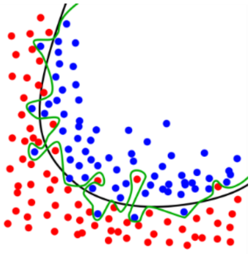
Computational Intelligence

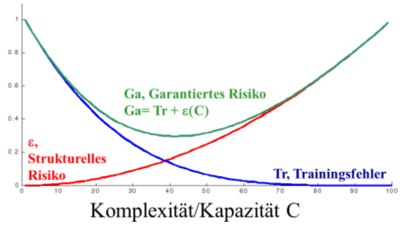
5. Radiale Basis-Funktions-netze, Konvolutionale NN

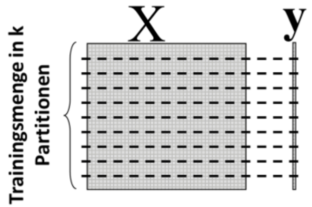
Prof. Dr. Sven Behnke

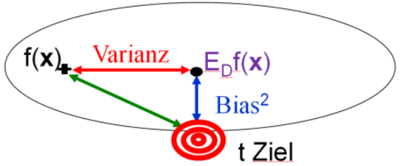
Letzte Vorlesung

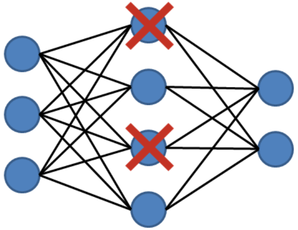
- Overfitting
- Minimierung des strukturellen Risikos
- Regularisierung
  - Geringe Anzahl der Modellparameter
  - Early Stopping
  - Weight Decay
- Kreuzvalidierung zur Selektion von Hyper-Parametern
- Bias-Varianz-Dilemma
- Mitteln von Prediktoren
  - Bagging
  - Boosting
  - Drop-out



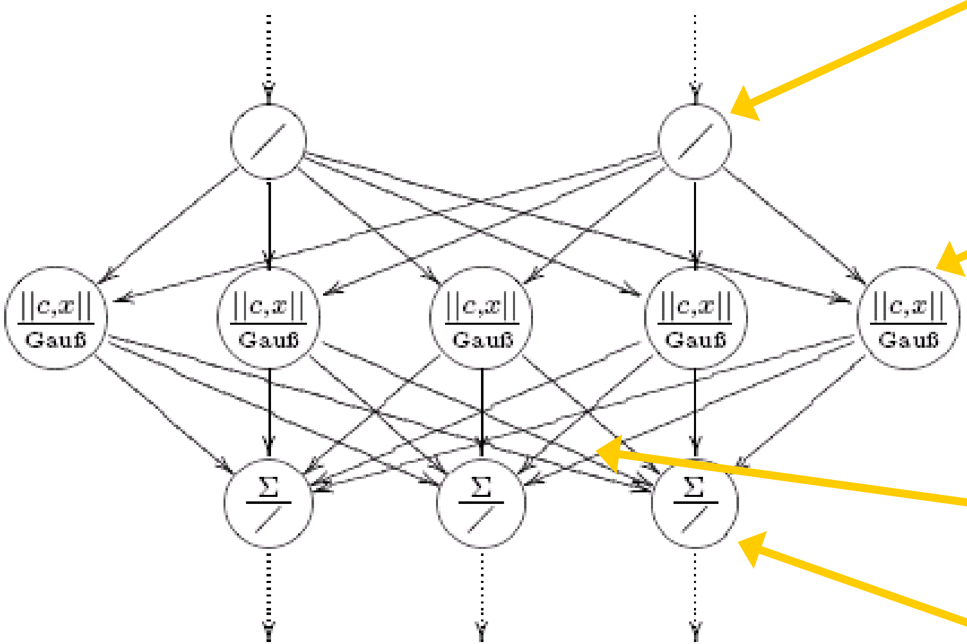








Radiale Basis-Funktions-Netze (RBF)

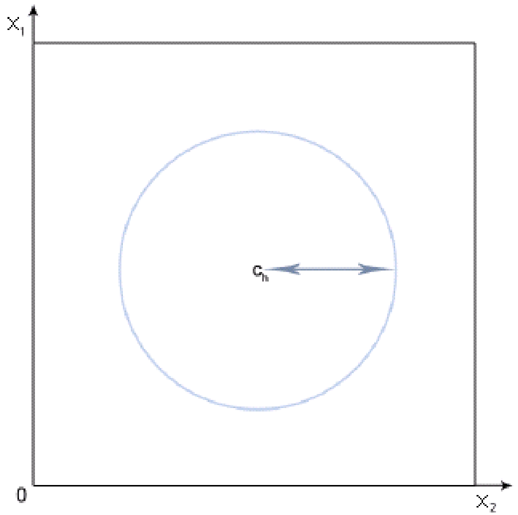


- Eingabeneuronen
- **RBF-Neuronen** $h_1,\ldots,h_k$ berechnen Abstände der Eingabe zu ihrem Gewichts-Vektoren $c_1,\ldots,c_k$, monoton fallende Aktivierungs-funktion
- Überwacht zu trainierende Gewichtsschicht
- Ausgabeneuronen $y_1,\ldots,y_k$

RBF-Neuron Integrationsfunktion

- Beim RBF-Neuron verwendet man den **Euklidischen Abstand** zwischen dem Eingabevektor $\boldsymbol{x}$ und dem Zentrum $\boldsymbol{c_h}$ des RBF-Neurons h als **Integrationsfunktion**:

$$r_h = f_{in}(x) = \|x - c_h\| = \sqrt{\sum_{i \in I} \left(x_i - c_{h,i}\right)^2}$$

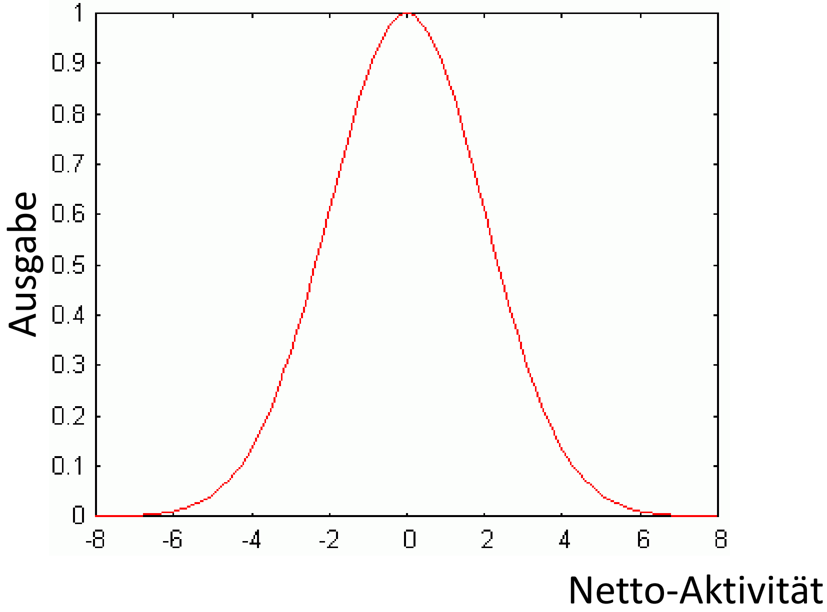


RBF-NEURON AKTIVIERUNGSFUNKTION

- Dieser Abstand wird dann durch eine monoton von Null aus fallende Aktivierungsfunktion geschickt.

- Üblicherweise wird die Gauß-Funktion verwendet:

$$\boxed{f_{act}(r_h) = e^{\left(\frac{-r_h^{\,2}}{2\sigma_h^{\,2}}\right)}}$$

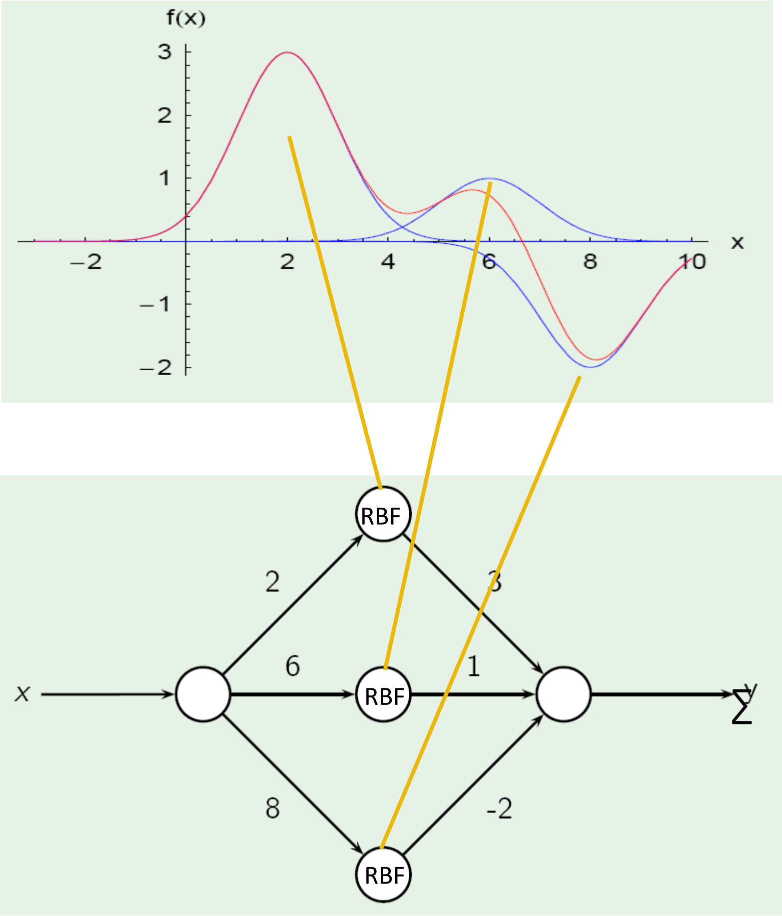


→ **Lokale Effekte der Hidden-Units**

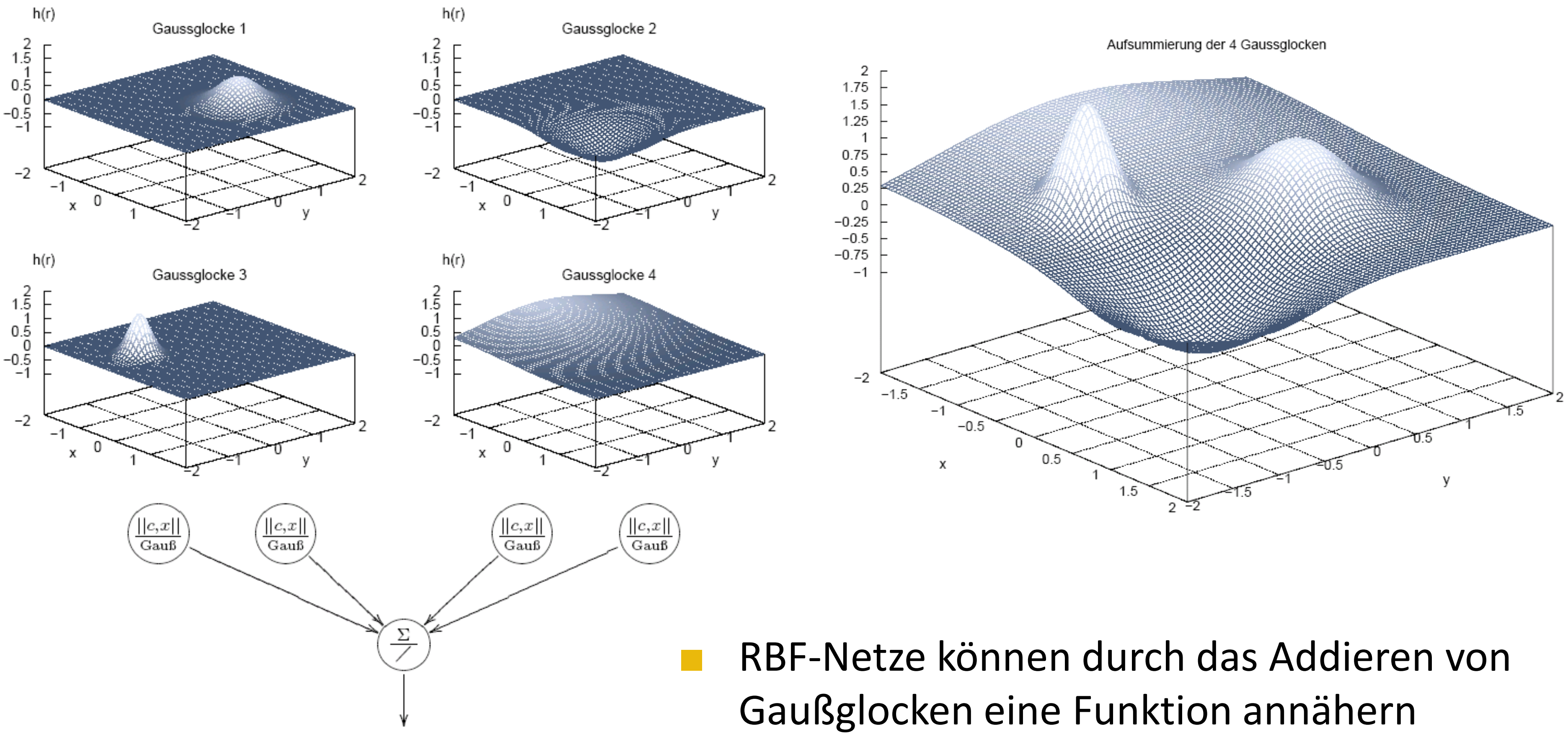
RBF-Funktionsapproximation 1D

- Ausgabe wird aus verschobenen und skalierten Gaußglocken zusammengesetzt

- Lokale Effekte von Gewichts-änderungen



RBF-Funktionsapproximation 2D



- RBF-Netze können durch das Addieren von Gaußglocken eine Funktion annähern

Training von RBF-Netzen

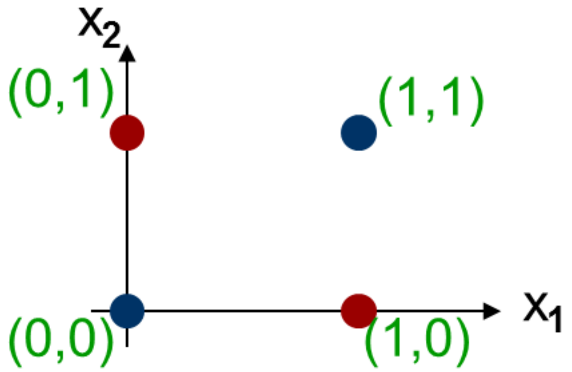
- Unüberwachte Initialisierung der RBF-Neuronen
  - Z.B. Gruppierung der Eingabevektoren oder zufällige Auswahl von Beispielen
- Überwachtes Training der Ausgabeschicht durch Delta-Regel minimiert quadratischen Regressionsfehler:

$$\boxed{\begin{aligned}\Delta w_{h,\Omega} &= \eta \cdot \delta_\Omega \cdot o_h \\ &= \eta \cdot \left(t_\Omega - o_\Omega\right) \cdot f_{act}\left(\left\| x - c_h \right\|\right)\end{aligned}}$$

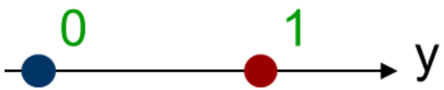
- Vorteil: Keine Backpropagation durch die verdeckte Schicht nötig
- Bei nichtlinearen Ausgabe-Neuronen auch Klassifikation möglich (Multiplikation mit Ableitung der Transfer-Fkt. nötig)
- Nichtlineares Mapping der Eingabe in höherdimensionalen Raum erleichtert lineare Trennbarkeit [Cover, 1965]

Beispiel: XOR-Problem

- **Eingaberaum**:



- **Ausgaberaum**:



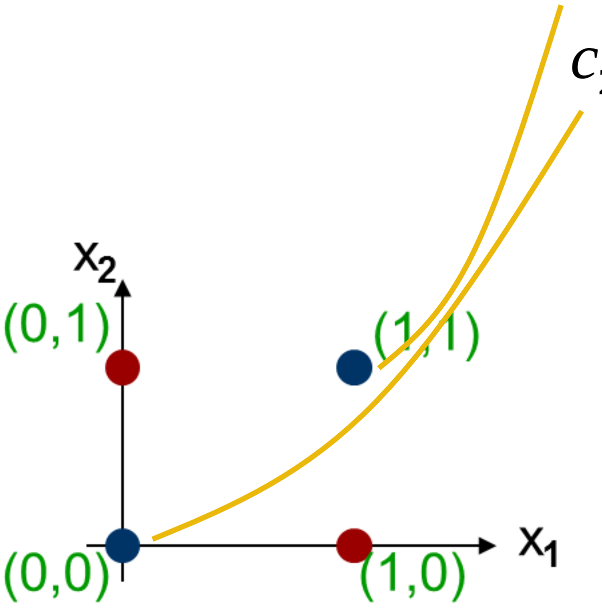
- Konstruiere RBF-Klassifikator, so dass:
  - **(0,0)** und **(1,1)** werden auf **0** abgebildet (**Klasse C1**)
  - **(1,0)** und **(0,1)** werden auf **1** abgebildet (**Klasse C2**)
- Klassen linear nicht trennbar

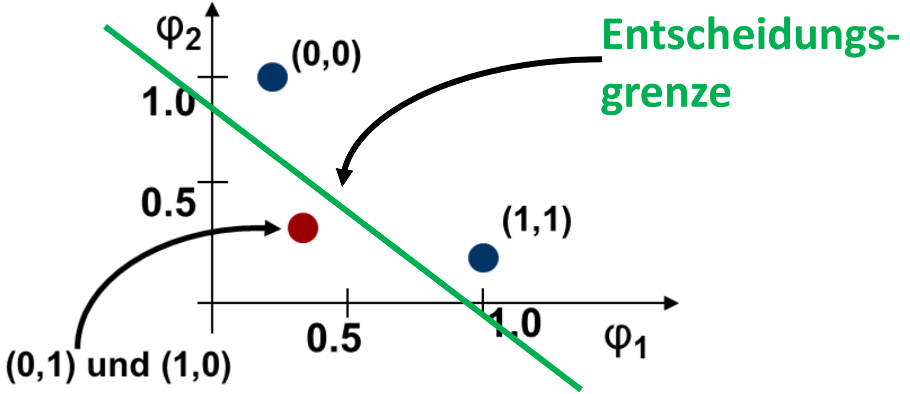
Beispiel: XOR-Problem

- Im **Merkmalsraum** der verdeckten Schicht:

$$c_1 = [1,1]^T \qquad \varphi_1(x_1, x_2) = e^{-\|x-c_1\|^2}$$

$$c_2 = [0,0]^T \qquad \varphi_2(x_1, x_2) = e^{-\|x-c_2\|^2}$$

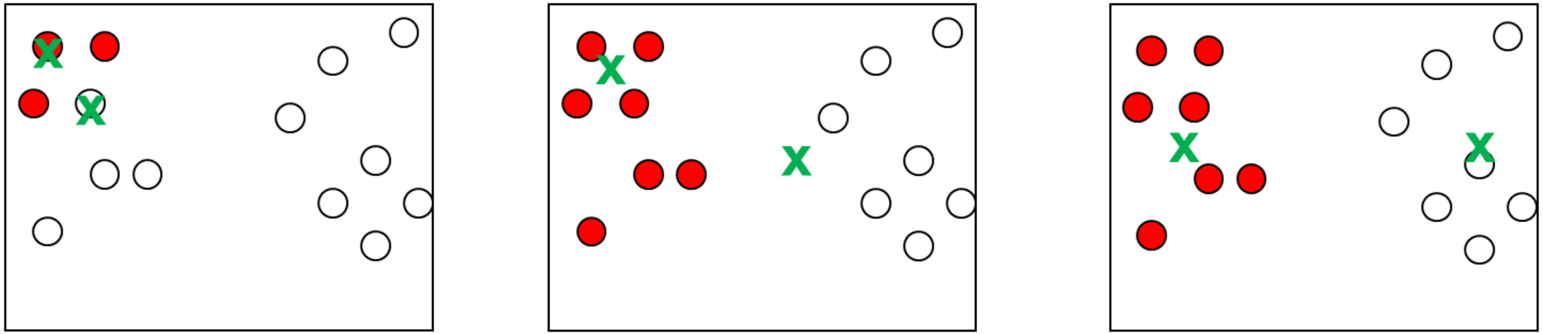




- **Linear trennbar!**

RBF-Initialisierung mit k-Means

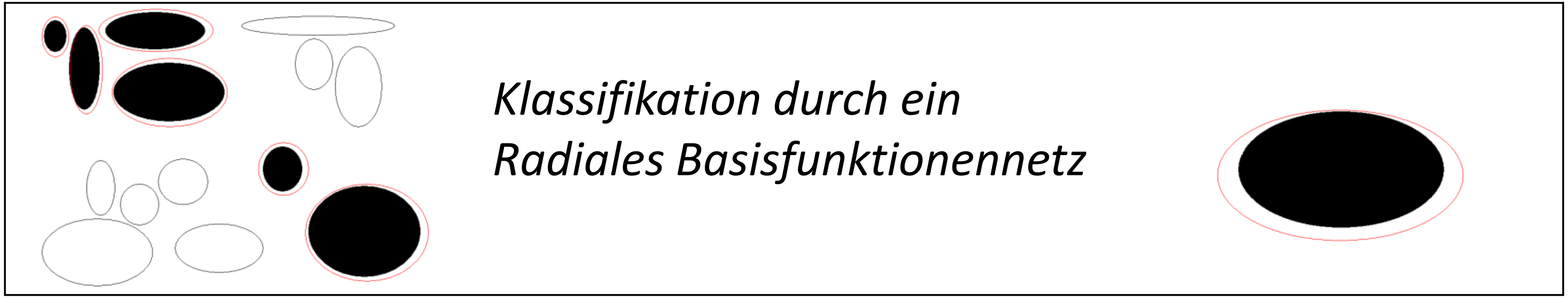
- Start: Zufällige Initialisierung oder zufällige Auswahl der Eingabevektoren
- **Expectation**: Zu jedem Eingabevektor wird das nächste RBF-Zentrum gewählt
- **Maximization**: Die RBF-Zentren werden zum Mittelwert der ihnen zugeordneten Eingabevektoren bewegt
- Iterativ bis die Qualität nicht mehr besser wird.

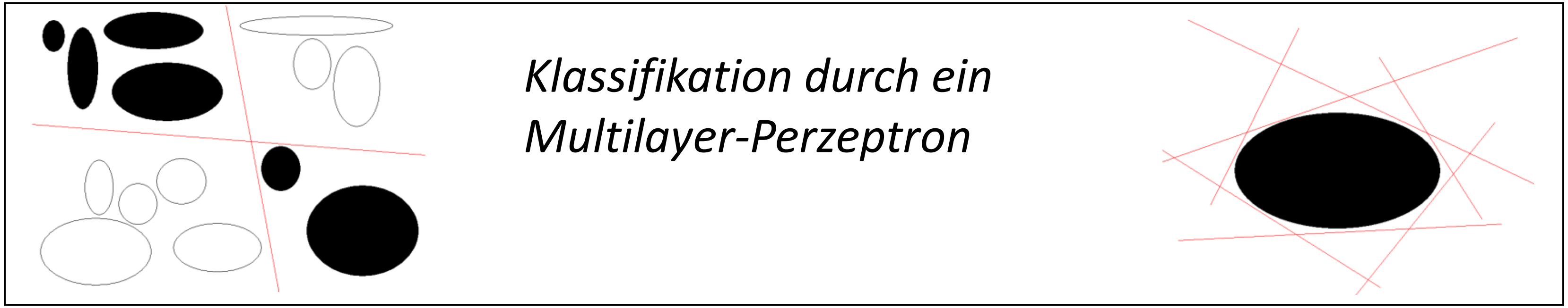


- Keine Garantie für optimale Aufteilung – lokale Optima
- Nicht robust – Ausreißer beeinflussen das Ergebnis stark
- Wahl der Zentrenzahl k beeinflusst das Ergebnis stark
- Variante: beginne mit kleinem k und erhöhe k im Laufe des Verfahrens, z.B. durch Spaltung des größten Clusters (oder des Clusters mit dem größten aggregierten Fehler)

Vergleich: RBF-Netze und MLPs

- Ob ein Problem einfacher von einem RBF-Netz oder von einem MLP gelöst werden kann, hängt von der Verteilung der Trainingsbeispiele ab:

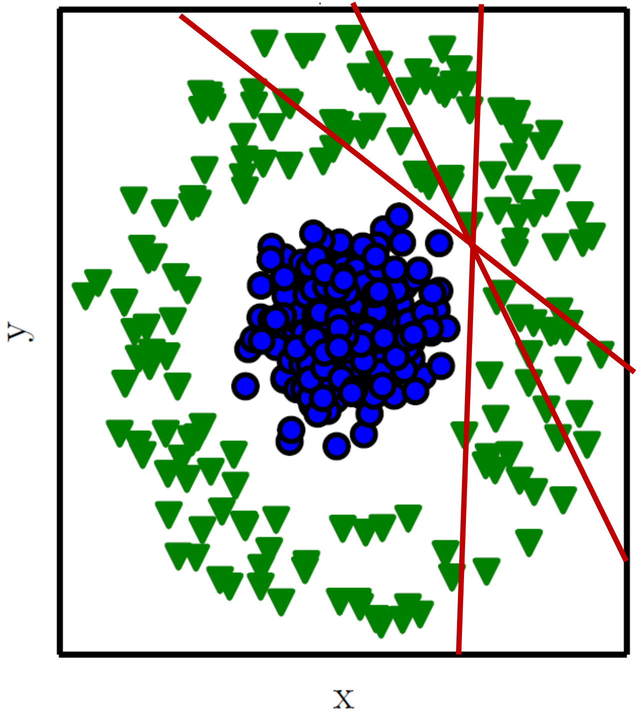




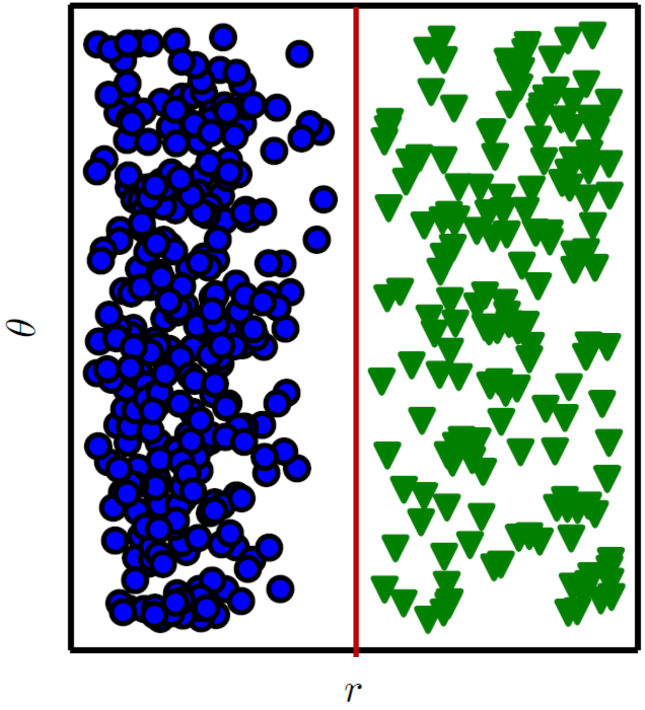
- Globale (MLP) vs. lokale (RBF) Klassifikation
- Trainigsaufwand vs. #Hidden Units (Recall-Aufwand)

Gewählte Repräsentation ist Wichtig

- Kartesische Koordinaten

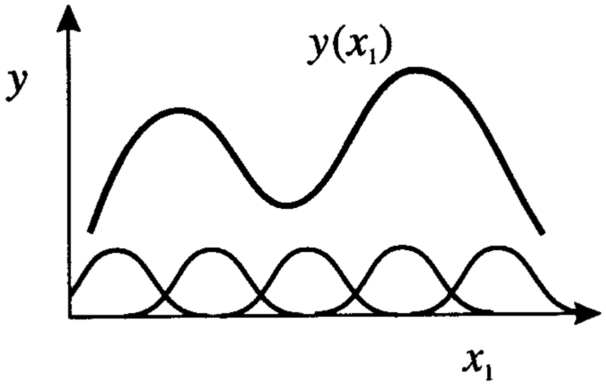


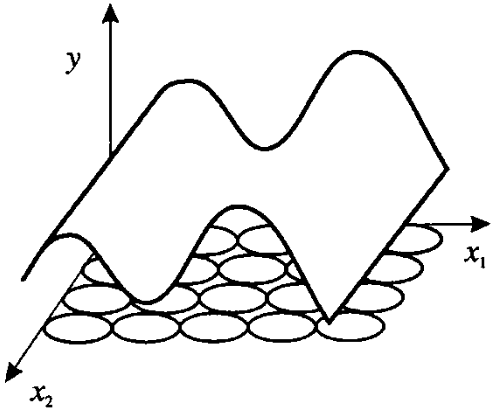
- Polarkoordinaten



[Goodfellow]

Ein Problem von RBF-Netzen





- Wenn die gewünschte Ausgabe nur von einer Eingabe-Komponente abhängt, kann ein RBF-Netz dies aufgrund des unüberwachten Trainings der Zentren nicht erkennen.
- Ein MLP kann die Gewichte zu den irrelevanten Eingaben auf Null setzen.
- Man kann auch Sigma-Units und RBF-Units in der verdeckten Schicht mischen.

Einige wichtige Prinzipien

- Occam‘s Razor
  - Wenn zwei Modelle die Daten gleich gut beschreiben, dann wähle das einfachere

  → **Komplexer (mächtiger) ist nicht automatisch besser**

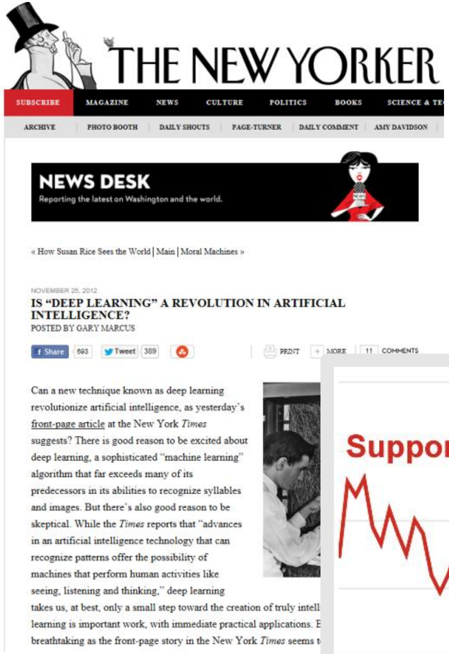
- Fluch der Dimension
  - Für komplexe Lerner steigt der Bedarf an Beispielen überlinear (exponentiell) mit der Zahl der Merkmale

  → **Nimm nur Merkmale, die notwendig sind**

- No free Lunch
  - Es gibt keinen Lerner, der für alle Probleme die beste Lösung liefert

  → **Wende komplexen Lerner nie blind ohne Wissen über die Daten an**

Trend in den letzten Jahren: Deep Learning


THE NEW YORKER

SUBSCRIBE | MAGAZINE | NEWS | CULTURE | POLITICS | BOOKS | SCIENCE & TE

ARCHIVE | PHOTO BOOTH | DAILY SHOUTS | PAGE-TURNER | DAILY COMMENT | AMY DAVIDSON

NEWS DESK
Reporting the latest on Washington and the world.

« How Susan Rice Sees the World | Main | Moral Machines »

NOVEMBER 25, 2012

IS "DEEP LEARNING" A REVOLUTION IN ARTIFICIAL INTELLIGENCE?

POSTED BY GARY MARCUS

Can a new technique known as deep learning revolutionize artificial intelligence, as yesterday's front-page article at the New York *Times* suggests? There is good reason to be excited about deep learning, a sophisticated "machine learning" algorithm that far exceeds many of its predecessors in its abilities to recognize syllables and images. But there's also good reason to be skeptical. While the *Times* reports that "advances in an artificial intelligence technology that can recognize patterns offer the possibility of machines that perform human activities like seeing, listening and thinking," deep learning takes us, at best, only a small step toward the creation of truly intell learning is important work, with immediate practical applications. E breathtaking as the front-page story in the New York *Times* seems t


The Register®

Data Centre | Cloud | Software | Networks | Security | Policy | Business | Jobs | Hardware | Science | Bootno

Financial News | Small Biz | CIO | Media

BUSINESS

Baidu muscles in on Google's turf with Silicon Valley deep learning lab

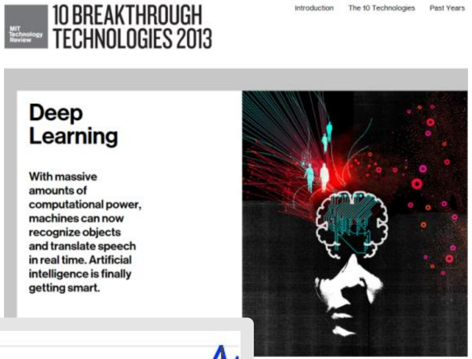
Chinese search giant beds down next to Apple in Cupertino

By Phil Muncaster, 15th April 2013 Follow 3,371 followers

Win a Samsung 40-inch LED HDTV with The Reg and HP!

Chinese search giant Baidu has opened the doors to a new research facility in Google's back yard where it's hoping to tap the local talent to consolidate early mover advantage in the burgeoning field of "deep learning".

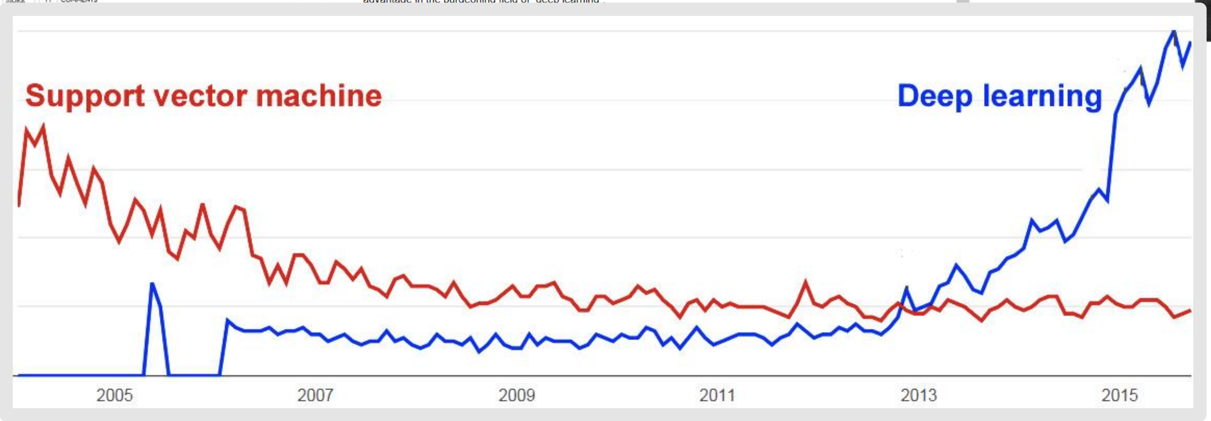
RELATED


10 BREAKTHROUGH TECHNOLOGIES 2013

Introduction | The 10 Technologies | Past Years

Deep Learning

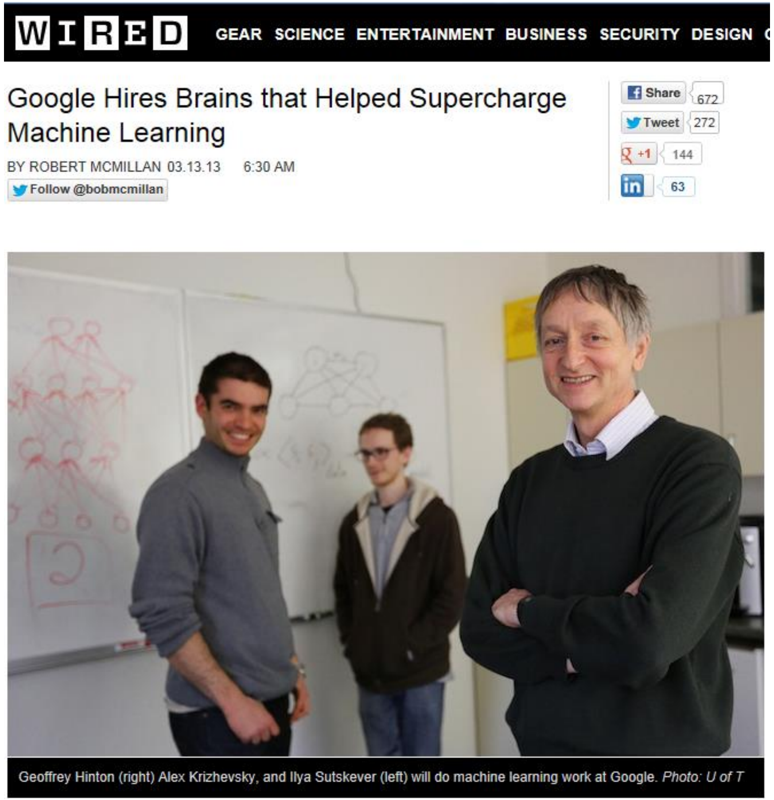
With massive amounts of computational power, machines can now recognize objects and translate speech in real time. Artificial intelligence is finally getting smart.



[Google Trends]

Starkes Interesse der Industrie

- Google
  - DNNresearch (Geoffrey Hinton)
  - DeepMind (Demis Hassabis)
- Baidu
  - Andrew Ng
- Facebook
  - Yann LeCun
- Microsoft
  - Li Deng



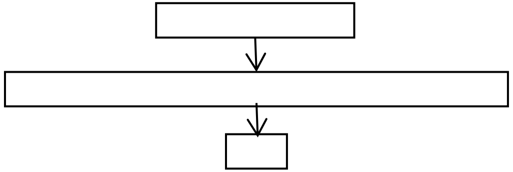
Definition: Deep Learning

- Deep Learning bezeichnet eine Menge von Algorithmen des maschinellen Lernens – häufig neuronale Netze – die versuchen **geschichtete Repräsentationen** von Eingabedaten zu lernen.
- Die Schichten in diesen Modellen entsprechen unterscheidbaren **Konzept-Ebenen**, wobei
  - Konzepte höherer Ebenen auf der Grundlage von Konzepten niederer Ebenen definiert werden und
  - Konzepte niederer Ebenen für die Definitionen vieler Konzepte höherer Ebenen verwendet werden können.

[Bengio 2009]

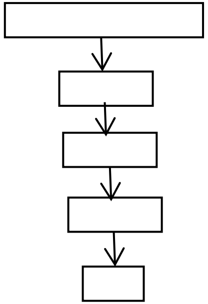
Flache vs. Tiefe Netzwerke

- Ein neuronales Netz mit einer einzelnen verdeckten Schicht, die breit genug ist, kann jede Funktion berechnen (Cybenko, 1989)



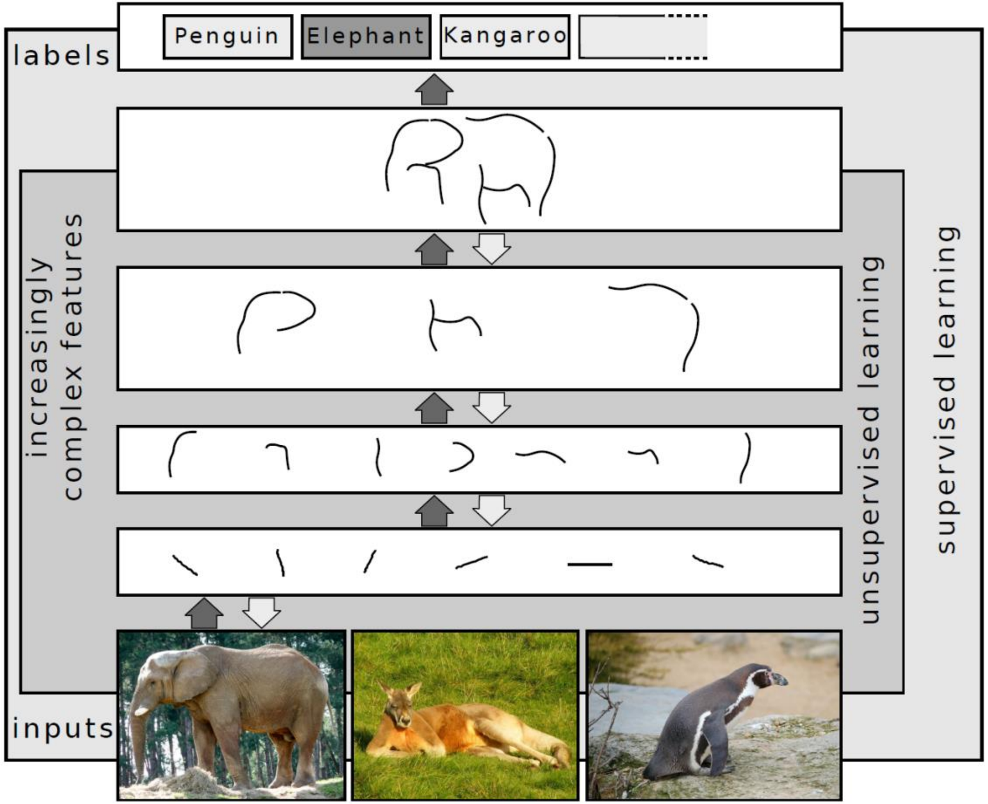
  - Manche Funktionen, z.B. die Paritätsfunktion, benötigen **exponentiell viele Hidden Units** (in der Anzahl der Eingaben)

- **Tiefe Netzwerke** (mit mehreren verdeckten Schichten) können exponentiell effizienter sein



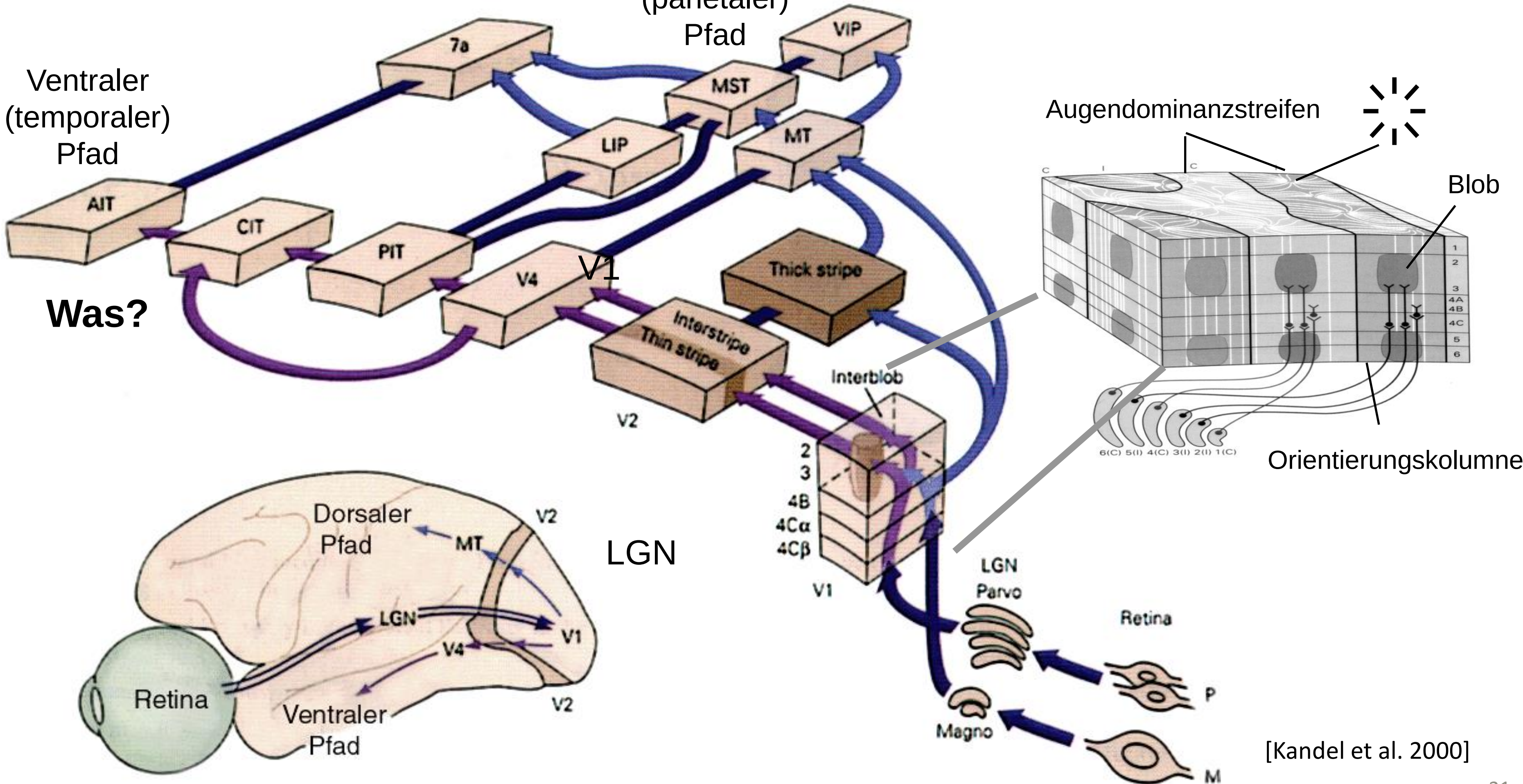
  - Paritätsfunktion: Sequentielle Berechnung des Übertragsbits

Repräsentationsebenen



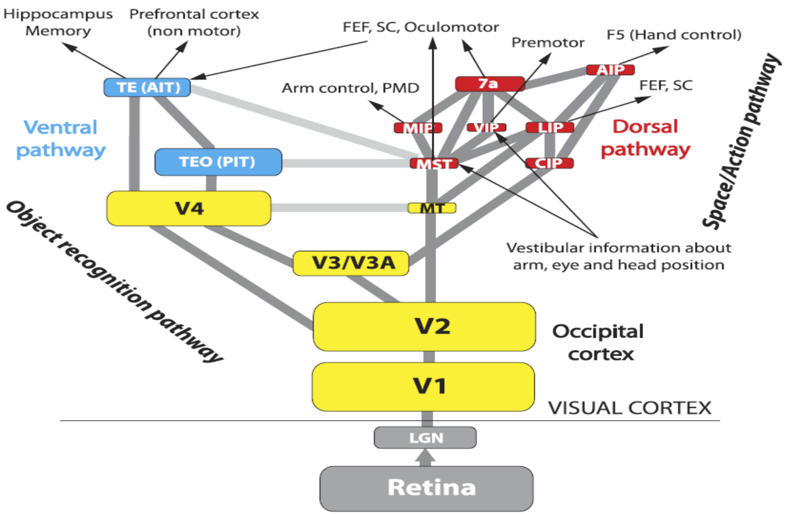
[Schulz and Behnke, KI 2012]

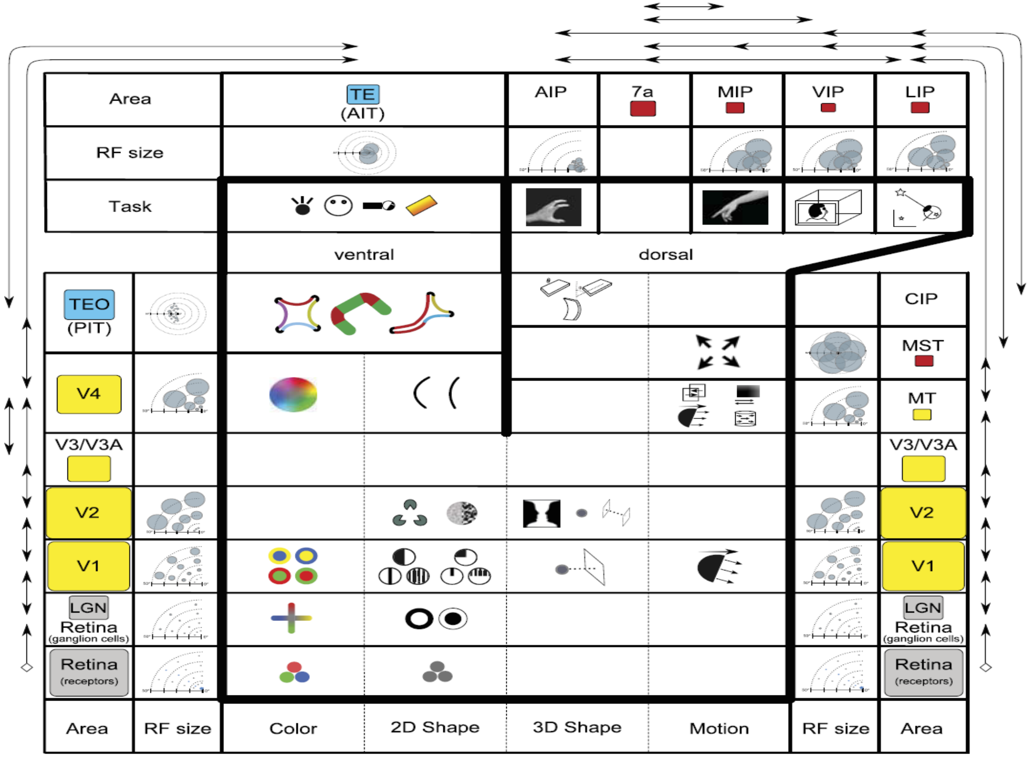
Visuelles System



[Kandel et al. 2000]

Visuelle Verarbeitungshierarchie



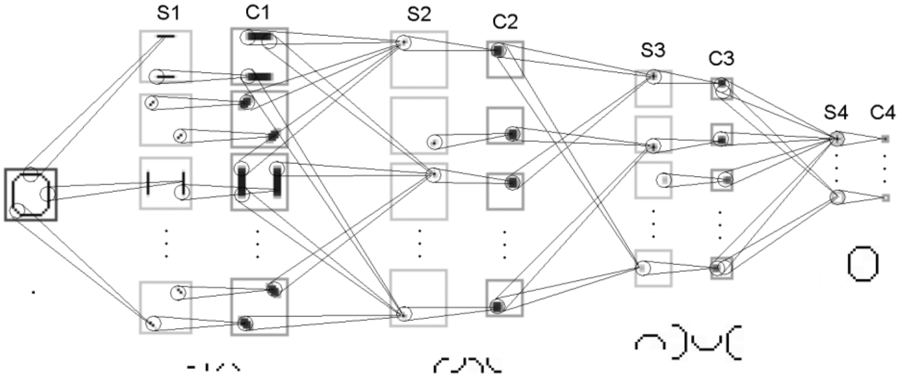


[Krüger et al., TPAMI 2013]

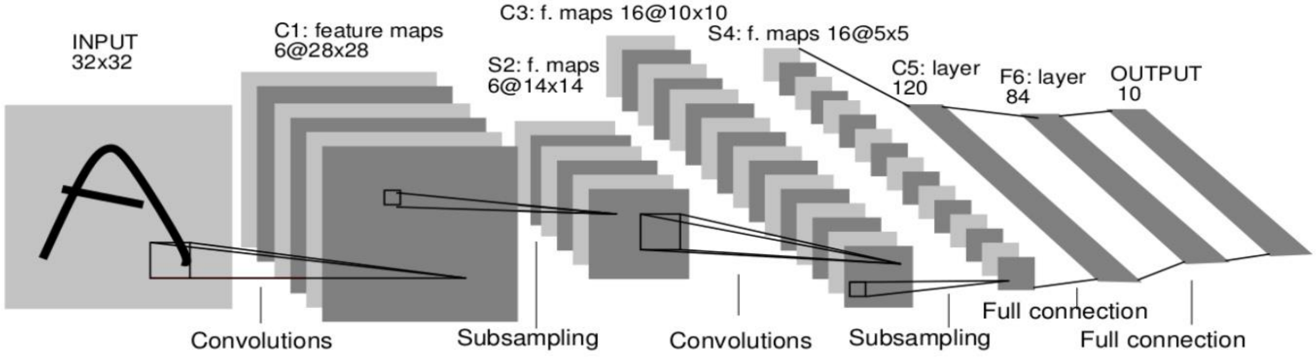
- Steigende Komplexität
- Zunehmende Invarianz
- Alle Verbindungen bidirektional
- Mehr Feedback als Feedforward
- Laterale Verbindungen wichtig

Konvolutionale Neuronale Netze

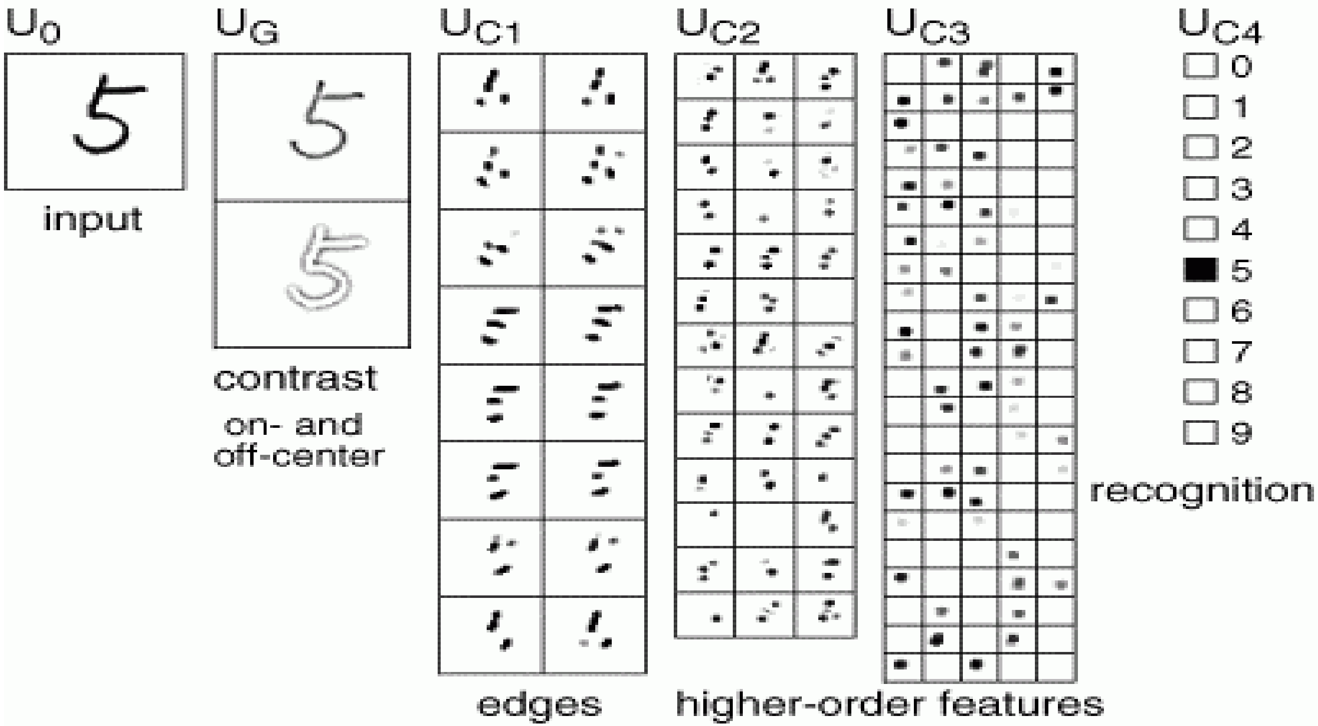
- Neokognitron: Fukushima 1980



- Überwachtes Training: LeCun 1989

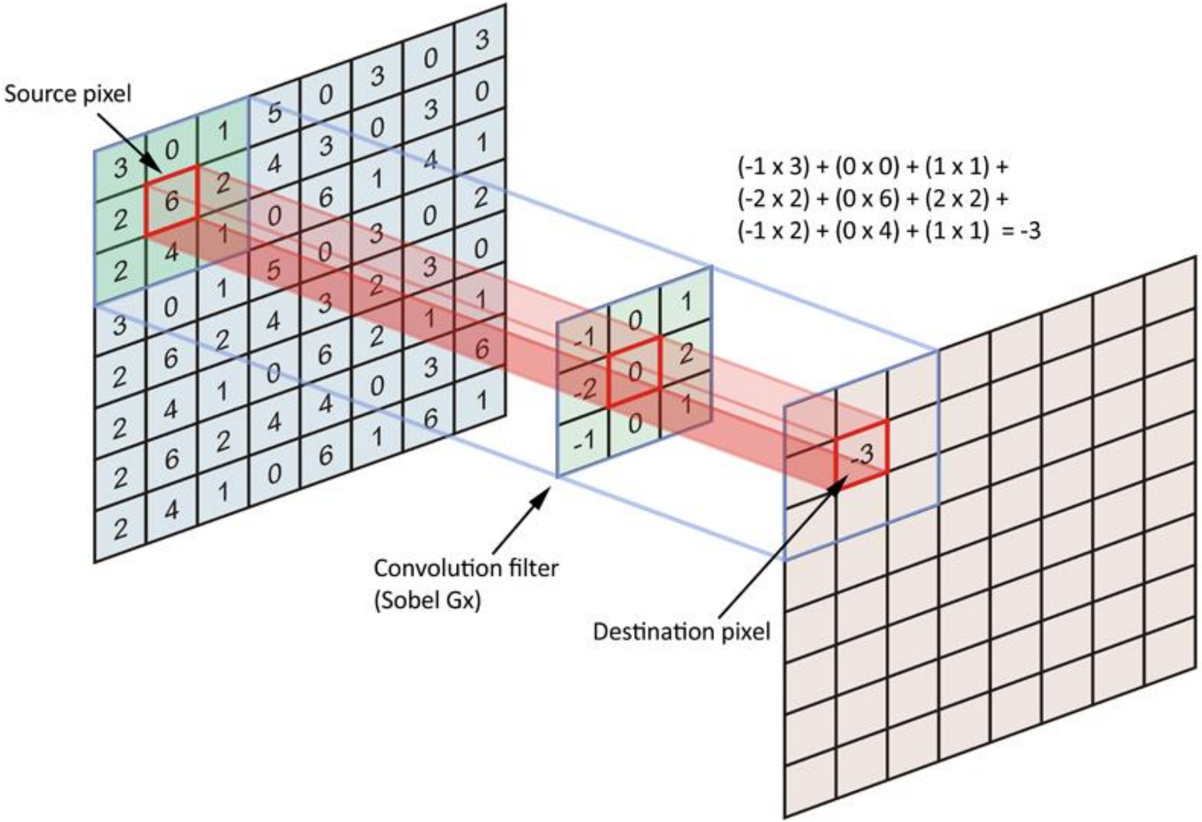


LeNet: Erkennung handgeschriebener Ziffern



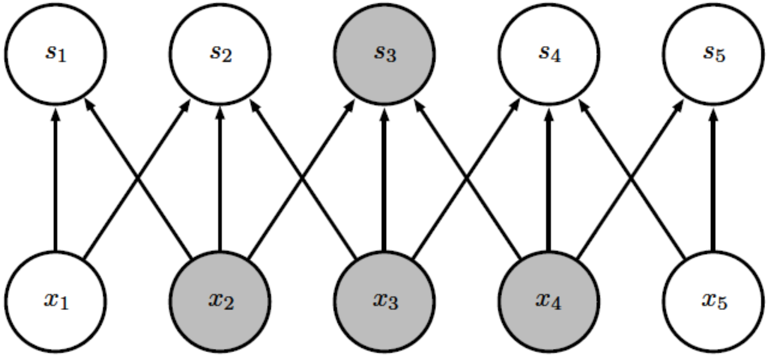
[LeCun]

Konvolution / Faltung

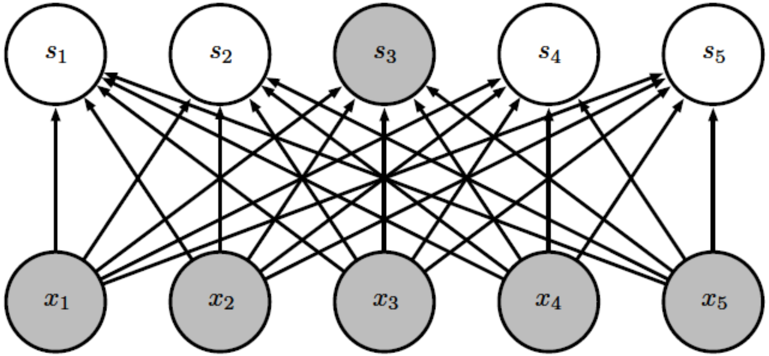


Spärliche, Lokale Konnektivität; Weight Sharing

- 1D-Konvolution

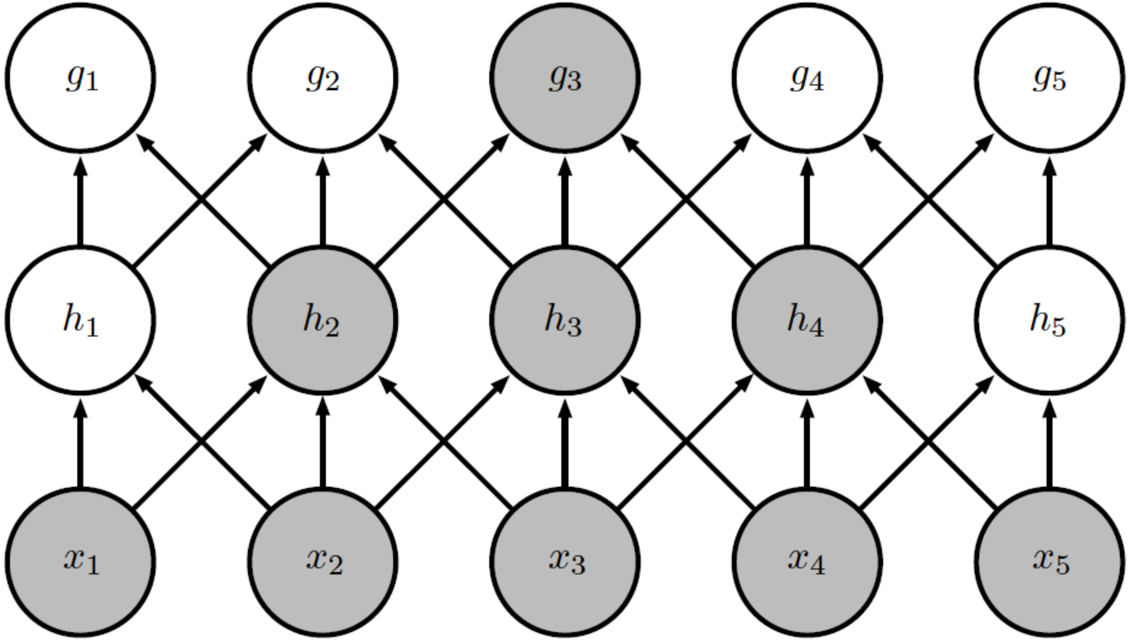


- Vollständig vernetzt



[Goodfellow]

Wachstum des Rezeptiven Feldes

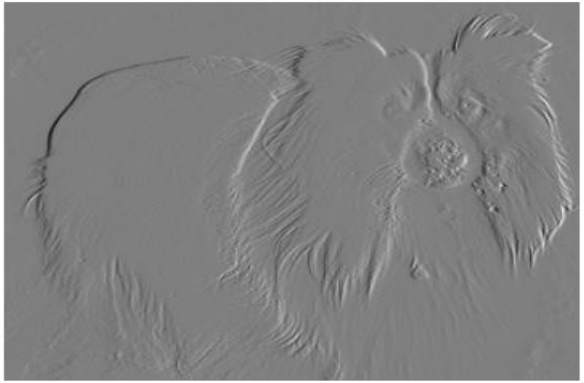


[Goodfellow]

Kantenerkennung durch Faltung mit Differenzkern

Eingabe

Ausgabe



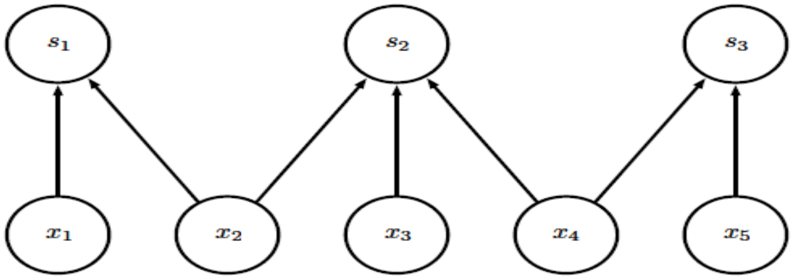
Kern

1	-1

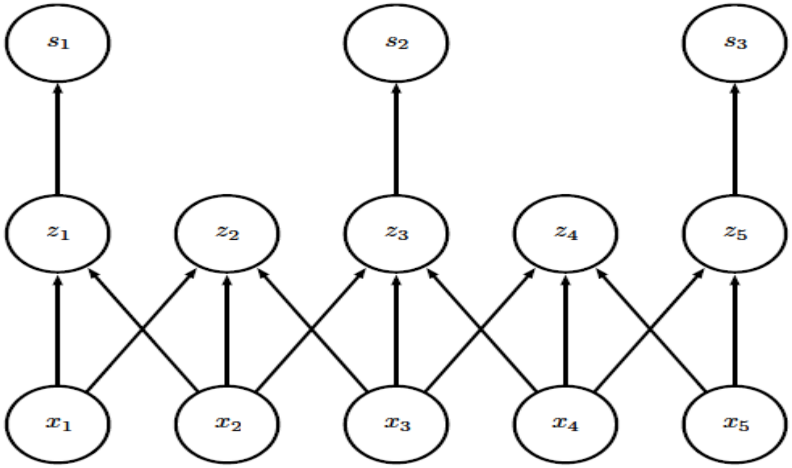
[Goodfellow]

Schrittweite (Stride)

- Konvolution mit Schrittweite 2:



- Konvolution, gefolgt von Unterabtastung (Subsampling)

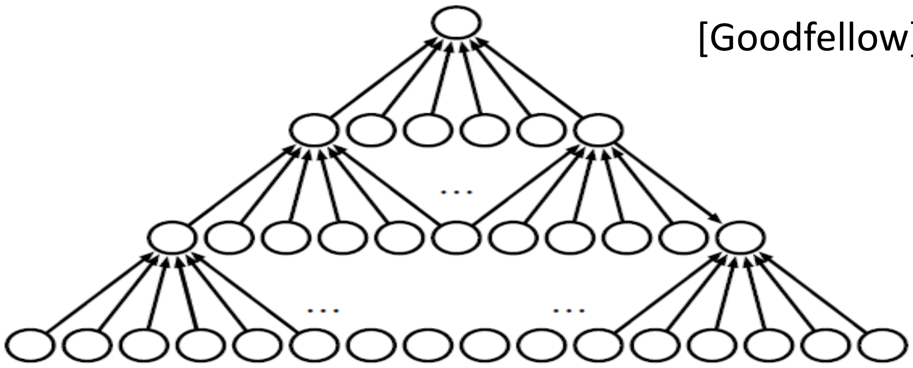


[Goodfellow]

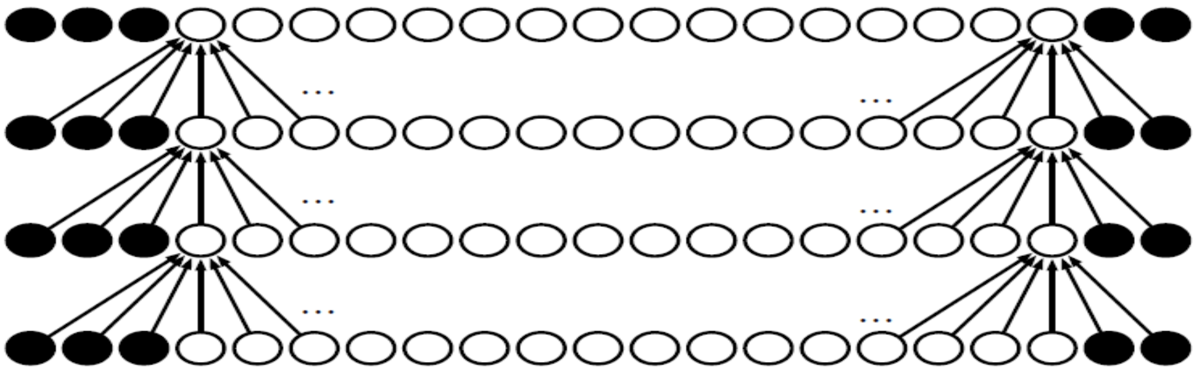
Ergänzung des Rands (Border Padding)

[Goodfellow]

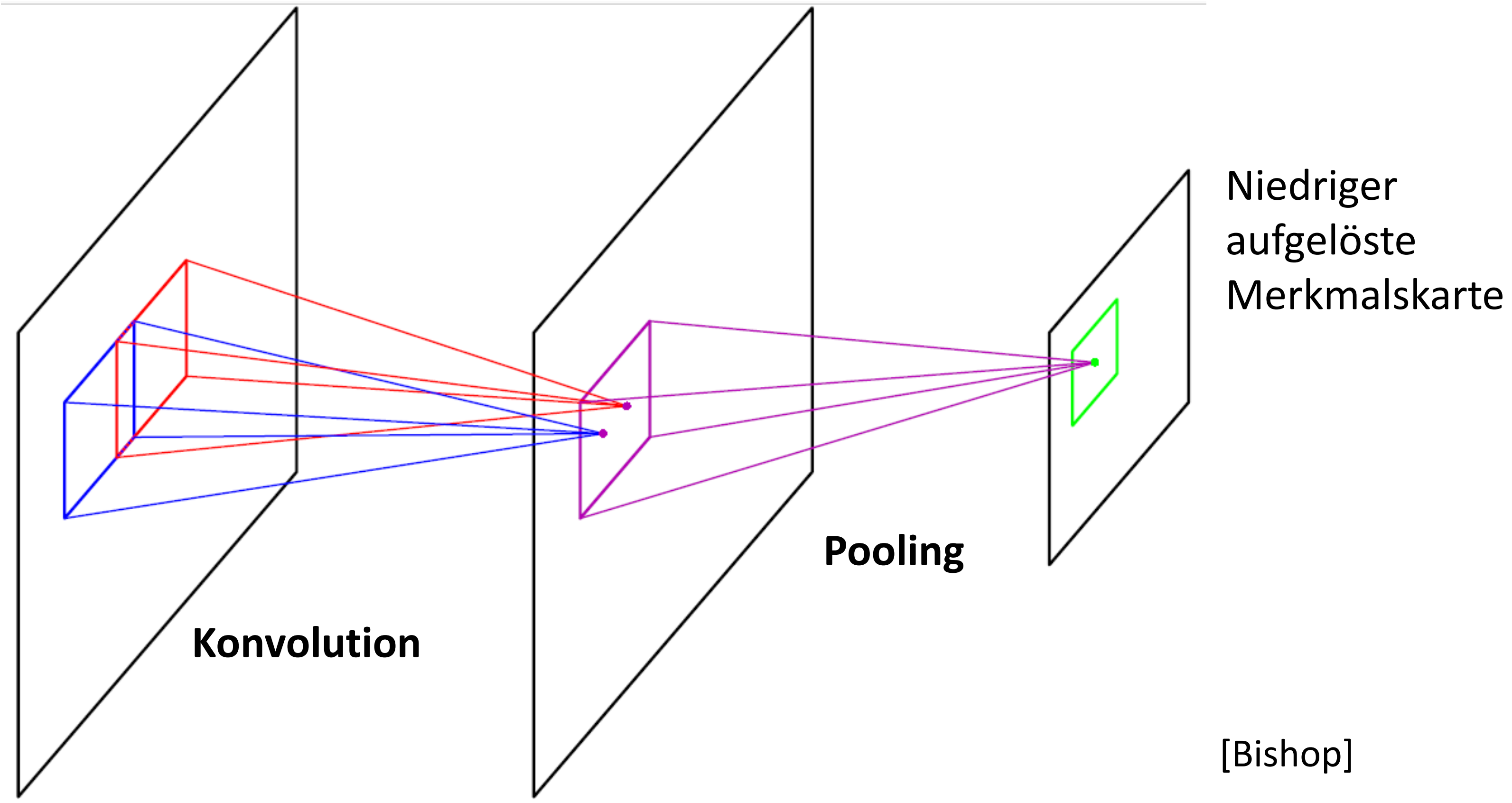
- Valide Konvolutionen reduzieren Bildgröße



- Auffüllen des Rands, um Bildgröße zu erhalten
  - Null (Zero padding)
  - Spiegeln von Aktivitäten
  - Kopieren von Aktivtäten



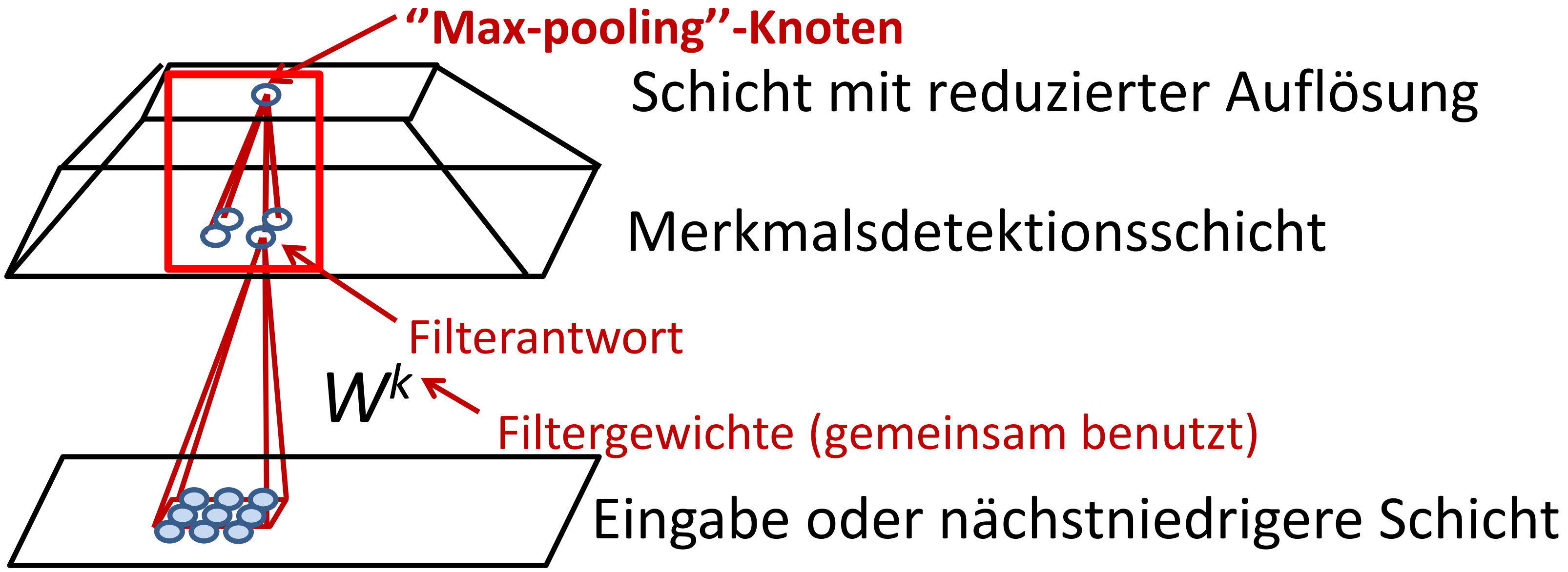
Konvolution und Pooling



[Bishop]

Max-Pooling

- $o'_{i,j} = \max(o_{2i,2j}, o_{2i+1,2j}, o_{2i,2j+1}, o_{2i+1,2j+1})$



- Erzeugt Invarianz gegen lokale Verschiebungen

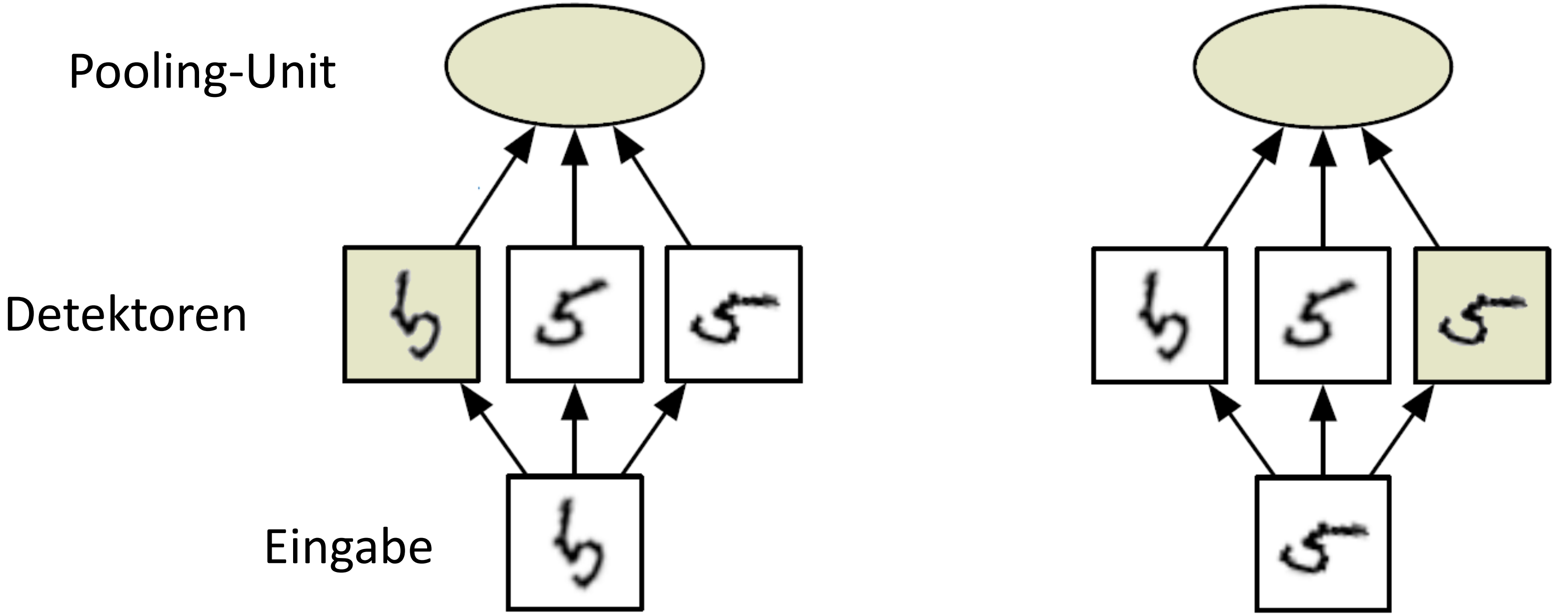
1	1	2	4
5	6	7	8
3	2	1	0
1	2	3	4

max pool with 2x2 filters and stride 2

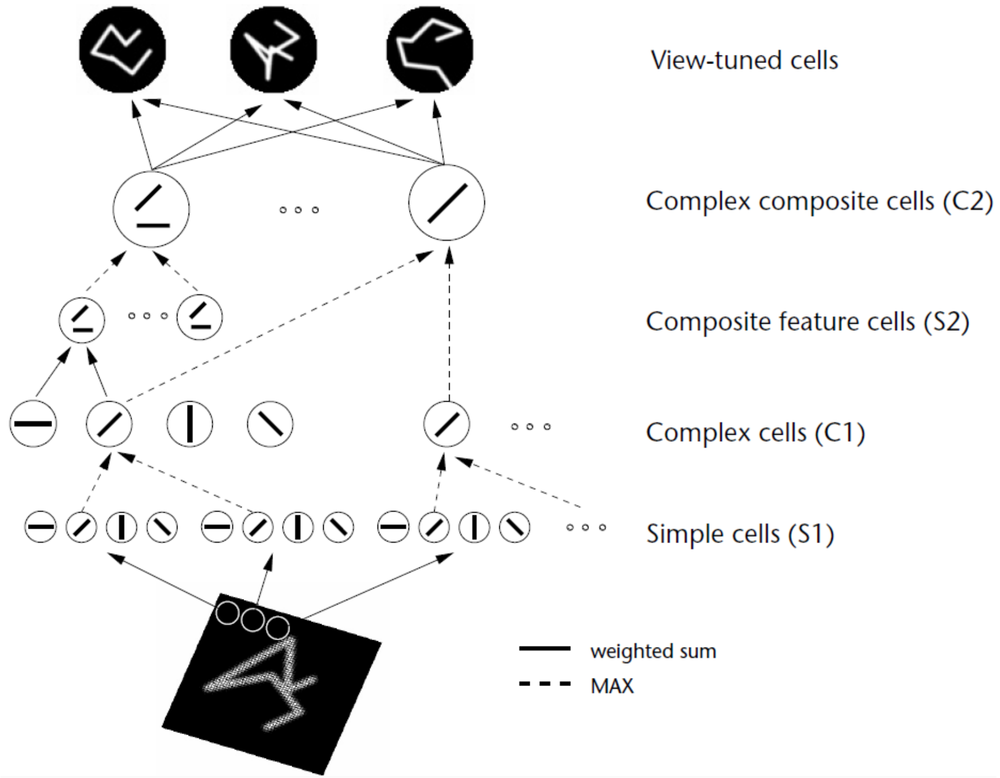
6	8
3	4

Pooling über Markmalskanäle (Cross-Channel Pooling)

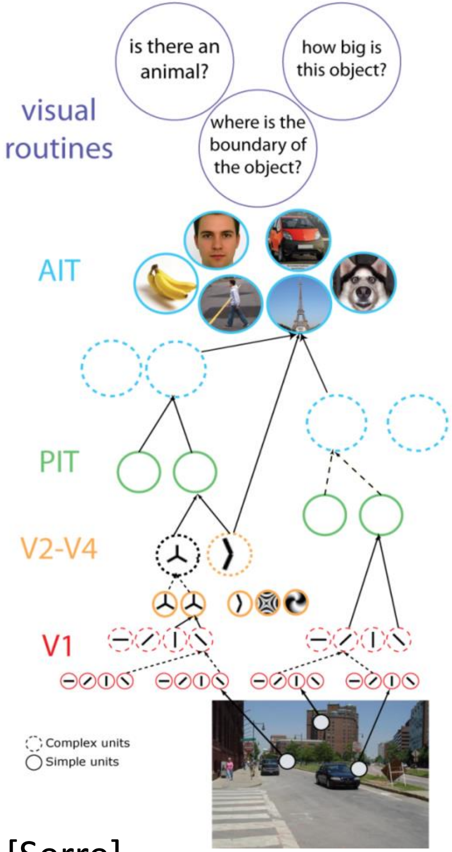
- Erzeugt Invarianz gegenüber erlernten Transformationen



HMAX-ModelL



[Riesenhuber and Poggio 1999]

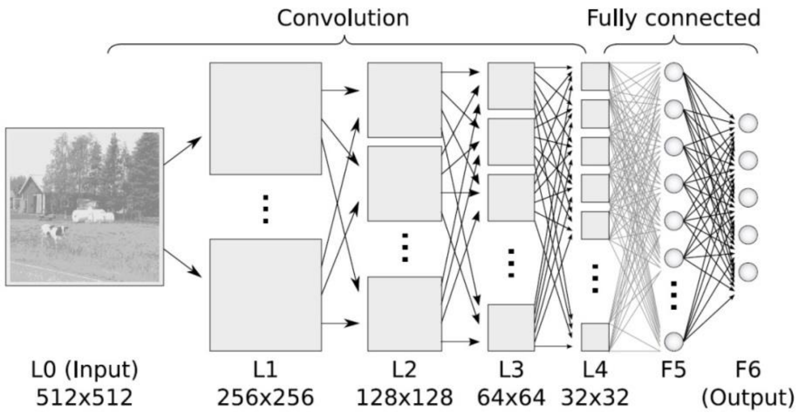


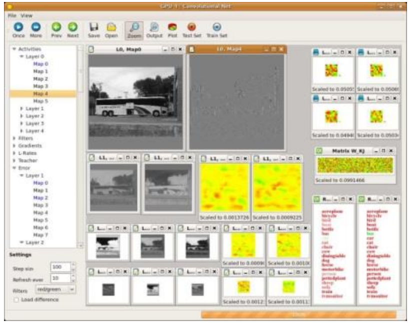
[Serre]

GPU-Implementierung (CUDA)

- Preiswerte Parallelrechner
- Programierbar durch CUDA-SDK
- Konvolutionale NN

[Scherer & Behnke, 2009]



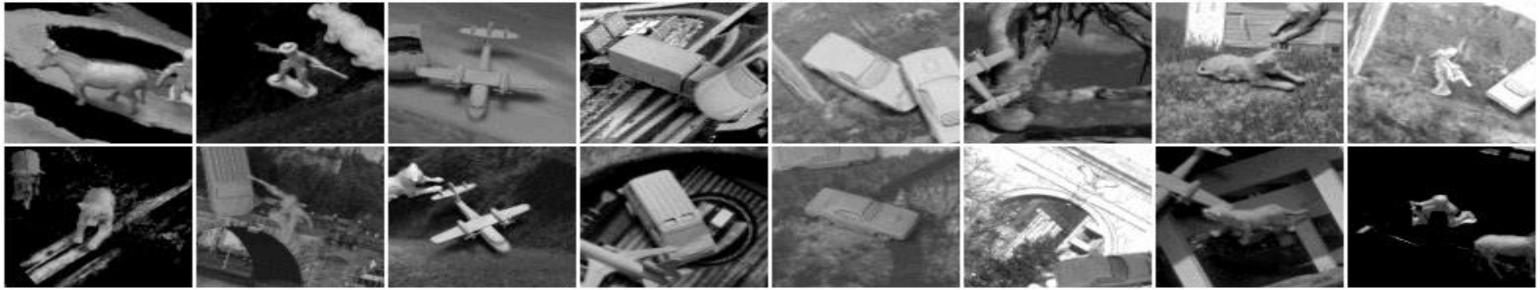


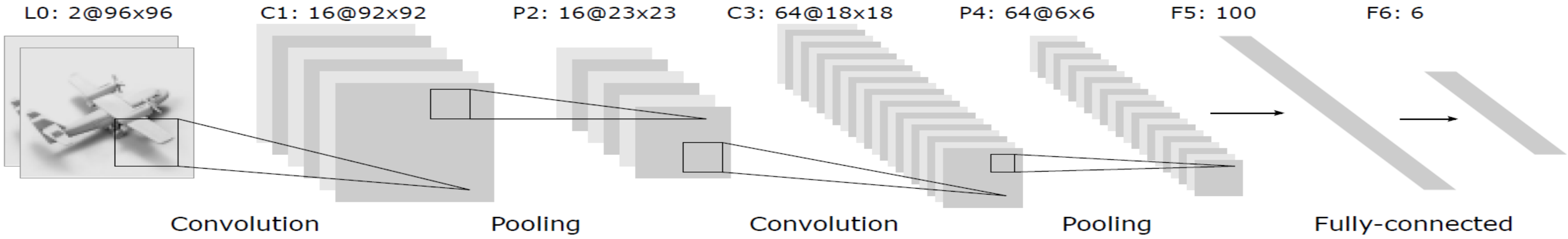
- Lokale Konnektivität

[Uetz & Behnke, 2009]

Kategorisierung von Bildern: NORB

- 10 Kategorien, jittered-cluttered



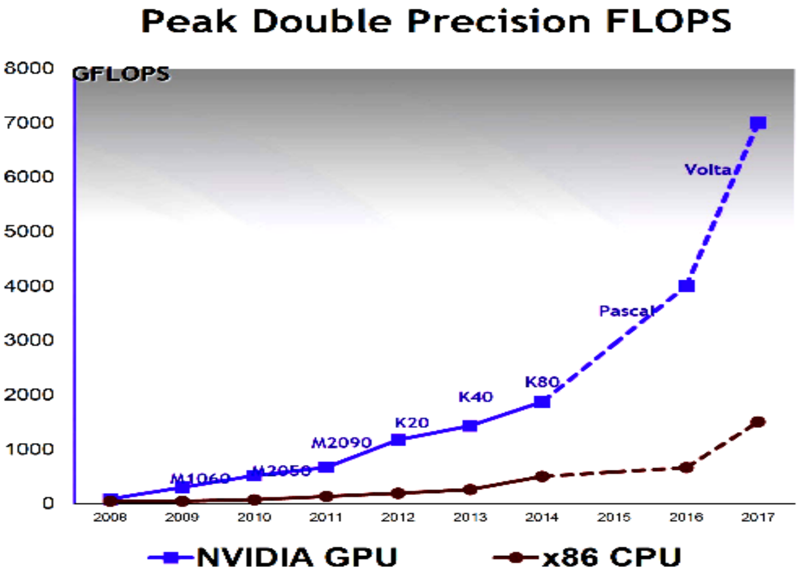


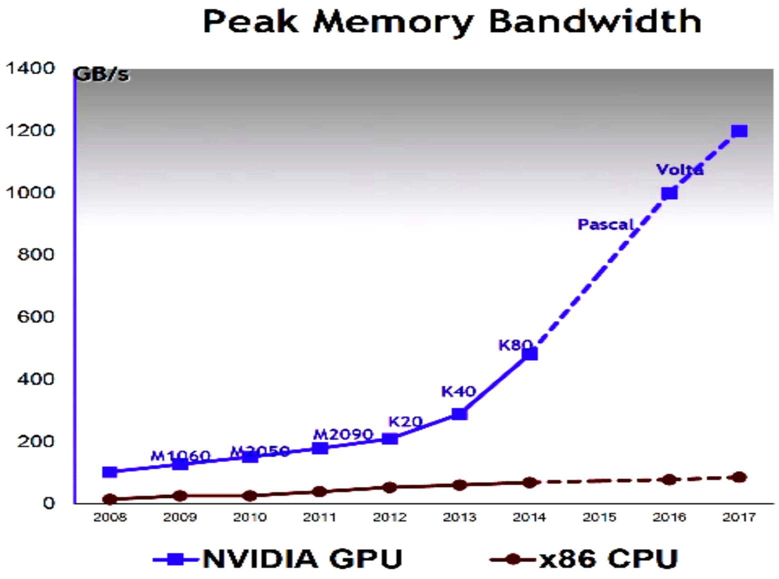
- **Max-Pooling**, Cross-Entropy-Training
- Testfehler: 5,6% (LeNet7: 7.8%)

[Scherer, Müller, Behnke, ICANN'10]

GPU vs. CPU-Performanz

- GPUs eine Größenordnung schneller





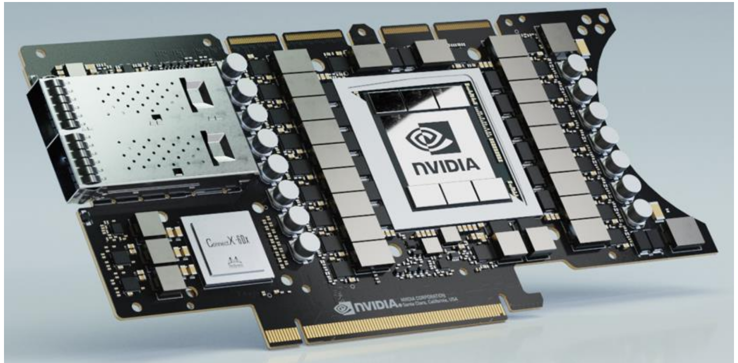
Beispiel: Nvidia A100 Tensore Core GPU

- 9.7 TFLOP/s doppelte Genauigkeit (FP64)
- 19.5 TFLOP/s einfache Genauigkeit (FP32)
- 78 TFLOP/s reduzierte Genauigkeit (FP16)

- 312 Tensor TFLOP/s (Spezialoperationen FP16), 624 Sparse Tensor TFLOP/s

$$\underbrace{D}_{\text{FP16 or FP32}} = \underbrace{\begin{pmatrix} A_{0,0} & A_{0,1} & A_{0,2} & A_{0,3} \\ A_{1,0} & A_{1,1} & A_{1,2} & A_{1,3} \\ A_{2,0} & A_{2,1} & A_{2,2} & A_{2,3} \\ A_{3,0} & A_{3,1} & A_{3,2} & A_{3,3} \end{pmatrix}}_{\text{FP16}} \underbrace{\begin{pmatrix} B_{0,0} & B_{0,1} & B_{0,2} & B_{0,3} \\ B_{1,0} & B_{1,1} & B_{1,2} & B_{1,3} \\ B_{2,0} & B_{2,1} & B_{2,2} & B_{2,3} \\ B_{3,0} & B_{3,1} & B_{3,2} & B_{3,3} \end{pmatrix}}_{\text{FP16}} + \underbrace{\begin{pmatrix} C_{0,0} & C_{0,1} & C_{0,2} & C_{0,3} \\ C_{1,0} & C_{1,1} & C_{1,2} & C_{1,3} \\ C_{2,0} & C_{2,1} & C_{2,2} & C_{2,3} \\ C_{3,0} & C_{3,1} & C_{3,2} & C_{3,3} \end{pmatrix}}_{\text{FP16 or FP32}}$$

- 40 GB HBM2-Speicher mit 1555 GB/sec Bandbreite



Pfingstwoche

- Die Woche nach Pfingsten (2.-5. Juni) **kann** für Lehrveranstaltungen genutzt werden
- Donnerstag, 11. Juni ist ein Feiertag (Fronleichnam)
- **Nächste Vorlesung am 4. Juni oder am 18. Juni?**

- **Die Abstimmung hat mit mehr als 2/3 Mehrheit ergeben:**
  Die nächste Vorlesung findet am 4. Juni statt.