 und dem vollen Rang von $\mathbf{H_{ii}}$ kann $\mathbf{x}$ so gewählt werden, dass $\alpha_{1,x} > 0$.

Somit kann man als erstes Resultat festhalten, dass $m_i(h)$ für große h ähnlich wächst

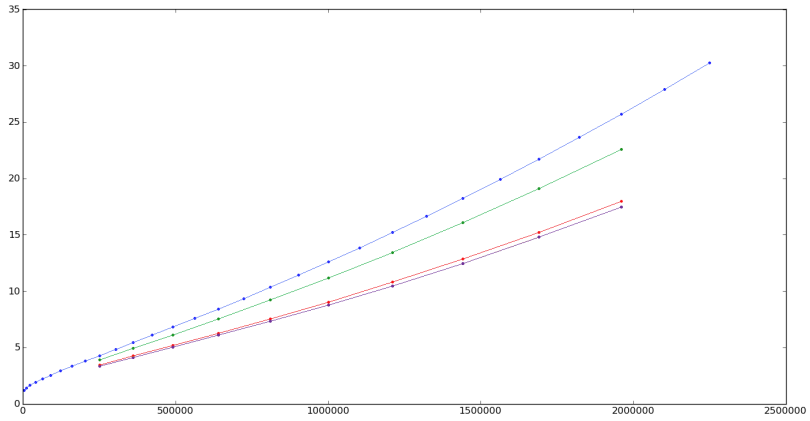

Abbildung 4.5: *$m_i(h)$-Simulationsergebnis. x-Achse: h, y-Achse: $m_i(h)$. Dargestellt sind die Kurven für die 4 quadratischen Unterformen von Zelle 2 aus Schicht 2. Oben: QT_4, Zweite von oben: QT_3, Dritte von oben: QT_1, unten: QT_2.*

wie eine quadratische Funktion. Betrachtet man die begrenzenden Funktionen in (4.21) genauer, so handelt es sich (bei durchweg positiven Koeffizienten) um Funktionen, die bis zu einem bestimmten Wert von h streng konkav sind, dann aber streng konvex werden und bleiben (Die zweite Ableitung beträgt bspw. für die rechts stehende Funktion $2\alpha_1 + \frac{3}{4}\alpha_2\frac{1}{\sqrt{h}} - \alpha_4\frac{1}{4\sqrt{h^3}}$).

Dass man $\mathbf{x}$ in (4.19) so wählen kann, dass z. B. $\alpha_{4,x} > 0$, ist nicht offensichtlich. Simulationen bestätigen dies jedoch für jede betrachtete Zelle. Auf eine detaillierte theoretische Herleitung wird an dieser Stelle jedoch verzichtet, Voraussetzung hierfür wären jedoch die Eigenwertverteilung der Matrizen $\mathbf{H}$ und die SFA-Eigenschaften, insbesondere die Unkorreliertheit.

Aus Obigem ergibt sich also, dass sich die quadratischen Unterformen QT_i bei kleinen h untereinander als konkurrierende konkave, dann aber streng konvexe Unterprobleme verhalten. Das gleiche gilt natürlich auch für die gemischten Formen mQT_{ij}.

Diese Geometrie von $m_i(h)$ wird durch die Simulation bestätigt (Abb. 4.5).

Entsprechend der obigen Theorie und Abschnitt 4.1.1 hieße das zudem, dass es bei Radien, die klein genug sind, vornehmlich zu einer Energieverteilung auf alle vier rezeptiven Felder kommen müsste, bei großen Radien hingegen sollte es eher zur Lokalisierung kommen. Auch dies bestätigt die Simulation (Abb. 4.6).

Es ist also für den Funktionswert ab einer bestimmten Größe von h sinnvoller, nahezu die gesamte Bildenergie in eine quadratische Unterform oder in eine quadratische Misch-

form zu investieren. Deswegen findet man ab Schicht 2 eine Lokalisierung auf ein oder zwei, aber nie auf mehr rezeptive Felder von Schicht 1. In höheren Schichten kommen noch höhere Potenzen von h hinzu und verstärken diesen Effekt. Dass manchmal eine quadratische Mischform präferiert wird, liegt an der Geometrie der Ausgangssignale der unteren Schichten.

Es seien noch zwei Bemerkungen ans Ende dieses Kapitels gestellt. Betrachtet man z.B. Abbildung (4.5), dann wird deutlich, dass die real vorliegenden QT_i und natürlich auch die Mischformen, nicht gänzlich gleich sind wie in der obigen Theorie angewandt.

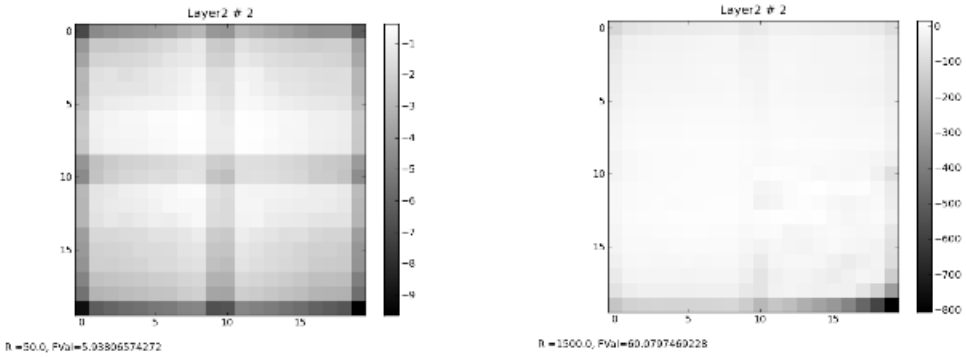

Abbildung 4.6: ***Auf der linken Seite:*** *das Maximum für* $h = 2500$. *Die* ***Normen*** *der einzelnen rezeptiven Felder sind:* $I = 20.75$, $II = 20.65$, $III = 28.24$, $IV = 29.08$. *Der* ***Funktionswert*** *teilt sich wiefolgt auf die quadratischen Unterformen auf:* $QT_{11} = 0.97$, $QT_{22} = 0.96$, $QT_{33} = 0.99$, $QT_{44} = 1.06$, $mQT_{12} = 0.47$, $mQT_{13} = 0.34$, $mQT_{14} = 0.10$, $mQT_{23} = 0.13$, $mQT_{24} = 0.31$, $mQT_{34} = 0.60$. *Man sieht, sowohl Norm als auch Funktionswert sind relativ gleichmäßig auf alle vier rezeptiven Felder verteilt.*
Auf der rechten Seite: *das Maximum für die gleiche Zelle für* $h = 2.25e06$. *Die* ***Normen*** *der einzelnen rezeptiven Felder sind:* $I = 264.31$, $II = 235.73$, $III = 451.11$, $IV = 1386.00$. *Der* ***Funktionswert*** *teilt sich wiefolgt auf die quadratischen Unterformen auf:* $QT_{11} = 1.90$, $QT_{22} = 1.71$, $QT_{33} = 3.27$, $QT_{44} = 35.18$, $mQT_{12} = 0.49$, $mQT_{13} = 1.20$, $mQT_{14} = 4.09$, $mQT_{23} = 0.91$, $mQT_{24} = 4.37$, $mQT_{34} = 6.90$. *Man sieht, sowohl Norm als auch Funktionswert sind im unteren rechten rezeptiven Feld konzentriert.*

So kommt es auch nicht zu einer ausschließlichen Verteilung auf genau ein rezeptives bzw. zwei rezeptive Felder von Schicht 1, sondern zu einer nur hauptsächlichen Verlagerung der Bildenergie. Zudem ist selbstverständlich nicht jedes Sub-Maximum einer quadratischen Unterform oder gemischten Form ein lokales Maximum der gesamten Zelle.

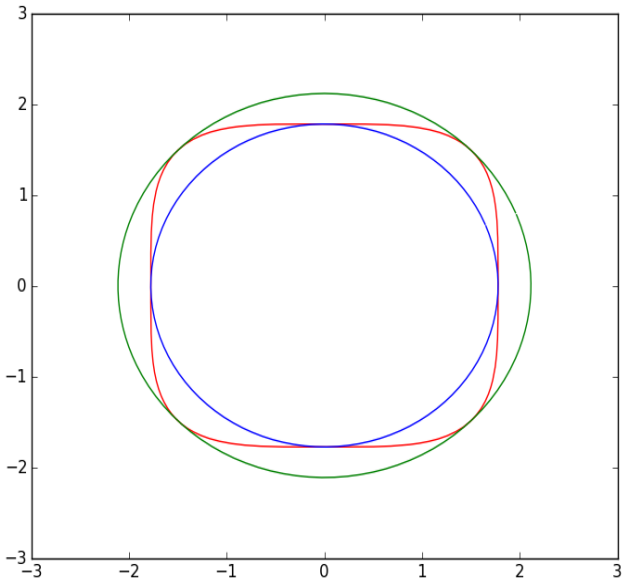

Abbildung 4.7: *Innen:* $\sum x_i^2 = \sqrt{10}$, *mittig:* $\sum x_i^4 = 10$, *außen:* $\sum x_i^2 = \sqrt{2 \cdot 10}$.

In den Optima von Schicht 2 (Abb. A.2) fällt auf, dass zumindest die Maxima von Zelle 1 und 2 gleichmäßig auf das gesamte rezeptive Feld verteilt sind. Das liegt an der quadratischen Form der beiden Zellen, die einen normmäßig überlegenen linearen Anteil ($\mathbf{f}^T\mathbf{x} \gg \mathbf{x}^T\mathbf{H}\mathbf{x}$) aufweisen.

4.2 Andere Nebenbedingungen

Dem Lokalisierungsverhalten intrinsisch war die zugrunde liegende Norm, also die Menge der zur Verfügung stehenden Bilder, im obigen Fall also alle Elemente auf einer Kugel eines bestimmten Radius'. Das Verhalten der Optima ändert sich drastisch, wenn diese verändert wird.

Betrachtet man obige Rechnungen, so könnte eine erste Änderung darin bestehen, die 4−Norm zu verwenden, also,

$$\sum_{i=1}^{dim} x_i^4 = h. \tag{4.22}$$

Diese Randbedingung entspricht einer ausgebeulten Kugel wie in Abbildung (4.7) dargestellt. Daraus folgt insbesondere, dass

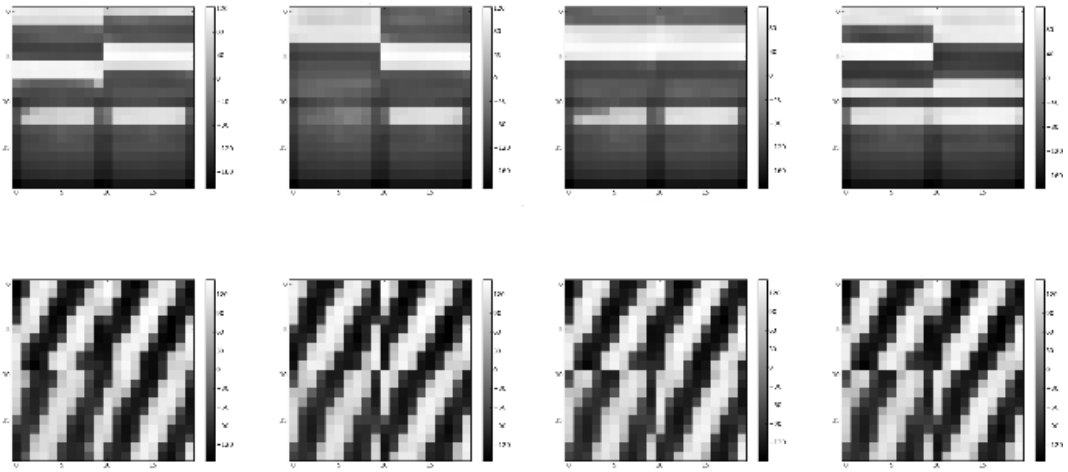

Abbildung 4.8: *Maxima unter der Nebenbedingung* $\sum x_i^4 = const$ *für zwei verschiedene Zellen in Schicht 2. Horizontal angeordnet sind verschiedene Durchläufe für die gleiche Zelle.*

$$m_K(\sqrt{h}) \leq m_{4K}(h) \leq m_K(\sqrt{dim \cdot h}), \tag{4.23}$$

mit

$$m_{nK}(h) := \max\left\{\frac{1}{2}y_i H y_i + f y_i + c : \sum [x_{i,j}]^n = h\right\}. \tag{4.24}$$

Mit m_k sei das Maximum einer bestimmten Zelle bezüglich Kugelnebenbedingung bezeichnet, mit *dim* die Dimensionalität des zugrunde liegenden Problems. Mit den Formeln für die Kugelrandbedingung ergibt dies z.B. für Schicht 2 vereinfacht geschrieben zusammen mit (4.23),

$$\begin{aligned} \alpha_1 h + \alpha_2\sqrt{h\sqrt{h}} + \alpha_3\sqrt{h} + \alpha_4\sqrt{\sqrt{h}} + \alpha_5 &\leq m_{4K}(h) \\ &\leq \alpha_6 dim \cdot h + \alpha_7\sqrt{dim \cdot h\sqrt{dim \cdot h}} \\ &\quad + \alpha_8\sqrt{dim \cdot h} + \alpha_9\sqrt{\sqrt{dim \cdot h}} + \alpha_{10} \end{aligned} \tag{4.25}$$

Die Größe $m_{4K}(h)$ ist also zwischen zwei streng konkaven Funktionen eingeschlossen und selbst natürlich streng monoton wachsend. Es sollte also keine Lokalisierung mehr auftreten, diese sollte erst wieder ab Schicht 3 vorhanden sein. Auch dies wird durch die Simulation bestätigt, in Abbildung 4.8 sind Maxima für verschiedene Zellen von Schicht 2 unter der Bedingung $\sum x_i^4 = const$ dargestellt. Es tritt keine Lokalisierung mehr auf.

Auch die theoretisch motivierte Geometrie der $m_{4K}(h)$ wird durch die Simulation bestä-

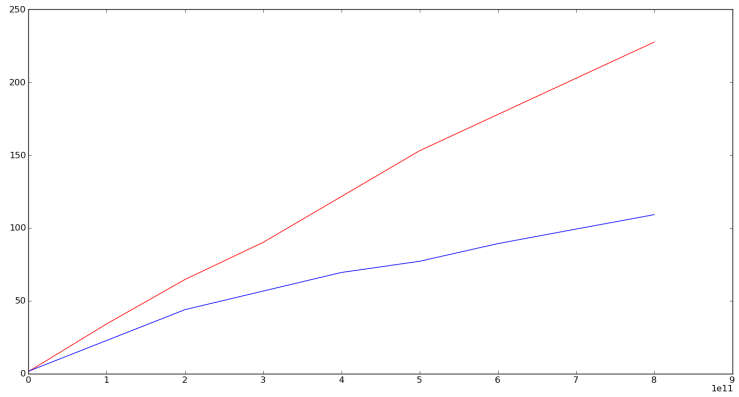

Abbildung 4.9: $m_{4K}(h)$-*Simulationsergebnis, also unter der Nebenbedingung* $\sum x_i^4 = const.$ *x-Achse: h, y-Achse:* $m_{4K}(h)$*. Dargestellt sind die Kurven für die 2 quadratischen Unterformen von Zelle 2 aus Schicht 2. Untere Kurve:* QT_{11}*, obere Kurve:* QT_{44}*.*

tigt (Abb. 4.9). Wie schon erwähnt, kommt es ab Schicht 3 unter dieser Nebenbedingung jedoch wieder zur Lokalisierung.

Ein weiterer sinvoller Ansatz für alle Schichten wäre aus diesen Überlegungen heraus die Unendlichnorm, also die Würfelnebenbedingung. Wie aus Abb. A.5 ersichtlich, wird entsprechend der vorher entwickelten Theorie somit das Lokalisierungsproblem für alle Schichten behoben.

Abschließend sei gesagt, dass neben den auftretenden mathematischen Phänomenen die Optima natürlich auch hinsichtlich der Frage nach biologischer Interpretierbarkeit betrachtbar sind. Wie eingangs erwähnt, wäre eine Nutzbarkeit der Optima höherer Schichten im SFA-Netz zur Klassifikation von Neuronen höherer Schichten im visuellen System natürlich sehr wünschenswert und soll nun im Rahmen der Abschlussdiskussion im letzten Kapitel besprochen werden.

Kapitel 5

Diskussion und Zusammenfassung

Diese Arbeit hat die maximal inhibitorischen und exzitatorischen Stimuli von Zellen eines hierarchischen, mit natürlichen Bildern trainierten SFA-Netzwerkes systematisch für alle vier Schichten unter verschiedenen Nebenbedingungen untersucht. Es konnten die bekannten Ergebnisse hinsichtlich der Optima für Schicht 1 aus den Arbeiten von Berkes and Wiskott (2005) sowie der MDP-Bibliothek (*http://mdp-toolkit.sourceforge.net/*) reproduziert werden. Unter der sonst verwendeten Kugelnebenbedingung zeigte sich in den Optima ab Schicht 2 des SFA-Netzwerkes ein Lokalisierungsphänomen, bei welchem nahezu die gesamte Bildenergie auf stets ein oder zwei rezeptive Felder von Schicht 1 verteilt wird (siehe Abb. 1.1 und Bildergalerie im Anhang). Diese Lokalisierung liegt am Zusammenspiel von Netzwerkarchitektur, Kugelnebenbedingung bzw. quadratischer Norm und Polynomeigenschaft der SFA-Zellen, die ab Schicht 2 Polynomen von Grad 4 entsprechen. Aufgrund der Netzwerkarchitektur lässt sich das globale Optimierungsproblem als Maximierung oder Minimierung konkurrierender Unterprobleme schreiben, dessen Verhalten sich von Schicht 1 zu Schicht 2 und dann zu allen höheren Schichten hin grundlegend verändert. Die Abhängigkeit 'maximal erreichbarer Funktionswert in Abhängigkeit von zur Verfügung stehender Bildenergie' produziert aufgrund der Kugelnebenbedingung in Schicht 1 eine Optimierung konkurrierender konkaver Unterprobleme, ab Schicht 2 eine Optimierung konkurrierender konvexer Unterprobleme (ab einem bestimmten Kugelradius), die die Resultate erklären. Entsprechend der entwickelten Theorie und den durchgeführten Simulationen tritt dieses Phänomen bei Nebenbedingungen höherer Ordnung ($n-$Norm, Supremumsnorm bzw. Würfelnebenbedingung) nicht mehr auf (siehe Abb. 4.8 und Bildergalerie im Anhang).

Die Interpretation der Ergebnisse ist nun schwierig, da es keine ausreichenden Referenzen

aus dem physiologischen Experiment gibt, die man zur Bewertung heranziehen könnte. Zudem kann man sich die Frage stellen, in wie weit das Konzept der optimalen Stimuli auf höheren Schichten sinnvoll ist, wo es sich doch um Zellen zu handeln scheint, die große Invarianzen aufweisen, deren Antwortverhalten nur im Zusammenspiel und Muster mit anderen Zellen verwertbar sein könnte und sich nicht oder nur selten im Bereich von maximalen oder minimalen Stimuli abzuspielen braucht, um komplexe Funktionen zu erfüllen (Gross 2008).

Zellen höherer Schichten der visuellen Bildverarbeitung weisen große Invarianzen auf und sind essenziell für höhere visuelle Funktionen wie Muster- und Objekterkennung (Gross 2008). Ob sich die Funktion dieser Neurone in den Bereichen maximaler Exhibition oder Inhibition abspielt oder vielmehr in Zwischenbereichen und im Zusammenspiel mit anderen Neuronen und ob das Konzept optimaler Stimuli für höhere Schichten sinnvoll ist oder nicht, wird diese Arbeit nicht endgültig klären können. Jedoch sind in der Literatur eine Vielzahl von Versuchen beschrieben, maximale (oder minimale) Reaktionen von Zellen oberhalb von V1 über verschiedene Stimulisets zu evozieren. Mit SFA bzw. durch SFA-Netzwerke hat man Resulate des physiologischen Experimentes für den primären visuellen Kortex sehr gut reproduzieren können und hat -im Gegensatz zum Experiment- die Möglichkeit, optimale Stimuli für beliebige Schichten aus den Zellen selbst zu generieren. Die Frage, wie die optimalen Stimuli für Zellen nach mehreren SFA-Schritten aussehen, stellt sich somit aus den Ergebnissen und den Bemühungen der physiologischen Experimente auf natürliche Weise.

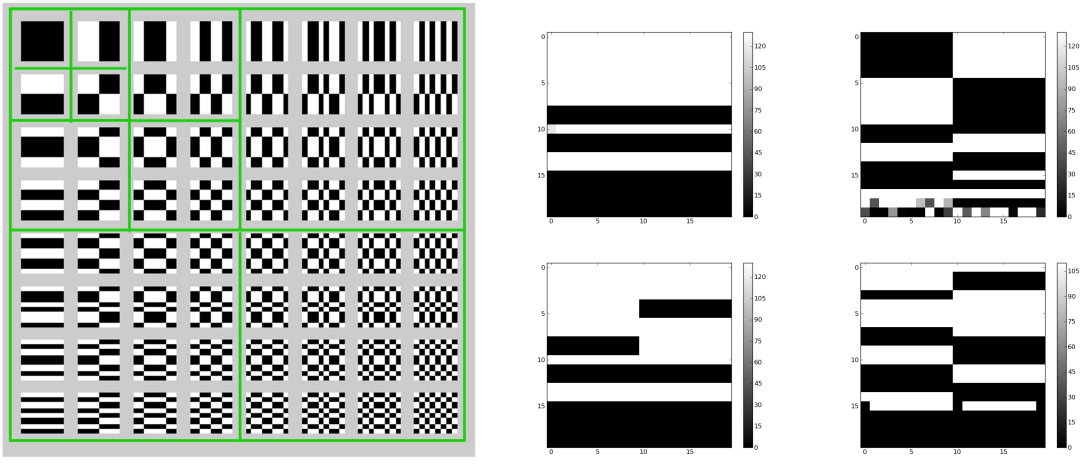

Abbildung 5.1: *Auf der* ***linken*** *Seite sind die Walsh-Hadamard-Basen für* 8×8 *Pixel messende Bilder dargestellt. Auf der* ***rechten*** *Seite exemplarische Maxima unter Würfelnebenbedingung von Schicht 2.*

Wendet man sich den optimalen Stimuli höherer Schichten in SFA-Netzwerken zu, so zeigt diese Arbeit, dass als Nebenbedingung für die Optimierung die quadratische Norm (Kugelnebenbedingung) nicht geeignet ist. Die unter Kugelnebenbedingung gefundenen Optima ab Schicht 2 zeigen alle die gleiche Struktur mit Lokalisierung der Bildenergie auf ein oder maximal zwei rezpetive Felder von Schicht 1, was, wie schon mehrfach angesprochen, nur der gewählten Nebenbedingung, der Netzwerkarchitektur und der Polynomeigenschaft der SFA-Zellen Rechnung trägt. Eine höhergradige, an das Netzwerk angepasste Norm oder gar die Supremumsnorm sind nicht weniger allgemeine Nebenbedingungen als die quadratische Norm, vermeiden aber dieses Phänomen und zeigen so in den Optima individuelle, strukturelle Eigenschaften der Zellen auf dem gesamten entsprechenden rezeptiven Feld. Möchte man die Supremumsnorm (Würfelnebenbedingung) vermeiden, so müsste man -bliebe man bei den $n-$Normen- eine Norm wählen, deren Grad mindestens so hoch ist wie der Grad der Polynome der zu optimierenden Schicht (also für 2 SFA-Schritte mindestens die $4-$Norm, für 3 Schritte die $8-$Norm etc.).

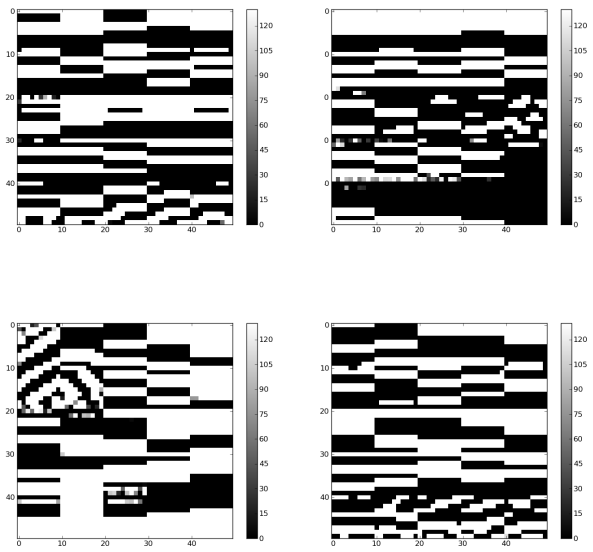

Abbildung 5.2: *Dargestellt sind Maxima einzelner Zellen in Schicht 3 unter Würfelnebenbedingung. Es findet sich auch hier gehäuft ein Muster mit horizontal versetzten Balken unterschiedlicher Breite, welches sich rein qualitativ auch bei der Hadamard-Basis findet (Abb. 5.1).*

Zum Abschluss sollen noch einmal die Optima unter Würfelnebenbedingung genauer betrachtet werden. Wie schon in der Einleitung erwähnt, wird hierfür besonders eine Arbeit aus dem Jahr 1987 von Richmond et al. (1987) hervor gehoben. Die Autoren

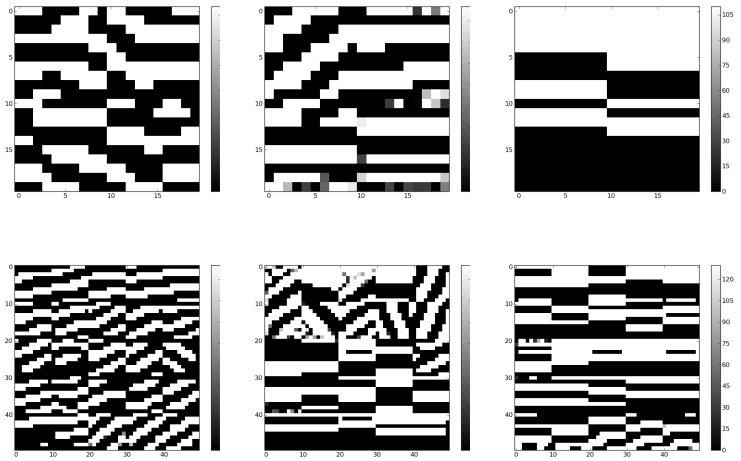

Abbildung 5.3: *Dargestellt sind die im Text genannten Elemente der Gratings und der horizontal versetzten Balken wie bei der Hadamard-Basis.* ***Oben von links nach rechts*** *sind verschiedene Maxima von Zellen aus Schicht 2 unter Würfelnebenbedingung dargestellt, in denen die beiden genannten Bildelemente unterschiedlich stark miteinander kombiniert sind. Links sieht man fast ausschließlich ein Grating, in der Mitte eine Kombination aus Grating und Balken und rechts fast ausschließlich horizontal versetzte Balken unterschiedlicher Dicke.* ***Unten von links nach rechts*** *ist das gleiche für Zellen aus Schicht 3 dargestellt.*

analysierten in dieser Arbeit das Antwortverhalten von Neuronen des IT bei Affen auf ein Set an Stimuli (schwarz-weiß, Auflösung 8×8 Pixel), welches aus einer vollständigen Walsh-Hadamard-Basis plus nochmals 64 Bildern mit umgekehrtem Kontrast (siehe z.B. http://en.wikipedia.org/wiki/Hadamard_transform) bestand. Die 64 verwendeten Bilder repräsentieren eine Orthonormalbasis auf dem $\mathbf{R}^{8\times8}$. Als Resultat fand man, dass die IT-Neurone sehr differenziert und Stimulus-abhängig auf die Testbilder reagierten. Die Autoren zogen letztlich aus den Experimenten den Schluss, dass die verwendeten Stimuli zur Testung und Klassifizierung von Neuronen im IT geeignet sind.

In unseren Simulationen finden sich unter Würfelnebenbedingung Optima, die qualitativ der in der genannten Publikation verwendeten Hadamard-Basis ähneln. In Abbildung 5.1 ist dies anhand von Schicht 2 dargestellt. Ähnliche Elemente finden sich auch in Maxima und Minima höherer Schichten wie für Schicht 3 in Abbildung 5.2 dargestellt. In der Tat fällt auf, dass sich die Optima höherer Schichten meist aus diagonalen Gratings einer bestimmten Frequenz (z.T. auch über Ecken und mit kurvigem Verlauf) und den genannten horizontal versetzten Balken wie bei der Hadamard-Basis zusammen setzen. Insgesamt werden die Muster mit höherer Schicht komplexer, die beiden Elemente finden sich jedoch stets wieder, z.T. mehr oder minder stark miteinander kombiniert wie in

Abbildung 5.3 dargestellt.

In wie weit sich nun die genannten Stimuli eignen, um weitere genauere Informationen über Neurone höherer Schichten des visuellen Kortex zu erlangen, muss das praktische Experiment zeigen, ebenso wie die Klärung der Frage, ob das Konzept der optimalen Stimuli für höhere Schichten weiterhin tragbar und weiterführend ist.

Anhang A

Bildergalerie

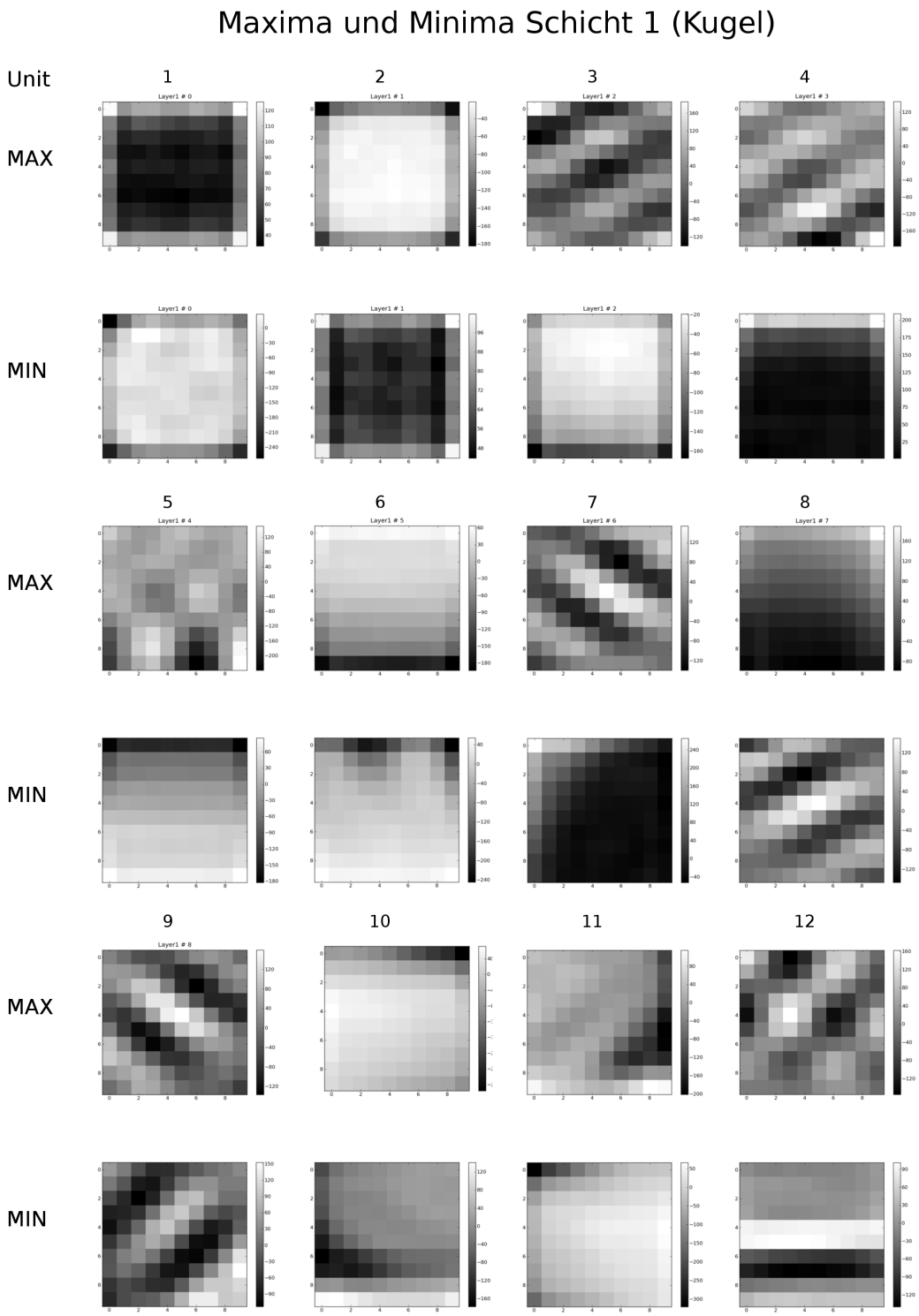

Abbildung A.1: *Maxima und Minima von Schicht 1 unter Kugelnebenbedingung ($r = 640$).*

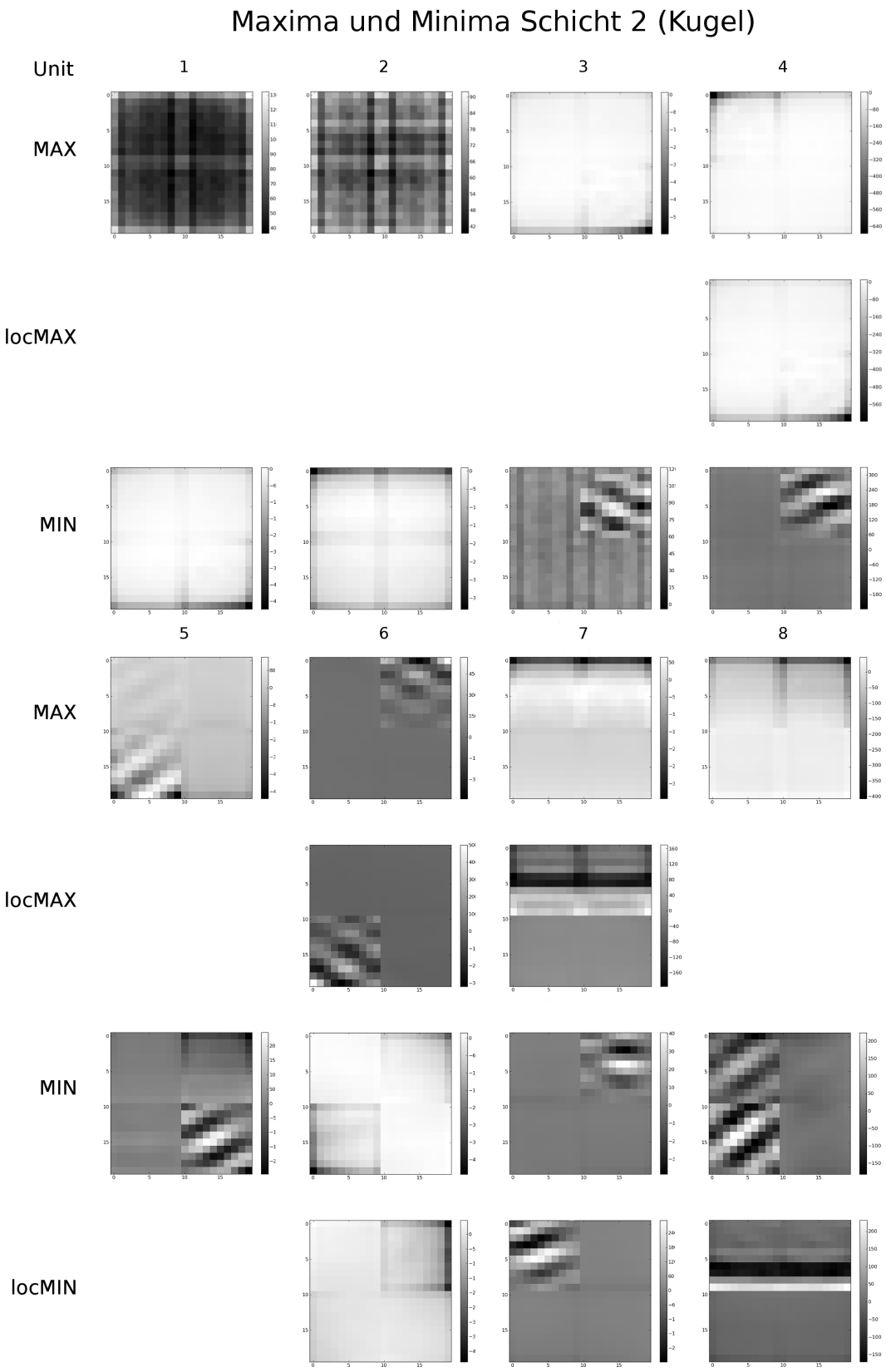

Abbildung A.2: *Maxima und Minima von Schicht 2 unter Kugelnebenbedingung sowie lokale Maxima und Minima (*locMIN, locMAX*).* ($r = 1280$).

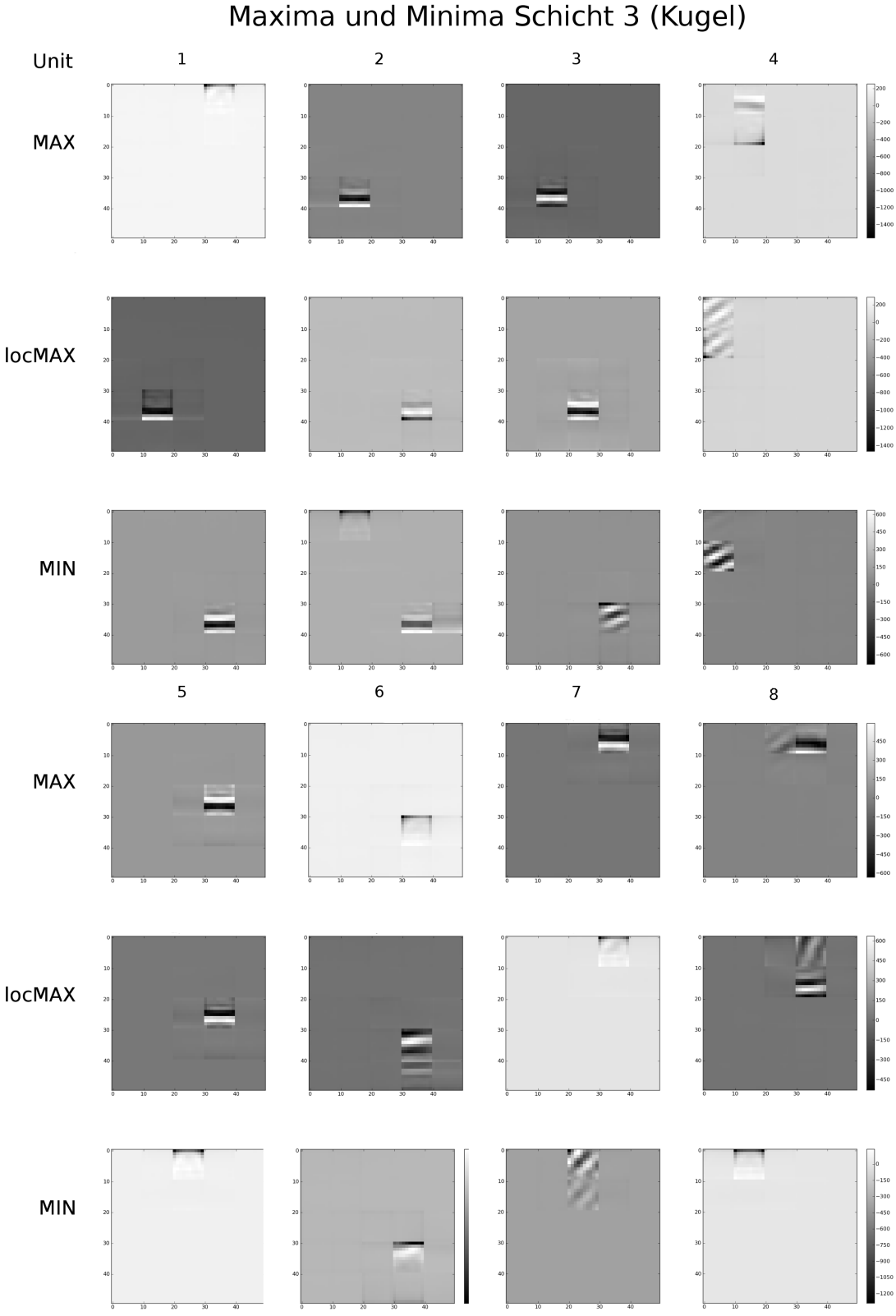

Abbildung A.3: *Maxima und Minima von Schicht 3 unter Kugelnebenbedingung sowie lokale Maxima und Minima (*locMIN, locMAX*) ($r = 3200$).*

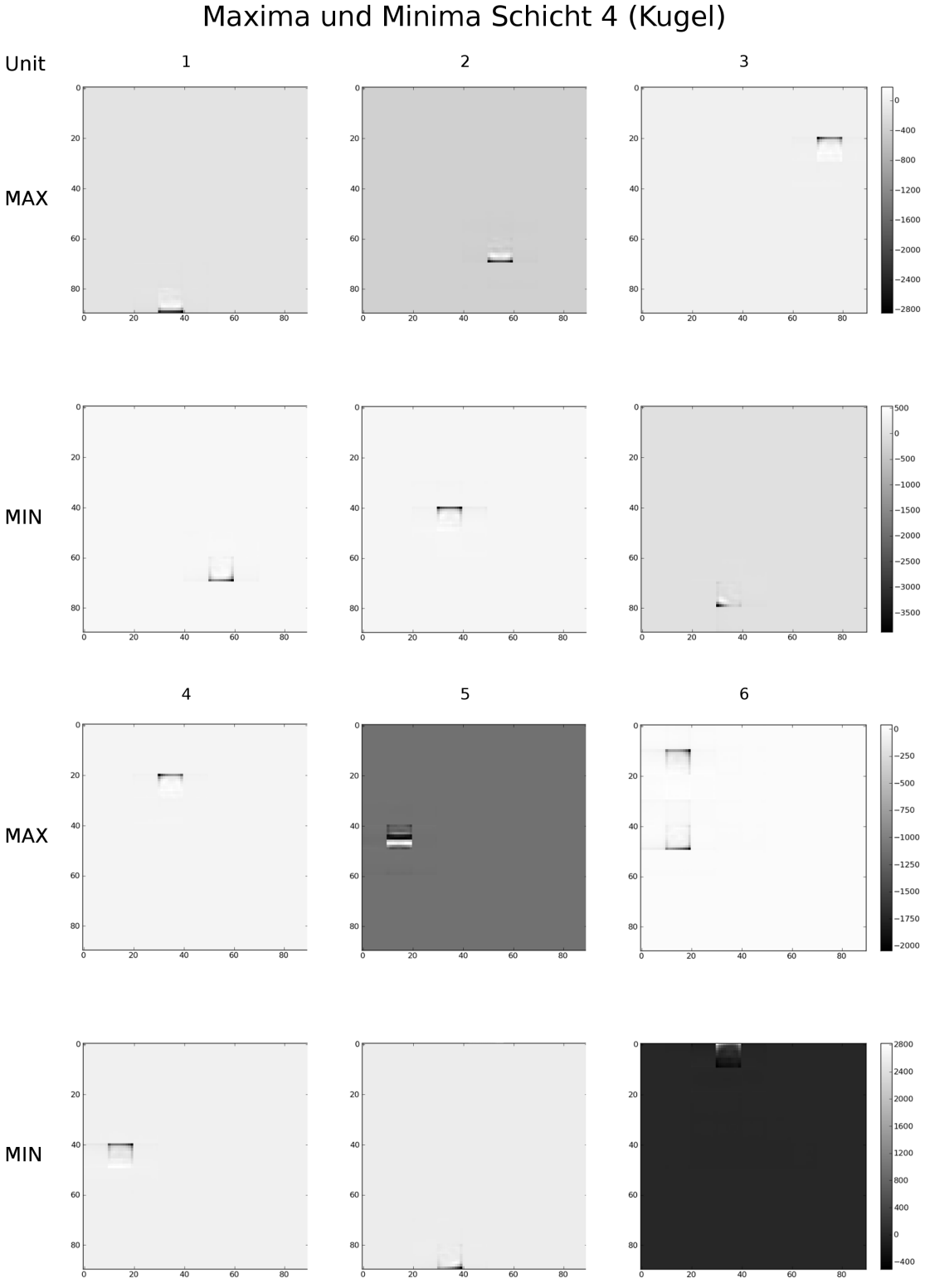

Abbildung A.4: *Maxima und Minima von Schicht 4 unter Kugelnebenbedingung ($r = 5760$).*

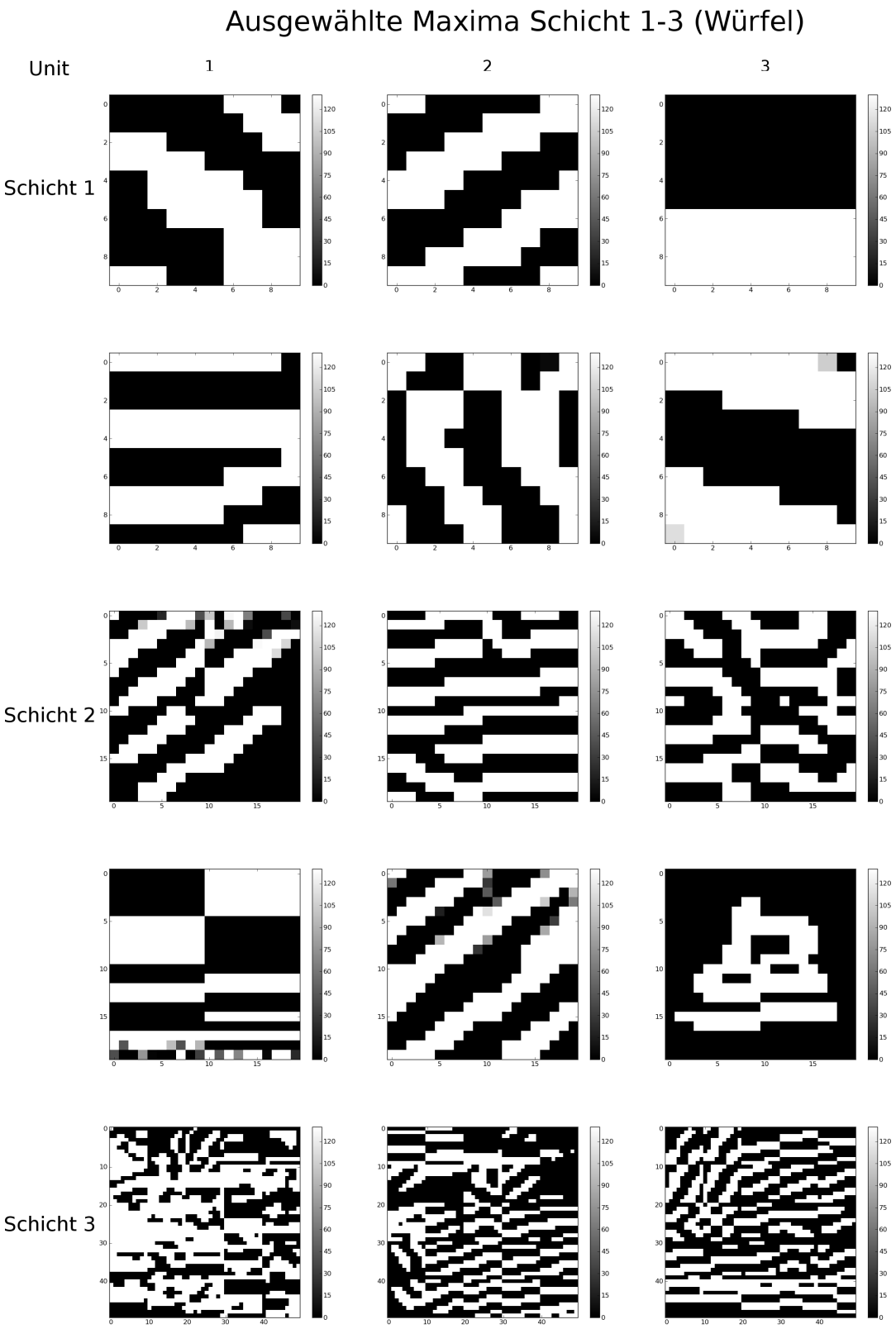

Abbildung A.5: *Ausgewählte Maxima von Schicht 1-3 unter Würfelnebenbedingung* ($a = 130$).

Literaturverzeichnis

E. H. Adelson and J. R. Bergen. Spatiotemporal energy models for the perception of motion. *Journal Optical Society of America*, 2:284–299, 1985.

J. S. Baizer. Receptive field properties of v3 neurons in monkey. *Investigative Ophthalmology & Vision Science*, 23:87–95, 1982.

P. Berkes and L. Wiskott. Slow feature analysis yields a rich repertoire of complex cell properties. *Journal of Vision*, 5(6):579–602, July 2005. `http://journalofvision.org/5/6/9/`, doi:10.1167/5.6.9.

C. Cadieu, M. Kouh, A. Pasupathy, C. E. Connor, M. Riesenhuber, and T. Poggio. A model of v4 shape selectivity and invariance. *Journal of Neurophysiology*, 98:1733–1750, 2007.

S. David, B. Hayden, and J. Gallant. Spectral receptive field properties explain shape selectivity in area v4. *J Neurophysiol*, 96:3492–3505, 2006.

R. Desimone. Face-selective cells in the temporal cortex of monkeys. *Journal of Cognitive Neuroscience*, 3:1–8, 1991.

R. Fletcher. *Practical Methods of Optimization*. John Wiley & Sons, 2000.

M. Franzius, H. Sprekeler, and L. Wiskott. Slowness and sparseness lead to place, head-direction, and spatial-view cells. *PLOS Computational Biology*, 3(8), 2007.

K. R. Gegenfurtner, D. C. Kiper, and J. B. Levitt. Functional properties of neurons in macaque area v3. *Journal of Neurophysiology*, 77:1906–1923, 1997.

C. G. Gross. Single neuron studies of inferior temporal cortex. *Neuropsychologia*, 46(3): 841–52, 2008.

D. Hubel and T. Wiesel. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *Journal of Physiology*, 160:106–154, 1962.

J. P. Jones and L. A. Palmer. An evaluation of the two-dimensional gabor filter model of simple recep-tive fields in cat striate cortex. *Journal of Neurophysiology*, 58:1233–1257, 1987.

A. Plebe. A model of the response of visual area v2 to combinations of orientations. *Computation in Neural Systems*, 23(3):105–122, 2012.

D. Pollen and S. Ronner. Phase relationship be-tween adjacent simple cells in the visual cortex. *Science*, 212:1409–1411, 1981.

D. Pollen, M. Nagler, J. Daugman, R. Kronauer, and P. Cavanagh. Use of gabor elementary functions to probe receptive field substructure of posterior inferotemporal neurons in the owl monkey. *Vision research*, 24(3):233–41, 1984.

R. Quiroga, L. Reddy, G. Kreiman, C. Koch, and I. Fried. Invariant visual representation by single neurons in the human brain. *Nature*, 435:1102–1107, 2005.

B. J. Richmond, L. M. Optican, and H. Spitzer. Temporal encoding of two-dimensional patterns by single units in inferior temporal cortex. i. *Journal of Neurophysiology*, 57: 132–146, 1987.

A. T. Smith, K. D. Singh, A. L. Williams, and M. W. Greenlee. Estimating receptive field size from fmri data in human striate and extrastriate visual cortex. *Cereb. Cortex*, 11 (12):1182–1190, 2001.

L. Wiskott and T. Sejnowski. Slow feature analysis: Unsupervised learning of invariances. *Neural Computation*, 14(4):715–770, 2002.