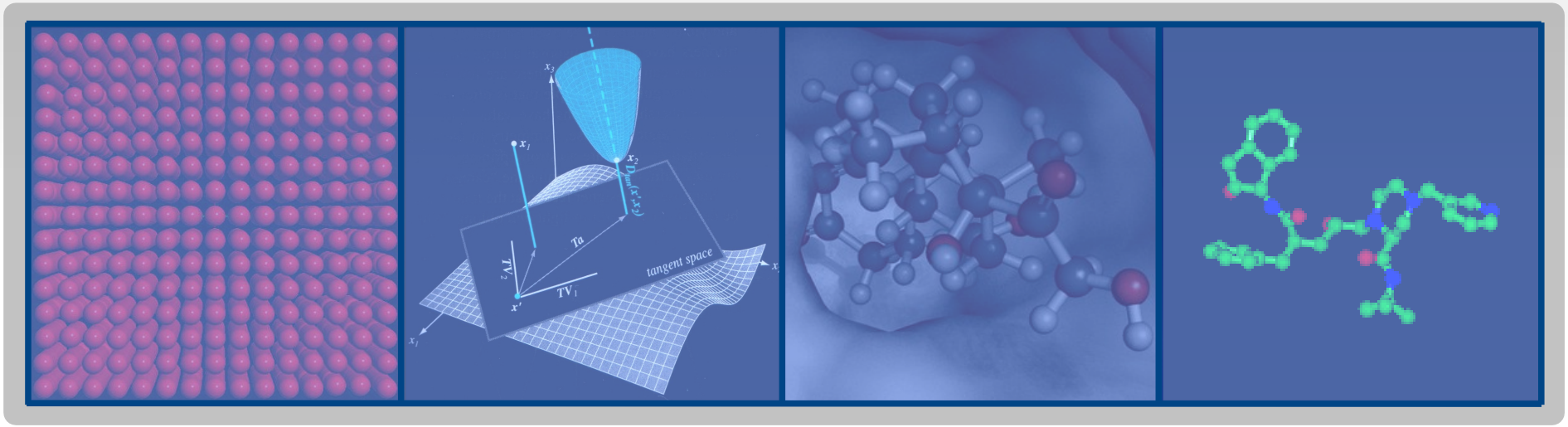
Pre-Docking Filter
Based on Image Recognition

Eva Kiszka, B.Sc.

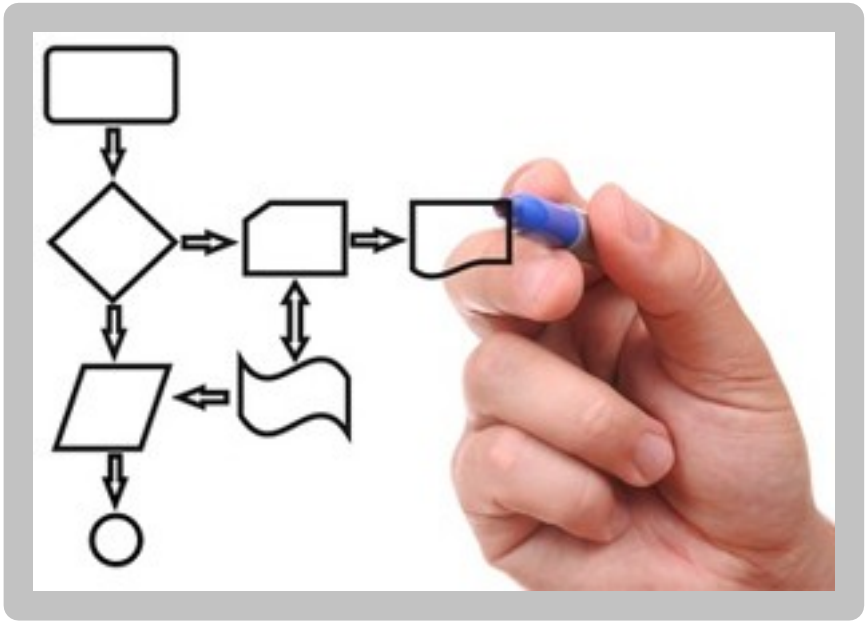


Master's Seminar Bioinformatics
Saarland University, Summer Term 2011
Supervisor: PD Dr. Michael C. Hutter

CBI CENTER FOR BIOINFORMATICS

Outline

- Introduction to docking basics
  - Relevance
  - Principles
  - Example molecules
- Idea
- Implementation
  - Status quo
  - Future work
  - Schedule
- Outlook



Introduction

~ Docking Basics ~

Definition „Docking“:

Prediction method for the orientation of one molecule to a second when bound to each other to form a stable complex.

Lengauer, Rarey (1996): Computational Methods for Biomolecular Docking. Curr. Opin. Struct. Biol. 6(3): 402-6

Introduction

~ Docking Basics ~

Definition „Docking“:

Prediction method for the orientation of one molecule to a second when bound to each other to form a stable complex.

Lengauer, Rarey (1996): Computational Methods for Biomolecular Docking. Curr. Opin. Struct. Biol. 6(3): 402-6

Approaches:

- MD simulation

Introduction

~ Docking Basics ~

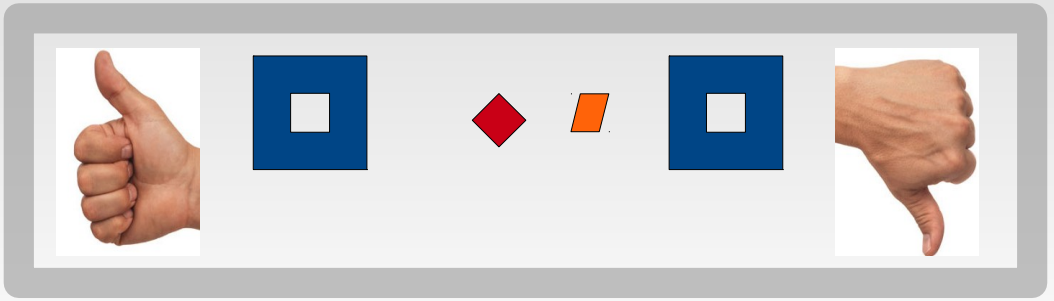
Definition „Docking“:

Prediction method for the orientation of one molecule to a second when bound to each other to form a stable complex.

Lengauer, Rarey (1996): Computational Methods for Biomolecular Docking. Curr. Opin. Struct. Biol. 6(3): 402-6

Approaches:

- MD simulation
- Shape complementarity



1. Molecule structures given (receptor, potential ligands)
2. Apply search algorithm
   → predicts ligand orientations
3. Apply scoring function
   → assigns ranking

Introduction

~ Docking Basics: Relevance ~

Problem in drug design: Vast amounts of potential ligands

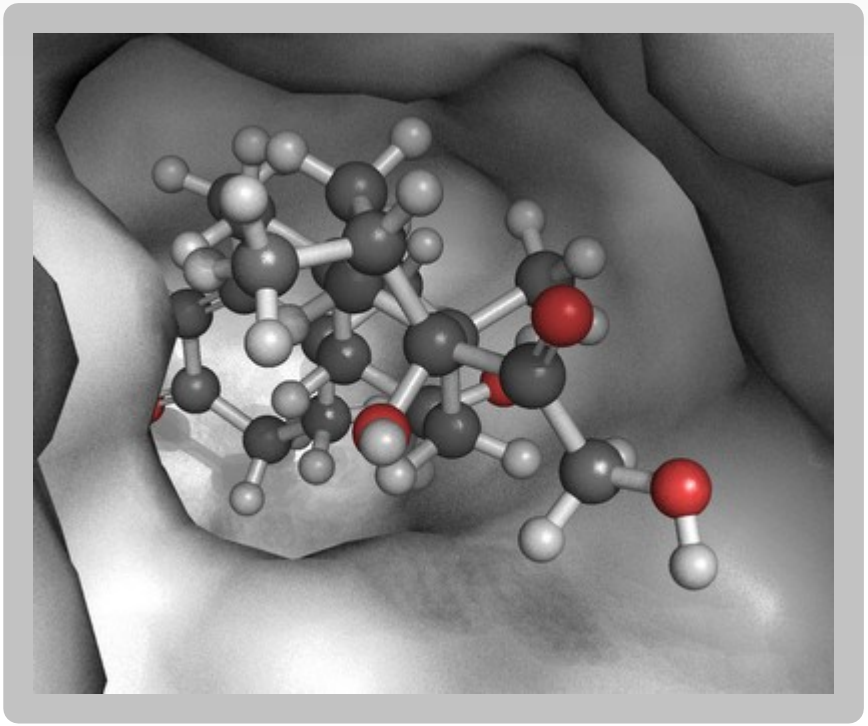
- Large variety of structures (algorithm with short runtime?)

Introduction

~ Docking Basics: Relevance ~

Problem in drug design: Vast amounts of potential ligands

- Large variety of structures (algorithm with short runtime?)
- Hard to identify the good ones (appropriate scoring function?)



Introduction

~ Docking Basics: Relevance ~

Problem in drug design: Vast amounts of potential ligands

- Large variety of structures (algorithm with short runtime?)
- Hard to identify the good ones (appropriate scoring function?)

A „good“ ligand?

- Fits into binding pocket of receptor

Introduction

~ Docking Basics: Relevance ~

Problem in drug design: Vast amounts of potential ligands

- Large variety of structures (algorithm with short runtime?)
- Hard to identify the good ones (appropriate scoring function?)

A „good“ ligand?

- Fits into binding pocket of receptor
- Strong, selective binding to receptor (high affinity, high specificity)

Introduction

~ Docking Basics: Relevance ~

Problem in drug design: Vast amounts of potential ligands

- Large variety of structures (algorithm with short runtime?)
- Hard to identify the good ones (appropriate scoring function?)

A „good“ ligand?

- Fits into binding pocket of receptor
- Strong, selective binding to receptor (high affinity, high specificity)
- Druglikeness (oral bioavailability, metabolic stability, ...)

Introduction

~ Docking Basics: Relevance ~

Problem in drug design: Vast amounts of potential ligands

- Large variety of structures (algorithm with short runtime?)
- Hard to identify the good ones (appropriate scoring function?)

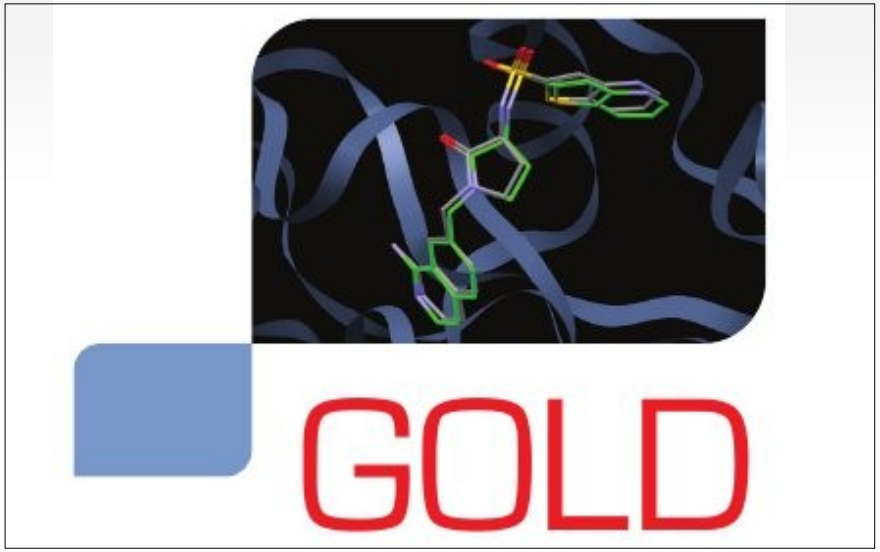
A „good“ ligand?

➢ Fits into binding pocket of receptor

- Strong, selective binding to receptor (high affinity, high specificity)
- Druglikeness (oral bioavailability, metabolic stability, ...)

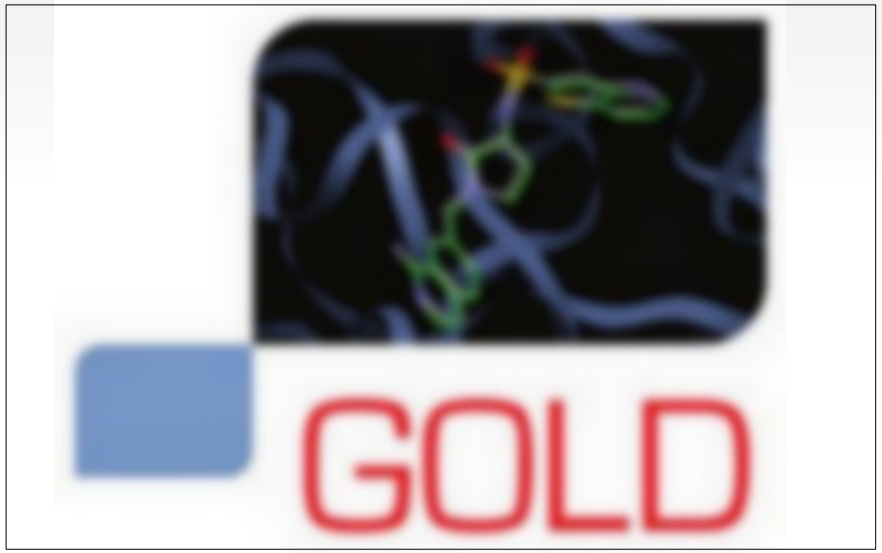
Introduction

~ Docking Basics: Software ~



Introduction

~ Docking Basics: Software ~



Introduction

~ Docking Basics: Software ~

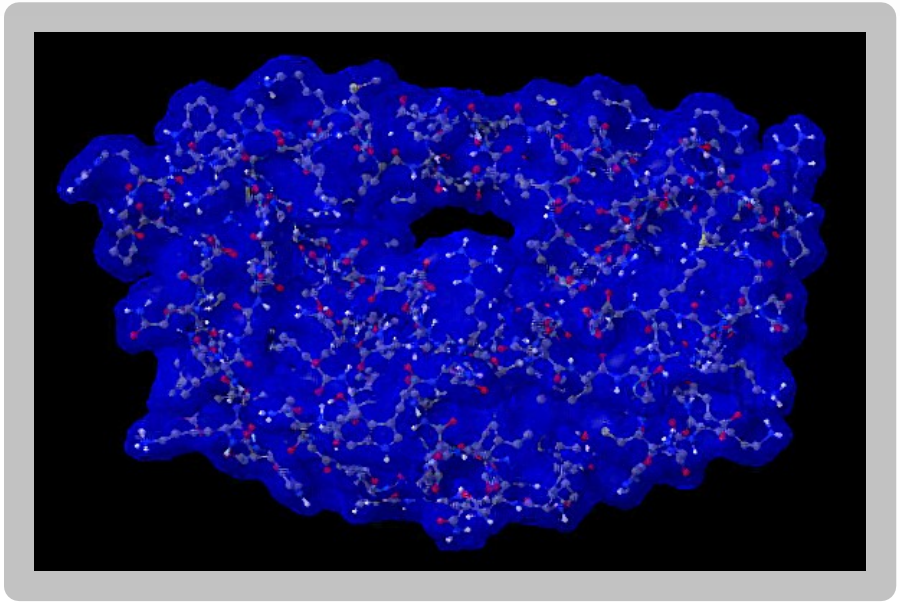
- Widely used in academia, open source software
- Maintainers: The Scripps Research Institute & Olson Laboratory
- Main components
  - ADT: Prepare coordinate files, analyze docking results
  - AutoGrid: Precalculate grid maps describing receptor
  - AutoDock: Dock ligand to set of grid maps

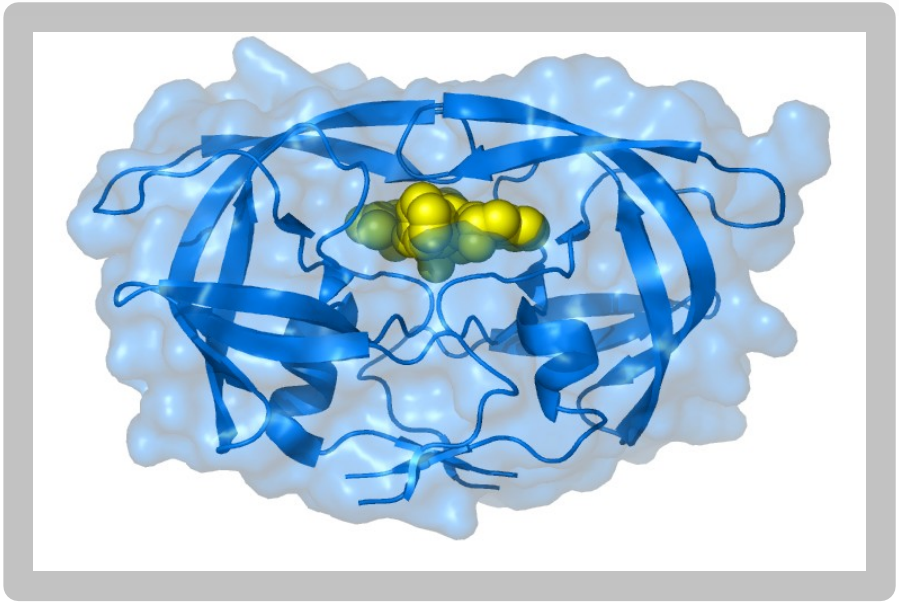
Introduction

~ Example Molecules: Protease / Indinavir ~

Receptor: **Human HIV II protease**

- Cleaves newly synthesized proteins, is essential in HIV lifecycle
- Homodimer with active site in the center of the complex



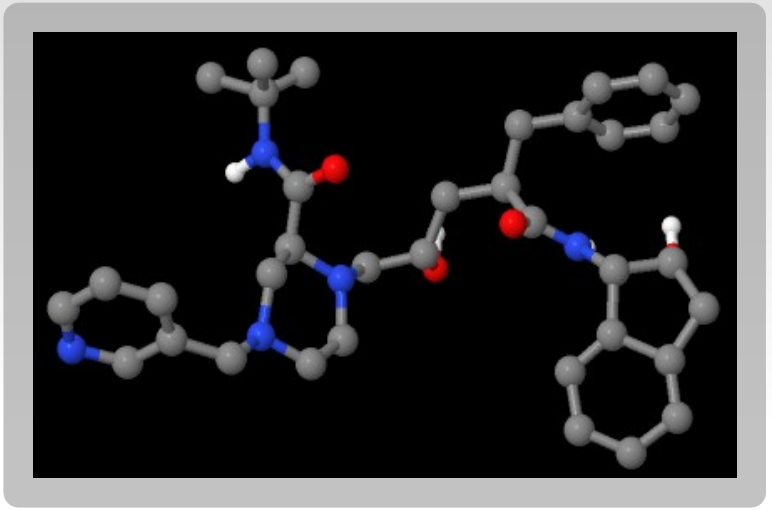


Introduction

~ Example Molecules: Protease / Indinavir ~

Ligand: **Indinavir**

- Protease inhibitor
- Discovered with AutoDock
- Milestone in the development of combination anti-retroviral therapy



Advantages of HIV protease & indinavir as an example:

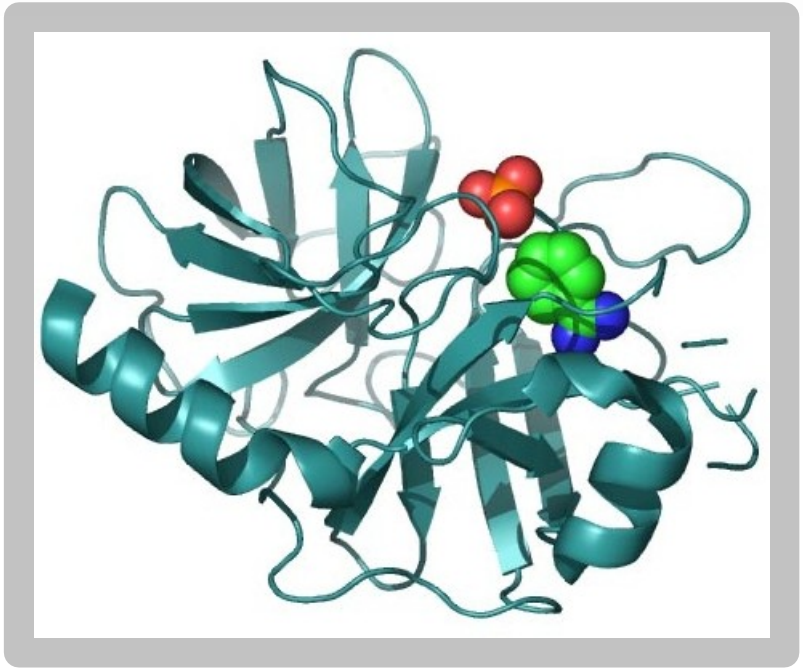
- Binding pocket is easy to visualize
- Official docking result available from Autodock (→ comparison)

Introduction

~ Example Molecules: Trypsin / Benzamidine ~

Receptor: **Bos taurus beta-trypsin**

- Produced in pancreas, cleaves peptide chains (→ protease)
- Widely used for protein digestion in biotechnology

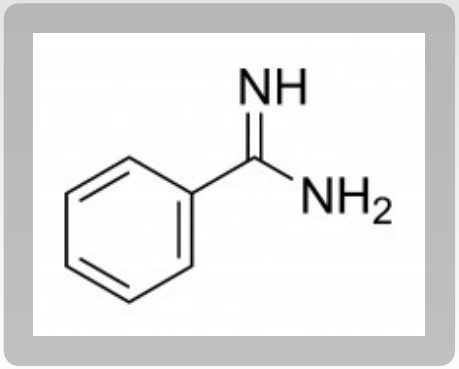


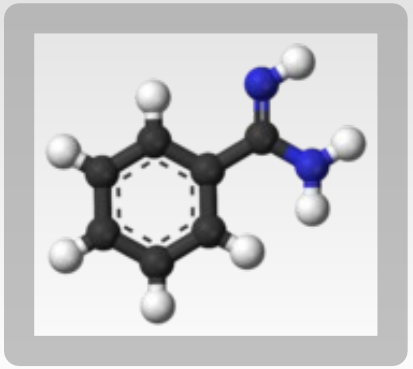
Introduction

~ Example Molecules: Trypsin / Benzamidine ~

Ligand: **Benzamidine**

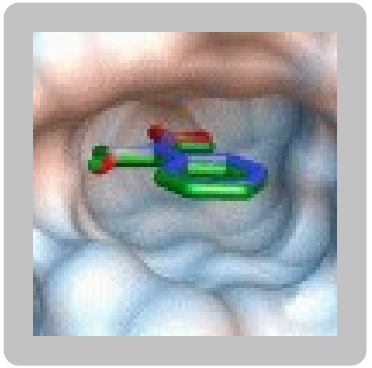
- Trypsin inhibitor
- Used e.g. in protein crystallo-graphy to avoid degradation





Advantages of trypsin & benzamidine as an example:

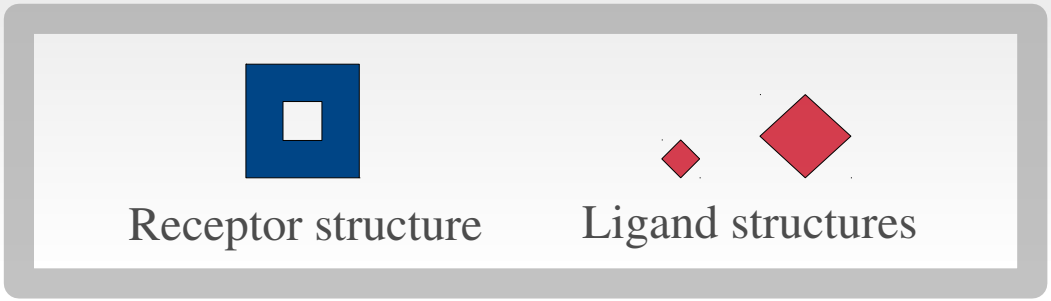
- Ligand benzamidine is rather rigid
  (→ software does not have to take rotatable bonds into account, initially)



Idea

~ Pre-Docking Filter ~

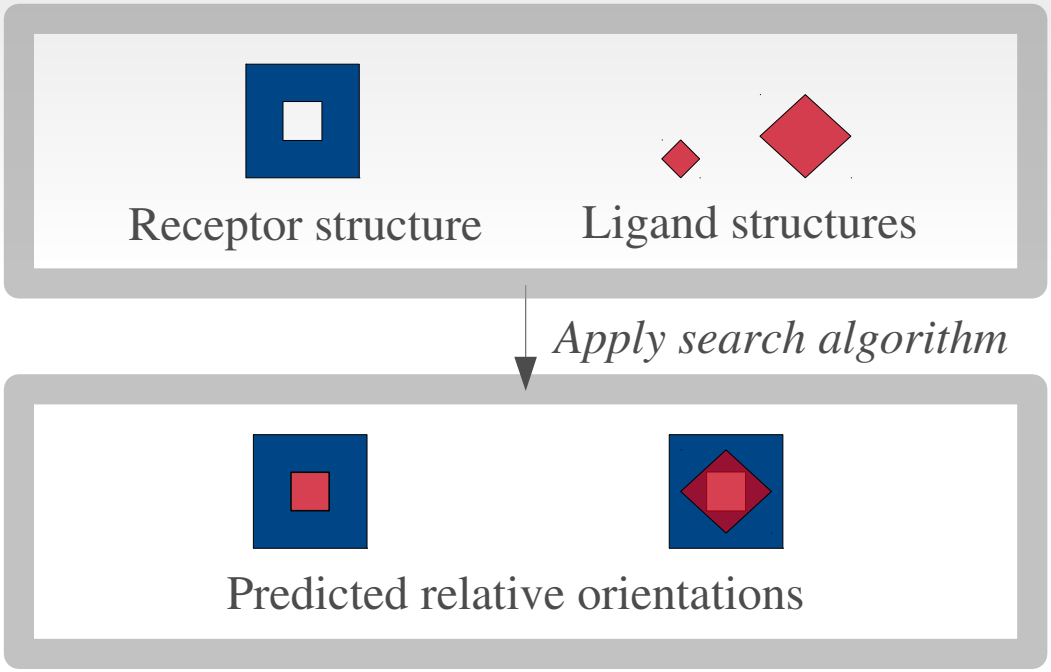
Why try to dock molecules that are much too large?



Idea

~ Pre-Docking Filter ~

Why try to dock molecules that are much too large?



Idea

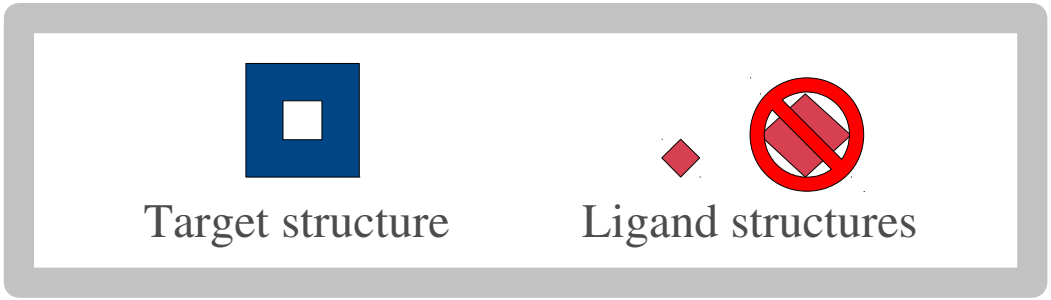
~ Pre-Docking Filter ~

Why try to dock molecules that are much too large?

→ Exclude non-matching ligands from docking runs (save runtime!)

Pre-docking filter performing vague shape-matching:

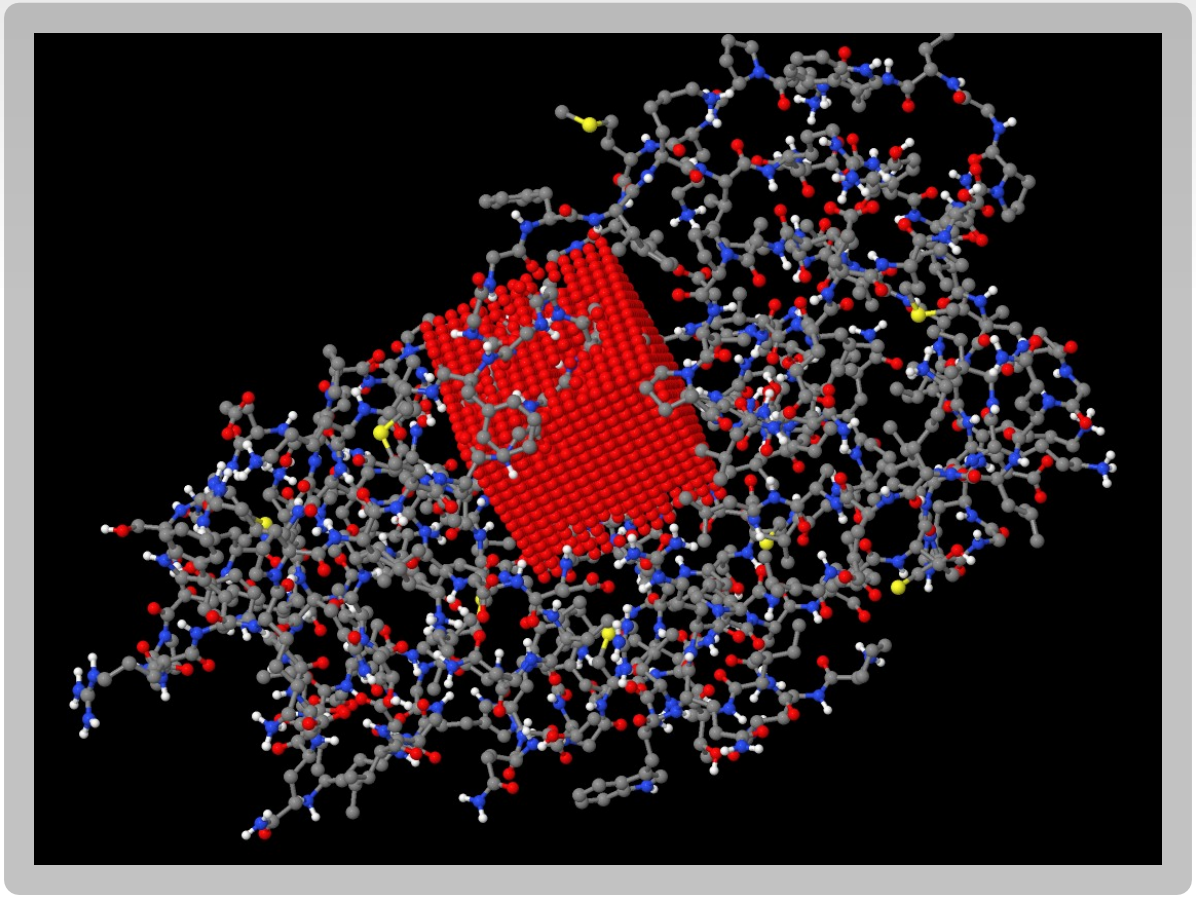
1. Evaluate binding pocket dimensions (create negative imprint)
2. Overlap pocket and ligand axes (principal component analysis)
3. Apply tangent distance algorithm



Idea

~ Pre-Docking Filter ~

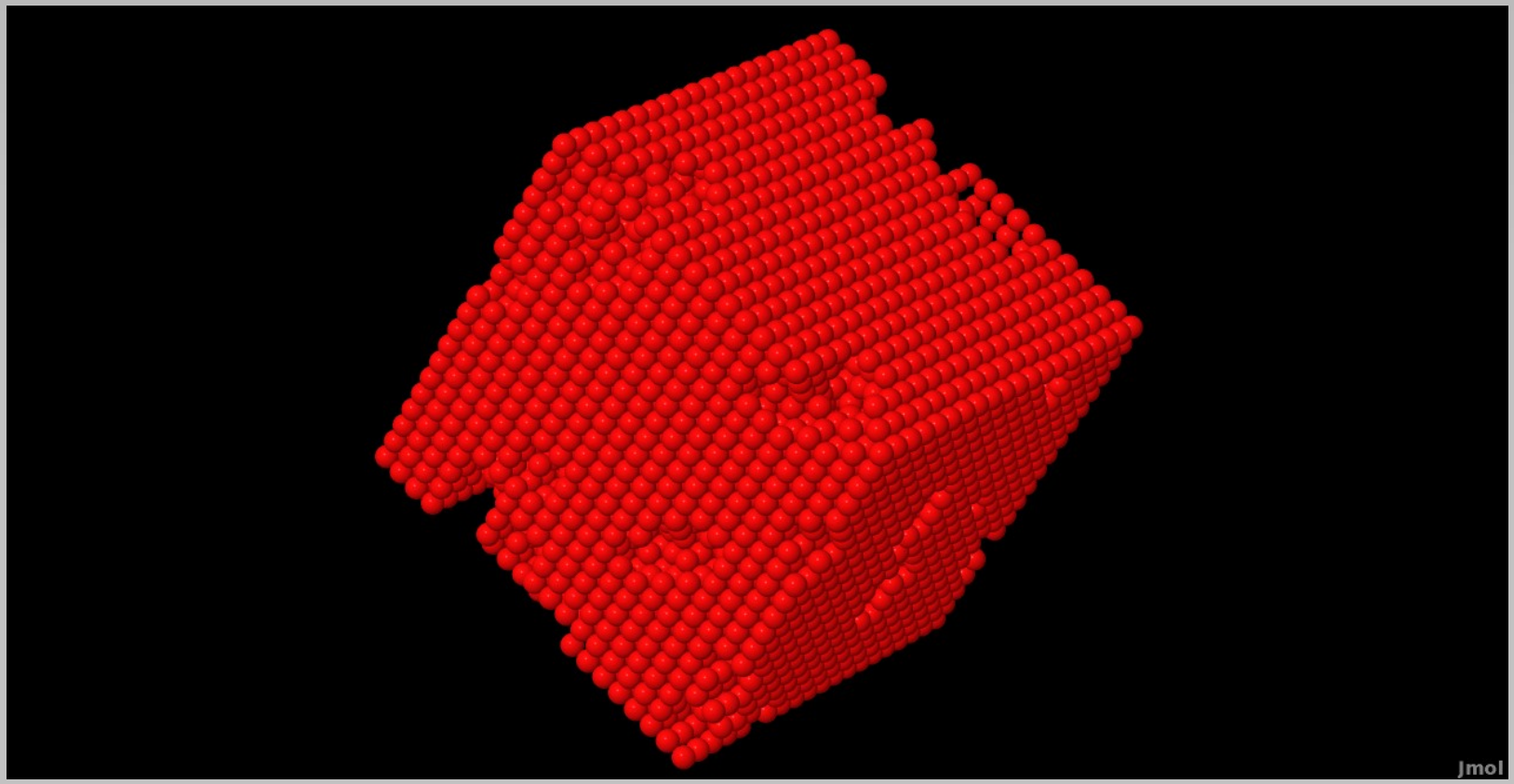
Evaluate binding pocket dimensions (create negative imprint)



Idea

~ Pre-Docking Filter ~

Evaluate binding pocket dimensions (create negative imprint)

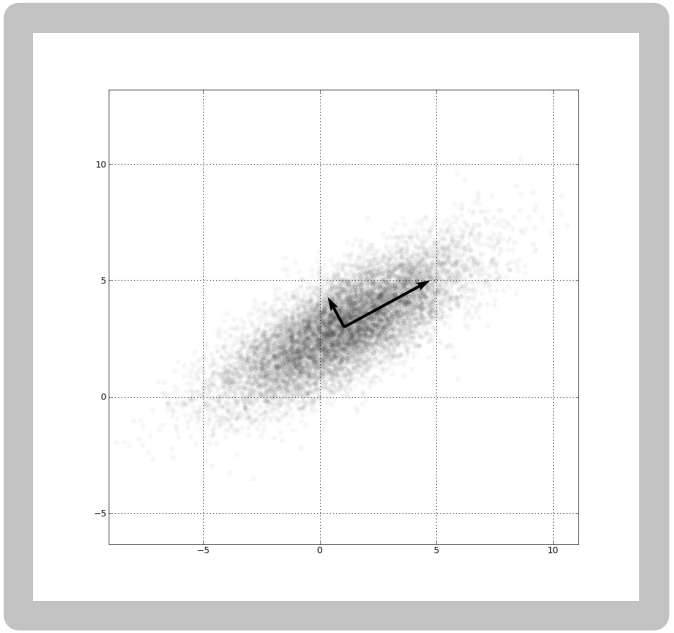


Idea

~ Pre-Docking Filter ~

Overlap pocket and ligand along centroidal axes: PCA

- Algorithm to find the longest stretches of our molecules
- Procedure: Create matrix from coordinates, compute Eigenvectors, Eigenvector with highest Eigenvalue equates to longest stretch



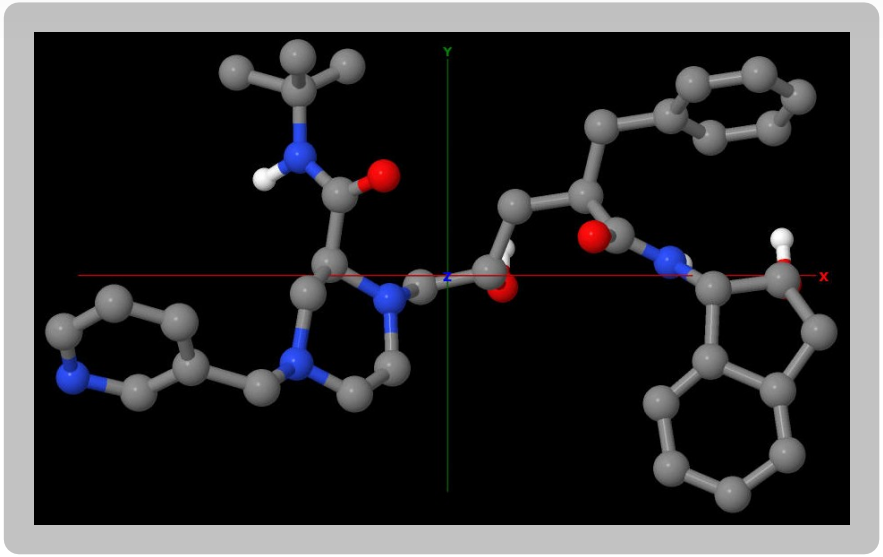
Example: Multivariate Gaussian distribution

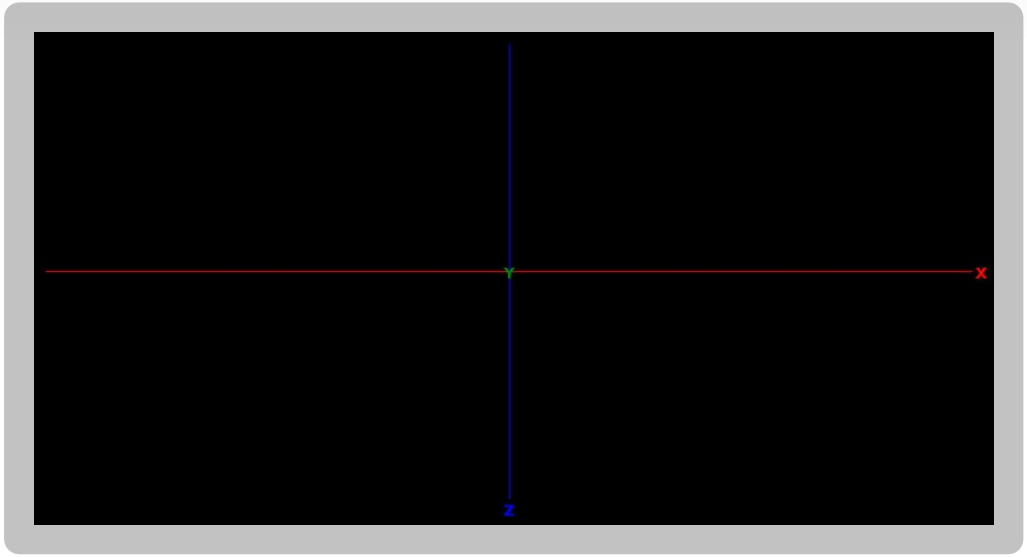
Idea

~ Pre-Docking Filter ~

Overlap pocket and ligand along centroidal axes: → Ligand

1. Where is the longest stretch (x, y, z)? → $(x > y)$ && $(x > z)$ → x



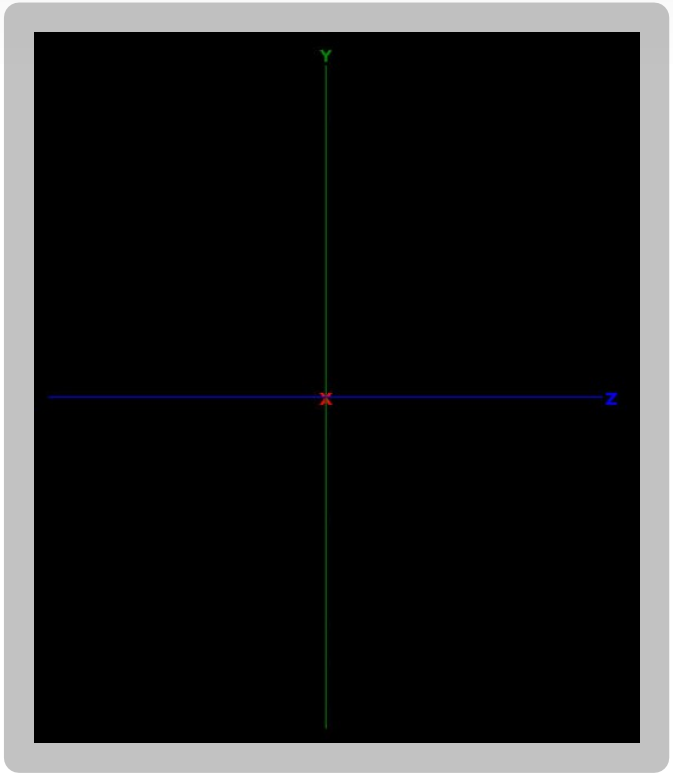


Idea

~ Pre-Docking Filter ~

Overlap pocket and ligand along centroidal axes: → Ligand

1. Where is the longest stretch (x, y, z)? → $(x > y)$ && $(x > z)$ → x
2. Where is the second longest stretch (y, z)? → $(y > z)$ → y

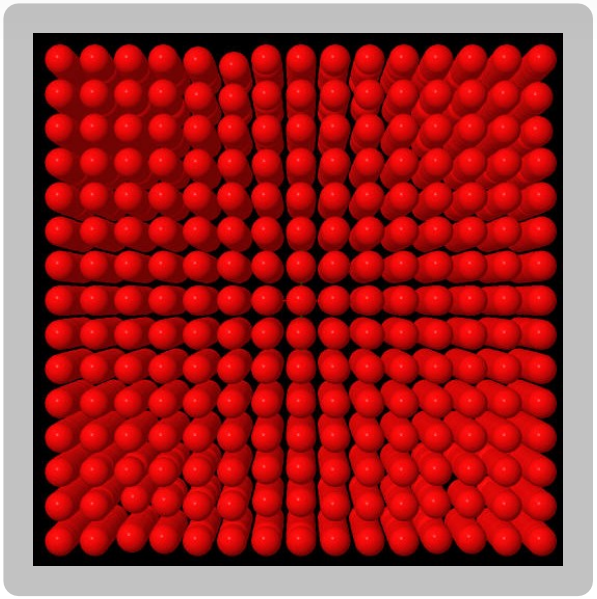


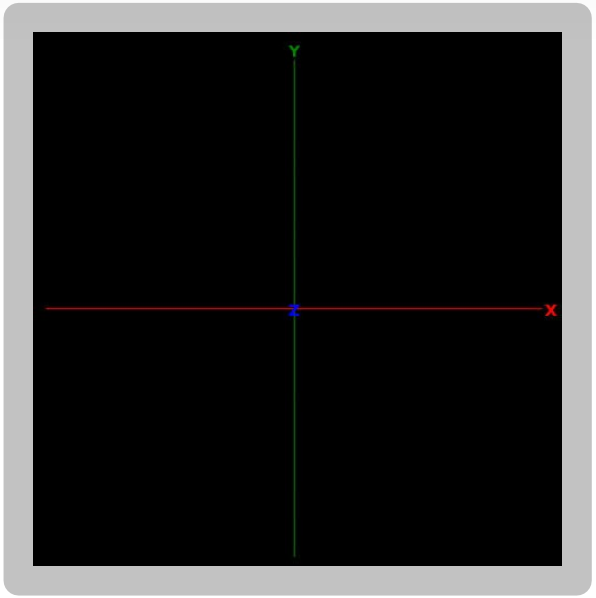
Idea

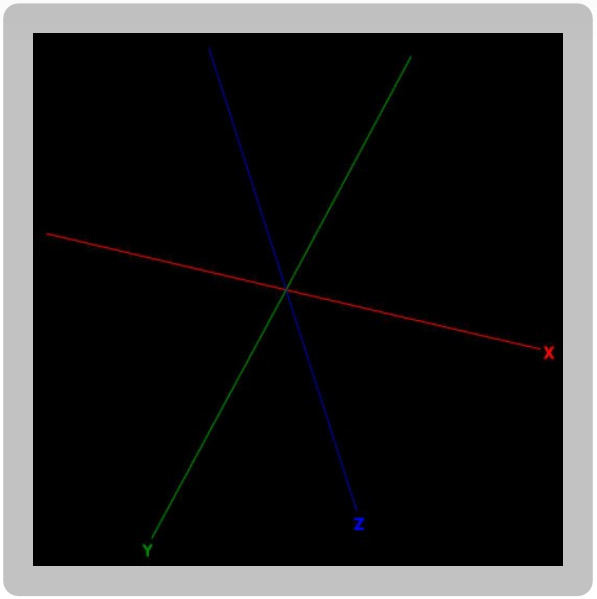
~ Pre-Docking Filter ~

Overlap pocket and ligand along centroidal axes: → Pocket

1. Where is the longest stretch *(x, y, z)*? → All same length → *x*
2. Where is the second longest stretch *(y, z)*? → Same length → *y*







Idea

~ Pre-Docking Filter ~

Overlap pocket and ligand along centroidal axes:

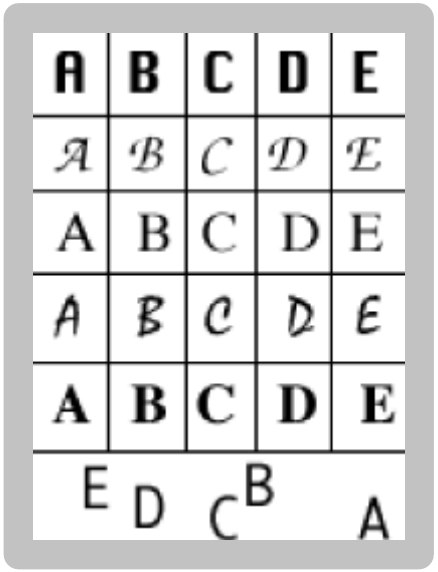
1. Superimpose the gravity centers of both molecules
2. Rotate the ligand, so that pocket and ligand have their longest stretches in the same direction (p: z, li: $z \rightarrow$ no rotation necessary)
3. Rotate the ligand, so that pocket and ligand also have their second longest stretches in the same direction (p: y, li: $y \rightarrow$ no rotation necessary)

Idea

~ OCR-Based Algorithm ~

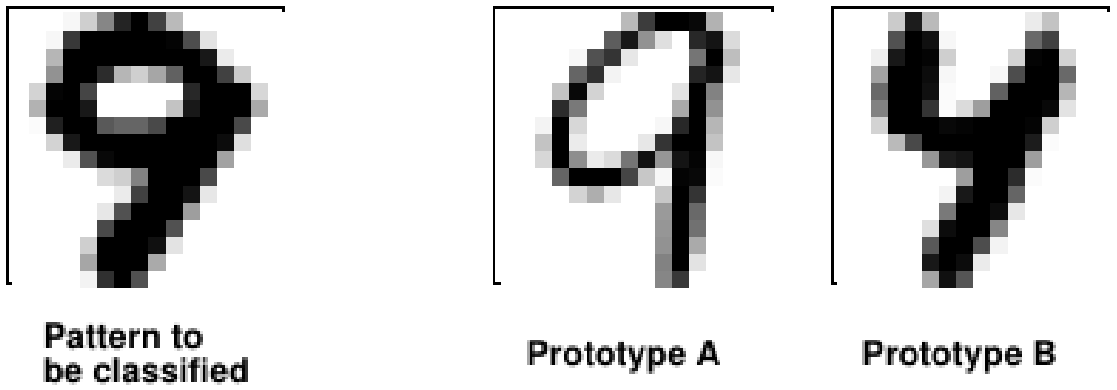
Tangent distance:

- Used in optical character recognition (OCR)
- Short definition: Shortest distance between two planes
- Linear approximations to arbitrary transforms
- More appropriate in OCR than Euclidean distance
- Context:
  - ➔ Classification / pattern recognition
    - ➔ Discriminative methods
      - ➔ Distance-based methods
        - ➔ Tangent distance



Idea

~ OCR-Based Algorithm: Euclidean Distance ~



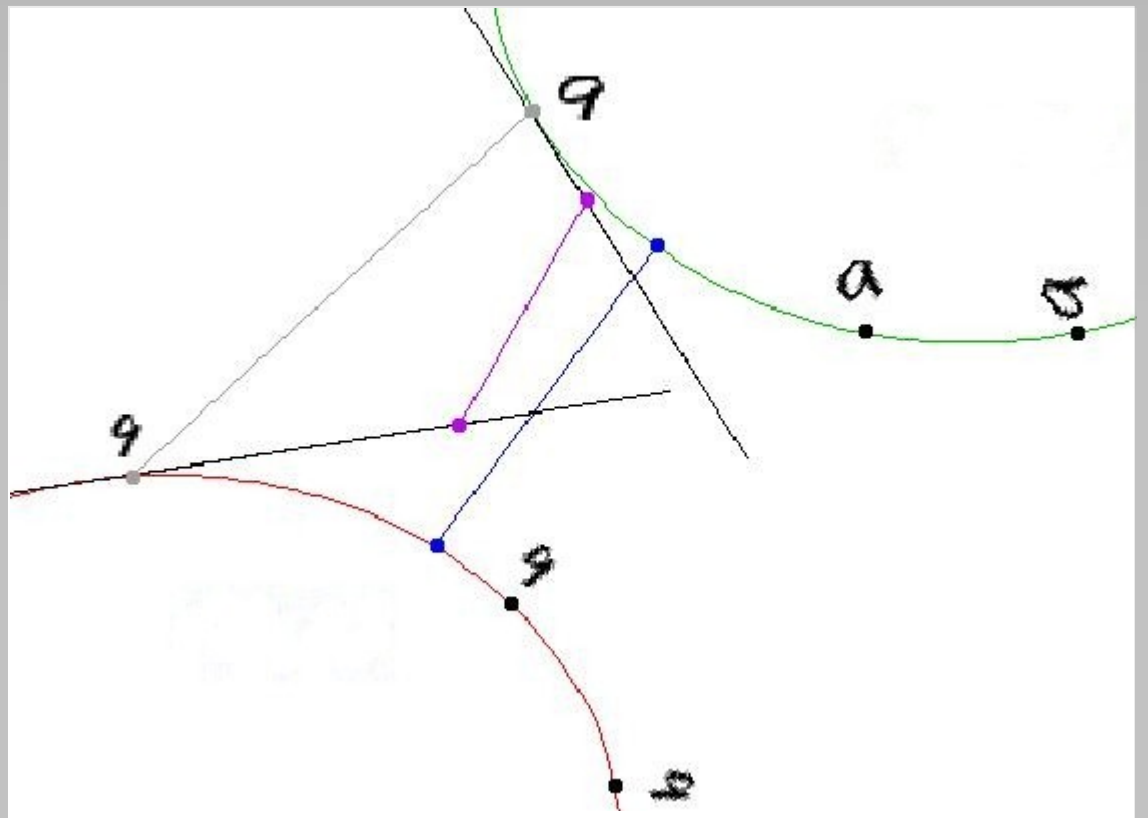
Simard, LeCun, Denker, Victorri (1998): Transformation Invariance in Pattern Recognition – Tangent Distance and Tangent Propagation. Neural Networks: Tricks of the Trade. Springer.

Task:	Classify digit using database of known prototypes
Data format:	256-D pattern vector (16*16 pixel grayscale image)
Euclidean dist.:	Sum of squares of pixel-to-pixel difference
Result:	Pattern is more similar to B (obviously wrong!)
Problem:	Inappropriate for „allowed“ transformations.

Idea

~ OCR-Based Algorithm: Distances ~

Illustration of real distance, Euclidean distance, and tangent distance:



Mazack (2009): Algorithms for Handwritten Digit Recognition. Master's thesis. Western Washington University.

Idea

~ OCR-Based Algorithm: Analogy ~

Analogy between OCR and docking problem set:
Existence of „allowed“ transformations (invariants) in both

OCR	Docking
Translation in x direction	Translation in x direction
Translation in y direction	Translation in y direction
Rotation	Translation in z direction
Shear	Rotation in x direction
Scale	Rotation in y direction
Line thickness transformation	Rotation in y direction

Idea

~ OCR-Based Algorithm: Tangent Distance ~

First part of the algorithm:
Construct classifier for given d-dimensional prototypes

- For each stored prototype x':
  - Perform each of the transformations $t_i(x'; a_i)$ on it (a_i = e.g. a small angle of rotation)
  - Construct tangent vector TV_i for each transformation:

$$TV_i = t_i(x'; a_i) - x'$$

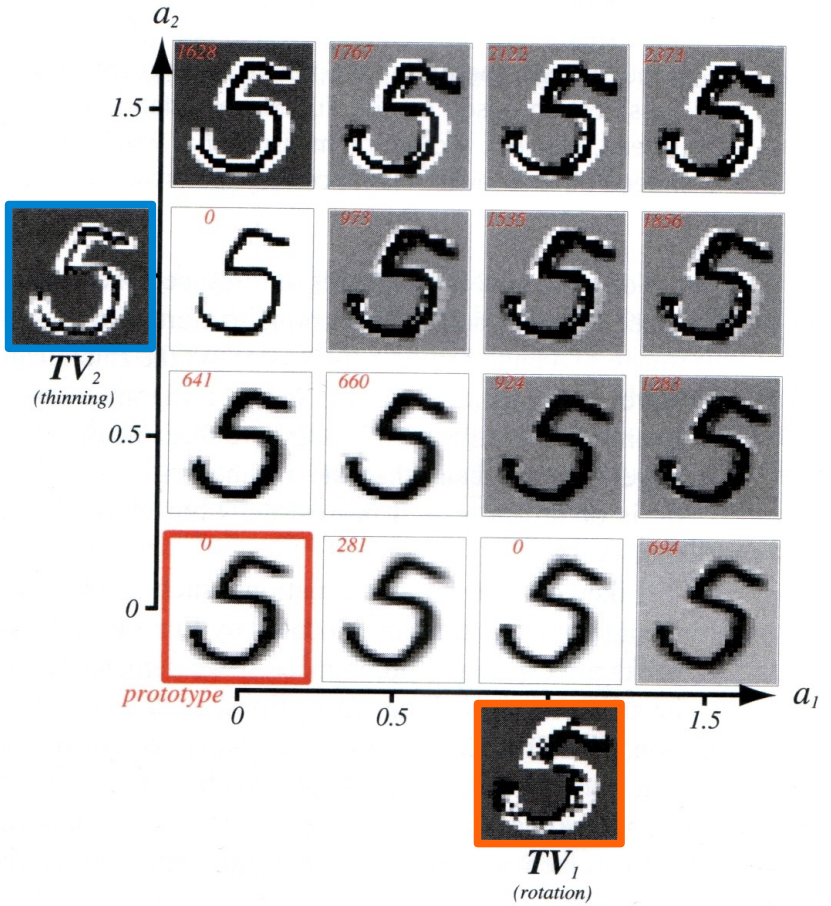
  - Construct $r * d$ matrix T consisting of the r TV's at x'

Idea

~ OCR-Based Algorithm: Tangent Distance ~

Matrix T for a handwritten 5, accounting for line thinning and rotation:

- Prototype
- TV_1: Prototype + rotation
- TV_2: Prototype + thinning
- Other 5's: Prototype + linear combination of TV_1 and TV_2 with coefficients a_1 and a_2



Duda, Hart, Stork (2000): Pattern Classification (2nd ed). Wiley-Interscience: 190

Idea

~ OCR-Based Algorithm: Tangent Distance ~

Match **OCR algorithm** to **docking problem**:
Construct classifier for given d-dimensional prototypes

- For each stored **prototype x'** (= **ligand structures**):
  - Perform each of the **transformations $t_i(x'; a_i)$** on it (a_i = e.g. a small angle of rotation) (= **3x rotation → drop shadows**)
  - Construct tangent vector TV_i for each transformation:

$$TV_i = t_i(x'; a_i) - x'$$

  - Construct $r * d$ matrix T consisting of the r TV's at x'

Idea

~ OCR-Based Algorithm: Tangent Distance ~

Second part:
Compute tangent distance *TD*

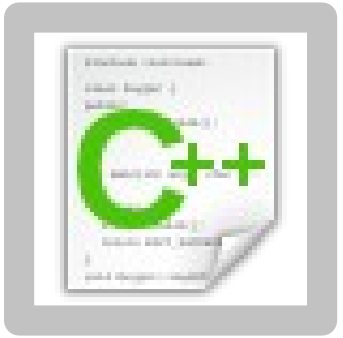
- Compute tangent distance from x' to test point x:
  $TD(x', x) = min_a[||(x' + Ta) - x||]$ *
- Find optimizing value of a

*Euclidean norm: $\|\boldsymbol{x}\| := \sqrt{x_1^2 + \cdots + x_n^2}$.

Implementation

~ Status Quo ~

1. Retrieve data from input
    1. Protein as PDB or PDBQT file → Coordinates, atom types
    2. Ligand as PDBQT file → Coordinates, atom types
    3. Grid as GPF file → Box constraints, grid spacing
2. Fill grid box with atoms (negative imprint)
3. Arbitrarily change ligand coordinates
4. Generate output
    1. Visualization: Superimposition of protein and ligand
    2. File writing: Ligand with changed coordinates (PDBQT format)



Implementation
~ Future Work ~

Coding:

1. Overlap pocket and ligand along centroidal axes
2. Apply tangent distance algorithm
3. If method works for rigid ligands, extend to flexible ligands

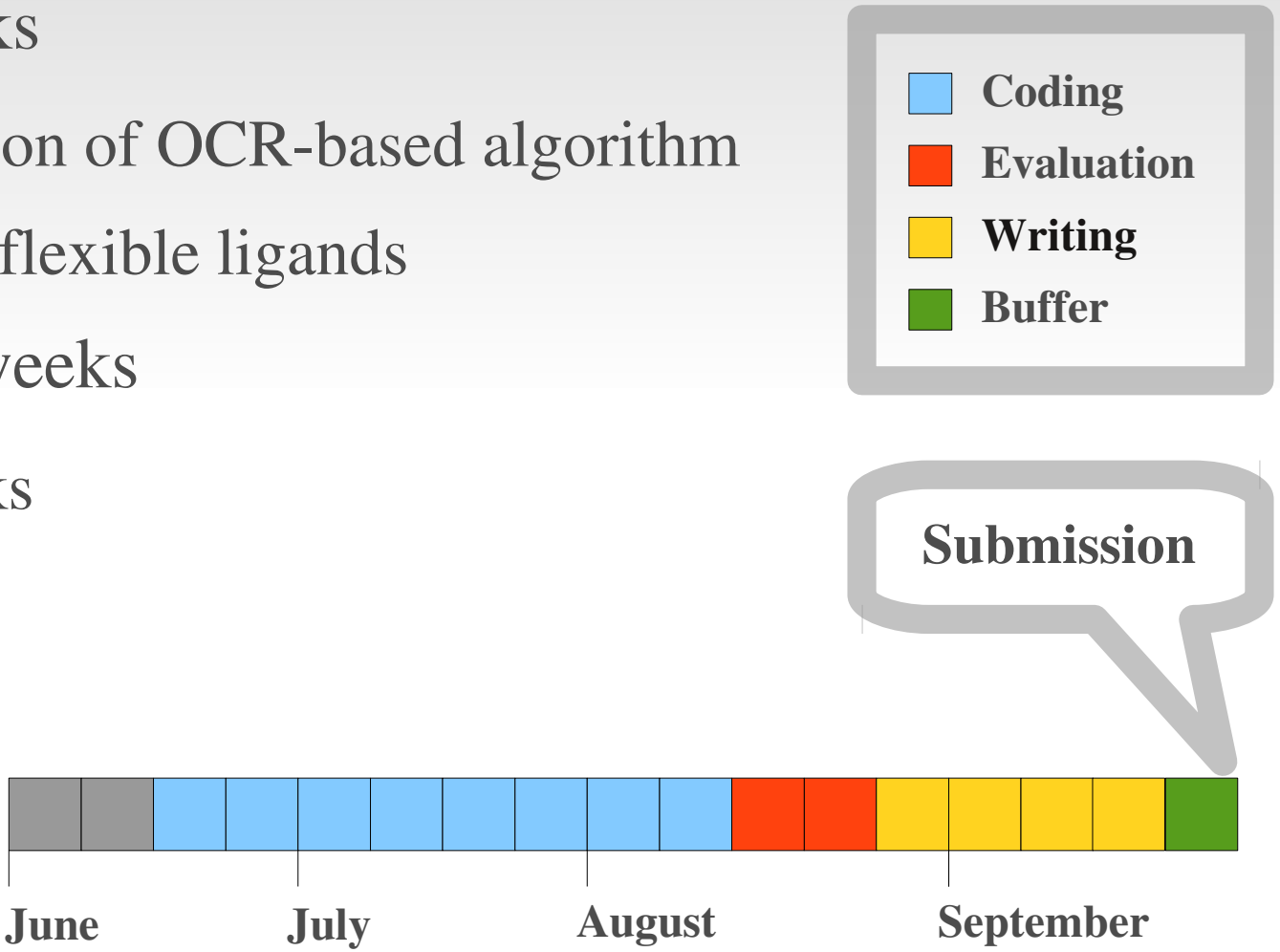
Evaluation: RMSD values of re-docking

- Run tests with gold standard set including decoys
- Compare runtime of AutoDock with runtime of AutoDock plus Pre-Docking

Implementation

~ Schedule ~

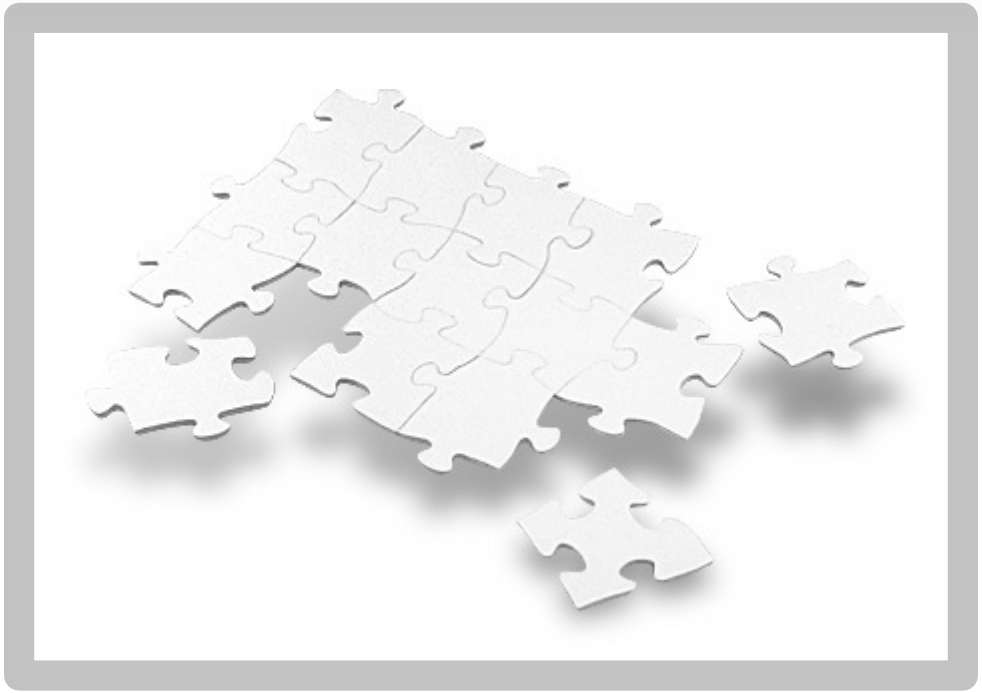
- Coding: 8 weeks
  - ➔ Implementation of OCR-based algorithm
  - ➔ Extension to flexible ligands
- Evaluation: 2 weeks
- Writing: 4 weeks
- Buffer: 1 week



Outlook

~ Envisaged Extensions ~

- Web server: Public availability
- Input formats: Broader support
- Output format: Compatibility with other docking tools



~ Thank you, ... ~

- dear audience, for your attention!
- PD Dr. Michael Hutter, for providing the idea and your support!
- Prof. Dr. Volkhard Helms, for an inspiring discussion and some useful hints!

Discussion

~ Docking Basics: AutoDock ~

- **File preparation:** add polar H's, partial charges, and atom types; compute torsional degrees of freedom (ligand only); create PDBQT files; define grid box; define docking parameters
- **AutoGrid:** Embed protein in 3-D grid; for each atom type in the ligand: place probe atom at each grid point; assess interaction energy between atom and protein; assign energy to grid point using AMBER force field; create grid map files
- **AutoDock:** Dock ligand to set of grid maps; available algorithms: different genetic algorithms, simulated annealing (SA)
- **Analysis:** Cluster and / or visualize results

Discussion

~ Docking Utilities: PASS ~

Predictive tool for binding site identification