On Efficient Bayesian Scene Interpretation: An Information Pursuit Approach

Ehsan Jahangiri

Johns Hopkins University
Center for Cognition, Vision, and Learning (CCVL)

April 20, 2017

Outline

1. Introduction
   - Scene Understanding: Machines vs. Humans
   - Convolutional Neural Networks
   - Scene Understanding and Context
   - Motivation and Goal
2. Information Pursuit
   - Information Pursuit Foundation
   - JHU Table-Setting Dataset
   - Prior Model
   - Classifiers
   - Data Model
   - Results
3. Conclusion

Outline

1. Introduction
   - Scene Understanding: Machines vs. Humans
   - Convolutional Neural Networks
   - Scene Understanding and Context
   - Motivation and Goal

2. Information Pursuit
   - Information Pursuit Foundation
   - JHU Table-Setting Dataset
   - Prior Model
   - Classifiers
   - Data Model
   - Results

3. Conclusion

Scene Understanding: Machines vs. Humans

- Interpreting scenes is effortless and instantaneous for people, even generating rich semantic annotations (“telling a story”).
- Machines lag very far behind in understanding images, and building a *description machine* remains a fundamental A.I. challenge.

Office

Dining Room

Farm

Outline

1. Introduction
   - Scene Understanding: Machines vs. Humans
   - Convolutional Neural Networks
   - Scene Understanding and Context
   - Motivation and Goal

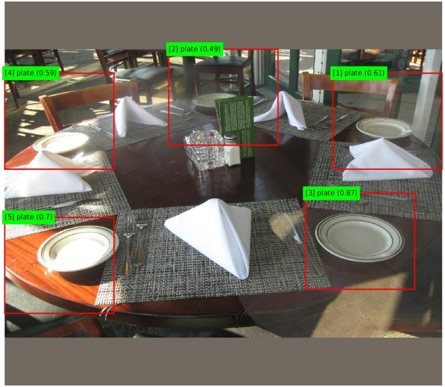
2. Information Pursuit
   - Information Pursuit Foundation
   - JHU Table-Setting Dataset
   - Prior Model
   - Classifiers
   - Data Model
   - Results

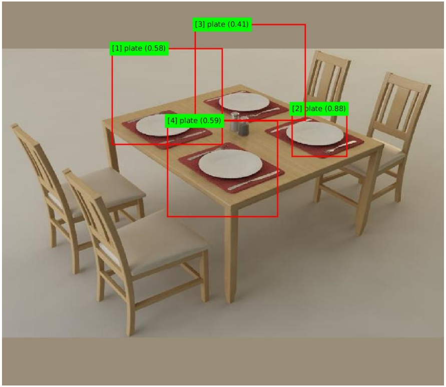
3. Conclusion

Deep Convolutional Neural Networks

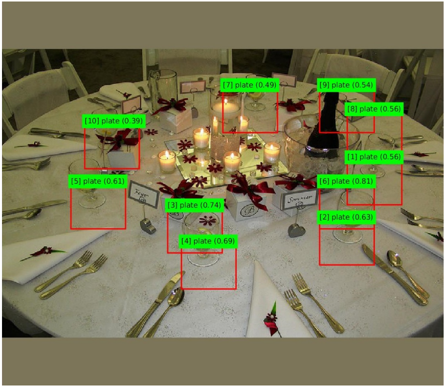
- Convolutional Neural Networks (CNNs) have received a flurry of interest in the past few years due to their superior performance.
- Deep neural networks are *loosely* inspired by how the brain works.
- Deep networks are computationally demanding and require large datasets for efficient training.
- Our understanding of the visual system seems to be roughly consistent with convolutional networks without benefiting from visual selective attention and top-down contextual feedback connections.

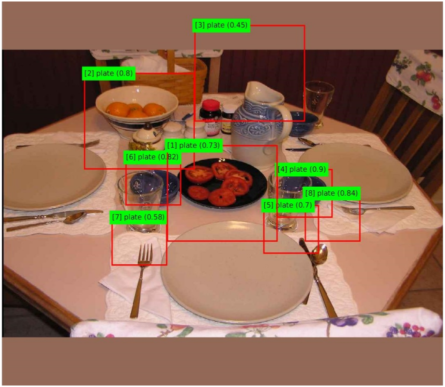
Perfect "plate" detections by CNNs



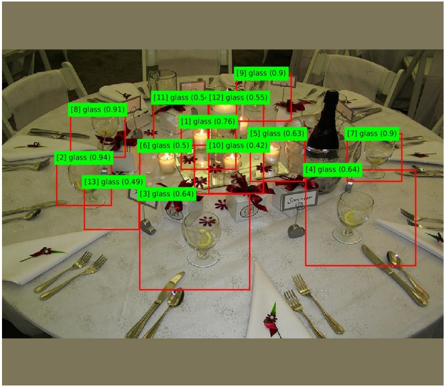


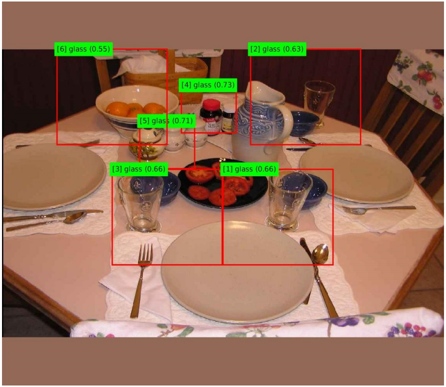
Poor “plate” detections by CNNs



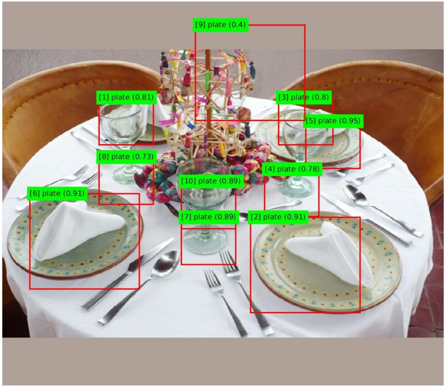


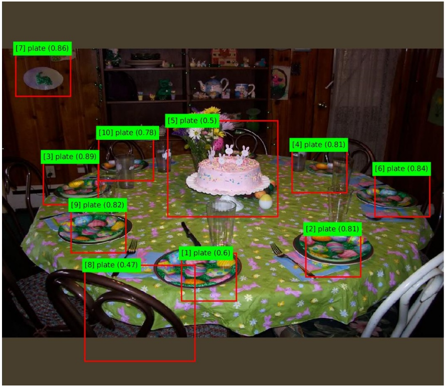
“glass” detections by CNNs





Contextually inconsistent detections





Outline

1. Introduction
   - Scene Understanding: Machines vs. Humans
   - Convolutional Neural Networks
   - Scene Understanding and Context
   - Motivation and Goal
2. Information Pursuit
   - Information Pursuit Foundation
   - JHU Table-Setting Dataset
   - Prior Model
   - Classifiers
   - Data Model
   - Results
3. Conclusion

Scene Understanding and Context

Two ways to incorporate context:

- **Based on common expert knowledge via pre-defined rules:**
  - **1973:** Fischler and Elschlager [2]
  - **1978:** Hanson and Riseman [3]
- **Based on a statistical model:**
  - **2006:** Jin and Geman [4]
  - **2010:** Porway et. al. [5]
  - **2010:** Torralba et. al. [6]
  - **2012:** Choi et. al. [1]

- The Bayesian approach provides a natural framework to integrate contextual relations and the evidence **E** collected using classifiers.

$$P(\mathbf{Y} \mid \mathbf{E}) = \frac{P(\mathbf{Y}, \mathbf{E})}{P(\mathbf{E})} \propto \underbrace{\boxed{P(\mathbf{Y})}}_{\text{Prior}} \times \underbrace{\boxed{P(\mathbf{E} \mid \mathbf{Y})}}_{\text{based on Data Model}}$$

Bayesian Approach

- The Bayesian approach provides a natural framework to integrate contextual relations and the evidence $\mathbf{E}$ collected using classifiers.

$$P(\mathbf{Y} \mid \mathbf{E}) = \frac{P(\mathbf{Y}, \mathbf{E})}{P(\mathbf{E})} \propto \underbrace{\boxed{P(\mathbf{Y})}}_{\text{Prior}} \times \underbrace{\boxed{P(\mathbf{E} \mid \mathbf{Y})}}_{\text{based on Data Model}}$$

- We use a Bayesian approach called "Information/Entropy Pursuit", where top-down contextual information are incorporated using a prior model.

Outline

1. Introduction
   - Scene Understanding: Machines vs. Humans
   - Convolutional Neural Networks
   - Scene Understanding and Context
   - Motivation and Goal

2. Information Pursuit
   - Information Pursuit Foundation
   - JHU Table-Setting Dataset
   - Prior Model
   - Classifiers
   - Data Model
   - Results

3. Conclusion

Motivation

Humans exploit two key strategies in searching spaces:

Motivation

Humans exploit two key strategies in searching spaces:

- **Divide-and-Conquer:**
  - The divide-and-conquer strategy is often used in parlor games such as the "20 Questions Game".
  - We ask the **right questions** in the **right order** - *for computational efficiency by prioritizing what to do next*.

Motivation

Humans exploit two key strategies in searching spaces:

- **Divide-and-Conquer:**
  - The divide-and-conquer strategy is often used in parlor games such as the "20 Questions Game".
  - We ask the **right questions** in the **right order** - *for computational efficiency by prioritizing what to do next*.
- **Selective Attention:**
  - We capture selective attention by selecting potential targets (**focusing**) and ignoring others.
  - We acquire evidence from various locations and at **different resolutions**, usually *coarse-to-fine*, and integrate it coherently by updating likelihoods.
  - **Coarse-to-fine** investigation is a natural byproduct of information/entropy pursuit.

Goal

- To study the stepwise evolution of multi-category object recognition when contextual information are incorporated.

Goal

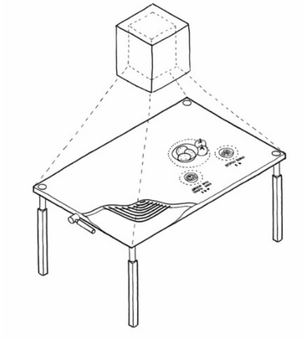
- To study the stepwise evolution of multi-category object recognition when contextual information are incorporated.
- To show that we can achieve almost the same detection accuracy by processing only a fraction of patches from the input image.

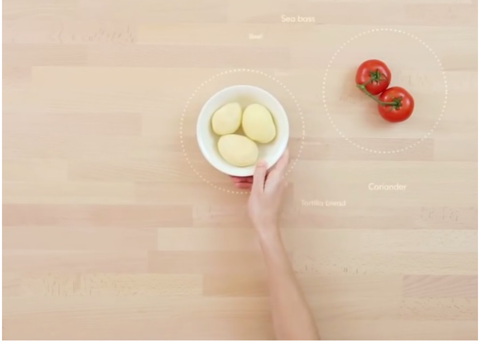
Goal

- To study the stepwise evolution of multi-category object recognition when contextual information are incorporated.
- To show that we can achieve almost the same detection accuracy by processing only a fraction of patches from the input image.
- As proof of concept we evaluated our approach for detecting objects in table-setting scenes.

IKEA Smart Kitchen Concept for 2025

- More than just a table:





Outline

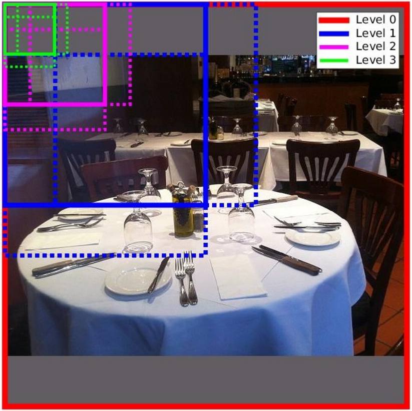
1. Introduction
   - Scene Understanding: Machines vs. Humans
   - Convolutional Neural Networks
   - Scene Understanding and Context
   - Motivation and Goal

2. Information Pursuit
   - Information Pursuit Foundation
   - JHU Table-Setting Dataset
   - Prior Model
   - Classifiers
   - Data Model
   - Results

3. Conclusion

Annocell Hierarchy

- Information Pursuit (IP) operates over a pre-defined collection of partial interpretation units defined based on an annocell hierarchy and the object categories of interest $\mathcal{C}$.
- The Annocell Hierarchy is a partitioning of the input image at different levels of resolution (coarse-to-fine).



Interpretation Units: Annobits and Annoints

- $\mathcal{A}$: hierarchy of image patches (sub-windows).
- Interpretation units:
  - **Annobit:** we define an annobit for every object category of interest $c \in \mathcal{C}$ and every patch from $\mathcal{A}$. An annobit is a high-level "yes-no" question about the scene that is basically the presence indicator of an object from a specific category inside the patch.
  - **Annoint:** an annoint is a composite interpretation unit determined by grouping a set of $|\mathcal{C}|$ annobits corresponding to the same image patch but associated with different categories.
- Y_A: an interpretation unit answering "*What is going on in A?*" for $A \in \mathcal{A}$. For example, for $\mathcal{C} = \{plate, bottle, glass, utensil\}$, an annoint indicates which categories have instances fully inside A.
- Y_A can take $2^{|\mathcal{C}|}$ values.
- Our objective is to recover the ground-truth interpretation units using our information/entropy pursuit approach.

Information Pursuit

- Information Pursuit (IP) is an *adaptive* and *stepwise* search strategy.
- Entropy Pursuit (EP) is a special case of IP under an approximation.
- We attempt to recover the ground-truth interpretation units by collecting evidence about the scene in an optimal order guided by the principle of uncertainty reduction.

Sequential Testing Strategy

- $\mathbf{q}_k = \{q_1, ..., q_k\} \subset \mathcal{Q}$: k previously asked questions.
- $\mathbf{E}_k = \{X_{q_1}(I) = x_1, \dots, X_{q_k}(I) = x_k\}$: evidence acquired after running k classifiers.
- $\mathbf{Y}_\mathcal{Q}$: collection of all interpretation units.

- *Information Pursuit:*

$$q_k = \underset{q \in \mathcal{Q}}{\arg\max}\ \mathcal{I}(X_q, Y_\mathcal{Q} | \mathbf{E}_{k-1}).$$

- *Second Interpretation:*

$$q_k = \underset{q \in \mathcal{Q}}{\arg\min}\, H(Y_\mathcal{Q} | X_q, \mathbf{E}_{k-1}).$$

- Note: $\mathcal{I}(X_q, Y_\mathcal{Q} | \mathbf{E}_{k-1}) = H(Y_\mathcal{Q} | \mathbf{E}_{k-1}) - H(Y_\mathcal{Q} | X_q, \mathbf{E}_{k-1})$.

- *Key Assumption: All classifiers have unit cost (met for CNNs).*

Understanding Selection Criterion

Returning to the interpretation of the selection criterion:

$$q_k = \arg\max_{q \in \mathcal{Q}} \ \mathcal{I}(X_q, Y_{\mathcal{Q}} | \mathbf{E}_{k-1}) = \arg\max_{q \in \mathcal{Q}} \Big\{ H(X_q | \mathbf{E}_{k-1}) - H(X_q | Y_{\mathcal{Q}}, \mathbf{E}_{k-1}) \Big\}.$$

Understanding Selection Criterion

Returning to the interpretation of the selection criterion:

$$q_k = \arg\max_{q \in \mathcal{Q}} \mathcal{I}(X_q, Y_{\mathcal{Q}} | \mathbf{E}_{k-1}) = \arg\max_{q \in \mathcal{Q}} \big\{ H(X_q|\mathbf{E}_{k-1}) - H(X_q|Y_{\mathcal{Q}}, \mathbf{E}_{k-1}) \big\}.$$

This implies that the next question is selected such that:

Understanding Selection Criterion

Returning to the interpretation of the selection criterion:

$$q_k = \arg\max_{q \in \mathcal{Q}} \mathcal{I}(X_q, Y_{\mathcal{Q}} | \mathbf{E}_{k-1}) = \arg\max_{q \in \mathcal{Q}} \big\{ H(X_q | \mathbf{E}_{k-1}) - H(X_q | Y_{\mathcal{Q}}, \mathbf{E}_{k-1}) \big\}.$$

This implies that the next question is selected such that:

- $H(X_q|\mathbf{E}_{k-1})$ is large, so that its answer is as unpredictable as possible given the current evidence, and

Understanding Selection Criterion

Returning to the interpretation of the selection criterion:

$$q_k = \arg\max_{q \in \mathcal{Q}} \mathcal{I}(X_q, Y_{\mathcal{Q}} | \mathbf{E}_{k-1}) = \arg\max_{q \in \mathcal{Q}} \big\{ H(X_q | \mathbf{E}_{k-1}) - H(X_q | Y_{\mathcal{Q}}, \mathbf{E}_{k-1}) \big\}.$$

This implies that the next question is selected such that:

- $H(X_q|\mathbf{E}_{k-1})$ is large, so that its answer is as unpredictable as possible given the current evidence, and
- $H(X_q|Y_{\mathcal{Q}}, \mathbf{E}_{k-1})$ is small, so that X_q is predictable given the ground truth i.e., X_q is a *good* classifier.

Understanding Selection Criterion

Returning to the interpretation of the selection criterion:

$$q_k = \arg\max_{q \in \mathcal{Q}} \mathcal{I}(X_q, Y_{\mathcal{Q}} | \mathbf{E}_{k-1}) = \arg\max_{q \in \mathcal{Q}} \{ H(X_q | \mathbf{E}_{k-1}) - H(X_q | Y_{\mathcal{Q}}, \mathbf{E}_{k-1}) \}.$$

This implies that the next question is selected such that:

- $H(X_q|\mathbf{E}_{k-1})$ is large, so that its answer is as unpredictable as possible given the current evidence, and
- $H(X_q|Y_{\mathcal{Q}}, \mathbf{E}_{k-1})$ is small, so that X_q is predictable given the ground truth i.e., X_q is a *good* classifier.

The two criteria are balanced so that one could accept a relatively poor classifier if it is (currently) highly unpredictable.

Some Assumption for Simplifications

The selection criterion can be simplified if one makes two independence assumptions:

- The classifiers are conditionally independent given $Y_{\mathcal{Q}}$.
- The classifier X_q is conditionally independent of $Y_{\mathcal{Q}\setminus q}$ given Y_q, i.e., the distribution of X_q depends on $Y_{\mathcal{Q}}$ only through Y_q.

Some Assumption for Simplifications

The selection criterion can be simplified if one makes two independence assumptions:

- The classifiers are conditionally independent given $Y_{\mathcal{Q}}$.
- The classifier X_q is conditionally independent of $Y_{\mathcal{Q}\setminus q}$ given Y_q, i.e., the distribution of X_q depends on $Y_{\mathcal{Q}}$ only through Y_q.

The selection criterion term:

$$\mathcal{I}(X_q, Y_{\mathcal{Q}}|\mathbf{E}_{k-1}) = H(X_q|\mathbf{E}_{k-1}) - H(X_q|Y_{\mathcal{Q}}, \mathbf{E}_{k-1}).$$

- **Turns into:** the entropy of a mixture distribution minus a mixture of entropies (namely mixture of $H(X_q|Y_q = y_q)$) with mixture weights $P(Y_q = y_q|\mathbf{E}_{k-1})$.

Some Assumption for Simplifications

The selection criterion can be simplified if one makes two independence assumptions:

- The classifiers are conditionally independent given $Y_{\mathcal{Q}}$.
- The classifier X_q is conditionally independent of $Y_{\mathcal{Q}\setminus q}$ given Y_q, i.e., the distribution of X_q depends on $Y_{\mathcal{Q}}$ only through Y_q.

The selection criterion term:

$$\mathcal{I}(X_q, Y_{\mathcal{Q}}|\mathbf{E}_{k-1}) = H(X_q|\mathbf{E}_{k-1}) - H(X_q|Y_{\mathcal{Q}}, \mathbf{E}_{k-1}).$$

- **Turns into:** the entropy of a mixture distribution minus a mixture of entropies (namely mixture of $H(X_q|Y_q = y_q)$) with mixture weights $P(Y_q = y_q|\mathbf{E}_{k-1})$.
- **Consequently:** given an explicit data model, the IP criterion can be easily computed from the posterior distribution.

Approximation

- A possible simplification is to make the approximation of neglecting the error rates of X_q at the selection stage, therefore replacing X_q by Y_q:

$$q_k = \underset{q \in \mathcal{Q} \setminus \{q_1, \ldots, q_{k-1}\}}{\arg\max} H(Y_q | \mathbf{E}_{k-1}).$$

Approximation

- A possible simplification is to make the approximation of neglecting the error rates of X_q at the selection stage, therefore replacing X_q by Y_q:

$$q_k = \underset{q \in \mathcal{Q} \setminus \{q_1, \ldots, q_{k-1}\}}{\arg\max} H(Y_q | \mathbf{E}_{k-1}).$$

- Hence, pursuing questions whose (true) answers are highly unpredictable.

Approximation

- A possible simplification is to make the approximation of neglecting the error rates of X_q at the selection stage, therefore replacing X_q by Y_q:

$$q_k = \underset{q \in \mathcal{Q} \setminus \{q_1, \dots, q_{k-1}\}}{\arg\max} H(Y_q | \mathbf{E}_{k-1}).$$

- Hence, pursuing questions whose (true) answers are highly unpredictable.

- One would not ask "Is it an urban scene?" after already having got a positive response to "Is there a skyscraper?" nor would one ask if there is an object instance from category c in patch "A" if we already know it is highly likely that there is an object instance from category c in patch "B", a subset of "A".

Approximation

- A possible simplification is to make the approximation of neglecting the error rates of X_q at the selection stage, therefore replacing X_q by Y_q:

$$q_k = \underset{q \in \mathcal{Q} \setminus \{q_1, \ldots, q_{k-1}\}}{\arg\max} H(Y_q | \mathbf{E}_{k-1}).$$

- Hence, pursuing questions whose (true) answers are highly unpredictable.
- One would not ask “Is it an urban scene?” after already having got a positive response to “Is there a skyscraper?” nor would one ask if there is an object instance from category c in patch “A” if we already know it is highly likely that there is an object instance from category c in patch “B”, a subset of “A”.
- We may ask questions in batch.

Relying on Prior and Data Model

- We choose an interpretation unit at step $(k+1)$ whose marginal posterior $P(Y_q = y_q | \mathbf{E}_k)$ has maximum entropy.

Relying on Prior and Data Model

- We choose an interpretation unit at step $(k+1)$ whose marginal posterior $P(Y_q = y_q|\mathbf{E}_k)$ has maximum entropy.

- Posterior:

$$P(\mathbf{Y} \mid \mathbf{E}_k) \propto P(\mathbf{Y}) \times P(\mathbf{E}_k \mid \mathbf{Y}).$$

Relying on Prior and Data Model

- We choose an interpretation unit at step $(k+1)$ whose marginal posterior $P(Y_q = y_q|\mathbf{E}_k)$ has maximum entropy.

- Posterior:

$$P(\mathbf{Y} \mid \mathbf{E}_k) \propto P(\mathbf{Y}) \times P(\mathbf{E}_k \mid \mathbf{Y}).$$

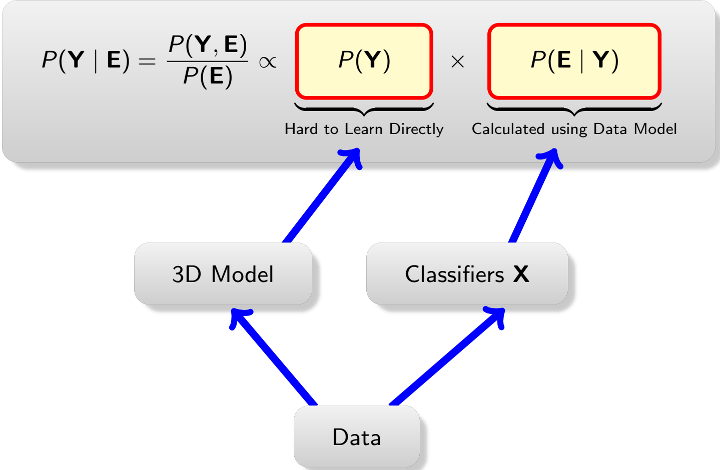
- A Bayesian approach relies on a suitable prior and data model:
  - $P(\mathbf{Y})$: Prior Model,
  - $P(X \mid Y)$: Data Model.

- Instead of designing the prior model in 2D we design in 3D i.e., world coordinate system.

- We project samples from the 3D model to the image coordinate system via perspective projection; then, aggregate the evaluated interpretation units for these samples to estimate $p(Y_q|\mathbf{E}_k)$ required by the IP/EP query engine.

Big Picture

IP/EP query engine needs $P(\mathbf{Y}|\mathbf{E})$.



$$P(\mathbf{Y} \mid \mathbf{E}) = \frac{P(\mathbf{Y},\mathbf{E})}{P(\mathbf{E})} \propto \underbrace{\boxed{P(\mathbf{Y})}}_{\text{Hard to Learn Directly}} \times \underbrace{\boxed{P(\mathbf{E} \mid \mathbf{Y})}}_{\text{Calculated using Data Model}}$$

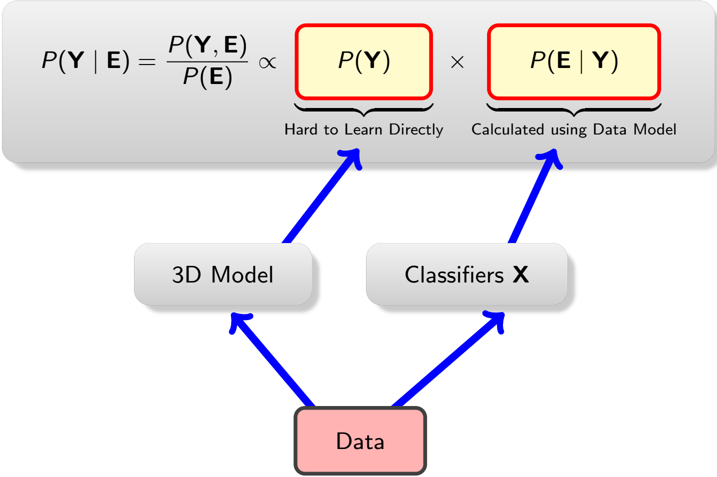
3D Model

Classifiers $\mathbf{X}$

Data

Big Picture

IP/EP query engine needs $P(\mathbf{Y}|\mathbf{E})$.



$$P(\mathbf{Y} \mid \mathbf{E}) = \frac{P(\mathbf{Y},\mathbf{E})}{P(\mathbf{E})} \propto \underbrace{\boxed{P(\mathbf{Y})}}_{\text{Hard to Learn Directly}} \times \underbrace{\boxed{P(\mathbf{E} \mid \mathbf{Y})}}_{\text{Calculated using Data Model}}$$

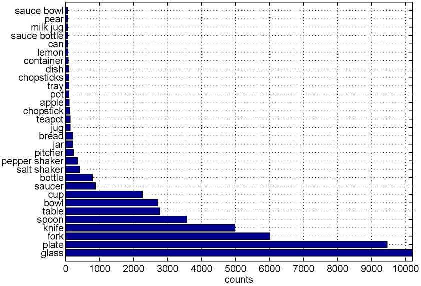
Outline

1. Introduction
   - Scene Understanding: Machines vs. Humans
   - Convolutional Neural Networks
   - Scene Understanding and Context
   - Motivation and Goal

2. Information Pursuit
   - Information Pursuit Foundation
   - JHU Table-Setting Dataset
   - Prior Model
   - Classifiers
   - Data Model
   - Results

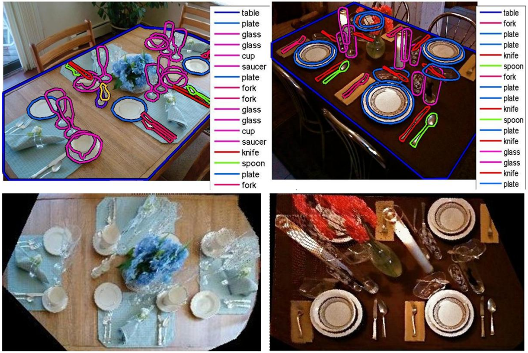
3. Conclusion

JHU Table Setting Dataset



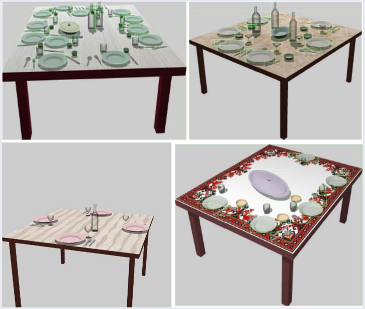
- ≈ 3000 annotated images.
- More than 30 object categories.
- Collected from multiple sources such as Google, Flickr, Altavista etc.
- Three annotators over a period of about ten months (LabelMe).

JHU Table Setting Dataset (Cont'd)



- The *Homography* matrixes are manually (visually) estimated.
- The homography enables us to undo the perspective effect.
- The homography matrixes are scaled appropriately such that the distance of objects in the table's coordinate system, in meters, can be computed.

Table Setting Scene Renderer



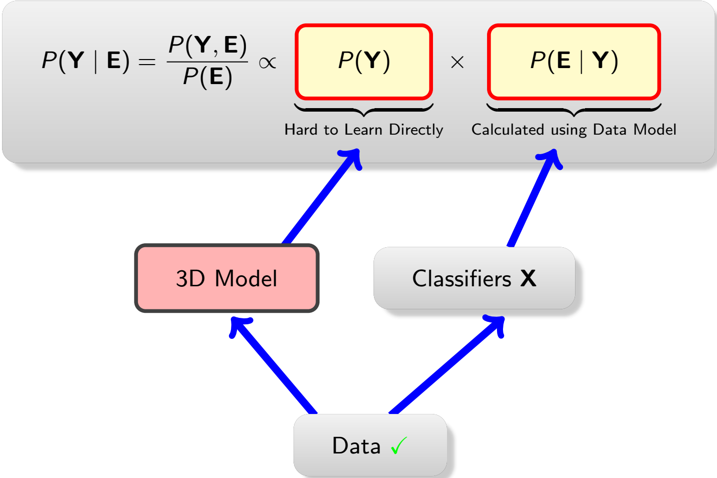
Inputs:

- Camera's calibration parameters, including focal length and pixel length (2 parameters),
- Rotation and translation of the camera (6 parameters),
- Table length and width (2 parameters),
- 3D object poses in the table coordinate system,

and outputs the corresponding table setting scene.

Big Picture

IP/EP query engine needs $P(\mathbf{Y}|\mathbf{E})$.



$$P(\mathbf{Y} \mid \mathbf{E}) = \frac{P(\mathbf{Y},\mathbf{E})}{P(\mathbf{E})} \propto \underbrace{\boxed{P(\mathbf{Y})}}_{\text{Hard to Learn Directly}} \times \underbrace{\boxed{P(\mathbf{E} \mid \mathbf{Y})}}_{\text{Calculated using Data Model}}$$

3D Model

Classifiers **X**

Data ✓

Outline

1. Introduction
   - Scene Understanding: Machines vs. Humans
   - Convolutional Neural Networks
   - Scene Understanding and Context
   - Motivation and Goal

2. Information Pursuit
   - Information Pursuit Foundation
   - JHU Table-Setting Dataset
   - **Prior Model**
   - Classifiers
   - Data Model
   - Results

3. Conclusion

Quote by George Box

"**Remember that all models are wrong; the practical question is how wrong do they have to be to not be useful.**"

George E. P. Box
(18 October 1919 - 28 March 2013)

Box, G. E. P., and Draper, N. R., (1987), Empirical Model Building and Response Surfaces, John Wiley Sons, New York, NY. (p. 74)

Prior Model

- We design a generative prior model in the world coordinate system based on attributed graphs with random structure.
- The generative attributed graph (GAG) model is at the level of objects encoding favored relationships among instances from a distinguished family of object categories.
- The GAG model has interpretable parameters and can be learned efficiently from limited number of annotated images; however, it suffers from slow conditional inference.
- To achieve faster inference we designed a second model based on MRF whose parameters are learned from the GAG model samples.
- Each sample from the GAG/MRF model is a 3D scene describing a table-setting.

Joint Prior Distribution

- Joint Prior Distribution

$$P(\underbrace{\xi_V}_{\text{2D poses}}, \underbrace{g}_{\text{attributed graph}}, \underbrace{\mathcal{W}}_{\text{homography free variables}}, \underbrace{T}_{\text{table geometry}}).$$

Joint Prior Distribution

- Joint Prior Distribution

$$P(\underbrace{\xi_V}_{\text{2D poses}}, \underbrace{g}_{\text{attributed graph}}, \underbrace{\mathcal{W}}_{\text{homography free variables}}, \underbrace{T}_{\text{table geometry}}).$$

- Chain rule:

$$P(\xi_V, g, \mathcal{W}, T) = \underbrace{P(\xi_V \mid g, \mathcal{W})}_{\text{Deterministic}} \times \underbrace{\boxed{P(g \mid T)}}_{\text{GAG model}} \times \underbrace{\boxed{P(\mathcal{W})}}_{\text{Hom. Param. Dist.}} \times \underbrace{\boxed{P(T)}}_{\text{Table Geom Dist.}}$$

Joint Prior Distribution

- Joint Prior Distribution

$$P(\underbrace{\xi_V}_{\text{2D poses}}, \underbrace{g}_{\text{attributed graph}}, \underbrace{\mathcal{W}}_{\text{homography free variables}}, \underbrace{T}_{\text{table geometry}}).$$

- Chain rule:

$$P(\xi_V, g, \mathcal{W}, T) = \underbrace{P(\xi_V \mid g, \mathcal{W})}_{\text{Deterministic}} \times \underbrace{\boxed{P(g \mid T)}}_{\text{GAG model}} \times \underbrace{\boxed{P(\mathcal{W})}}_{\text{Hom. Param. Dist.}} \times \underbrace{\boxed{P(T)}}_{\text{Table Geom Dist.}}$$

- The relationship between ξ_V, g and $\mathcal{W}$ is deterministic via perspective projection (homography).

Joint Prior Distribution

- Joint Prior Distribution

$$P(\underbrace{\xi_V}_{\text{2D poses}}, \underbrace{g}_{\text{attributed graph}}, \underbrace{\mathcal{W}}_{\text{homography free variables}}, \underbrace{T}_{\text{table geometry}}).$$

- Chain rule:

$$P(\xi_V, g, \mathcal{W}, T) = \underbrace{P(\xi_V \mid g, \mathcal{W})}_{\text{Deterministic}} \times \underbrace{\boxed{P(g \mid T)}}_{\text{GAG model}} \times \underbrace{\boxed{P(\mathcal{W})}}_{\text{Hom. Param. Dist.}} \times \underbrace{\boxed{P(T)}}_{\text{Table Geom Dist.}}$$

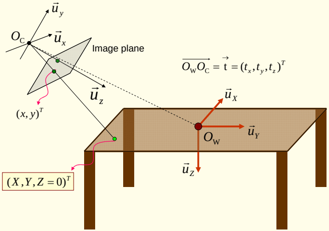
- The relationship between ξ_V, g and $\mathcal{W}$ is deterministic via perspective projection (homography).
- $P(\xi_V \mid g, \mathcal{W})$ is a degenerate distribution but included anyways to make the point that samples from the 3D GAG model are projected onto the 2D image coordinate system.

Homography

- The homography matrix H is a deterministic function of the camera's extrinsic and intrinsic parameters i.e. $H = H(\mathcal{W})$, where

$$\mathcal{W} = (\underbrace{\phi_x, \phi_y, \phi_z}_{\text{angular variables}} , \underbrace{t_x, t_y, t_z}_{\text{translation vector}} , \underbrace{f}_{\text{focal length}} , \underbrace{s_x, s_y}_{\text{pixel size}} , \underbrace{\dot{x}_0, \dot{y}_0}_{\text{image center}}).$$

$(\dot{x}_0, \dot{y}_0)$: the point on the image (in pixels) where the camera's principal axis meets the image plane. In case that the image is not cropped $\dot{x}_0 = \frac{\#\text{of image columns}}{2}$ and $\dot{y}_0 = \frac{\#\text{of image rows}}{2}$.



We choose a coordinate whose origin is at the center of the table, whose X and Y axes are on the table, and whose Z axis is orthogonal to the table.

Specifying Distributions

- Prior model:

$$P(\xi_V, g, \mathcal{W}, T) = \underbrace{P(\xi_V \mid g, \mathcal{W})}_{\text{Deterministic}} \times \underbrace{\boxed{P(g \mid T)}}_{\text{Gen. Attrib. Graph model}} \times \underbrace{\boxed{P(\mathcal{W})}}_{\text{Hom. Param. Dist.}} \times \underbrace{\boxed{P(T)}}_{\text{Table Geom Dist.}}$$

Assumptions

Assumptions:

1. Motivated by our application to table-settings scenes we assume that the scene contains a dominant world plane (the table plane) where different objects lie (spoons, plates, cups, etc.)
2. The height of objects is small relative to their distance to the camera (planar objects).

Assumptions

Assumptions:

1. Motivated by our application to table-settings scenes we assume that the scene contains a dominant world plane (the table plane) where different objects lie (spoons, plates, cups, etc.)
2. The height of objects is small relative to their distance to the camera (planar objects).

- Many human-created scenes are composed of parallel supporting surfaces where different objects lie.

Overview of the GAG Prior Model

- A scene is described as a collection of object instances from different categories at different poses.

Overview of the GAG Prior Model

- A scene is described as a collection of object instances from different categories at different poses.
- Each object instance is associated with a vertex $v \in V$ of a *base* graph $g_0 \in \mathcal{G}_0$ which captures contextual relationships among object instances.

Overview of the GAG Prior Model

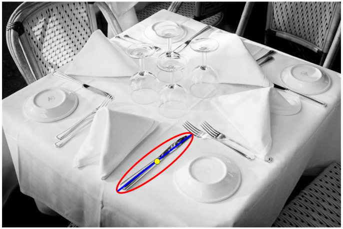
- A scene is described as a collection of object instances from different categories at different poses.
- Each object instance is associated with a vertex $v \in V$ of a *base* graph $g_0 \in \mathcal{G}_0$ which captures contextual relationships among object instances.
- The underlying base graph is a forest of directed trees in special case.

Overview of the GAG Prior Model

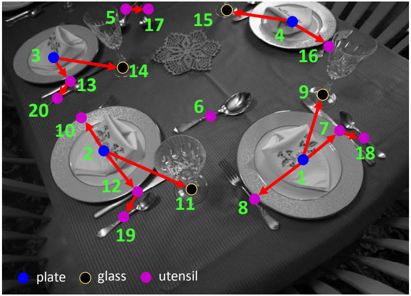
- A scene is described as a collection of object instances from different categories at different poses.
- Each object instance is associated with a vertex $v \in V$ of a *base* graph $g_0 \in \mathcal{G}_0$ which captures contextual relationships among object instances.
- The underlying base graph is a forest of directed trees in special case.
- An attributed graph is a triple $g = (g_0, c_V, \theta_V)$, where:
    - $c_V = \{c_v, v \in V\}$: category labels,
    - $\theta_V = \{\theta_v, v \in V\}$: 3D poses of objects.

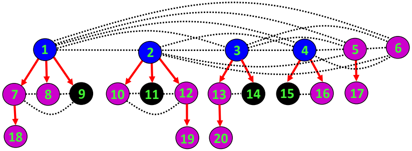
Overview of the GAG Prior Model

- A scene is described as a collection of object instances from different categories at different poses.
- Each object instance is associated with a vertex $v \in V$ of a *base* graph $g_0 \in \mathcal{G}_0$ which captures contextual relationships among object instances.
- The underlying base graph is a forest of directed trees in special case.
- An attributed graph is a triple $g = (g_0, c_V, \theta_V)$, where:
  - $c_V = \{c_v, v \in V\}$: category labels,
  - $\theta_V = \{\theta_v, v \in V\}$: 3D poses of objects.
- Pose of an object can be represented by its fitting ellipse:



Generative Attributed Graph model





Removing the undirected edges (dashed lines) leads to a Bayesian tree.

- The Generative Attributed Graph (GAG) model:

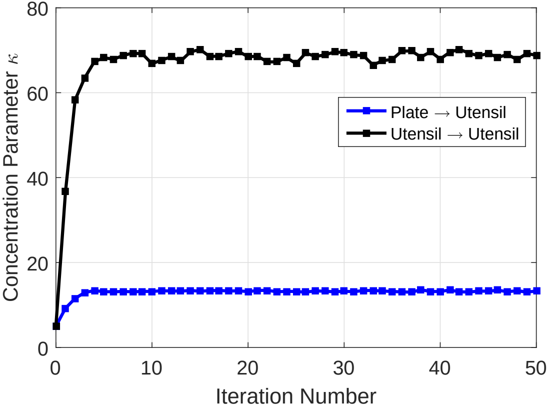
$$
\begin{aligned}
p(g|T) &= p(g_0, c_V|T) \times p(\theta_V|g_0, c_V, T) \\
&= \underbrace{p^{(0)}(n_{(0,1)}, \cdots, n_{(0,|\mathcal{C}|)}|T)}_{\text{Poisson Dist.}} \; p(\theta_{V_0}|c_{V_0}, T)\times \\
&\qquad \prod_{v \in V \setminus V_T} \underbrace{p^{(c_v)}(n_{(v,1)}, \cdots, n_{(v,|\mathcal{C}|)})}_{\text{Multi-type Branching Process}} \times p(\theta_{ch(v)}|c_{ch(v)}, c_v, \theta_v, T).
\end{aligned}
$$

Learning the GAG Model (Details Skipped)

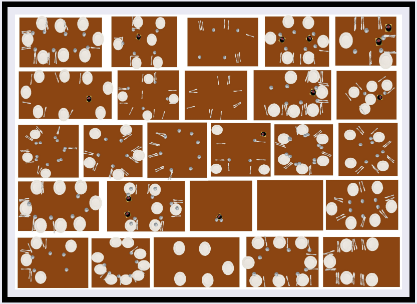
- We used *stochastic Expectation-Maximization* to learn the GAG model.

Learning the GAG Model (Details Skipped)

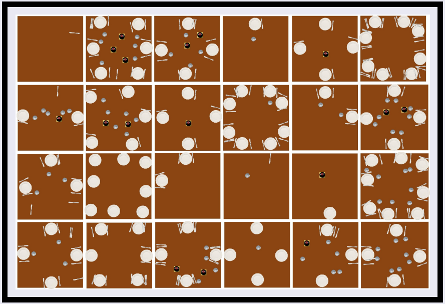
- We used *stochastic Expectation-Maximization* to learn the GAG model.
- After only a few iterations (< 10) the MCEM algorithm for parameter learning converged.



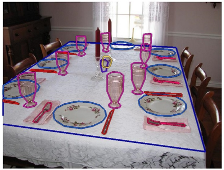
Samples from the dataset vs samples from the GAG model

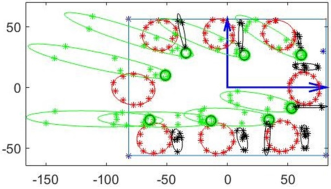


samples from JHU-Dataset



samples from GAG model





GAG Model Conditional Sampling

- Even though conditional sampling from $p(g|T, X)$ is not a necessary building block in the process of our application it was still studied.
- Conditional MCMC sampling based on Metropolis-Hastings with moves including:
  1. Birth and Death of Nodes,
  2. Edge Deletion/Addition,
  3. Pose Change.
- Conditional sampling from a distribution on graphs with random structure is interesting (yet difficult) and could be applied to other problems e.g., a model-based Visual Turing Test (see Geman et al., 2015 in PNAS).

"**My personal bet:** **we may need to understand visual cortex (and the brain!) to achieve scene understanding at human level, and thereby develop systems that pass a full Turing test.**"

Tomaso Poggio

Specifying Distributions

- Prior model:

$$P(\xi_V,\omega,\mathcal{W},T) = \underbrace{P(\xi_V \mid \omega,\mathcal{W})}_{\text{Deterministic}} \times \underbrace{\boxed{P(\omega \mid T)}}_{\text{Replaced by MRF model}} \times \underbrace{\boxed{P(\mathcal{W})}}_{\text{Hom. Param. Dist.}} \times \underbrace{\boxed{P(T)}}_{\text{Table Geom Dist.}}$$

MRF Model

- Samples from $p(g|T)$ were used to generate statistics required to learn the Markov Random Field (MRF) Model.

MRF Model

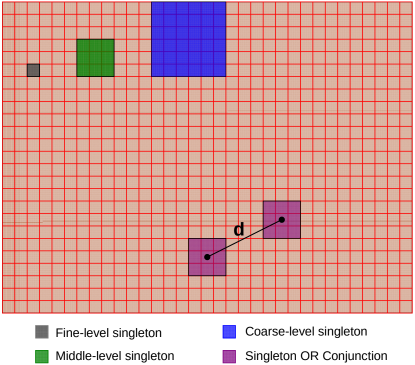
- Samples from $p(g|T)$ were used to generate statistics required to learn the Markov Random Field (MRF) Model.
- MRF model:

$$p_\lambda(\omega) = \frac{1}{Z(\lambda)} \exp\left(\lambda^T.\mathbf{f}(\omega)\right),$$

$\mathbf{f}(\omega) = [f_1(\omega), f_2(\omega), \cdots, f_M(\omega)]^\top$: feature functions (all binary),

$\lambda = [\lambda_1, \lambda_2, \cdots, \lambda_M]^\top$: weights (model parameters).

MRF features

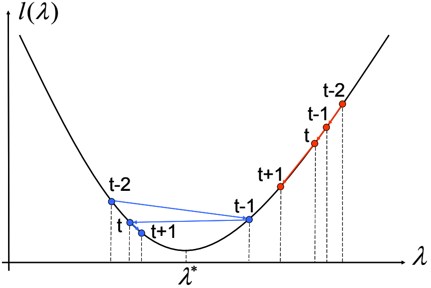


- The **singleton** feature functions are incorporated to preserve the overall empirical statistics on the existence of an object category at a particular location on the table.
- The **conjunction** feature functions are aimed to incorporate the contextual relations between different object categories.

MRF Learning

(DETAILS SKIPPED)

- We took advantage of the symmetry in table-settings ("Invariance Property") to lower the number of parameters.
- We learned 10 MRF models for 10 different table sizes using an accelerated version of the stochastic gradient descent (SGD), by iteratively minimizing the KL divergence between the Gibbs and empirical distribution (equivalent to ML for the exponential family).



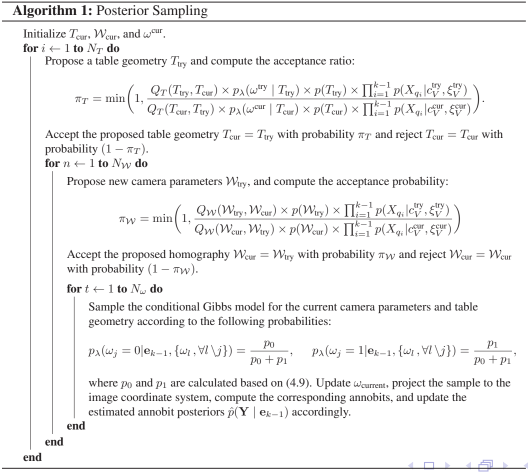
Posterior Sampling
(DETAILS SKIPPED)

- Posterior sampling was carried out in three nested loops:

Outer Loop:	sampling table size (Metropolis-Hastings)
Middle Loop:	sampling homography (Metropolis-Hastings)
Inner Loop:	sampling MRF model (Gibbs sampling)

Posterior Sampling
(DETAILS SKIPPED)

- Posterior sampling was carried out in three nested loops:
  Outer Loop: sampling table size (Metropolis-Hastings)
  Middle Loop: sampling homography (Metropolis-Hastings)
  Inner Loop: sampling MRF model (Gibbs sampling)

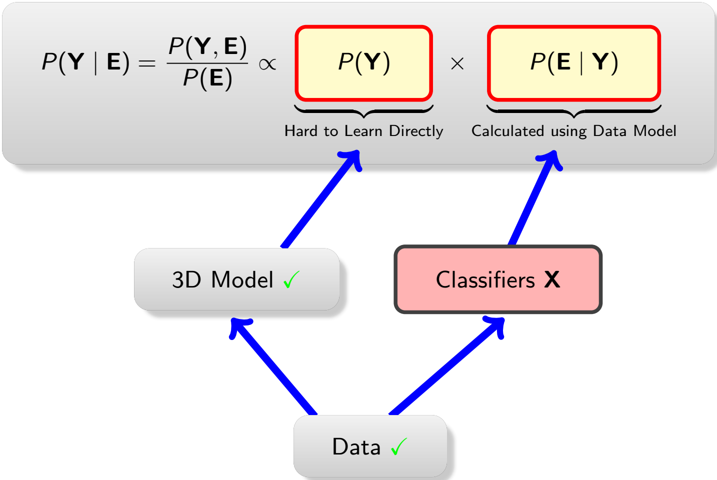


Algorithm 1: Posterior Sampling

```
Initialize $T_{\text{cur}}$, $\mathcal{W}_{\text{cur}}$, and $\omega^{\text{cur}}$.
for $i \leftarrow 1$ to $N_T$ do
    Propose a table geometry $T_{\text{try}}$ and compute the acceptance ratio:
        $$\pi_T = \min\left(1, \frac{Q_T(T_{\text{try}}, T_{\text{cur}}) \times p_\lambda(\omega^{\text{try}} \mid T_{\text{try}}) \times p(T_{\text{try}}) \times \prod_{i=1}^{k-1} p(X_{q_i}|c_V^{\text{try}}, \xi_V^{\text{try}})}{Q_T(T_{\text{cur}}, T_{\text{try}}) \times p_\lambda(\omega^{\text{cur}} \mid T_{\text{cur}}) \times p(T_{\text{cur}}) \times \prod_{i=1}^{k-1} p(X_{q_i}|c_V^{\text{cur}}, \xi_V^{\text{cur}})}\right).$$
    Accept the proposed table geometry $T_{\text{cur}} = T_{\text{try}}$ with probability $\pi_T$ and reject $T_{\text{cur}} = T_{\text{cur}}$ with
    probability $(1 - \pi_T)$.
    for $n \leftarrow 1$ to $N_{\mathcal{W}}$ do
        Propose new camera parameters $\mathcal{W}_{\text{try}}$, and compute the acceptance probability:
            $$\pi_{\mathcal{W}} = \min\left(1, \frac{Q_{\mathcal{W}}(\mathcal{W}_{\text{try}}, \mathcal{W}_{\text{cur}}) \times p(\mathcal{W}_{\text{try}}) \times \prod_{i=1}^{k-1} p(X_{q_i}|c_V^{\text{try}}, \xi_V^{\text{try}})}{Q_{\mathcal{W}}(\mathcal{W}_{\text{cur}}, \mathcal{W}_{\text{try}}) \times p(\mathcal{W}_{\text{cur}}) \times \prod_{i=1}^{k-1} p(X_{q_i}|c_V^{\text{cur}}, \xi_V^{\text{cur}})}\right)$$
        Accept the proposed homography $\mathcal{W}_{\text{cur}} = \mathcal{W}_{\text{try}}$ with probability $\pi_{\mathcal{W}}$ and reject $\mathcal{W}_{\text{cur}} = \mathcal{W}_{\text{cur}}$
        with probability $(1 - \pi_{\mathcal{W}})$.
        for $t \leftarrow 1$ to $N_\omega$ do
            Sample the conditional Gibbs model for the current camera parameters and table
            geometry according to the following probabilities:
                $$p_\lambda(\omega_j = 0|\mathbf{e}_{k-1}, \{\omega_l, \forall l \setminus j\}) = \frac{p_0}{p_0 + p_1}, \qquad p_\lambda(\omega_j = 1|\mathbf{e}_{k-1}, \{\omega_l, \forall l \setminus j\}) = \frac{p_1}{p_0 + p_1},$$
            where $p_0$ and $p_1$ are calculated based on (4.9). Update $\omega_{\text{current}}$, project the sample to the
            image coordinate system, compute the corresponding annobits, and update the
            estimated annobit posteriors $\hat{p}(\mathbf{Y} \mid \mathbf{e}_{k-1})$ accordingly.
        end
    end
end
```

Big Picture

IP/EP query engine needs $p(\mathbf{Y}|\mathbf{E})$.



$$P(\mathbf{Y} \mid \mathbf{E}) = \frac{P(\mathbf{Y},\mathbf{E})}{P(\mathbf{E})} \propto \underbrace{\boxed{P(\mathbf{Y})}}_{\text{Hard to Learn Directly}} \times \underbrace{\boxed{P(\mathbf{E} \mid \mathbf{Y})}}_{\text{Calculated using Data Model}}$$

3D Model ✓

Classifiers **X**

Data ✓

Outline

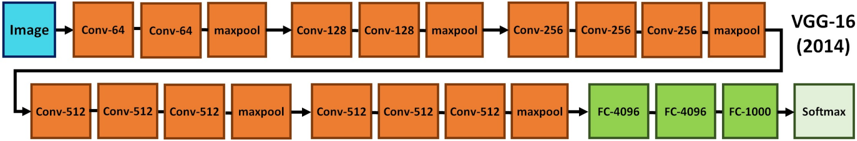
1. Introduction
   - Scene Understanding: Machines vs. Humans
   - Convolutional Neural Networks
   - Scene Understanding and Context
   - Motivation and Goal

2. Information Pursuit
   - Information Pursuit Foundation
   - JHU Table-Setting Dataset
   - Prior Model
   - Classifiers
   - Data Model
   - Results

3. Conclusion

CNN Classifiers

- We trained three deep CNNs, all based on the VGG-16 network (up to layer 15):
  - CatNet: for category classification,
  - ScaleNet: to estimate the scale of detected object instances (skipped),
  - TableNet: to detect the table surface area in a given image (skipped).
- The last fully-connected layer (16-th weight layer) and the following softmax layer of these three CNNs are modified to accommodate the design needs.

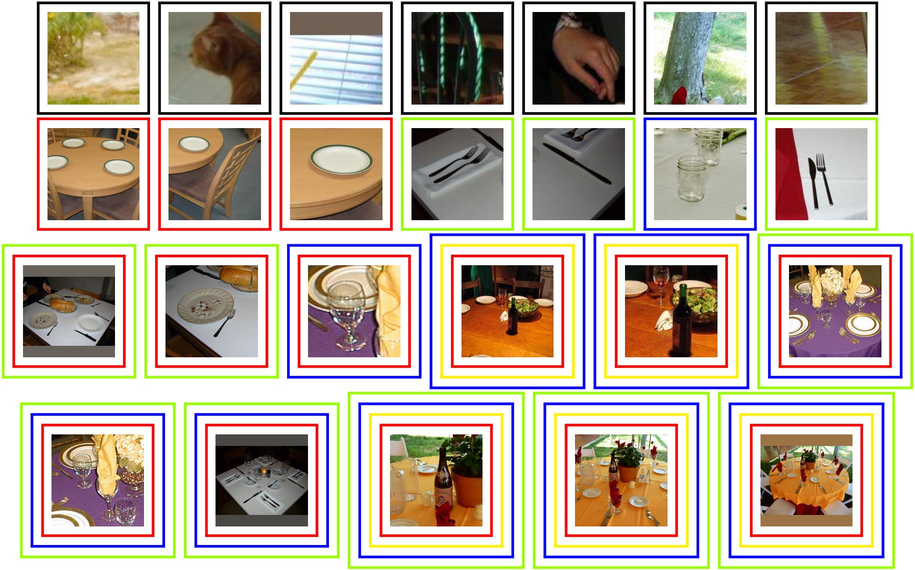


CatNet

- The CatNet is a CNN with a 5-way softmax output layer used to predict the ground-truth annoint associated with the input patch, with:
    - OUTPUT 1: estimating **“No Object”** proportion,
    - OUTPUT 2: estimating **“Plate”** proportion,
    - OUTPUT 3: estimating **“Bottle”** proportion,
    - OUTPUT 4: estimating **“Glass”** proportion,
    - OUTPUT 5: estimating **“Utensil”** proportion.

CatNet: Training/Test Patch Examples

- Color coded categories: **“No Object”**, **“Plate”**, **“Bottle”**, **“Glass”**, and **“Utensil”**.



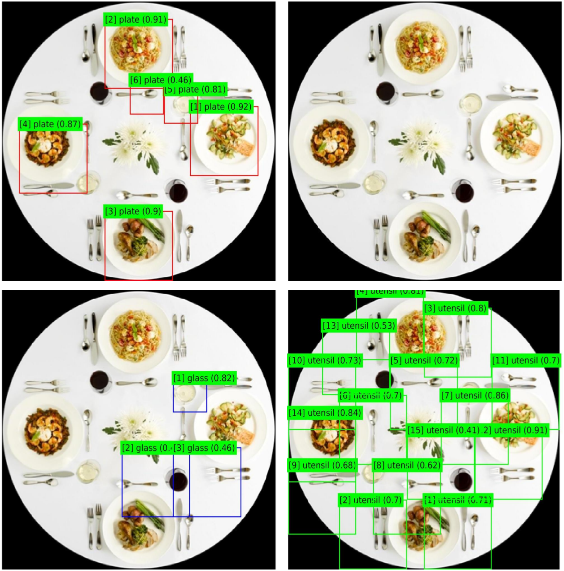
CatNet Testing

- CNN output proportions are processed to obtain binary classification per category.
- We define two parameters (k, S_g) for considering the top-k scores with less than S_g consecutive score gap (distance).
- Suppose $k = 3$ with score gap $S_g = 0.2$, and the CatNet outputs are:

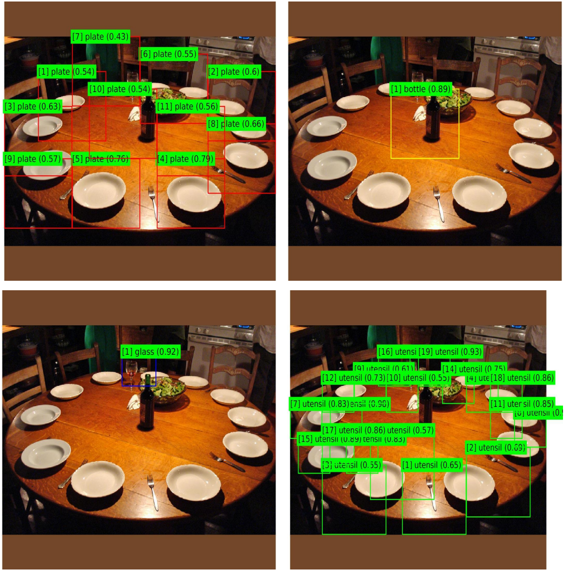
$$(s_1 = 0.05, s_2 = 0.45, s_3 = 0.05, s_4 = 0.1, s_5 = 0.35)$$

since $(s_2 - s_5) < S_g$ but $(s_5 - s_4) \nless S_g$, then categories "2" and "5" are declared as positive detections.

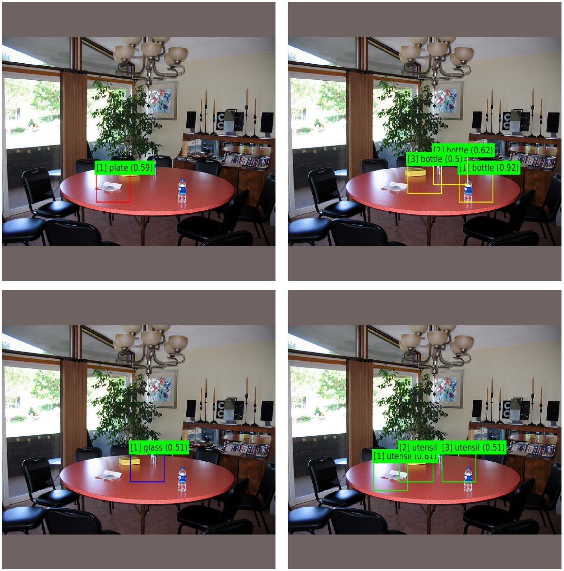
CNN detection examples



CNN detection examples



CNN detection examples



CNN detection examples

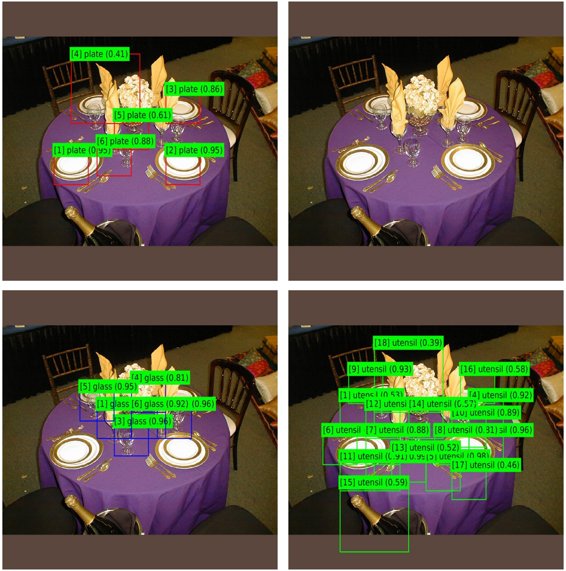
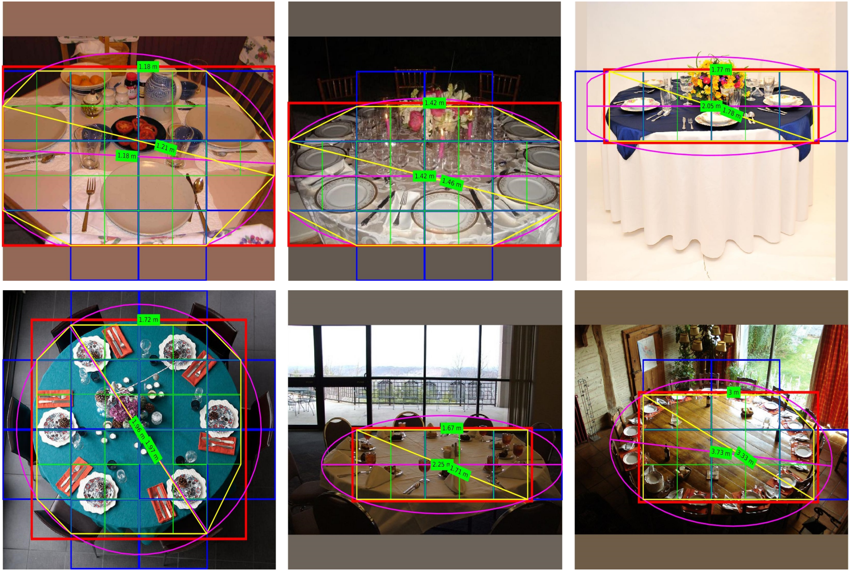
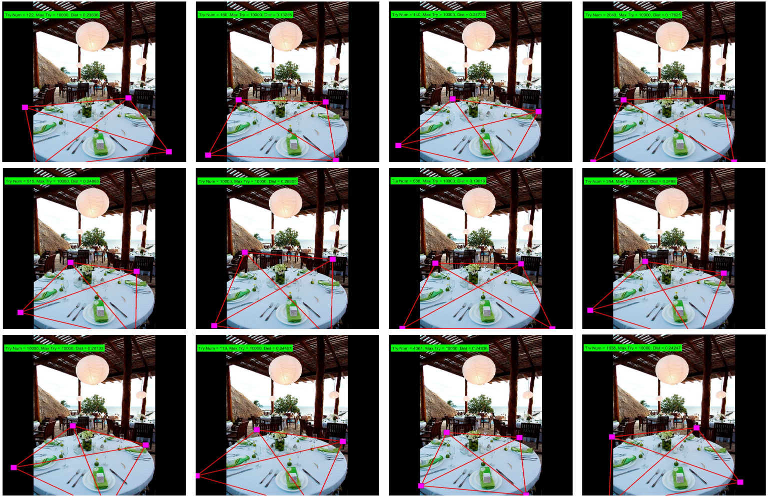


Table detection examples by TableNet

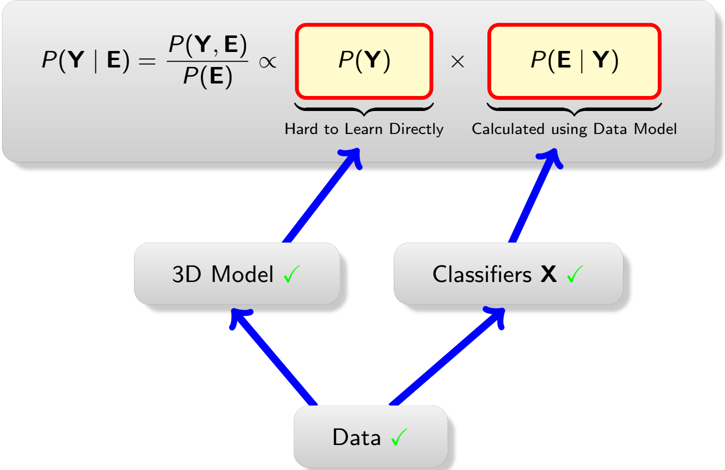


Homography Samples Consistent with Detected Table


Try Num = 122, Max Try = 10000, Dist = 0.23636
Try Num = 166, Max Try = 10000, Dist = 0.13285
Try Num = 140, Max Try = 10000, Dist = 0.24733
Try Num = 2043, Max Try = 10000, Dist = 0.17625
Try Num = 515, Max Try = 10000, Dist = 0.24863
Try Num = 10000, Max Try = 10000, Dist = 0.28803
Try Num = 558, Max Try = 10000, Dist = 0.19016
Try Num = 384, Max Try = 10000, Dist = 0.2488
Try Num = 10000, Max Try = 10000, Dist = 0.29132
Try Num = 119, Max Try = 10000, Dist = 0.24457
Try Num = 4061, Max Try = 10000, Dist = 0.24836
Try Num = 1838, Max Try = 10000, Dist = 0.24247

Big Picture

IP/EP query engine needs $P(\mathbf{Y}|\mathbf{E})$.



$$P(\mathbf{Y} \mid \mathbf{E}) = \frac{P(\mathbf{Y}, \mathbf{E})}{P(\mathbf{E})} \propto \underbrace{\boxed{P(\mathbf{Y})}}_{\text{Hard to Learn Directly}} \times \underbrace{\boxed{P(\mathbf{E} \mid \mathbf{Y})}}_{\text{Calculated using Data Model}}$$

3D Model ✓

Classifiers **X** ✓

Data ✓

Outline

1 Introduction
- Scene Understanding: Machines vs. Humans
- Convolutional Neural Networks
- Scene Understanding and Context
- Motivation and Goal

2 Information Pursuit
- Information Pursuit Foundation
- JHU Table-Setting Dataset
- Prior Model
- Classifiers
- Data Model
- Results

3 Conclusion

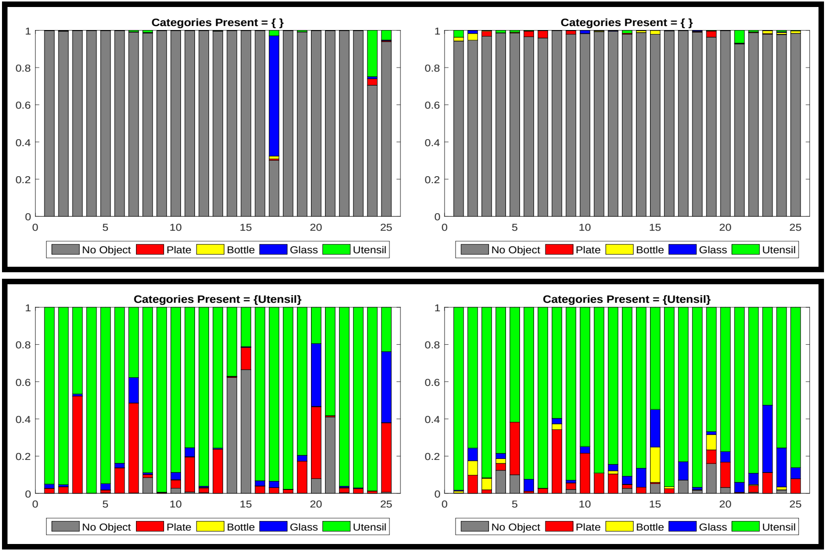
Dirichlet Data Model

- The Dirichlet distribution is a density on probability vectors $\mathbf{x} \in [0,1]^K$.

$$p(\mathbf{x}) \sim \mathrm{Dir}(\alpha_1, ..., \alpha_K) = \frac{\Gamma(\sum_k \alpha_k)}{\prod_k \Gamma(\alpha_k)} \prod_k x_k^{\alpha_k - 1}.$$

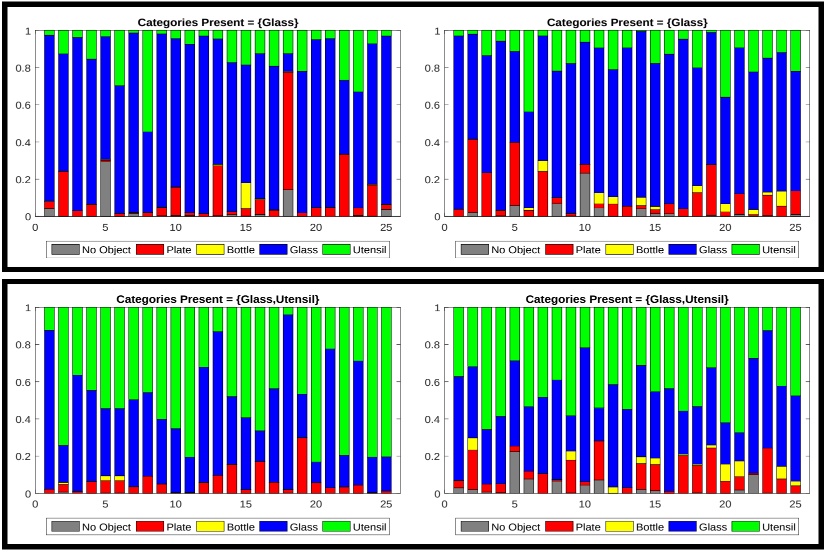
- We learned 16 conditional CatNet data models (MLE) (i.e., 16 Dirichlet models) for the 16 possible configurations of object categories.
- The training data are obtained by running the CNNs on patches with matching configuration.
- Similarly for ScaleNet.

Stacked-Bar Samples



Left: CNN on patches with matching configs. Right: Dirichlet model samples.

Stacked-Bar Samples



Left: CNN on patches with matching configs. Right: Dirichlet model samples.

Outline

1. Introduction
   - Scene Understanding: Machines vs. Humans
   - Convolutional Neural Networks
   - Scene Understanding and Context
   - Motivation and Goal

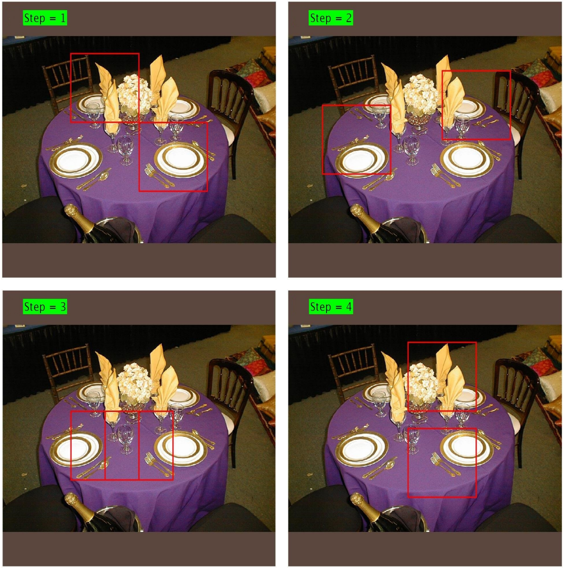
2. Information Pursuit
   - Information Pursuit Foundation
   - JHU Table-Setting Dataset
   - Prior Model
   - Classifiers
   - Data Model
   - Results

3. Conclusion

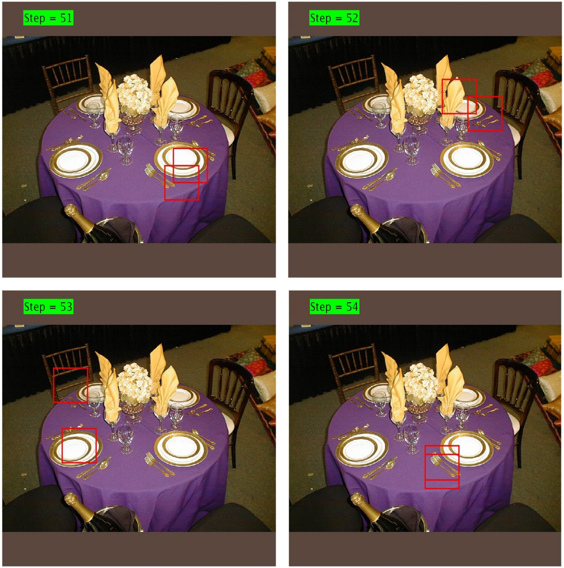
Great Expectations

- *Does coarse-to-fine search emerge naturally from IP/EP?*
- *Can a fraction of the classifiers do as well as all of them?*

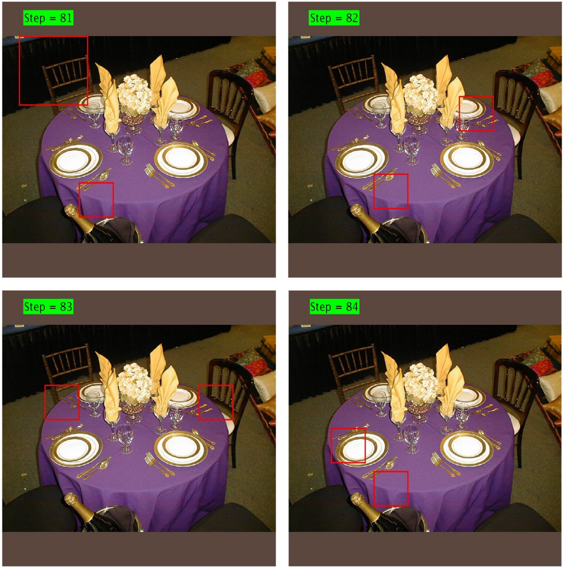
IP questions (steps 1-4)



IP questions (steps 51-54)



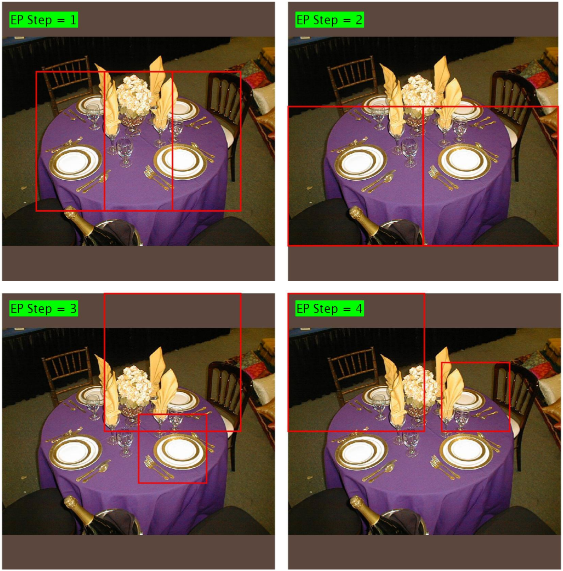
IP questions (steps 81-84)



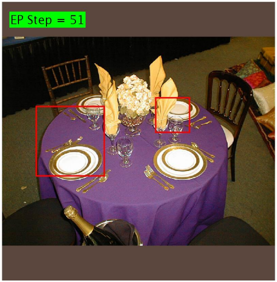
Questions

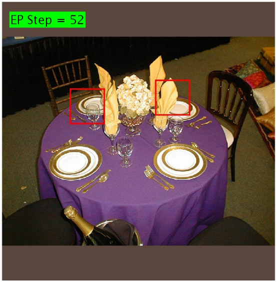
EP vs. IP

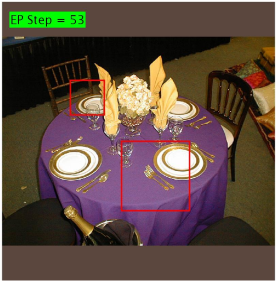
EP questions (steps 1-4)

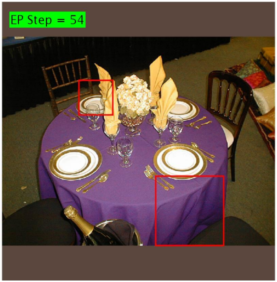


EP questions (steps 51-54)

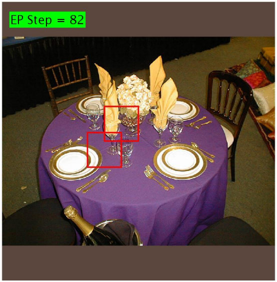


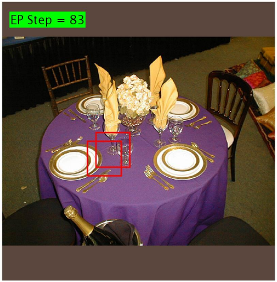


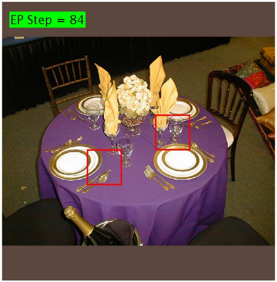




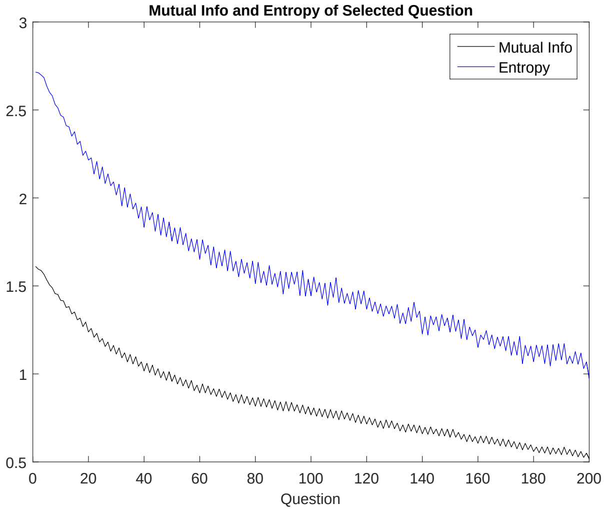
EP questions (steps 81-84)







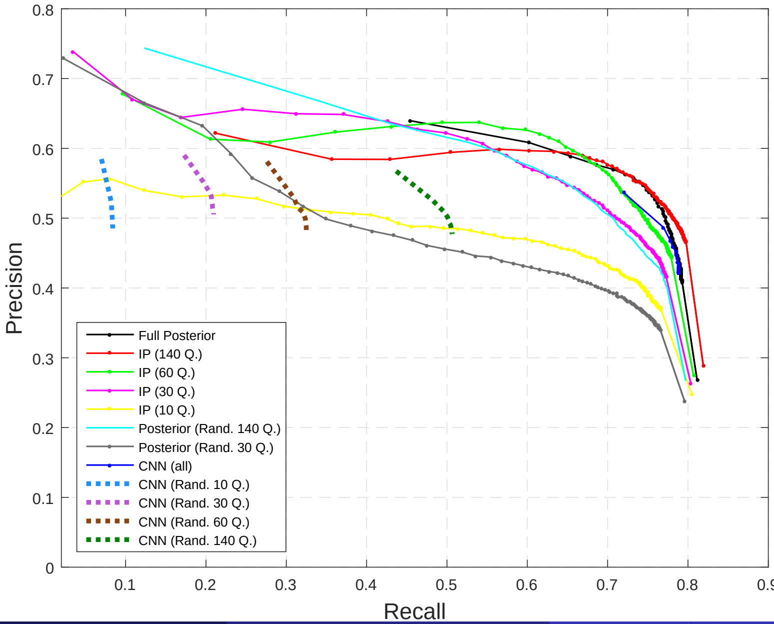
Mutual Information and Entropy of IP Questions



Great Expectations

- *Does coarse-to-fine search emerge naturally from IP/EP?*
- *Can a fraction of the classifiers do as well as all of them?*

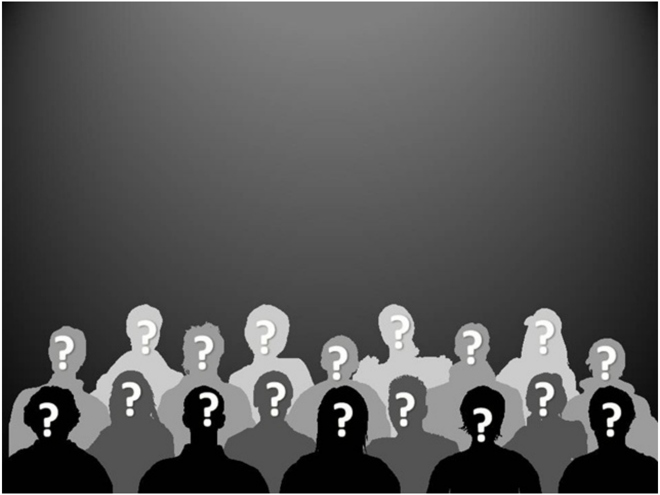
Precision-Recall Curves



Conclusion

- Does coarse-to-fine search emerge naturally from IP/EP? YES
- Can a fraction of the classifiers do as well as all of them? YES

Questions?



A Few of References

M. J. Choi, A. Torralba, and A. S. Willsky.
A tree-based context model for object recognition.
IEEE Trans. Pattern Anal. Mach. Intell., 34(2):240–252, Feb. 2012.

M. A. Fischler and R. A. Elschlager.
The representation and matching of pictorial structures.
IEEE Trans. Comput., 22(1):67–92, Jan. 1973.

A. Hanson and E. Riseman.
Visions: A computer vision system for interpreting scenes.
Computer Vision Systems., pages 303–334, 1978.

Y. Jin and S. Geman.
Context and hierarchy in a probabilistic image model.
2013 IEEE Conference on Computer Vision and Pattern Recognition, 2:2145–2152, 2006.

J. Porway, K. Wang, and S. C. Zhu.
A hierarchical and contextual model for aerial image understanding.
Int'l Journal of Computer Vision, 88(2):254–283, 2010.

A. Torralba, K. P. Murphy, and W. T. Freeman.
Using the forest to see the trees: exploiting context for visual object detection and localization.
Commun. ACM, 53(3):107–114, Mar. 2010.