ENSC 482

Introduction to Decision Making in Engineering

Instructor:

Prof. Shahram Payandeh
Networked Robotics and Sensing Laboratory

In performing any tasks, it is usually required for a person, an automated system or a person in collaboration with an automated system to reason about the objectives of the task.

This is required in order for a person (or an automated system) to reach a decision about the expected plans of action based on available information from the task and the environment where the task needs to be performed.

Objectives of the task are evaluated in conjunction with available resources (e.g. person's capabilities, method of actuation, hardware , computation resources or man-power) and available methods for collecting measurements associated with the task performance (if available) and the environmental factors (e.g. constraints) in order to rank the plan of actions for executing the task.

In some applications, ranking of the tasks can be accomplished without presence of any measures regarding to the performance of their outcomes.

In some applications, the ranking of the tasks and the eventual decisions are based on the various measures and consequences associated with possible actions.

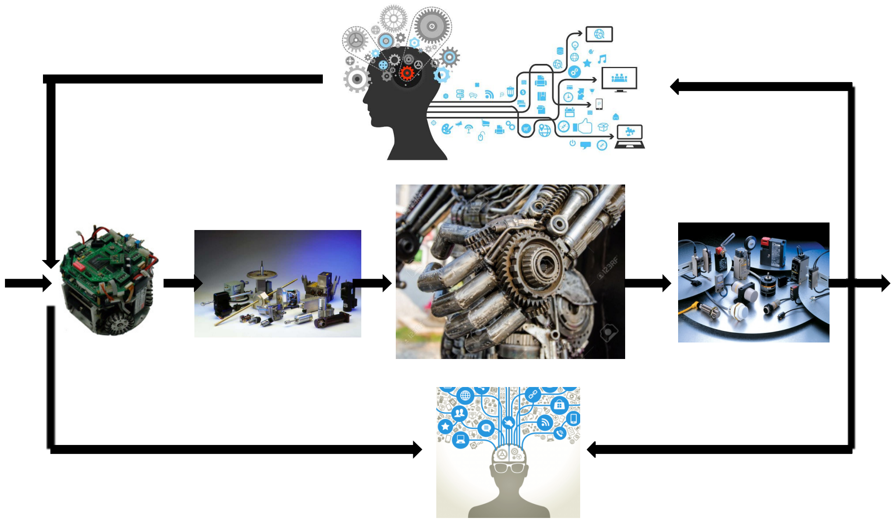
Ranking of possible actions can be structured in various forms for better management of data and/or better visualization of the dependencies of the incremental outcomes.

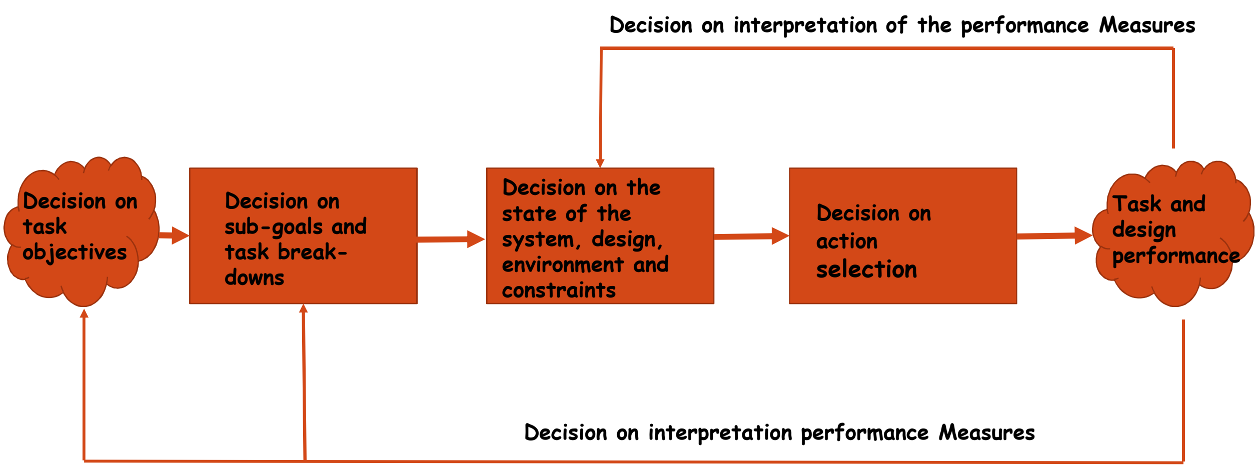
In most tasks where the monitoring is done through utilization of various sensing modalities, the decision on ranking of actions is mostly done by associating the actions to various interpretation of sensed data.

In most cases, the decision on selecting a suitable action item is based on classification of noisy data or uncertain information with some statistical properties.

In some design cases, decision on selection of the best action (i.e. a suitable design) can be determined based on exact definition of range of parameters, constrain functions and a suitable definition of the design objectives.

Various Stages of Decision making

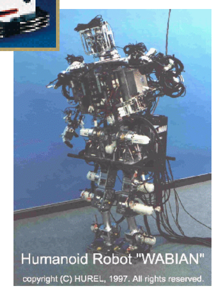


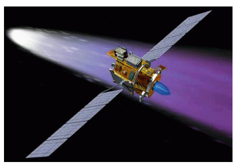
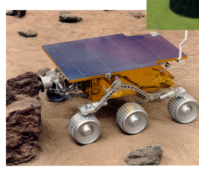


Decision Making in Robotics and Automation





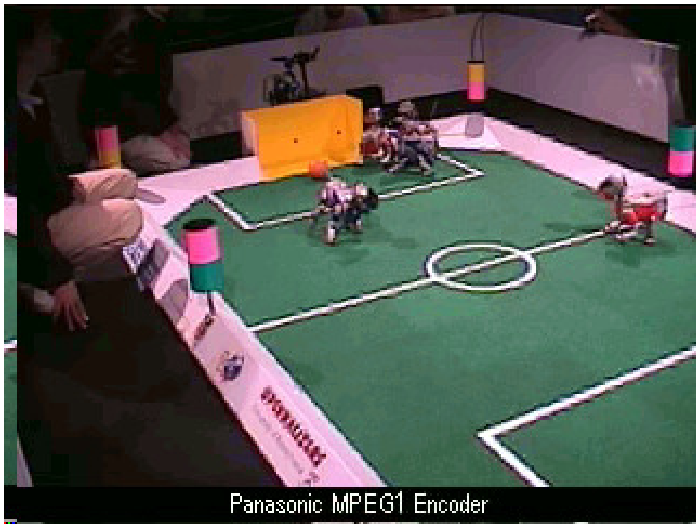



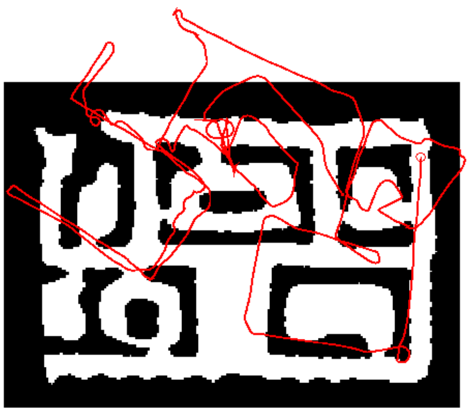

Uncertainty is Inherent and Fundamental

- Uncertainty can arises from a number of interrelated factors such as:
  1. Human movements and actions can be unpredictable
  2. Environment can be stochastic and unpredictable
  3. Robots actions are stochastic
  4. Sensors are limited and noisy
  5. Models can be inaccurate and incomplete

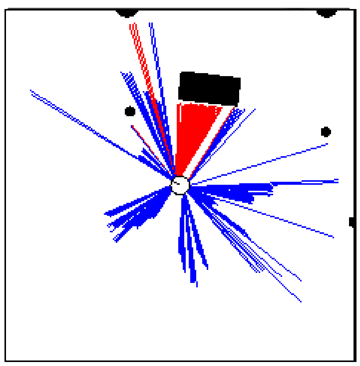
Educational Robotics



Nature of Sensed Data



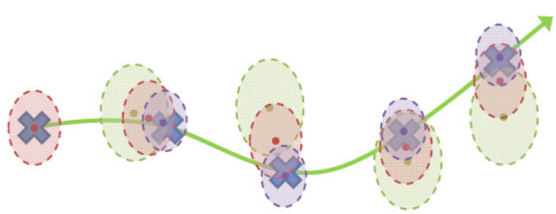
Odometry Data

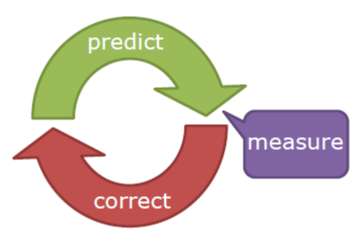


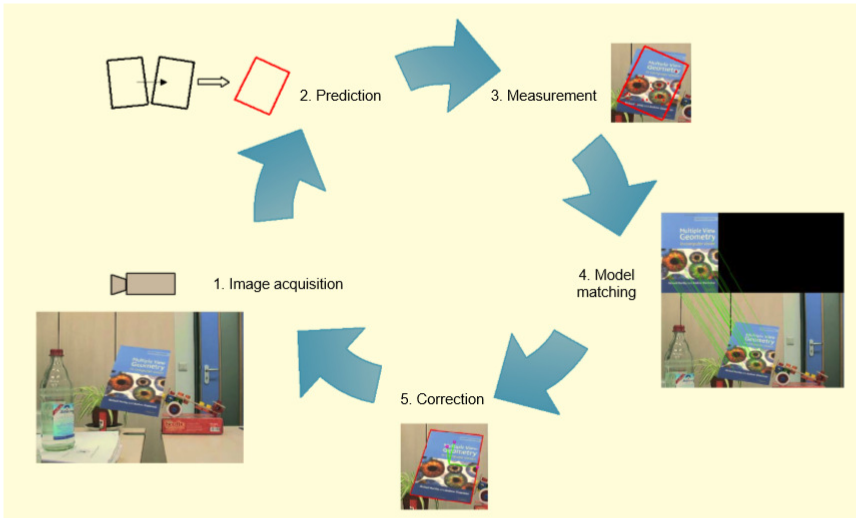
Range Data

Autonomous Flying Machines

Tracking Problem



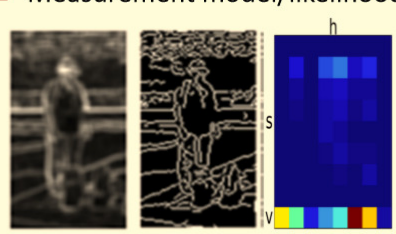




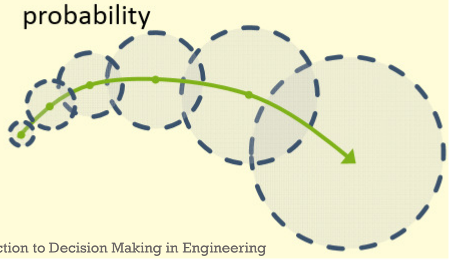
OpenCV – Raspberry PI

- State

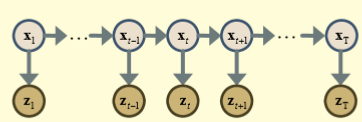
- Measurement model/likelihood



- Motion model/transition probability

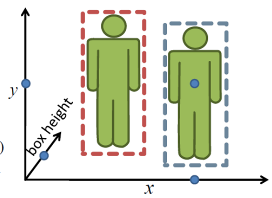


- Inference



Pedestrian Detection OpenCV

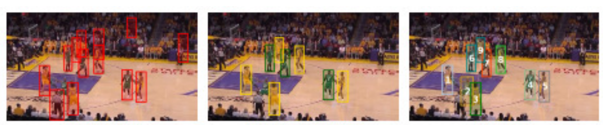
$\mathbf{x}_t = x$
$\mathbf{x}_t = (x, y)$
$\mathbf{x}_t = (x, y, h)$
$\mathbf{x}_t = \{\mathbf{x}_t^1, \mathbf{x}_t^2\}$



$\mathbf{x}_t = (x, y)$
$\mathbf{x}_t = (x, y, \dot{x}, \dot{y})$
$\mathbf{x}_t = (x, y, \dot{x}, \dot{y}, \ddot{x}, \ddot{y})$

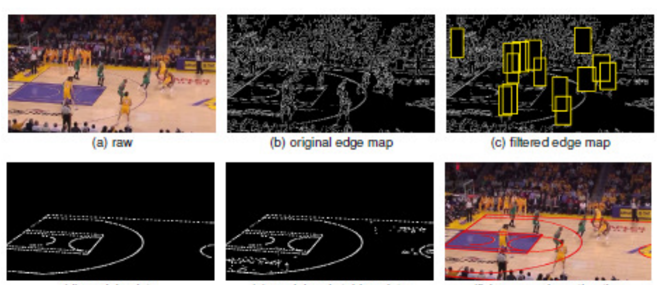
https://www.youtube.com/watch?v=OfDdNv-bMBo

https://github.com/yquemener/HeadCounter



(a) detection (b) team classification (c) tracking

https://docs.opencv.org/3.1.0/d2/d0a/tutorial_introduction_to_tracker.html

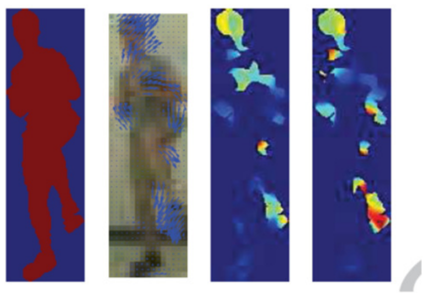


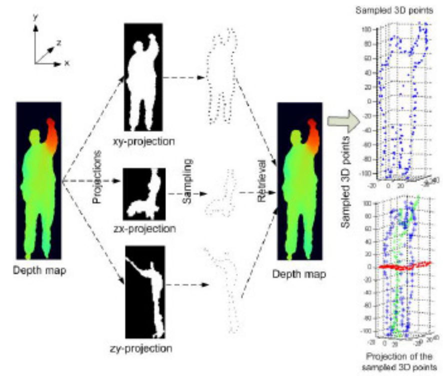
(a) raw (b) original edge map (c) filtered edge map

(d) model points (e) model and stable points (f) homography estimation

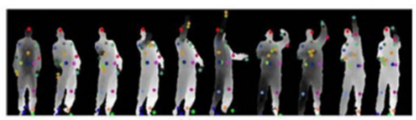
http://mathalope.co.uk/video-tracking-project-with-opencv/

Action and motion recognition





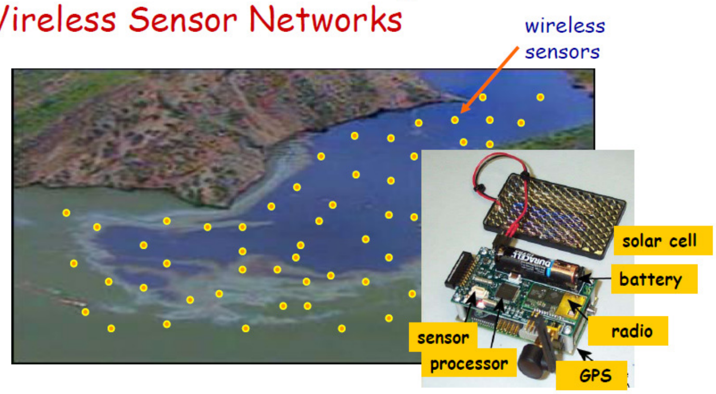
https://software.intel.com/en-us/realsense/d400



Compressed Sensing

Surveillance and Monitoring with Wireless Sensor Networks

https://hackaday.io/project/9754-raspberry-pi-sensor-network



Goal: Reconstruct an accurate map of contamination while keeping number of samples/communications to a bare minimum

Classical Approach:

$x \longrightarrow$ **Sensing/Sampling** $\xrightarrow{N}$ **Compression** $\xrightarrow{k}$ **Reconstruction** $\xrightarrow{N} \hat{x}$

- Sensing/Sampling:
  - Linear processing.
- Compression:
  - Non-linear processing.

$\rightsquigarrow$ *Why acquire N samples only to discard all but k pieces of data?*

Fundamental Idea:

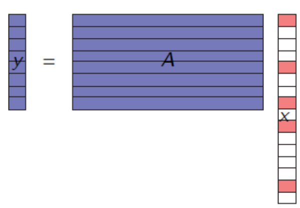
- Directly acquire "compressed data", i.e., the information content.
- Take more universal measurements:

$x \longrightarrow$ **Compressed Sensing** $\xrightarrow{(k <)n(<< N)}$ **Reconstruction** $\xrightarrow{N} \hat{x}$

What is Compressed Sensing...?

General Procedure:

- Signal $x \in \mathbb{R}^N$.
- x is k-sparse.
- Take $n << N$ linear, non-adaptive measurements using a matrix A.



Fundamental Questions:

- What are suitable signal models?
- When and with which accuracy can the signal be recovered?
- What are suitable sensing matrices?
- How can the signal be algorithmically recovered?

http://www.maths.manchester.ac.uk/~mlotz/teaching/jake/

Predicting spatial phenomena in large environments

Biomass in lakes

Salt concentration in rivers

Constraint: Limited energy source for making observations

Fundamental Problem: Where should we observe to maximize the collected information?

Challenges for informative path planning

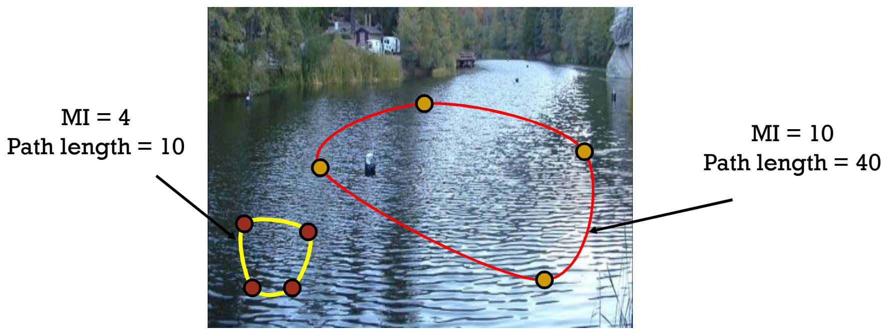
Use robots to monitor environment

Not just select best k locations A for given F(A). Need to
… take into account cost of traveling between locations
… cope with environments that change over time
… need to efficiently coordinate multiple agents

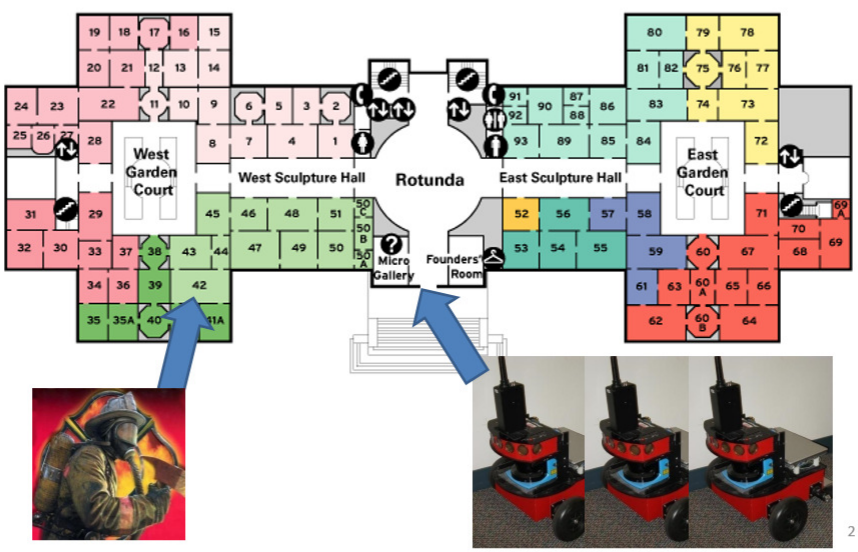
Want to scale to very large problems and have guarantees

How to quantify collected information?

- ***Mutual Information (MI):*** reduction in uncertainty (entropy) at unobserved locations

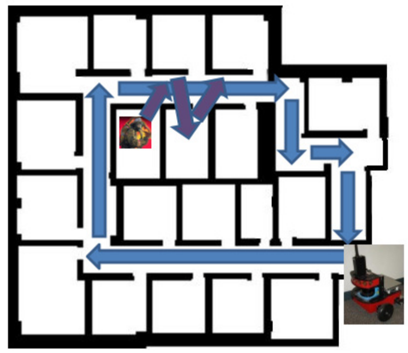


Multi-Robot Search for a Moving Target

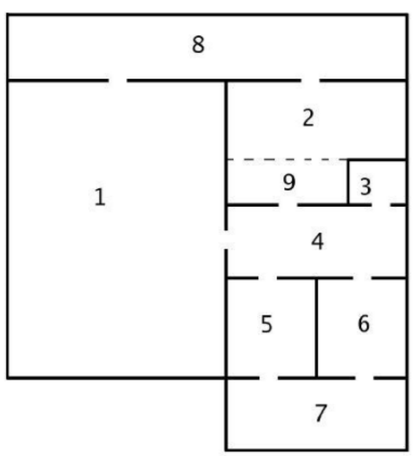


Problem: Efficient Search

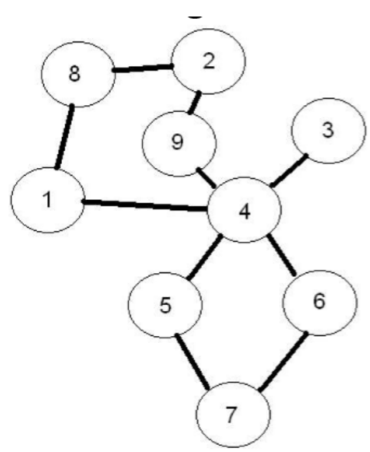
Quickly locate a non-adversarial target using prior knowledge and a rough model of how target moves.



Environment Representation



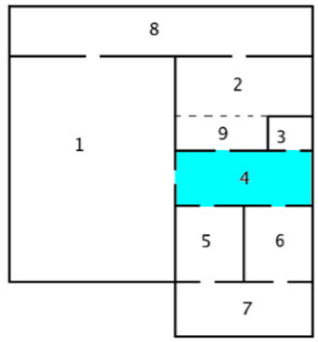
Discretization of the environment



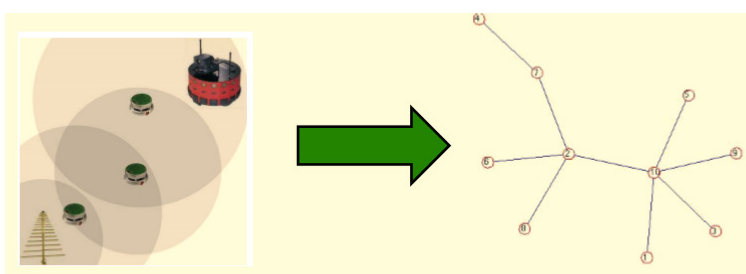
An example of undirected graph resulting from the discretization

Target motion modeling and estimation:
Define probability distribution over target's state
Use matrices to model capture and (Markov) diffusion.

$$
D = \begin{array}{cc}
 & \begin{array}{cccccccccc} C & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \end{array} \\
 & \left(\begin{array}{cccccccccc}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \frac{1}{3} & 0 & 0 & \frac{1}{3} & 0 & 0 & 0 & \frac{1}{3} & 0 \\
0 & 0 & \frac{1}{3} & 0 & 0 & 0 & 0 & 0 & \frac{1}{3} & \frac{1}{3} \\
0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 \\
0 & \frac{1}{6} & 0 & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & 0 & 0 & \frac{1}{6} \\
0 & 0 & 0 & 0 & \frac{1}{3} & \frac{1}{3} & 0 & \frac{1}{3} & 0 & 0 \\
0 & 0 & 0 & 0 & \frac{1}{3} & 0 & \frac{1}{3} & \frac{1}{3} & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & 0 & 0 \\
0 & \frac{1}{3} & \frac{1}{3} & 0 & 0 & 0 & 0 & 0 & \frac{1}{3} & 0 \\
0 & 0 & \frac{1}{3} & 0 & \frac{1}{3} & 0 & 0 & 0 & 0 & \frac{1}{3}
\end{array}\right)
\begin{array}{c} C \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \\ 9 \end{array}
\end{array}
$$



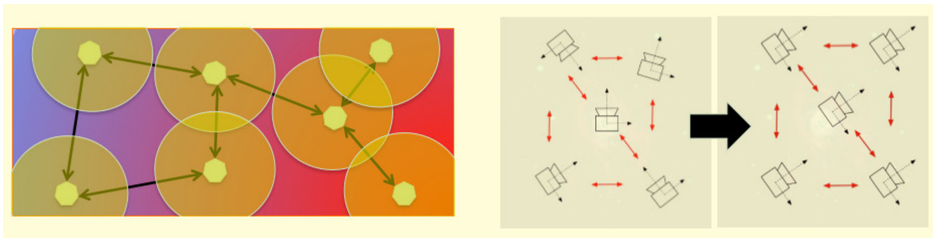
Every robot has a limited ability to: a) perceive the environment with onboard sensors (e.g. other robots); b) communicate information to other robots (via communication medium) and c) process information (i.e. gathered from onboard sensors or other sources on the sensor network)



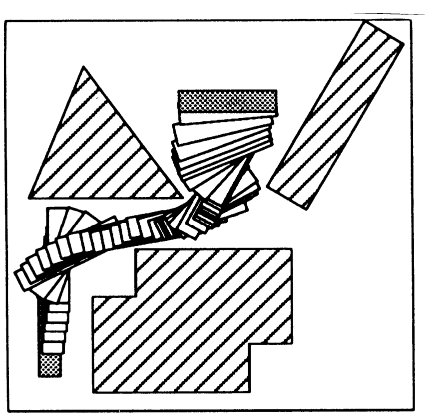
For example we can have: <u>Sensing graphs</u>: for each sensors, encode what robots can be locally sensed. <u>Communication graphs</u>: for each communication medium, encode with robots a communication link. <u>Action graph</u>: for each control action, encode what robots will be locally affected.

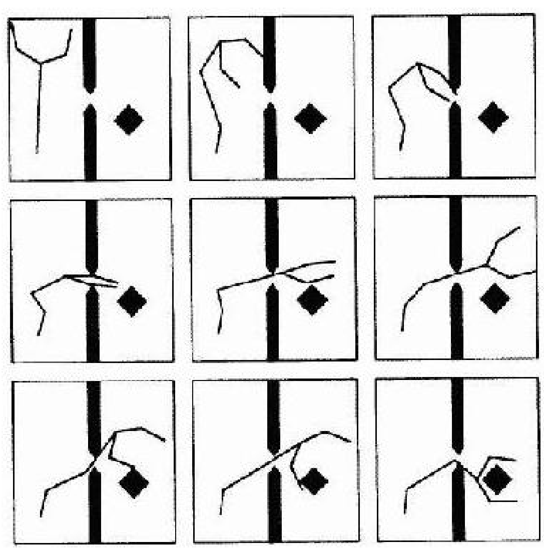
Example - The consensus protocol

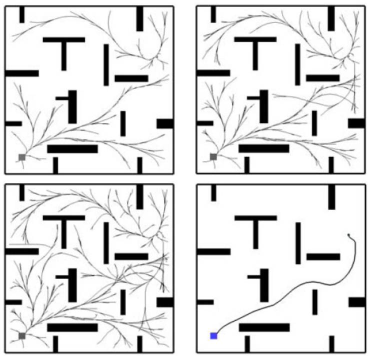
a) rendezvous: meet at a common point;
b) alignment: point in the same direction;
c) distributed estimation: agree on the estimation of some distributed quantity;
d) synchronization: agree on the same time.

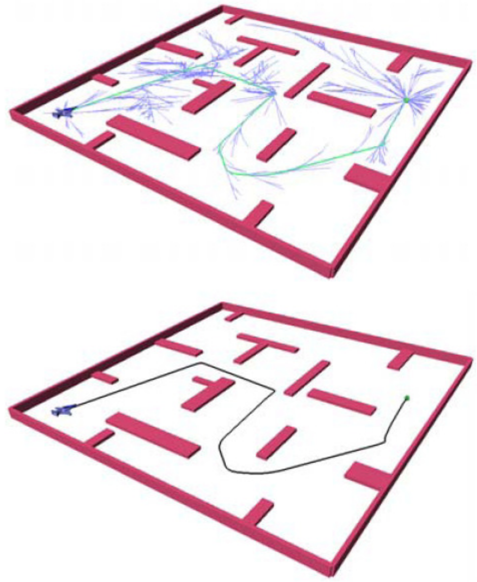


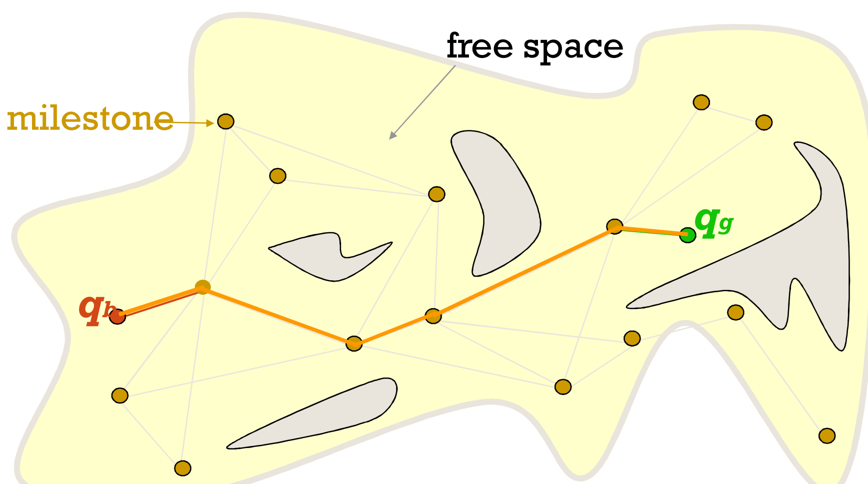
Path Planning

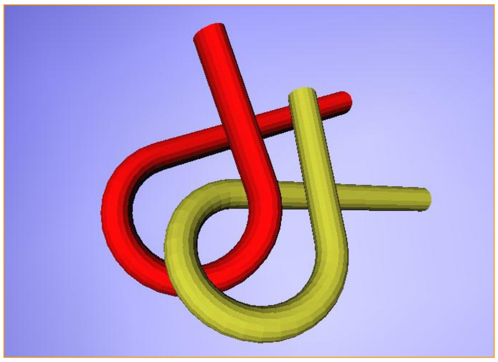


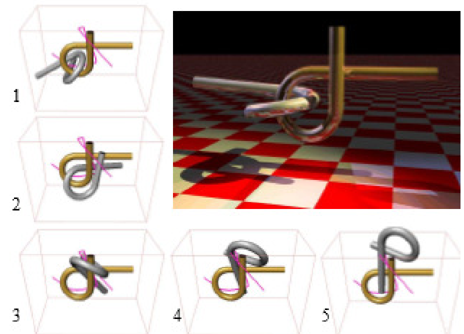




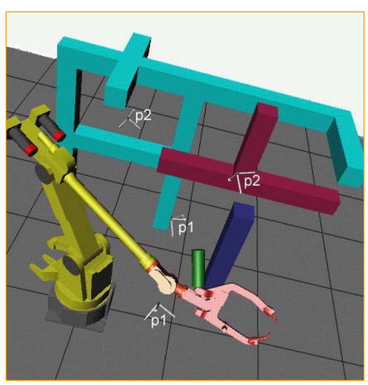


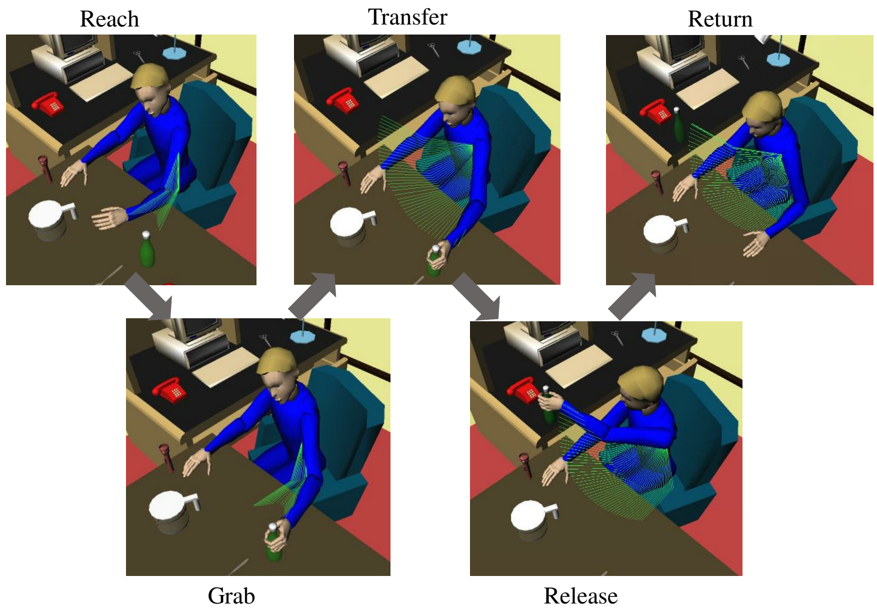

free space
milestone
q_b
q_g





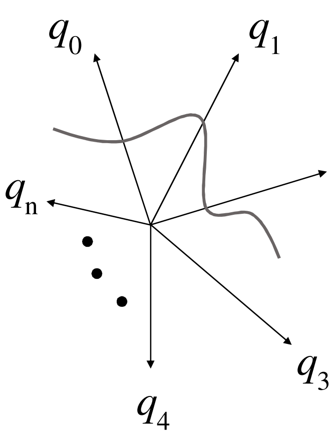
Trajectory and Motion Planning





Introduction to Decision Making in Engineering

35



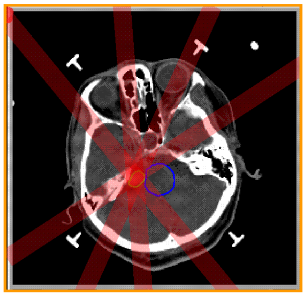
$$Q(t) = \begin{bmatrix} q_0(t) \\ \vdots \\ q_n(t) \end{bmatrix}$$

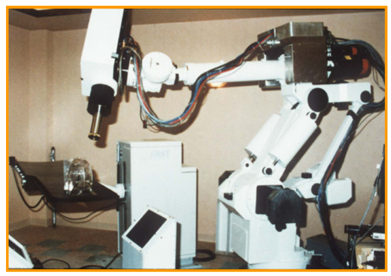
$$t \in [0, T]$$

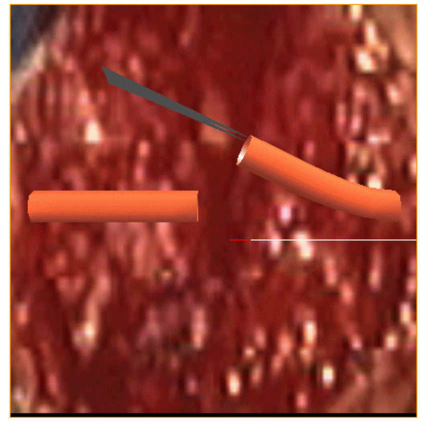
	Parts	DOF
L	19	68
H	51	118

Introduction to Decision Making in Engineering

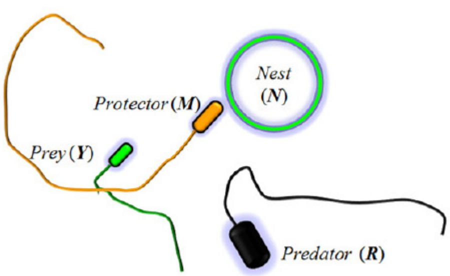
36

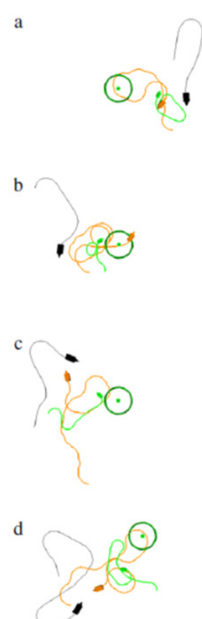




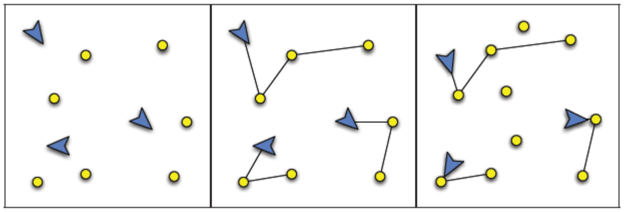


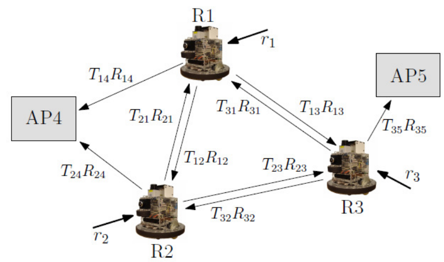
Predator-protector-prey

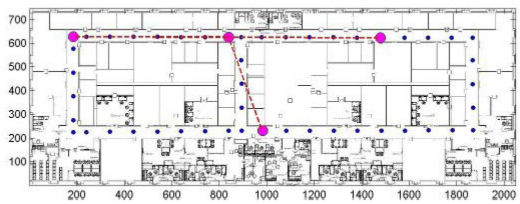




Robot Routing

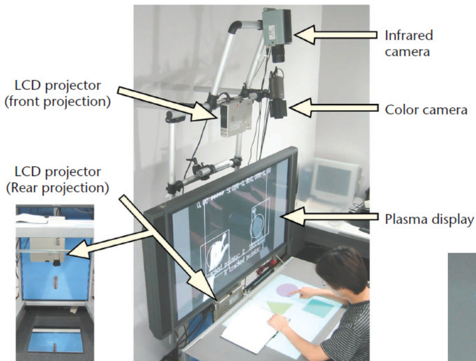


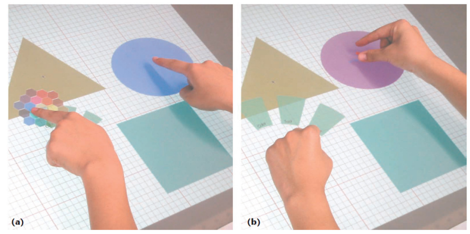


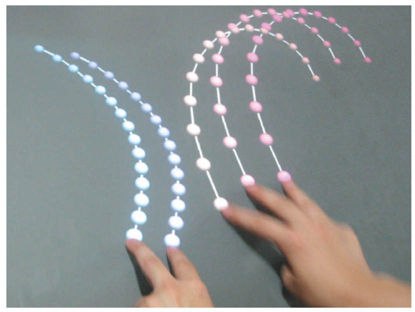


Hand gesture Recognition

Gesture Tracking

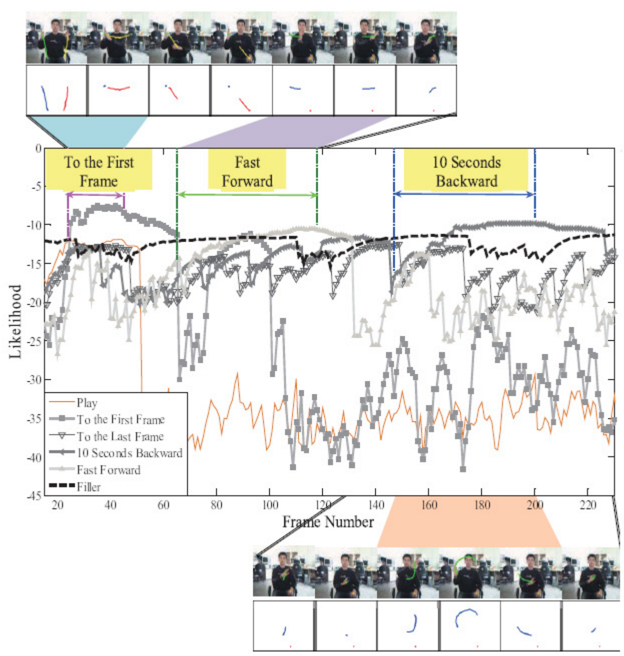






Introduction to Decision Making in Engineering

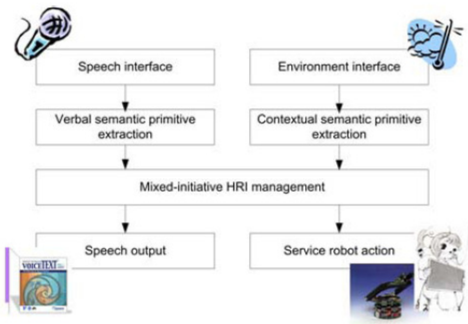
41

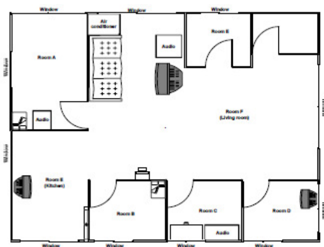


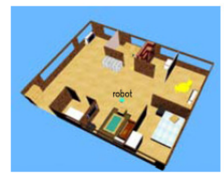
Introduction to Decision Making in Engineering

42

Human Robot Interaction

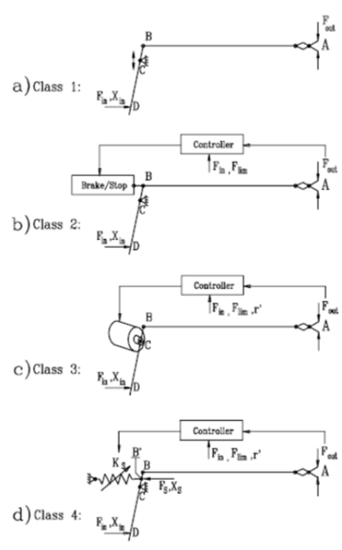


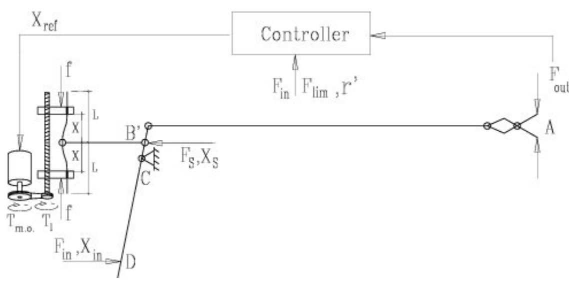


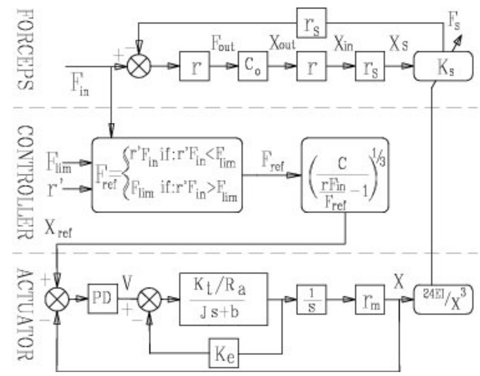


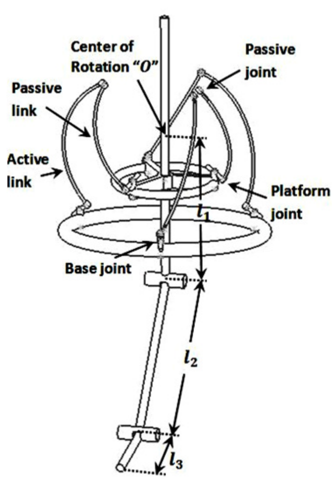
Personal Research Experiences in Using Decision Making Design Tools

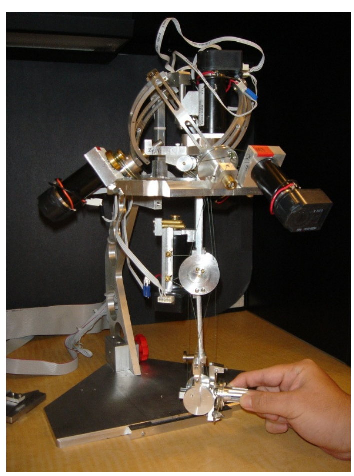
Novel Design of Surgeon-Computer User Interface Devices (Haptic Devices)

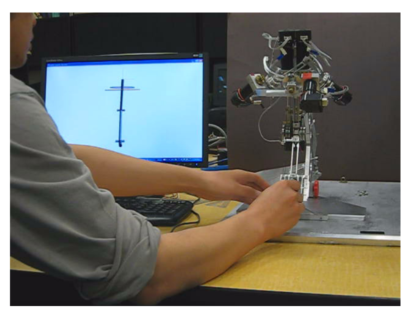


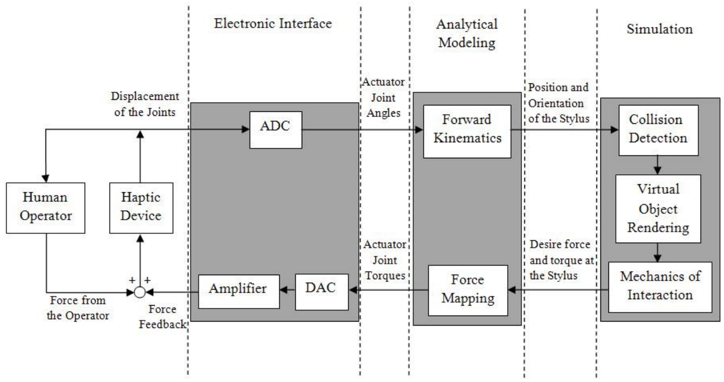

Controller
X_{ref}
F_{in} F_{lim}, r'
F_{out}
A
B'
C
D
F_s, X_s
F_{in}, X_{in}
f
x
L
$T_{m.o.}$
T_1


FORCEPS
CONTROLLER
ACTUATOR
r_s
F_s
F_{in}
F_{out}
X_{out}
X_{in}
X_s
r
C_o
K_s
F_{lim}
r'
X_{ref}
F_{ref}
PD
V
K_t/R_a
J s+b
$\frac{1}{s}$
r_m
X
$24EI/X^3$
Ke


Center of Rotation "O"
Passive joint
Passive link
Active link
Platform joint
Base joint
l_1
l_2
l_3

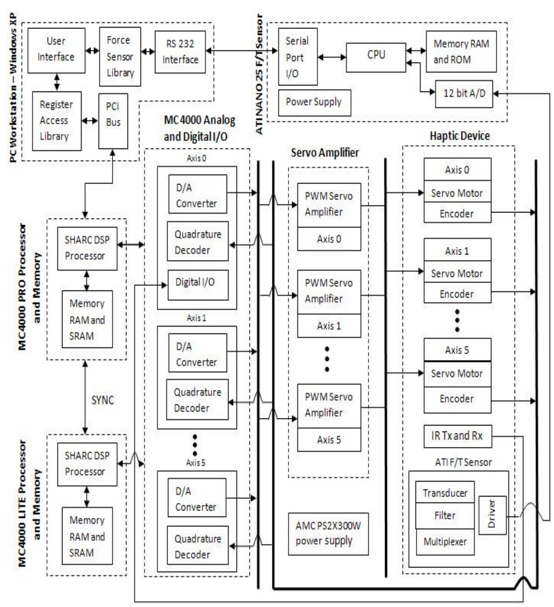






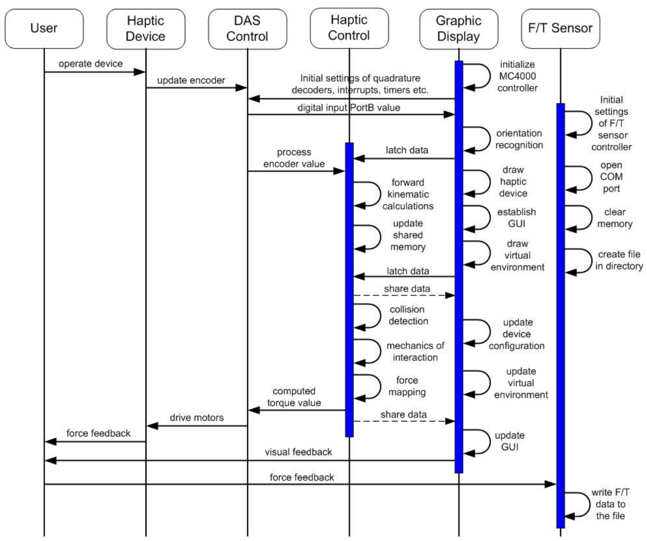
Introduction to Decision Making in Engineering

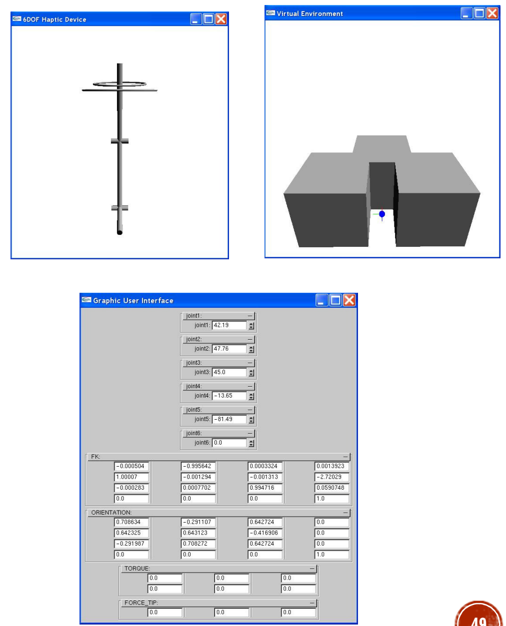
47

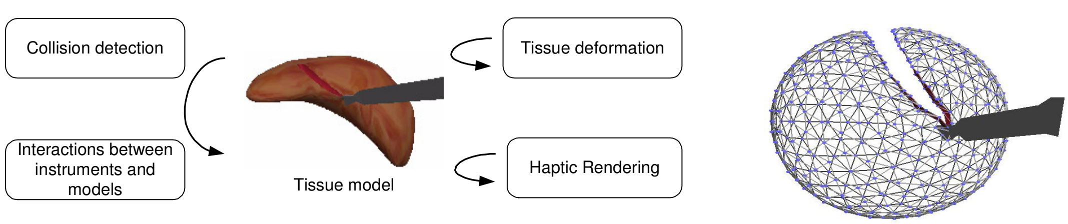


Introduction to Decision Making in Engineering

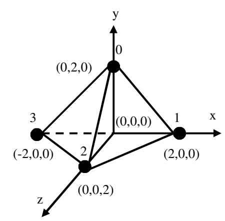
48

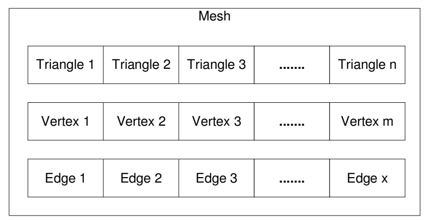

User
Haptic Device
DAS Control
Haptic Control
Graphic Display
F/T Sensor
operate device
update encoder
Initial settings of quadrature decoders, interrupts, timers etc.
digital input PortB value
initialize MC4000 controller
Initial settings of F/T sensor controller
orientation recognition
latch data
process encoder value
draw haptic device
open COM port
forward kinematic calculations
establish GUI
clear memory
update shared memory
draw virtual environment
create file in directory
latch data
share data
collision detection
update device configuration
mechanics of interaction
force mapping
update virtual environment
computed torque value
share data
drive motors
force feedback
update GUI
visual feedback
force feedback
write F/T data to the file


6DOF Haptic Device
Virtual Environment
Graphic User Interface

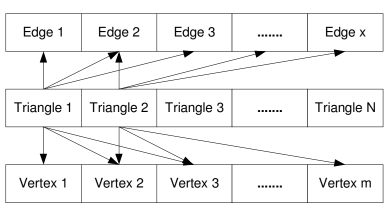

Collision detection
Interactions between instruments and models
Tissue model
Tissue deformation
Haptic Rendering

```
#VRML V1.0 ascii
Coordinate3 {
point [
   0 2 0,
   2 0 0,
   0 0 2,
   -2 0 -0,
]}
IndexedFaceSet {
coordIndex [
  0,2,1,-1,
  0,1,3,-1,
  0,3,2,-1,
  2,3,1,-1,
]}
```

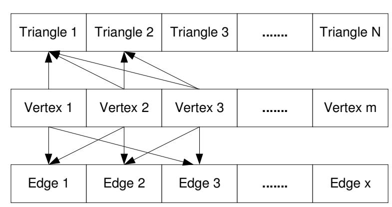




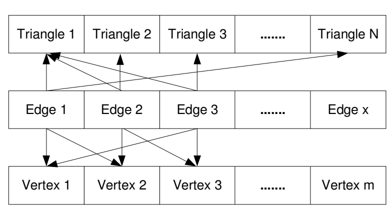
Mesh data structure



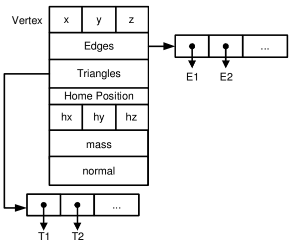
Triangle's pointers to its vertices and edges



Vertex's pointers to adjacent triangles and edges



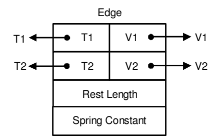
Edge's pointers to its end vertices and adjacent triangles



Vertex data structure

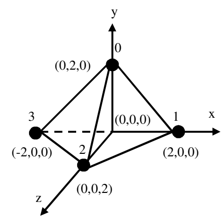
```
x=0; y=2; z=0;
vector<edge*> adjacentEdges
adjacentEdges [0] = *e01;
adjacentEdges [1] = *e02;
adjacentEdges [2] = *e03;

vector<triangle*> adjacentTriangles
adjacentTriangles [0] = *t021;
adjacentTriangles [1] = *t013;
adjacentTriangles [2] = *t032;
hx=0; hy=2; hz=0;
```



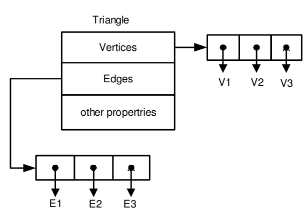
Edge data structure

```
class edge
{
    vertex* v1;
    vertex* v2;
    vector<triangle*> adjacentTriangles;
float springK;
float restLength;
}
```




```
v1 = *vertex0;
v2 = *vertex3;
adjacentTriangles[0] = *triangle032;
adjacentTriangles[1] = *triangle013;
springK = 2;
restLength = 0.7*2.828;
//the rest length is set 70% of the edge length
```

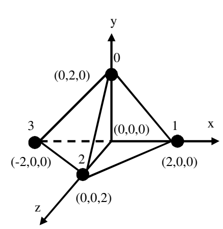
Example: edge 03



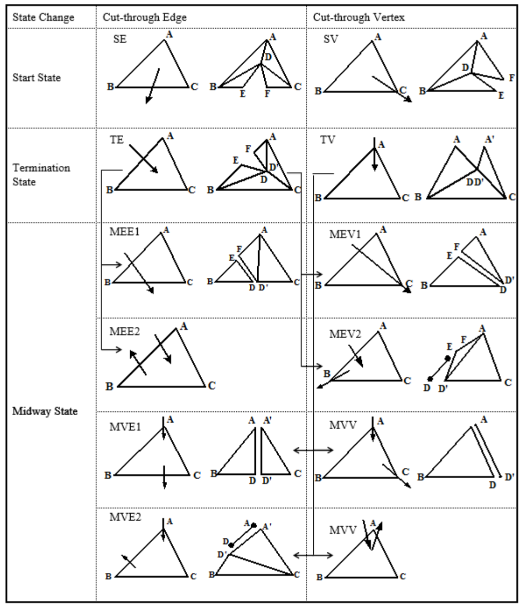
Triangle data structure

```
vector<vertex*> vertices
    vertices[0]=vertex0;
    vertices[1]=vertex2;
    vertices[2]=vertex1;
vector<edge*> edges
    edges[0]=edge02;
    edges[1]=edge21;
    edges[2]=edge10;
```

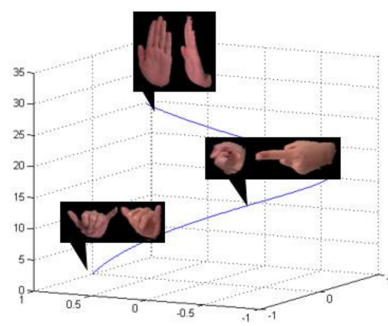
Example: Triangle 021

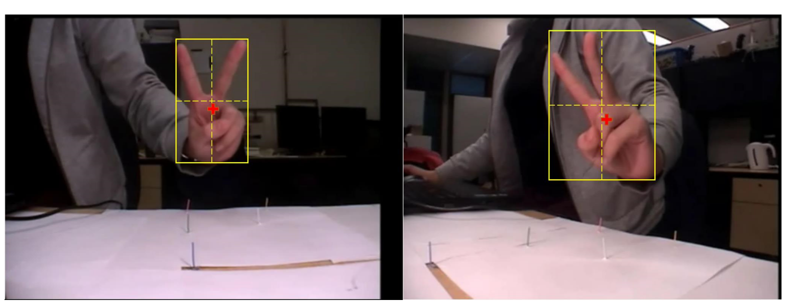


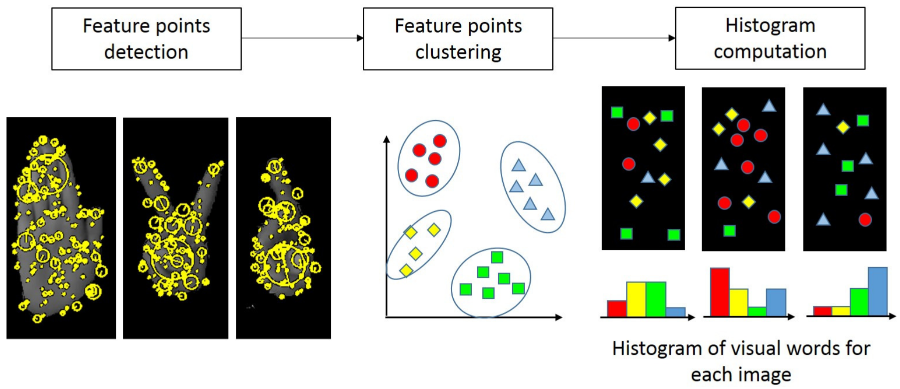
Decision making matrix
for dividing triangle



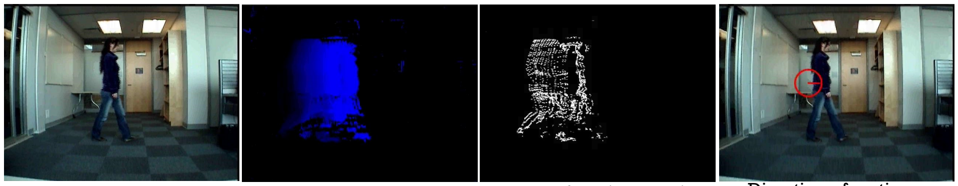
Hand Gesture Recognition



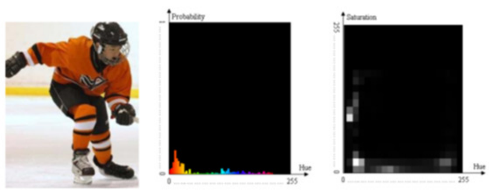


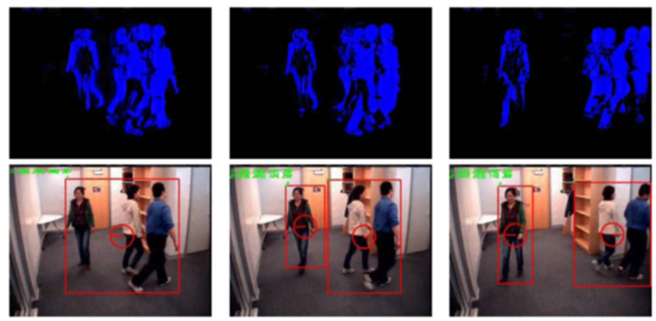


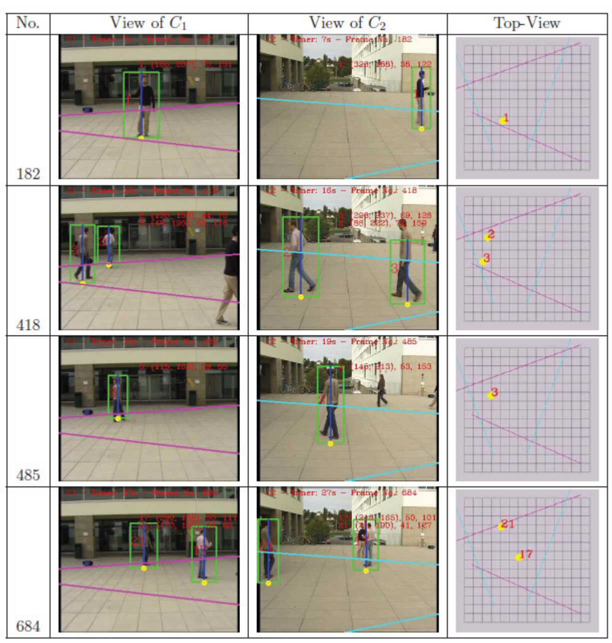
Testing Postures	Eight	Fist	Palm	Point	Six	V
Eight	71	0	1	4	2	0
Fist	0	46	0	15	9	0
Palm	0	0	61	0	0	2
Point	0	17	3	52	7	2
Six	1	9	0	1	43	1
V	0	0	7	0	11	67
Accuracy	98.6%	63.9%	84.7%	72.2%	59.7%	93.06%
	78.7% (340/432)					



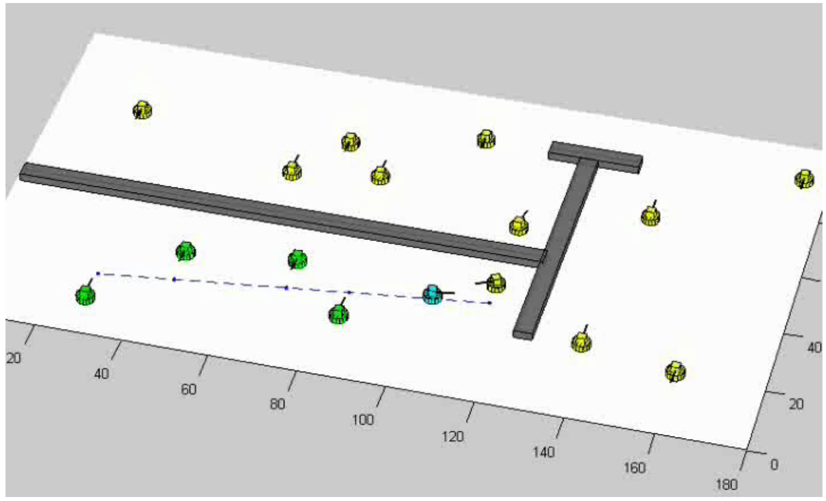
Source image | Motion history image | Image of motion gradient | Direction of motion



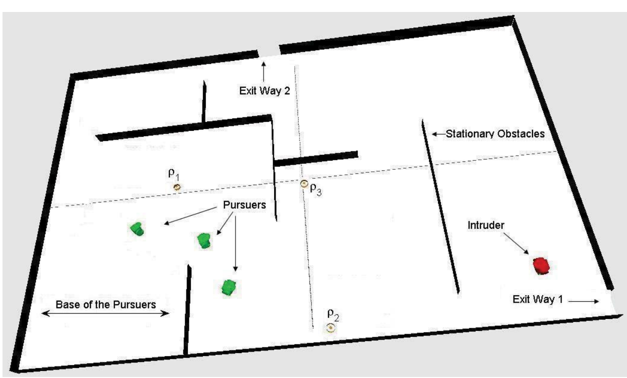




Robot Recharging Problem



Multi-Robot Pursuit Problem

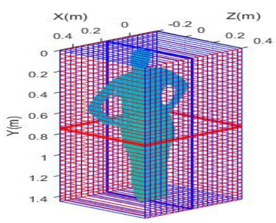


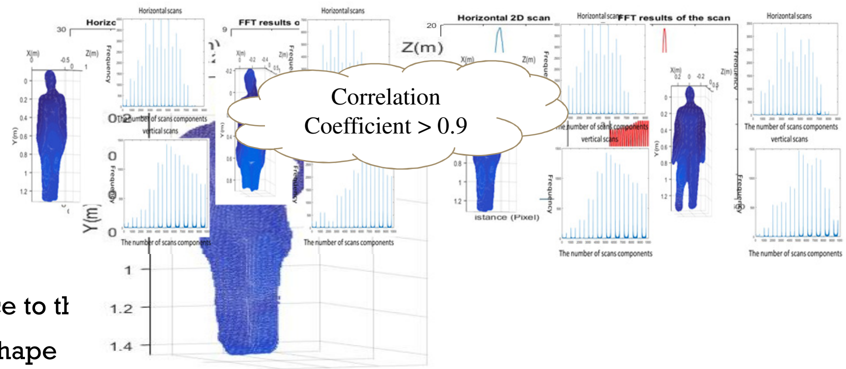
Group Profit Matrix

$$\Phi_{n\times m} = \begin{bmatrix} \Pi_1 \\ \Pi_2 \\ \vdots \\ \Pi_n \end{bmatrix} = \begin{bmatrix} \pi_1(\rho_1) & \pi_2(\rho_2) & \cdots & \pi_1(\rho_m) \\ \pi_2(\rho_1) & \pi_2(\rho_2) & \cdots & \pi_2(\rho_m) \\ \vdots & \vdots & \ddots & \vdots \\ \pi_n(\rho_1) & \pi_n(\rho_2) & \cdots & \pi_n(\rho_m) \end{bmatrix}$$

FEATURE SELECTION

2D cross-sectional scans | Distance | FFT | Body shape

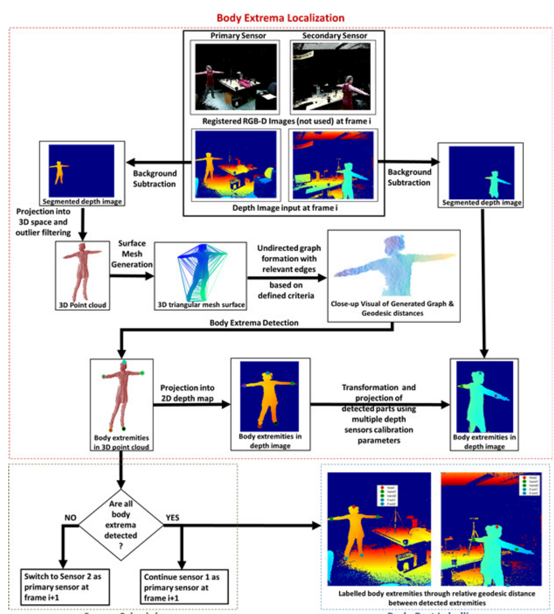




- ❖Not sensitive to distance to tl
- ❖Not sensitive to body shape

The proposed method

63



Introduction to Decision Making in Engineering

64

Sample references:

C. Barca, A. Sekercioglu and A. Ford, "Controlling Formation of Robots with Graph Theory".

Amarjeet Singh, Andreas Krause, Carlos Guestrin and William J. Kaiser, "Efficient information sensing with multiple robots"

Geoffrey Hollinger, Sanjiv Singh, Joseph Djugash, "Efficient Multi-robot Search for a Moving Target", The International Journal of Robotics Research, Vol. 28, No. 2, pp. 201-209, 2009

L. Kavraki, P. Svestka, J. Latombe and M. Overmars, "Probabilistic roadmaps for path planning in high-dimensional configuration space, IEEE Transaction on Robotics and Automation, Vol. 12, Nov. 4, 1996

S. La Valle and S. Hutchinson, "Path Selection and Coordination for Multiple robots vi nah Equilibria", Proceedings of IEEE international Conference on Robotics and Automation, pp. 1847-1852, 1994

S. Bhattacharya and S. Hutchinson, "On the existence of nash Equilibrium for a two player pursuit-Evasion game with visibility constraints", Algorithm foundations of robotics VIIII, pp. 251-265

G.N. Givigi and H.M. Schwartz, "A Game Theoretic Approach to swarm robotics", Journal of Applied Bionics and Biomechanics, Vol. 3, Issue 3, pp. 131-142, 2006

K. Oka and Y. Sato, "Real-time fingertip tracking and gesture recognition", IEEE Computer Graphics and Applications, Vol. 22, Issue 6, pp. 64-71, 2002.

- A. Rezaee, A. Raie, A. nadi and S. Shiry, "using Bayesian Network for Robot Behaviour with sensor fault", Electronics and electrica engineering, 2011, No. 2(108),
- A. Boyali, N. Hashimoto and O. Matsumoto, "Paradigm shift in continuous signal pattern classification: mobile ride assistance system for two-wheeled mobility robots",
- S. Khare, "Finger gesture and pattern recognition based device security system", Proceedings of IEEE international conference on signal processing and communication, 2015
- Z. Chen, J. Kim, J. Liang, J. Xhang and Y. Yuan, "real-time hand gesture recognition using finger segmentation", The Scientific World Journal, Vol. 2014.
- J.K. Aggarwal and L. Xia, "Human activity recognition from 3D data: A review", Pattern Recognition Letters, Vol. 48, pp. 70-80, 2014
- S. Lavalle and J. Kuffner, "Randomized kinodynamic planning", The International Journal of Robotics Research, Vol. 20, Issue 5, pp. 378-400, 2001.
- J. Martin, J. de Lope and D. Maravall, "Analysis and solution of a predator–protector–prey multi-robot system by a high-level reinforcement learning architecture and the adaptive systems theory", Robotics and autonomous systems, Vol. 58, pp. 1266-1272, 2010.
- F. Bullo, E. Frazzoli, M. Pavone, K. Savla and S. Smith, "Dynamic Vehicle Routing for Robotic Systems", Proceedings of IEEE, Vol. 99, Issue 9, pp. 1482-1504, 2011.

- M. Zavlanos, A. Ribeiro and G. Pappas, "Mobility & Routing Control in Networks of Robots", Proceedings of IEEE Conference on Decision and Control, 2010.
- O. Tekdas, W. Yang and V. Isler, "Robotic routers: algorithm and implementation", The International Journal of Robotics Research, Vol. 29, Issue 1, pp. 110-126, 2009.
- W. Caselton and J. Zidek, "Optimal monitoring network designs", Journal of Statistics & Probability Letters, Vol. 2, Issue 4, pp. 223-227, 1984.
- S. M. La Valla, "Planning algorithms", Cambridge University Press, 2006.
- J. P. Wachs, M. Kolsch, H. Stern and Y. Edan, "Vision-based
- Hand-gesture applications", Communication of the ACM, Vol. 54, No. 2, 2011.
- J. Hong, Y. Song and S. Cho, "A hierarchical Bayesian network for mixed-initiative human-robot interaction", Proceedings of IEEE Internation Conference on Robotics and Automation, pp. 3808-3813, 2005.
- W. Lu, J. Ting, J. Little and K. Murphy, "Learning to track and identify players from broadcast sports videos", Transaction of IEEE on Pattern Analysis and Machine Intelligence, Vol. 35, Issue 7, pp. 1704-1716, 2013.

http://www.sciencedirect.com/science/article/pii/S1077314215001794

http://www.sciencedirect.com/science/article/pii/S1077314215001794

http://www.sciencedirect.com/science/article/pii/S1077314213000295

http://www.sciencedirect.com/science/article/pii/S1077314215002568

http://www.sciencedirect.com/science/article/pii/S1077314216000680

http://www.sciencedirect.com/science/article/pii/S1077314216000138

datasets
https://rgbd-dataset.cs.washington.edu/
http://www0.cs.ucl.ac.uk/staff/M.Firman/RGBDdatasets/

OpenCV (*Open Source Computer Vision*)
http://http://opencv.org/

OpenGL (Open Graphics Library)
https://www.opengl.org