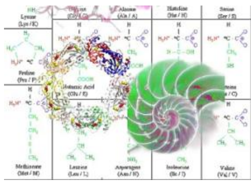
Computer Vision Lab

http://vision.unipv.it/



PROTEIN STRUCTURE ANALYSIS THROUGH HOUGH TRANSFORM AND RANGE TREE

VIRGINIO CANTONI, Dipartimento di Informatica e Sistemistica, Università di PAVIA, virginio.cantoni@unipv.it

ELIO MATTIA, Center for Systems Chemistry, Rijksuniversiteit Groningen, Groningen, The Netherlands, E.Mattia@rug.nl

Overview

- Searching in a database of protein structures
  - Pairwise comparison
  - All-to-All comparison
  - Search for a structural “motif”

General Hough transform approach to protein structure comparison

3C Vision

cues, contexts and channels

Elsevier (April 2011)

V. Cantoni, S. Levialdi, B. Zavidovique

Università di Pavia, Roma, Université de Paris XI

Paul Hough , 1959 : straight lines

$y = mx+q$

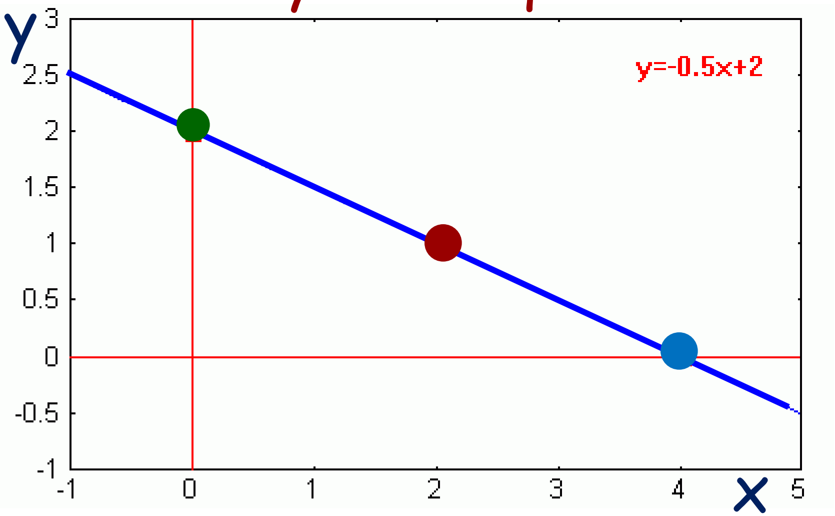
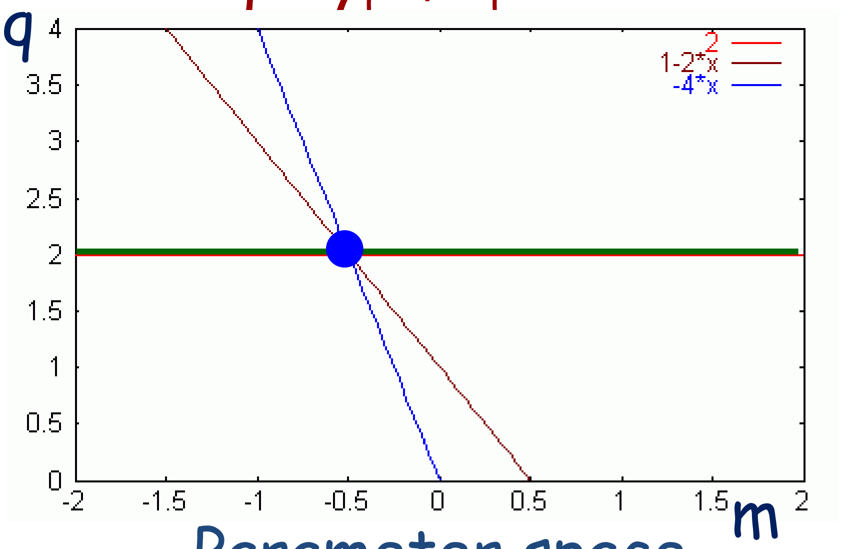
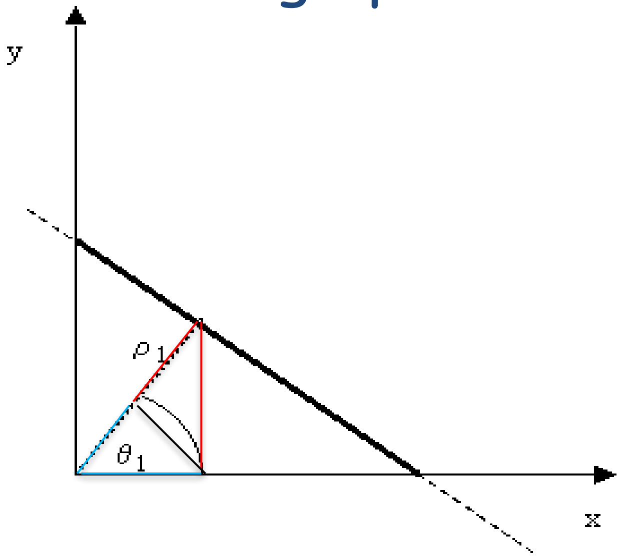


Image plane

$q = y_i-mx_i$



Parameter space

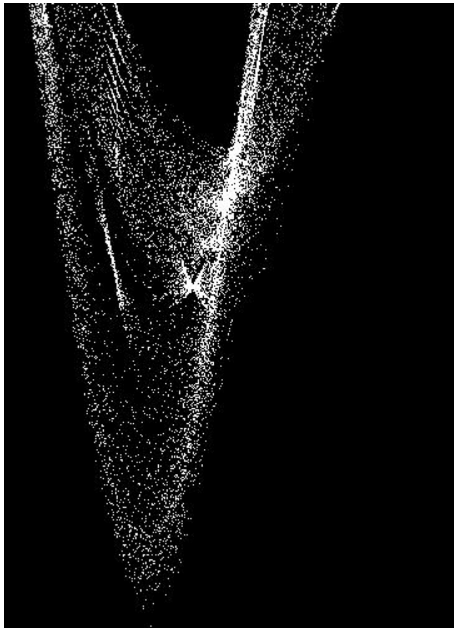


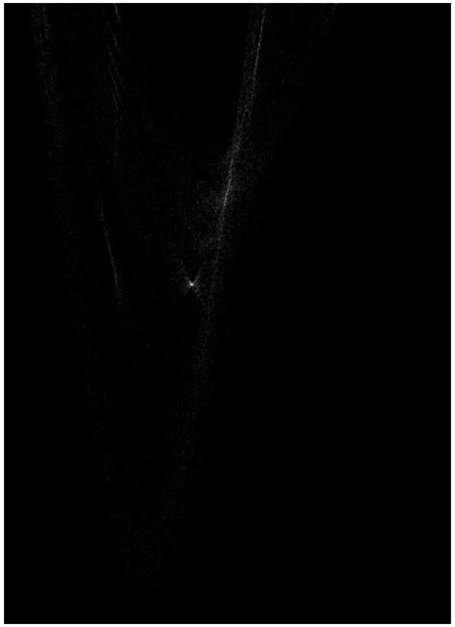
$-\propto < m, q < +\propto$

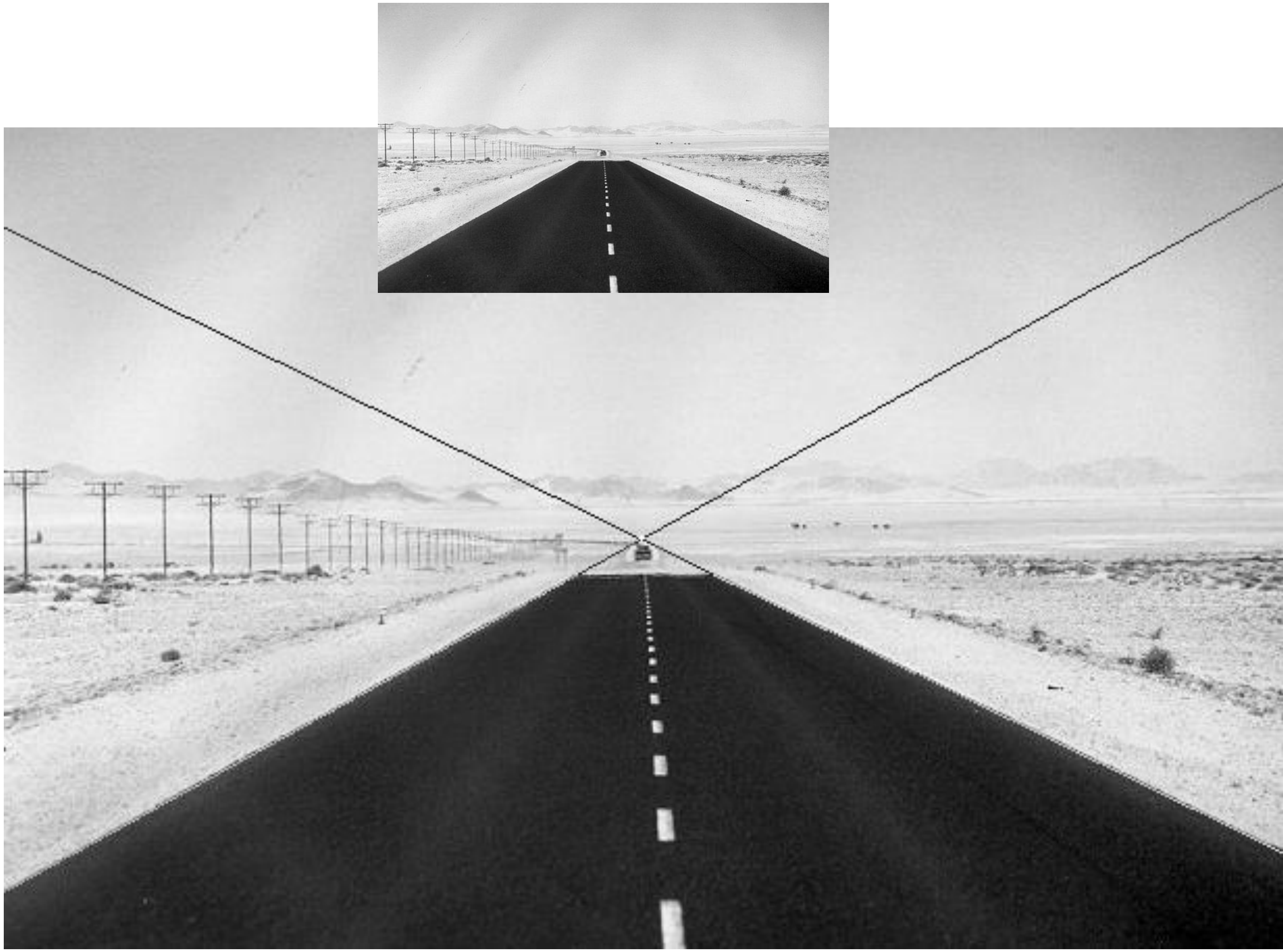
$\rho = x\cos(\theta) + y\sin(\theta)$

$0 < \rho < L\sqrt{2}; -\pi \leq q \leq \pi$

Example







Richard Duda and Peter Hart 1972: Circles

$$f((x,y),x_c,y_c,r) = (y-y_c)^2+(x-x_c)^2-r^2=0$$

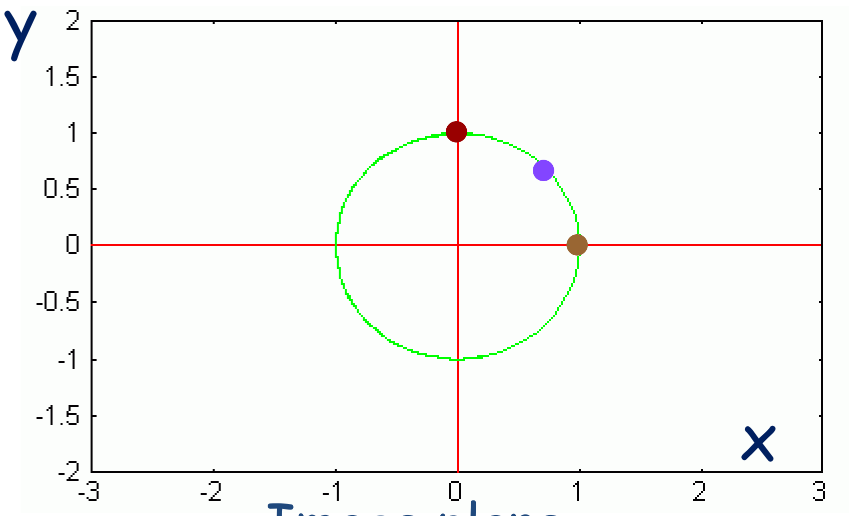
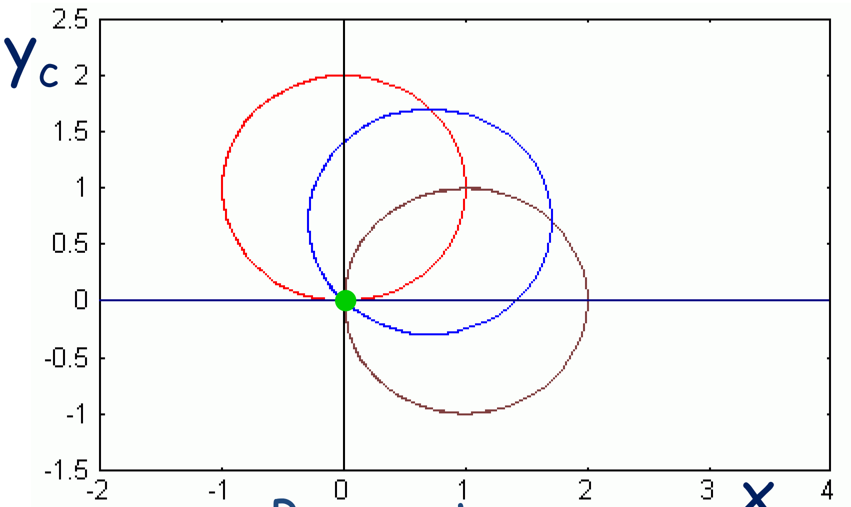
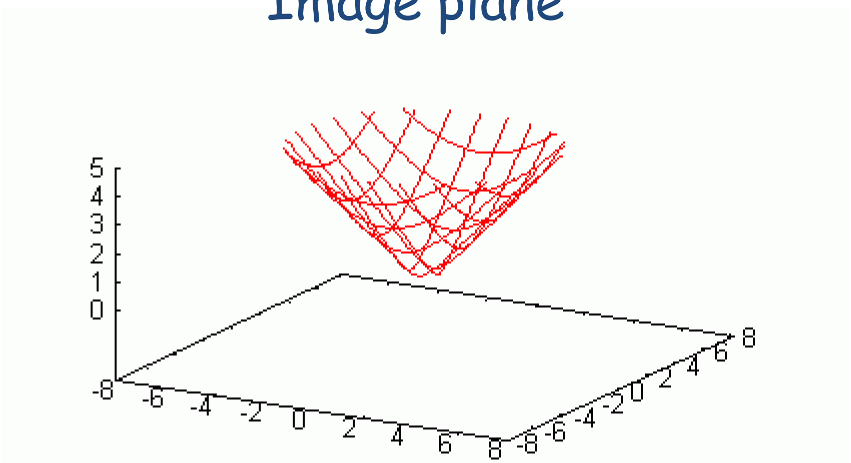


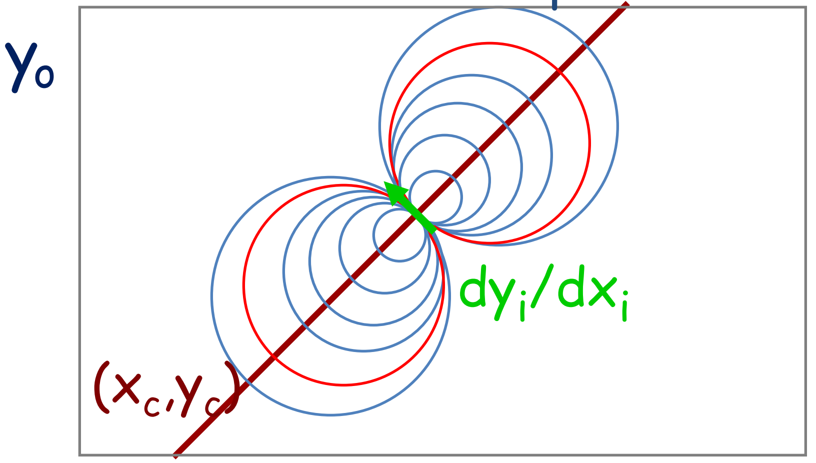
Image plane



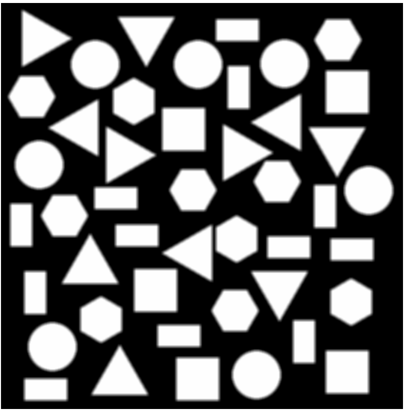
Parameter spaces



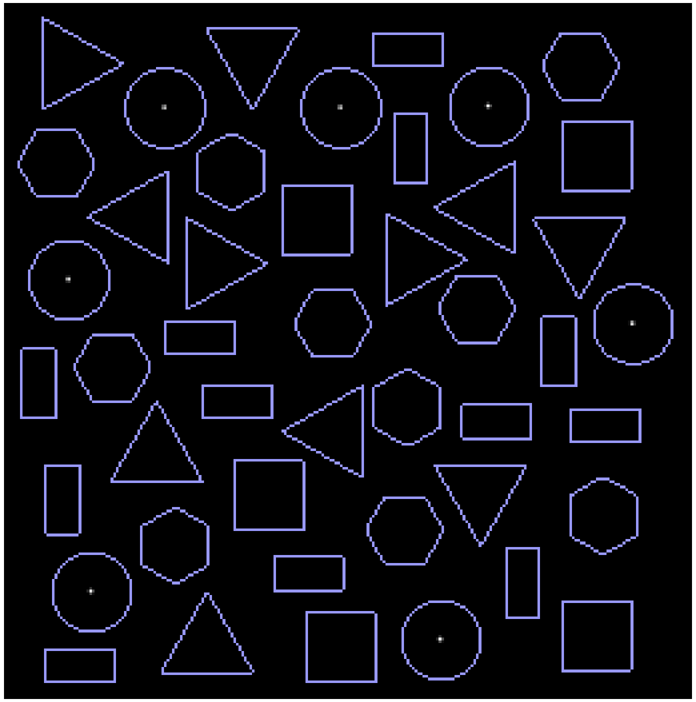
Parameter space

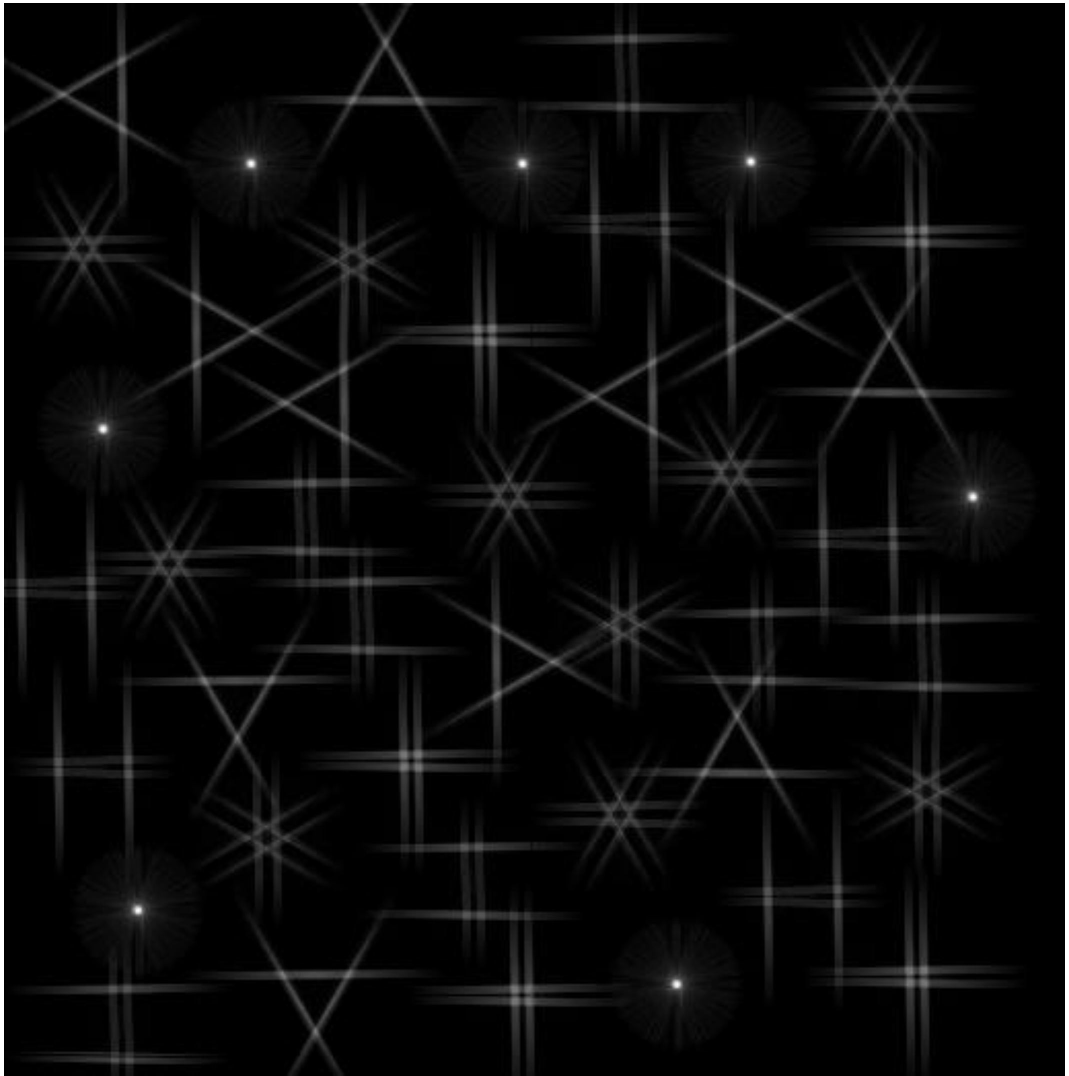


$$y_c = -1/m_i\, x_c + (y_i - m_i x_i)$$



Exemple de vote: cercle



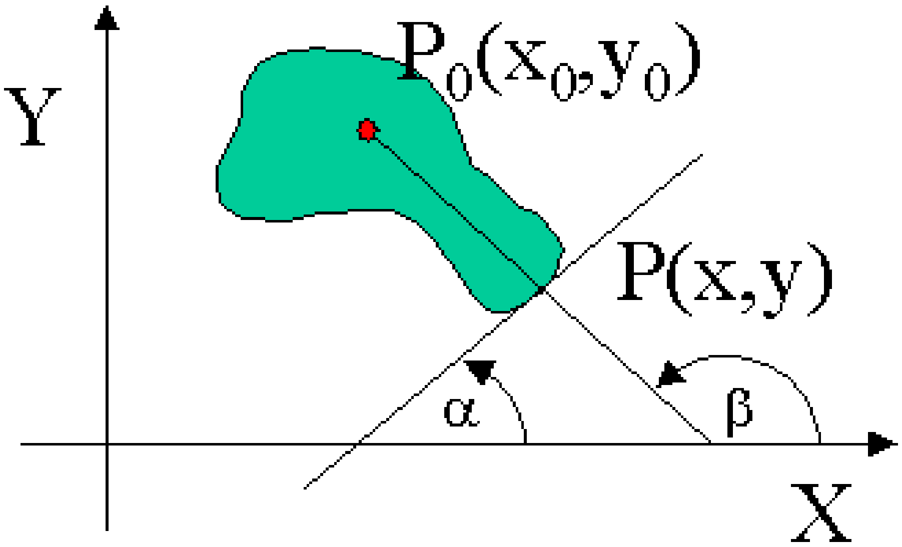


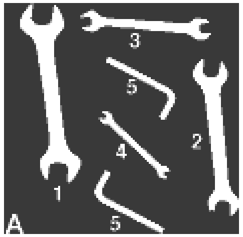
Dana H. Ballard 1981: Generalized HT

Mapping rule

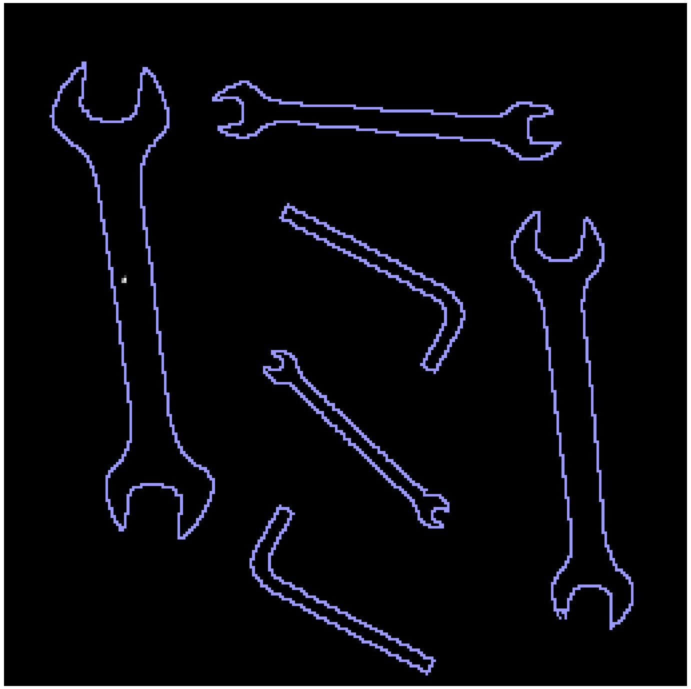
$X_0 = x + \rho \cos(\alpha);$

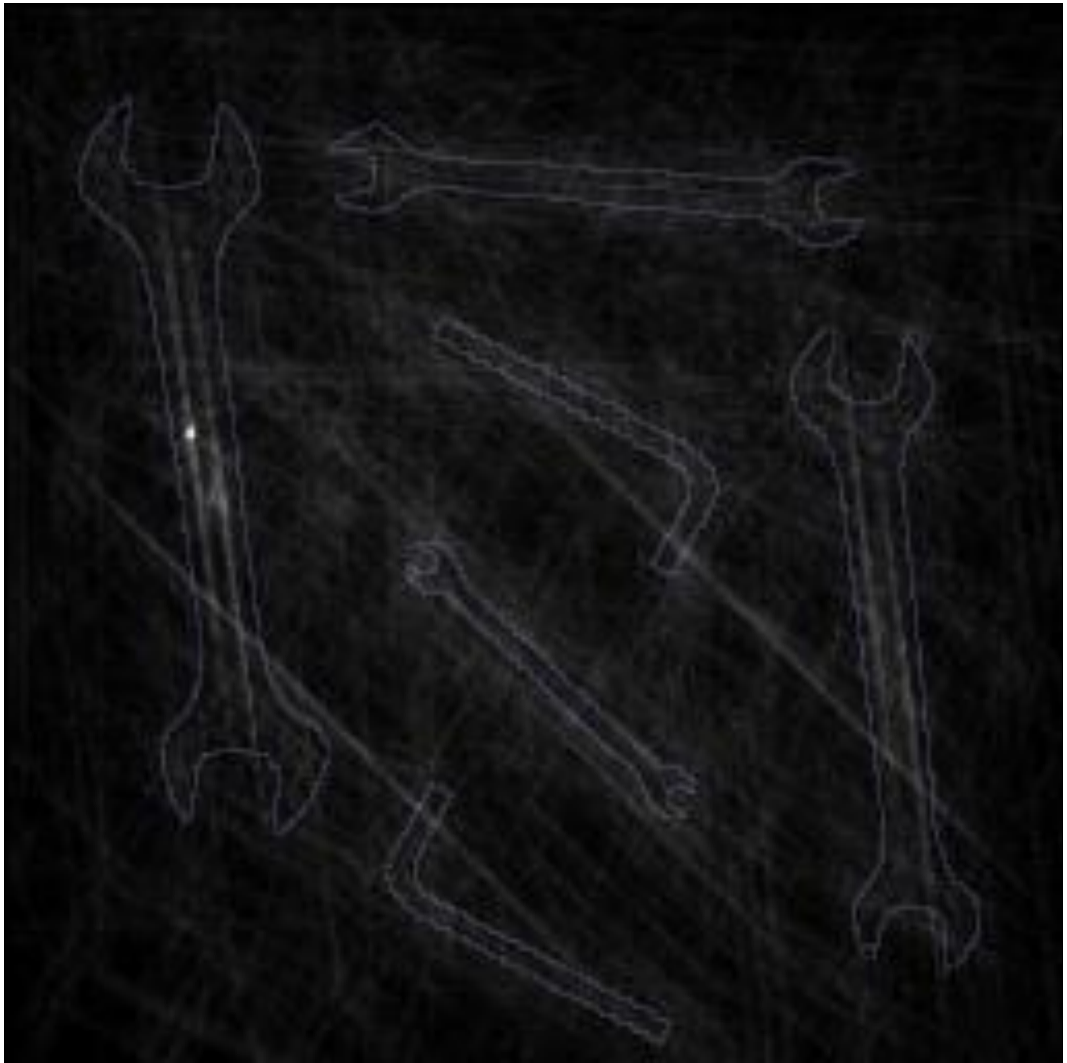
$Y_0 = y + \rho \sin(\alpha)$





Exemple de vote : clé 1





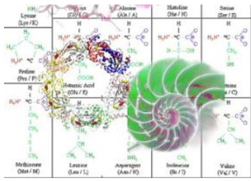
Basics on Proteins

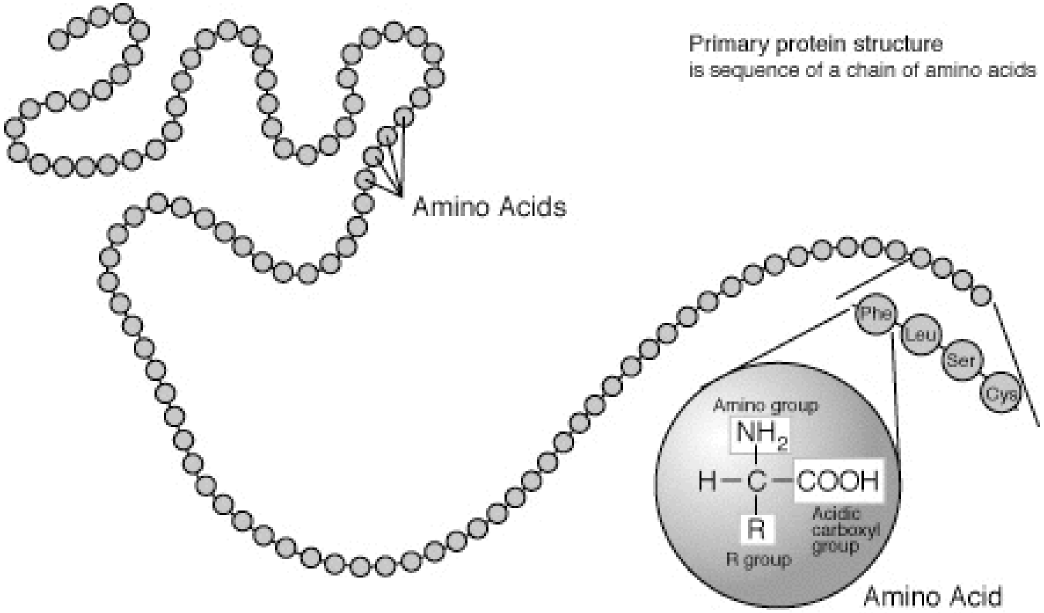
- A protein is an ordered sequence of amino acids
- Building blocks: 20 amino acid residues.
- Three-dimensional shapes (“fold”) vary enormously.

Levels of protein structure representation

- Primary structure
- Secondary structure
- Tertiary structure
- Quaternary structure

Primary structure: the sequence of amino acids





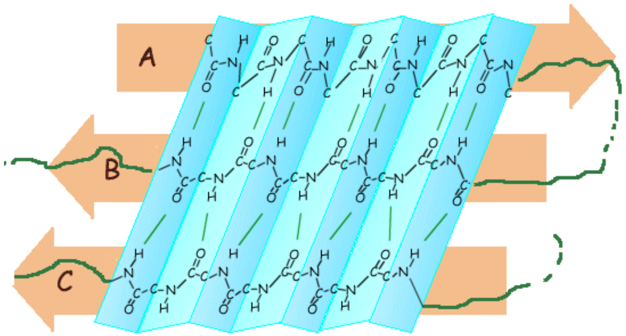
MHGAYRTPRSKTDAYGCQILETRAS

Secondary structures

Three basic components:

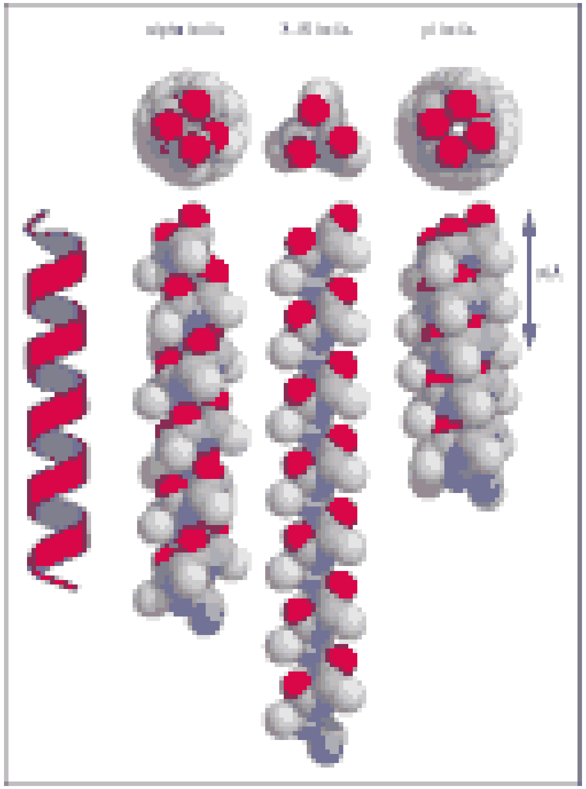
- helix

- sheet



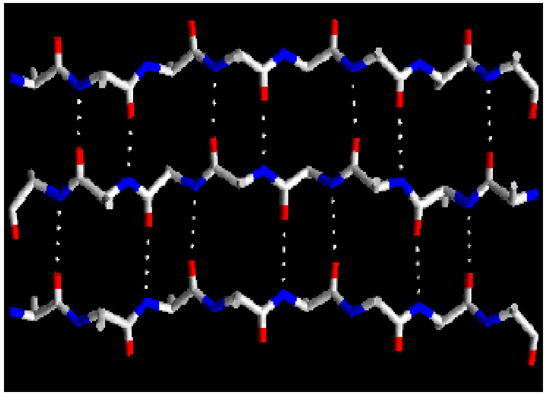
- Loops (linear connections between the components)

The helix

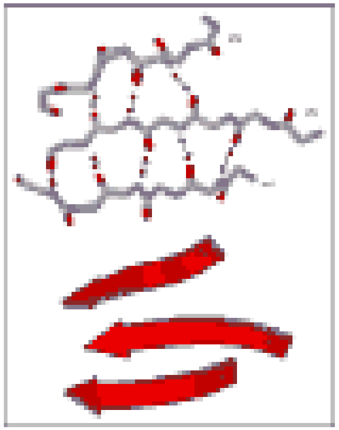


- One of the most closely packed arrangement of residues.
- ~40% of residues in globular proteins

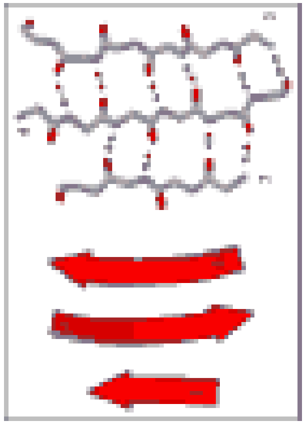
The sheet



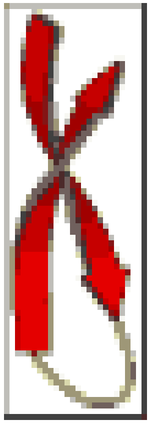
loosely packed arrangement of residues.



Parallel



Antiparallel



Twisted

Secondary Structures Representation

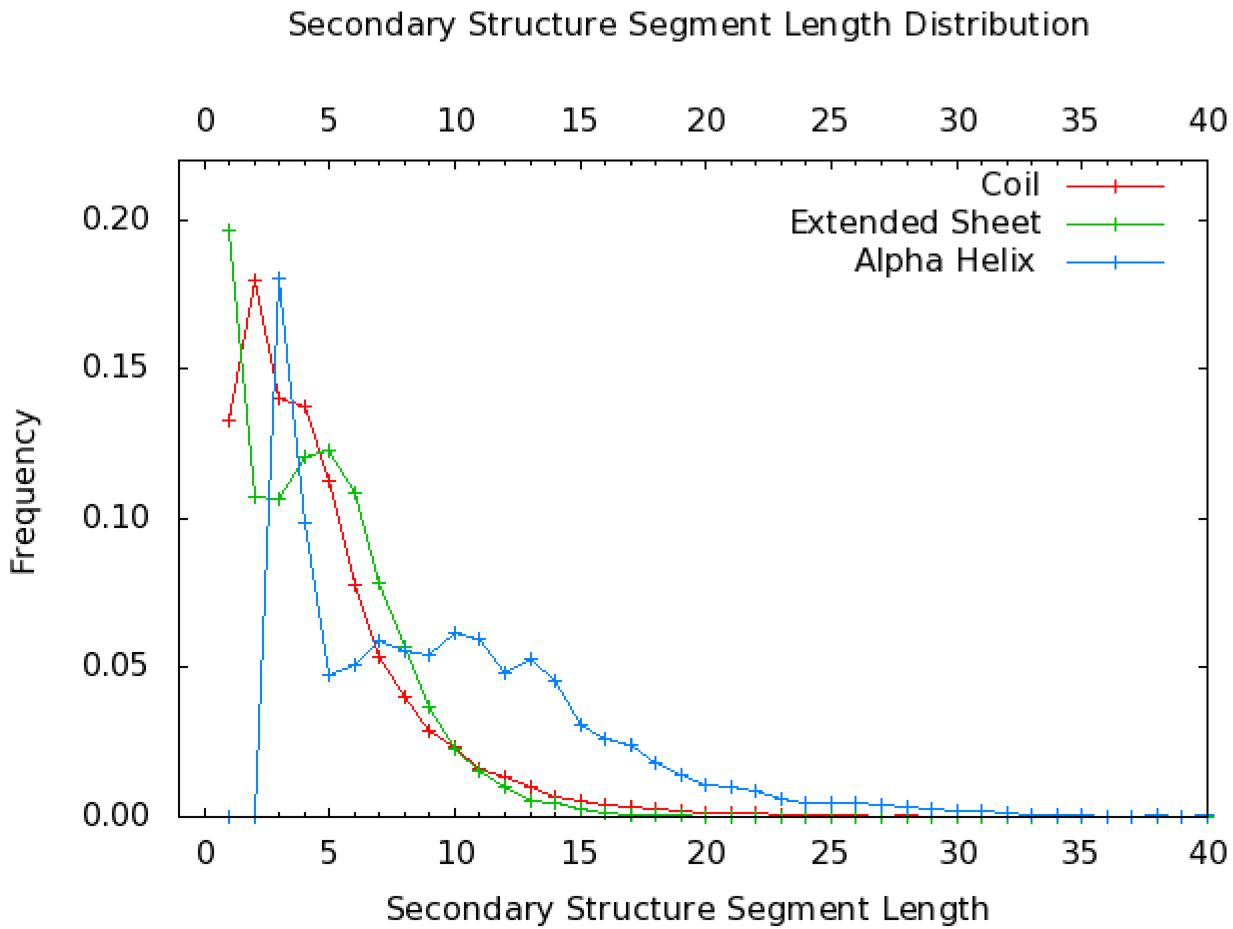
- Secondary structures are represented as linear vectors (segments):

  the axis for the alpha helix and the best fit segment for a strand
- An alignment algorithm is used to match an helix segments with known axes to determine helix axis. Direct segment fits are made to fit sheet strands.

Secondary Structure Determination

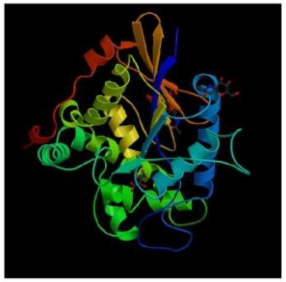
- Programs: DSSP and STRIDE.
- On the average 4.8% of the target residues were differently assigned, this number reaching 12% for certain targets.

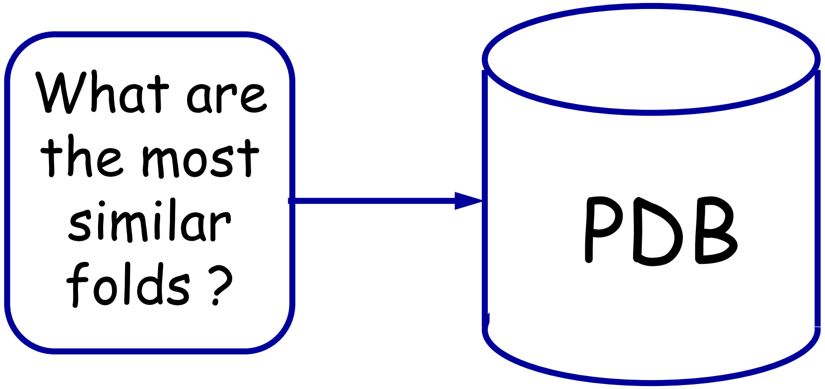
Distribution of segment lenght



Protein Structure Comparison

Given a motif or domain or protein



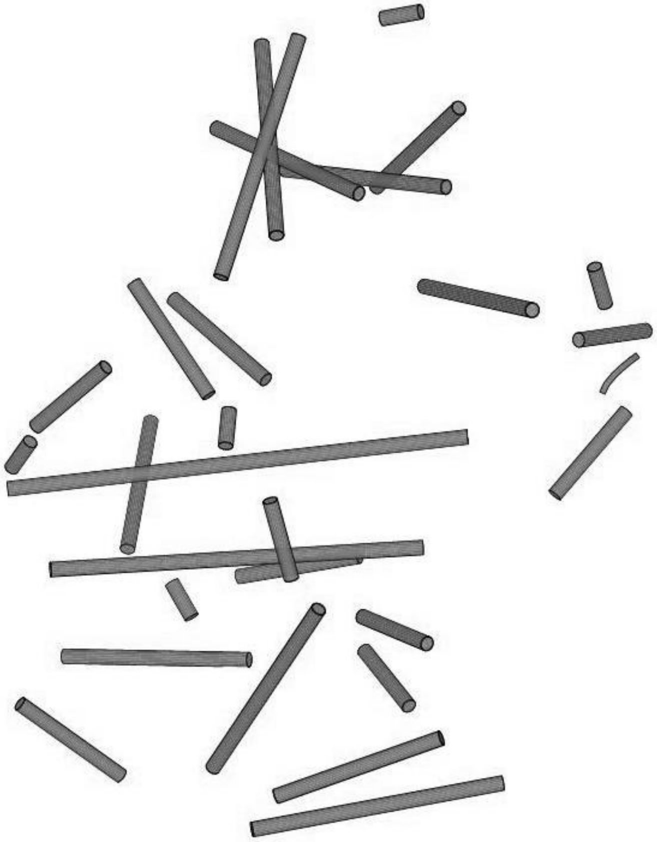


What are the most similar folds ?

PDB

Secondary structure representation

- Each segment is associated to a secondary structure and is displayed as a cylinder
- The protein is represented by and ordered sequence of cylinder with two labels: helices or strands

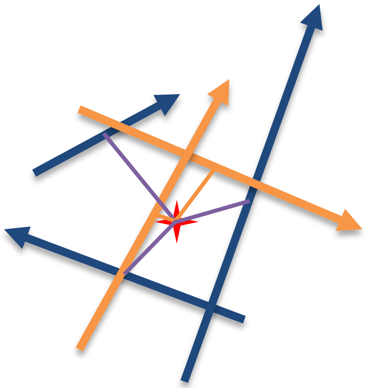


GHT applied to proteins

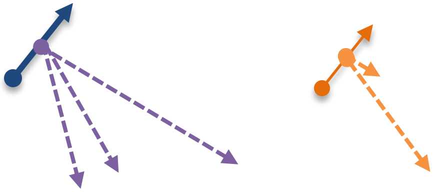
- For every protein, the distance (ρ) of every secondary structure from a reference point (RP, eg the geometric center of the protein) and the angle (theta) between the direction of the secondary structure in the 3D space and the segment linking the center of that secondary structure with the RP are first calculated. (GH reference table RT)

In the way of GHT
(simplified 2D representation)

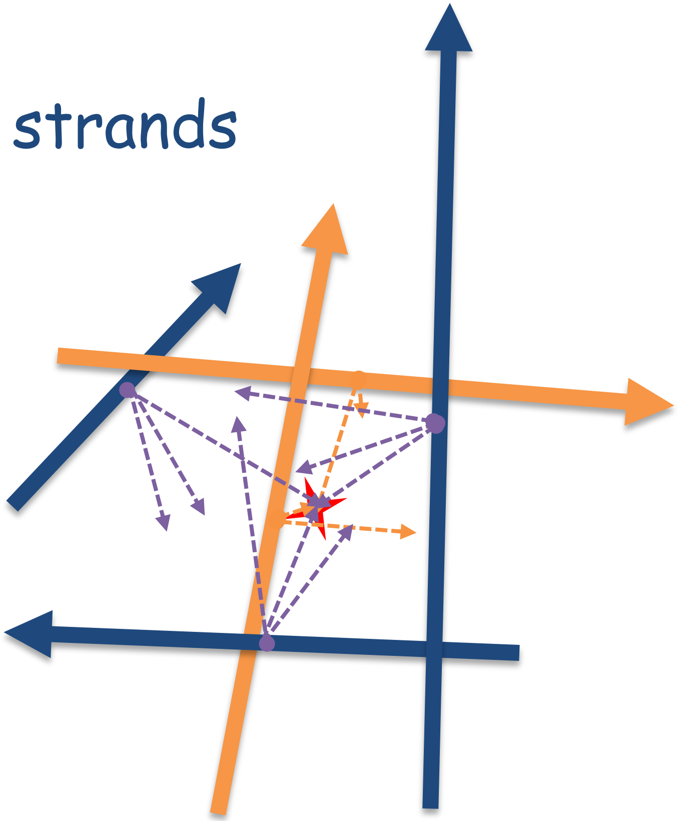
helices and strands



Query protein
(scaled 0.5)



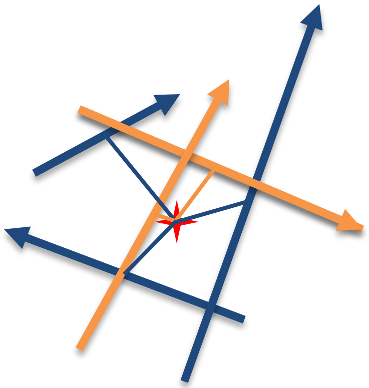
Mapping Rule



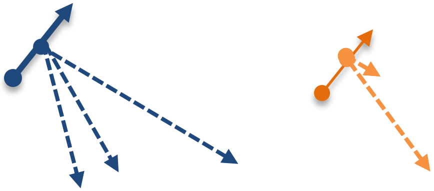
Votes Space

In the way of GHT

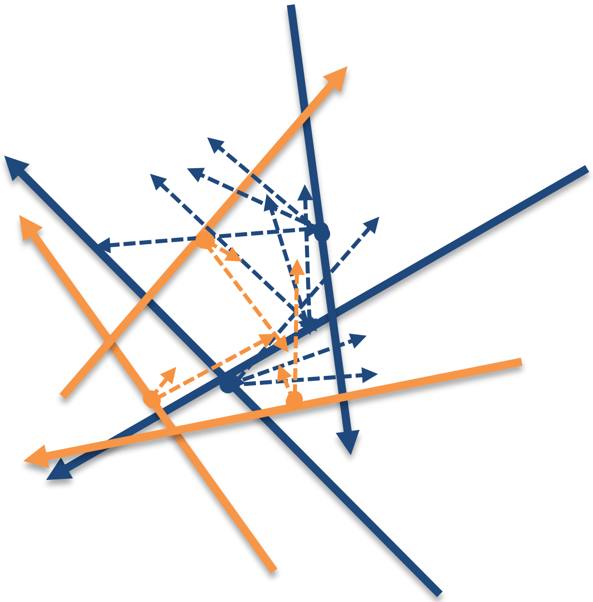
helices and strands



Query protein

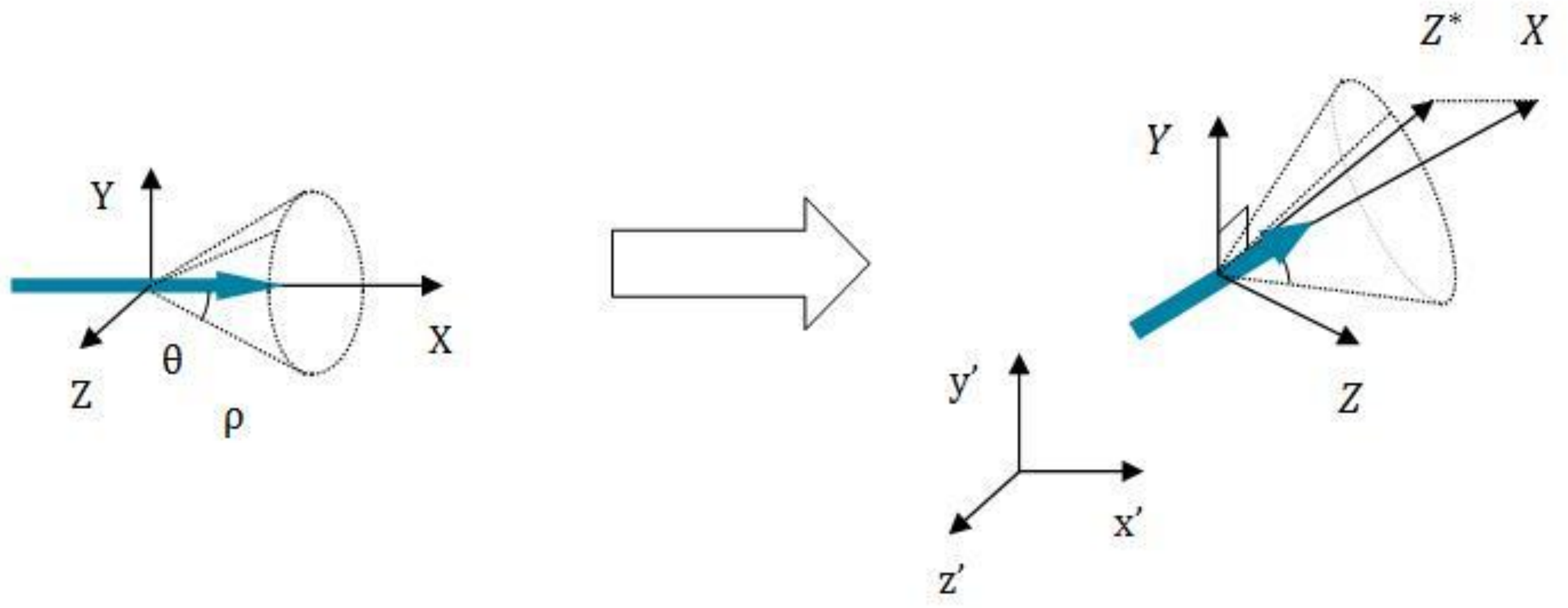


Mapping Rule

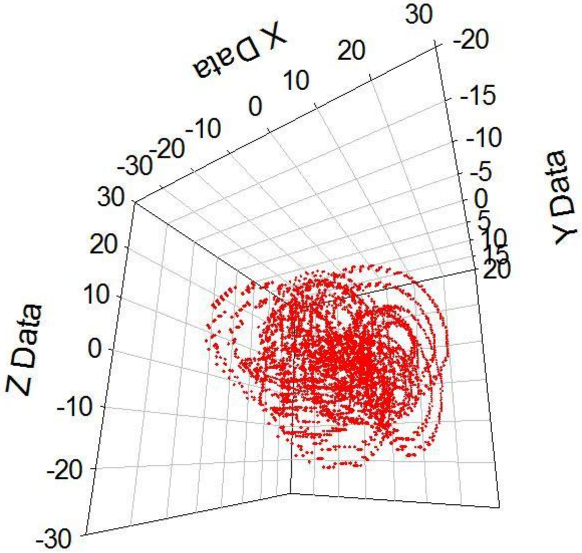


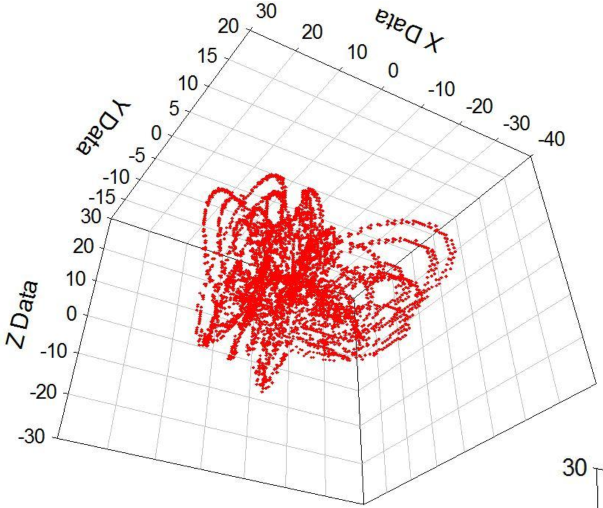
Votes Space

Proteins: the 3D solution

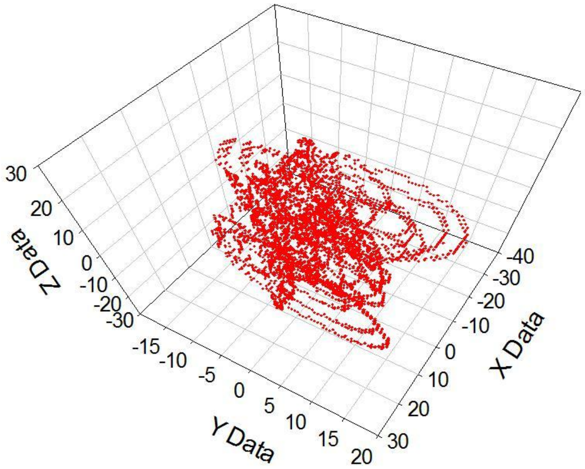


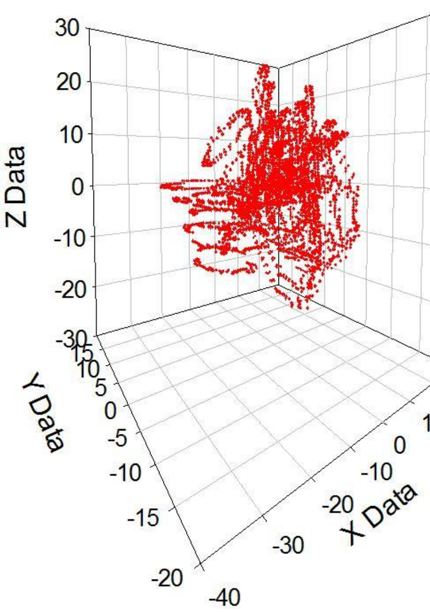
GH parameters spaces





Credits:
Elio Mattia





GHT applied to proteins

- In the 3D space of a given “object protein”, every secondary structure of a “model protein” votes a circumference of points starting from every secondary structure of the object protein.
- If the proteins are similar in shape, the circumferences will all intersect in a given point.

Main characteristics

- the mapping rule, for each compatible correspondent, in 3D is a circle on a plane perpendicular to the axis of the secondary structure
- Other information can be exploited to increase the S/N ratio:
  - the length of the secondary structure
  - the residues properties contained in the SS
  - any other (biochemical, morphological, etc.) peculiarities.

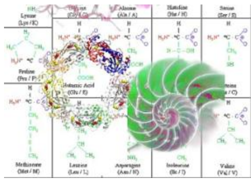
The implementation

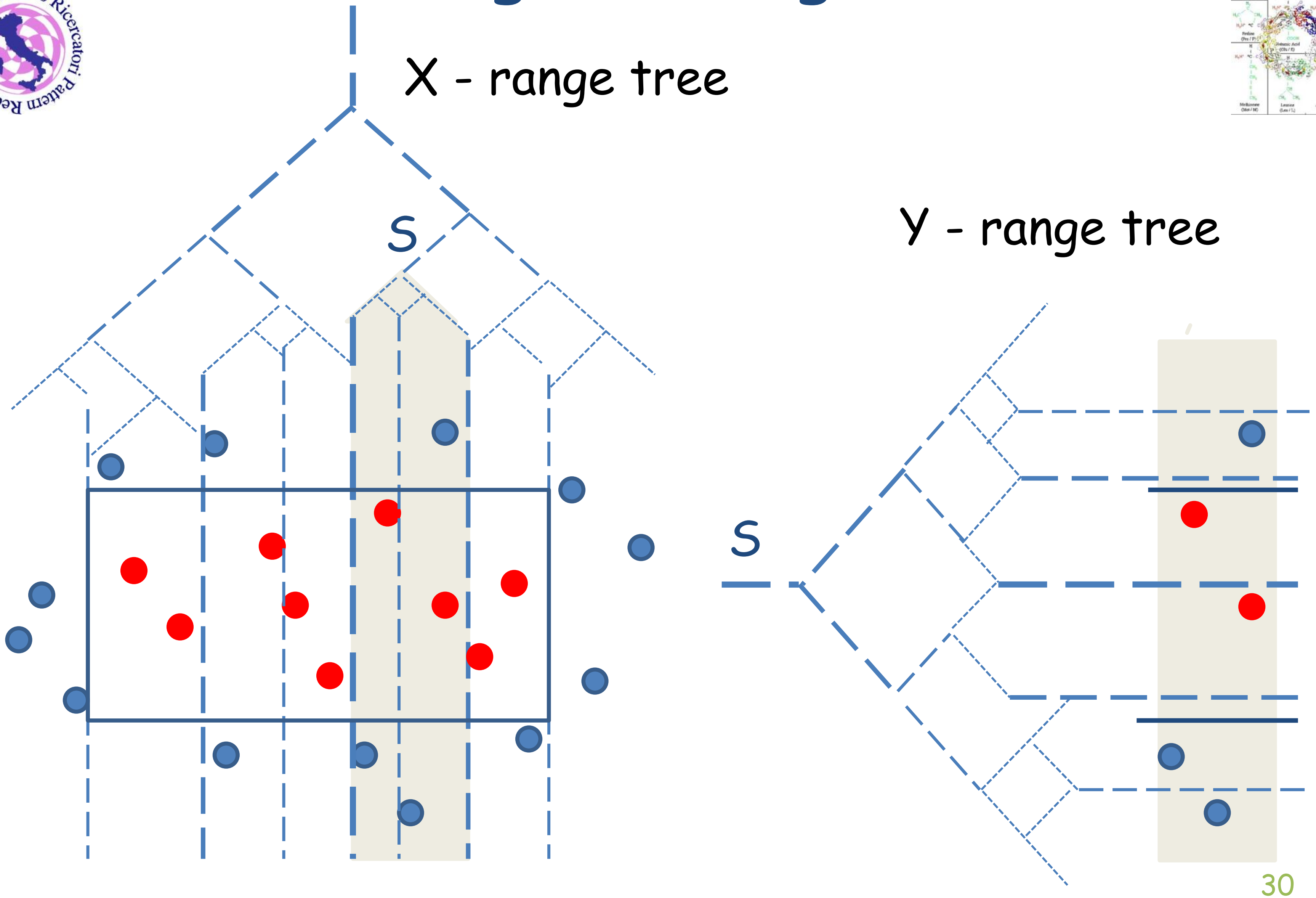
- The voting space is smoothed by accumulation of nearby votes (within a given radius) for each point
- After smoothing, the highest peaks in the voting space are detected (avoiding to pick high votes that however are not the top of a peak but lie close to one such peak)
- Only the relevant votes are stored in memory: there isn't a matrix with all the possible cells.

Smoothing Algorithm

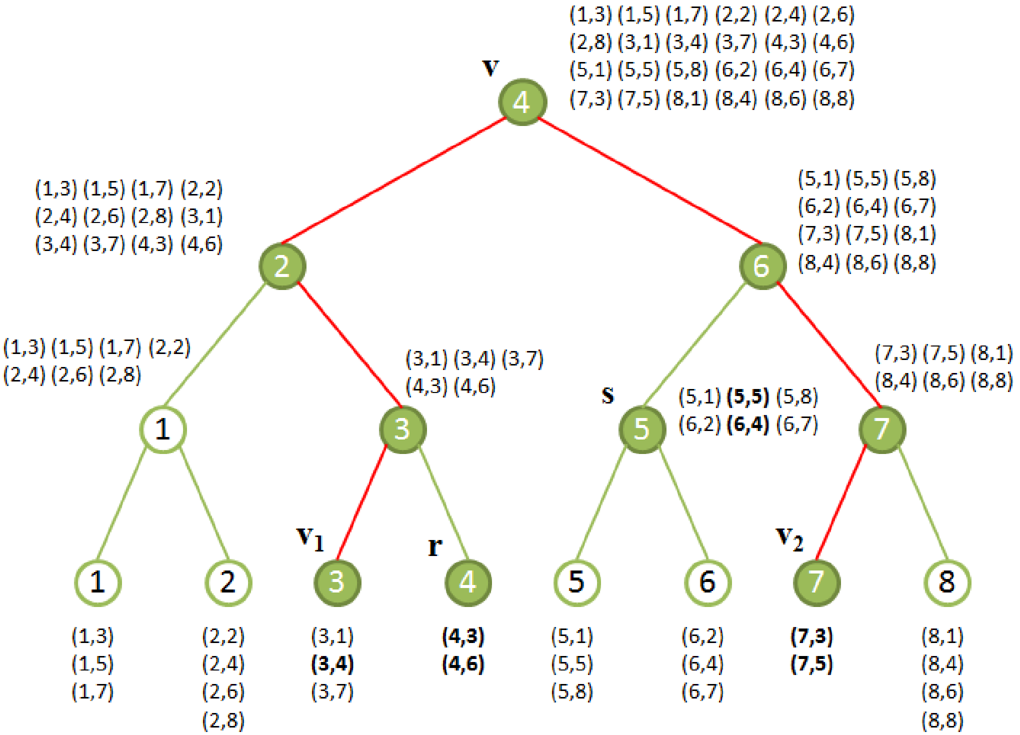
- Smoothing is performed by accumulating votes within a given radius, for every point in the vote space.
- The classic version, i.e., checking every vote for the vicinity condition, has been proven to be too time-consuming for applications, with a time complexity of $O(n^2)$, where n is the number of votes in the vote space.
- The smoothing problem can be seen as an "orthogonal search" problem, i.e., finding points within a given cube in space.
- A particular structure has been implemented for solving this problem with a $O(n \log^3(n))$ complexity: Range Trees.

Ortogonal range tree





Ortogonal range tree



v

(1,3) (1,5) (1,7) (2,2) (2,4) (2,6)
(2,8) (3,1) (3,4) (3,7) (4,3) (4,6)
(5,1) (5,5) (5,8) (6,2) (6,4) (6,7)
(7,3) (7,5) (8,1) (8,4) (8,6) (8,8)

(1,3) (1,5) (1,7) (2,2)
(2,4) (2,6) (2,8) (3,1)
(3,4) (3,7) (4,3) (4,6)

(5,1) (5,5) (5,8)
(6,2) (6,4) (6,7)
(7,3) (7,5) (8,1)
(8,4) (8,6) (8,8)

(1,3) (1,5) (1,7) (2,2)
(2,4) (2,6) (2,8)

(3,1) (3,4) (3,7)
(4,3) (4,6)

s

(5,1) **(5,5)** (5,8)
(6,2) **(6,4)** (6,7)

(7,3) (7,5) (8,1)
(8,4) (8,6) (8,8)

v_1 r v_2

(1,3)
(1,5)
(1,7)

(2,2)
(2,4)
(2,6)
(2,8)

(3,1)
(3,4)
(3,7)

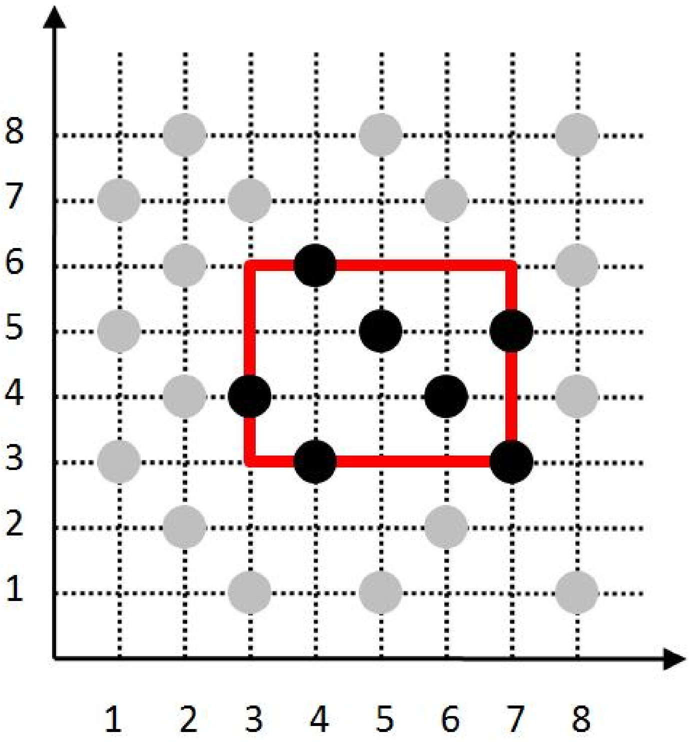
(4,3)
(4,6)

(5,1)
(5,5)
(5,8)

(6,2)
(6,4)
(6,7)

(7,3)
(7,5)

(8,1)
(8,4)
(8,6)
(8,8)



The implementation

- The comparison of ONE (1) object protein with MANY (N) model proteins is accomplished by sorting the votes of the top peaks in the spaces of each of the (N) model proteins.
- The sorting is carried out in TWO ways: either the smoothed votes themselves are sorted, or the differences between the two highest peaks in each of the (N) voting spaces are sorted.

First results

		Main Peak		First Difference	
obj. protein	nr. of secndry str's	L	Q	L	Q
1	20	✗	✗	✗	⚠
2	17	⚠	✓	✓	✓
3	18	✗	✓	✓	✓
4	5	✗	✗	⚠	⚠
5	5	✓	✓	✓	✓
6	18	✗	✗	✗	⚠
7	5	✗	✗	⚠	⚠
8	6	✓	✓	✓	✓
9	4	✓	✓	✓	✓
10	13	✗	⚠	⚠	✓
11	6	✓	✓	✓	✓
12	11	✗	⚠	✓	✓

Many errors to **Good results**

L	Linear weights
Q	Square-root weights
✗	Protein failed to be matched with itself
⚠	Protein was matched correctly, but with only a slightly better score than the second one in list
✓	Protein was matched correctly and with a fairly better score than the second one in list

Testing on Motif Retrieval

- The developed algorithm makes a new approach for protein structural comparison available.
- The main application of this new approach is to classify protein structures and to retrieve structural motifs which are common of a given protein function.
- Indeed, tests were performed on motif retrieval.
- As an example, a motif (present in the Ubiquitin Conjugating Enzyme) was found in other proteins which are known to contain it.
- Further testing will be done with the parallel implementation of the software.

Much experimentation allowed

- Computationally, the results might vary substantially if any of the following parameters are varied:
  - The mesh of the voting space (in Ångström)
  - The mesh of the voting circumference (how many votes in each circumference)
  - The radius of smoothing
  - The radius of tolerance for avoiding “false peaks” when detecting peaks
  - The normalization factor (linear, square root, etc.)