

Knowledge-based Computer Vision

What is Computer Vision?

Computer Vision is the academic discipline dealing with task-oriented reconstruction and interpretation of a scene by means of images.

scene:	section of the real world stationary (3D) or moving (4D)
image:	view of a scene projection, density image (2D) depth image (2 1/2D) image sequence (3D)
reconstruction and interpretation:	computer-internal scene description quantitative + qualitative + symbolic
task-oriented:	for a purpose, to fulfill a particular task context-dependent, supporting actions of an agent

Basic System Architecture

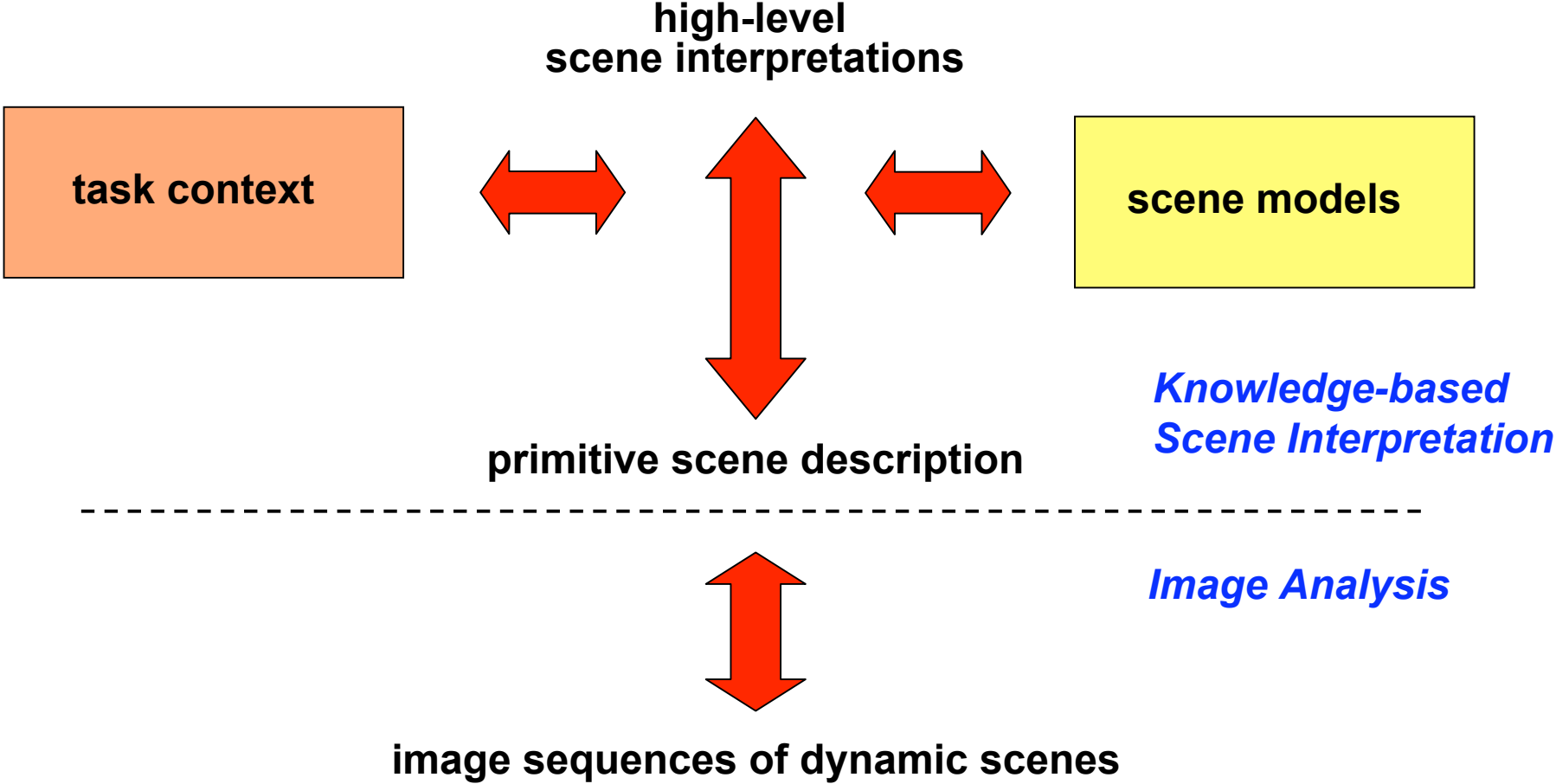


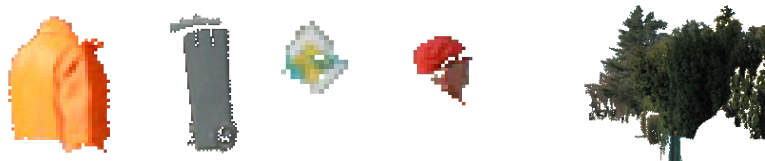
Illustration of Scene Interpretation

Typical results of Image Analysis:

- spatial configurations of interest points and their surroundings



- more or less meaningful regions



Typical result of Scene Interpretation:

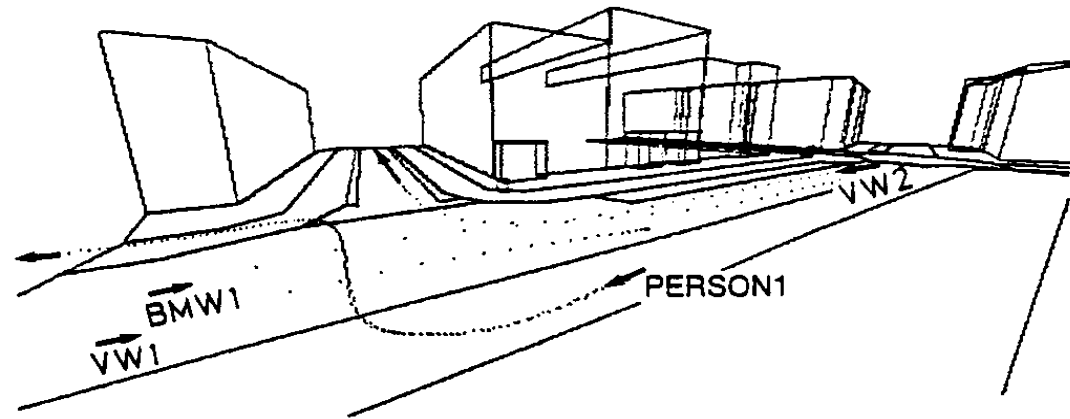
"There is garbage collection in a street,
and a mailman distributes mail"



Computer Vision research has dealt
almost exclusively with Image Analysis,
(except of a few unperturbable
researchers in Germany ...)

Historical Example

Interpretation of a simulated street scene with the system NAOS (Neumann & Novak 1986)



English paraphrase of automatically generated description:

The scene contains four moving objects: three cars and a pedestrian.

A VW **drives** from the Alte-Post to the front of the FBI. It **stops**.

Another VW **drives** towards Dammtor. It **turns off** Schlueterstrasse. It **drives** on Bieberstrasse towards Grindelhof.

A BMW **drives** towards Hallerplatz. While doing so, it **overtakes** the VW which has stopped, before Bieberstrasse. The BMW **stops** in front of the traffic lights.

The pedestrian **walks** towards Dammtor. While doing so, he **crosses** Schlueterstrasse in front of the FBI.

State-of-the-art Example

Ph.D. research of Somboon Hongeng (2003)



Recognition of an assault



Recognition of a theft

Aggregates for Scene Interpretation

What kind of concepts must be represented for scene interpretation?

Concepts for

- **object constellations**
e.g. laid-table, kitchen, parking ground, town
- **activities, events, episodes**
e.g. operating a CD-player, one car overtaking another, playing soccer

Typical scene interpretation concepts describe entities composed of sub-entities related to each other in space and time. We call such entities **"aggregates"**.

Aggregate Structure

Basic structure of a frame-based representation of an aggregate concept:

aggregate name
parent concepts
external properties
parts
constraints between parts

- *aggregate name* contains a symbolic ID
- *parent concepts* contains IDs of taxonomical parents
- *external properties* provide a description of the aggregate as a whole
- *parts* describe the subunits out of which an aggregate is composed
- *constraints* specify which relations must hold between the parts

Occurrence Model for Overtaking

name: overtake
:local-name ov

parents: :is-a occurrence-model

arguments: (?veh1 :is-a vehicle)
(?veh2 :is-a vehicle)

properties: (ue.B ue.E)

parts : (mv1 :is-a (move ?veh1 mv1.B mv1.E))
(mv2 :is-a (move ?veh2 mv2.B mv2.E))
(bh :is-a (behind ?veh1 ?veh2 bh.B bh.E))
(bs :is-a (beside ?veh1 ?veh2 bs.B bs.E))
(bf :is-a (before ?veh1 ?veh2 bf.B bf.E))
(ap :is-a (approach ?veh1 ?veh2 ap.B ap.E))
(rc :is-a (recede ?veh1 ?veh2 rc.B rc.E))

constraints: (ov.B = bh.B)
(ov.E = bf.E)
(ap :during mv1)
(ap :during mv2)
(rc :during mv1)
(rc :during mv2)
(bh :overlaps bs)
(bs :overlaps bf)
(bh :during ap)
(bf :during rc)

Note:

Aggregate format
may vary
according to
expressiveness of
knowledge
representation
language and
syntactic
conventions

Occurrence Model for Placing a Cover

Recognizing table-laying actions



```

name:           place-cover
parents:        :is-a agent-activity
parts:          pc-tp1 :is-a (transport with (tp-obj :is plate))    %transport of a plate
                pc-tp2:is-a (transport with (tp-obj :is saucer))   %transport of a saucer
                pc-tp3 :is-a (transport with (tp-obj :is cup))      %transport of a cup
                pc-cv :is-a cover                                   %cover configuration
properties:     tb, te :is-a timepoint                             %begin and end timepoint of place-cover
constraints:    pc-tp1.tp-ob = pc-cv.cv-pl                         %transport-plate object same as cover-plate
                pc-tp2.tp-ob = pc-cv.cv-sc                         %transport-saucer object same as cover-saucer
                pc-tp3.tp-ob = pc-cv.cv-cp                         %transport-cup object same as cover-cup
                pc-cv.tb ≥ pc-tp1.te                               %cover begins after plate transport
                pc-cv.tb ≥ pc-tp2.te                               %cover begins after saucer transport
                pc-cv.tb ≥ pc-tp3.te                               %cover begins after cup transport
                pc-tp3.tp-te ≥ pc-tp2.tp-te                       %cup transport ends after saucer transport
                tb = pc-tp1.tb min pc-tp2.tb min pc-tp3.tb
                te = pc-tp1.te max pc-tp2.te max pc-tp3.te
                te ≤ tb + 80Δt                                     %place-cover may not last more than 80 time units
    
```

Model for a Cover Configuration

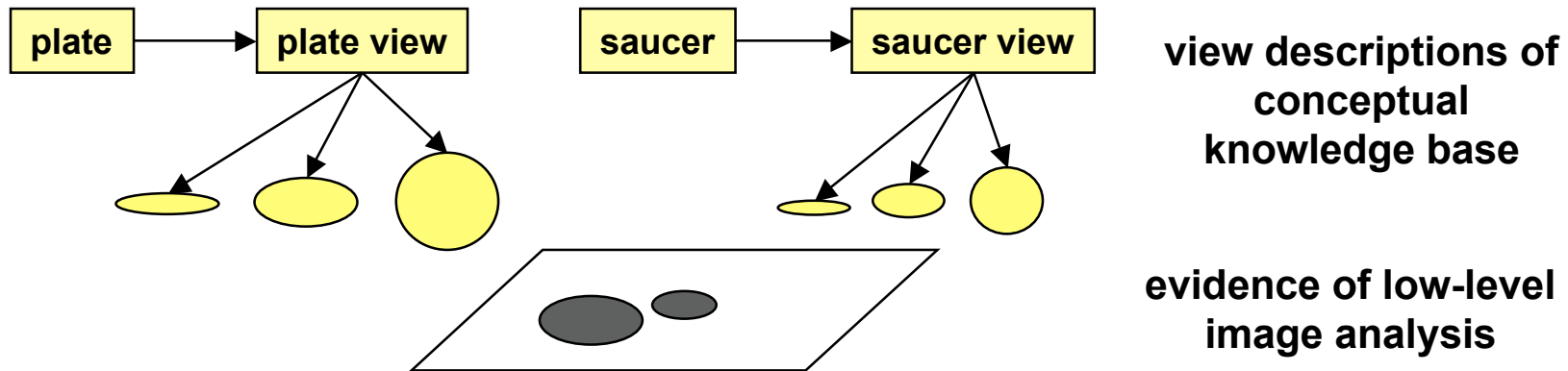
name:	cover	
parents:	:is-a configuration	
parts:	cv-pl :is-a plate	
	cv-sc :is-a saucer	
	cv-cp :is-a cup	
	cv-tt :is-a table-top	
properties:	w, h, tb, te	%width and height of cover
constraints:	cv-sc.pos NE cv-pl.pos	%saucer position northeast of plate position
	cv-sc.rim CLOSE cv-pl.rim	%saucer rim close to plate rim
	cv-cp.pos = cv-sc.pos	
	cv-tt.rim SO cv-pl.rim	%table-top rim south of plate rim

**Spatial relations NO (north), NE (northeast), ... , SO (south), ... ,
CLOSE must be defined and computable based on parts properties.**

Signal-Symbol Interface

Assumptions

- Low-level image analysis provides evidence which can be matched with object views of the conceptual knowledge base.



- Evidence is represented in metric space.
- Evidence may be
 - regions corresponding to objects
 - blobs corresponding to object parts
 - descriptive features around interest points
 - ...

} depending on sophistication of object recognition and categorisation

Matching Evidence with Views

Bottom-up classification

Assign evidence to one of several view classes.

Model-based recognition problem with view classes as models.

In a probabilistic setting same as Bayes classification, except that a priori class probabilities depend on interpretation context.

Top-down hypothesis verification

Check compatibility of top-down view hypothesis with available evidence and other top-down hypotheses.

Checking with evidence is similar to bottom-up classification, except that model is given and evidence is selected.

Checking with other top-down hypotheses is a harder task, as all hypotheses may have uncertainty ranges. How can several hypotheses with uncertain views and locations fit into an image, observing factual evidence and occlusion rules?

Stepwise Scene Interpretation

Given taxonomical and compositional concept hierarchies, there are five kinds of interpretation steps for constructing interpretations consistent with evidence:

Evidence matching

Assign evidence to object view classes or verify view hypotheses.

Aggregate instantiation

Infer an aggregate from (not necessarily all) parts.

Instance specialization

Refine instances along specialisation hierarchy or in terms of aggregate parts.

Instance expansion

Instantiate parts of an instantiated aggregate.

Instance merging

Merge identical instances constructed by different interpretation steps.

Repertoire of interpretation steps allows flexible interpretation strategies
e.g. mixed bottom-up and top-down, context-dependent, task-oriented

Basic Interpretation Algorithm

Enter context information

Repeat

Check for goal completion

Check for new evidence

Determine possible interpretation steps and update agenda

Select from agenda one of

**{ evidence matching,
 aggregate instantiation,
 aggregate expansion,
 instance specialization,
 parameterization,
 constraint propagation }**

Check for conflict

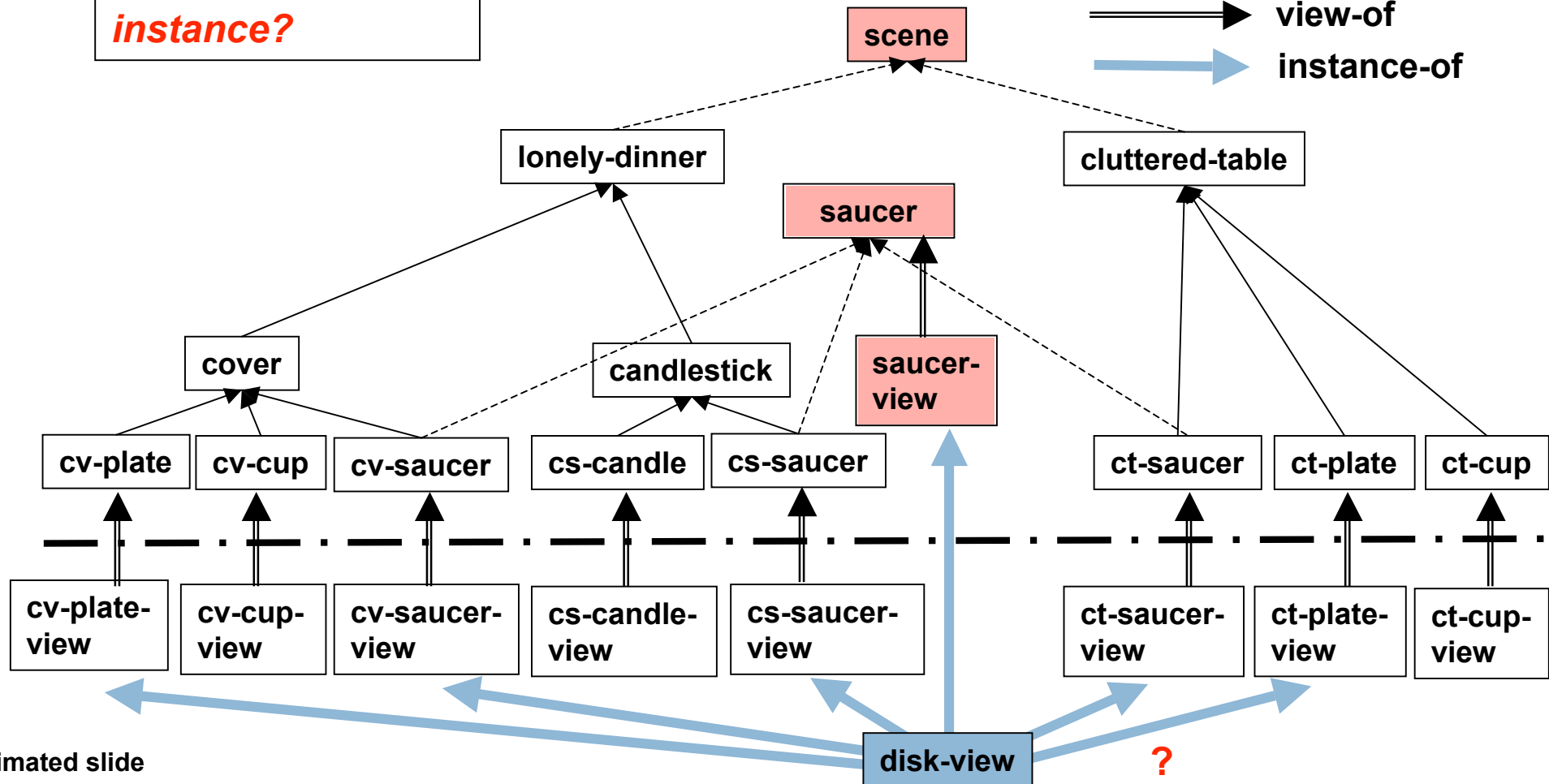
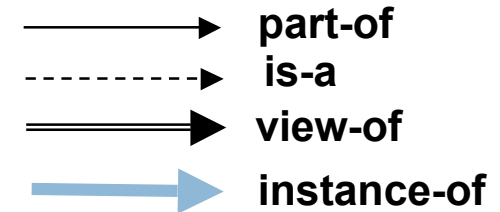
end

Conflict = unsatisfiable constraint net

=> need for backtracking

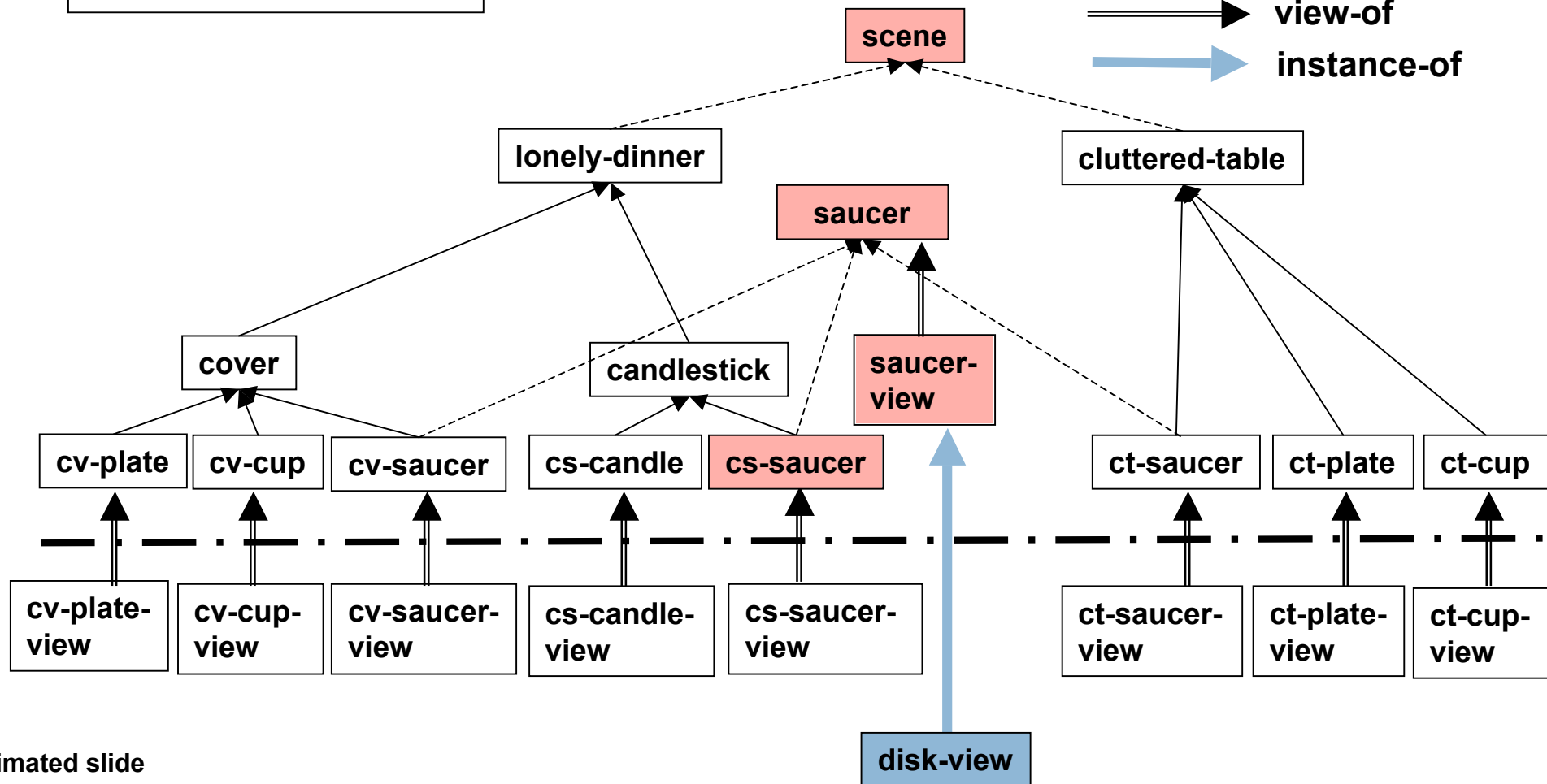
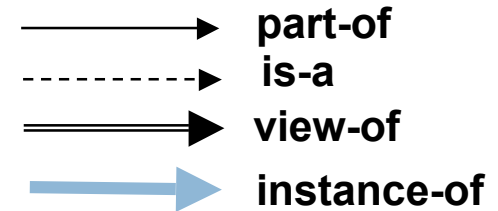
Example for Interpretation Steps (1)

Of what view-class is disk-view an instance?



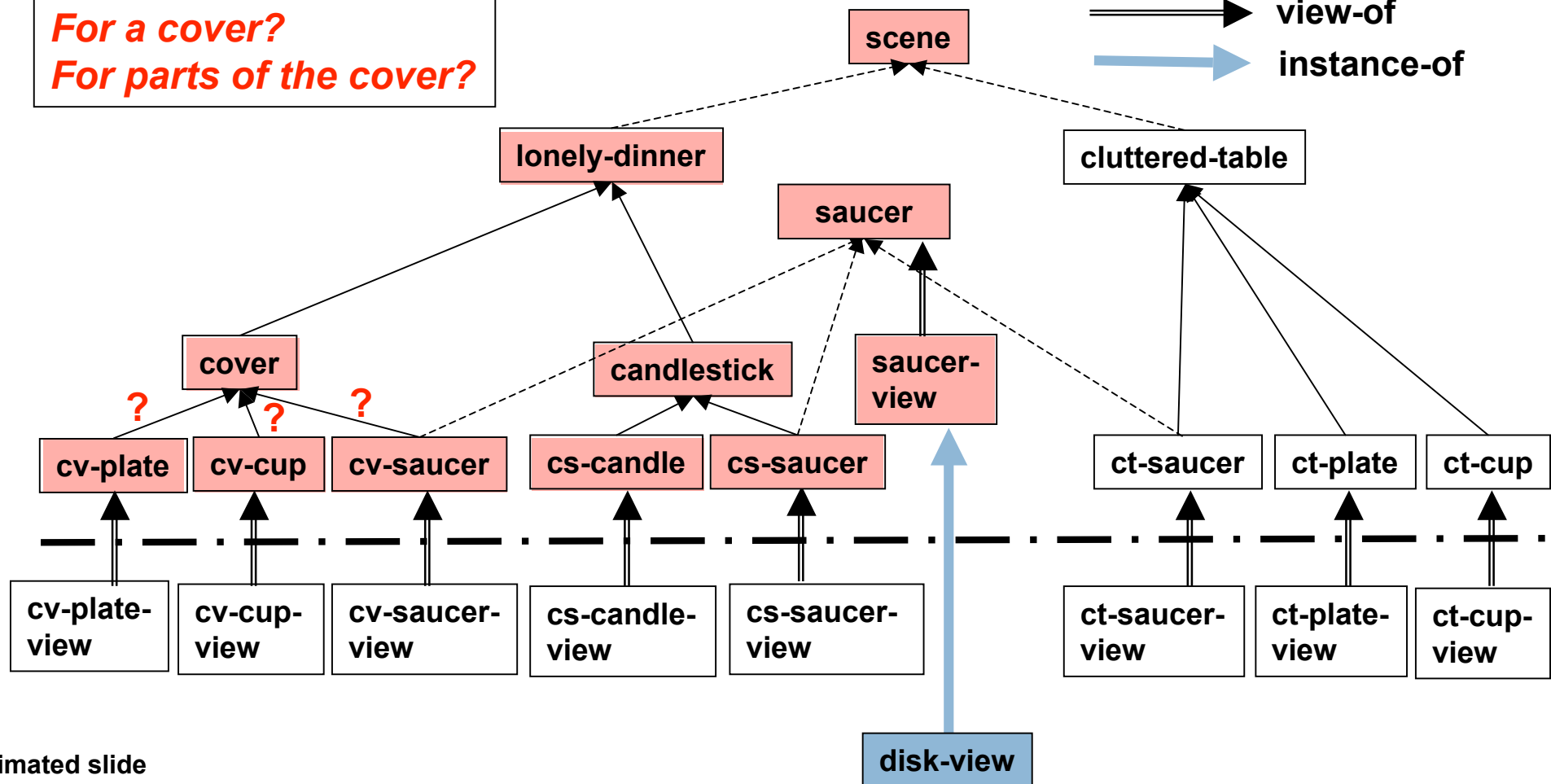
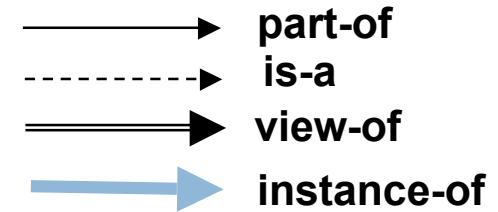
Example for Interpretation Steps (2)

For which role is the saucer a filler?



Example for Interpretation Steps (3)

*Where should one look for a candle?
For a cover?
For parts of the cover?*



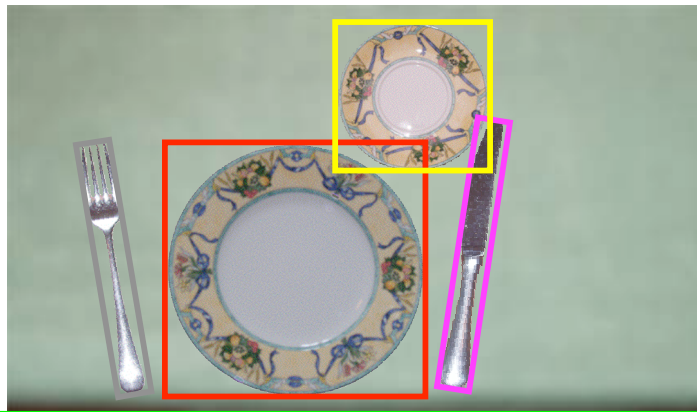
Constraints

Constraints are used in Scene Interpretation for

- conceptual descriptions of aggregates (constraints between parts)
- checking the consistency of parts before aggregate instantiation

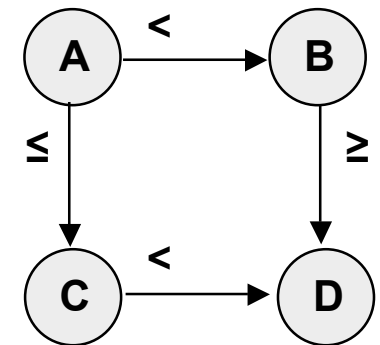
Spatial and temporal constraints are most important for scene interpretation

constraining a spatial configuration



constraining a temporal configuration

A = red_traffic_light.beg
B = red_traffic_light.end
C = pass_traffic_light.beg
D = pass_traffic_light.end



Checking Temporal Constraints (1)

- Variables:** Time variables of an aggregate model
- Domains:** Time points covering the period of interest
- Constraints:**
1. Constraints imposed by aggregate model
 2. Constraints arising from evidence

Example:

Aggregate model:

```
name:      traffic_light_violation
parts:     red_traffic_light
           pass_traffic_light
constraints: pass_traffic_light during red_traffic_light
```

Scene:



Checking Temporal Constraints (2)

Nodes:

A = red_traffic_light.beg

B = red_traffic_light.end

C = pass_traffic_light.beg

D = pass_traffic_light.end

Arcs:

$A \leq C$

$B \geq D$

$A < B$

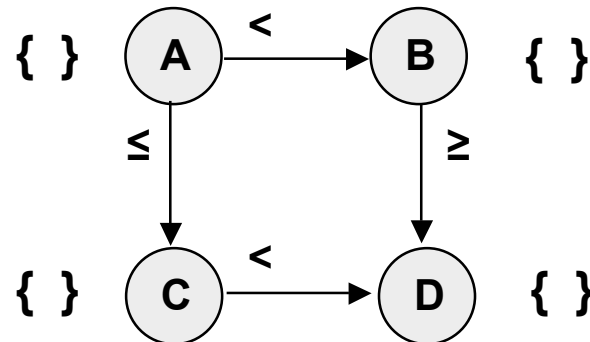
$C < D$



pass_traffic_light during red_traffic_light

begin of occurrence before end

Domains: $\text{dom}(A) = \text{dom}(B) = \text{dom}(C) = \text{dom}(D) = \{ 0:0:0 \dots 23:59:59 \}$



Step 1: Obtain consistency for initial constraint net

Step 2: Observe $A=10:05:08$, prune $\text{dom}(A)$, obtain consistency

Step 3: Observe $C=10:05:30$, prune $\text{dom}(C)$, obtain consistency

Step 4: Observe $B=10:05:33$, prune $\text{dom}(B)$, obtain consistency

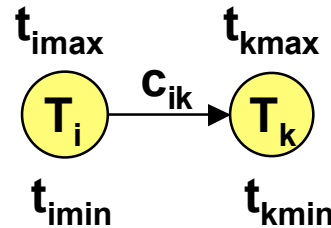
Step 5: Observe $D=10:05:36$, prune $\text{dom}(D)$, obtain consistency, no solution is possible

Animated slide!

Convex Time-Point Algebra

Variables:	time variables	T_i
Domain of a variable:	range of integers	$[t_{imin} .. t_{imax}]$
Constraints:	inequalities with offset	$T_i + c_{ik} \leq T_k$

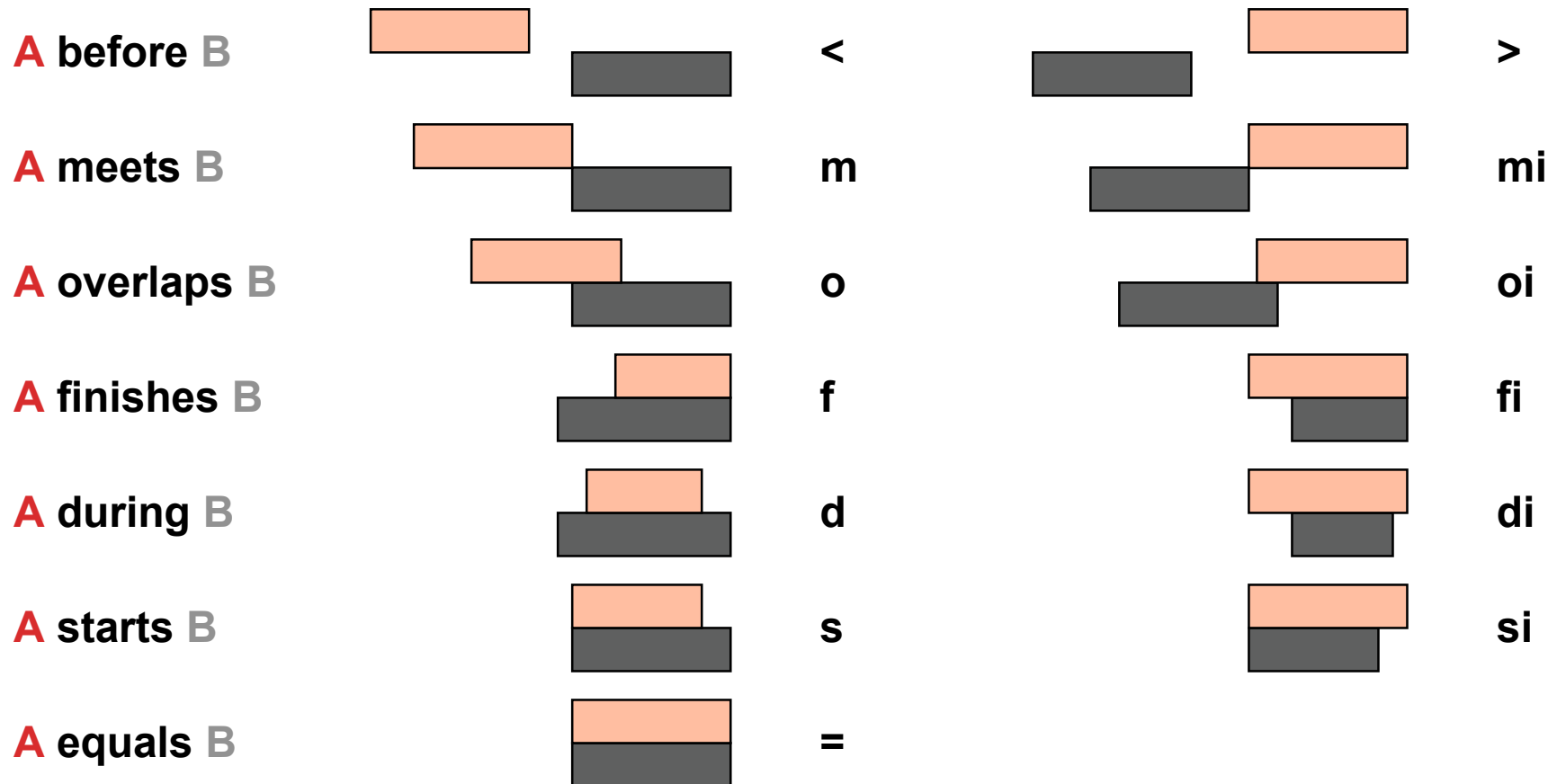
Graphical representation:



- Domains may always be represented by min- and max-values ("convexity property").
- An increase of a min-value affects only time variables connected in edge direction.
- A decrease of a max-value affects only time variables connected against edge direction.
- In a cycle-free constraint net with N variables, any change of a domain can be propagated in at most $N(N-1)$ steps.

Allen's Interval Algebra

Basic relations:



Composition Table for Interval Algebra (1)

For $I_1 R_{12} I_2$ and $I_2 R_{23} I_3$, the table specifies possible relations $I_1 R_{13} I_3$.
 => enables spatial reasoning

	<	m	o	fi	di	si	=
<	<	<	<	<	<	<	<
m	<	<	<	<	<	m	m
o	<	<	< m o	< m o	< m o fi di	o fi di	o
fi	<	m	o	fi	di	oi mi >	fi
di	< m o fi di	o fi di	o fi di	di	di	di	di
si	< m o fi di	o fi di	o fi di	di	di	si	si
=	<	m	o	fi	di	si	=
s	<	<	< m o	< m o	< m o fi di	s = si	s
d	<	<	< m o s d	< m o s d	full	d f oi mi >	d
f	<	m	o s d	f = fi	di si oi mi >	oi mi >	f
oi	< m o fi di	o fi di	o fi di si = s d f oi	di si oi	di si oi mi >	oi mi >	oi
mi	< m o fi di	s = si	d f oi	mi	>	>	mi
>	full	d f oi mi >	d f oi mi >	>	>	>	>

Composition Table for Interval Algebra (2)

	=	s	d	f	oi	mi	>
<	<	<	< m o s d	< m o s d	< m o s d	< m o s d	full
m	m	m	o s d	o s d	o s d	fi = f	di si oi mi >
o	o	o	o s d	o s d	o f d s = si di fi oi	di si oi	di si oi mi >
fi	fi	o	o s d	fi	di si oi	di si oi	di si oi mi pi
di	di	o fi di	o fi di si = s d f oi	di	di si oi	di si oi	di si oi mi pi
si	si	s = si	d f oi	di	oi	mi	>
=	=	s	d	f	oi	mi	>
s	s	s	d	p m o	d f oi	mi	>
d	d	d	d	< m o s d	d f oi mi >	>	>
f	f	d	d	f = fi	oi mi >	>	>
oi	oi	d f oi	d f oi	di si oi	oi mi >	>	>
mi	mi	d f oi	d f oi	mi	>	>	>
>	>	d f oi mi >	d f oi mi >	<	>	>	>

Note that only 27 disjunctive combinations out of 8192 possible combinations occur.

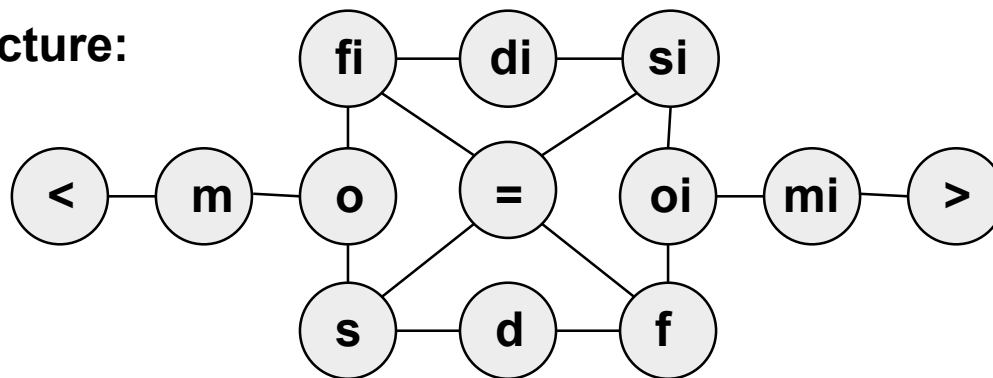
Conceptual Neighborhoods

C. Freksa: Conceptual Neighborhood and its role in temporal and spatial reasoning. In: M. Singh, L. Trave-Massuyes (eds.), Proc. IMACS Workshop on Decision Support Systems and Qualitative Reasoning, North-Holland, 1991, 181-187

In order to permit coarse reasoning, it is useful to identify "neighboring" interval relations.

Two relations between pairs of events are conceptual neighbors if they can be directly transformed into one another by continuous deformation (i.e. shortening or lengthening) of the events.

Conceptual neighborhood structure:



Note that entries of the composition table contain only conceptual neighbors.

Spatial Constraints

In scene interpretation, spatial constraints restrict the relative position and orientation of parts of aggregates.

Example:

Relative positions of plate, saucer and table boundary as parts of a cover

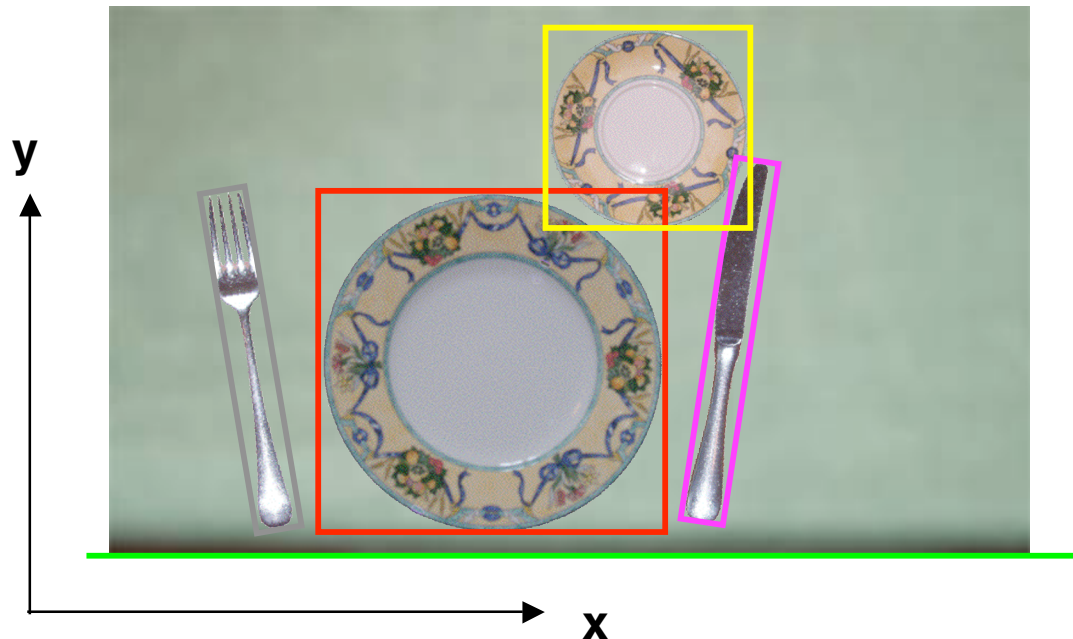


Several ways to represent 2D spatial constraints:

- Bounding box constraints
- Topological relations
- Various other qualitative spatial representations
- Grid region constraints
- Probability distributions

Bounding Boxes

A bounding box is an approximate 2D shape description



A bounding box is specified by

x_{min} , x_{max} , y_{min} , y_{max}

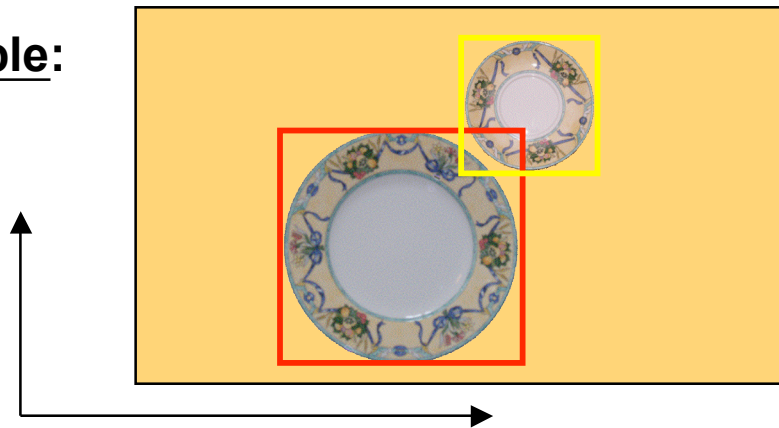
relative to a reference coordinate system

- object-centric vs. global reference coordinate system
- position constraints in terms of relative distances between bounding-box boundaries
- orientation constraints in terms of angles between object axes

Extending Discrete Time-Point Algebra to 2D-Space

Use linear inequalities independently in two spatial dimensions.
(Bounding boxes must be parallel to reference system.)

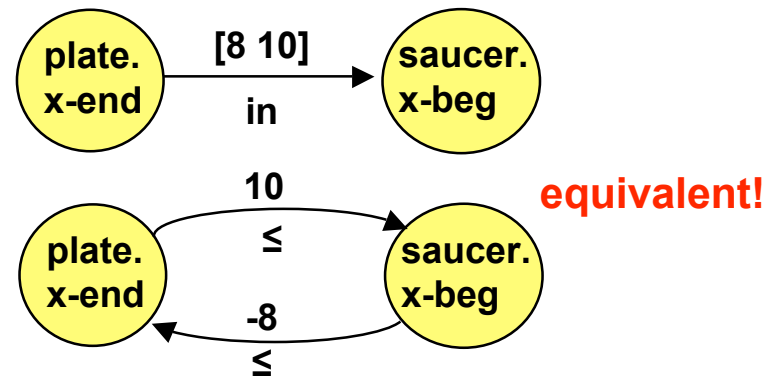
Example:



$\text{plate.x-end} \leq \text{saucer.x-beg} + 10$
 $\text{plate.x-end} \geq \text{saucer.x-beg} + 8$
 $\text{plate.y-end} \leq \text{saucer.y-beg} + 5$
 $\text{plate.y-end} \geq \text{saucer.y-beg} + 3$
 $\text{plate.x-beg} \geq \text{table.x-beg}$
 $\text{plate.x-end} \leq \text{table.x-end}$
 $\text{plate.y-beg} \leq \text{table.y-beg} + 5$
 $\text{plate.y-beg} \geq \text{table.y-beg}$

Pairwise constraints can be combined to (quantitative) interval constraints:

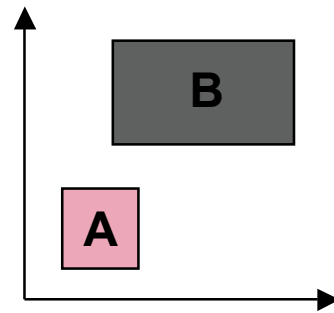
plate.x-end in $\text{saucer.x-beg} + [8 \ 10]$
 plate.y-end in $\text{saucer.y-beg} + [3 \ 5]$
 plate.x-beg in $\text{table.x-beg} + [0 \ \text{inf}]$
 plate.x-end in $\text{table.x-end} + [-\text{inf} \ 0]$
 plate.y-beg in $\text{table.y-beg} + [0 \ 5]$



Extending Allen's Interval Algebra to 2D-Space

Use Allen's interval relations independently for two spatial dimensions.

Example:



horizontal relation: $A \circ B$

vertical relation: $A < B$

combination: $A \circ | < B$

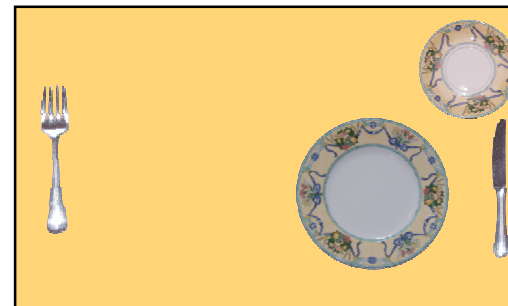
Interval relations are often not restrictive enough to describe the variability of realistic spatial configurations.

Example: Cover configuration

Also covered by this description:

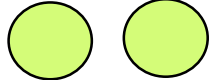
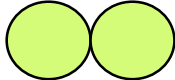
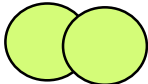





plate $\circ | m$ saucer
 plate $d | d$ table
 plate $> | s$ fork
 plate $< | s$ knife
 saucer $d | d$ table
 fork $d | d$ table
 knife $d | d$ table



Topological Relations in RCC8

Elementary relations (disjunct):

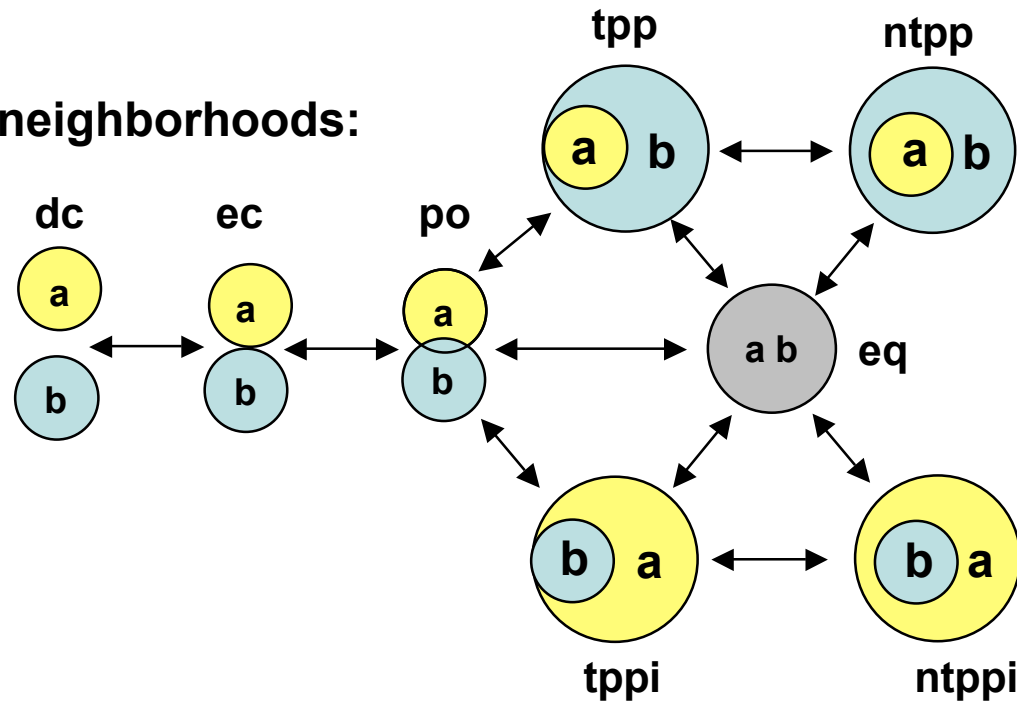
- disconnected  dc
- externally connected  ec
- partial overlap  po
- tangential proper part  tpp tppi
- non-tangential proper part  ntp ntppi
- equal  eq

Composed relations:

- spatially_related
- connected
- overlapping
- inside

RCC8 Conceptual Neighborhoods

Conceptual neighborhoods:



Observations of two regions at two time points must be connected by transitions along a conceptual-neighborhood path.

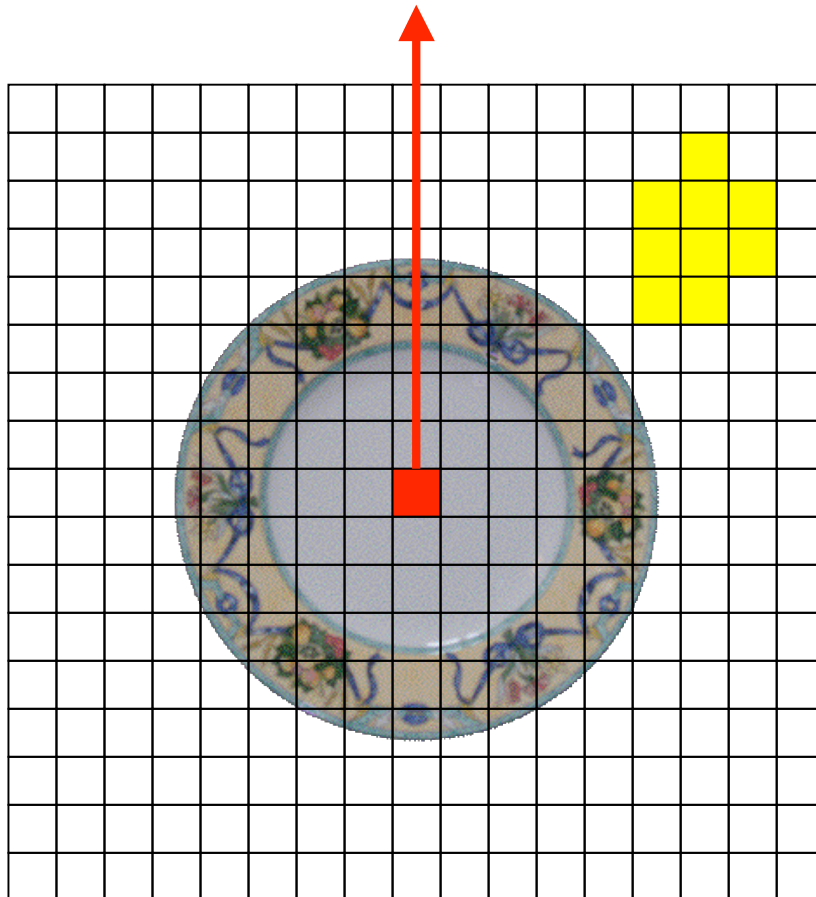
RCC8 Composition Table

Table entries denote possible relations R_{AC} , given R_{AB} and R_{BC}

o	DC	EC	PO	TPP	NTPP	TPPi	NTPPi	EQ
DC	DC,EC,PO TPP,NTPP TPPi,= NTPPi	DC,EC PO TPP NTPP	DC,EC PO TPP NTPP	DC,EC PO TPP NTPP	DC,EC PO TPP NTPP	DC	DC	DC
EC	DC,EC,PO TPPi NTPPi	DC,EC,PO =,TPP TPPi	DC,EC,PO TPP NTPP	EC,PO TPP NTPP	PO TPP NTPP	DC EC	DC	EC
PO	DC,EC,PO TPPi NTPPi	DC,EC,PO TPPi NTPPi	DC,EC,PO TPP,TPPi,= NTPP,NTPPi	PO TPP NTPP	PO TPP NTPP	DC,EC,PO TPPi NTPPi	DC,EC,PO TPPi NTPPi	PO
TPP	DC	DC EC	DC,EC PO,TPP NTPP	TPP NTPP	NTPP	DC,EC,PO =,TPP TPPi	DC,EC,PO TPPi NTPPi	TPP
NTPP	DC	DC	DC,EC PO TPP NTPP	NTPP	NTPP	DC,EC PO TPP NTPP	DC,EC,PO TPP,TPPi NTPP,= NTPPi	NTPP
TPPi	DC,EC,PO TPPi NTPPi	EC,PO TPPi NTPPi	PO TPPi NTPPi	PO,= TPP TPPi	PO TPP NTPP	TPPi NTPPi	NTPPi	TPPi
NTPPi	DC,EC,PO TPPi NTPPi	PO TPPi NTPPi	PO TPPi NTPPi	PO TPPi NTPPi	PO,TPP,= NTPP,TPPi NTPPi	NTPPi	NTPPi	NTPPi
EQ	DC	EC	PO	TPP	NTPP	TPPi	NTPPi	EQ

Spatial Relations as Grid Point Sets

A grid region describes the possible locations (implicit OR) of a point r relative to a reference point and a reference orientation of an object o .



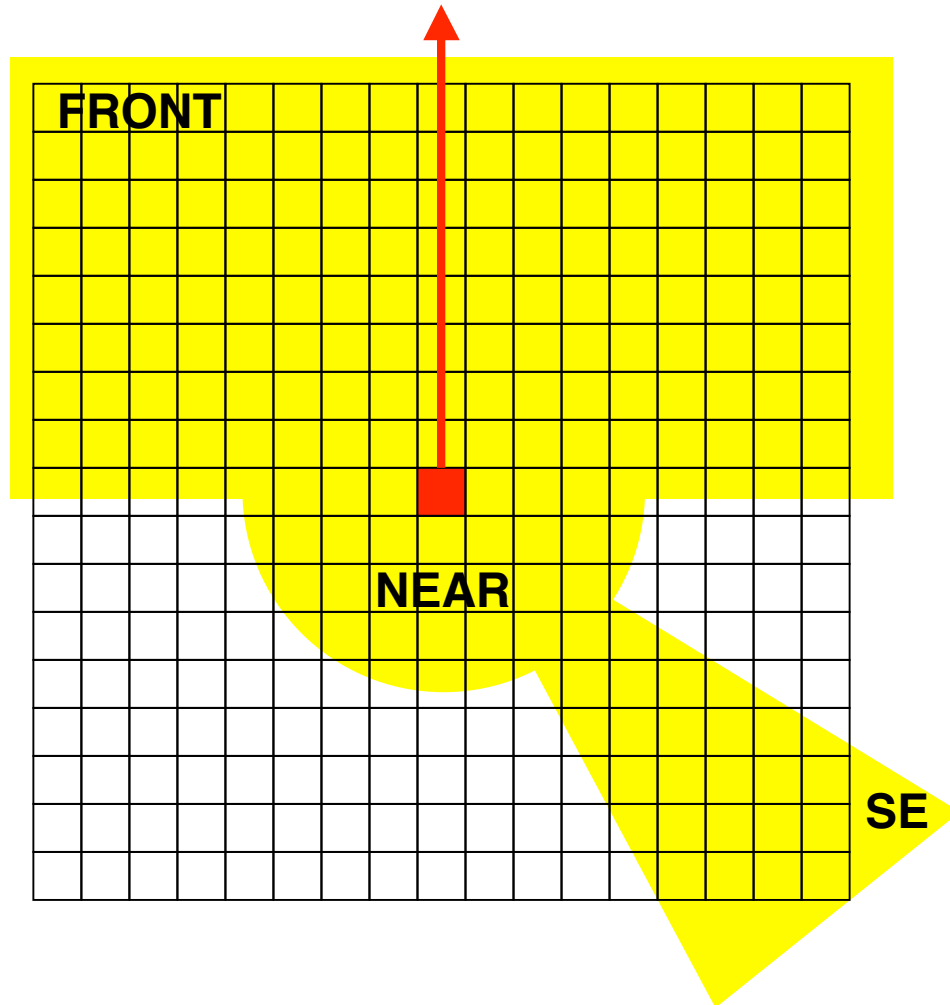
Relative location is a relation
 $O \times R$
between an object o and some
point r .

Example:

O = plate

r = center-of-gravity of saucer

Qualitative Spatial Relations



Grid-point sets constitute qualitative location concepts

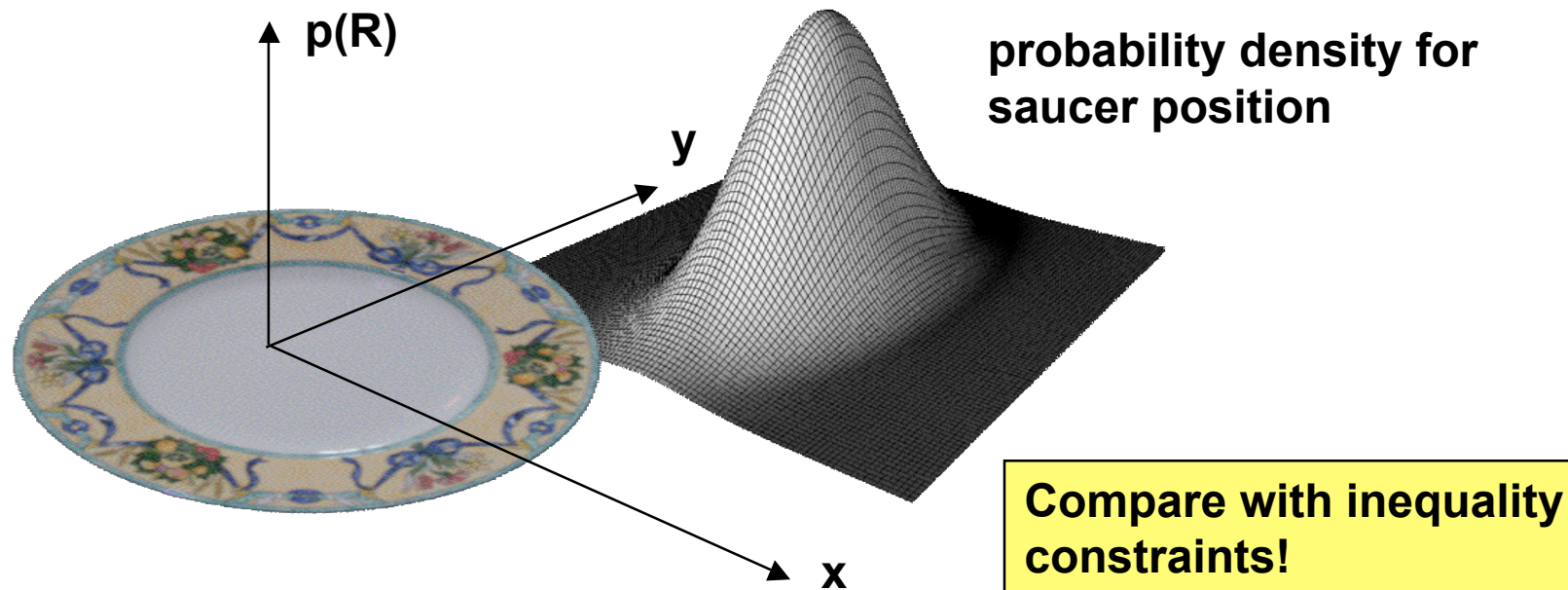
Constraint propagation is possible via set relationships

Example:

(SE plate saucer) ^
(FRONT plate saucer)
=> inconsistent

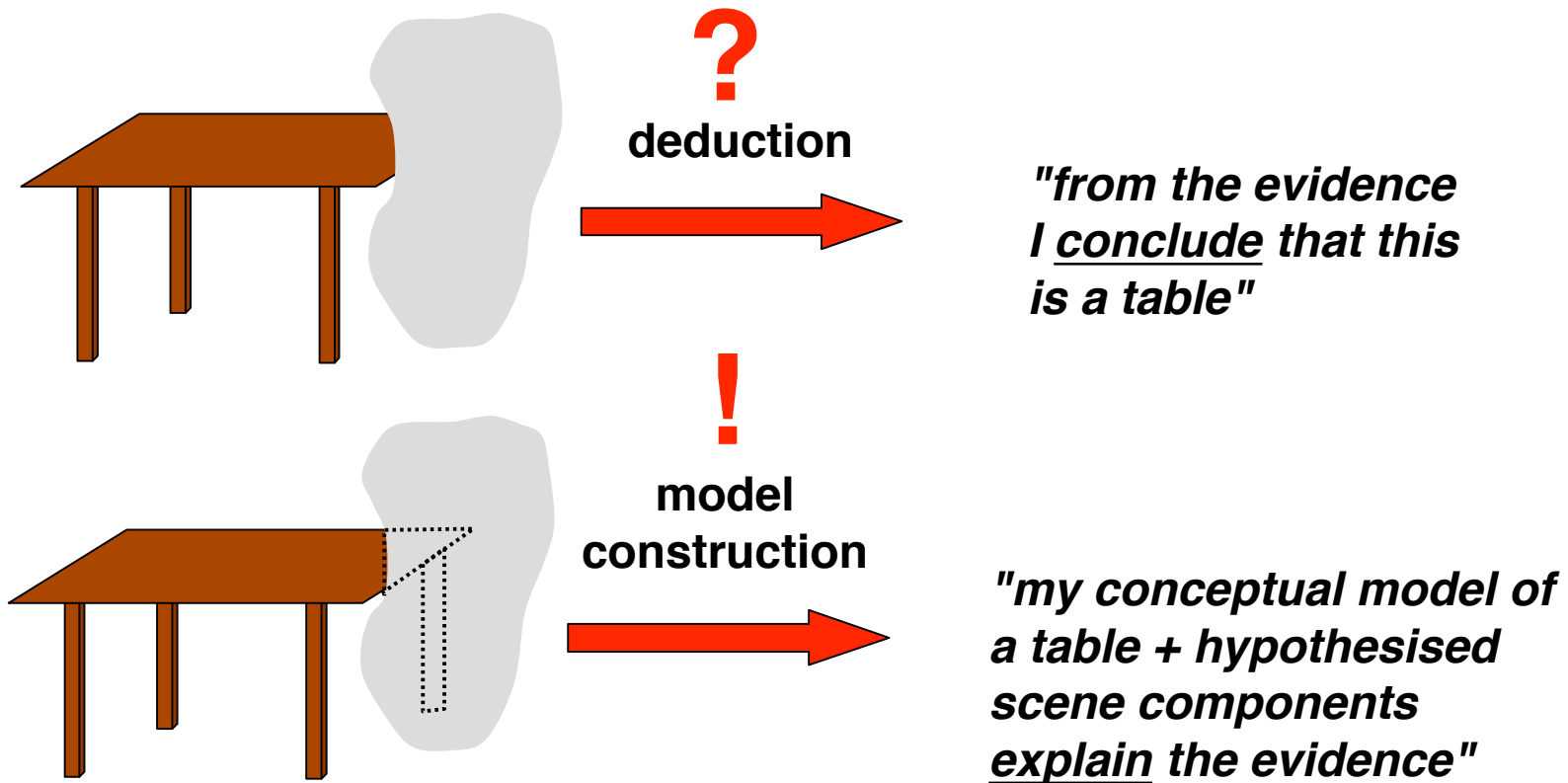
Spatial Relations as Probability Distributions

Constraints on the coordinates (x, y) of a point relative to a reference coordinate system can be expressed in terms of a probability distribution (density).



Logics of Knowledge-based Computer Vision

In 2D images (with possible occlusions) we never see the complete 3D reality.



Reiter & Mackworth 87, Matsuyama 90, Schröder 99

Animated slide!

Definition of Model Construction

An interpretation $I = [D, \varphi, \pi]$ of a logical language maps

- constant symbols of the language into individuals of a real-world domain D
- N-ary predicate symbols of the language into predicate functions over D^N

A model of some clauses is an interpretation for which all clauses are true.

How to do model construction:

- Establish mapping φ by assigning segmentation results to constant symbols
- Establish mapping π by assigning computational procedures to predicate symbols
- Construct model by finding clauses which are true

Deciding whether a model exists is undecidable in FOPC!
There may be infinitely many models!

Problems with Model Construction

Mapping φ

Establish mapping between real-world objects (as delivered by image analysis procedures) and constant symbols (as used in symbolic knowledge representation)

Problems: Segmentation performance, real-world objects not visible in a scene

Mapping π

Establish mapping between procedures which compute real-world relations (e.g. "touch") and predicate symbols of symbolic knowledge representation.

Problems: View-based procedures vs. 3D real-world relations, classification uncertainty

True clauses

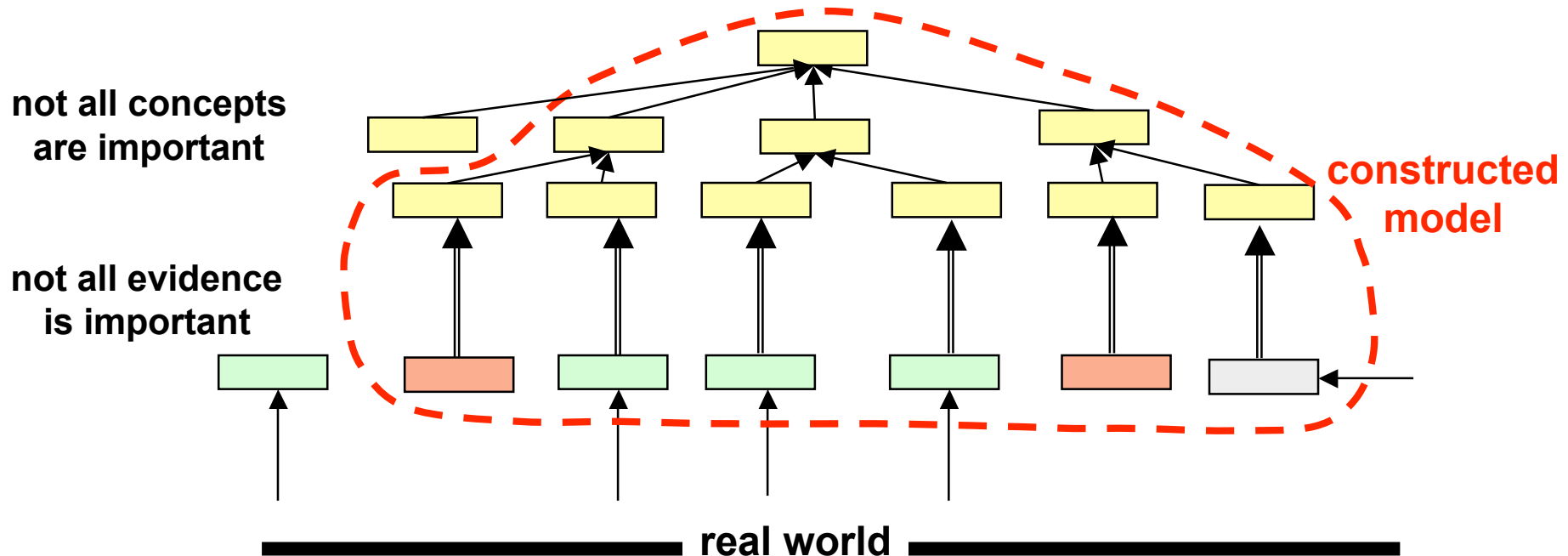
Establish that all clauses of the symbolic knowledge base are true for the mappings φ and π .

Problems: Many clauses of the knowledge base may be irrelevant for a concrete scene. A partial model may suffice for the vision task on hand.

So what is Knowledge-based Computer Vision?

Intuitively:

A scene interpretation is a scene description in terms of instantiated concepts consistent with evidence and context information.



concepts context hypotheses evidence

■ ■ ■ ■

Animated slide!

Partial Models

It seems plausible that a scene interpretation must not be proved to be consistent with all clauses of the conceptual knowledge base.

Example: Outdoor knowledge (e.g. about street traffic behavior) may not be relevant for indoor scenes (e.g. setting a table).

But there may be scenes where knowledge beyond the concrete scene may influence the interpretation.

Example: Knowing that a person has arrived outside of a house may affect the expected behavior of persons inside.

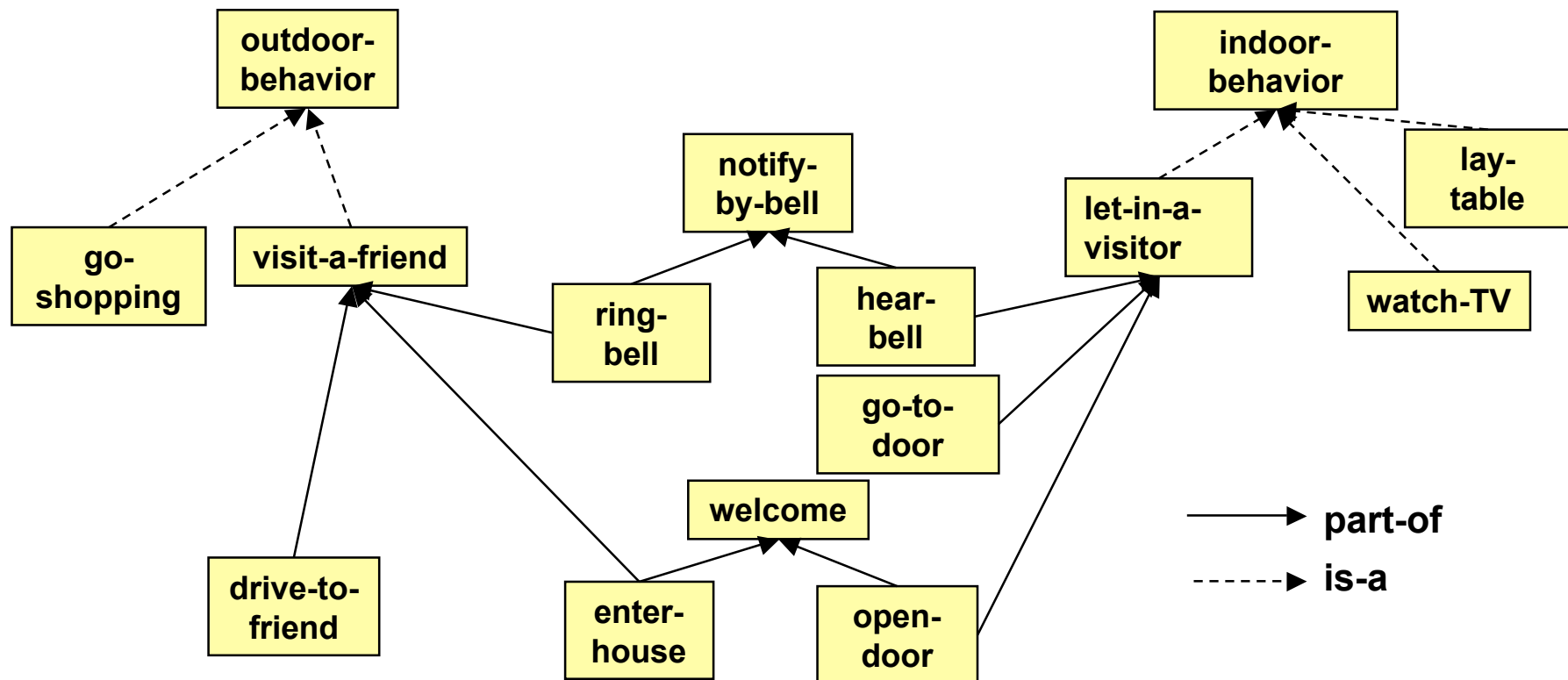
A good knowledge base provides aggregate concepts for all interrelated entities, often overlapping specific domains. In general, any two conceptual entities may be (indirectly) structurally connected (s. example next slide).

Partiality of scene interpretations depends on vision goals and context, not on structural boundaries of the conceptual knowledge base.

Interrelated Domains

Conceptual entities in seemingly disjoint domains may be interrelated, hence model construction for scene interpretations cannot be restricted by obvious boundaries.

Example: Outdoor behavior connected to indoor behavior



Finite Model Construction

(Reiter & Mackworth 87, Poole)

- An image consists of regions and chains (edges)
- The image elements constitute all constant symbols of an interpretation (domain closure assumption)
- Different constant symbols denote different image elements and vice versa (unique name assumption)

 Problem can be expressed in Propositional Calculus and solved as a constraint satisfaction problem.

For MAPSEE, scene interpretation amounts to finding a mapping p for predicates *road*, *river*, *shore*, *land*, *water*.

Constructing Partial Models

If image analysis provides the intended mappings φ and π from symbols into a real-world domain, model construction amounts to instantiating clauses of the conceptual knowledge base such that all clauses are true.

The interpretation steps introduced earlier allow to instantiate all concepts of the knowledge base.

Evidence matching

Assign evidence to object view classes or verify view hypotheses.

Aggregate instantiation

Infer an aggregate from (not necessarily all) parts.

Instance specialization

Refine instances along specialisation hierarchy or in terms of aggregate parts.

Instance expansion

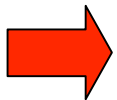
Instantiate parts of an instantiated aggregate.

Instance merging

Merge identical instances constructed by different interpretation steps.

Practical Requirements for Partial Models

- **Task-dependent scope and abstraction level**
 - **no need for checking all predicates**
e.g. propositions outside a space and time frame may be uninteresting
 - **no need for maximal specialization**
e.g. geometrical shape of "thing" suffices for obstacle avoidance
- **Partial model may not have consistent completion**
 - **uncertain propositions due to inherent ambiguity**
 - **predictions may be falsified**
- **Real-world agents need single "best" scene interpretation**
 - **requires uncertainty rating for evidence and context (propositions)**
 - **requires preference measure for scene interpretations**



Logical model property provides only loose frame for possible scene interpretations.

Stepwise Model Construction

