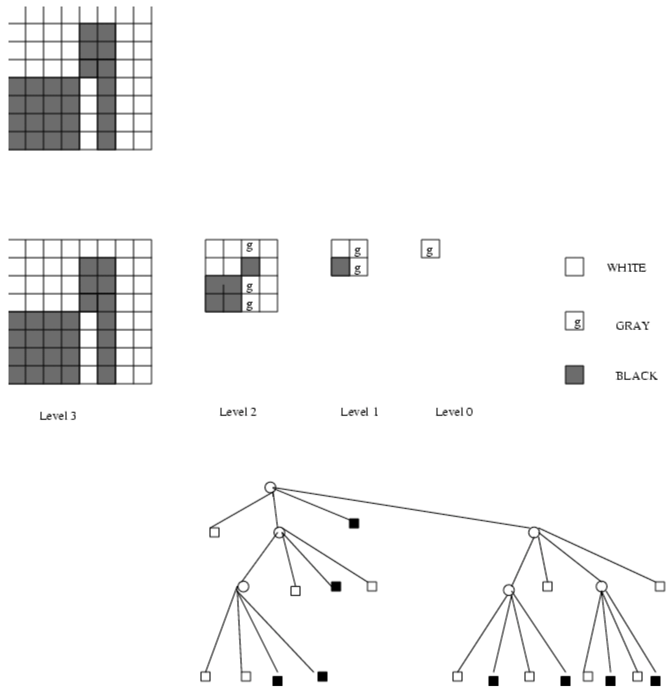
Lecture-15

Quad Trees, Chain Code, Shape number & Moravec's interest operator

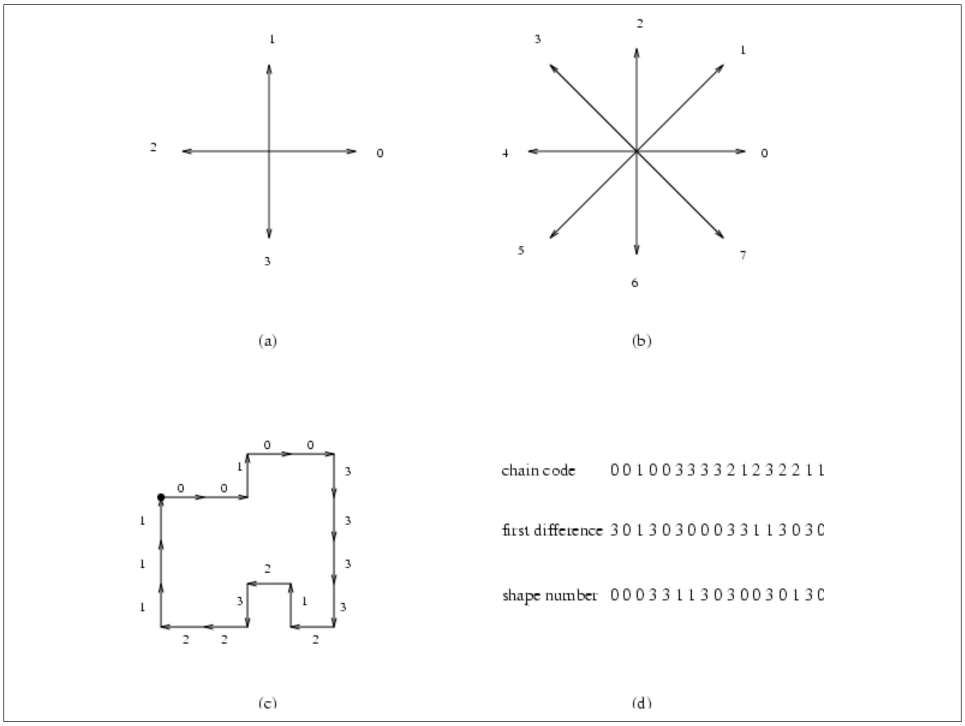
Quad Trees

- Data structure to represent regions
- Three types of nodes: gray, black and white
- First generate the pyramid, then:
- If type of pyramid is black or white then return else
  - Recursively find quad tree of SE quadrant
  - Recursively find quad tree of SW quadrant
  - Recursively find quad tree of NE quadrant
  - Recursively find quad tree of NW quadrant
  - Retrun

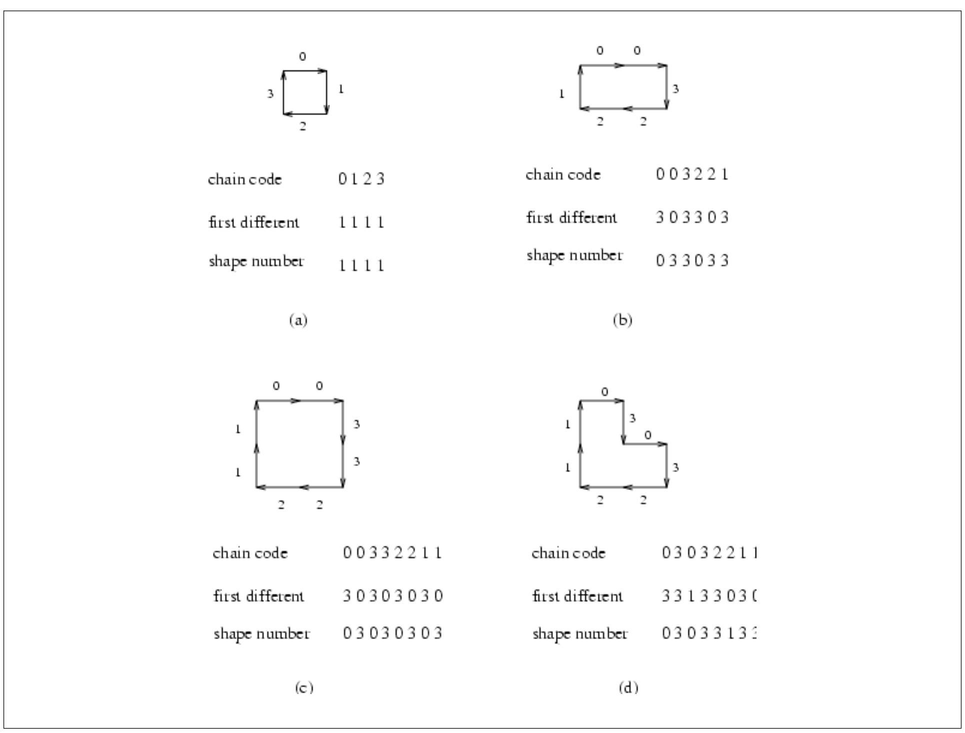


Chain Code

- A simple technique to represent a shape of boundary.
- Each directed line segment is assigned a code.
- Chain code is integer obtained by putting together the codes of all consecutive line segments.
- Shape number is a normalized chain code, which is invariant to translation and rotation.



1
2
0
3
(a)
2
3
1
4
0
5
7
6
(b)
0 0
1
0 0
3
1
3
1
3
1
2
3
1
3
2 2
2
(c)
chain code 0 0 1 0 0 3 3 3 3 2 1 2 3 2 2 1 1
first difference 3 0 1 3 0 3 0 0 0 3 3 1 1 3 0 3 0
shape number 0 0 0 3 3 1 1 3 0 3 0 0 3 0 1 3 0
(d)



0
3
1
2
chain code 0 1 2 3
first different 1 1 1 1
shape number 1 1 1 1
(a)
0 0
1
3
2 2
chain code 0 0 3 2 2 1
first different 3 0 3 3 0 3
shape number 0 3 3 0 3 3
(b)
0 0
1
3
1
3
2 2
chain code 0 0 3 3 2 2 1 1
first different 3 0 3 0 3 0 3 0
shape number 0 3 0 3 0 3 0 3
(c)
0
1
3
0
1
3
2 2
chain code 0 3 0 3 2 2 1 1
first different 3 3 1 3 3 0 3 0
shape number 0 3 0 3 3 1 3 3
(d)

Moravec's Interest Operator

Algorithm

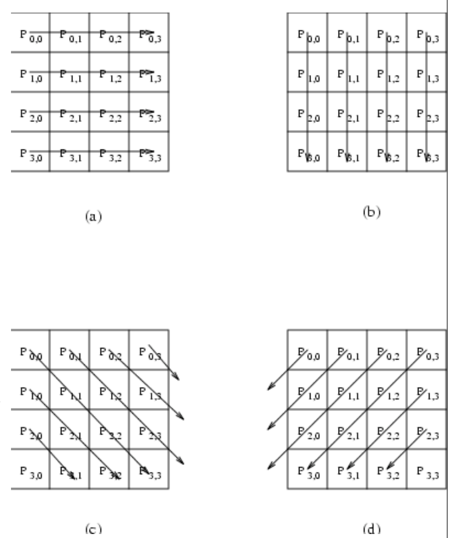
- Compute four directional variances in horizontal, vertical, diagonal and anti-diagonal directions for each 4 by 4 window.
- If the minimum of four directional variances is a local maximum in a 12 by 12 overlapping neighborhood, then that widow (point) is interesting.

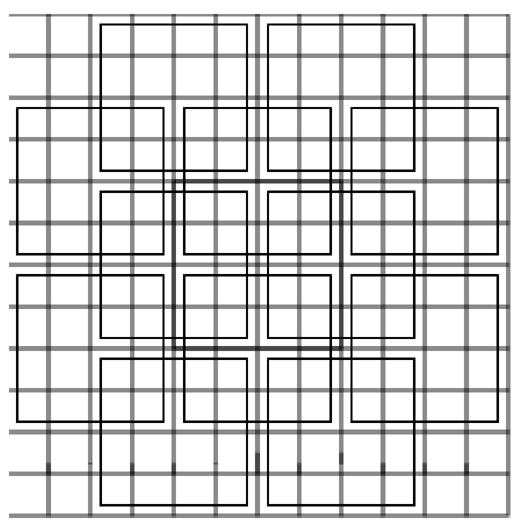
$$V_h = \sum_{j=0}^{3} \sum_{i=0}^{2} (P(x+i, y+j) - P(x+i+1, y+j))^2$$

$$V_v = \sum_{j=0}^{2} \sum_{i=0}^{3} (P(x+i, y+j) - P(x+i, y+j+1))^2$$

$$V_d = \sum_{j=0}^{2} \sum_{i=0}^{2} (P(x+i, y+j) - P(x+i+1, y+j+1))^2$$

$$V_a = \sum_{j=0}^{2} \sum_{i=1}^{3} (P(x+i, y+j) - P(x+i-1, y+j+1))^2$$





$$V_h = \sum_{j=0}^{3} \sum_{i=0}^{2} (P(x+i, y+j) - P(x+i+1, y+j))^2$$

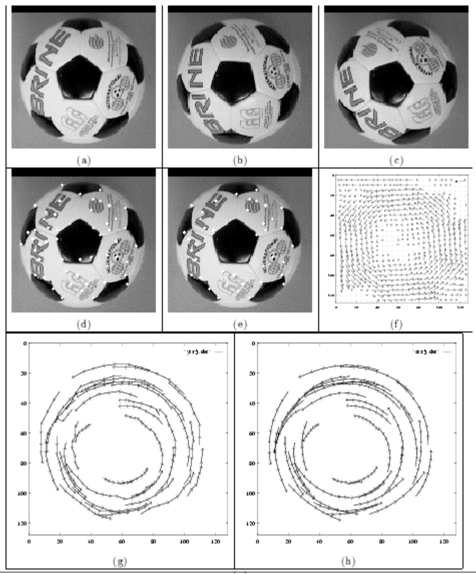
$$V_v = \sum_{j=0}^{2} \sum_{i=0}^{3} (P(x+i, y+j) - P(x+i, y+j+1))^2$$

$$V_d = \sum_{j=0}^{2} \sum_{i=0}^{2} (P(x+i, y+j) - P(x+i+1, y+j+1))^2$$

$$V_a = \sum_{j=0}^{2} \sum_{i=1}^{3} (P(x+i, y+j) - P(x+i-1, y+j+1))^2$$

$$V(x,y) = \min(V_h(x,y), V_v(x,y), V_d(x,y), V_a(x,y))$$

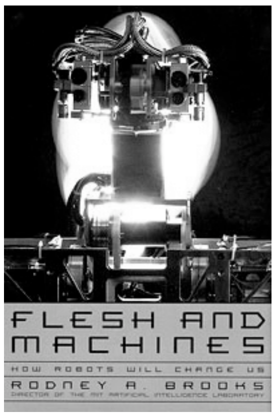
$$I(x,y) = \begin{cases} 1 & if V(x,y) local \max \\ 0 & 0therwise \end{cases}$$



Books by Hans Moravec

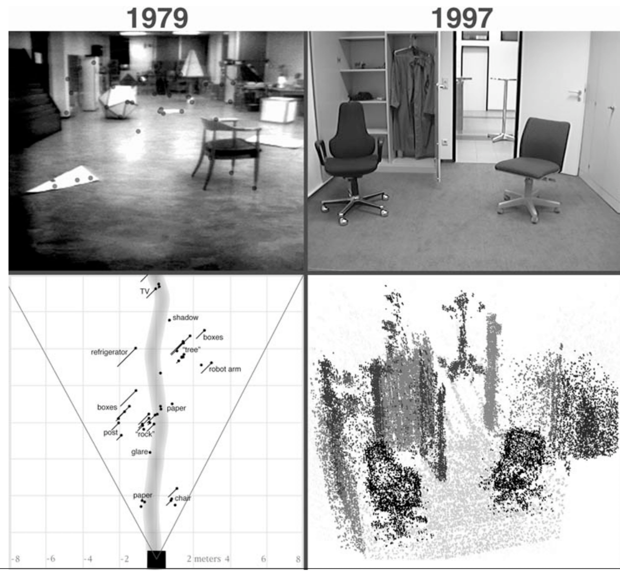
- Robot Rover Visual Navigation
- Mind Children: The future of Robot and Human Intelligence
- Robot, Being
  - Website http://www.frc.ri.cmu.edu/~hpm/

New Book



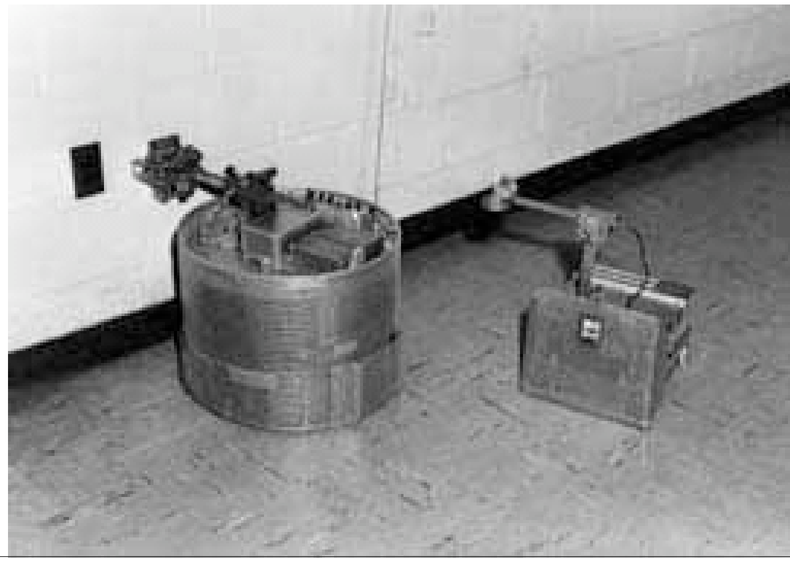
Cart under SAIL

1979 and 1997 3D Maps from stereo

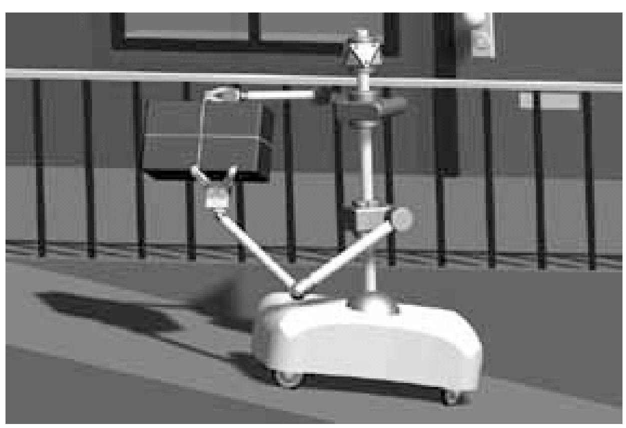


NAVLABS

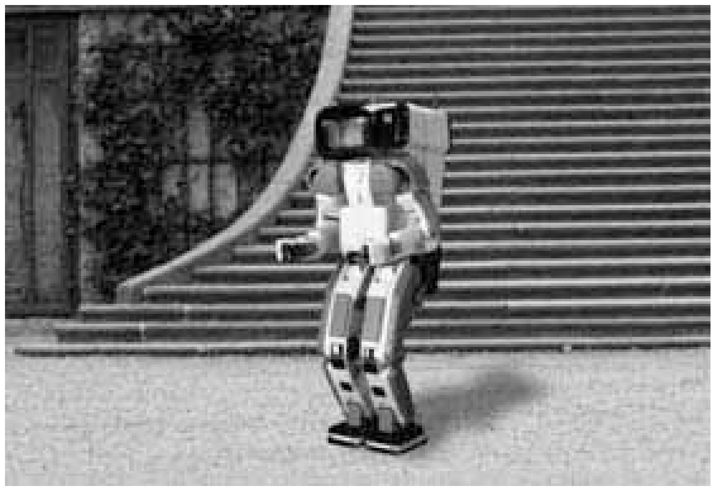
Beast



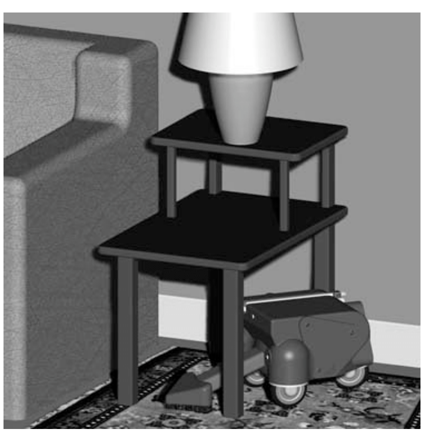
Universal Delivery

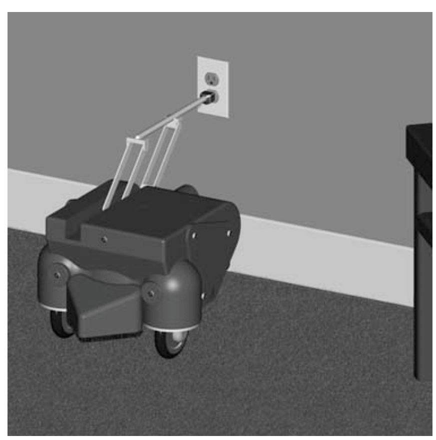


Potsdam

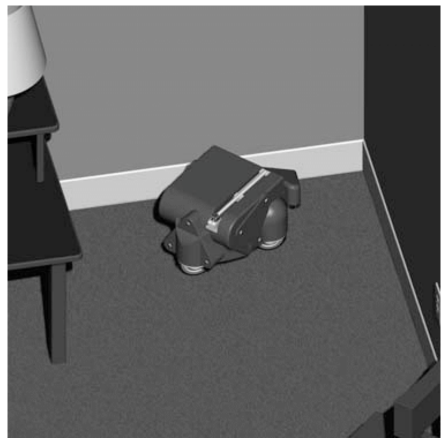


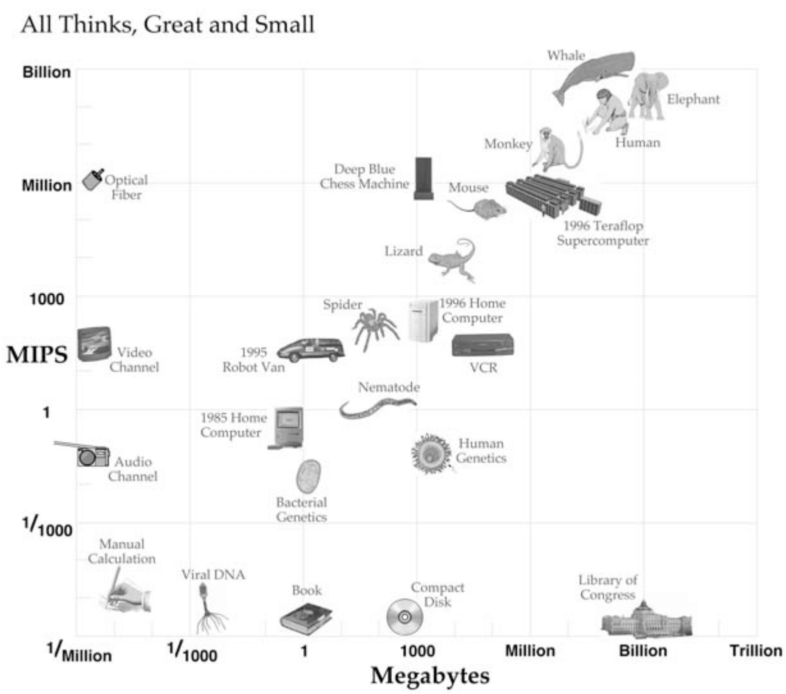
Home Vacuum Cleaning Robot (Dustbot)

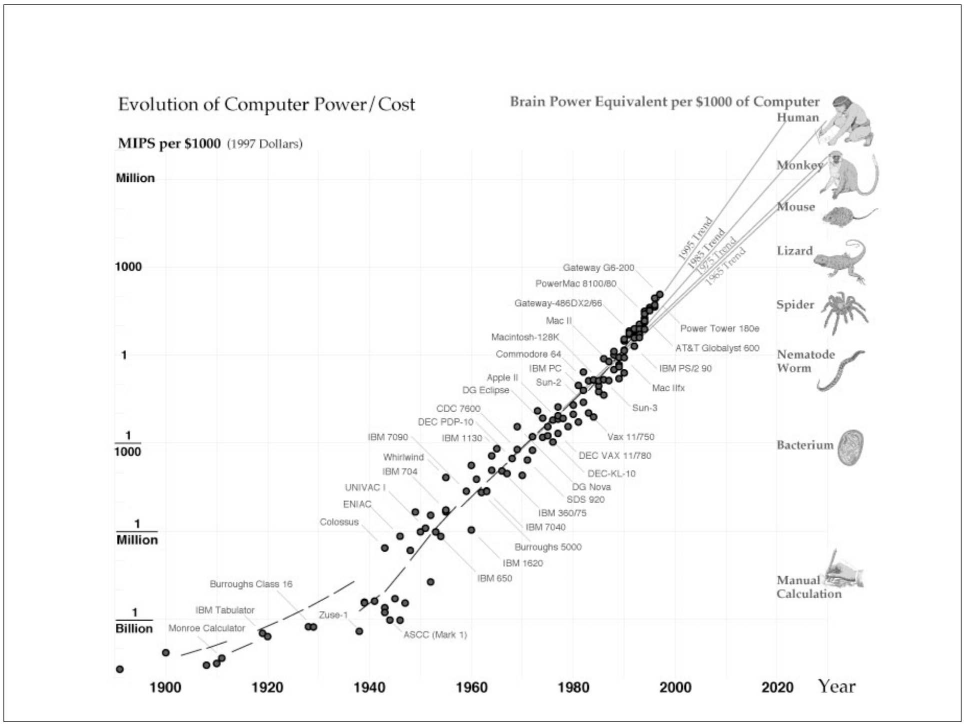




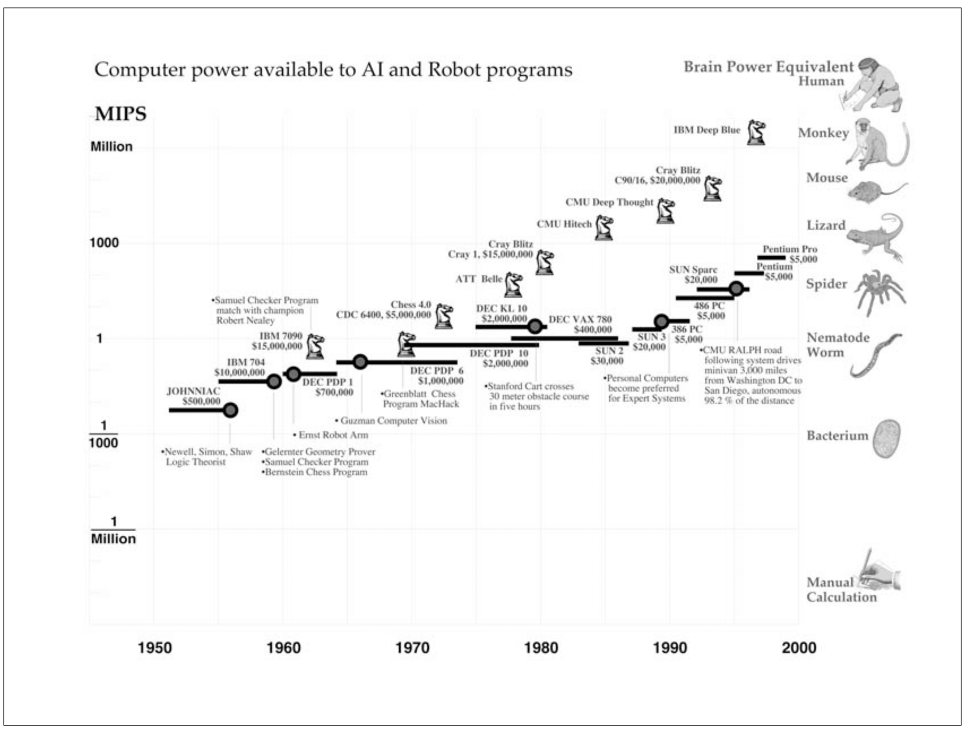
Home Vacuum Cleaning Robot (Dustbot)



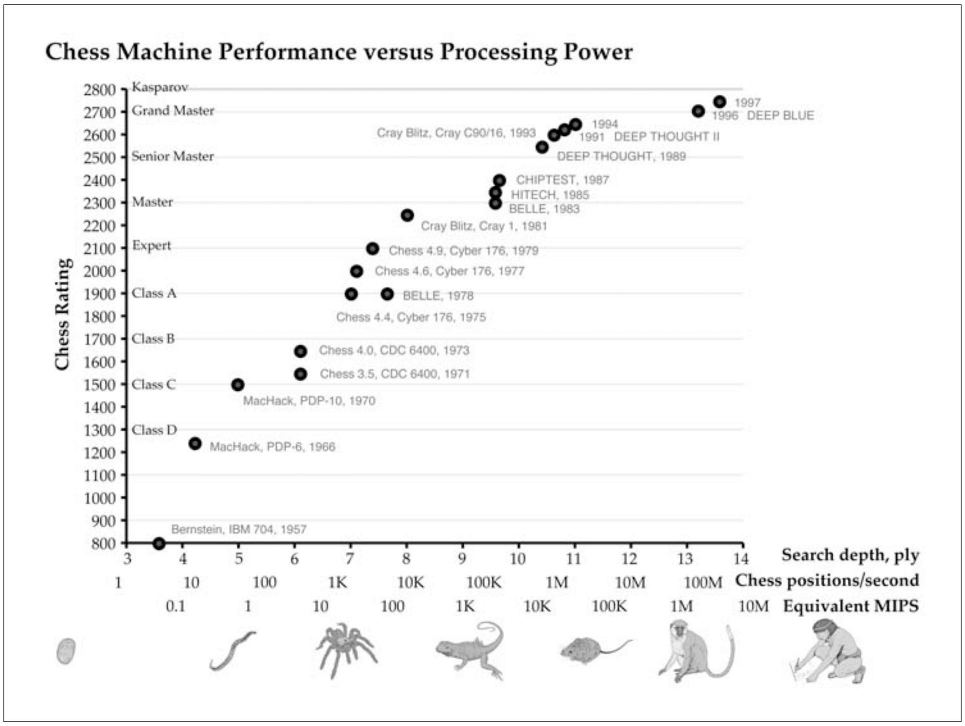




Evolution of Computer Power/Cost
Brain Power Equivalent per $1000 of Computer
MIPS per $1000 (1997 Dollars)
Million
1000
1
1/1000
1/Million
1/Billion
Human
Monkey
Mouse
Lizard
Spider
Nematode Worm
Bacterium
Manual Calculation
1995 Trend
1985 Trend
1975 Trend
1965 Trend
Gateway G6-200
PowerMac 8100/80
Gateway-486DX2/66
Mac II
Macintosh-128K
Commodore 64
IBM PC
Apple II
Sun-2
DG Eclipse
CDC 7600
DEC PDP-10
IBM 7090
IBM 1130
Whirlwind
IBM 704
UNIVAC I
ENIAC
Colossus
Burroughs Class 16
IBM Tabulator
Monroe Calculator
Zuse-1
ASCC (Mark 1)
Power Tower 180e
AT&T Globalyst 600
IBM PS/2 90
Mac IIfx
Sun-3
Vax 11/750
DEC VAX 11/780
DEC-KL-10
DG Nova
SDS 920
IBM 360/75
IBM 7040
Burroughs 5000
IBM 1620
IBM 650
1900
1920
1940
1960
1980
2000
2020
Year



Computer power available to AI and Robot programs
Brain Power Equivalent
MIPS
Million
1000
1
1/1000
1/Million
Human
Monkey
Mouse
Lizard
Spider
Nematode Worm
Bacterium
Manual Calculation
IBM Deep Blue
Cray Blitz C90/16, $20,000,000
CMU Deep Thought
CMU Hitech
Cray Blitz Cray 1, $15,000,000
ATT Belle
Chess 4.0 CDC 6400, $5,000,000
•Samuel Checker Program match with champion Robert Nealey
IBM 7090 $15,000,000
IBM 704 $10,000,000
JOHNNIAC $500,000
DEC PDP 1 $700,000
DEC PDP 6 $1,000,000
DEC KL 10 $2,000,000
DEC VAX 780 $400,000
DEC PDP 10 $2,000,000
SUN 2 $30,000
SUN 3 $20,000
386 PC $5,000
486 PC $5,000
SUN Sparc $20,000
Pentium $5,000
Pentium Pro $5,000
•CMU RALPH road following system drives minivan 3,000 miles from Washington DC to San Diego, autonomous 98.2 % of the distance
•Personal Computers become preferred for Expert Systems
•Stanford Cart crosses 30 meter obstacle course in five hours
•Greenblatt Chess Program MacHack
• Guzman Computer Vision
• Ernst Robot Arm
•Newell, Simon, Shaw Logic Theorist
•Gelernter Geometry Prover
•Samuel Checker Program
•Bernstein Chess Program
1950
1960
1970
1980
1990
2000


Chess Machine Performance versus Processing Power
Chess Rating
Kasparov
Grand Master
Senior Master
Master
Expert
Class A
Class B
Class C
Class D
1997
1996 DEEP BLUE
1994
1991 DEEP THOUGHT II
Cray Blitz, Cray C90/16, 1993
DEEP THOUGHT, 1989
CHIPTEST, 1987
HITECH, 1985
BELLE, 1983
Cray Blitz, Cray 1, 1981
Chess 4.9, Cyber 176, 1979
Chess 4.6, Cyber 176, 1977
BELLE, 1978
Chess 4.4, Cyber 176, 1975
Chess 4.0, CDC 6400, 1973
Chess 3.5, CDC 6400, 1971
MacHack, PDP-10, 1970
MacHack, PDP-6, 1966
Bernstein, IBM 704, 1957
Search depth, ply
Chess positions/second
Equivalent MIPS