Scene Interpretation by Entropy Pursuit

Donald Geman
Johns Hopkins University

MIA'16
Institut Henri Poincare
Paris, January 18-20, 2015

COLLABORATORS

- Ehsan Jahangiri (PhD student, JHU)
- Erdem Yoruk (former PhD student, JHU)
- Laurent Younes
- Rene Vidal

OUTLINE

- Scene Interpretation
- Matched Bayesian Model
- Entropy Pursuit
- Application to Table Settings

Machines vs. Humans

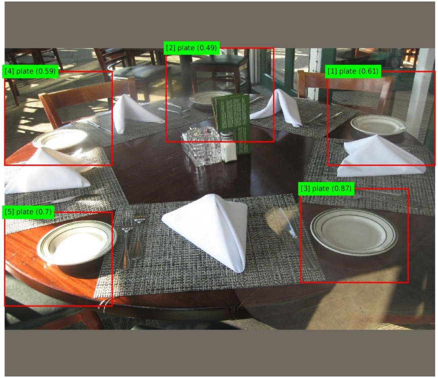
- Interpreting scenes is effortless and instantaneous for people, even generating rich semantic annotations ("telling a story").
- Machines lag very far behind in understanding images, and building a *description machine* remains a fundamental A.I. challenge.
- This remains true even for the restricted task of detecting and localizing all instances from a set of object categories.

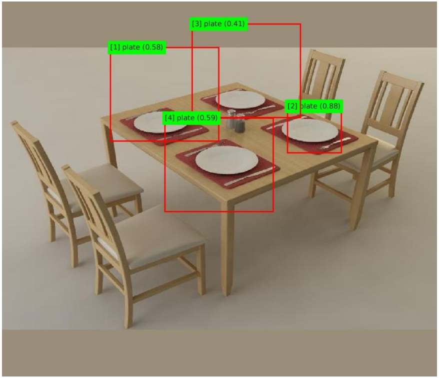
STREET SCENES

Table Scenes

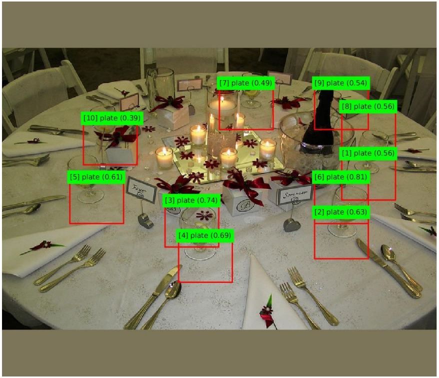


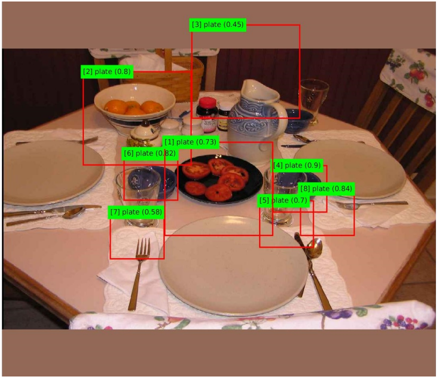

PERFECT "PLATE" DETECTIONS BY CNNS



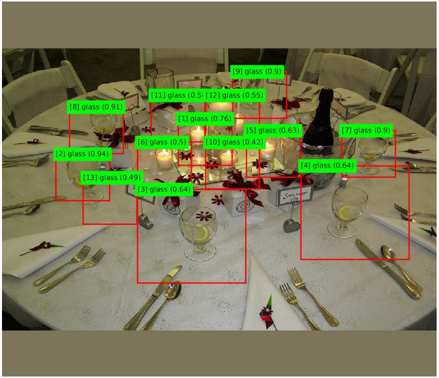


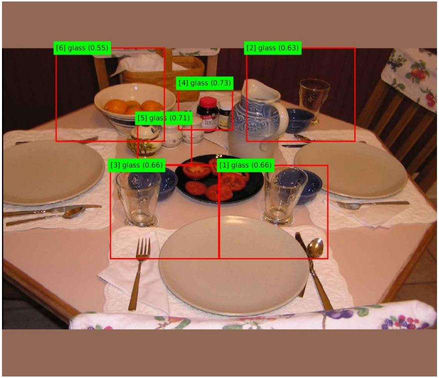
Poor "Plate" Detections by CNNs



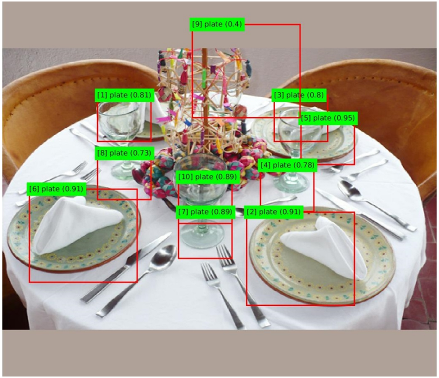


“GLASS” DETECTIONS BY CNNS





CONTEXTUALLY INCONSISTENT DETECTIONS



OUTLINE

- Scene Interpretation
- Matched Bayesian Model
- Entropy Pursuit
- Application to Table Settings

MATCHED BAYESIAN

- Combine discriminative (parsing by scanning with trained classifiers) and model-based (identifying likely interpretations under the posterior) approaches.
- Replace the usual features in Bayesian data models with high-level classifiers; define latent variables (almost) one-to-one correspondence with classifiers.
- In particular, no low-level or mid-level features in the model; all variables have semantic content.
- The prior model encodes knowledge about relative sizes and likely configurations (spatial context).
- The posterior distribution modulates or *contextualizes* raw classifier output.

SEQUENTIAL BAYESIAN

- Model construction also motivated by efficient search and evidence integraion.
- Scene annotation is procedural, inspired by divide-and-conquer querying and selective attention.
- Computational efficiency by prioritizing what to do next - a *process of discovery*.
- Prioritization by *entropy pursuit*.
- Processing can be terminated at any point, ideally when the posterior is peaked.

Scenes and Images

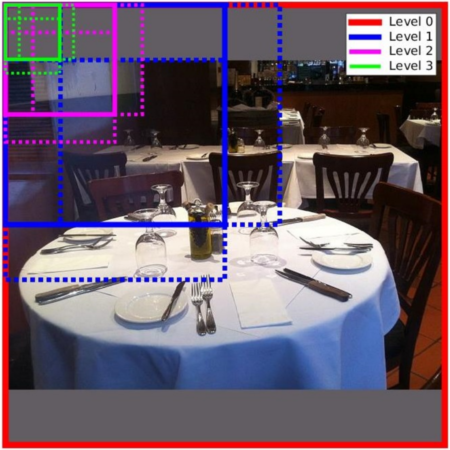
- $\mathcal{C}$: object categories of interest.
- ω: 3D scene, described by instances from $\mathcal{C}$ and their 3D poses.
- H: scene-to-image transformation.
- $I = I(\omega, H)$: image over an image domain $\mathcal{L}$ for FOV of the camera
- $p \in \mathcal{P}$: pose space in image coordinates.
- $\{(c_k, p_k), k = 1, \ldots, N\}$: image description, where N is random.

ANNOCELLS AND ANNOBITS

- $\mathcal{A}$: hierarchy of image patches (sub-windows) $W \subset \mathcal{L}$.
- Y_A: "*What is going on in* A?" for $A \in \mathcal{A}$. For example, for $\mathcal{C} = \{plate, bottle, glass, utensil\}$, which categories have instances fully inside A?
- More generally, yes-no questions ("annobits") about subsets of $\mathcal{C}$ and subsets of $\mathcal{P}$ ("pose cells"), e.g.,
  - *"Is there a plate in W?"*
  - *"Is there a bottle or glass centered in W in the scale range* $[s, S]$*?"*
- Y_A corresponds to $|\mathcal{C}|$ such annobits with $2^{|\mathcal{C}|}$ possible values.

ANNOCELL HIERARCHY

- A partitioning of the input image at different levels of spatial resolution.



PRIOR MODELS

- $P(\omega)$: 3D scene model.
- $P(H)$: distribution on homographies.
- $\mathbf{Y} = Y_A, A \in \mathcal{A}$
- $P(\omega, H, \mathbf{y}) = P(\omega)P(H)\delta(\mathbf{y} = \mathbf{y}(\omega, H))$.

Data Model

- X_A: classifier to predict Y_A.
- In practice, X_A assumes $|\mathcal{C}| + 1$ values, not $2^{|\mathcal{C}|}$.
- $\mathbf{X} = X_A, A \in \mathcal{A}$
- $P(\mathbf{x} \mid \mathbf{y})$: conditional distribution of classifiers
- Will assume conditional independence:

$$P(\mathbf{x} \mid \mathbf{y}) = \prod_{A \in \mathcal{A}} P_A(x_A \mid \mathbf{y})$$

and

$$P_A(x_A \mid \mathbf{y}) = P_A(x_A \mid y_A).$$

OUTLINE

- Scene Interpretation
- Matched Bayesian Model
- Entropy Pursuit
- Application to Table Settings

SEQUENTIAL TESTING STRATEGY

- We will collect evidence by asking questions sequentially and adaptively.
- $\mathbf{q}_t = \{q_1, ..., q_t\} \subset \mathcal{A}$: annocells previously processed
- $\mathbf{x}_{\mathbf{q}_t} = \{X_{q_1}(I), ..., X_{q_t}(I)\}$: corresponding classifier results
- $\mathbf{e}_t = (\mathbf{q}_t, \mathbf{x}_{\mathbf{q}_t})$: evidence acquired from I after t classifiers
- *Entropy Pursuit:*

$$q_{t+1} = \arg\min_{A \in \mathcal{A}} H(\mathbf{Y} | \mathbf{e}_t, X_A).$$

- *Key Assumption: All classifiers have unit cost.*

More Precisely

- $\mathcal{A}_t(I) \subset \mathcal{A}$: the annocells previously processed. This is a random subset depending on I, the image being processed.
- $\mathbf{e}_t(I) = \{X_A = X_A(I), A \in \mathcal{A}_t(I)\}$: history as an event, that is, $\mathbf{e}_t(I)$ is the set of images with X_A values identical to those for image I for each $A \in \mathcal{A}_t(I)$.
- $\mathcal{A}_{t+1}(I) = \{A\} \cup \mathcal{A}_t(I)$, where

$$A = \arg\min_{A \in \mathcal{A}} H(\mathbf{Y}|\mathbf{e}_t(I), X_A).$$

APPROXIMATION

- Replace

$$q_{t+1} = \arg\min_{A \in \mathcal{A}} H(\mathbf{Y}|\mathbf{e}_t, X_A)$$

by

$$q_{t+1} = \arg\min_{A \in \mathcal{A}} H(\mathbf{Y}|\mathbf{e}_t, Y_A).$$

- It then follows that

$$q_{t+1} = \arg\max_{A \in \mathcal{A}} H(Y_A|\mathbf{e}_t)$$

.

- Hence, “pursue” highly uncertain annocells under the current posterior.

GREAT EXPECTATIONS

- *Does coarse-to-fine search emerge naturally from EP?*
- *Are ambiguities due to conflicting evidence resolved?*
- *Can a fraction of the classifiers do as well as all of them?*

Outline

- Scene Interpretation
- Matched Bayesian Model
- Entropy Pursuit
- Application to Table Settings

JHU Table-setting Dataset

PRIOR MODEL ON TABLE SETTINGS

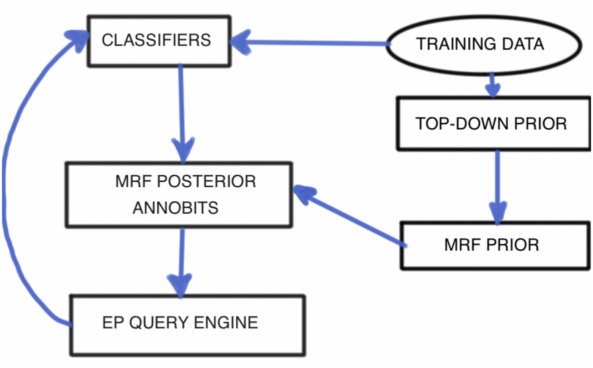
- T: Table dimensions (geometry).
- ω: $|\mathcal{C}|$ binary variables for each 5cm $\times$ 5cm table cell indicating the presence of at least one instance from the corresponding category.
- $P(\omega|T)$: 3D scene model (Gibbs distribution) on the table.
- $P(H)$: distribution on homographies.
- $\mathbf{Y} = Y_A, A \in \mathcal{A}$ determined by ω, H.
- $P(\omega, H, T) = P(H)P(T)P(\omega|T)$ where:

$$p_\lambda(\omega|T) = \frac{1}{Z(\lambda)} \exp\big(\lambda.\mathbf{f}(\omega)\big).$$

Actually Two Prior Scene Models

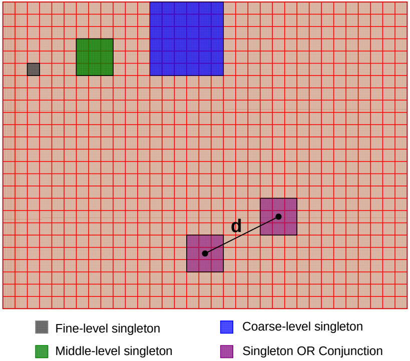
- *First:* A generative attributed graph (GAG) prior model in the world coordinate system (skipped).
- The GAG model has interpretable parameters and was efficiently learned from limited number of annotated images.
- But: conditional inference is slow.
- *Second:* The MRF *whose parameters are learned from GAG model samples.*

OVERVIEW



- To estimate $p(Y_A|\mathbf{e}_t)$, posterior model samples are projected to the image coordinate system via perspective projection and the interpretation units are aggregated.

MRF FEATURES



- The singleton features accommodate the overall empirical statistics for localized object instances.
- The conjunction feature functions incorporate contextual relations between different object categories.

MRF Learning (Skipping Details)

- We exploit symmetry in table-settings to reduce the number of parameters.
- We learned 10 MRF models $P(\omega|T)$ for 10 different table sizes using stochastic gradient descent, iteratively minimizing the KL divergence between the Gibbs and empirical distribution.

POSTERIOR SAMPLING

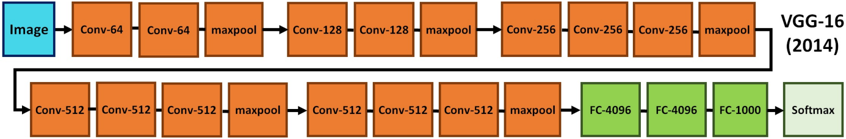
- Posterior sampling was carried out in three nested loops corresponding to factoring the posterior at step t:

$$P(\omega, T, H|\mathbf{e}_t) = P(T|\mathbf{e}_t)P(H|T, \mathbf{e}_t)P(\omega|T, H, \mathbf{e}_t).$$

  - Outer Loop: sampling table size (Metropolis-Hastings)
  - Middle Loop: sampling homography (Metropolis-Hastings)
  - Inner Loop: sampling MRF model (Gibbs sampling)
- Given posterior samples of (ω, H), directly obtain posterior samples of $\mathbf{Y}$, and hence can estimate $H(Y_A|\mathbf{e}_t)$ for all A.

CNN Classifiers

- We trained (the last layers of) 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,
  - TableNet: to detect the table surface area in a given image.



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.
- Reducing the $2^4 = 16$ possible states of a patch to only 5, whereas crude, does scale linearly with the number of categories (rather than exponential $2^{|\mathcal{C}|}$).

CatNet Training

- A patch including multiple object instances appears multiple times in the training set, each time with the category label of one of the existing instances.
- The CatNet was trained by minimizing the cross-entropy loss function using stochastic gradient descent.
- Training took about 24 hours when the first 15 weight layers were initializing by the first 15 weight layers from the VGG-16 network.

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:

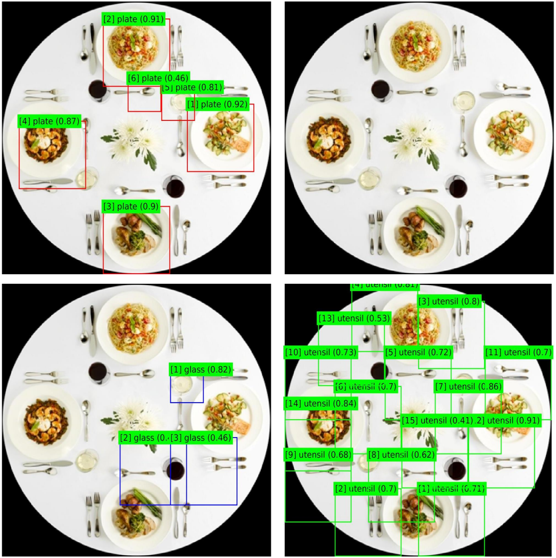
$$(s_1 = 0.05, s_2 = 0.45, s_3 = 0.05, s_4 = 0.1, s_5 = 0.35)$$

  Then categories "2" and "5" are declared as positive detections.

ScaleNet (In Brief)

- ScaleNet estimates the ratio of an object's scale (in pixels) to the size of the input patch, which stays unchanged after resizing the original input to 224×224.
- For an object that is fully inside a patch the scale ratio is within the range $(0, 1]$.
- We declare an annocell patch as a positive detection (bounding box) for category c if both $S_{\text{scale}} \geq 0.5$ and c is detected.

CNN DETECTION EXAMPLES



CNN DETECTION EXAMPLES

CNN DETECTION EXAMPLES

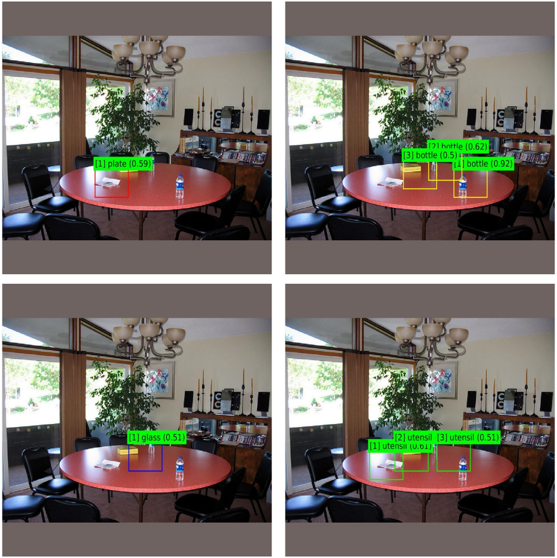
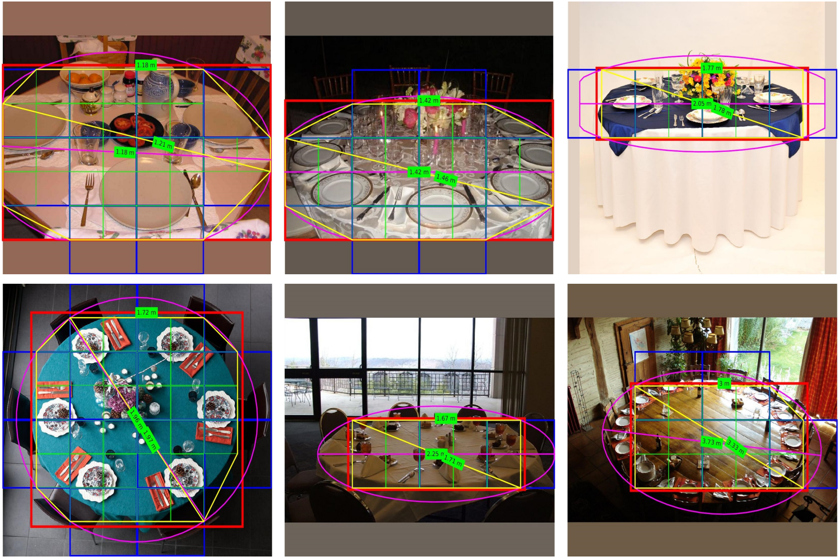


Table Detection by TableNet



Dirichlet Data Model

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

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

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

RECALL

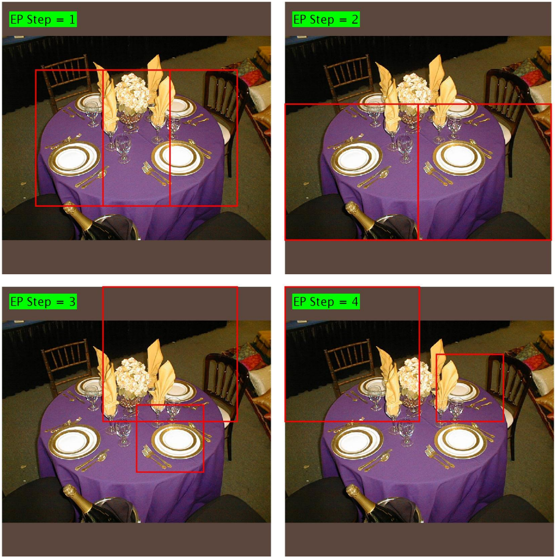
- Y_A: "*What is going on in A*?" for $A \in \mathcal{A}$.
- $P(\omega, H, T) = P(H)P(T)P(\omega|T)$.
- X_A: CNN to predict Y_A.
-
$$P(\mathbf{x} \mid \mathbf{y}) = \prod_{A \in \mathcal{A}} P_A(x_A \mid y_A).$$
- $\mathbf{e}_t = (\mathbf{q}_t, \mathbf{x}_{\mathbf{q}_t})$: evidence acquired from I after t annocells processed with both CatNet and ScaleNet.
- Next annocell examined is
$$q_{t+1} = \arg\max_{A \in \mathcal{A}} H(Y_A|\mathbf{e}_t)$$
.

FULL POSTERIOR DETECTIONS

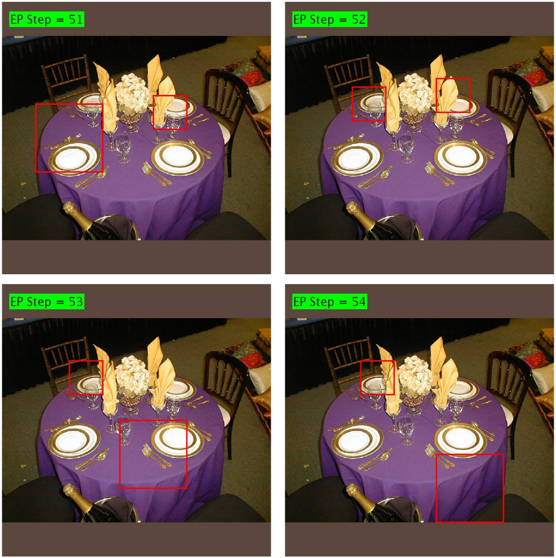
EP DETECTIONS (STEP 40)

CNN DETECTIONS

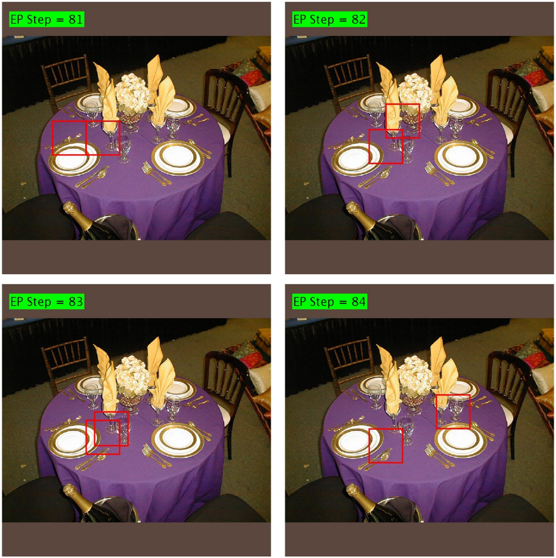
EP QUESTIONS (STEPS 1-4)



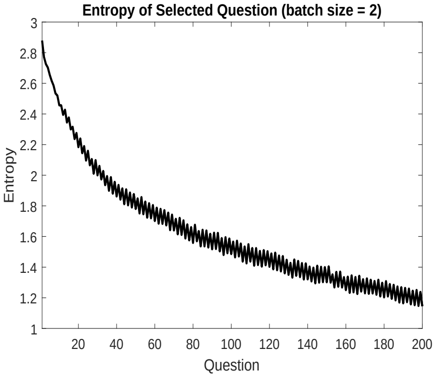
EP QUESTIONS (STEPS 51-54)

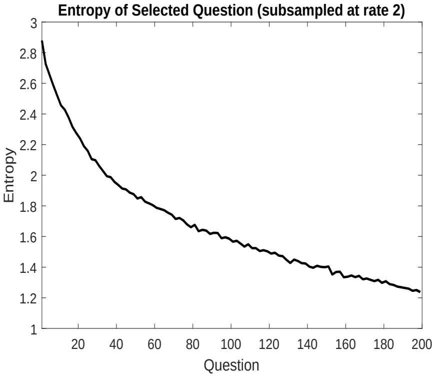


EP QUESTIONS (STEPS 81-84)

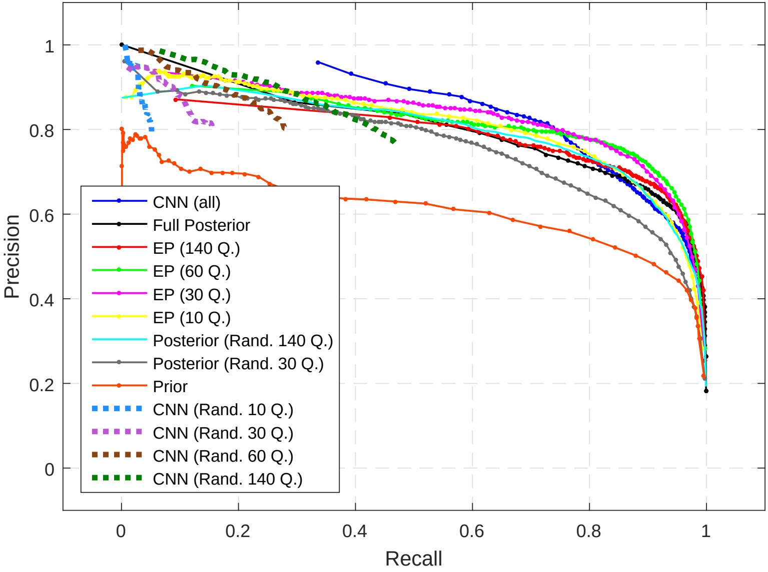


Entropy of EP Questions

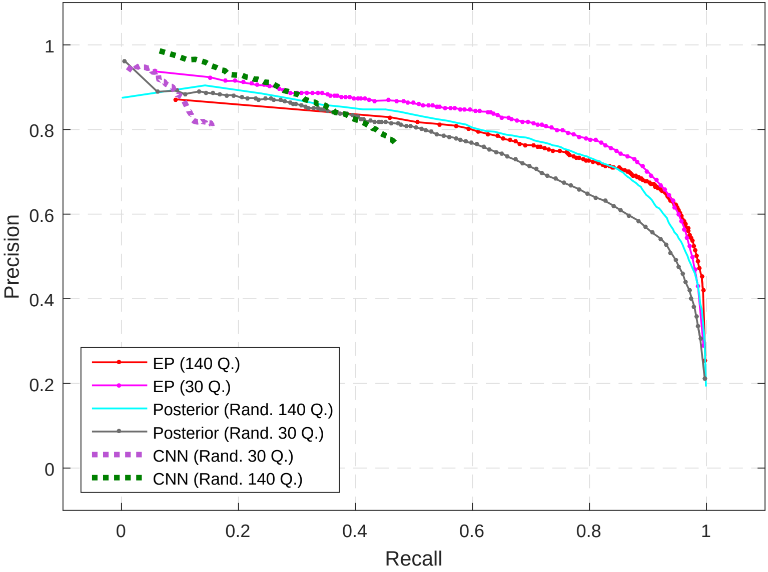




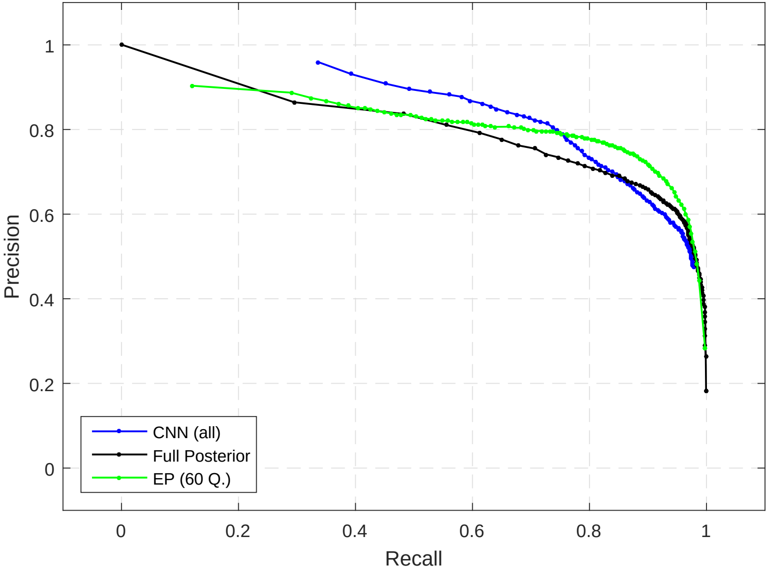
PRECISION-RECALL CURVES



PRECISION-RECALL CURVES



Precision-Recall Curves



Concluding Remarks

- Some ad hoc aspects and lots to integrate.
- Many improvements are possible, e.g., better integration of scale and table prediction into the matched Bayesian framework.
- Also, dropping the “oracle approximation” in EP deserves investigation.
- But does serve as a *proof of concept*.