Lecture 26: Introduction to Computer Vision

December 5, 2019

Real-World Machines

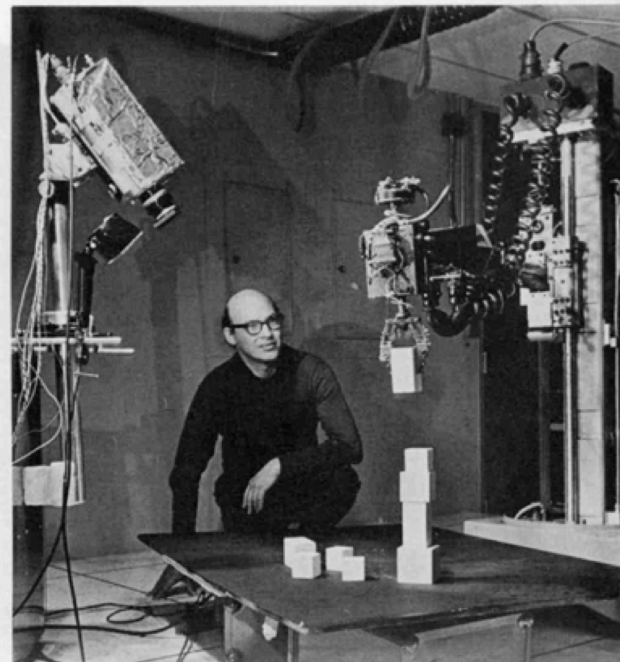
A computer, a television camera, and a mechanical arm have now been combined into a system with enough artificial intelligence to recognize blocks of various sizes and shapes and to assemble them into structures without step-by-step instructions from an operator. The system can perceive the blocks visually, determining their size and their location on a table. It can stack them into a tower while accomplishing another goal, for example, of making the tower as high as possible with the given blocks. Or, it can be told to sort the blocks by size into neat, separate stacks.

Development of this kind of system, which was demonstrated at M.I.T. this spring, is an early stage of research on principles that will give machines engaged in routine tasks greater flexibility through their ability to see their work. Even simple vision would allow a machine to grasp one object without relying on its being absolutely positioned, or to pick up an object it had dropped, or to recognize defects.

Long range goals of work directed by Marvin L. Minsky, Professor of Electrical Engineering, and Seymour A. Papert, Visiting Professor of Applied Mathematics, envisage machines with finer and more varied visual abilities and more manual dexterity than are required for such semi-routine tasks. Work is progressing on binocular vision, color vision, the ability to perceive textures, touch sensors, improved mechanical hands and other areas whose development is necessary for accomplishing significant real-world tasks. Outlining goals such as these, especially the ability to program machines to acquire and use a substantial fund of knowledge about the real world, reveals the extent of scientific and engineering progress toward "artificial intelligence."

For vision, the system demonstrated at M.I.T. this spring uses "image dissector" cameras, controlled by a

A computer, a television camera and a mechanical arm have been combined at M.I.T. into a system with enough artificial intelligence to recognize blocks of various sizes and shapes and to assemble them into structures without step-by-step instructions from an operator. The work is part of research on artificial intelligence being conducted by Professor Marvin L. Minsky (below) and his associates in the Department of Electrical Engineering and Project MAC. In the picture, the camera is at the upper left and the mechanical arm with a vise-like hand is shown holding a block to be stacked. The large-scale PDP-6 computer which is programmed to co-ordinate eye and arm is located elsewhere in the laboratory.



From ***The Tricky Challenge of Making Machines That "See"*** in the MIT Technology Review. Article above originally published in 1968.

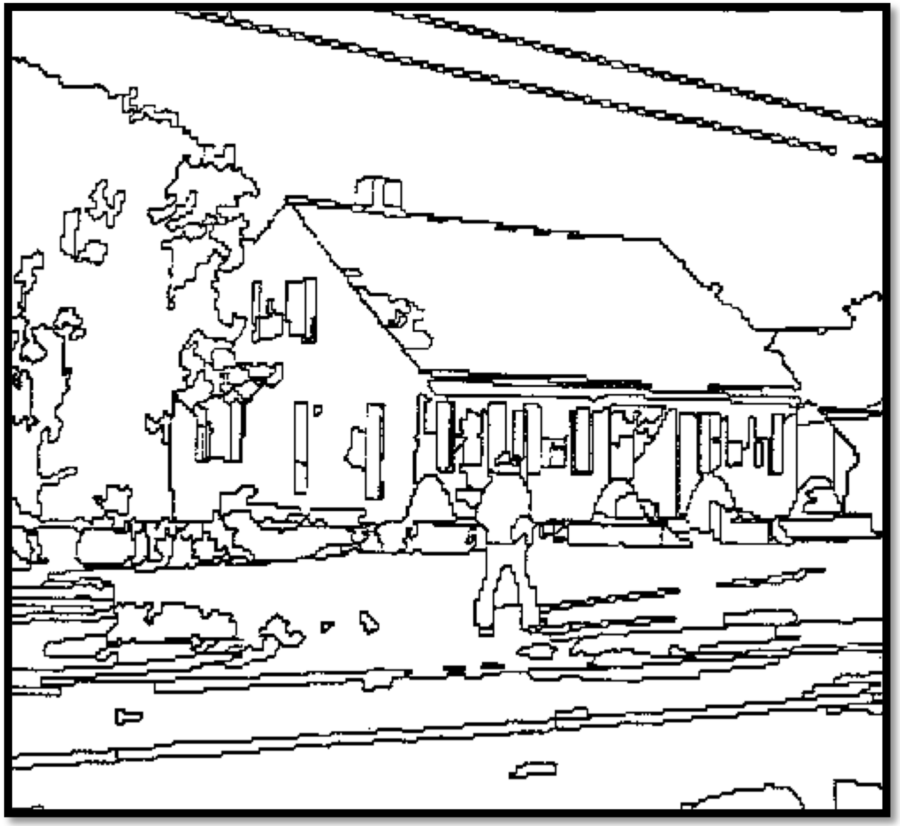
A Bit Personal

International Journal of Computer Vision
January 1989, Volume 2, Issue 3, pp 311–347 | Cite as

Segmenting images using localized histograms and region merging

Authors | Authors and affiliations

J. Ross Beveridge, Joey Griffith, Ralf R. Kohler, Allen R. Hanson, Edward M. Riseman



And from Bruce Draper

Published: January 1989

The schema system

Bruce A. Draper, Robert T. Collins, John Brolio, Allen R. Hanson & Edward M. Riseman

International Journal of Computer Vision **2**, 209–250(1989) | Cite this article

CSU History (Very Partial)

- Image Understanding
- 3D Model Based Object Recognition
- Satellite Reconnaissance
- Object Recognition in General
- Face Recognition
- Video Understanding
- Gesture Recognition
- Communicating with Computers

CSU Software

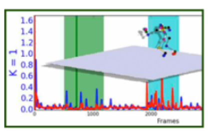
CSU Computer Vision Group Software Downloads

Here are major software downloades provided by the group in approximate chronological order. Annotation with each will provide some guideance about which we consider to be current and relevant versus those that are still provided but for more archival purposes.

A visualization of RNNs in Skeleton based Action Recognition (SkeletonVis)

The visualization of RNNs provided by the Skeleton based Action Recognition Toolkit provides insights into models embedded in Recurrent Neural Networks in the domain of skeleton based Action Recognition. Using two primary methods for visualizing the properties learnt by a trained LSTM Network, namely, Sensitivity analysis and Activation Maximization, we present case studies on datasets commonly used for Action Recognition. Additionally, users can upload their own models and visualize their trained models.

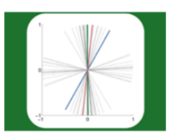
Last updated August 30, 2019.



Generalized Curvature Analysis Toolkit (GeCAT)

The Generalized Curvature Analysis Toolkit (GeCAT) is useful for estimating generalized curvatures of curves lying in an n-dimensional space. As GeCAT calculates the curvatures using a curve localized SVD approach, it is a robust alternative to numerical differentiation for estimating curvature in high dimensions and works well even in the presence of noise. GeCAT segments pose streams into motions without knowing the set of motions in advance.

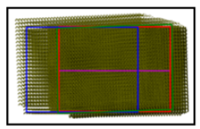
Last updated May 30 2018.



Subspace Mean and Median Evaluation Toolkit (SuMMET)

The Subspace Mean and Median Evaluation Toolkit (SuMMET) is a Matlab-based software package. There are two parts to the download. The first is the core Matlab code used to compute the Karcher mean, the L2-median, the extrinsic manifold mean, and the flag mean for collections of linear subspaces. The second part includes datasets and associated Matlab scripts that will reproduce results from the CVPR 2014 paper.

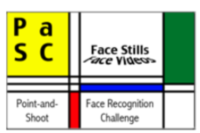
Last updated June 24 2014.



Robust Staged RANSAC Tracking

The Robust Staged RANSAC Tracking algorithm is useful for stabilizing video from a handheld camera. It is particularly well suited to the handheld video in the Point-and-Shoot Face Recognition Challenge where there is distinct camera motion following a person and yet for much of the video much of the scene background remains in view. The goal of this algorithm is to generate a new video where the background becomes fixed as though the camera had been locked down on a tripod.

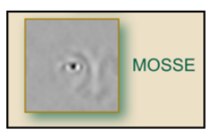
Last updated February 28 2014.



PaSC Software Support Package

CSU distributes a package of software for working with the Point-and-Shoot Face Recognition Challenge. This software comes in two forms, a completely self contained virtual machine and source code. Both distributions include meta-data useful for working with PaSC. They also include everything needed to run the CSU provided baseline algorithms and to generate ROC curves.

Last updated June 11 2013.



Optimized Correlation Output Filters Toolset (OCOFTools)

The Optimized Correlation Output Filters Toolset (OCOFTools) is a software package offered for those intersted in experimenting with David Bolme's Optimized Correlation Output Filters. This work is summarized in David Bolme's Dissertation as well as CVPR papers from 2009 and 2010. Be aware that Colorado State University has a patent covering the filter construction technique: the code is available for non-commercial research and education purposes only. License details are availble through the download page.

Last updated October 10 2012.

The 2011/2012 Face Recognition Baselines

This release was prepared to support the Good, Bad and Ugly (GBU) Challenge problem. More information about the GBU Challenge may be found in An Introduction to the Good, the Bad, and the Ugly Face Recognition Challenge Problem and at the NIST Site. In terms of what is most current, the PaSC is a more recent and better challenge problem for most purposes. However, GBU remains a challenging face recognition problem with amble opportunity for the community to benchmark improvement.

Last updated August 7 2012.

FaceL

FaceL is a simple turnkey demonstration of live face recognition over video. It is constructed to work with video feeds from laptop cameras and webcams. It is designed to be easily trained for up to about 6 people and serve as a demonstration of technology largely for educational purposes. Unfortunately, the turnkey version form Mac OS is not compatible with mopre recent releases of OS X. With appropriate knowledge of Mac OS or Windows FaceL can still be installed from the source distribution and run successfully. However, as it is based on older versions of Python and OpenCV it should not be viewed as trivial to install.

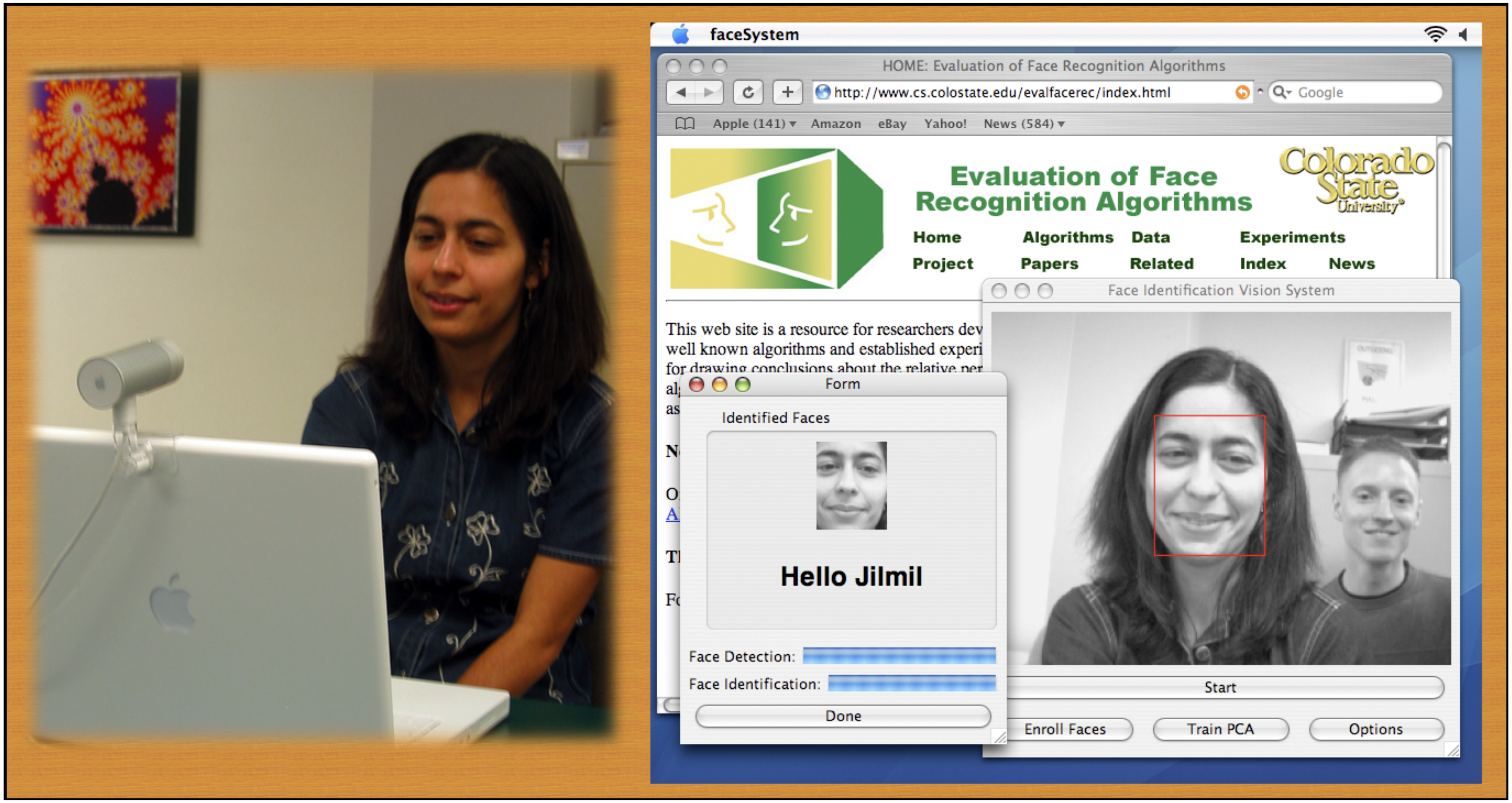
Last updated September 21, 2009.

Evaluation of Face Recognition Algorithms

This system was our first major effort at releasing a turnkey series of face recognition algorithms along with evaluation support softwared. As of now, 2014, it is eleven years old and still being downloaded by some. As an educational tool we believe it can still be of value. The associated information concerning evaluation of algorithm using the original FERET protocol may be of use to anyone wishing to replicate much earlier experiments on arguably the first major face recognition challenge problem. The algorithms themselves in this package are dated and should not be taken as meaningful benchmarks of modern algorithm performance.

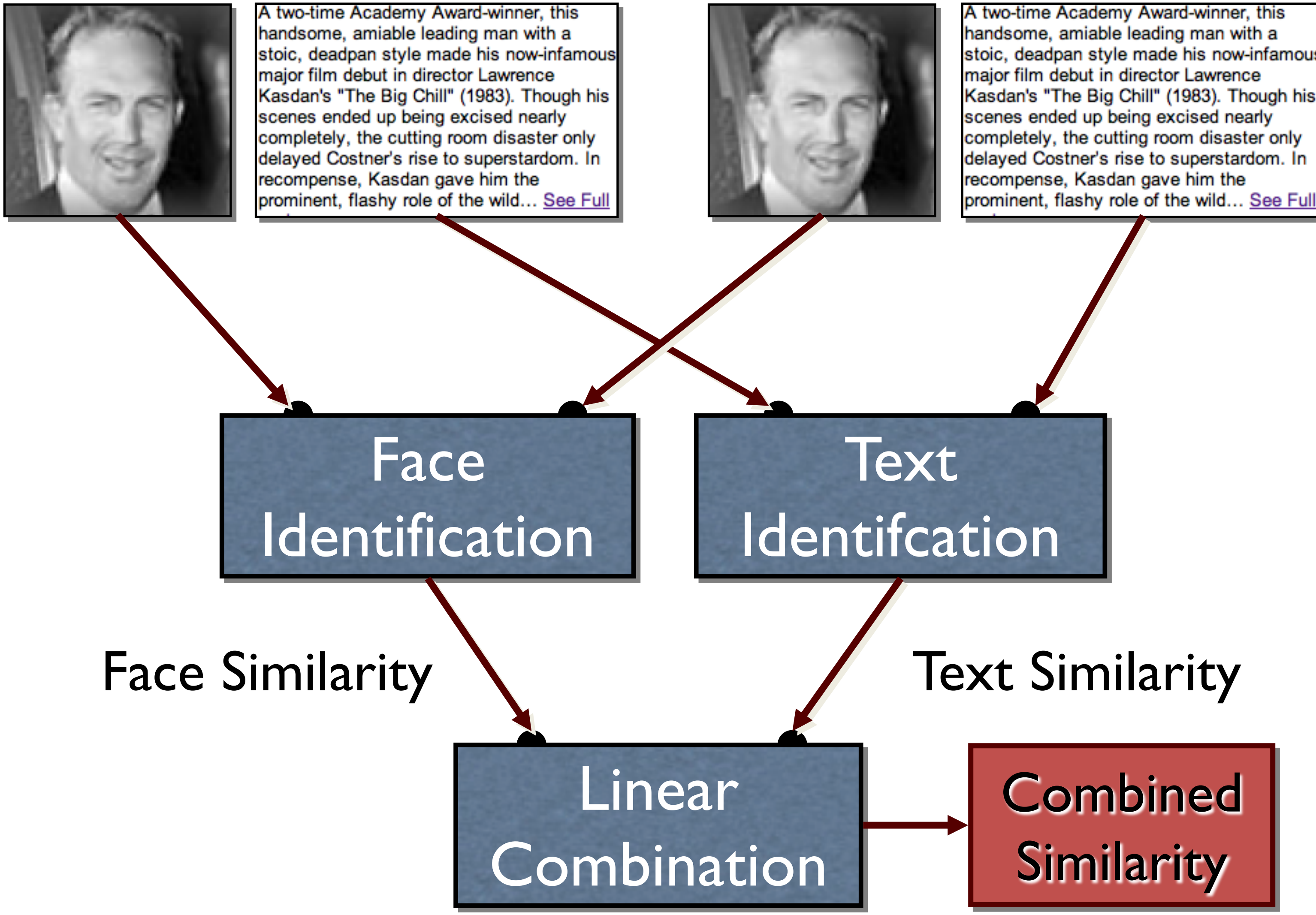
Last significant update May 2003.

Some Samples of Past Work



Face Recognition in real-time at CSU around 2006

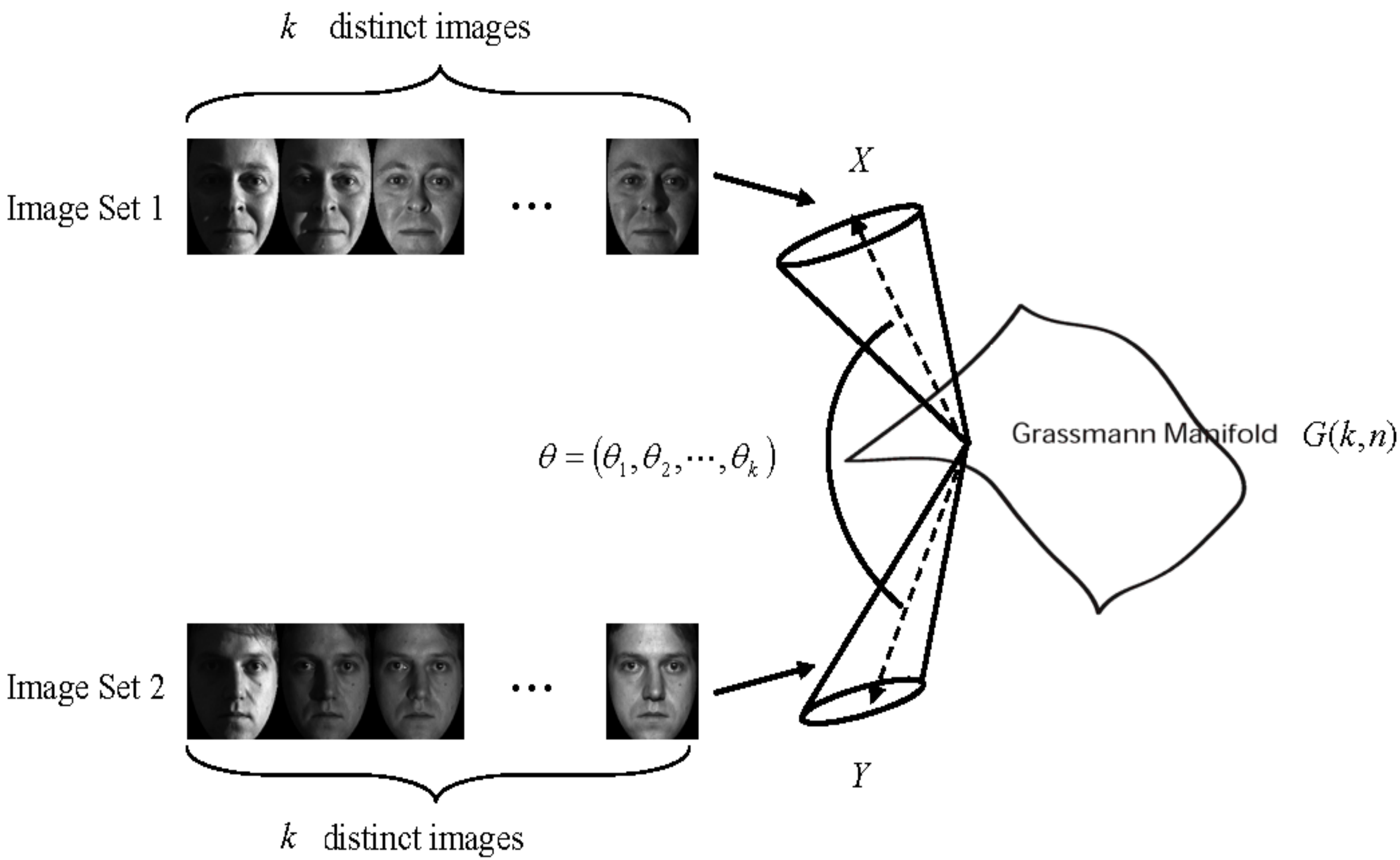
Text and Face Images (~2006)



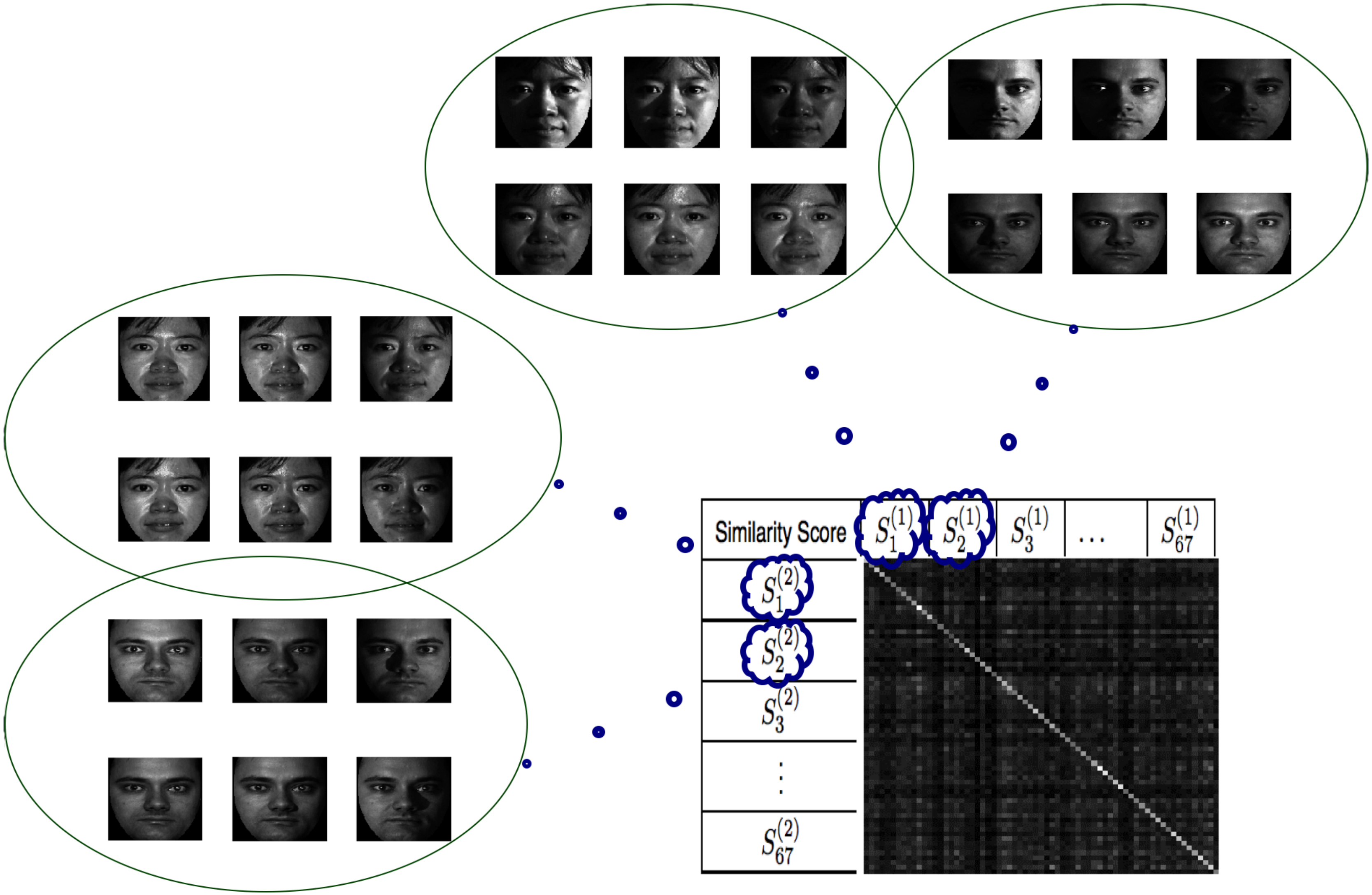
Faces & Illumination Part 1

J. Ross Beveridge, Bruce A. Draper, Jen-Mei Chang, Michael Kirby, Holger Kley, Chris Peterson, “Prin- cipal Angles Separate Subject Illumination Spaces in YDB and CMU-PIE”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 2, pp. 351-363, February, 2009.

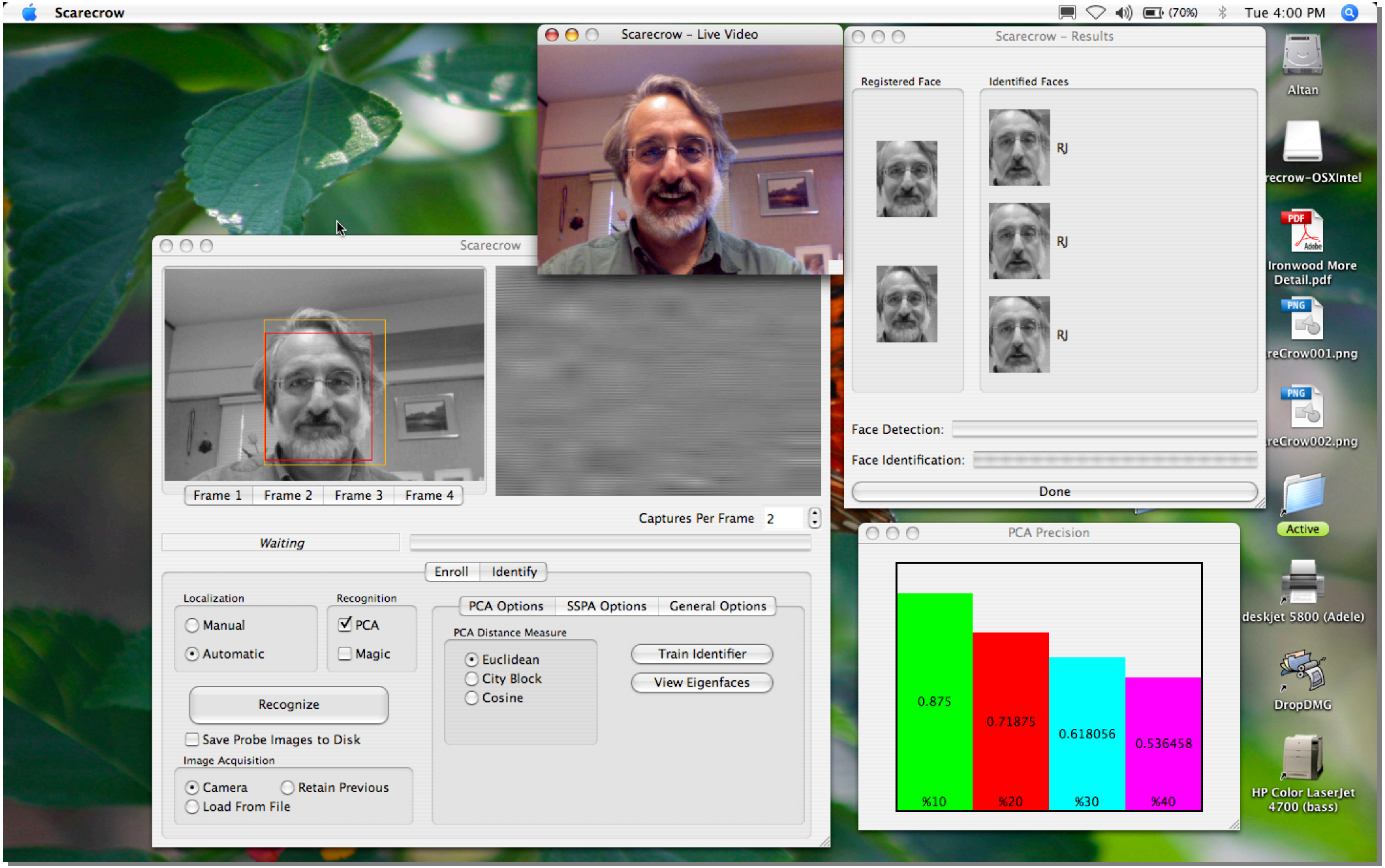
Faces & Illumination Part 2



Faces & Illumination Part 3



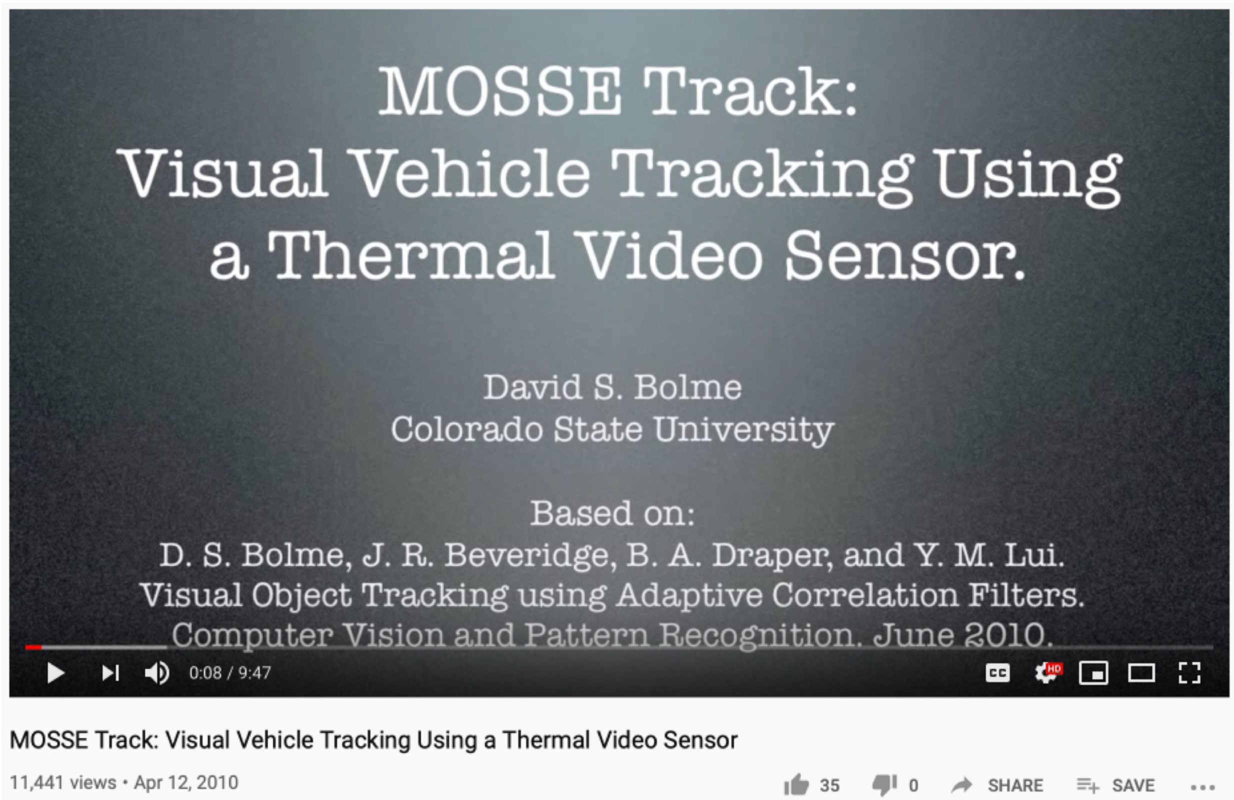
More Interactive Play



FaceL


CSU FaceL
FaceL
Facile Face Labeling
Controls
Click a face in the video to enroll.
Labels: 0
Faces: 0
Enrollment Count: 64
Train Labeler
Clear Labels
Colorado State University

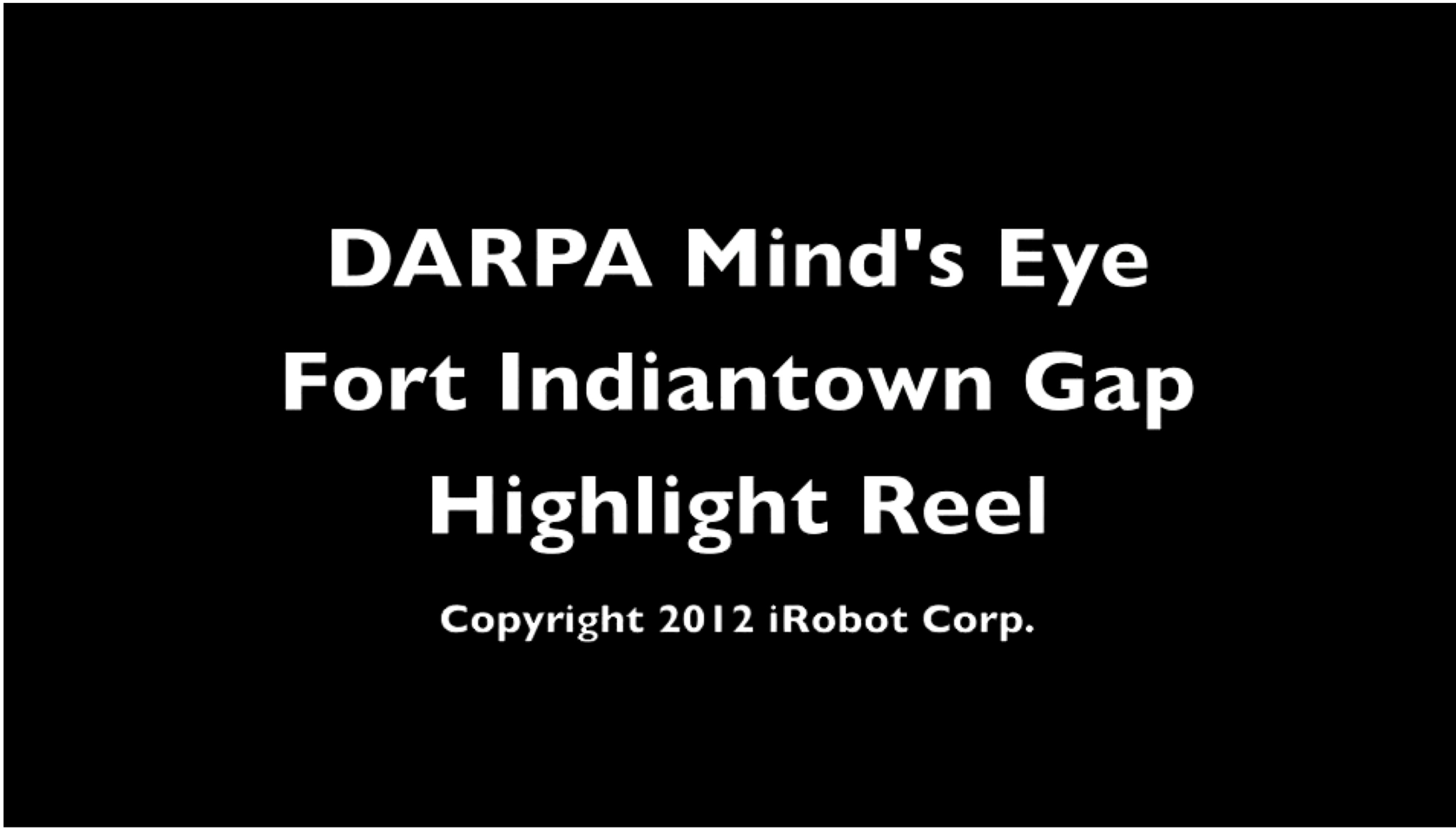
How to Start a Trend



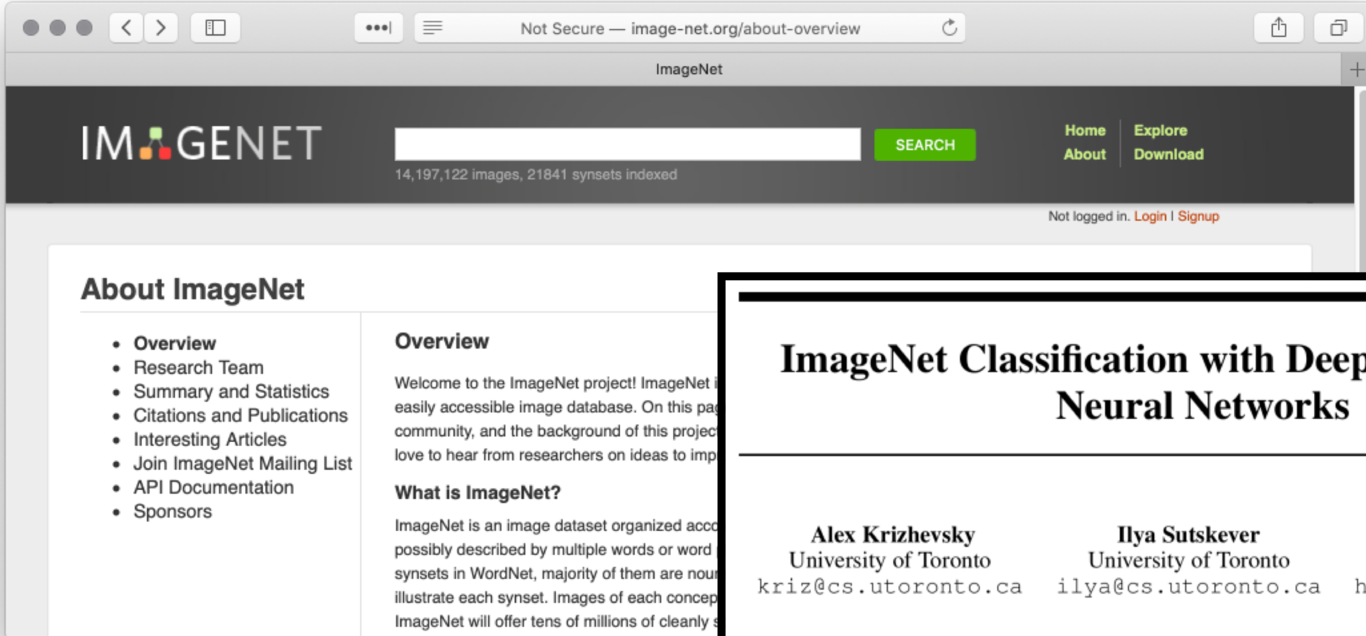
FOLLOW

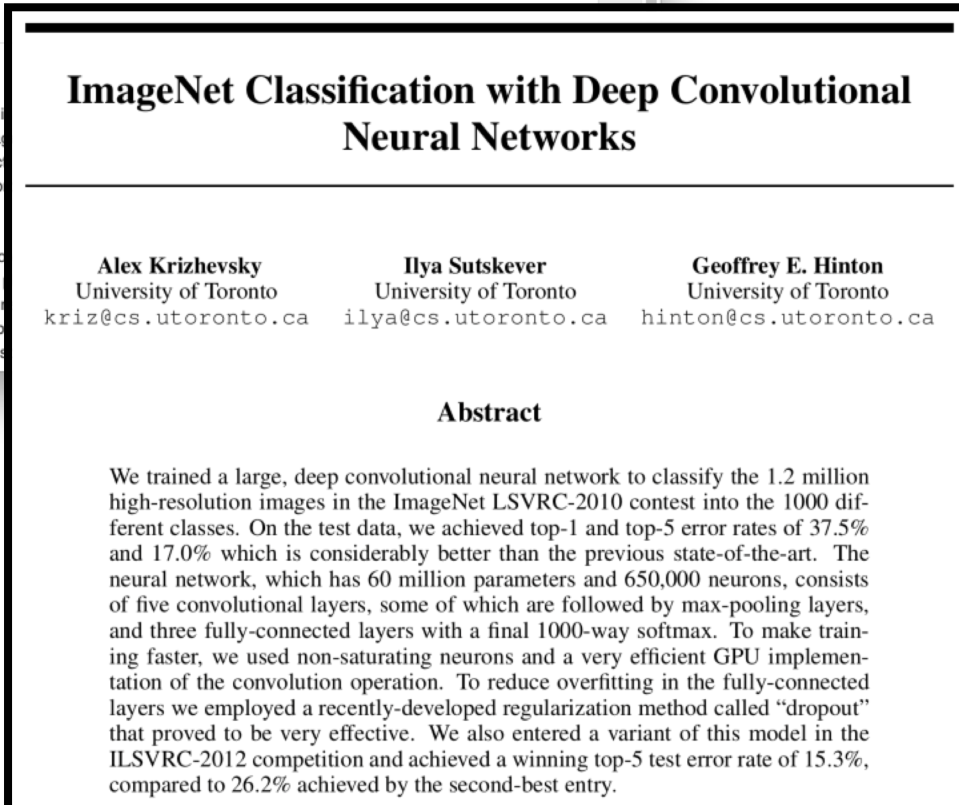
TITLE	CITED BY	YEAR
Visual object tracking using adaptive correlation filters DS Bolme, JR Beveridge, BA Draper, YM Lui 2010 IEEE Computer Society Conference on Computer Vision and Pattern …	1622	2010
Recognizing faces with PCA and ICA BA Draper, K Baek, MS Bartlett, JR Beveridge Computer vision and image understanding 91 (1-2), 115-137	737	2003

Action Recognition



… and then came CNNs





ImageNet Classification with Deep Convolutional Neural Networks

Alex Krizhevsky
University of Toronto
kriz@cs.utoronto.ca

Ilya Sutskever
University of Toronto
ilya@cs.utoronto.ca

Geoffrey E. Hinton
University of Toronto
hinton@cs.utoronto.ca

Abstract

We trained a large, deep convolutional neural network to classify the 1.2 million high-resolution images in the ImageNet LSVRC-2010 contest into the 1000 different classes. On the test data, we achieved top-1 and top-5 error rates of 37.5% and 17.0% which is considerably better than the previous state-of-the-art. The neural network, which has 60 million parameters and 650,000 neurons, consists of five convolutional layers, some of which are followed by max-pooling layers, and three fully-connected layers with a final 1000-way softmax. To make training faster, we used non-saturating neurons and a very efficient GPU implementation of the convolution operation. To reduce overfitting in the fully-connected layers we employed a recently-developed regularization method called "dropout" that proved to be very effective. We also entered a variant of this model in the ILSVRC-2012 competition and achieved a winning top-5 test error rate of 15.3%, compared to 26.2% achieved by the second-best entry.

Communicating with Computers

User-Aware Shared Perception for Embodied Agents

David G. McNeely-White
Francisco R. Ortega
J. Ross Beveridge
Bruce A. Draper
Rahul Bangar
Dhruva Patil
Department of Computer Science
Colorado State University
Fort Collins, Colorado 80523
Email: Ross.Beveridge@colostate.edu

James Pustejovsky
Nikhil Krishnaswamy
Kyeongmin Rim
Department of Computer Science
Brandeis University
Waltham, Massachusetts 02453
Email: jamesp@cs.brandeis.edu

Jaime Ruiz
Isaac Wang
Department of Computer & Information
Science & Engineering
University of Florida
Gainesville, FL 32611
Email: jaime.ruiz@ufl.edu

Hello

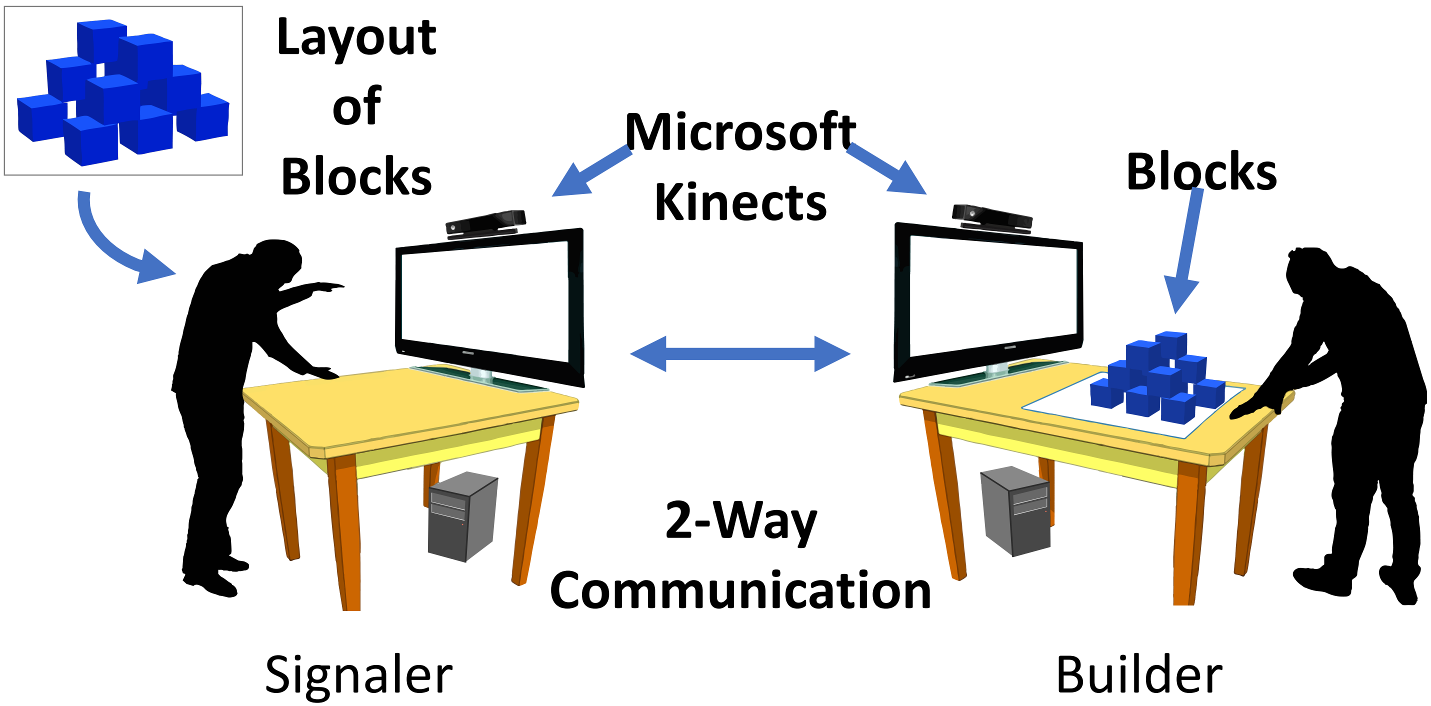
Diana is an embodied agent who can hear, speak and see.

Our Goal

- Better communication when …
    - A person and computer are focused upon a physical task
    - Tasks are AI classics: e.g. Blocks World, Set a Table
- Why is this interesting
- Our Tasks stand in for most everyday tasks where …
    - There is shared perception of a common setting
    - Verbal communication is grounded by context
    - There is shared awareness of body/embodiment
    - Communication is grounded by seeing each other

How We Started

- Elicitation studies between two people solving blocks world tasks.



What We Learned … Examples

RA:

move, front;

RH:

into point, down

RA:

move, up;

RH:

into thumbs, up

Body:

move, back;

What We Learned … Gestures

24,503 labeled instances

8:08:02 of video

~550 hours of labeling effort

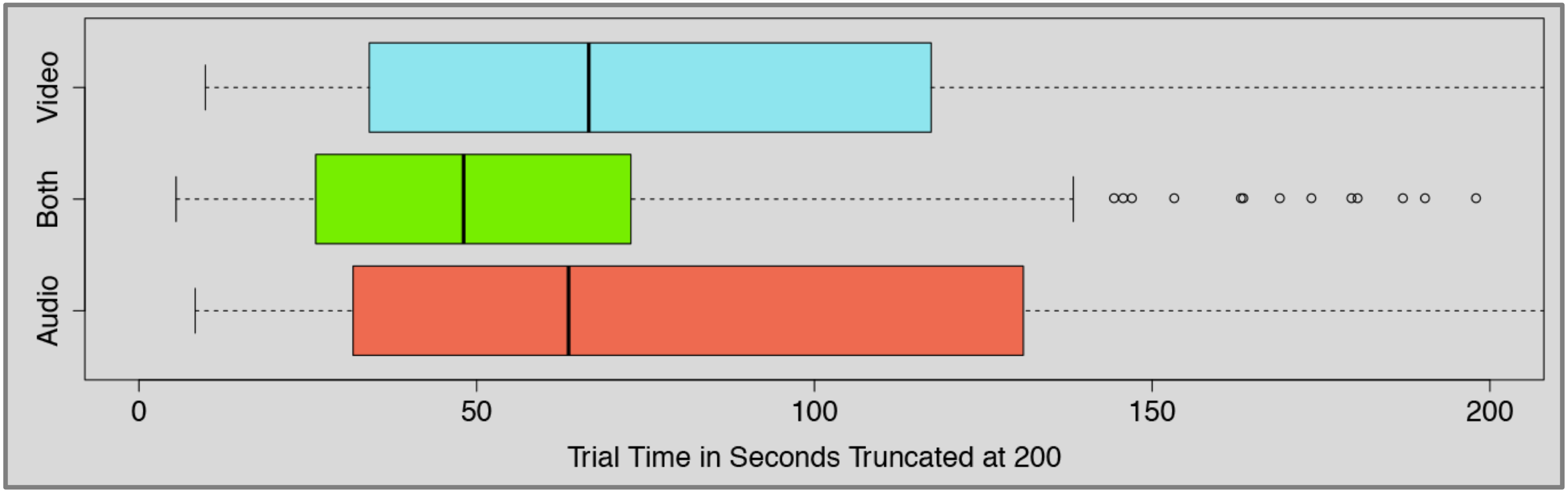
5,060 unique physical movements

110 Gestures occur 20 or more times and from multiple people.

Our current system uses primitive 32 hand poses
Combined with arm and body motion, there are 31 distinct non-verbal communicative actions.

Speech, Gesture, or Both

Gestures alone are roughly as effective as words alone in blocks world: a physical and cooperative task.



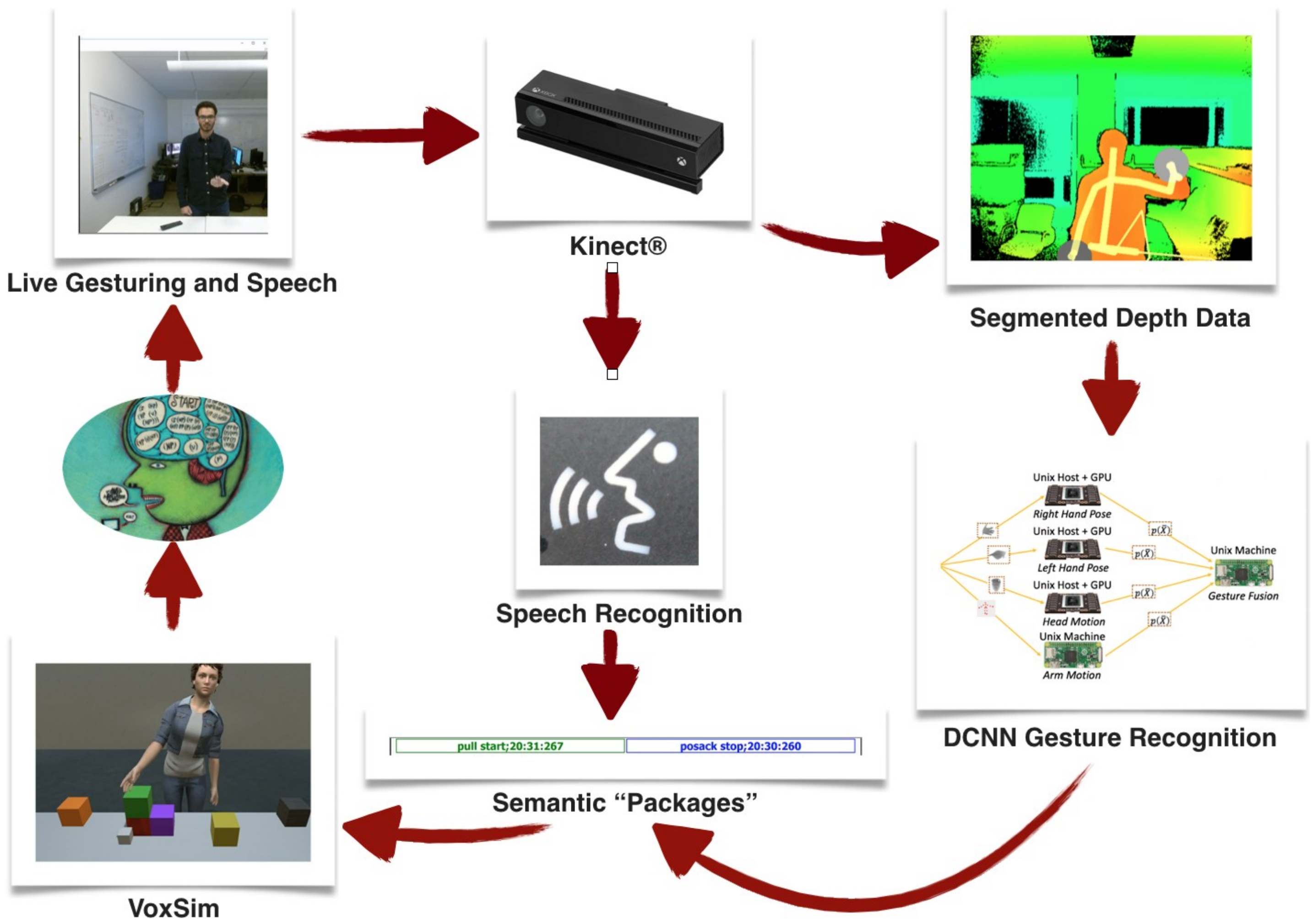
Gestures and words used together are more effective than either used alone.

What ~~We~~ Diana Can Do

Recap What you Just Saw

- Diana sees and hears
  - She understands the user's speech and is conversant
  - She understands the user's non-verbal communication
  - She integrates verbal and non-verbal communication
  - Communication is grounded in shared perception and a task
- In just the first few seconds
  - "Hello Nikhil, I am ready to go"
    - Diana saw Nikhil approach, she waves in greeting
    - Diana focuses her attention, her gaze, on Nikhil
    - Thumbs up gesture while saying "ready to go"
- ...
- And one minute forty seconds later
  - Diana and Nikhil have together built a staircase

How ~~Does Diana~~ Do We Do It?



Just the Gesture Side

- CNNs trained on Kinect depth images recognize 31 distinct hand poses
- LSTMS trained on Kinect skeleton data recognize 8 distinct arm motions
- Training is derived from our 8 hours of labeled data
  - The EGGNOG (Elicited Giant Gallery of Naturally Occurring Gestures) dataset is publicly available
- Typically neural networks require multiple GPUs
  - But a very high-end laptop with GPUs now supports the full system
- Gesture label data as it is recognized gets passed along to the cognitive agent, i.e. Diana

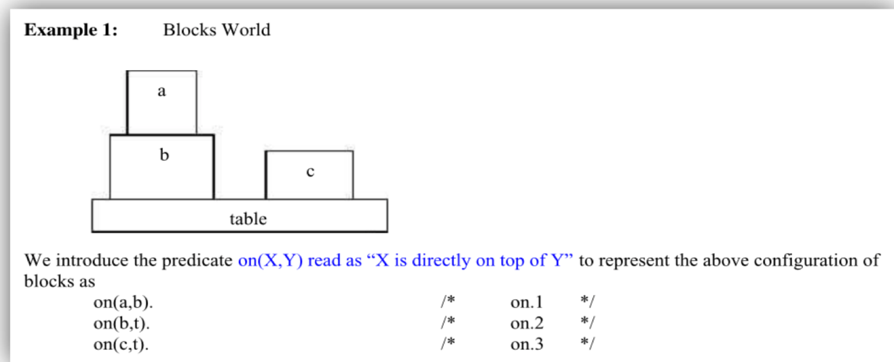
The Cognitive Agent, aka Diana

Key Idea: Grounded Language

Risking gross oversimplification, two approaches

Traditional

Final 'truth' lies in predicates.



Example 1: Blocks World

We introduce the predicate on(X,Y) read as "X is directly on top of Y" to represent the above configuration of blocks as

on(a,b).	/*	on.1	*/
on(b,t).	/*	on.2	*/
on(c,t).	/*	on.3	*/

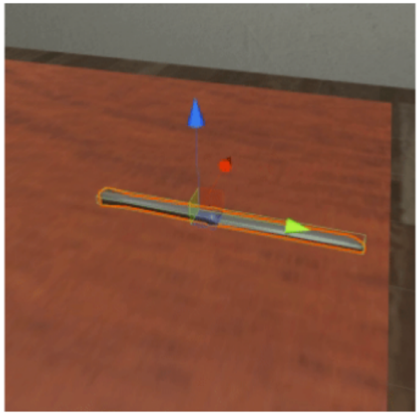
*

Blocks world actually used to illustrate value of logic based representation, and ONLY predicates

Brandeis

Final 'truth' ties directly into physical simulation

$$\left[\begin{array}{l}\textbf{knife}\\ \text{LEX} = \left[\begin{array}{l}\text{PRED} = \textbf{knife}\\ \text{TYPE} = \textbf{physobj, artifact}\end{array}\right]\\ \text{TYPE} = \left[\begin{array}{l}\text{HEAD} = \textbf{rectangular\_prism[1]}\\ \text{COMPONENTS} = \textbf{handle[2], blade}\\ \text{CONCAVITY} = \textbf{flat}\\ \text{ROTATSYM} = nil\\ \text{REFLECTSYM} = \{XY\}\end{array}\right]\\ \text{HABITAT} = \left[\begin{array}{l}\text{INTR} = {}_{[3]}\left[\begin{array}{l}\text{CONSTR} = \{X > Y, X \gg Z\}\\ \text{FRONT} = front(+X)\end{array}\right]\\ \text{EXTR} = \ldots\end{array}\right]\\ \text{AFFORD\_STR} = \left[\begin{array}{l}A_1 = H_{[3]} \rightarrow [grasp(x,[1])]\\ A_2 = H_{[3]} \rightarrow [grasp(x,[2]) \rightarrow grasp(x,[1])]\end{array}\right]\\ \text{EMBODIMENT} = \left[\begin{array}{l}\text{SCALE} = \textbf{<agent}\\ \text{MOVABLE} = \textbf{true}\end{array}\right]\end{array}\right]$$



Concepts grounded in physical modeling, e.g. The Unity Engine

VoxML

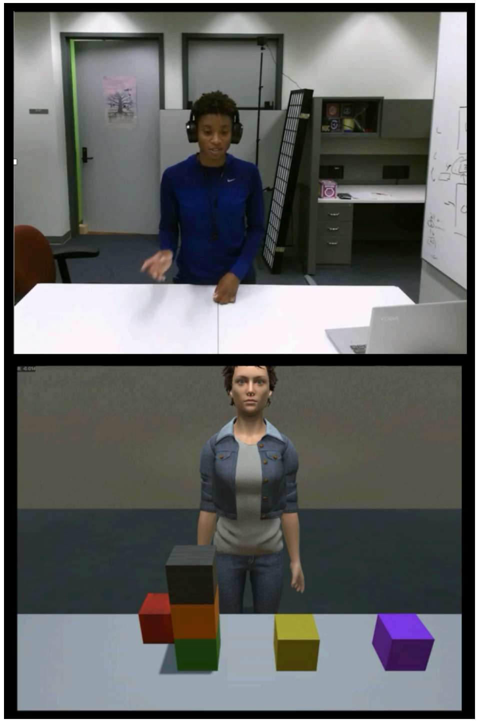
* Example from: https://www.scribd.com/document/326487422/prolog

The Embodied Avatar Conceit

There is now a growing conceit about AI agents. Interact with your agent as you would with a person. We have taken a step down this road in terms of an agent that can see, speak, listen, share perception, and interact with a person to solve tasks.

Person
to
Person

Person
to
Avatar



We Still Do Vision!

TITLE	CITED BY	YEAR
Inception and ResNet features are (almost) equivalent D McNeely-White, JR Beveridge, BA Draper Cognitive Systems Research 59, 312-318		2020
Looking Under the Hood: Visualizing What LSTMs Learn D Patil, BA Draper, JR Beveridge International Conference on Image Analysis and Recognition, 67-80		2019
Inception and ResNet: Same Training, Same Features DG McNeely-White, JR Beveridge, BA Draper Biologically Inspired Cognitive Architectures Meeting, 352-357		2019
Continuous gesture recognition through selective temporal fusion P Narayana, JR Beveridge, BA Draper 2019 International Joint Conference on Neural Networks (IJCNN), 1-8	3	2019
Analyzing multi-channel networks for gesture recognition P Narayana, JR Beveridge, BA Draper 2019 International Joint Conference on Neural Networks (IJCNN), 1-8	3	2019
Face Detection in Repeated Settings MN Teli, BA Draper, JR Beveridge arXiv preprint arXiv:1903.08649		2019
Adapting RGB Pose Estimation to New Domains G Mulay, BA Draper, JR Beveridge 2019 IEEE 9th Annual Computing and Communication Workshop and Conference …		2019
Rotary manifold for automating a paper-based Salmonella immunoassay CS Carrell, RM Wydallis, M Bontha, KE Boehle, JR Beveridge, BJ Geiss, ... RSC Advances 9 (50), 29078-29086	1	2019

And from CS 510 Last Spring

The images were collected on the Barry M. Goldwater Air Force Range (BMGR) in Arizona. Provided as a courtesy to CS 510 to demonstrate proof of concept.

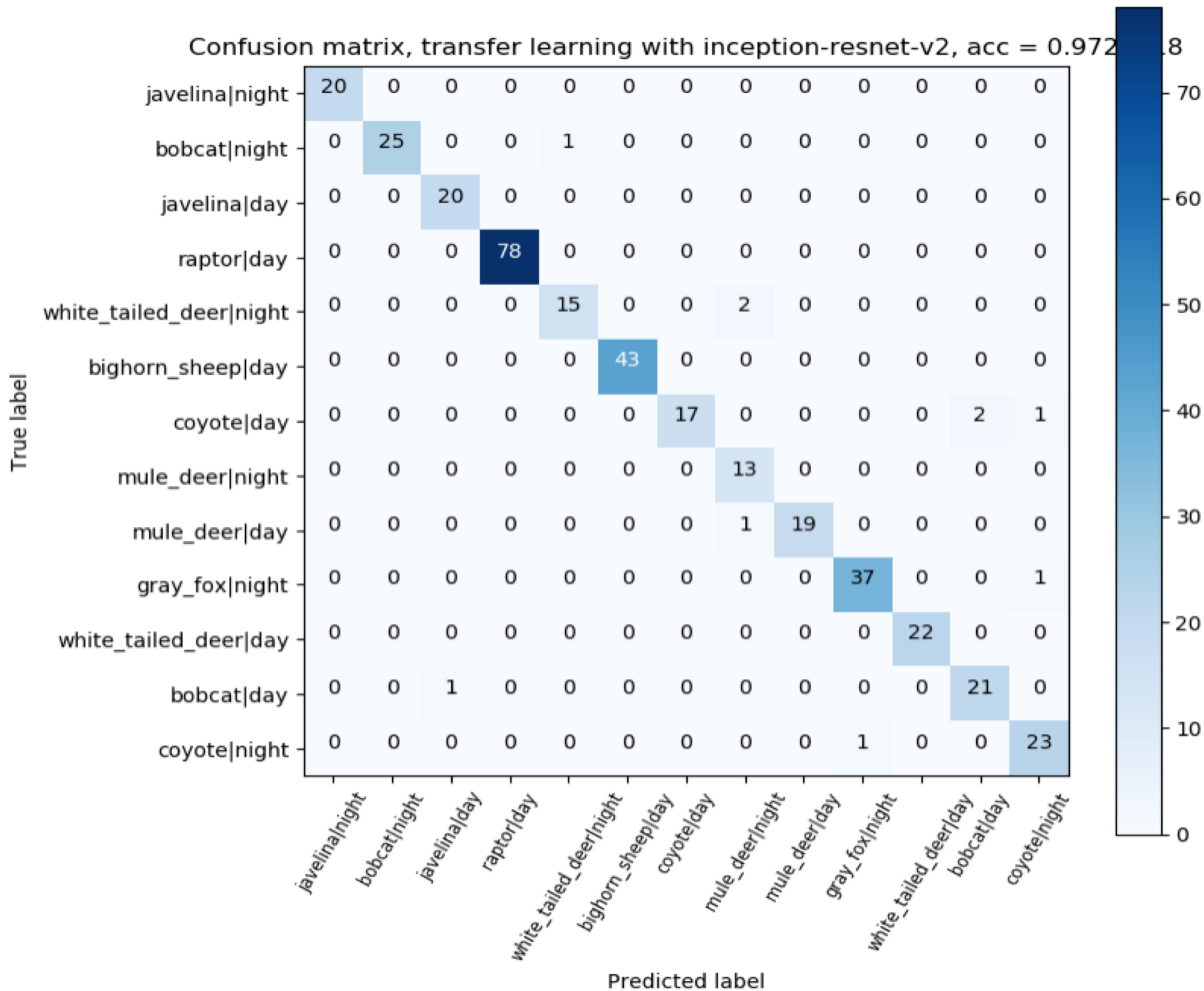
Label: Mule Deer
white tailed deer 0.75579
mule deer 0.23304494
bobcat 0.005833398

Label: White Tail Deer
bobcat 0.5680608
white tailed deer 0.3562632
gray fox 0.046345647

Label: Mule Deer
Top 3: Classification
white tailed deer 0.499
mule deer 0.4859644
coyote 0.0058485395

Label: White Tale Deer
Top 3: Classification
mule deer 0.62270
white tailed deer 0.3751
coyote 0.0005578

Recognition Accuracy Example



From students Brandon Gildemaster and Yan Wang

So What About Jobs

Quick informal survey of LinkedIn on December 10 2019

Software Engineer	262,000
Computer Vision	47,000
Machine Learning	46,000
Cybersecurity	25,000
Bioinformatics	3,000
Computer Graphics	3,000
Natural Language Processing	1,000
High Performance Computing	989

CS410 – A Good Word: Much CS 410 is designed to train skills that feed directly into Computer Vision. We even published a paper on this design.