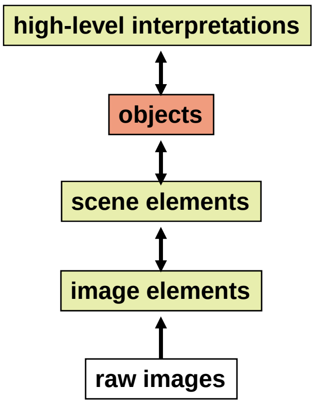
Object Recognition



Object recognition

- object recognition is a typical goal of image analysis
- object recognition includes

- object identification
  recognizing that one object instance is (physically) identical to another object instance
- object classification
  assigning an object to one of a set of predetermined classes
- object categorization
  assigning an object to an object category (as proposed in biological vision)

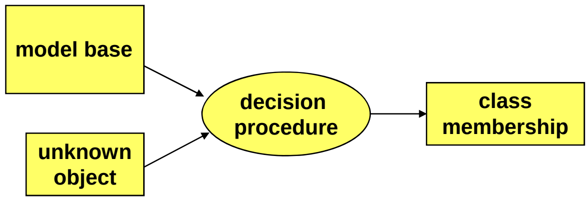
1

The Chair Room

About Model-based Recognition

"model" = generic description of a class of objects

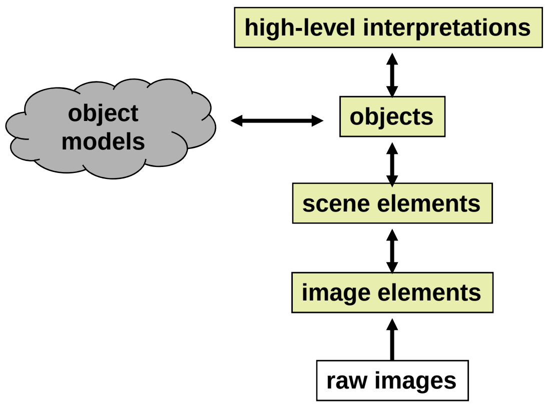
- explicit representation of object properties (as opposed to decision procedures which incorporate class properties implicitely)
- generic (class-independent) decision procedure
- reusable and incremental model bases
- no strict correspondence with biological vision



3

Model-based Object Recognition

How to classify objects based on a generic description.

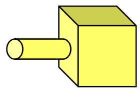


4

3D Models vs. 2D Models

1. Requirement:
Object models must represent invariant class properties
=> 3D models, properties independent of views

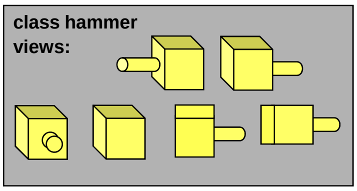
e.g.



class hammer
is-a aggregat
has-parts part1, part2
is-a part1 cube
is-a part2 cylinder
coaxially-connected part1 part2

2. Requirement:
Object models must support recognition
=> 2D models, view-dependent properties

Modern approaches to object recognition are typically a compromise of Requirements 1 and 2.

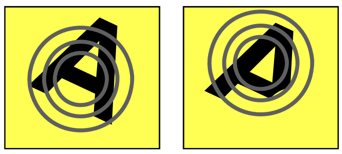


5

Holistic Models vs. Component Models

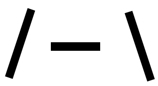
Holistic ("global") models:
- properties refer to complete object
- local disturbances may jeopardize all properties

e.g. area, polar signature, NN classifier



Component models:
- object model is described by components and relations between components
- properties refer to individual components
- local disturbances affect only local properties

Example of components:



6

3D Shape Models

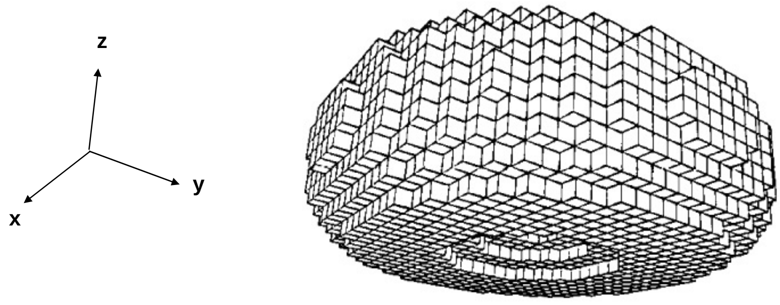
Several 3D shape models have been developed for engineering applications:
- 3D space occupancy
- Oct-trees
- CSG ("Constructive Solid Geometry") models
- 3D surface triangulation

In general, pure 3D models are not immediately useful for Computer Vision because they do not support recognition.

In support of recogntion, special 3D models have been developed which include view-related information:
- EGI ("Extended Gaussian Image")
- Generalized cylinders

7

3D Space Occupancy Model

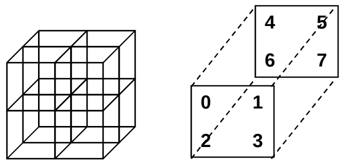


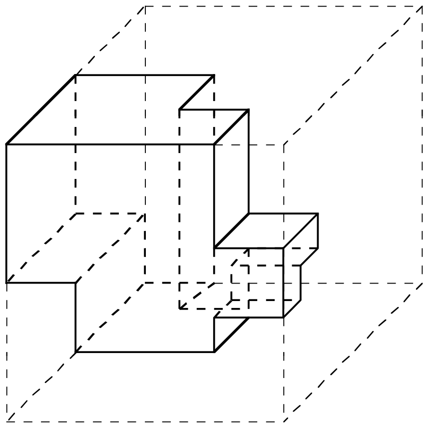
3D shape represented by cube primitives
- useful for highly irregular shapes (e.g. medical domain)
- useful for robotics applications (e.g. collision avoidance)
- interior cubes do not provide information relevant for views
- no explicit surface properties (e.g. surface normals)

8

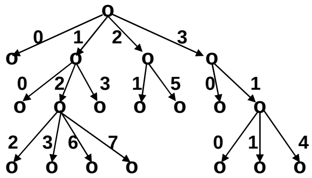
Oct-trees

- hierarchical 3D shape model
- analog to 2D quad-trees
- each cube is recursively decomposed into 8 subcubes
- access via numbering code



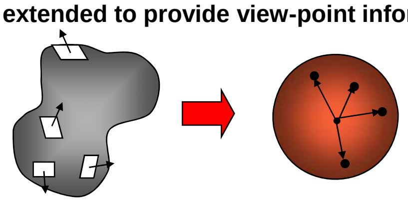


Oct-tree for example (right):



Extended Gaussian Image (EGI)

- 3D shape model based on a surface slope histogram
- extended to provide view-point information for recognition



example of a 3D surface

entries on Gaussian sphere

Each entry represents information for a particular 3D slope and viewing direction:

1. quotient of surface area with this slope and total surface area
2. quotient of visible 3D surface area and area of its 2D projection (as viewed from this direction)
3. direction of axis of minimal inertia of 2D projection of visible surface (as viewed from this direction)

Recognition with EGI Models

Properties of EGIs:

- scale invariant
- rotation of object corresponds to equivalent rotation of EGI
- convex shapes can be uniquely reconstructed
  In particular: A convex polyhedron can be reconstructed from the set of orientations and associated areas $\{(o_1\ a_1)\ (o_2\ a_2)\ \dots\ (o_N\ a_N)\}$
- In general, reconstruction requires an iterative algorithm

Recognition procedure:

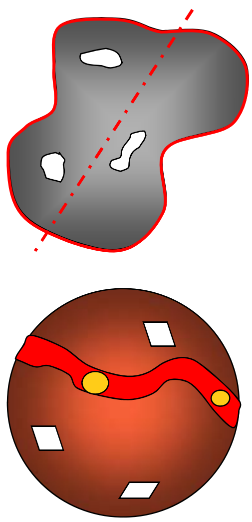
It is assumed that 3D surface normals are determined
(e.g. by laser measurements)

- determine direction of axis of minimal inertia
- determine projected surface area
- determine patches of (approximately) constant 3D surface inclination
- constrained search for models which match the measurements

11

Illustration of EGI Recognition Procedure

1. Determine direction of axis of minimal inertia
   => locations on EGI with corresponding entries
2. Determine projected surface area
   => subset of locations determined by 1)
3. Determine patches of constant 3D surface inclination
   => rotate EGI into viewing direction of 1) and 2), compare surface area with corresponding entries
4. Constrained search for models which match the measurements
   => if 1) to 3) do not match, choose other models

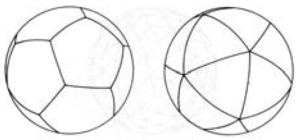


12

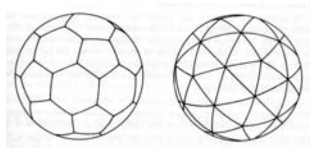
Discretizing the Surface of a Sphere

For a computer representation of an Extended Gaussian Sphere, the surface of the sphere has to be tessellated into patches of approximately equal size.

geodesic tessellation has undesirable properties



dodecahedron (left) and icosahedron (right) provide tessallations with 12 and 20 cells, respectively



truncated icosahedron (left) provides 12 pentagonal and 20 hexagonal faces, pentakis dodekahedron (right) provides 60 triangular faces

Further refinements can be obtained by triangularization within the cells of a regular or semiregular solid.

13

Representing Axial Bodies

Picasso´s "Rites of Spring" shows bodies composed of roughly cylindrical and cone-shaped pieces.

What representations capture the inherent restrictions of such shapes?

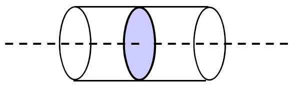
(a)

14

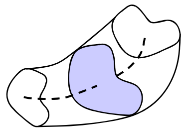
Generalized Cylinders

3D surface determined by sweeping a closed curve along a line

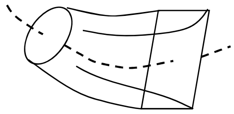
ordinary <u>cylinder</u> swept out by a circle along a straight line



<u>generalized cone</u> swept out by an arbitrary planar cross section, varying in size, along a smooth axis (Binford 71)



<u>generalized cylinder</u> swept out by a closed curve at an angle to a curved axis subject to a deformation function



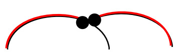
Generalized cones were used in ACRONYM (Brooks et al. 79) to model mainly artificial objects, e.g. airplanes. Under certain conditions, the 3D surface may be reconstructed from the contours of many views.

15

Conditions for 3D Reconstruction from Contours

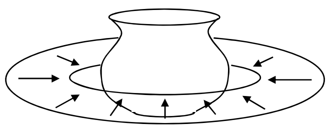
1. Each line of sight touches the body at a single point
   => we see a "contour generator"
2. Nearby points on the contour in the image are also nearby in 3D (with only few exceptions)

   2 distant points projected onto nearby contour points



3. The contour generator is planar
   => hence inflections of the contour in 2D correspond to inflections in 3D

If a surface is smooth and if conditions 1 to 3 hold for all viewing directions in any plane, then the viewed surface is a generalized cone. (Marr 77)



16

Object Recognition using Relational Matching

17

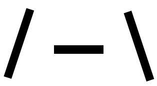
Relational Models

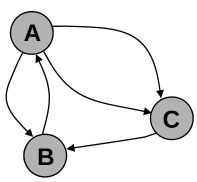
Relational models describe objects (object classes) based on parts (components) and relations between the parts

Relational model can be represented as structure with nodes and edges:

Nodes: parts with properties

e.g.





Edges: relations between parts

e.g.

- obtuse-angle
- 2cm-distance
- touches
- surrounds
- left-of
- after

18

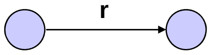
Relations between Components

unary relation: **property**

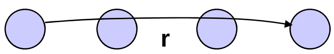
n-ary relation: **relation, constraint**

Graphical representation

binary relation:



n-ary relation:



"hypergraph"

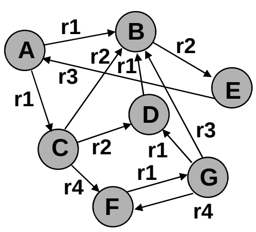
19

Object Recognition by Relational Matching

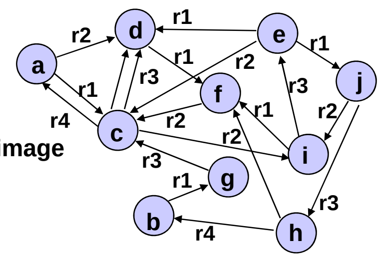
Principle:

- **construct relational model(s) for object class(es)**
- **construct relational image description**
- **compute R-morphism (best partial match) between image and model(s)**
- **top-down verification with extended model**

model



image



20

Compatibility of Relational Structures

Different from graphs, nodes and edges of relational structures may represent entities with rich distinctive descriptions.

<u>Example</u>: **nodes = image regions with diverse properties**
edges = spatial relations

1. Compatibility of nodes

An image node is compatible with a model node, if the properties of the nodes match.

2. Compatibility of edges

An image edge is compatible with a model edge, if the edge types match.

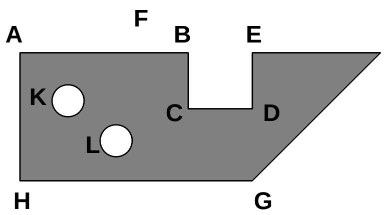
3. Compatibility of structures

A relational image description B is compatible with a relational model M, if there exists a bijective mapping of nodes of a partial structure B´of B onto nodes of a partial structure M´of M such that

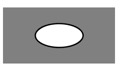
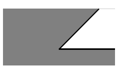
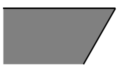
- corresponding nodes and edges are compatible
- M is described by M´ with sufficient completeness

Example of a Relational Model (1)

shape to be recognized:

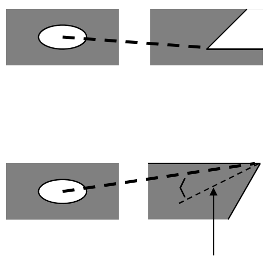


primitive descriptive elements (nodes) **properties**

	**hole**	**t** **f** **a**	**type T1** **area** **axes relation**
	**interior corner**	**t** **w**	**type T2** **angle**
	**exterior corner**	**t** **w**	**type T3** **angle**

Example of a Relational Model (2)

relations between primitive descriptive elements (edges)



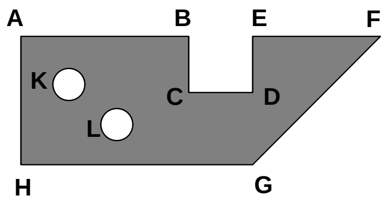
bisector of angle

...	
d10	distance 10 ± 1
d12	distance 12 ± 1
d14	distance 14 ± 1
...	

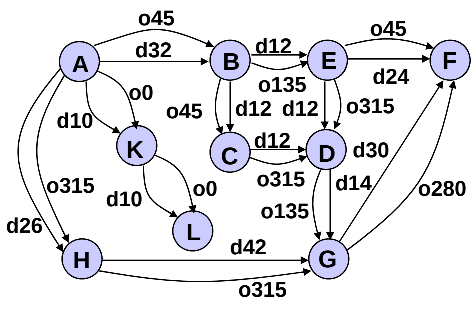
...	
o10	orientation 10 ± 5
o20	orientation 20 ± 5
o30	orientation 30 ± 5
...	

23

Example of a Relational Model (3)



A t T3 w 90	E t T3 w 90	K t T1 f 48 a 1
B t T3 w 90	F t T3 w 45	L t T1 f 48 a 1
C t T2 w 90	G t T3 w 135	
D t T2 w 90	H t T3 w 90	

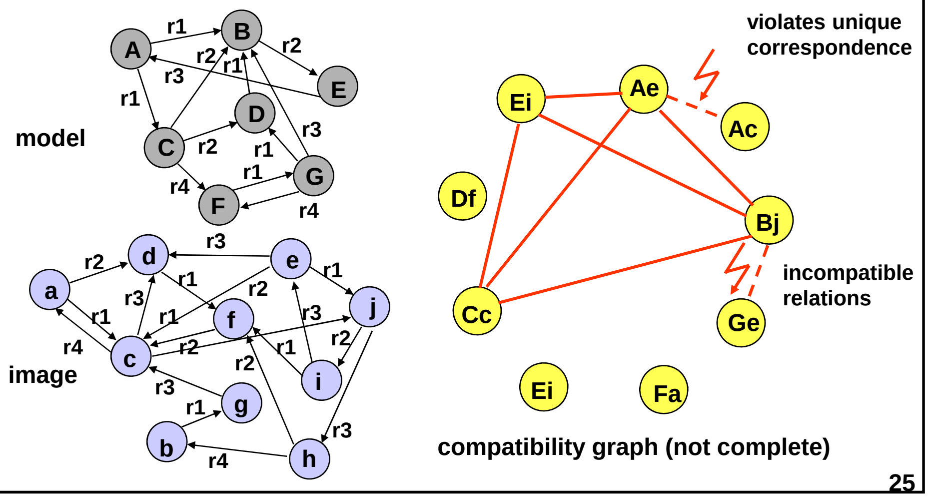


(not all edges are shown)

24

Relational Match Using a Compatibility Graph

nodes of compatibility graph = pairs with compatible properties
edges of compatibility graph = compatible pairs
cliques in compatibility graph = compatible partial structures



25

Finding Maximal Cliques

clique = complete subgraph

Find maximal cliques in a given compatibility graph

Algorithms are available in the literature, e.g.

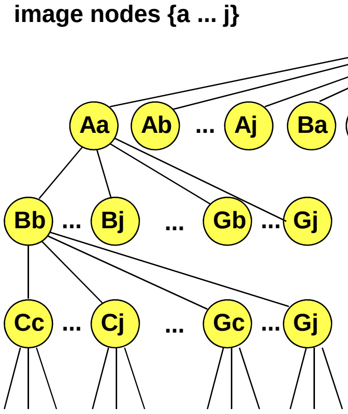
Bron & Kerbusch, Finding all Cliques of an Undirected Graph, Communications of the ACM, Vol. 16, Nr. 9, S. 575 - 577, 1973.

- Complexity is exponential relative to number of nodes of compatibility graph
- Efficient (suboptimal) solutions based on heuristic search

26

Relational Matching with Heuristic Search

Stepwise correspondence search between model nodes {A ... G} and image nodes {a ... j}



- **quality function evaluates partial matches**
- **accept a partial match if**
  quality > acceptance threshold
- **refute a partial match, if**
  quality < refutation threshold

27