Convolutional Neural Networks (CNN)

Kognitiv teknologi och artificiell intelligens
729G83

LINKÖPINGS UNIVERSITET

Översikt

- Visuell perception i mänskliga hjärnan
  - Stegvis bearbetning
- CNN (Convolutional Neural Networks)
  - Convolution
  - Channels och feature maps
  - Pooling
  - Softmax

Inneboende svårigheter

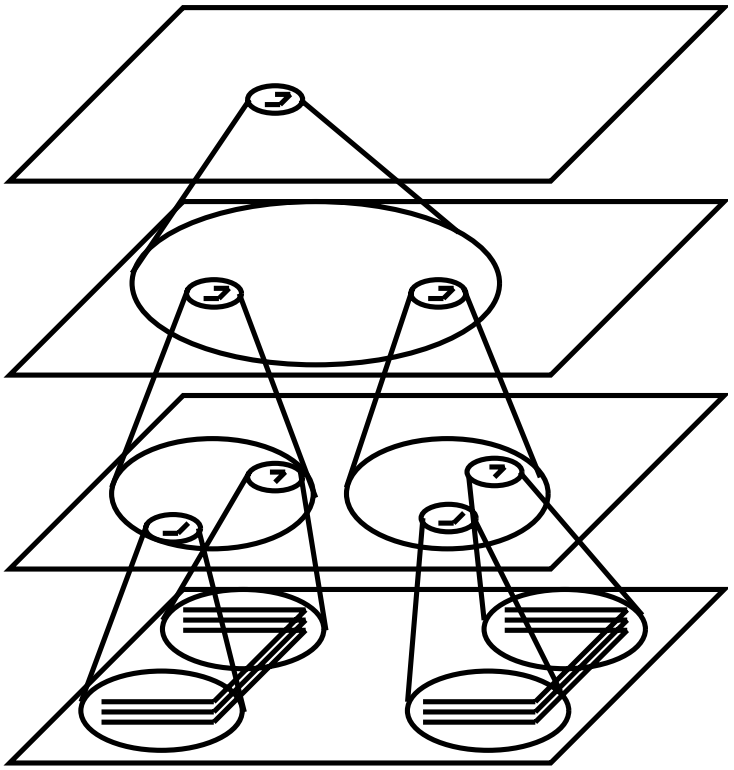
- Matchar inputmallen katt?
  - Måste zooma i alla möjliga skalor
  - Måste rotera i alla möjliga vinklar—och i tre plan
  - Förskjuta så att ”grejen” hamnar på samma ställe som kattmallen
- Objektigenkänning måste vara
  - Storleks-, rotations-, och positionsoberoende (size, rotation, and position invariant)

Lösningen som människan tillämpar

- Typisk ”ful”-lösning
  - Viss tolerans i matchning
  - Lagra flera mallar (för olika rotationer)
    - Matcha mot alla parallellt för att se vilken som matchar bäst

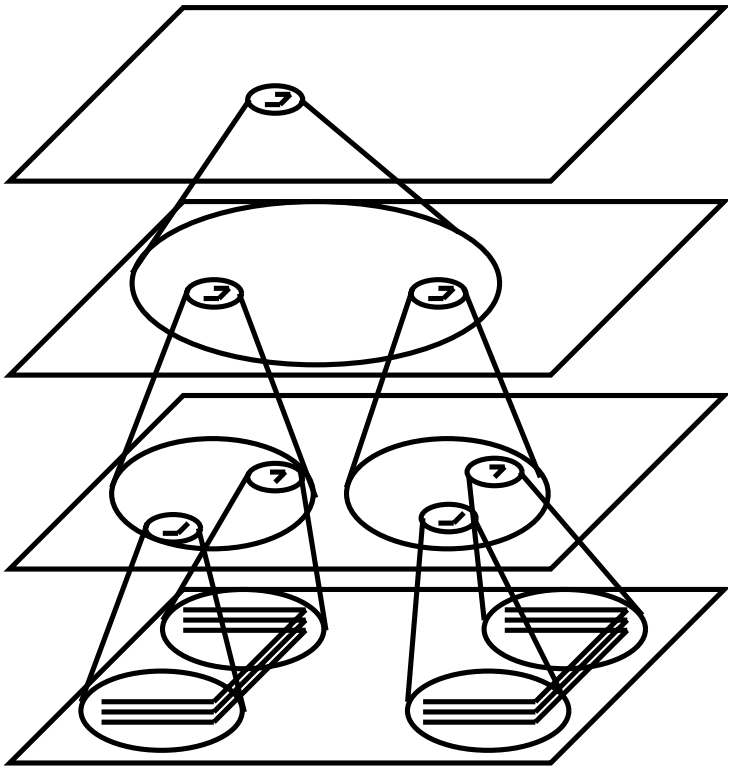
Modell av visuell perception i emergent

- ”Mallar” byggs upp
  - Börjar med enkla linjer, färgfläckar
  - Kombineras *stegvis* till mer komplexa särdrag
  - Slutligen objekt



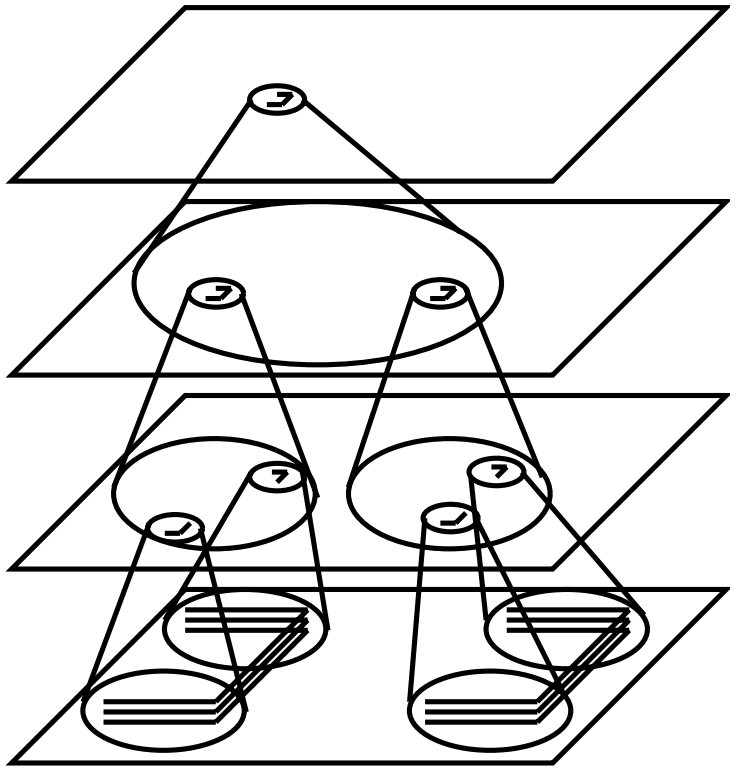
Modell av visuell perception i emergent

- Varje neuron har ett Receptive Field (RF)
- Inom RF
  - Komb. särdrag
  - Relativ position mellan särdrag



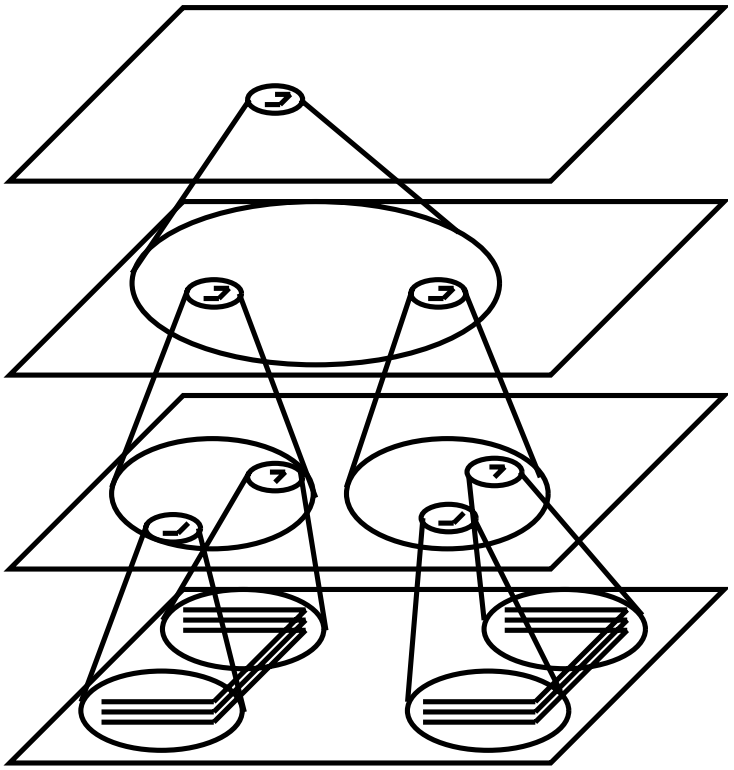
Modell av visuell perception i emergent

- Positionsinvarians
  - Små upptagningsområden (receptive field)
  - Stegvis större
  - Stegvis bättre pos.invarians
    - Storleks-invarians

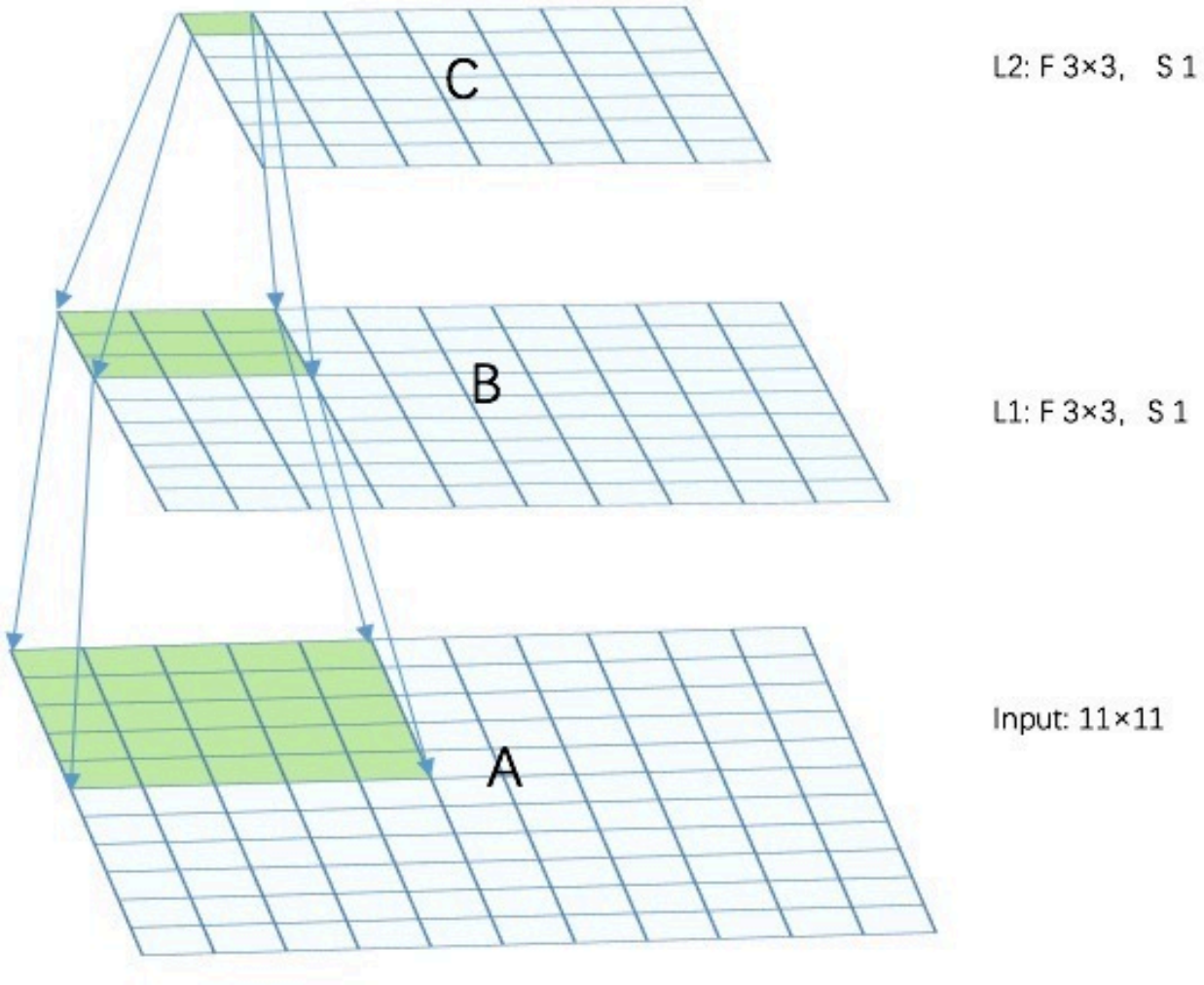


Större och större *indirekt* RF

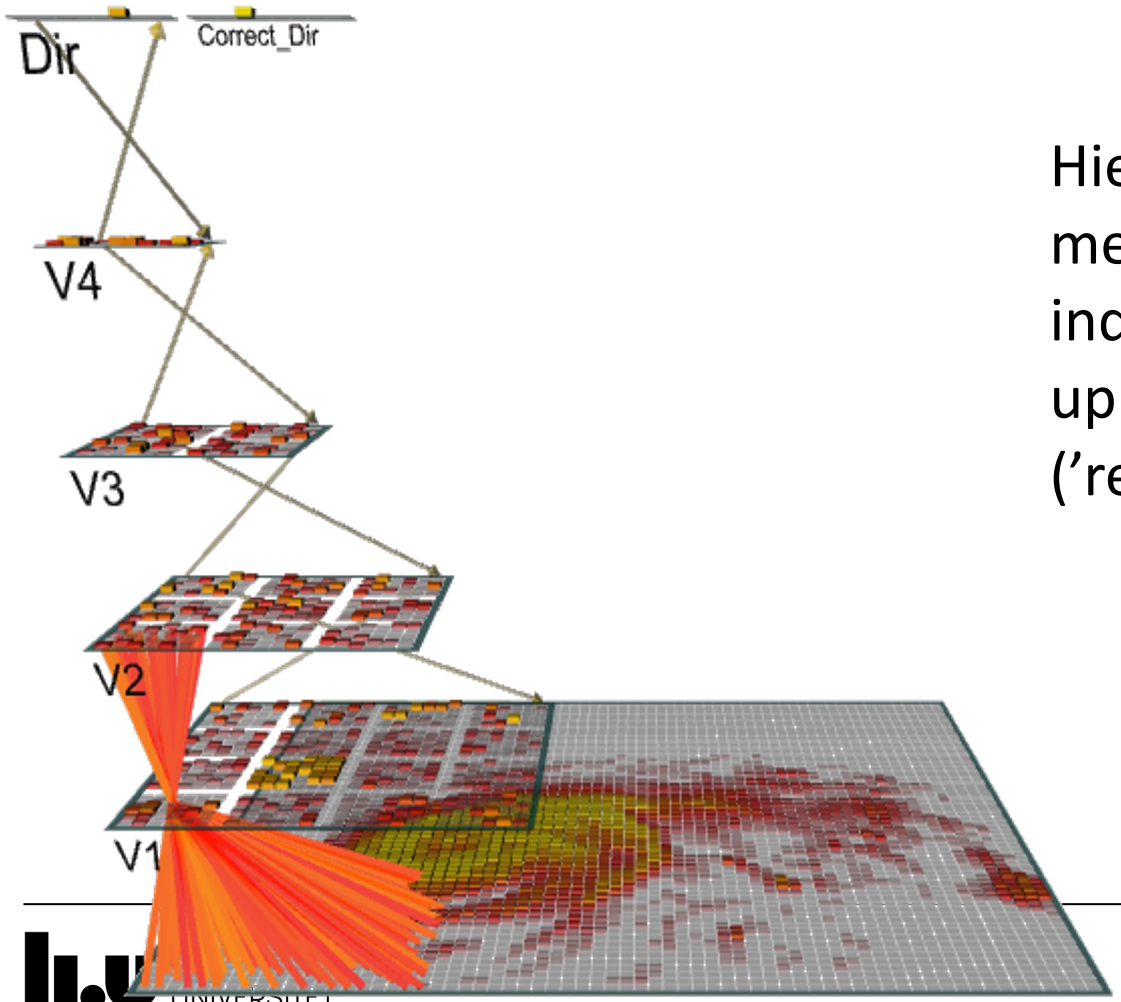
- Neuroner högre upp i nätet tittar på större delar av bilden



Större och större *indirekt* RF



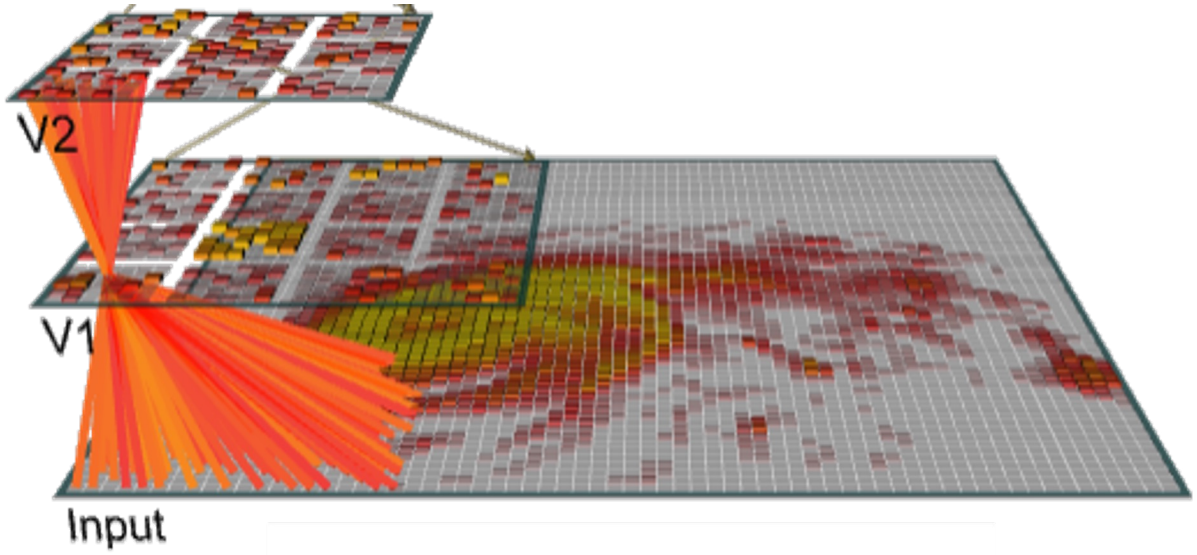
Exempel på nät i emergent



Hierarki av neuroner med allt större indirekt (’effective’) upptagningsområde (’receptive field’)

Grupper = feature djup

- Grupper av 3x3 noder samarbetar
  - Tittar på samma RF
  - Kodar för olika särdrag



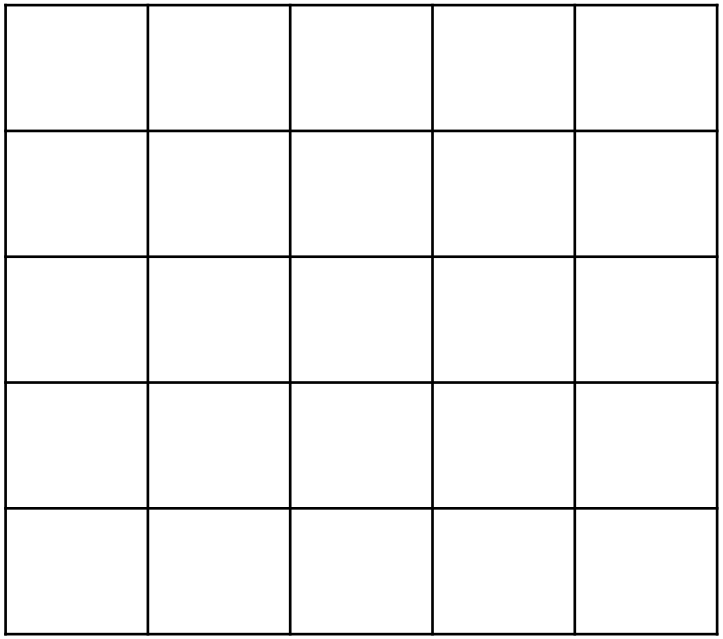
Sammanfattningsvis

- Kombinera särdrag till mer komplexa särdrag
- Mottagaren tar hänsyn till:
  - Angränsande features
  - Hur dessa features ligger i förhållande till varandra
- Ovan gäller för storlek och position
- Rotation hanteras genom att flera mallar lärs in för varje objekt/klass

AI-nät

Convolution (= ”RF”)

- Flytta en viktmatris (filter, kernel) över en inputmatris



1	1.2	0.9
-0.3	-0.1	0
0.2	0.1	-0.2

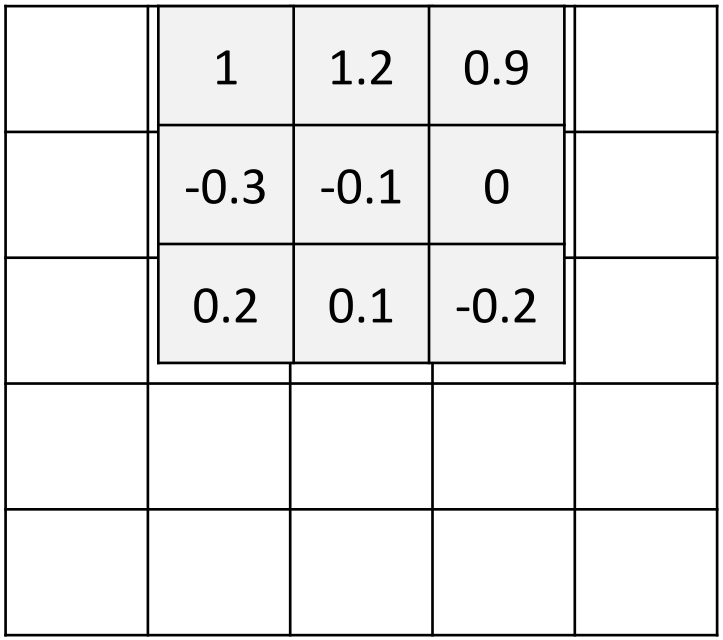
Convolution

- Multiplicera och summera vikt och input i varje position (precis om vanligt), $y = wx + b$

1	1.2	0.9		
-0.3	-0.1	0		
0.2	0.1	-0.2		

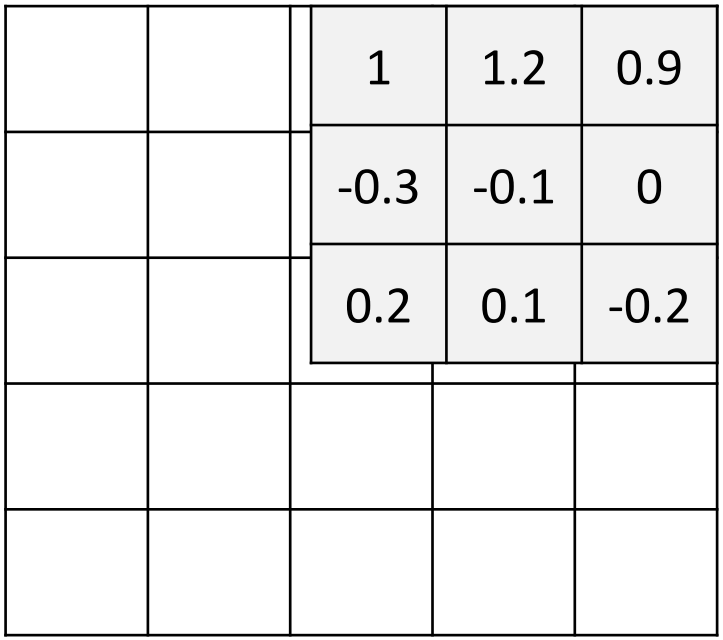
Convolution

- Flytta ett steg, multiplicera och summera



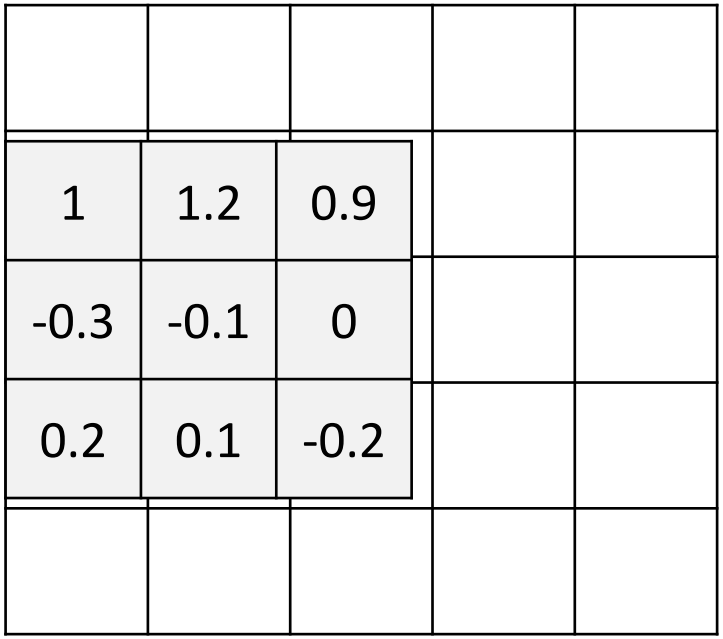
Convolution

- Flytta ett steg



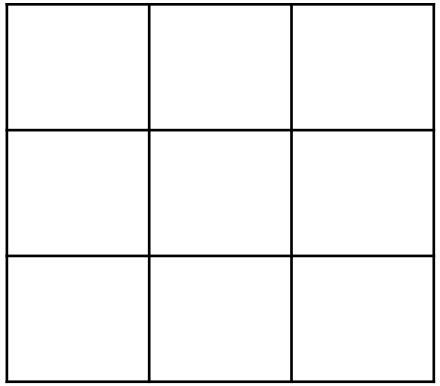
Convolution

- Flytta neråt ett steg, osv.



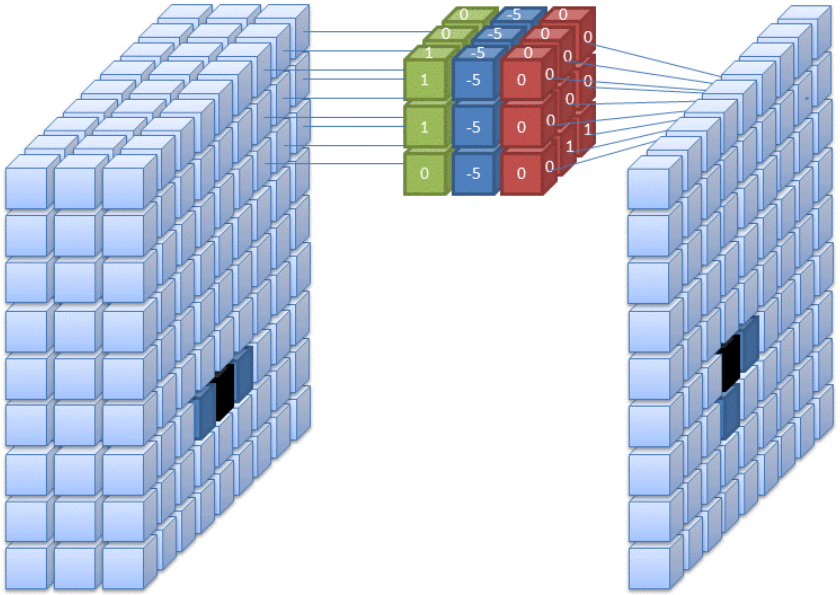
Convolution

- Mottagarlagret har en nod för varje position där filtret användes
- Varje nod kodar för horisontell linje överst i respektive RF
  - (Givet vårt exempel)
- Lagret bildar en feature map (”var förekommer horisontell linje i bilden?”)



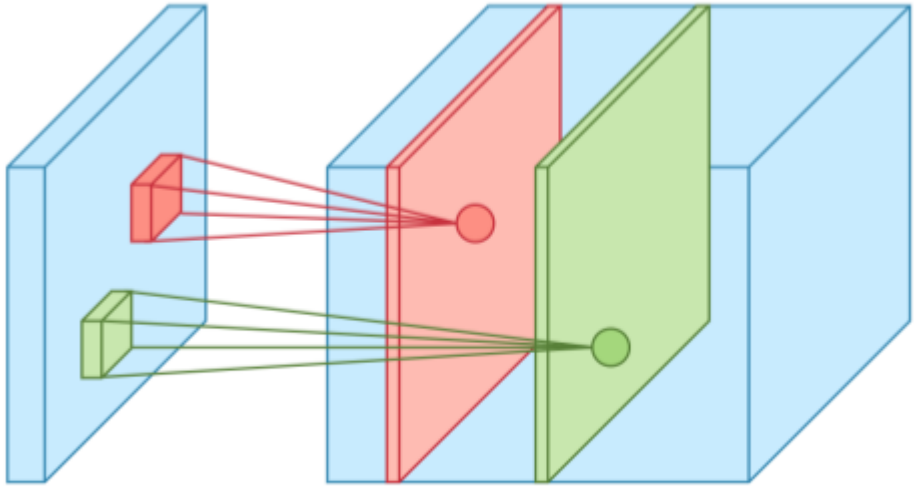
Ofta flera kanaler av input (bilder)

- Ofta flera input-channels
  - T.ex. 3 kanaler för red, green, blue



Vill koda för flera typer av särdrag

- Behöver flera feature maps (activation maps) i mottagarlagret
  - Varje feature map har sitt eget filter (vikter)



4D kernel

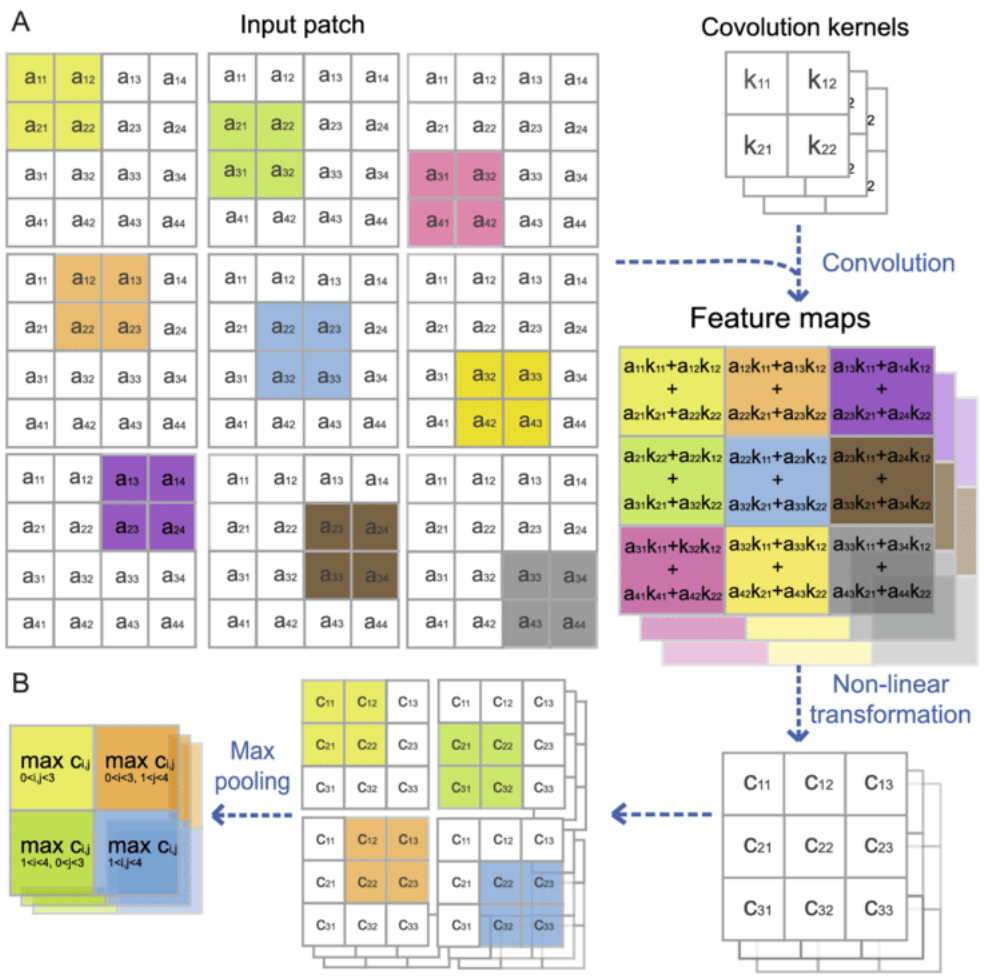
Steg mot objekt

- Varje conv. lager kombinerar enklare särdrag till en uppsättning mer komplexa
  - Kan kombinera olika kanaler el. typer av särdrag inom RF
    - T.ex. hörn och linje om utgör en avgränsad form inom RF

Vill ha positionsinvarians

- Vill inte bara kombinera features, utan även dra ihop angränsande positioner i bilden (”kisa med ögonen”)
  - Bortser var i bilden en feature är
  - Viktiga är ungefärliga relativa positionen till andra features
  - ”Finns det en sned linje någonstans inom RF?”

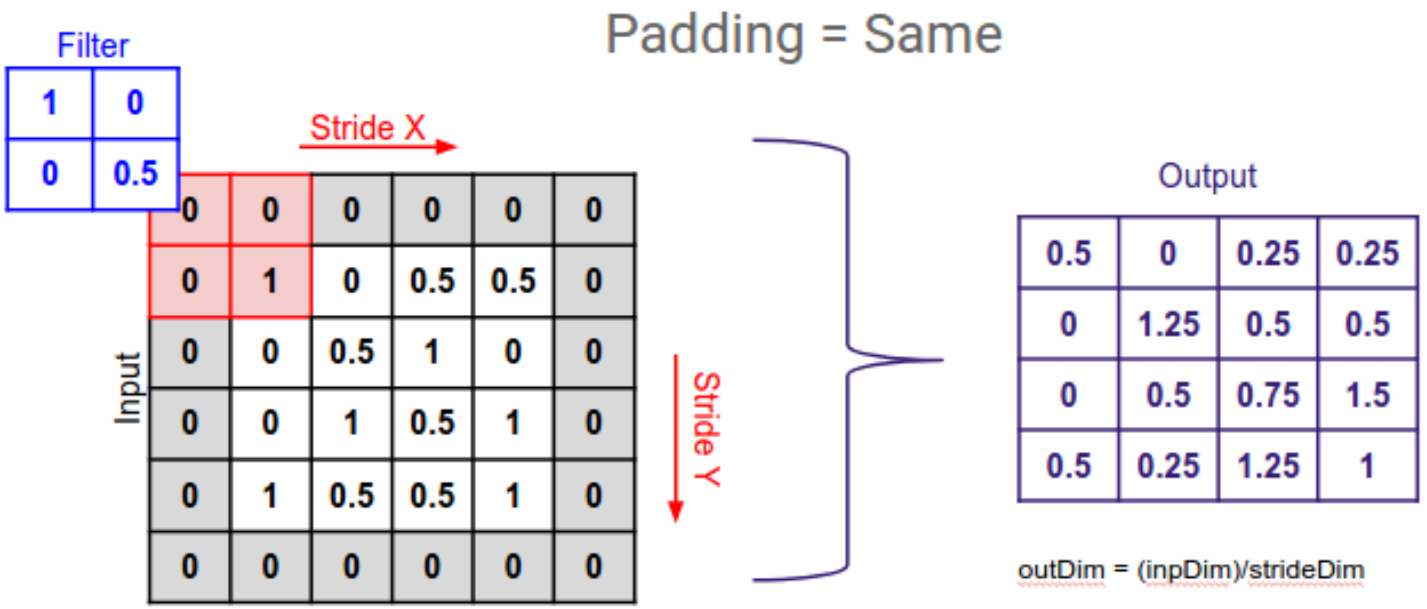
Max pooling


A
Input patch
Covolution kernels
Convolution
Feature maps
Non-linear transformation
B
Max pooling

Pooling

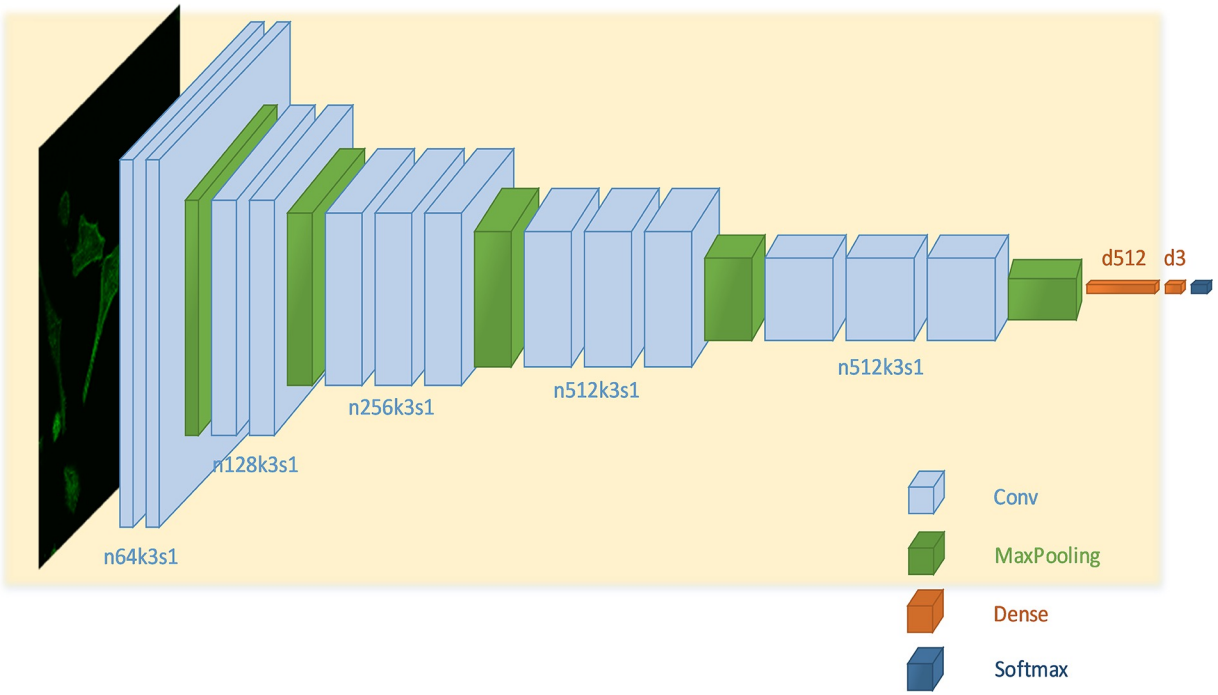
- Max pooling
  - Vad finns det för evidens för särdrag X inom RF?
- Resultatet är en sammanslagning av flera noder till en nod med större (indirekt) RF
  - Feature:n kodas som närvarande om den förekom i någon av de RF som har slagits samman

Annars (oftast) ingen avsmalning av nätverket



Pooling

- Pooling för att få hierarkin av lager att smalna av



Mer avancerade CNN

AlexNet (vann ImageNet LSVRC-2012)

- 5 conv + 3 fully-connected (dense) lager
- 15.3% fel (2:a plats: 26.2% fel)
  - Relu (snabbar upp träningen)
  - Dropout (mindre overfitting)
  - Overlapping pooling (mindre overfitting)
    - Större pooling-fönster ger mer pos.invarians

VGG (Visual Geometry Group, Oxford)

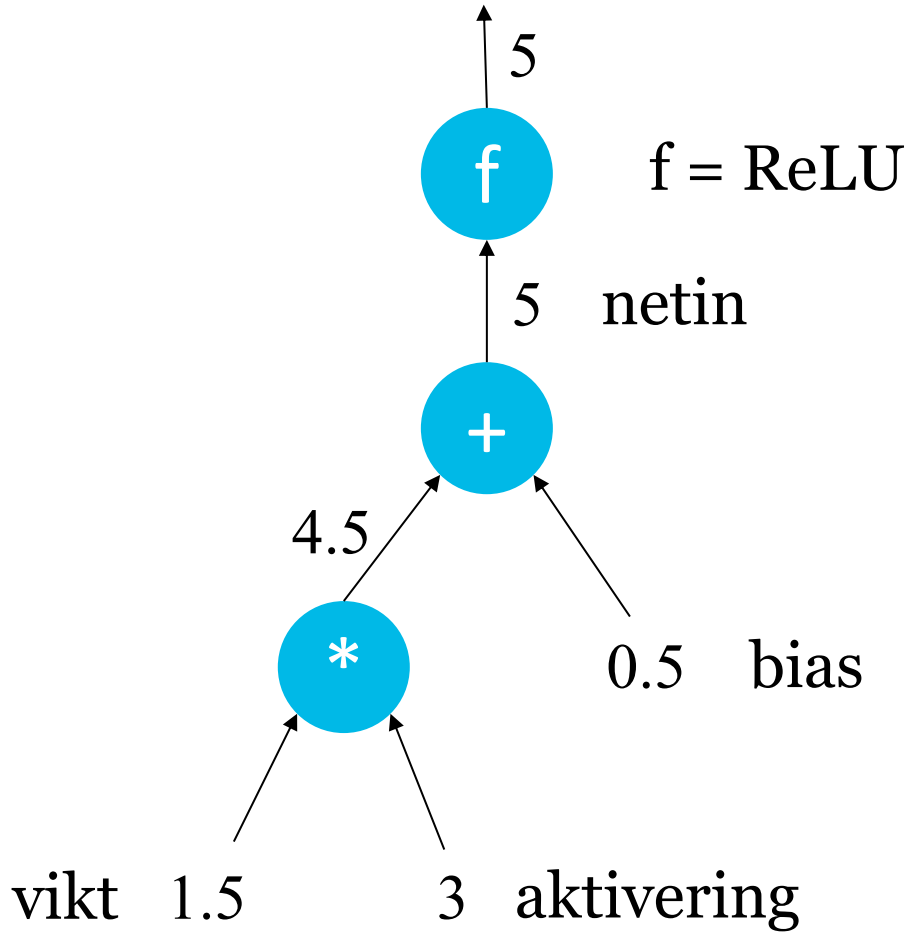
- Ändrade hyperparametrar på AlexNet
  - Mindre filter/kernel-storlek
    - Spar vikter
  - Djupare nät genom att bygga på flera lager
    - Fler lager ger större möjlighet att få fram komplexa features
  - Större feature-djup i lagren, dvs. fler typer av features som kan kännas igen i varje steg
    - Ökad representationsförmåga

GoogLeNet

- Fully conv., dvs. inga FC-lager (fully connected) i toppen
  - 12 ggr färre vikter än AlexNet
- Inception moduler
  - Olika spatiala upplösningar parallellt
  - Fångar fina och grova spatiala mönster samtidigt
  - Tricks med 1x1 conv för att få ner feature-djupet i lagren
- Extra 2 klassificerings-lager för att staga upp inlärningen

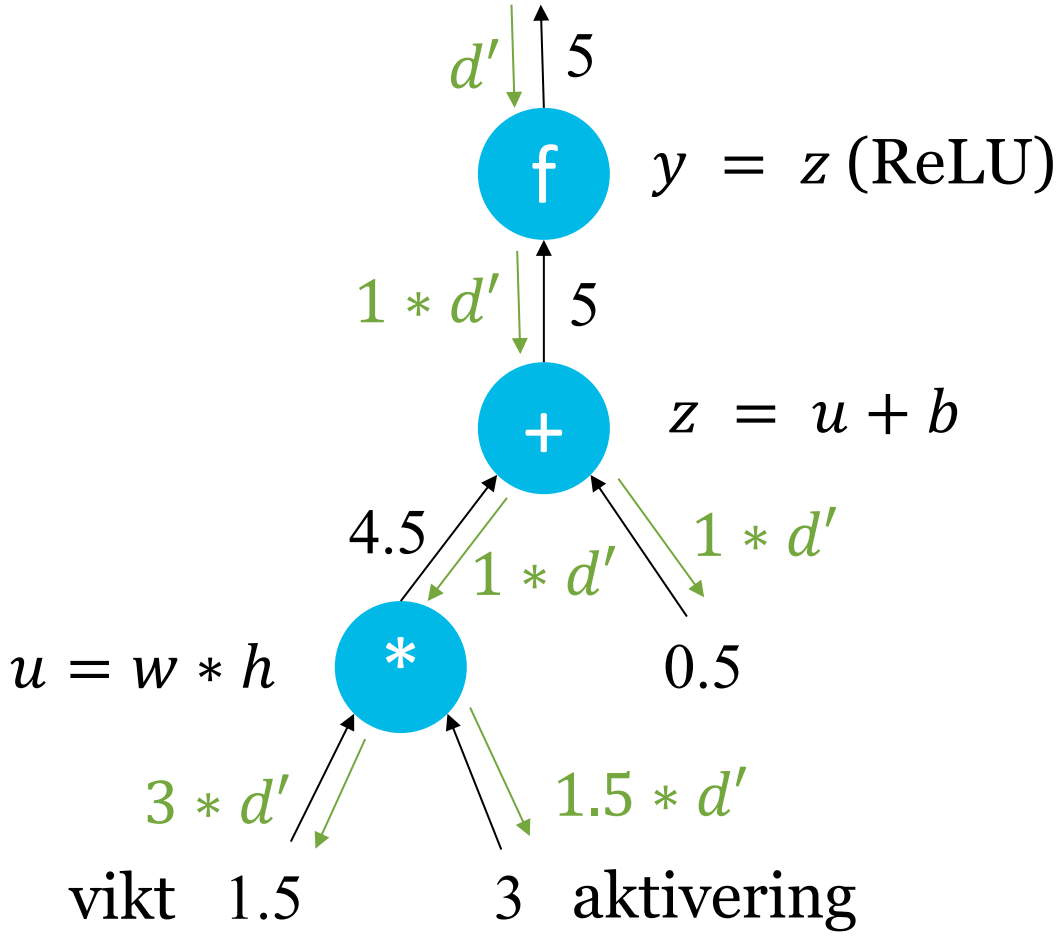
Gradientflöde

- Forward pass
  - Aktiveringar flyter uppåt i nätet



Gradientflöde

- Backward pass
  - $\frac{d}{dx}(c) = 0$
  - $\frac{d}{dx}(x) = 1$
  - $\frac{d}{dx}(x + c) = 1$
  - $\frac{d}{dx}(x * c) = c$



Multiplikativa gates känsliga

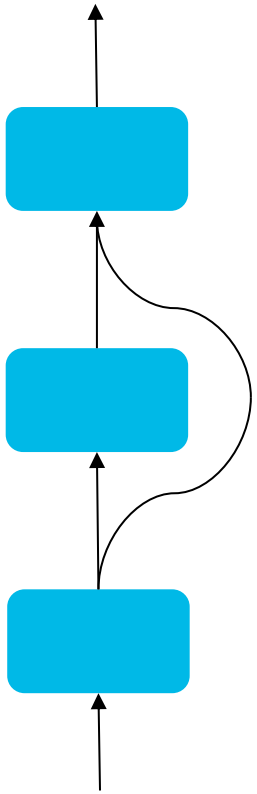
- Multiplikativa gates ($w * h$) gör gradientflödet känsligt för värdet på vikt och aktivering
- Särskilt känsligt i djupa nät
  - Sannolikheten för lågt värde ökar för varje lager (varje multiplikation av gradienten på dess väg neråt i nätet)

Residual Networks (He et al., 2015)

Prof. Tang Xiaoou (left) and his PhD student, Mr. He Kai-Ming, of the Department of Information Engineering, CUHK

ResNet

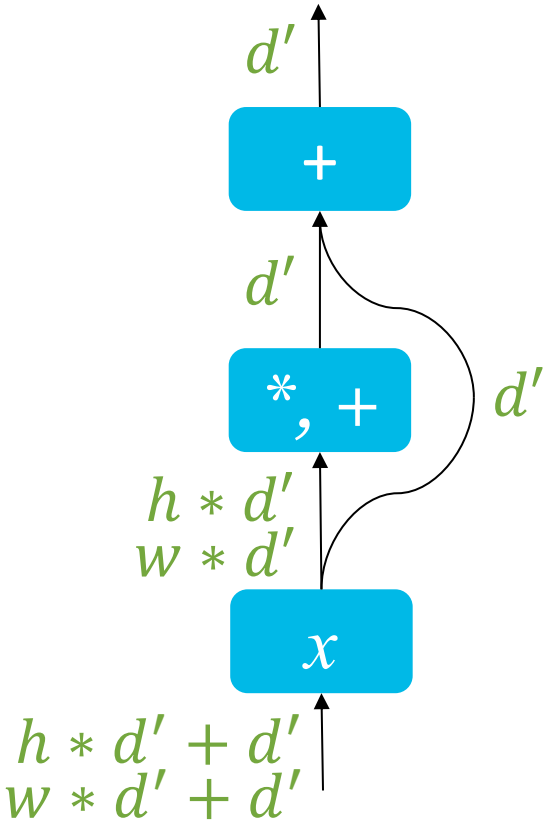
- 152 lager
- Förbikopplingar*
  - Uppdelat
    - Vidarebefordra input
    - ± diff



* Kan även ses i hjärnan!

ResNet

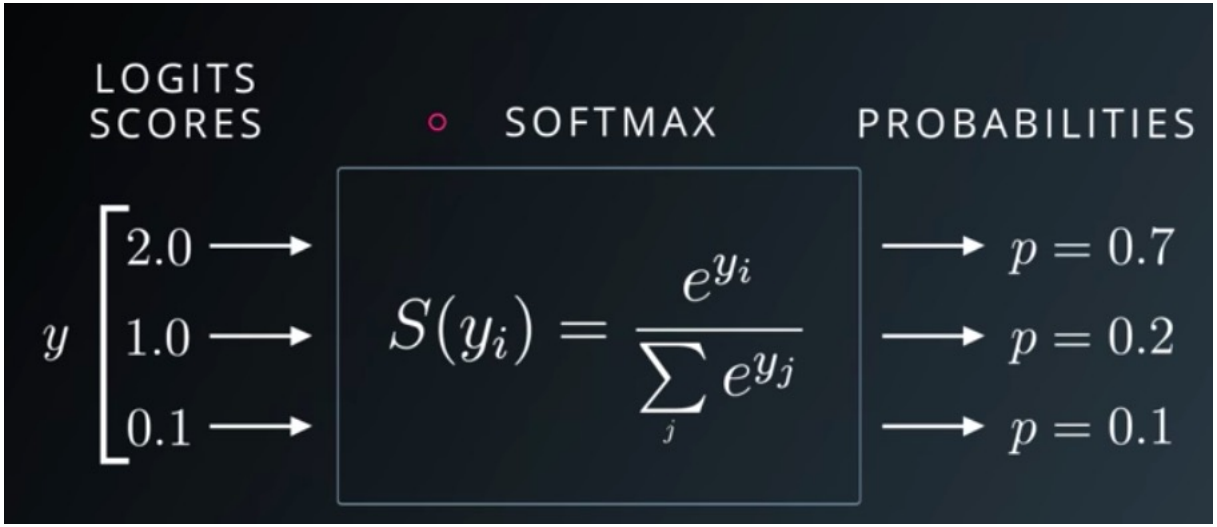
- Förbikoppling
  - Gradienten kan flyta bakåt obehindrat
- Gradienterna läggs ihop där de sammanstrålar i x-noden



Softmax

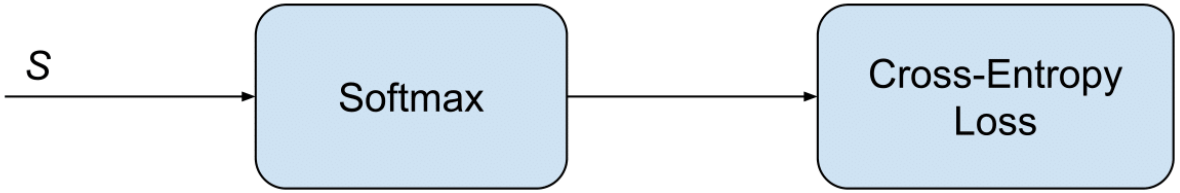
”Softmax-loss”

- Cross entropy loss
  - Förutsätter ett softmax-utlager:



Categorical cross entropy loss (C > 2)

- Mått på överensstämmelse mellan önskad och erhållen sannolikhet, summerat över alla ut-noder



$$f(s)_i = \frac{e^{s_i}}{\sum_j^C e^{s_j}} \qquad CE = -\sum_i^C t_i log(f(s)_i)$$

- Ofta bara p:te elementet i label $t_p = 1$, så bara en term i summan är kvar: $CE = -\log(f(s)_p)$

rita.kovordanyi@liu.se

www.liu.se

LINKÖPINGS UNIVERSITET