Künstliche Neuronale Netzwerke zur Vorhersage der Hirnkontur

Jens Rittscher, Jens Hiltner und Claudio Moraga

Universität Dortmund – Lehrstuhl Informatik I
Otto-Hahn-Straße 16, 44221 Dortmund
Email: [hiltner|moraga]@ls1.cs.uni-dortmund.de

Zusammenfassung. Die innerhalb der medizinische Bildverarbeitung zu analysierenden Muster erfordern in der Regel die Anwendung von sogenannten "High-Level"-Bildverarbeitungsoperatoren. Um trotzdem eine schnelle Verfügbarkeit gewährleisten zu können, muß das Operationsgebiet ("Region Of Interest", ROI) der höherwertigen Bildverarbeitungsalgorithmen geeignet eingeschränkt werden. Der Vorteil dieser Vorgehensweise liegt darin, daß innerhalb des gefundenen Suchbereiches deutlich komplexere (zeitaufwendigere) Methoden eine anschließende Segmentierungs- und Klassifikationsaufgabe auf einem stark reduzierten Datenmaterial durchführen können. Innerhalb dieses Beitrags wird sowohl der theoretische als auch der praktische Ansatz eines neuen –auf Neuronalen Netzwerken basierenden– Konzeptes zur Vorhersage der Hirnkontur auf MRT-Kopfdatensätzen beschrieben. Schließlich werden anhand von Beispielen die Ergebnisse der Konturberechnung präsentiert.

Schlüsselwörter: Medizinische Bildverarbeitung, Neuronale Netzwerke, Bildsegmentierung, Datenreduktion, Region-of-Interest-Bestimmung

1 Einleitung

Grundsätzlich ist die medizinische Bildverarbeitung, wie sich in den vergangenen Jahren gezeigt hat, nicht trivial und die Erfüllung der hier gestellten Anforderungen nicht leicht. Zum einen werden die Ergebnisse sehr schnell benötigt, zum anderen müssen sie sehr genau sein. Eines der wichtigsten ungelösten Probleme in der Bildvorverarbeitung stellt die *Bildsegmentierung* dar, wobei es das Ziel der Bildsegmentierung ist, das Bild in sinnvolle Teilbereiche zu partitionieren, die im Idealfall den realen Objekten im Bild entsprechen. Bildsegmentierung ist somit der Schlüsselschritt zur eigentlichen *Bilderkennung*. Das Problem in der medizinischen Bildverarbeitung ist hierbei die große Variabilität in Form, Lage und Grauwertverhalten des vorliegenden Bildmaterials, die die Anwendung einfacher und damit schneller Algorithmen aufgrund fehlender Genauigkeit verhindert. Umgekehrt sind komplexe Algorithmen, wie sie durch die Anwendung von Methoden aus der *Computational Intelligence* entstehen, zeitaufwendig. Will oder kann man dennoch nicht auf die Anwendung dieser Algorithmen verzichten, muß eine zusätzliche Verarbeitungsstufe eingeführt werden, in welcher die *Region of Interest* (in folgenden kurz *ROI*) für komplexere Algorithmen ermittelt wird.

Im allgemeinen wird der Segmentierungsprozeß dadurch schneller bzw. durch die möglich gewordene Verwendung komplexer Algorithmen genauer. Wir haben im Rahmen unserer Forschung die Anwendbarkeit dieser Technik auf die Segmentierung des Gehirns auf MRT-Datenbildern hin untersucht und verschiedene Neuronale Netze zur Lösung dieser Aufgabe hin trainiert. Im folgenden werden die Begriffe *Neuronales Netzwerk*, *Künstliches Neuronales Netzwerk* oder einfach nur *Netz* synonym verwendet.

2 Grundlagen Neuronaler Netze

Neuronale Netze sind informationsverarbeitende Systeme, die aus einer Vielzahl von miteinander verschalteten Neuronen bestehen. Dabei handelt es sich –bei mathematische Betrachtungsweise– um ein Berechenbarkeitsmodell (d.h. jede berechenbare Funktion kann mit Hilfe eines Neuronalen Netzes berechnet werden). Der Unterschied zu herkömmlichen Modellen besteht darin, daß zur Lösung einer Aufgabe nicht der Lösungsweg explizit vorgegeben werden muß, sondern das Netz im Rahmen einer Trainingsphase die Aufgabenlösung durch Verallgemeinerung des Lernproblems *erlernt*.

Ein Neuronales Netz wird im Modell als gerichteter Graph mit einer Zustandsmenge Z für jeden Knoten (Neuron), einer Zustandsmenge W mit Gewichten für jede Verbindung zwischen Neuronen und je einer Menge von Ein- und Ausgaben verstanden. Die Zustandsmenge σ eines Netzes mit k Knoten und v Verbindungen ist ein $\Sigma = Z^k \times W^v$, wobei ein einzelner Netzzustand durch das Tupel $\sigma = (z, w)$ mit $z \in Z^k$ und $w \in W^k \times W^k$ beschrieben wird. Gruppen von Neuronen werden in der Literatur häufig gemäß ihrer Funktion in Schichten (z.B. Eingangsschicht, Ausgangsschicht, verdeckte Schichten) zusammengefaßt. Abbildung 1 zeigt ein einfaches Beispiel eines Neuronalen Netzes.

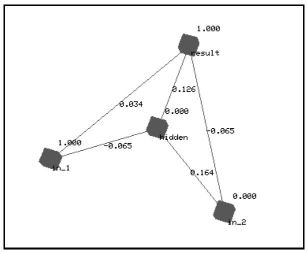


Abb. 1. Ein einfaches Neuronales Netz zur Realisierung der XOR-Funktion, bestehend aus einer Eingabe-, einer Ausgabe- und einer verdeckten Schicht

Das Verhalten eines Neuronalen Netzes wird einerseits durch seine Kurzzeitdynamik (das Verhalten der Neuronen selbst) und andererseits durch die Änderungen in der Gewichtsmatrix (Konnektivitätsmatrix) W in der Lernphase bestimmt, der Langzeitdynamik. Gleichung 1 zeigt die Veränderung des Gewichtes einer Neuronenverbindung mittels einer Lernfunktion g.

$$w_{ij}^{t+1} = w_{ij}^t + g_{ij}(w_{ij}^t, z_i^t, z_j^t) \tag{1}$$

Ziel des Lernvorgangs ist es, das Netz durch Justieren der Neuronenverbindungsgewichte so einzustellen, daß bei einem bestimmten an der Eingabeschicht angelegten Muster ein gewünschtes Ausgabenmuster berechnet wird. Die Abweichung zwischen Soll- und Istausgabe wird in der Praxis mit Hilfe einer mathematischen Fehlerfunktion gemessen, wobei das Lernen dann als Minimierung der Fehlerfunktion verstanden wird.

Das Problem für mehrschichtige Netzwerke ist, daß keine gewünschten Sollausgaben der Lerneingaben für die Neuronen in den verdeckten Schichten zur Verfügung stehen. Aus diesem Grund muß der an den Ausgabeneuronen beobachtbare Fehler E an die Neuronen in den verdeckten Schichten zurückpropagiert werden. Wendet man das Gradientenabstiegsverfahren auf die quadratische Fehlerfunktion an, so ergibt sich eine Gewichtsänderung w_{ij} für die Elemente der Gewichtsmatrix W aus $\Delta w_{ij} = \mu \delta_i o_j$, mit

$$\delta_i = \begin{cases} \sum\limits_n (y_i^n - o_i(W, (x_i^n))) \cdot f'(c_i) & \text{falls } i \text{ Ausgangsneuron ist} \\ \sum\limits_{k \in S(i)} \delta_k w_{ki} \cdot f'(c_i) & \text{sonst} \end{cases} \tag{2}$$

Das ist die bekannte *Backpropagation-Regel*. Innerhalb unserer Arbeit haben wir noch mit den Lernalgorithmen *Resilient Propagation* und *Quickpropagation* gearbeitet, da sie bzgl. ihres Konvergenzverhaltens Verbesserungen darstellen. Detaillierte Informationen dazu können etwa in [1] nachgelesen werden.

3 Ermittlung der ROI mittels Neuronaler Netze

Die ROI beinhaltet den Teil des Bildes, der für eine weitere Bearbeitung wichtig ist. Dabei müssen mindestens alle Punkte des gesuchten Objektes O in der ROI enthalten sein (d.h. $O \subseteq ROI$). Für die Repräsentation des Bildes im Netz bieten sich grundsätzlich zwei Möglichkeiten an. Entweder werden die Bildpunkte über ihre Grauwerte im Netz repräsentiert oder es werden nur Objektkonturen betrachtet. Wir haben uns für die zweite Möglichkeit entschieden, da Netze der Größe 256×256 recht komplex sind und außerdem eine statistisch ausreichende Menge an Daten nicht zur Verfügung stand (hierfür wären mehrere hundert Datensätze notwendig). Uns interessierte zudem auch noch die Fragestellung, ob die ROI im Sinne einer schnellen Vorverarbeitung nur aufgrund der vorliegenden Kopfform gefunden werden kann. Erste Netztypen werden u.a. in [2, 3] behandelt. Die Aufgabe unserer Netze bestand darin, zu einer bestehenden Eingabekontur eine passende Ausgabekontur zu generieren. Experimentell haben wir die besten Ergebnisse mit 29 Konturpunkten erzielt. Die getesteten vorwärtsgerichteten Netze bestanden somit aus mindestens $2 \cdot 29$ Neuronen in der Eingangsschicht und $2 \cdot 29$ Neuronen in der Ausgangsschicht, d.h. je 2 Neuronen repräsentierten einen Konturpunkt (je Koordinate je Punkt ein Neuron). Abbildung 2 veranschaulicht unser Vorgehen nochmals.

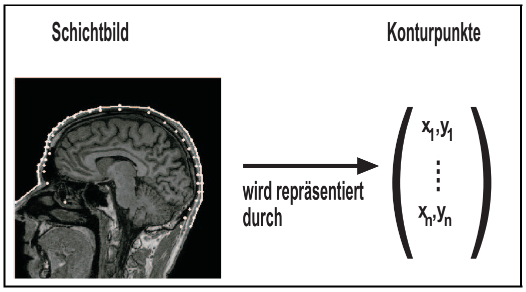


Abb. 2. Die den Kopf umschließende Kontur wird durch eine Menge von Punkten im Netz repräsentiert. Das Netz selbst berechnet wieder eine Menge von Punkten, die der ROI des Gehirns entsprechen.

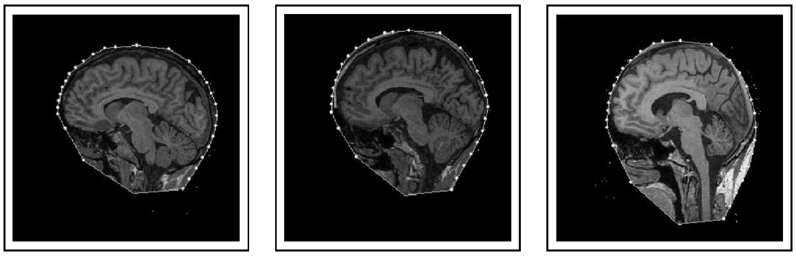
Ein Problem bei dieser Repräsentationsform sind die Variabilitäten in Form, Lage und Größe, so existieren Stauchungen, Streckungen und Verschiebungen in Höhe, Breite und Länge. In ersten Testläufen ergab sich, daß die von uns getesteten Netze nicht in der Lage waren, hier eine Invarianz zu entwickeln (was an der noch geringen Trainingsmenge gelegen hat). Wir haben daher zwei Strategien angewandt, um die Variabilitäten abzuschwächen. Zum einen wurde die Kopfgröße normiert [4], zum anderen ein Konturpunktrasters [1] verwendet.

Die Notwendigkeit der Verwendung eines Punktrasters ergab sich aus den Erfahrungen in der Erstellung von Trainingsdaten, wo sehr schnell deutlich wurde, daß manuelle Konturpunktpositionierung zu ungenau und zu aufwendig ist. Deshalb wurde für die semiautomatische Erstellung von Testdaten von uns ein entsprechendes Werkzeug entwickelt. Die eigentliche Trainingsdatenerstellung zerfiel in drei sequentiell abzuarbeitende Schritte:

1. Automatische Ermittlung der Kopfkontur mittels eines Sobeloperators
2. Interaktive Festlegung der ROI mittels Rasterschablone
3. Generierung eines Testmusters in einem SNNS-spezifischen Format

Alle Neuronalen Netze wurden jeweils mit Hilfe unterschiedlicher Gradientenabstiegsverfahren trainiert. Die meisten Netze konnten nach Abschluß der Lernphase einen nummerischen Lernfehler, bzw. SSE (sum of squared errors), < 1 erreichen, wobei der *StdProp* [5] den Algorithmen *RProp* [5] und *QuickProp* [5] in Konvergenzgechwindigkeit als auch im Endergebnis z.T. deutlich unterlegen war [1].

Nachdem die Netze trainiert waren, wurden sie anhand zuvor erstellten Validierungsdatensätze getestet, bzw. die von den Netzen erzeugte ROI wurde visuell begutachtet (die numerische Bewertung ist aufgrund des Fehlens von Sollvorgaben hier nicht mehr möglich). Fast alle Netze konnten die ROI des Gehirns auch auf den untrainierten Datensätzen finden. Beispiele der ROI-Berechnung sind in Abbildung 3 zu finden.



(a) (b) (c)

Abb. 3. Einige Ergebnisse der Segmentierung der ROI mittels Neuronaler Netze

4 Zusammenfassung und Ausblick

Ausgangspunkt unserer Untersuchungen war die Frage, ob die ROI für Segmentierungsalgorithmen des Gehirns mit Hilfe Neuronaler Netze einfach berechnet werden kann. Diese Frage kann positiv beantwortet werden. Ein wesentlicher Vorteil unseres Verfahrens liegt in der Geschwindigkeit, mit der die Ergebnisse berechnet werden. Mit Ausnahme der größeren Netze standen diese auf einem Pentium 133Mhz praktisch sofort zur Verfügung. Durch Hinzunahme weiterer Informationen wie z.B. Texturparameter, Grauwertinformation, o.ä. kann die Qualität der Netzausgaben vermutlich noch erheblich verbessert werden.

Derzeit wird außerdem durch die Verfügbarkeit von deutlich mehr Datensätzen getestet, ob auch kleinere Strukturen, etwa das Kleinhirn oder der Hirnstamm, mit diesem Verfahren schon grob segmentiert werden können.

Entwicklung, Training und z.T. auch die numerische Bewertung der entwickelten Neuronalen Netze geschah mit Hilfe des Stuttgarter Neuronalen Netz Simulators (SNNS) [5].

Literatur

1. Rittscher J: Einsatz Künstlicher Neuronaler Netze in der medizinischen Bildverarbeitung. Diplomarbeit am Fachbereich Informatik I, Universität Dortmund, 1998
2. Aizenberg I: Processing of noisy and small-detailed gray-scale images using cellular neural networks. Journal of Electronic Imaging, 6(3):272-285, 1997
3. Hiltner J, Aizenberg I, Meyer zu Bexten E, Moraga C: Neural Networks and Fuzzy Logic in Medical Image Processing. 5th International Conference on Soft Computing (Iizuka '98), Fukuoka, Japan, 1998
4. Hiltner J: Operatoren zur deskriptiven und modellbasierten unscharfen Wissensbeschreibung in der medizinischen Bildverarbeitung. Bildverarbeitung für die Medizin, Springer-Verlag, Berlin, 114:118, 1998
5. Zell A: Simulation Neuronaler Netze. Addison-Wesley, 1994