{Human, Soar}-in-the-loop

Visual Guidance through Reasoning

Mohamed El Banani

University of Michigan - Ann Arbor
37th Soar Workshop

June 7, 2017

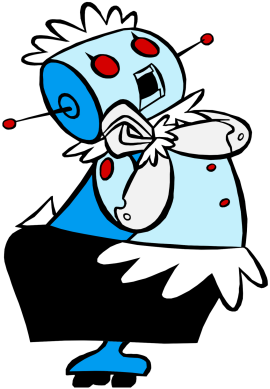
Overview

1. Motivation
2. Hybrid Intelligence
3. Application: Viewpoint Estimation
4. Conclusion

Motivation

Motivation

Can an agent use its reasoning and interaction capabilities to improve its perception ?



Source: thejetsons.wikia.com

Hybrid Intelligence

What is Hybrid Intelligence ?

Hybrid Intelligence has been studied under different names:

- Hybrid Intelligent Systems: "Computational architectures integrating neural and symbolic processes." [4]
- Human-in-the-loop Systems: "The system asks humans to make judgments whenever the computer is less confident – resulting in the most accurate, trustworthy system."[1]
- Symbiotic Autonomy: " [...] a robot reasons about, plans for, and overcomes its limitations by proactively asking humans in the environment for help". [6]

Reasons for Hybrid Intelligence

Different intelligent systems have different strengths:

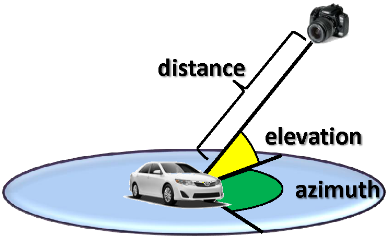
- Symbolic Rule-based systems work very well when the environment can be abstracted in such a way that allows rules to be applied.
- Connectionist systems work very well when a general pattern exists in the data.
- Humans can flexibly deal with anomalies and novel cases, and they think out of the "box".

Systems that are able to leverage the strengths of all those systems are likely to perform better than any single one of those sytems.

Viewpoint Estimation

Task Description

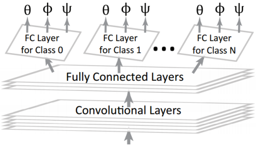
- Estimate the agent's viewpoint from a 2D image.
- A richer description than location or object class.



Source: PASCAL 3D+ Dataset [7]

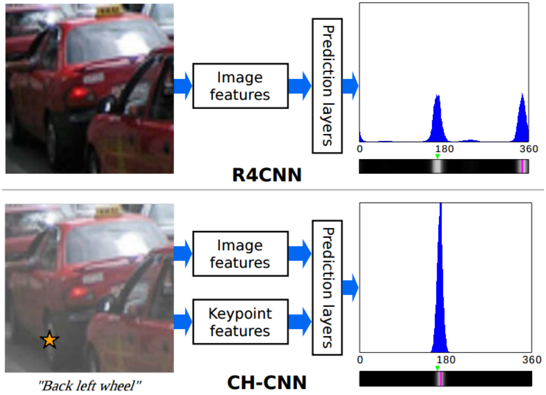
Typical Approaches

- Match image to a 3-D model.[2]
- Train a neural network.[3]



Source: Top: FPM [2]. Bottom: Render For CNN [3]

Human-in-the-loop Viewpoint Estimation

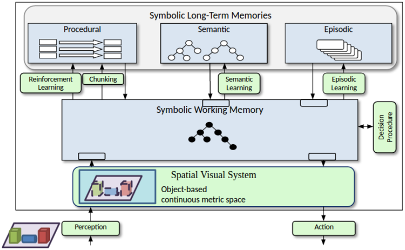


Source: Click-Here CNN [5]

Soar-in-the-loop Viewpoint Estimation

Why use Soar ?

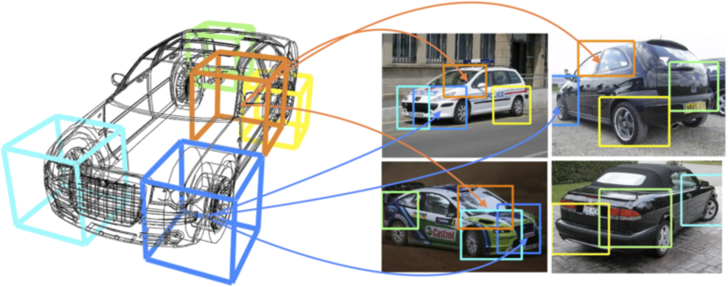
1. Interface to minimize expected human input.
2. Provide an autonomous agent with more control over its perception.



Source: Nate Derbinsky

SVS meets Deformable Parts Models

- Part detectors combined with a parts model could allow for reasoning about part relations.
- SVS would support the parts model and spatial reasoning.



Source: Max Planck Institute

Conclusion

Conclusion

Nuggets:

1. Hybrid Intelligence allows one to leverage different intelligent systems (including humans).
2. Auxillary input can be used to improve the performance of a deep learning vision system.
3. Soar could use its reasoning and interaction capabilities to improve its perception.
4. Integrating deep learning with Soar.

Coal:

1. To be implemented!

References I

- Clare Corthell.
  Hybrid intelligence: How artifical assistants work, May 2016.
  [Online; posted 28-May-2017].

- Joseph J Lim, Aditya Khosla, and Antonio Torralba.
  Fpm: Fine pose parts-based model with 3d cad models.
  In *European Conference on Computer Vision*, pages 478–493. Springer, 2014.

- Hao Su, Charles R. Qi, Yangyan Li, and Leonidas J. Guibas.
  Render for cnn: Viewpoint estimation in images using cnns trained with rendered 3d model views.
  In *The IEEE International Conference on Computer Vision (ICCV)*, December 2015.

References II

Ron Sun and Lawrence A Bookman.
Computational architectures integrating neural and symbolic processes: A perspective on the state of the art, volume 292.
Springer Science & Business Media, 1994.

Ryan Szeto and Jason J Corso.
Click here: Human-localized keypoints as guidance for viewpoint estimation.
arXiv preprint arXiv:1703.09859, 2017.

Manuela M Veloso, Joydeep Biswas, Brian Coltin, and Stephanie Rosenthal.
Cobots: Robust symbiotic autonomous mobile service robots.
In *IJCAI*, page 4423, 2015.

References III

Yu Xiang, Roozbeh Mottaghi, and Silvio Savarese.
Beyond pascal: A benchmark for 3d object detection in the wild.
In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2014.

Human-in-the-loop Performance

	$Acc_{\pi/6}$			
	bus	car	motor	*mean*
R4CNN [22]	92.4	78.5	81.4	84.1
R4CNN [22], fine-tuned	90.6	82.4	84.1	85.7
Keypoint features (Gaussian fixed attn.)	88.9	81.3	82.8	84.4
Keypoint features (uniform fixed attn.)	90.6	82.0	83.7	85.4
CH-CNN (keypoint map only)	90.6	82.0	84.2	85.6
CH-CNN (keypoint class only)	90.9	86.3	83.1	86.8
CH-CNN (keypoint map + class)	**96.8**	**90.2**	**85.2**	**90.7**

Source: Click-Here CNN [5]