Introduction to Computer Vision

Dr. Gerhard Roth

COMP 4102A
Winter 2013

General Information

Instructor: Adjunct Prof.

Gerhard Roth gerhardroth@rogers.com – read hourly

groth@connect.carleton.ca – read daily

Office Hours: 10:00 to 11:30, Mon/Wed HP5331

(right after the lectures)

TA: none, I am the TA and marker for everything.

Course website:

http://people.scs.carleton.ca/~roth/comp4102a-13/

Linked from http://www.scs.carleton.ca/courses/

Lectures in pdf are dated by name, so you can check for updates – which are periodic

Course Organization

Textbook: Introductory Techniques for 3-D Computer Vision, by Trucco and Verri

No longer in print, CD has scanned chapters

Also contains SzeliskiBook_20100903_draft which will be used for some of the course (to be decided)

Two parts:

Part I (Before Feb. break) – Introduction, Review of linear algebra, Image formation, Image processing, Edge detection, Corner detection.

Part II (After Feb. break) – Camera calibration, Homography, Correspondence, Stereo, Epipolar geometry, Active sensors

Evaluation

Four or five assignments (40%)

-two before break and two after

-three or four of them will involve programming

-submit via CuLearn or put in the dropbox at 3115 HP (or e-mail me)

Two mid-terms (60% - 30% each)

-one just before reading week break, one at end of term (just before the exam period)

-will cover material up to that point, so once tested material in first part is not tested again

What is Computer Vision?

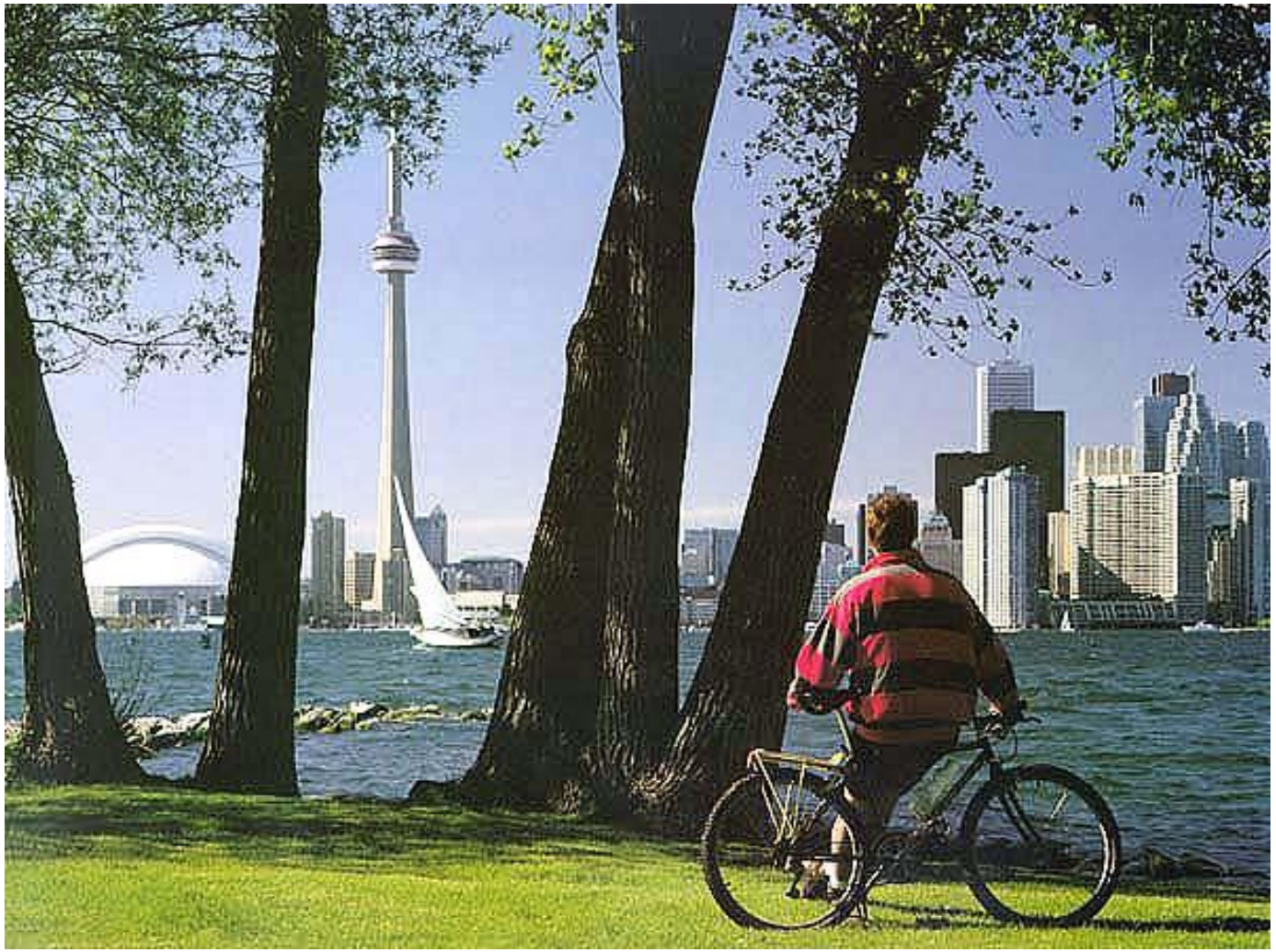
The goal of computer vision is to develop algorithms that allow computer to “see”.

Computer vision is sometimes called the inverse problem of computer graphics

Also called

- Image Understanding
- Image Analysis
- Machine Vision

General visual perception is hard



Digital Image

232	182	143	151	151	148	148	143	145	139	143	136	139	136	134	132	129	130	126	124	115	116	115	104	109	102	100	101
244	218	160	149	145	147	145	143	139	142	140	139	134	134	130	131	125	120	120	116	110	110	107	100	100	97	95	97
246	233	196	145	145	146	141	141	137	134	140	133	133	125	131	125	114	121	116	116	109	101	95	101	97	87	89	91
248	242	222	161	142	140	145	137	138	135	129	127	127	122	124	118	116	113	102	110	99	102	98	94	91	88	91	90
252	246	234	192	143	139	136	134	133	129	131	127	124	121	117	114	111	105	108	95	101	102	86	88	91	84	84	99
252	249	242	215	151	137	134	134	129	126	126	121	120	116	113	111	108	104	99	94	102	93	89	96	79	87	92	112
252	248	242	227	169	134	135	124	122	120	125	121	116	115	105	112	102	99	92	98	93	88	89	74	87	65	97	111
253	246	244	236	192	134	125	123	119	120	118	116	112	107	110	95	104	94	89	96	84	86	79	77	65	79	105	119
252	250	246	238	210	144	126	118	120	115	116	116	98	105	103	102	96	93	91	82	80	79	75	70	82	81	108	119
250	251	247	239	219	161	127	117	117	109	105	107	100	104	99	100	98	79	98	70	75	80	72	65	86	83	113	124
252	249	247	241	226	177	122	120	116	106	108	110	91	103	93	99	89	88	79	80	72	74	76	65	84	87	109	123
249	249	247	241	231	191	131	116	110	109	106	98	95	90	102	83	78	86	85	80	70	75	69	81	79	96	122	121
248	249	247	244	238	206	133	121	101	103	94	97	91	87	87	83	83	82	77	78	81	61	73	65	86	99	118	120
247	250	248	244	237	215	149	115	102	105	91	94	80	91	79	83	81	70	71	75	74	71	78	74	76	108	117	119
250	247	246	243	239	218	159	108	100	87	100	88	92	83	85	77	81	63	80	70	63	73	70	78	81	110	120	116
248	245	244	241	239	224	170	113	103	94	89	86	84	83	74	81	68	78	76	66	66	70	73	65	92	108	115	123
248	244	244	242	237	226	179	123	98	94	84	74	88	77	71	76	71	78	68	67	63	72	72	75	94	109	115	124
247	244	245	241	238	221	183	123	95	87	89	73	77	79	71	65	78	56	69	66	62	61	70	69	90	113	118	118
247	246	244	242	236	219	185	120	100	84	82	79	66	67	76	72	69	55	61	56	65	57	70	78	95	106	119	116
246	245	244	241	231	216	190	126	91	86	77	77	72	71	76	60	69	60	57	52	66	55	62	75	87	110	110	117
245	244	244	237	231	221	189	133	97	83	70	73	62	59	77	44	65	66	60	70	51	43	67	75	95	107	116	111
244	244	241	237	230	222	188	133	90	83	77	77	59	78	60	67	62	61	66	72	62	51	62	71	96	105	115	108
242	242	237	236	232	219	187	126	83	79	70	64	58	66	63	67	54	65	51	65	58	54	62	73	77	92	107	94
241	241	238	236	229	216	186	125	85	77	70	66	64	53	63	55	56	53	67	39	52	25	23	9	11	51	66	77
241	239	237	237	228	214	185	127	92	83	64	66	69	62	61	65	32	42	12	7	6	15	65	123	146	160	167	172
240	239	237	236	225	208	178	123	89	67	72	67	49	54	27	10	7	23	103	142	162	167	169	168	171	172	172	178
238	236	236	229	221	203	174	125	77	82	55	33	23	9	79	135	163	173	174	175	174	170	171	167	167	172	169	173
235	235	231	228	215	198	165	122	84	43	14	57	132	166	176	175	179	177	176	178	178	173	169	172	167	168	171	162
231	231	227	223	210	191	163	110	44	95	159	174	175	179	178	180	183	180	179	177	175	175	174	173	169	168	171	156
230	226	225	220	202	187	169	151	175	180	182	177	182	182	183	184	184	184	181	182	181	178	182	179	172	161	160	155
223	224	220	213	198	191	185	186	182	182	178	179	184	185	191	189	189	192	188	192	193	194	192	187	179	161	153	147
220	219	213	203	191	182	181	177	176	173	175	180	182	184	192	192	193	195	200	203	203	206	205	202	192	164	150	151
212	209	200	188	177	173	174	171	169	165	173	176	180	187	191	192	195	195	201	203	207	210	208	212	201	177	147	143

A brief history of computer vision

- 1960s - started as a student summer project at MIT.
- 1970s and 80s – part of AI – understanding human vision and emulating human perception.
- 1990s – depart from AI , geometric approach.
- Today – various mathematical methods (statistics, differential equations, optimization), applications (security, robotics, graphics).
- Many applications of computer vision are now in common usage

What is Computer Vision?

Trucco & Verri:

Computing properties of the 3-D world from one or more digital images.

Properties: mainly physical (geometric, dynamic, etc.)

One common definition:

Computer vision is inverse optics or inverse graphics.

Related fields

- Image Processing
  - Generates one image from another – i.e. sharpen
- Pattern Recognition
  - Learn patterns for classifying or analyzing data – i.e. character recognition
- Photogrammetry
  - Measures 3d quantities from multiple 2d images – i.e. model building
- Computer graphics
  - Take a virtual scene and render it – i.e. gaming
- Human/Computer Interfaces
  - Ways of interacting between a computer and a human – i.e. kinect for xbox

Our Time

It is a good time to do computer vision now, because:

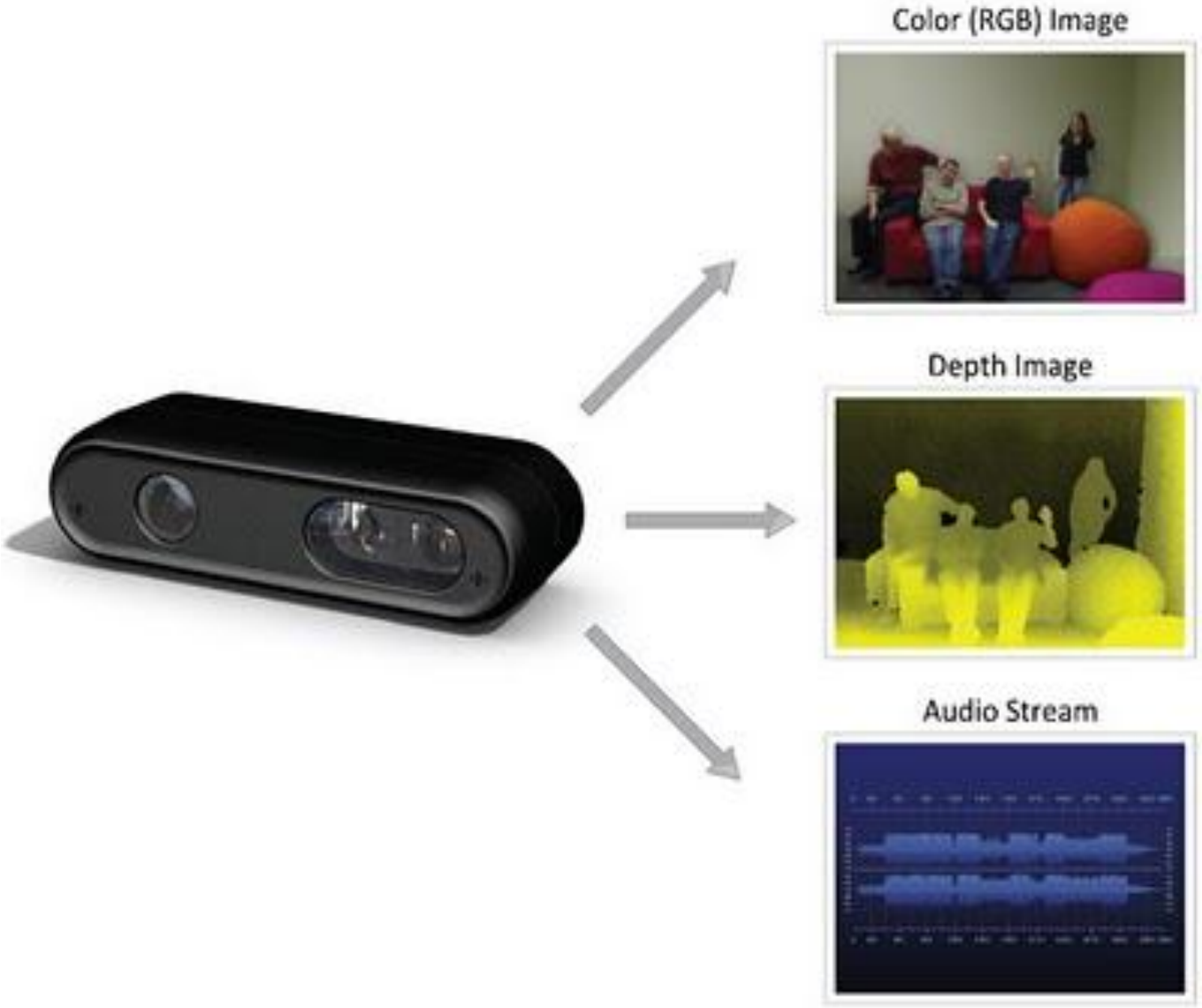
- Powerful computers
- Inexpensive cameras
- Algorithm improvements
- Understanding of vision systems
- Modern computer networks
- Cameras everywhere
  - Including mobile devices
- However one problem!
  - Doing anything reliably is very difficult!

Human Computer Interfaces

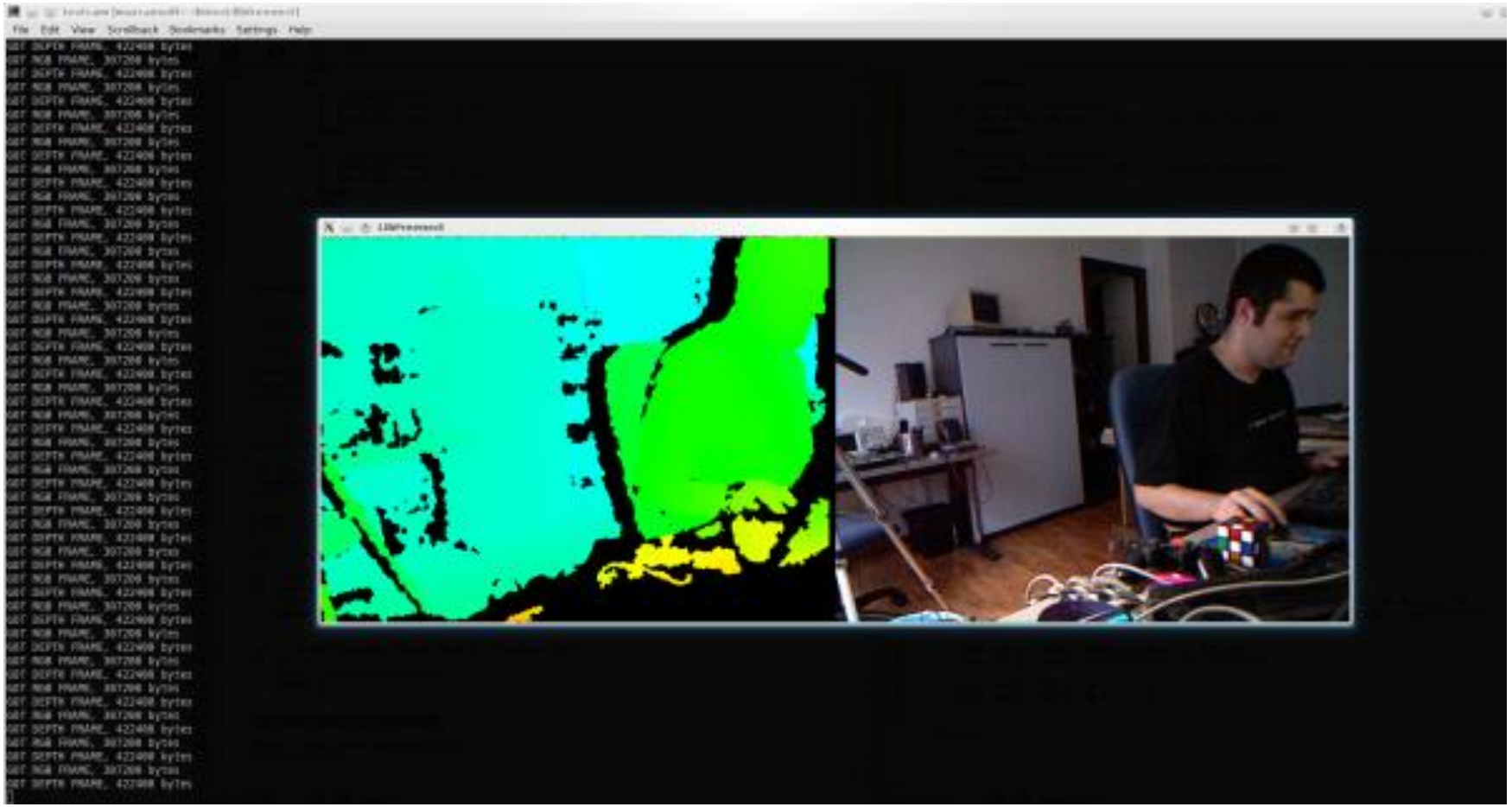
- Microsoft Kinect camera
  - Uses infrared projected patterns
  - Single camera sees these and computes 3d data
  - Basically depth values on top of image pixel values
- 3d depth data is easier to use than 2d
  - At least for applications such as motion detection
  - Can detect motion of a person (including arms/legs)
- Use this motion information to control game
  - A more complex input device than the Wii
  - Wii also uses computer vision technology

Typical structure light system

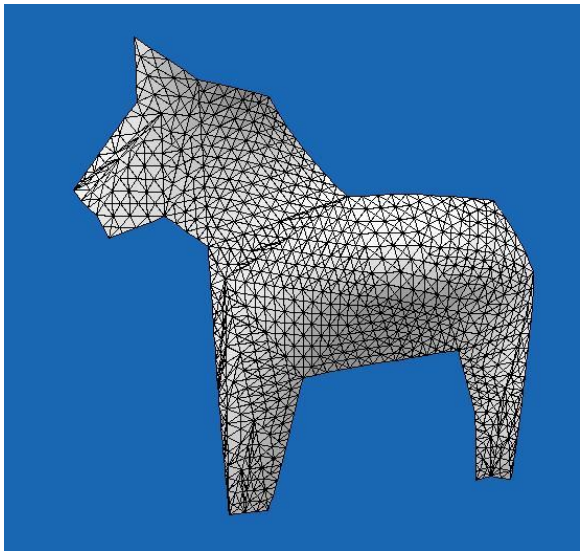
Kinect system



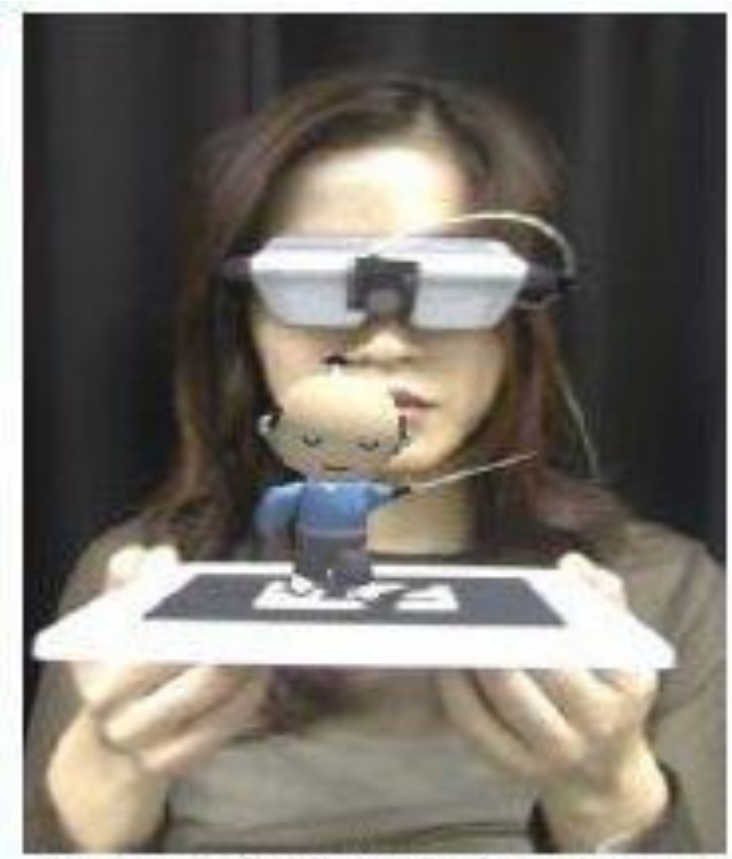
Kinect system



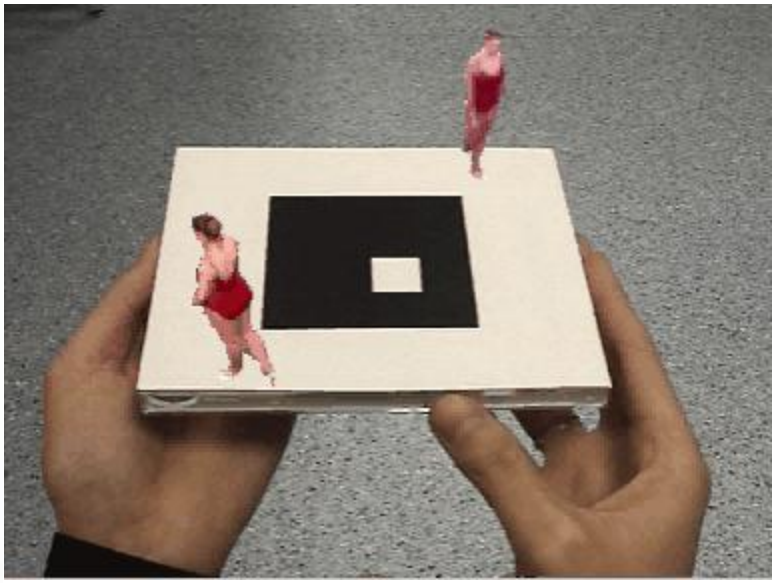
3D Reconstruction



Augmented Reality



Ftg. 2: A Virtual Object on a Card



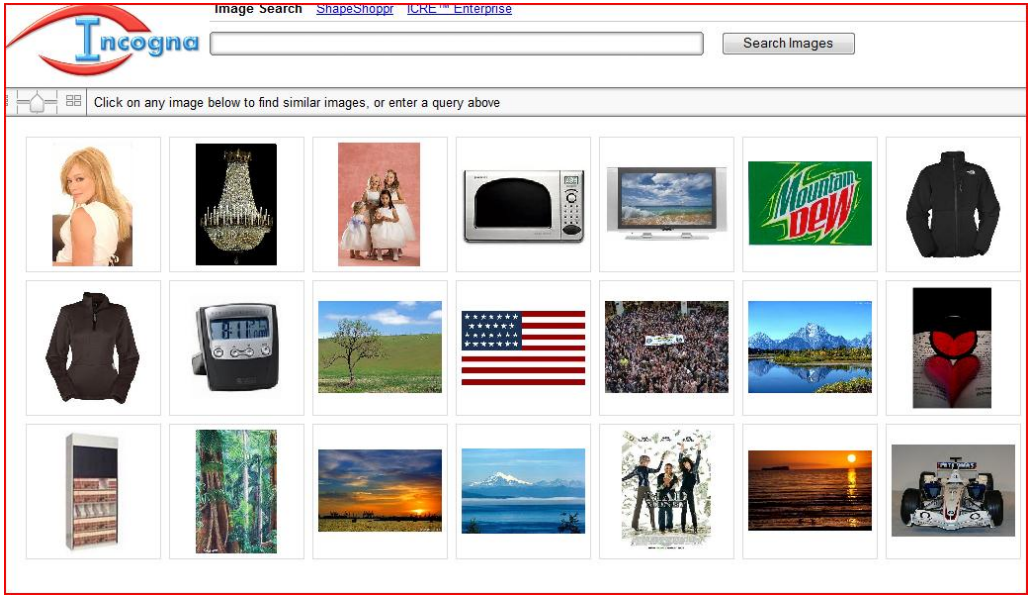
Panoramic Mosaics

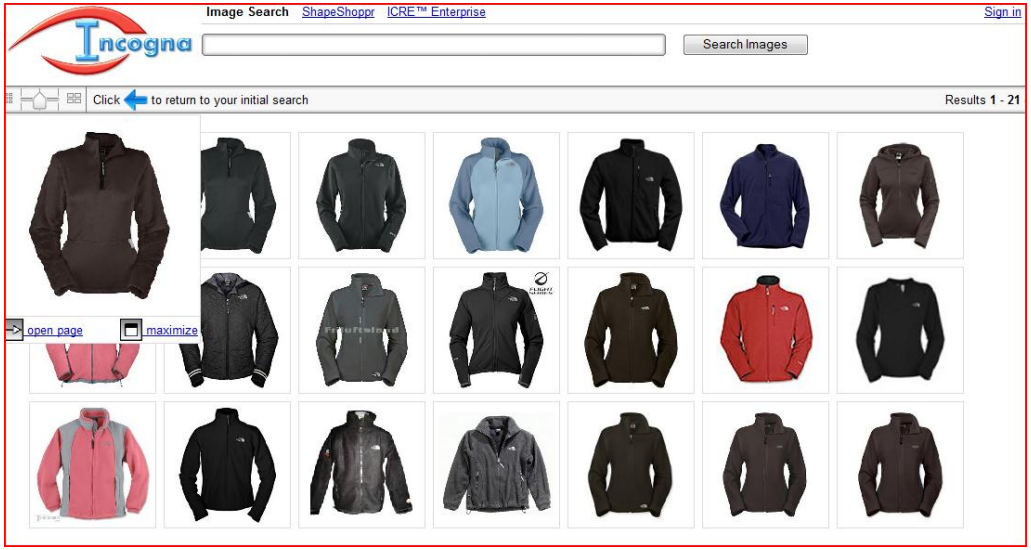

+

+ … +

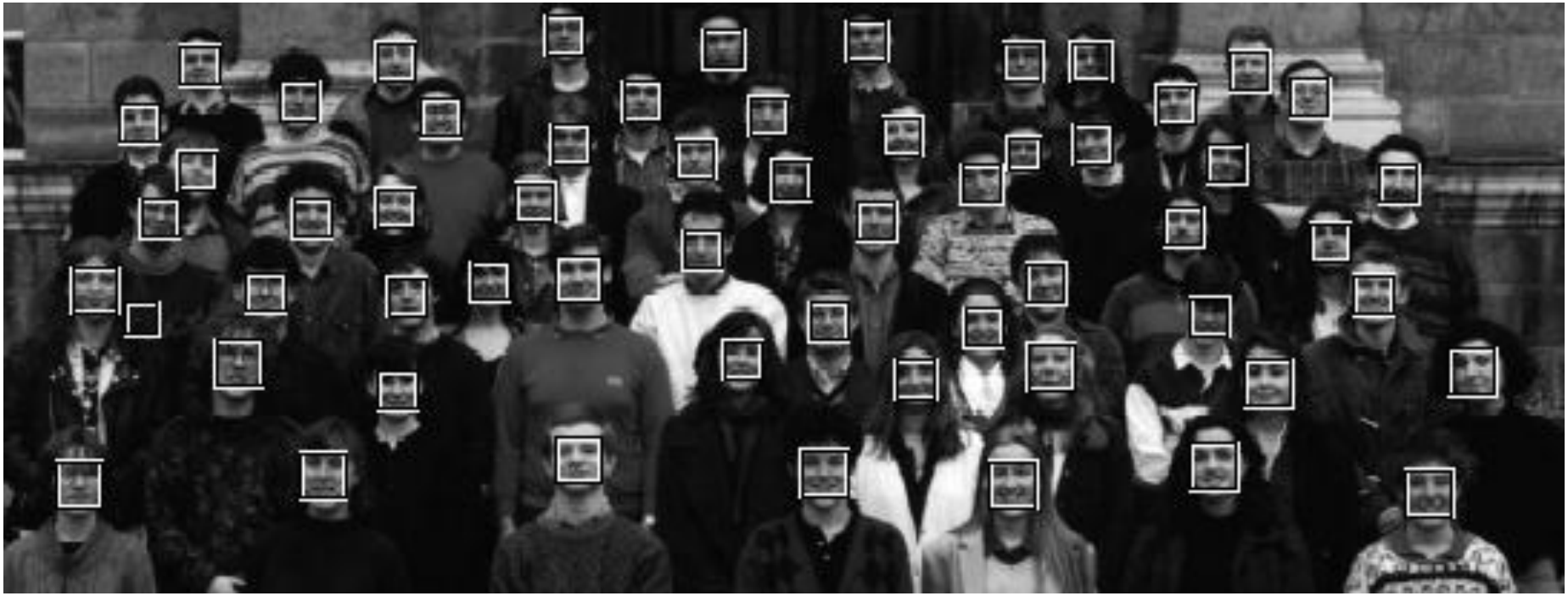
=

Image Search: www.incogna.com





Applications: Recognition



Applications: Special Effects

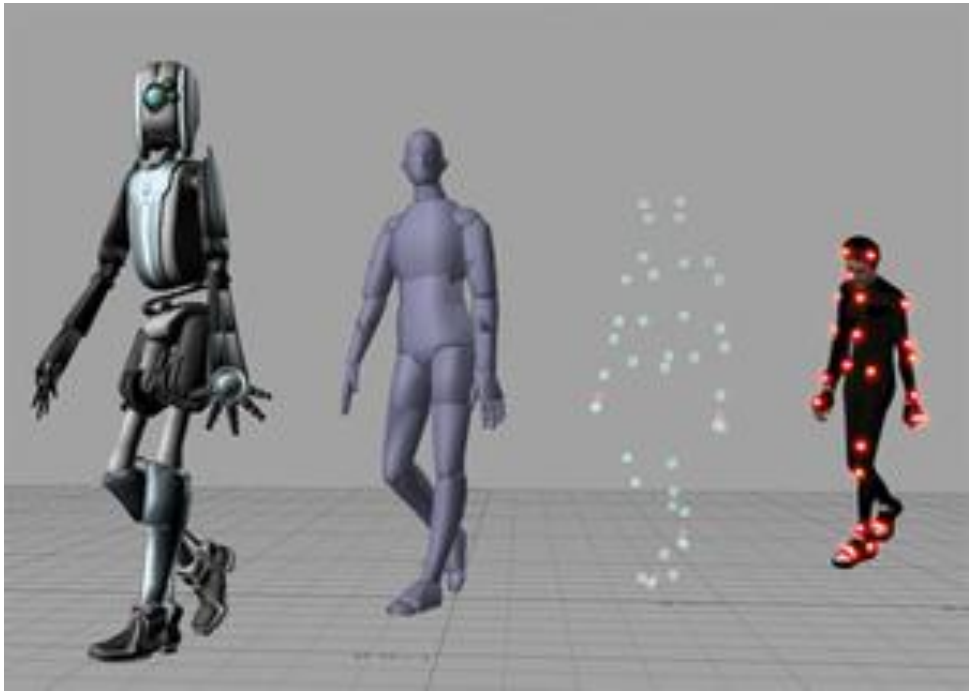
ESC Entertainment, XYZRGB, NRC

Applications: Special Effects

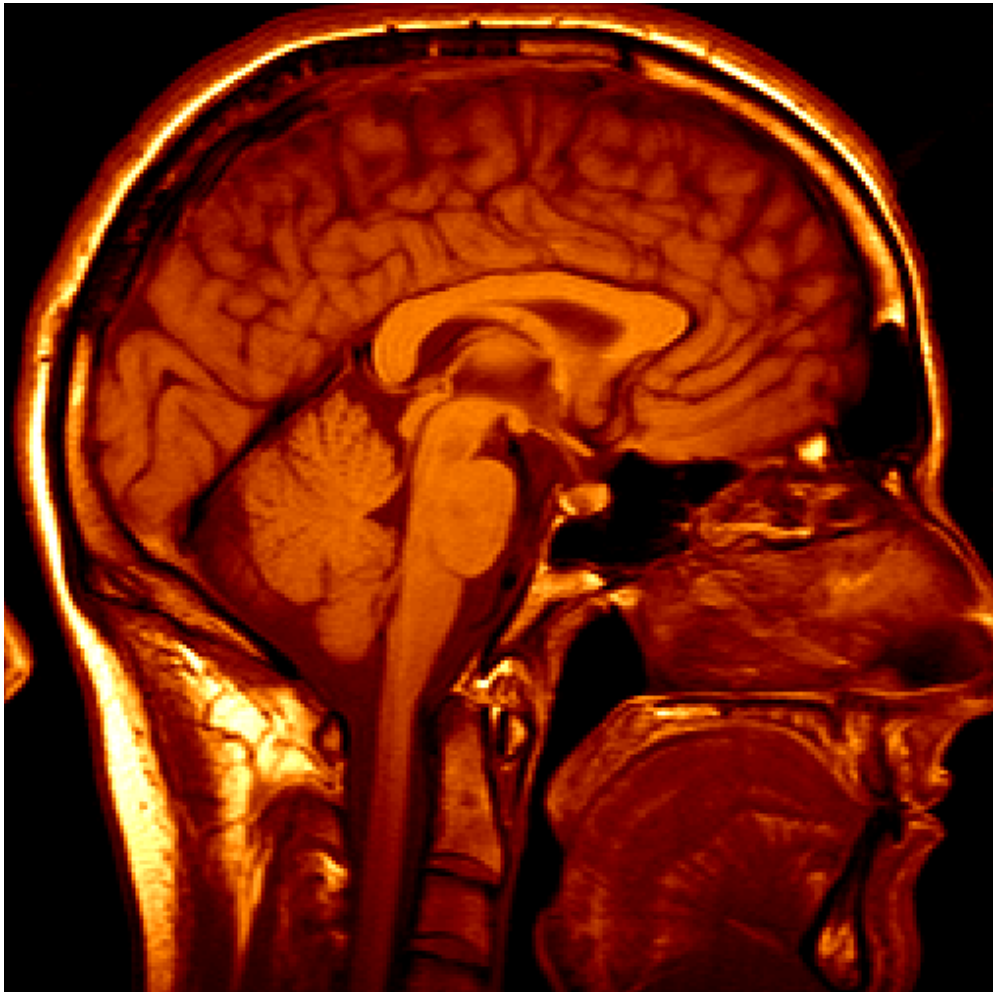
Andy Serkis, Gollum, Lord of the Rings

Motion capture systems

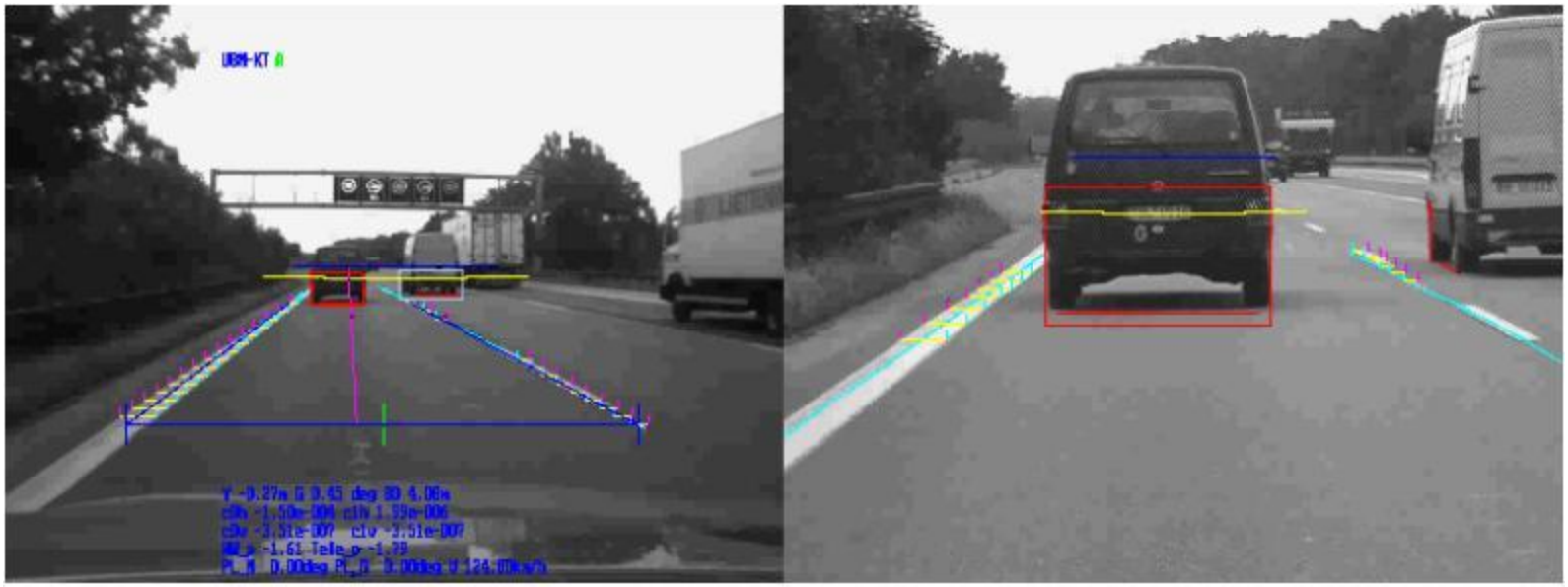
- Put retro-reflective balls everywhere
- Track these in real-time to get 3d positions



Applications: Medical Imaging

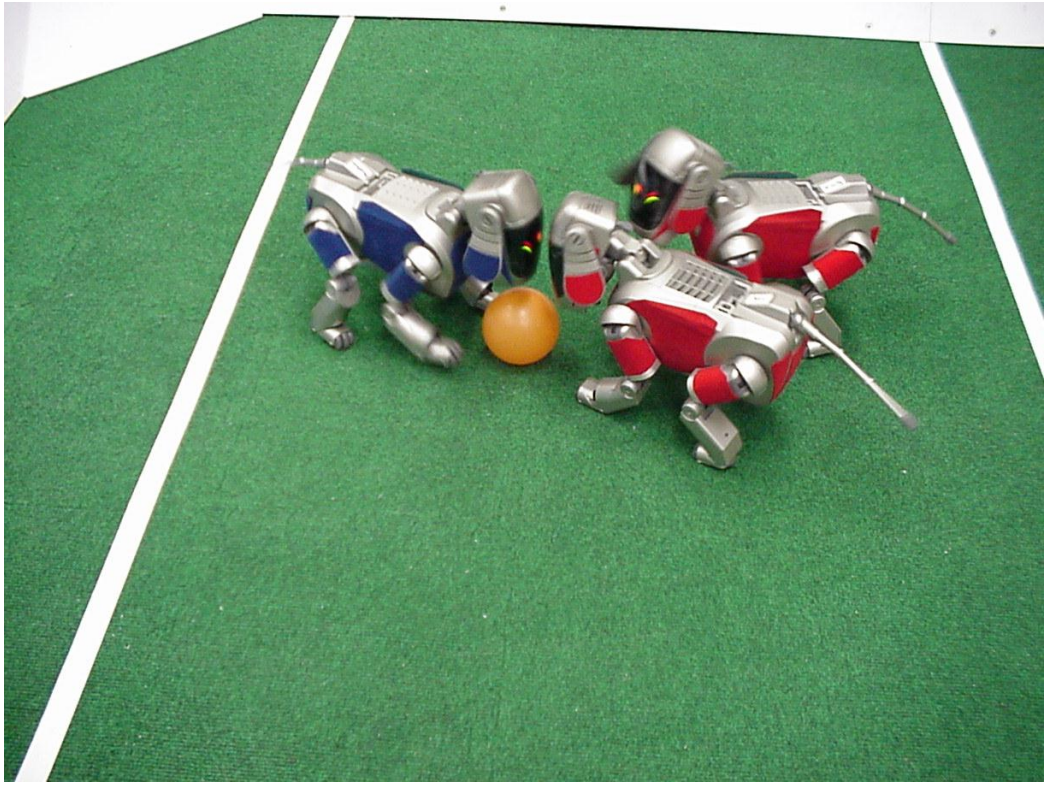


Autonomous Vehicle

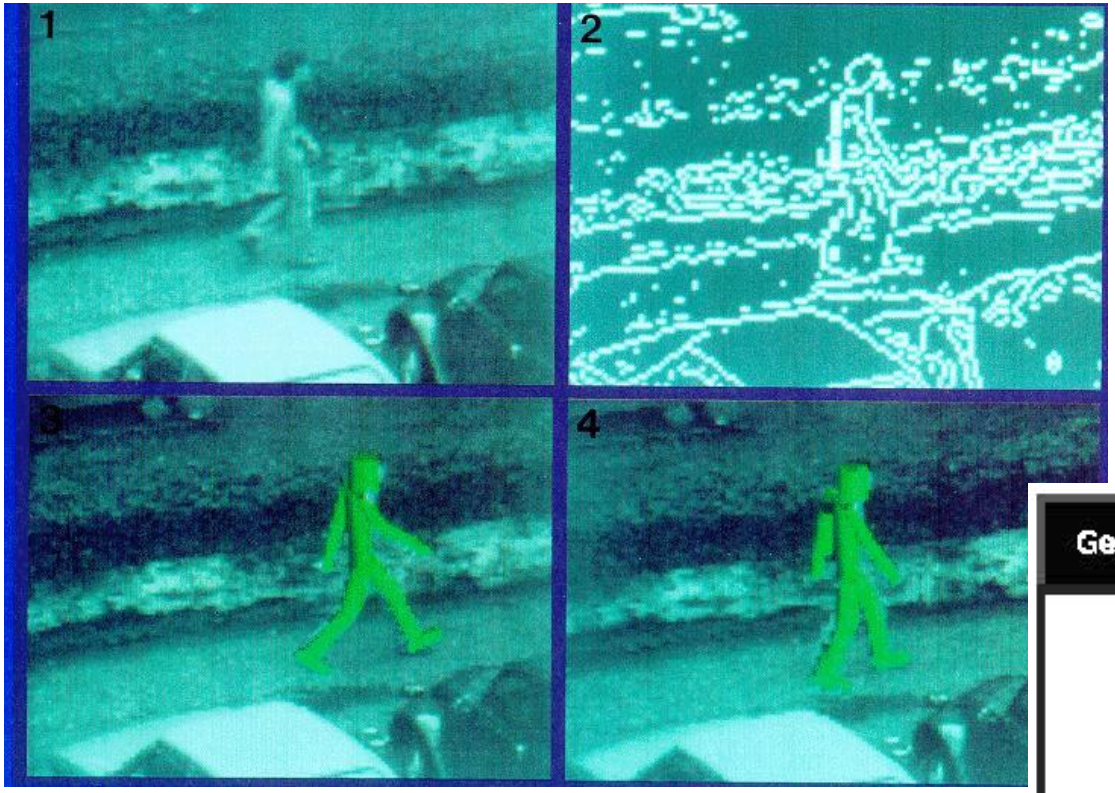


Flakey, SRI

Applications: Robotics



Applications: Surveillance



Mathematical tools

- Linear algebra
- Vector calculus
- Euclidean geometry
- Projective geometry
- Differential geometry
- Differential equations
- Numerical analysis
- Probability and statistics

Programming tools

- OpenCV
  - A universal toolbox for research and development in the field of Computer Vision
  - Widely used OpenSource library written in C/C+
  - Capable of running in real-time applications
  - OS/hardware/window-manager independent
    - Simple Gui environment with sliders/mouse interaction
  - Routines for a large variety of computer vision algorithms
  - Works on Windows, Linux and on Android phone
  - In windows require Visual Studio environment
  - Easiest way to learn is to look at the sample code
  - OpenCV Wiki and on-line documentation
  - A number of different versions (Older version 2.1.0) is easiest but you can use latest version (more complex)

OpenCV Functionality

Image data manipulation
- allocation, release, copying, setting, conversion

Image and video I/O
- file and camera based input, image/video file output

Matrix and vector manipulation, and linear algebra routines
- products, solvers, eigenvalues, SVD

Various dynamic data structures
- lists, queues, sets, trees, graphs

Basic image processing
- filtering, edge detection, corner detection, sampling and interpolation, color conversion, morphological operations, histograms, image pyramids

OpenCV Functionality (*cont.*)

Structural analysis

- connected components, contour processing, distance transform, various moments, template matching, Hough transform, polygonal approximation, line fitting, ellipse fitting, Delaunay triangulation

Camera calibration

- finding and tracking calibration patterns, calibration, fundamental matrix estimation, homography estimation, stereo correspondence

Motion analysis

- optical flow, motion segmentation, tracking

Object recognition

- eigen-methods, HMM

Basic GUI

- display image/video, keyboard and mouse handling, scroll-bars

Image labeling

- line, conic, polygon, text drawing

OpenCV Modules

OpenCV Functionality

- more than 350 algorithms

Cxcore

- Data structures and linear algebra support.

cv

- Main OpenCV functions.

Cvaux

- Auxiliary (experimental) OpenCV functions.

Highgui

- GUI functions.

Course DVD – OpenCV plus lectures

- Contains OpenCV and an example program along with a copy of the course web site
  - But you should look at course web site for current updates
- http://opencv.willowgarage.com/wiki/ has detailed instructions on most recent version
  - But read my notes in the CD to see easiest approach!
- Easiest approach is different
  - Install some version of Visual Studio (Version 8 or later)
  - Install OpenCV (OpenCV-2.1.0-win32-vs2008)
  - In directory OpenCV example open the HarrisCorner project
  - Use debug mode, copy link directory to use release mode
  - To try out different example programs replace HarrisCorner.cpp with the other .cpp and .c files