The importance of a good representation

- **Properties of a good representation:**

- Reveals important features
- Hides irrelevant detail
- Exposes useful constraints
- Makes frequent operations easy-to-do
- Supports local inferences from local features
  - Called the “soda straw” principle or “locality” principle
  - Inference from features “through a soda straw”
- Rapidly or efficiently computable
  - It’s nice to be fast

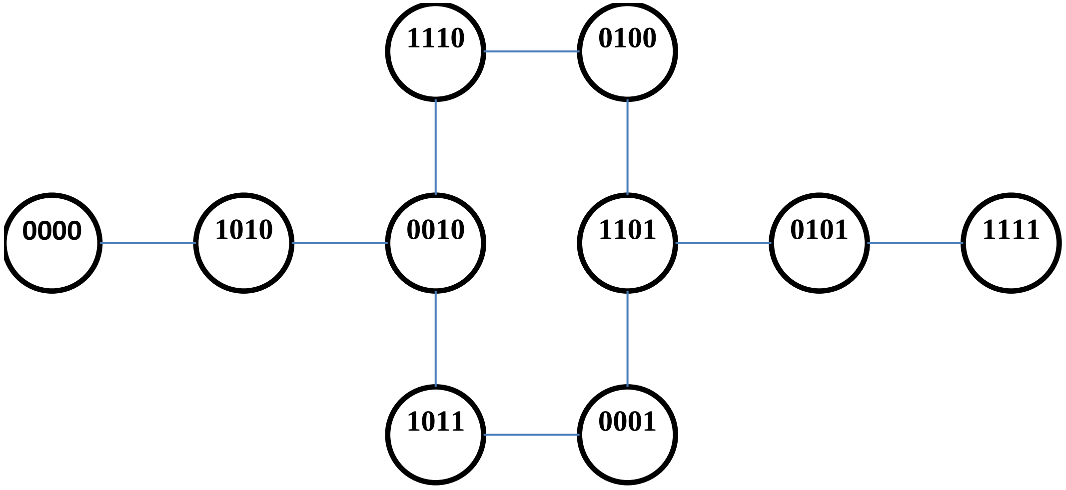
Reveals important features / Hides irrelevant detail

- **"You can't learn what you can't represent."** --- G. Sussman

- **In search:** *A man is traveling to market with a fox, a goose, and a bag of oats. He comes to a river. The only way across the river is a boat that can hold the man and exactly one of the fox, goose or bag of oats. The fox will eat the goose if left alone with it, and the goose will eat the oats if left alone with it.*

  How can the man get all his possessions safely across the river?

- **A good representation makes this problem easy:**



MFGO

M = man
F = fox
G = goose
O = oats
0 = starting side
1 = ending side

Exposes useful constraints

- **"You can't learn what you can't represent."** --- G. Sussman
- **In logic:** *If the unicorn is mythical, then it is immortal, but if it is not mythical, then it is a mortal mammal. If the unicorn is either immortal or a mammal, then it is horned. The unicorn is magical if it is horned.*

  Prove that the unicorn is both magical and horned.
- **A good representation makes this problem easy:**

(¬ Y v ¬ R) ^ (Y v R) ^ (Y v M) ^ (R v H) ^ (¬ M v H) ^ (¬ H v G)

Makes frequent operations easy-to-do

Roman numerals

- M=1000, D=500, C=100, L=50, X=10, V=5, I=1
- 2000 = MM; 1776 = MDCCLXXVI

- Long division is **very tedious** (try MDCCLXXVI / XVI)
- Testing for N < 1000 is very easy (first letter is not “M”)

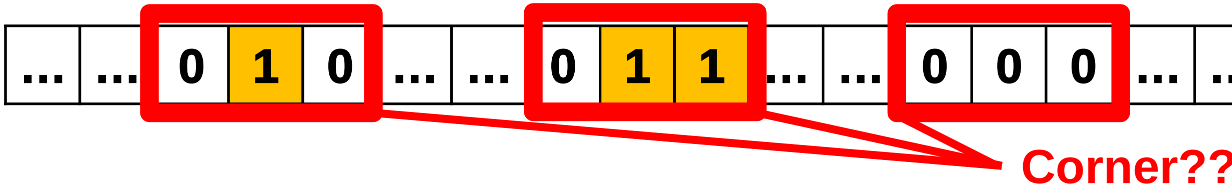
Arabic numerals

- 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, “.”

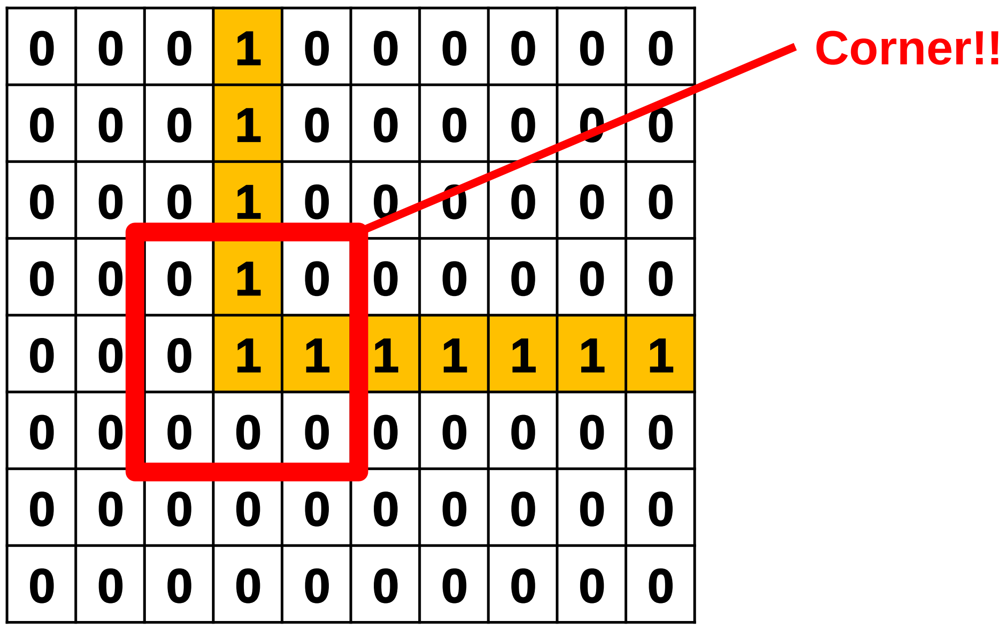
- Long division is **much easier** (try 1776 / 16)
- Testing for N < 1000 is slightly harder (have to scan the string)

Supports local inferences from local features

- Linear vector of pixels = highly non-local inference for vision

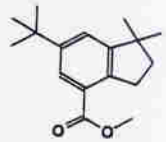


- Rectangular array of pixels = local inference for vision

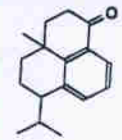


COLLECTION: 026160

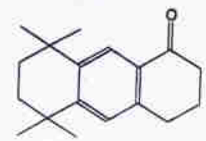
COLLECTION: 019486



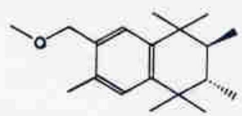
COLLECTION: 004285



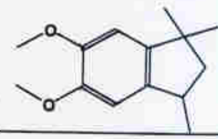
COLLECTION: 019449



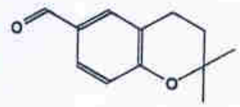
COLLECTION: 025684



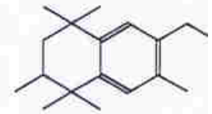
COLLECTION: 021409



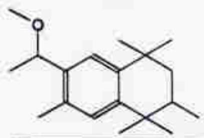
COLLECTION: 021004



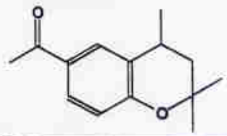
COLLECTION: 019491



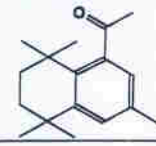
COLLECTION: 025685



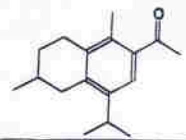
COLLECTION: 020830



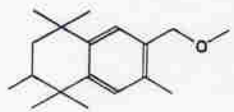
COLLECTION: 021961



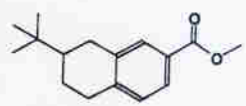
COLLECTION: 012774



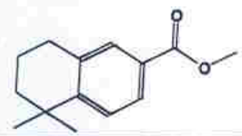
COLLECTION: 019492



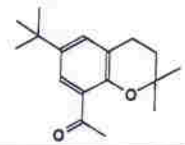
COLLECTION: 022001



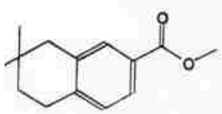
COLLECTION: 021999



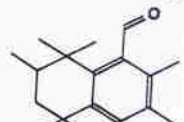
COLLECTION: 019194



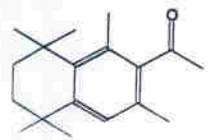
COLLECTION: 022000



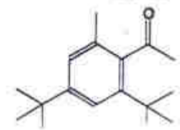
COLLECTION: 021107



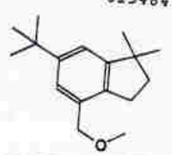
COLLECTION: 021111



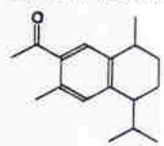
COLLECTION: 019726



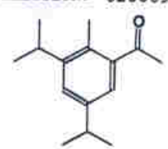
COLLECTION: 019484



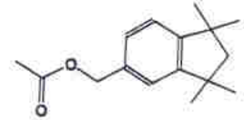
COLLECTION: 012773



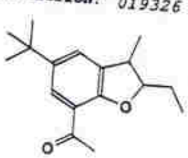
COLLECTION: 020069



COLLECTION: 020074

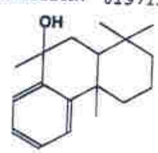


COLLECTION: 019326

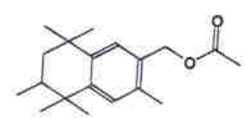


COLLECTION: 020075

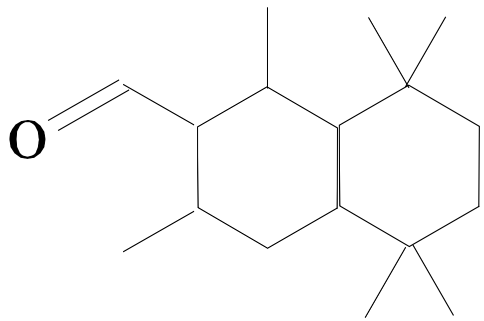
COLLECTION: 019713

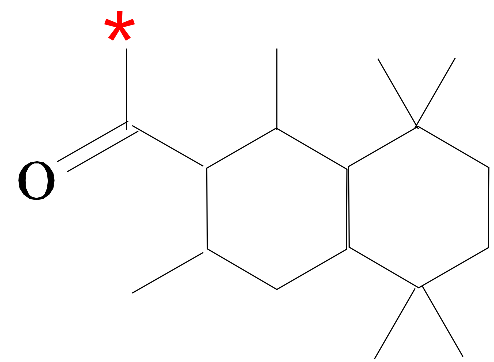


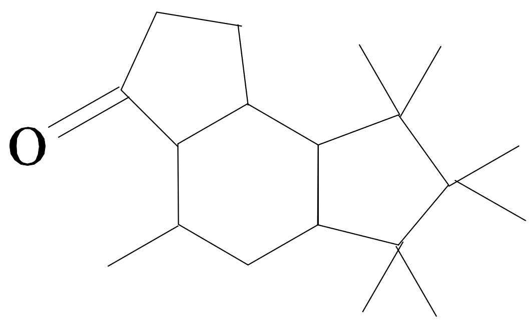
COLLECTION: 019494

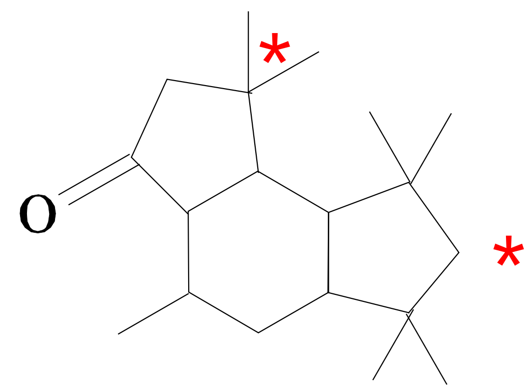


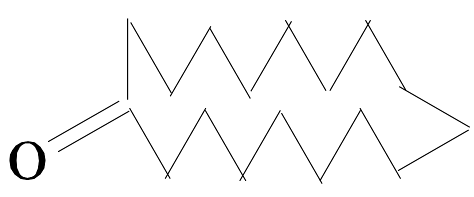
Positive Examples | Negative Examples



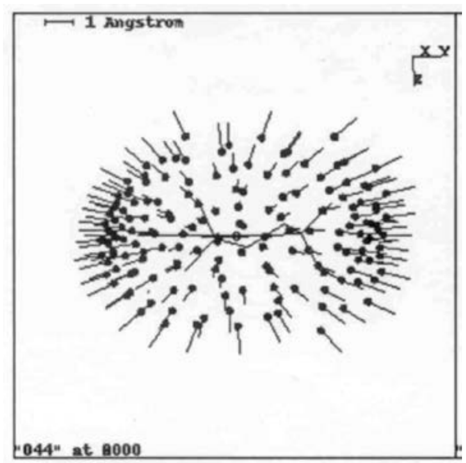


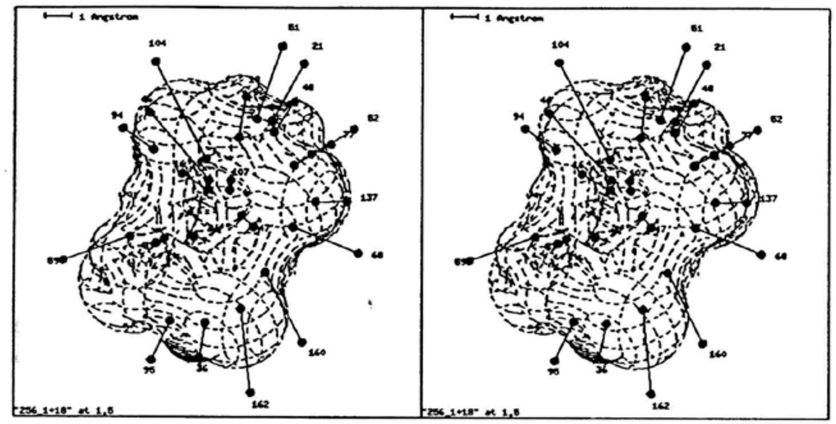


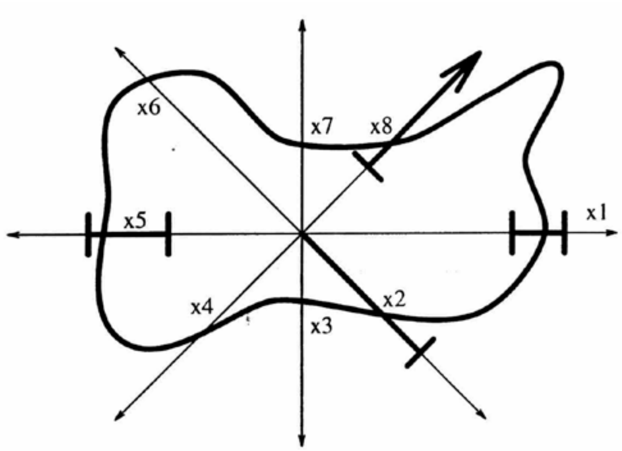


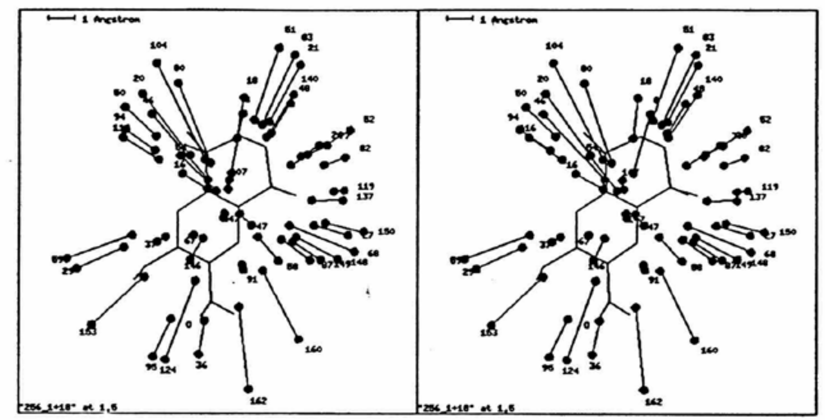


Digital 3D Shape Representation

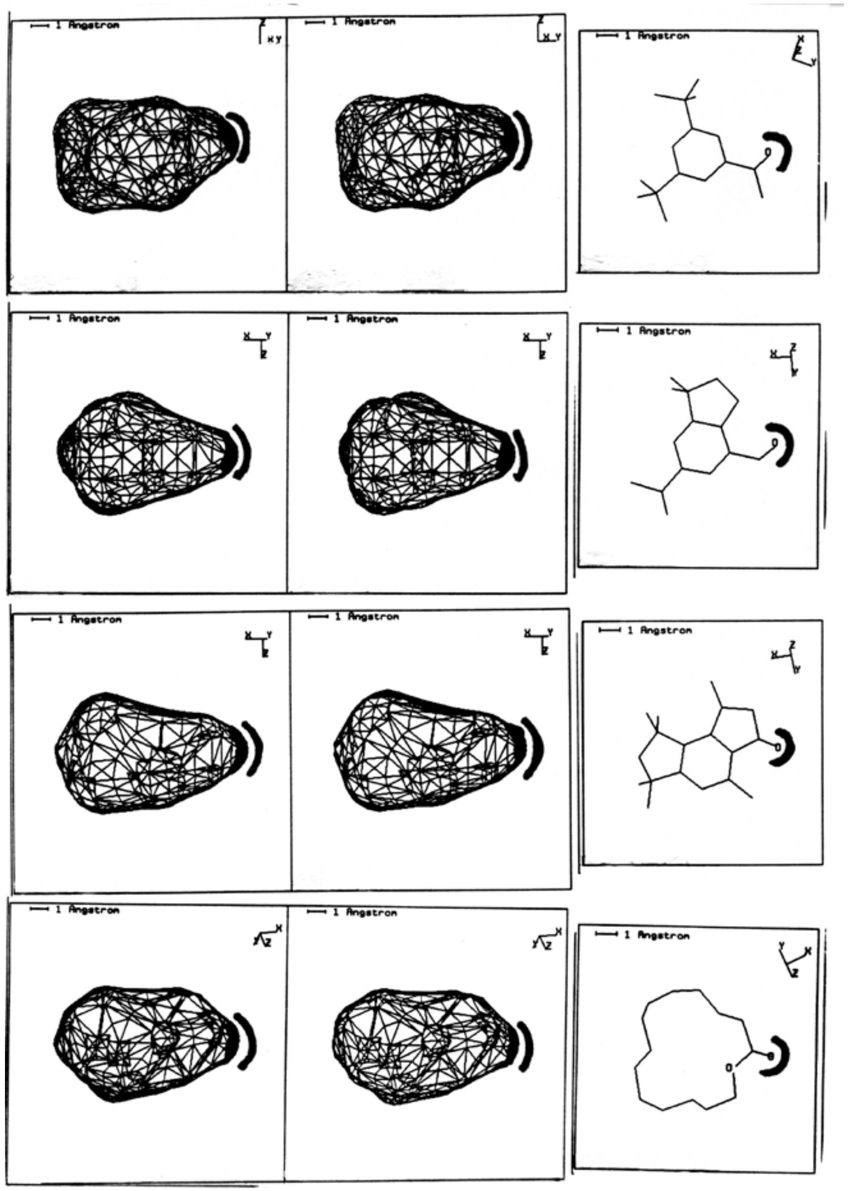




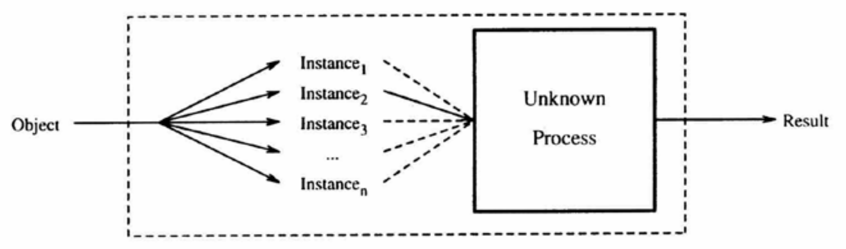


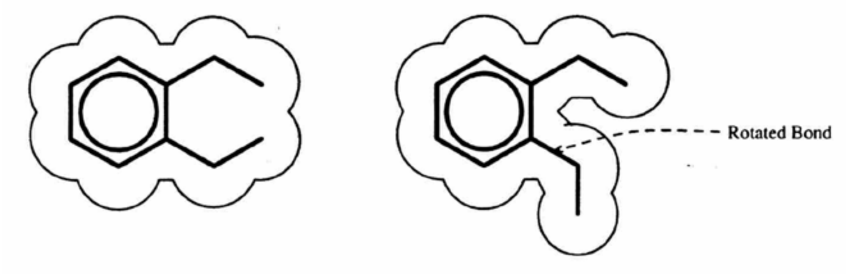


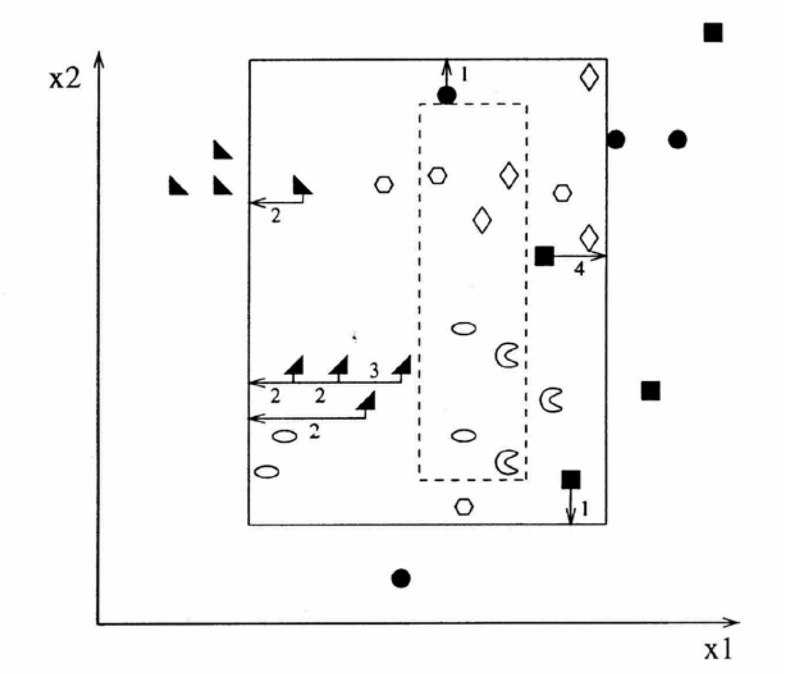
The Power of a Good Representation



Learning the “Multiple Instance” Problem

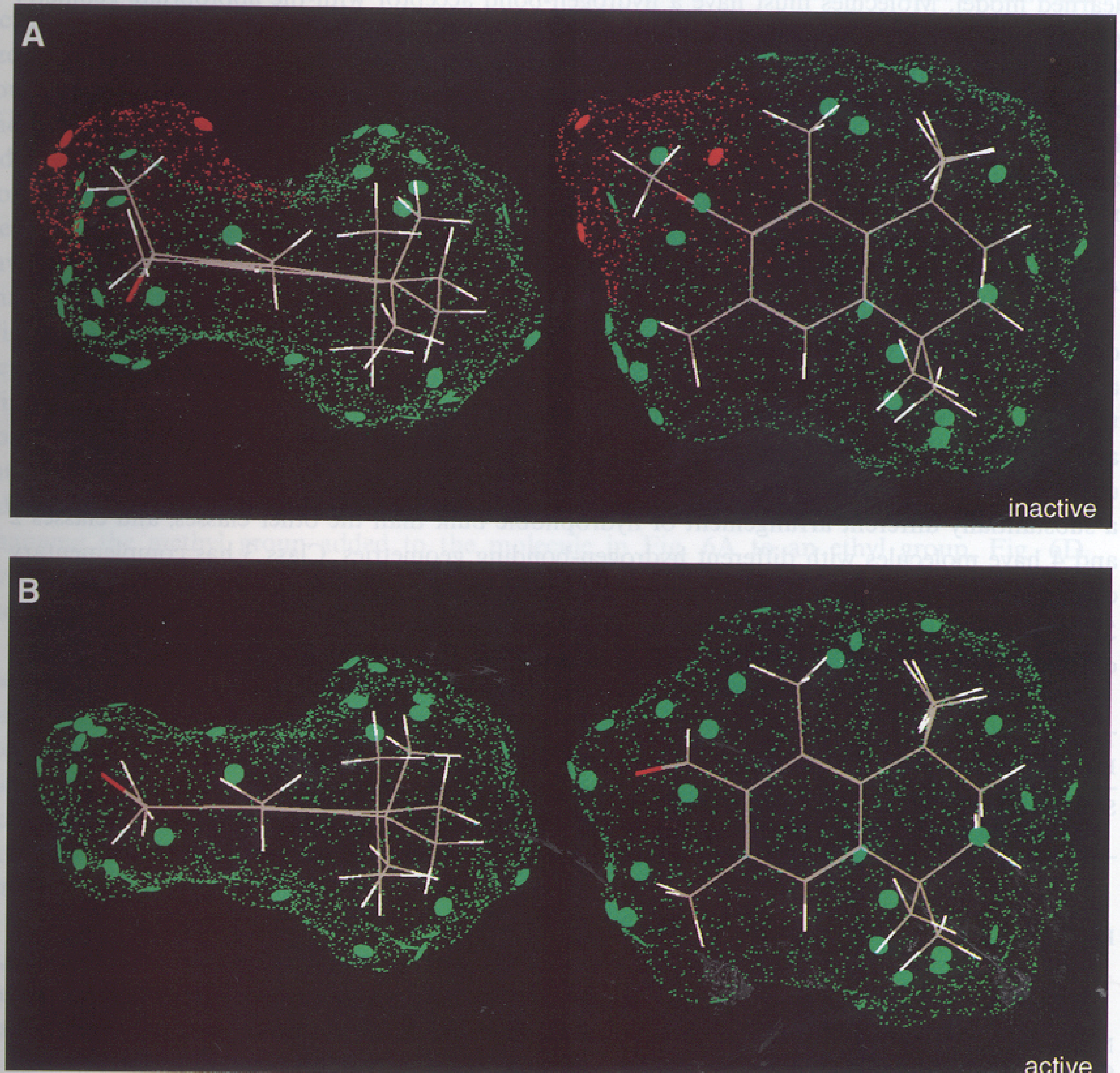






“Solving the multiple instance problem with axis-parallel rectangles”

Dietterich, Lathrop, Lozano-Perez, Artificial Intelligence 89(1997) 31-71



“Compass: A shape-based machine learning tool for drug design,” Jain, Dietterich, Lathrop, Chapman, Critchlow, Bauer, Webster, Lozano-Perez,

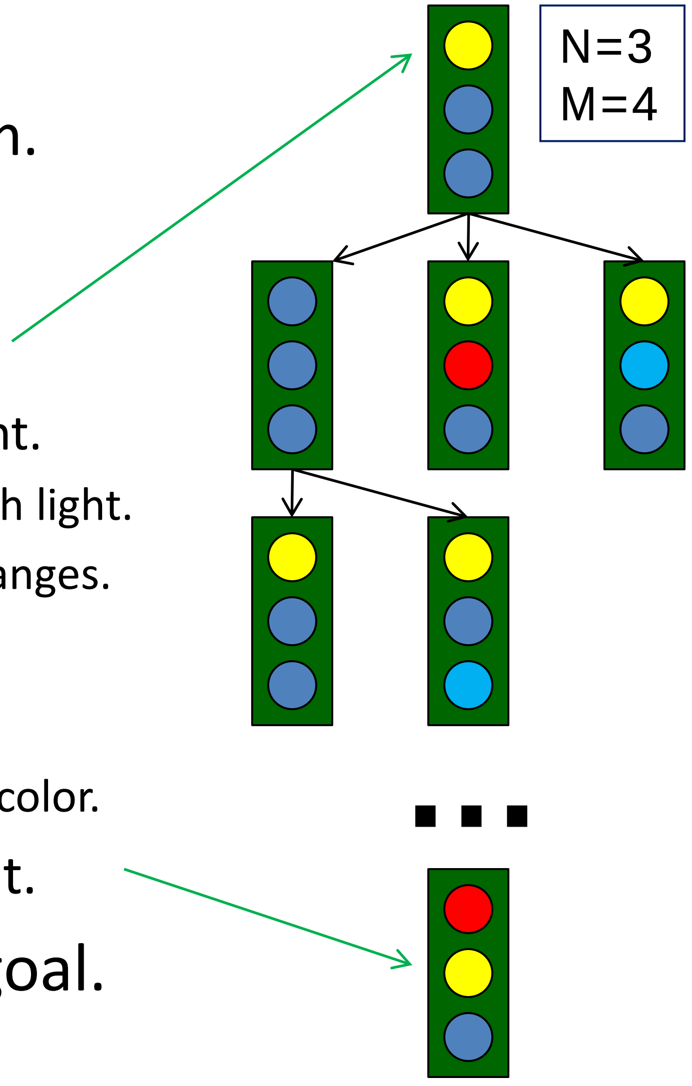
J. Of Computer-Aided Molecular Design, 8(1994) 635-652

Fig. 7. Remediation of unfavorable steric interaction. (A) Non-musk

From last class meeting – a "non-distance" heuristic

- The "N Colored Lights" search problem.
  – You have N lights that can change colors.
    - Each light is one of M different colors.
  – <u>Initial state</u>: Each light is a given color.
  – <u>Actions</u>: Change the color of a specific light.
    - You don't know what action changes which light.
    - You don't know to what color the light changes.
    - Not all actions are available in all states.
  – <u>Transition Model</u>: **RESULT**(s,a) = s'

    where s' differs from s by exactly one light's color.
  – <u>Goal test</u>: A desired color for each light.
- Find: Shortest action sequence to goal.



From last class meeting – a "non-distance" heuristic

- The "N Colored Lights" search problem.
  - Find: Shortest action sequence to goal.
- h(n) = number of lights the wrong color
- f(n) = g(n) + h(n)
  - f(n) = (under-) estimate of total path cost
  - g(n) = path cost so far = number of actions so far
- Is h(n) admissible?
  - Admissible = never overestimates the cost to the goal.
  - Yes, because: (a) each light that is the wrong color must change; and (b) only one light changes at each action.
- Is h(n) consistent?
  - Consistent = h(n) ≤ c(n,a,n') + h(n'), for n' a successor of n.
  - Yes, because: (a) c(n,a,n')=1; and (b) h(n) ≤ h(n')+1
- Is A* search with heuristic h(n) optimal?