# Motion Understanding

- **The Problem:** “What happened in this movie?”

- **The Approach:** Physics-based analysis of simplified scene model.
- **Example:**

  - Can is a ‘passive object’ (resting on table).
  - Hand is an ‘active object’ (moving above table).
  - Hand lifts can off table (by attaching to it).

# Framework

**Computational Perceiver** (Jepson & Richards, 1991, 1993)

- **Ontology:** representation of scene properties.
  - Force-dynamic properties of simplified scene model (eg., ‘active’ vs. ‘passive’ objects, attached vs. contacting objects, etc.).
- **Domain theory:** constraints used to determine feasible interpretations.
  - Consistency with Newtonian mechanics.
- **Preferences:** to choose most plausible interpretation(s).

# Simplifications (Base system)

- Analysis of motion at a single point in time (relaxed later).
- Continuous motion
  (ie., no changing contact relations or collisions).
- 2D polygonal scene model.

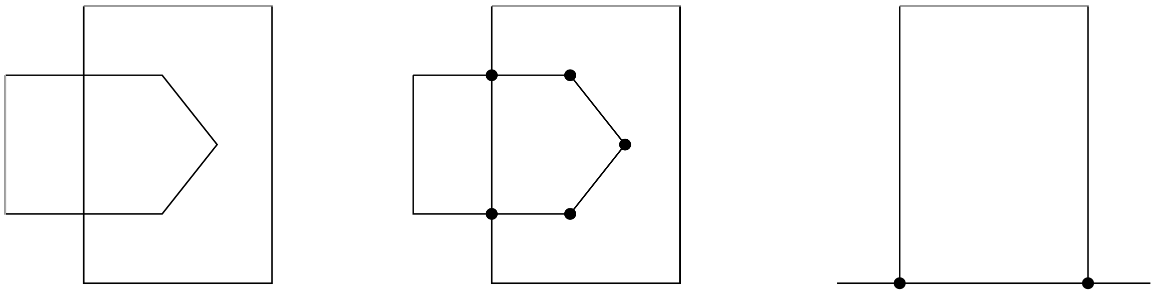
(This base system provides a framework for an enhanced system that integrates information over time.)

**(Overview)**

# Representation: Kinematic Model

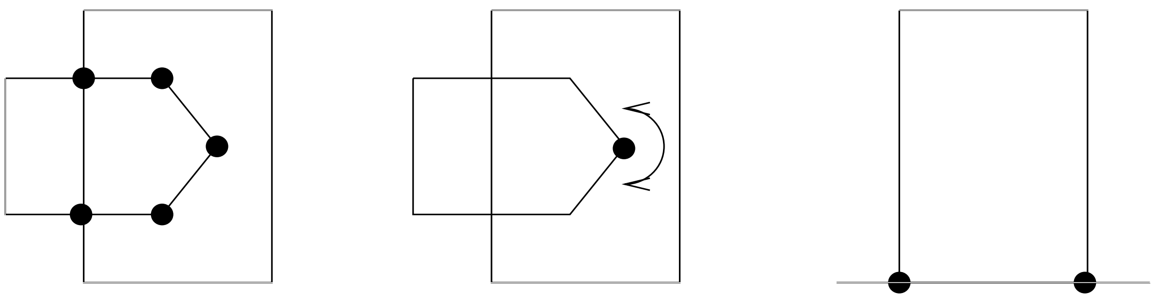
(object shapes, velocities, contact geometry)

- **Objects.**
  2D rigid polygons (made up of convex parts).
- **Contact.**
  Layered scene model with the following contact relations between objects:



Disjoint Overlap Abut

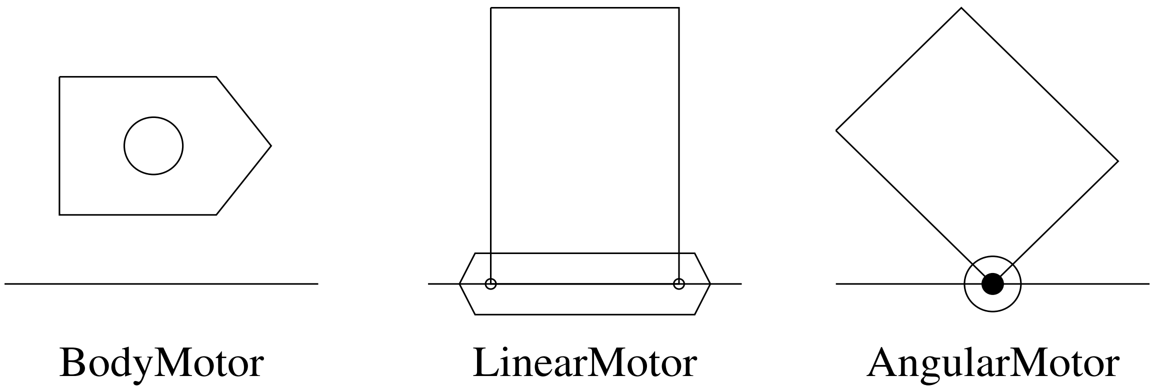
- **Attachment.**
  Model attachment by fixing contact points (analogy to *rivets*).



# Representation: Dynamic Model

(object masses, accelerations, forces)

- **Masses**
  - **Mass.** Variable mass; assumed to be positive (and bounded above).
  - **Center of gravity.** Taken to be center of object.
  - **Inertia.** Constrained by mass and bounds on mass distribution.
- **Forces**
  - **Gravity** acts on all objects.
  - **Contact forces.** normal forces, tangential forces (friction).
- **Ground plane:** used to represent table.
- **Motors** (Our assertions about 'active' objects.)



# Interpretation

**Definition:** An *interpretation* is an unambiguous assignment of the terms in our representation:

- contact and attachment relations between objects,
- all force-generators (*motors*) in the scene.

# Feasible Interpretations

**Definition:** An interpretation is *feasible* if the motion of the objects is consistent with the physics-based model.

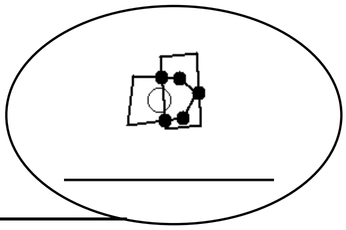
**Testing Feasibility by 'Force-balancing'**

- For each object, sum forces due to gravity, contact with other objects, and forces due to motors.
- masses are variable (constrained to be positive).
- solution must explain observed accelerations according to Newtonian mechanics.
- formulate as *constraint satisfaction problem*.
  (Reduces to *linear programming* in our case)

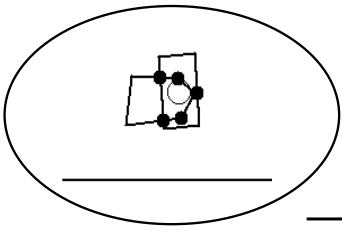
**Sample inferences:**

- 'passive' objects: must be *supported* (otherwise *falling*).
- non-attached objects: can 'push' but not 'pull'.

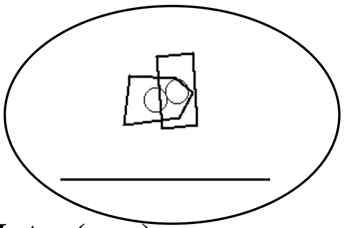
# Example: Feasible Interpretations (Overview)



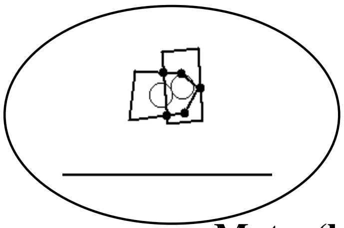
Motor(hand),________
Attach(hand,can)
Contact(hand,can)



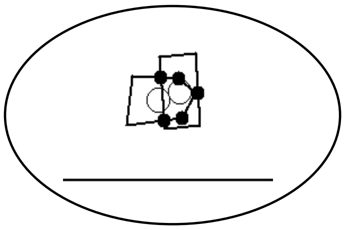
________,Motor(can)
Attach(hand, can)
Contact(hand, can)



Motor(hand), Motor(can)
________
________



Motor(hand), Motor(can)
________
Contact(hand,can)



Motor(hand), Motor(can)
Attach(hand,can)
Contact(hand,can)

**Observation:** trivial interpretations can always be found by adding extra motors!

# Preferred Interpretations

- **Preferences:**

  *Minimize* set of motors:

  - $P_{bm}$ : $\neg\textsc{BodyMotor}(o) \succ \textsc{BodyMotor}(o)$;
  - $P_{lm}$ : $\neg\textsc{LinearMotor}(o_1, o_2) \succ \textsc{LinearMotor}(o_1, o_2)$;
  - $P_{am}$ : $\neg\textsc{AngularMotor}(o_1, o_2) \succ \textsc{AngularMotor}(o_1, o_2)$.

  *Minimize* set of attachments:

  - $P_a(o_1, o_2)$ : $\neg\textsc{Attach}(o_1, o_2) \succ \textsc{Attach}(o_1, o_2)$,

  *Indifferent* to contacts:

  - $P_c(o_1, o_2)$ : $\neg\textsc{Contact}(o_1, o_2) \sim \textsc{Contact}(o_1, o_2)$,

- **Priorities on preferences:**

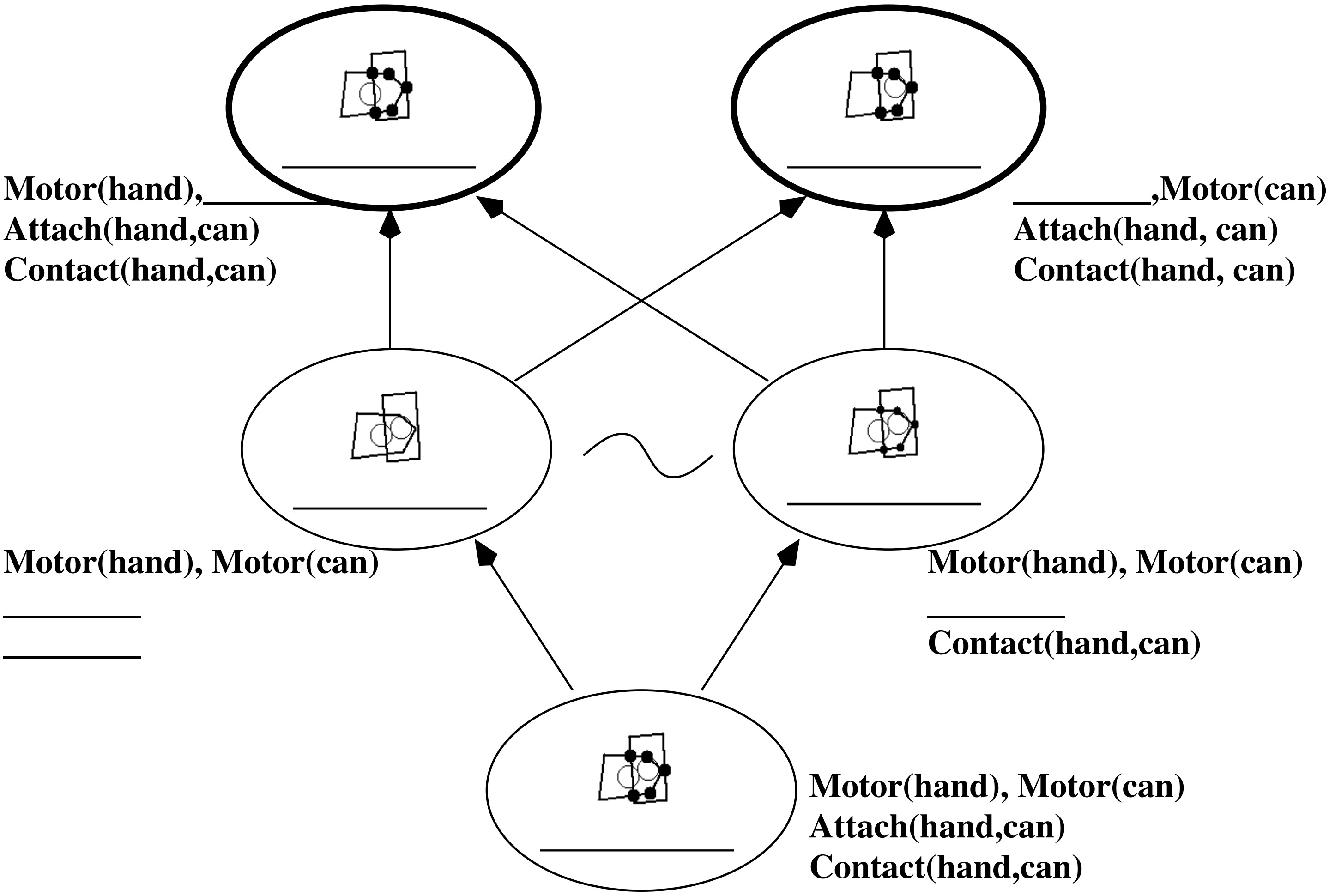
  - $\{P_{bm}, P_{lm}, P_{am}\} > P_a > P_c$

**Key observation:** Preferred models defined by *prioritized subset ordering*. (This is a special case of *circumscription* (McCarthy, 1980).)

(Overview)

# Preference Ordering

- hand lifts can.
- can lifts hand.



# Summary so Far

**Computational Perceiver:**

- **Ontology:** represention of scene properties.
  - *Kinematic* and *dynamic* models of the scene.
- **Domain theory:** constraints used to determine feasible interpretations.
  - Consistency with Newtonian mechanics of simplified scene model.
- **Preferences:** to choose most plausible interpretation(s).
  - Seek 'minimal' or 'simplest' interpretations that explain each frame (ie., minimal sets of motors and attachments).

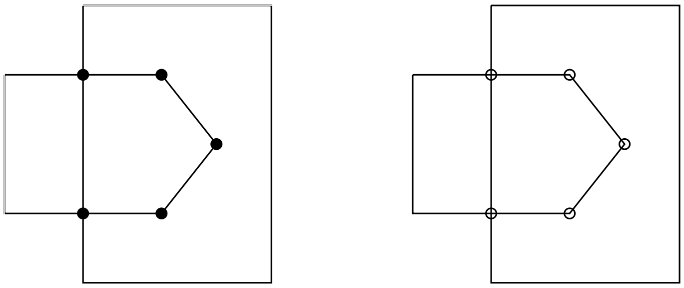
# Implementation (Base System)

- Input object shapes along with estimates of their motions.
- Compute possible contact relations between objects.
- Test feasibility of interpretations.
- Search for preferred interpretations.

(Implementation)

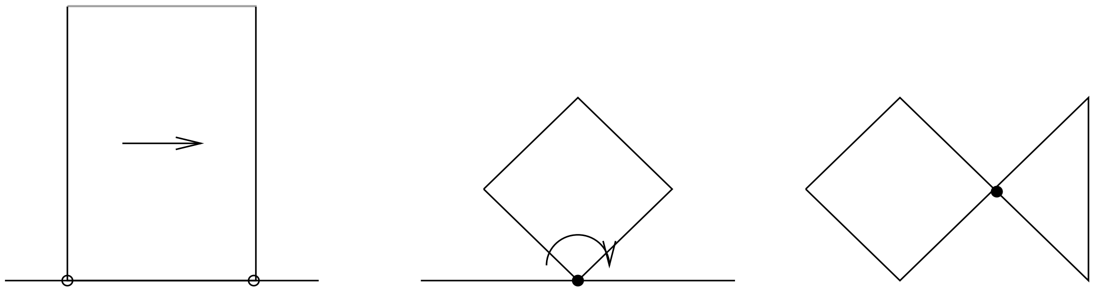
# Determine Contact Relations

- **Overlapping contacts**



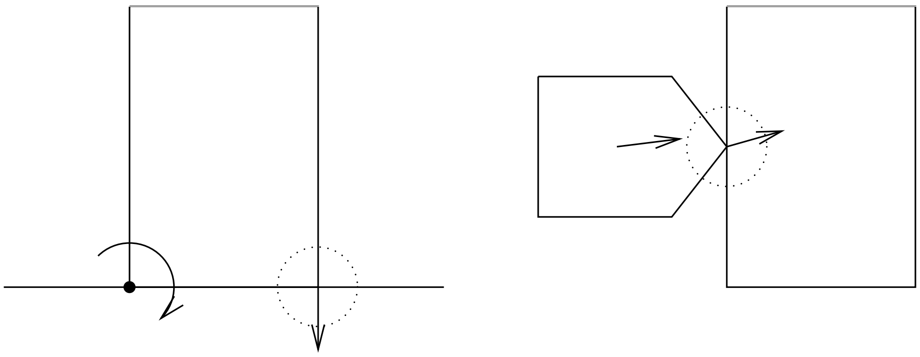
Stationary    Moving

- **Abutting contacts**



edge-edge    point-edge    point-point

- **Inadmissible Contacts**



- **Note:** Consider only *pairwise* contact between objects.

# Feasible Interpretations

- **Definition:** An *interpretation* is a complete description of the scene (wrt the granularity of our representation).

  Specifically, we denote an interpretation, $I = (C, A)$:

  - *Configuration*, $C$, specifies components of the representation which are uniquely specified by the image data and the ontology.
    (e.g., centers of mass, velocities, accelerations, ...).
  - *Assertions*, $A$, specify our hypotheses about the scene
    (e.g., contact and attachment between objects, motors, ...)

- **Definition:** An interpretation is *feasible* if there exists a some assignment of the unknowns (i.e., the contact geometry in the scene, the masses of the objects and forces between the objects) which is consistent with our domain theory.

# Feasibility Test

**Newton-Euler equations** (rigid-body motion)

$$\begin{aligned} \mathbf{F} &= \dot{\mathbf{p}} = M\dot{\mathbf{v}} \\ \mathbf{N} &= \dot{\mathbf{L}} = \mathbf{I}\dot{\omega} + \omega \times \mathbf{I}\omega \end{aligned}$$

- Place force vector at each contact point. (Represent contact regions by convex hull of contact set (Featherstone, 1987; Baraff, 1995).)
- Sum forces and torques for each object.
  (Include forces and torques MOTOR assertions.)

**Constraints:**

- Normal contact forces (non attached objects), $\mathbf{F} \cdot \mathbf{n} \geq 0$
- Static friction (linear model), $||\mathbf{F_t}|| \leq \mu \mathbf{F_n}$
- Sliding friction (additional constraint), $\mathbf{F} \cdot \mathbf{v} \leq 0$

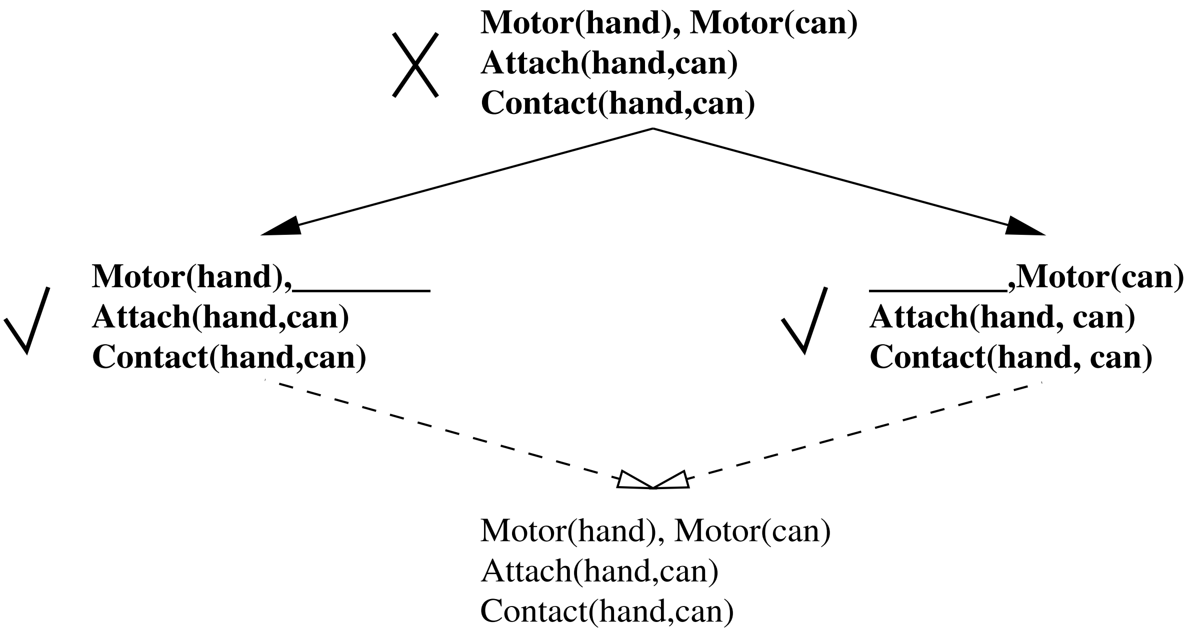
**Testing Feasibility:** Approximate by a linear constraints. (Computed by *simplex algorithm*.)

**Note:** feasibility test applies in both 2D and 3D scenes.

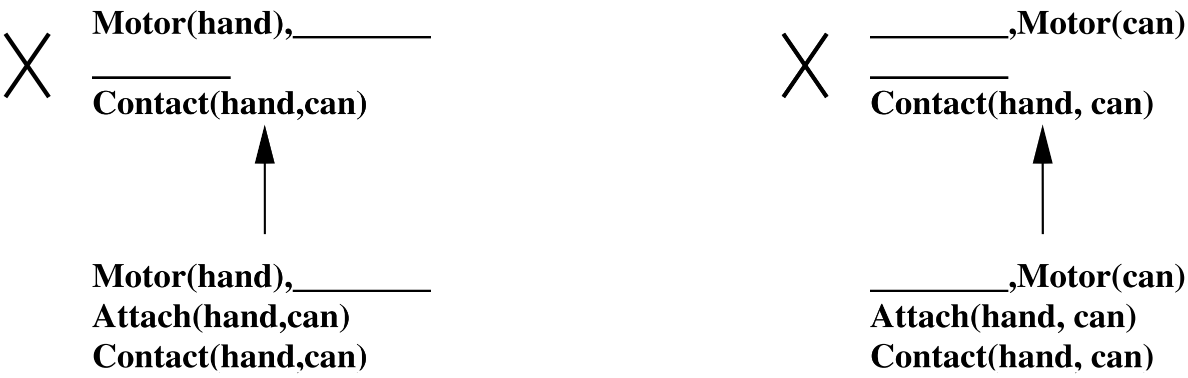
## (Implementation)

# Search Example

- **First pass:** Minimize motors with all possible attachments.




- **Second pass:** Minimize attachment for each setting of motors.



# Search Algorithm

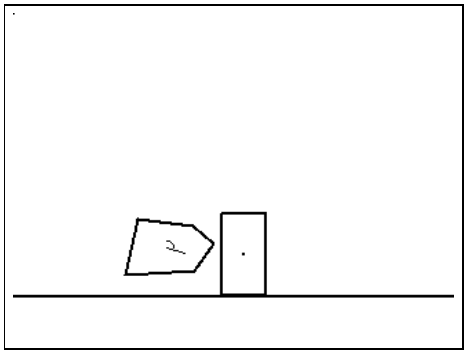
- **Efficient algorithm**
  - Set assertions at lower priorities to ‘most permissive’ settings.
    (e.g., assert all possible attachments when searching for motors)
  - Breadth-first search at highest priority.
  - For each minimal model, perform breadth first search at lower priority, and so on.
- **Issues**
  - Algorithm relies on the structure of the domain. (i.e., adding assertions never makes an interpretation infeasible)
  - Refer to this property as *monotonicity*.

# Examples (Base System)

- **Image Processing:**
  - Outline objects (including table) in first frame.
  - *View-based* tracking of object templates (Jepson & Black, 1993).
  - Robust estimatation of velocities and accelerations.
- **Search for preferred interpretations.**
- **Output interpretations for each frame.**

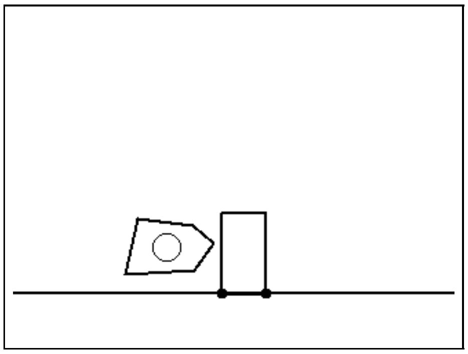
# Coke (frame 32)

**Input**



(objects and accelerations)

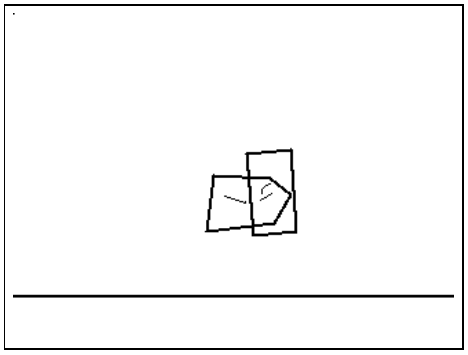
**Interpretation**



$\text{B}\textsc{ody}\text{M}\textsc{otor}(hand).$

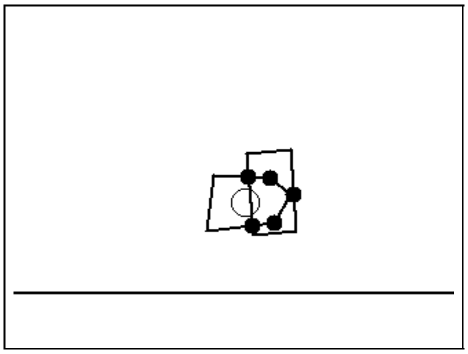
# Coke (frame 63)

**Input**



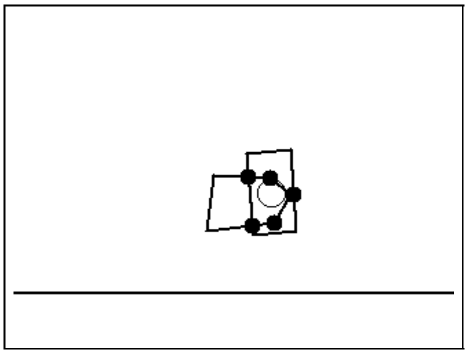
(objects and accelerations)

**Interpretation #1**



BODYMOTOR$(hand)$,
ATTACH$(hand, can)$.

**Interpretation #2**

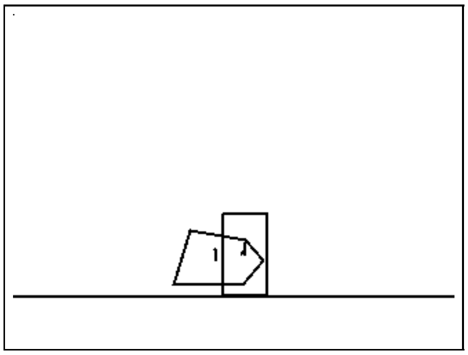


BODYMOTOR$(can)$,
ATTACH$(hand, can)$.

(Examples)

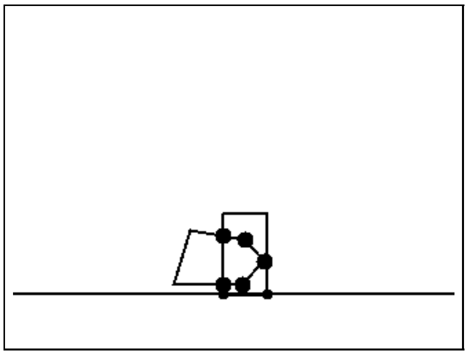
# Coke (frame 50)

**Input**



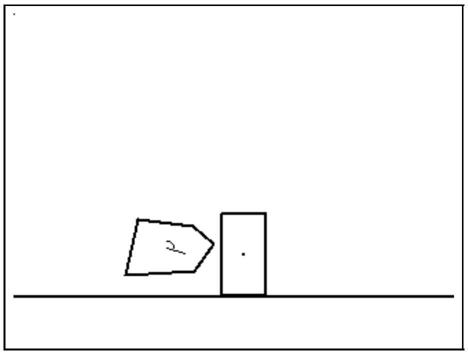
(objects and accelerations)

**Interpretation**

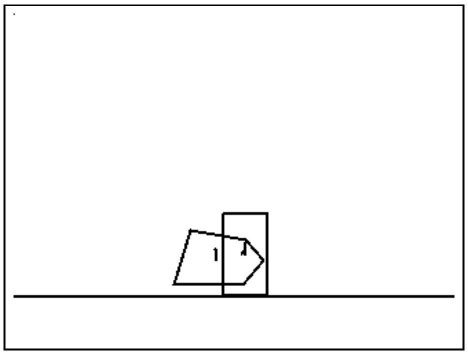


ATTACH$(hand, can)$.

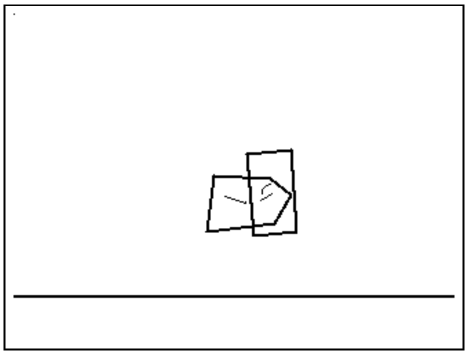
# Coke



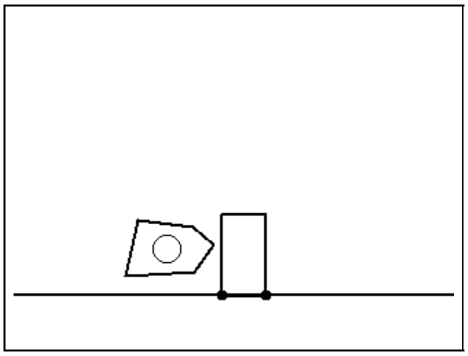
*32*



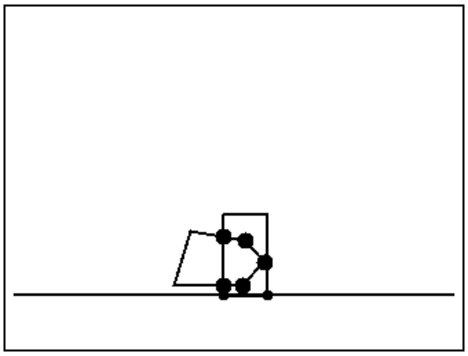
*45*



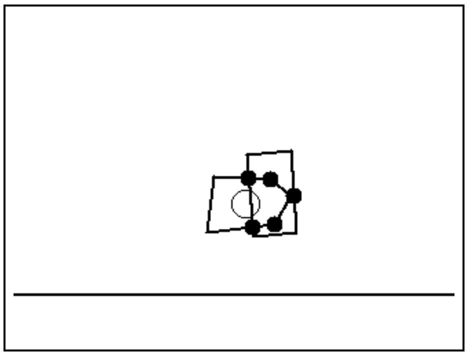
*63*



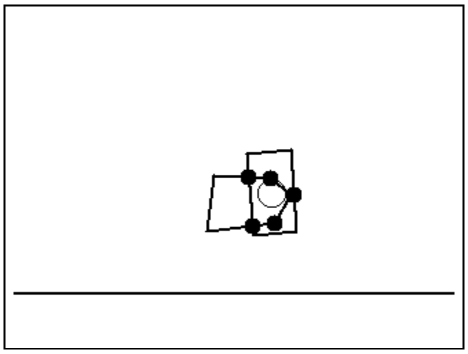
BODYMOTOR$(hand)$.



BODYMOTOR$(hand)$,
ATTACH$(hand, can)$.



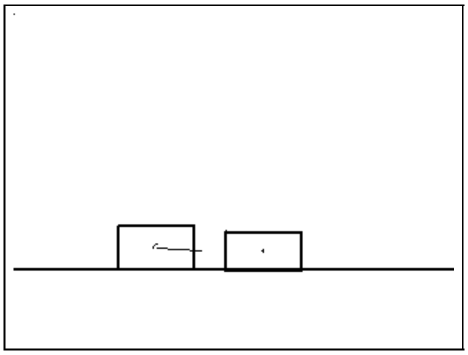
BODYMOTOR$(can)$,
ATTACH$(hand, can)$.



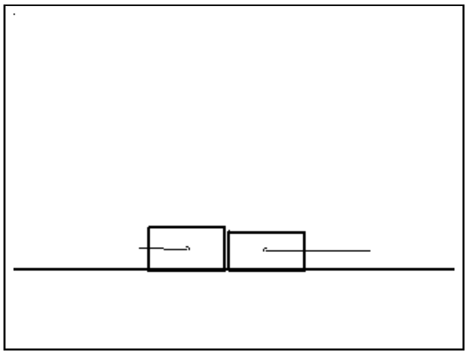
BODYMOTOR$(can)$,
ATTACH$(hand, can)$.

**(Examples)**

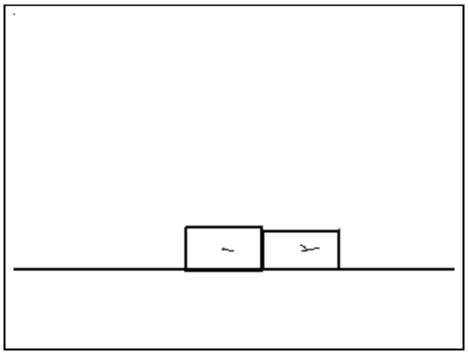
# Cars



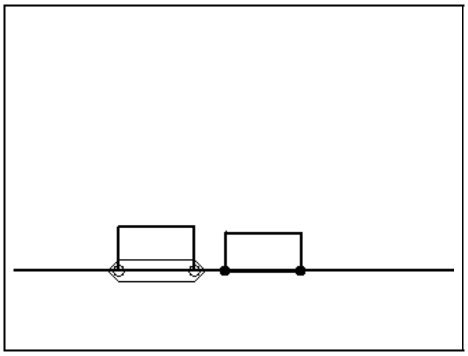
*23*



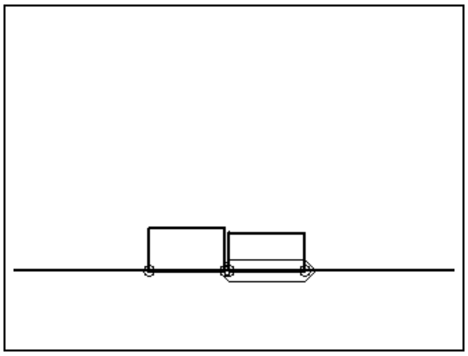
*28*



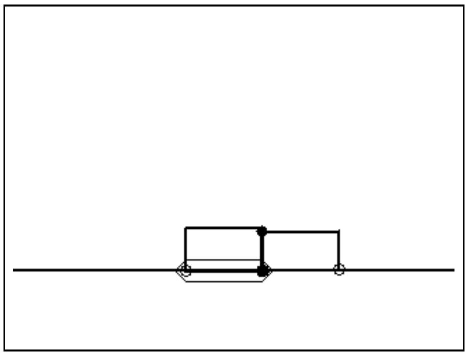
*34*



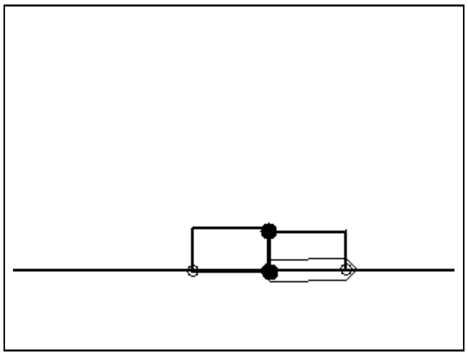
LINEARMOTOR$(left, table)$.



LINEARMOTOR$(right, table)$.



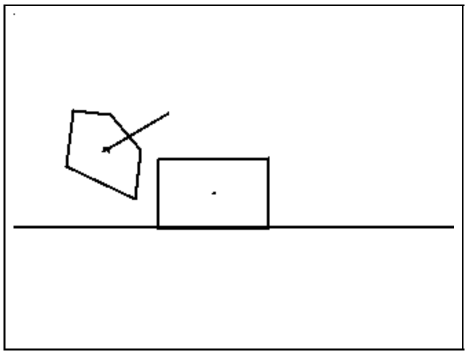
LINEARMOTOR$(left, table)$.



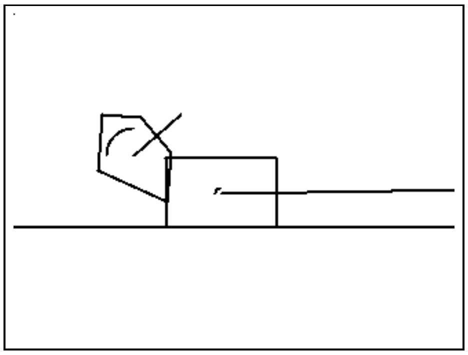
LINEARMOTOR$(right, table)$,
ATTACH$(left, right)$.

**(Examples)**

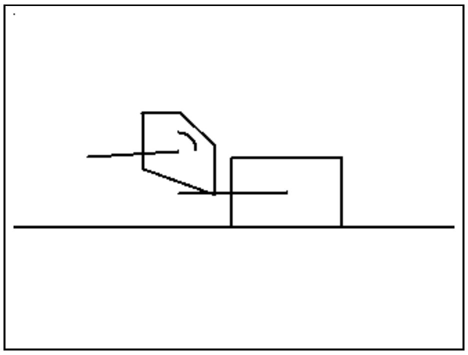
# Hit



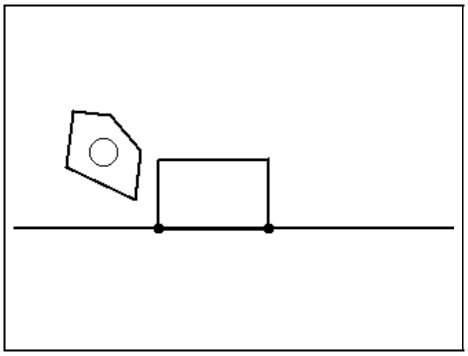
*28*



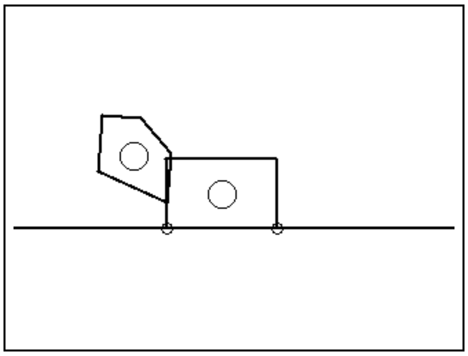
*31*



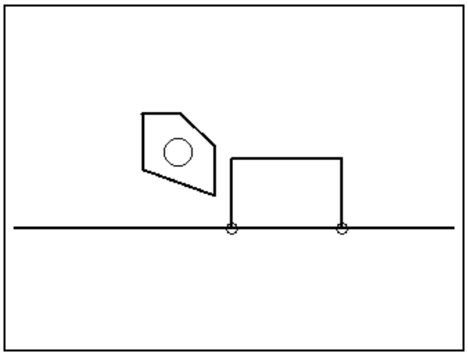
*36*



BODYMOTOR$(hand)$.



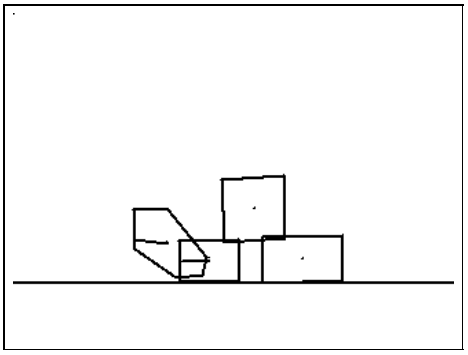
BODYMOTOR$(hand)$,
BODYMOTOR$(box)$.



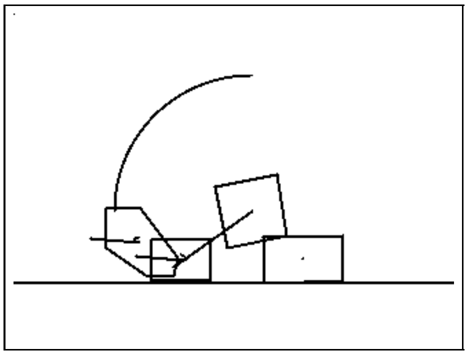
BODYMOTOR$(hand)$.

**(Examples)**

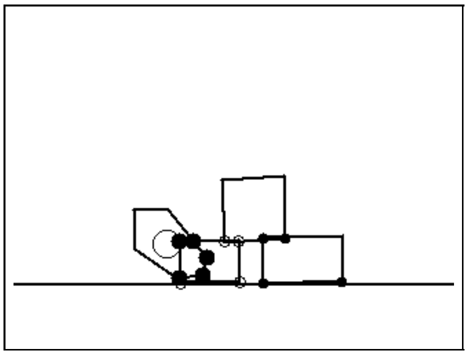
# Arch



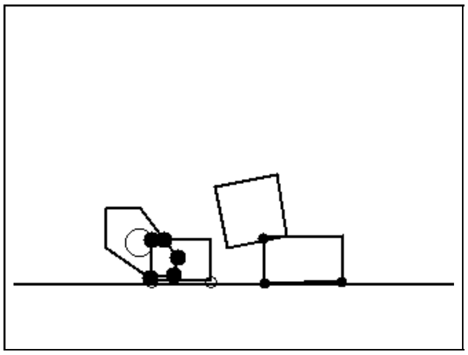
*45*



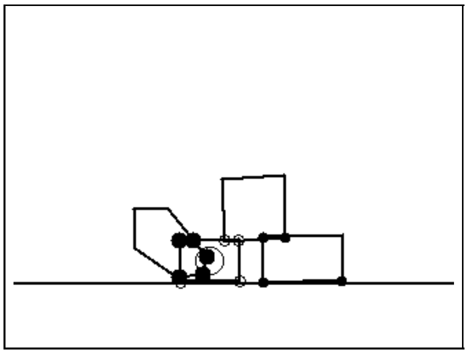
*52*



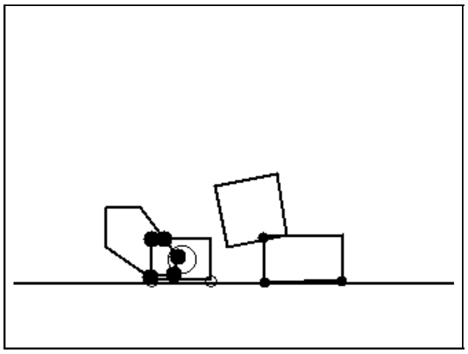
BODYMOTOR$(hand)$,
ATTACH$(hand, left)$.



BODYMOTOR$(hand)$,
ATTACH$(hand, left)$.

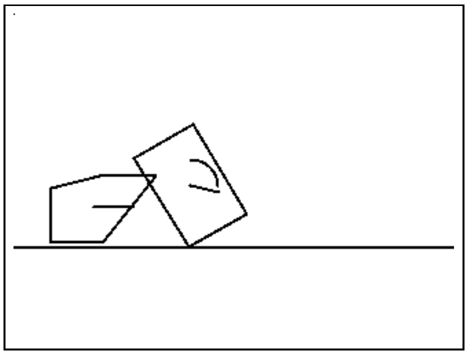


BODYMOTOR$(left)$,
ATTACH$(hand, left)$.

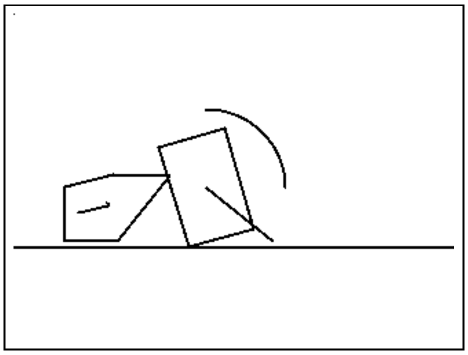


BODYMOTOR$(left)$,
ATTACH$(hand, left)$.

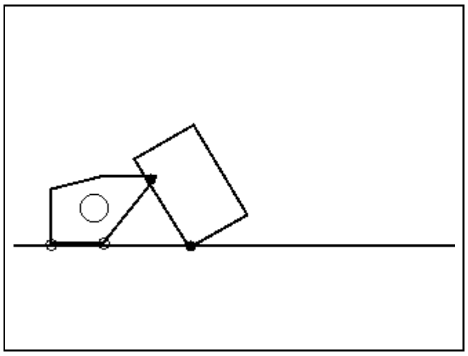
# Tip

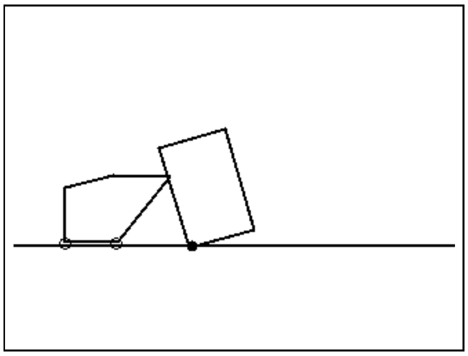


*30*

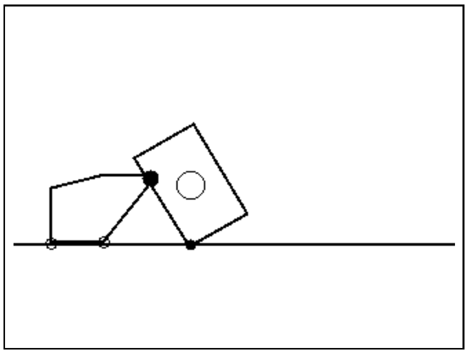


*34*





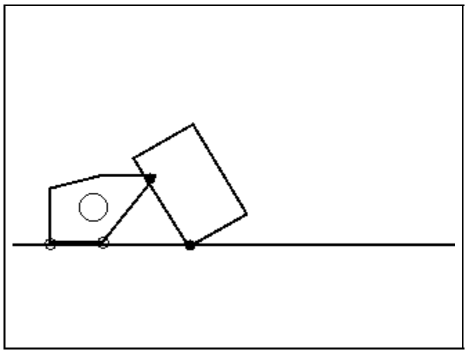
BodyMotor$(hand)$.



BodyMotor$(box)$,
Attach$(hand, box)$.

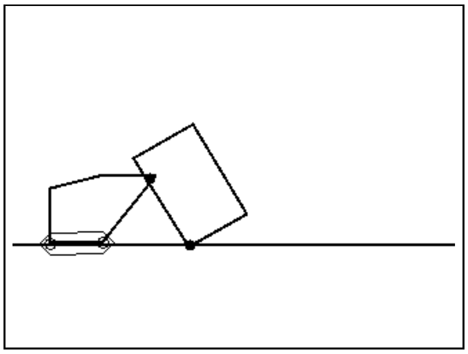
# Tip Detail (frame 30)

**Interpretation #1**



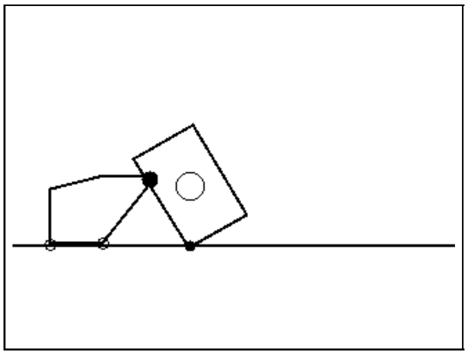
$\textsc{BodyMotor}(hand)$.

**Interpretation #2**



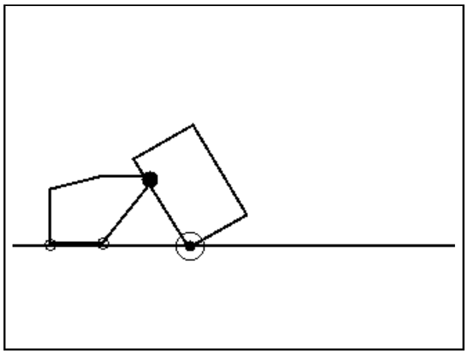
$\textsc{LinearMotor}(hand, table)$.

**Interpretation #3**



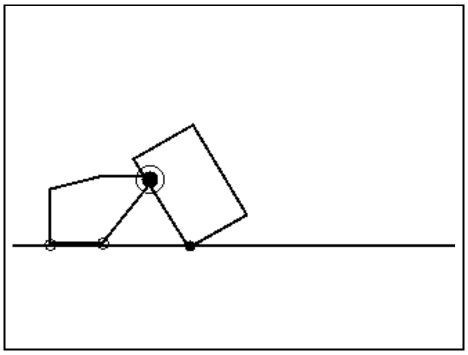
$\textsc{BodyMotor}(box)$,
$\textsc{Attach}(hand, box)$.

**Interpretation #4**



$\textsc{AngularMotor}(box, table)$,
$\textsc{Attach}(box, table)$,
$\textsc{Attach}(hand, box)$.

**Interpretation #5**



$\textsc{AngularMotor}(hand, box)$,
$\textsc{Attach}(hand, box)$.

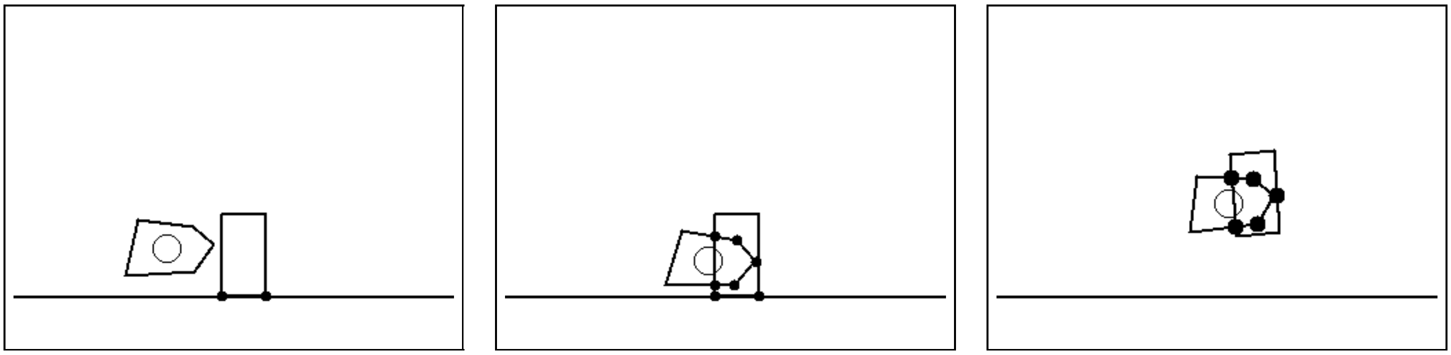
- All interpretations have a *force-balance*.
- Only interpretations #1 and #2 are realizable.

# Enhanced System
## (Integrate Information over Time)

- Ascribe properties to objects.
  - FLYER$(o)$: Object $o$ has arbitrary force and torque acting on it.
  - DRIVER$(o)$: Object $o$ can apply *tangential force* at any contact region.
  - ROTOR$(o)$: Object $o$ can apply a *torque* at any contact point.
- Object properties *persist* over time.
- Find minimal *covering* which explains whole sequence.

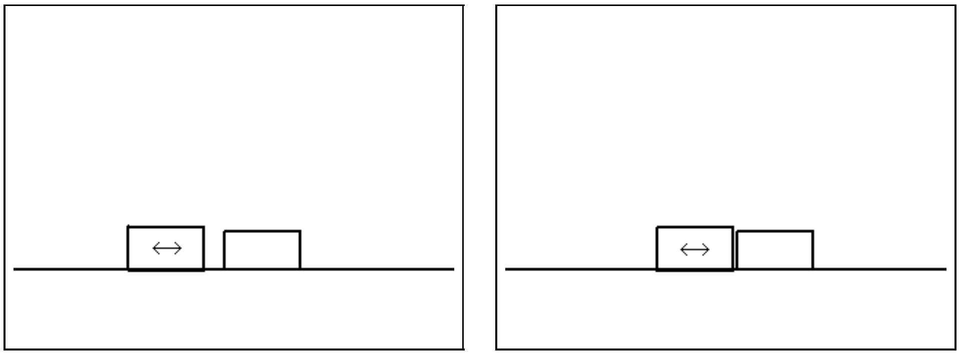
# Examples (Enhanced system)

## coke



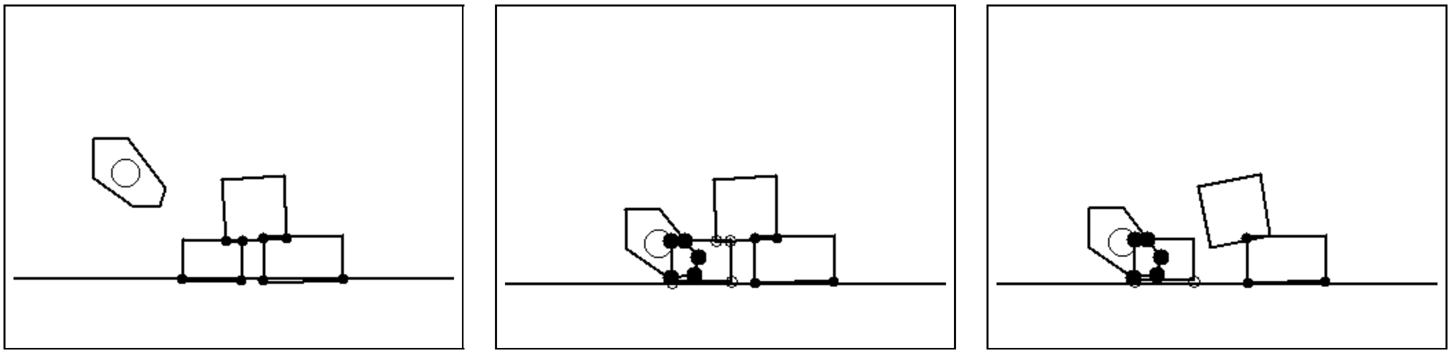
FLYER($hand$).

## cars



DRIVER($left$).

## arch



FLYER($hand$).

# Extensions

- **Transitions (changing contact relations)**
  (e.g., Frame $28$ of cars).
  - Detection of changing contact relations.
  - Dealing with collisions.
    (Theory of transfer of *momentum* as well as transfer of forces.)
- **More complete kinematic model**
  (e.g., Frame $30$ of tip).
  - Currently only check for pairwise motions of objects.
- **Full 3D model**
  - Feasibility test currently in 3D.
  - Requires 3D shape and motion data.
  - Alternative is *qualitative* depth model.

# Contributions of this work

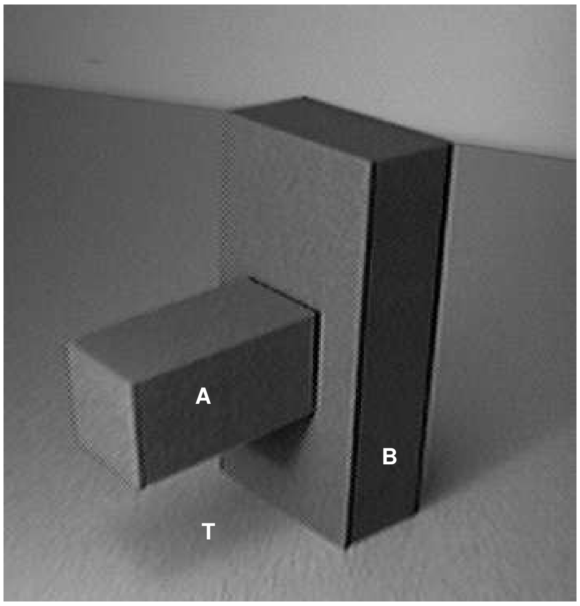
- First implementation of the 'computational perceiver' framework for scene interpretation (Jepson & Richards, 1991, 1993).
- Physics-based representation suitable for describing scene dynamics.
- Explicit computational procedure to test feasibility of interpretations. (Efficient implementation by linear programming.)
- Theory of preference ordering between multiple interpretations. (Efficient computational procedure exploits structure of the domain.)

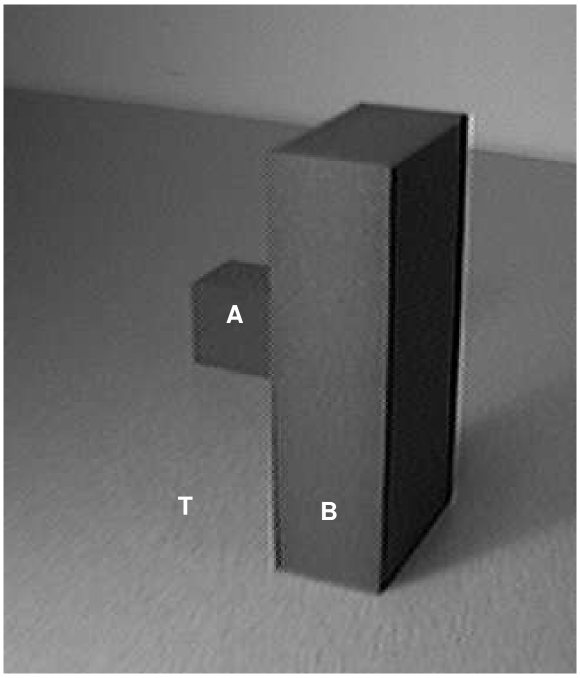
# Future Work

- **Potential applications of the force-dynamic representation:**
  - Reasoning about *support* in 3D scenes (Mann & Jepson, *in progress*).
  - Event recognition.
  - Recognition of objects based on *function*.
- **Other potential applications of the 'perceiver' approach:**
  - Inference of shape and form.
  - Scene labeling and figure completion.
  - Lighting, shadows, and scene structure.
- **Designing perceivers:**
  **Link between Vision and Knowledge Representation**
  - Logical framework for depiction and image interpretation. (Reiter & Mackworth, 1989)
  - Connection between preferences and *default reasoning* (Reiter, 1980) and *circumscription* (McCarthy, 1980).
  - Connection between preferences and *qualitative probabilities* (Richards, Jepson, & Feldman, *to appear*).

(Future Work)

# Support in 3D Scenes




**Preferred Interpretations:**

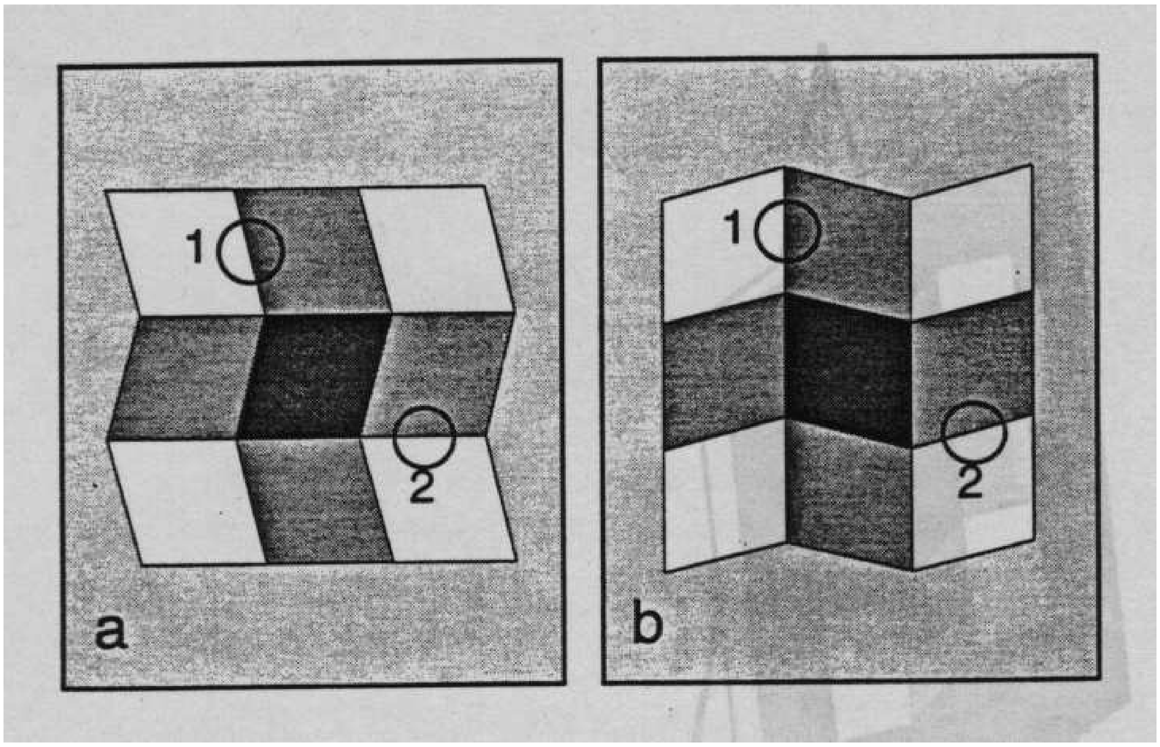
$\{C_{bt}, A_{ab}, \neg C_{at}\}$

$\{C_{bt}, \neg C_{ab}, C_{at}\}$
$\{C_{bt}, A_{ab}, \neg C_{at}\}$

(Future Work)

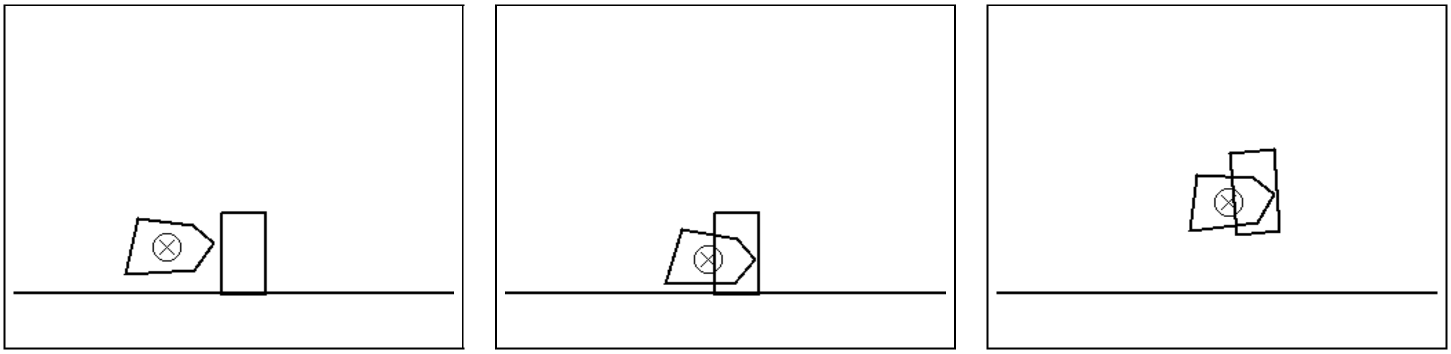
# Lighting, Shadows, and Scene Structure

- **Shadows and motion** (Kersten, Mamassian, & Knill)
- **Shading and reflectance** (Adelson & Pentland, 1990; Sinha & Adelson, 1993)

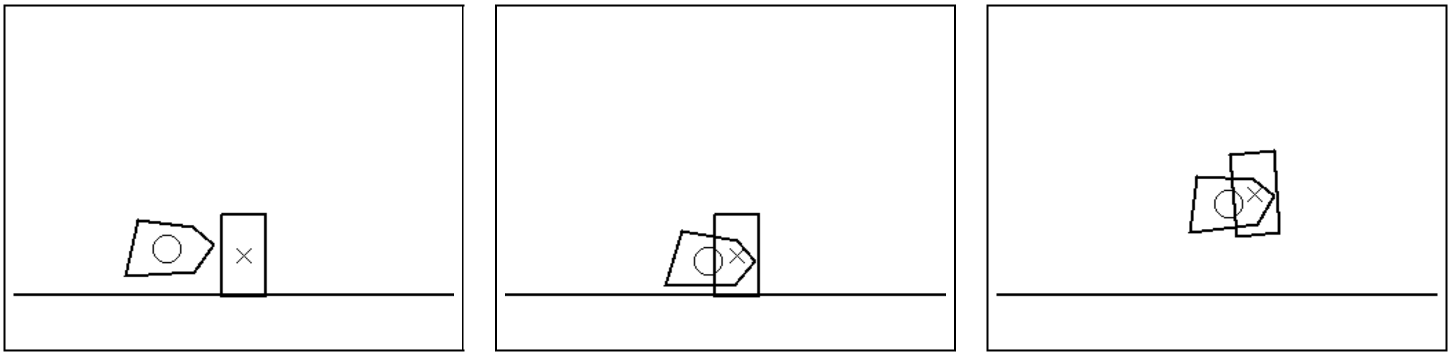


# Integration Over Time (example #2)

- Is the attachment *caused* by the hand or by the can?



FLYER($hand$), STICKY($hand$)



FLYER($hand$), STICKY($can$)

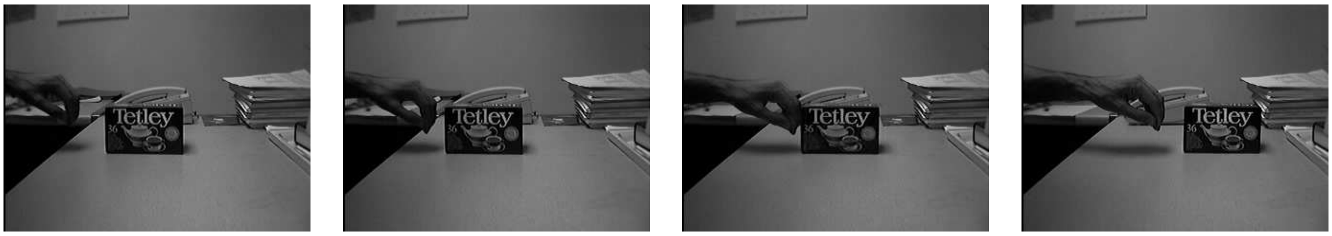
coke

30 37 48 60

cars

10 26 28 35

hit




25 29 32 40

arch

35 49 55 65

tip


21 31 35 39