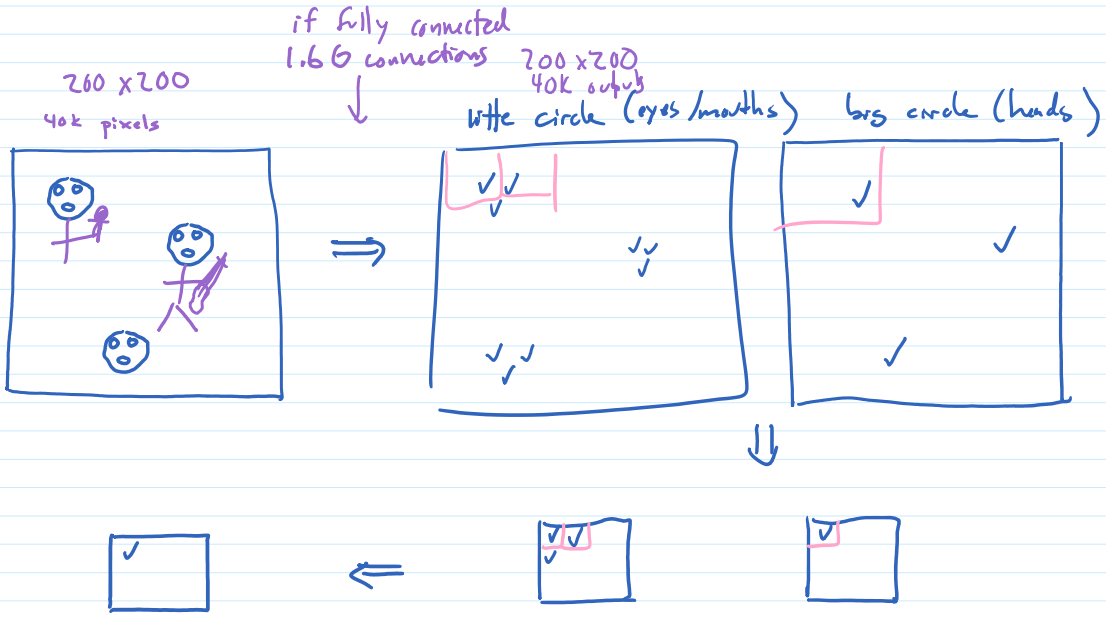
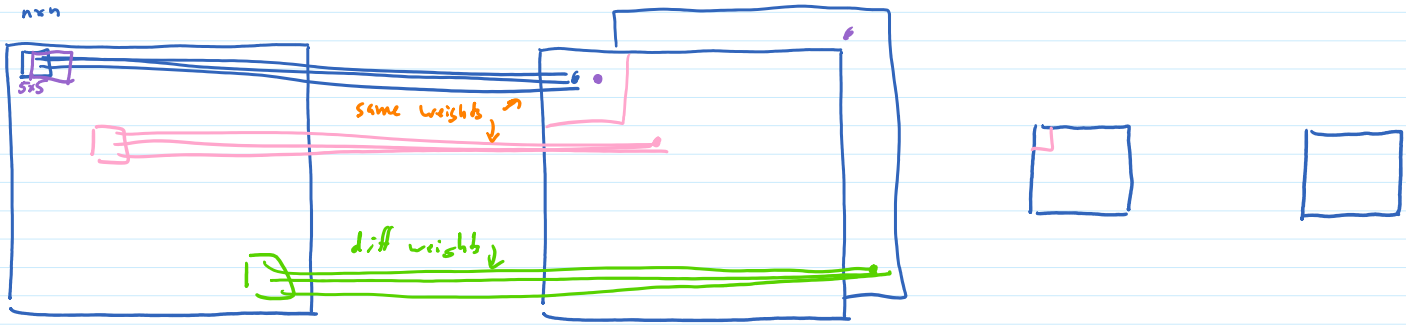
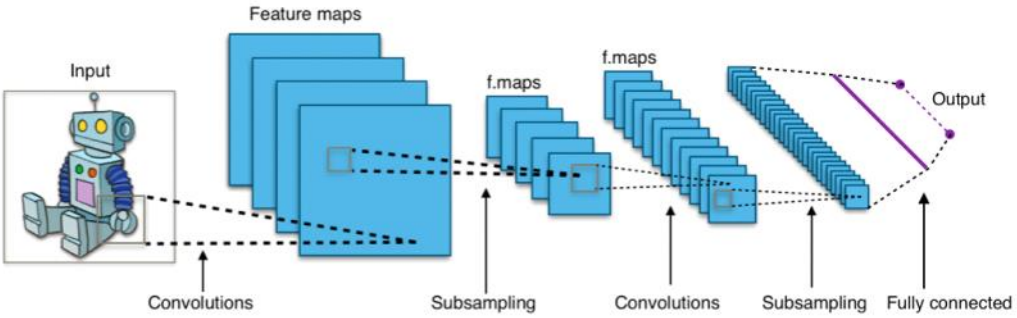
Convolutional Neural Networks





A much better picture from Wikipedia user cpburnz34, who does not endorse these notes.



https://upload.wikimedia.org/wikipedia/commons/6/63/Typical_cnn.png

AlphaGo

policy network - outputs confidence in each possible move

valve network - outputs confidence that position is a winning position

inputs : for each position on board
- black or white 2
- # stones captured
  - opponent 8
  - own 8
- features deemed important by experts
  - liberties 8
  - ladder capture
  - escape
  - ⋮

Step 1: supervised learning for convolutional deep neural network (policy network)

→ match expert play 55% match

3 weeks

+ smaller, faster network

Step 2: reinforcement learning for convolutional deep neural network (policy network)

beats SL net from Step 1 80%

1 day

Step 3: reinforcement learning for valve network

play step 2 RL network against itself 30M times
1 pos from each game

1 week

Step 4: MCTS

5 sec/move

modified UCB $Q(s,a) + c \cdot \underbrace{P(s,a)}_{\text{from SL policy network}} \cdot \frac{\sqrt{\sum_b N(s,b)}}{1+N(s,a)}$

when expand a node, initialize Q using valve network from step 3

playouts use fast step1 policy SL network

AlphaGo Zero — uses no expert knowledge

Inputs 19×19×17 → white/black for current pos + 7 previous
+ all 1's if black's move
all 0's if white's turn

Architecture

Outputs — policy and value