Active Learning for Structured Probabilistic Models with Histogram Approximation

Qing Sun
Virginia Tech
sunqing@vt.edu

Ankit Laddha
CMU
aladdha@andrew.cmu.edu

Dhruv Batra
Virginia Tech
dbatra@vt.edu

Abstract

This paper studies active learning in structured probabilistic models such as Conditional Random Fields (CRFs). This is a challenging problem because unlike unstructured prediction problems such as binary or multi-class classification, structured prediction problems involve a distribution with an exponentially-large support, for instance, over the space of all possible segmentations of an image. Thus, the entropy of such models is typically intractable to compute. We propose a crude yet surprisingly effective histogram approximation *to the Gibbs distribution, which replaces the exponentially-large support with a* coarsened *distribution that may be viewed as a histogram over M bins. We show that our approach outperforms a number of baselines and results in a 90%-reduction in the number of annotations needed to achieve nearly the same accuracy as learning from the entire dataset.*

1. Introduction

A number of problems in Computer Vision – image segmentation, geometric labeling, human body pose estimation – can be written as a mapping from an input image $\mathbf{x} \in \mathcal{X}$ to an exponentially large space $\mathcal{Y}$ of *structured outputs*. For instance, in semantic segmentation, $\mathcal{Y}$ is the space of all possible (super-)pixel labelings, $|\mathcal{Y}| = L^n$, where n is the number of (super-)pixels and L is the number of object labels that each (super-)pixel can take.

As a number of empirical studies have found [1–3], the amount of training data is one of the most significant factors influencing the performance of a vision system. Unfortunately, unlike *unstructured* prediction problems – binary or multi-class classification – data annotation is a particularly expensive activity for structured prediction. For instance, in image segmentation annotations, we must label every (super-)pixel in every training image, which may easily run into millions. In pose estimation annotations, we must label 2D/3D locations of all body parts and keypoints of interest in thousands of images. As a result, modern dataset collection efforts such as PASCAL VOC [4], ImageNet [5], and MS COCO [6] typically involve spending thousands of human-hours and dollars on crowdsourcing websites such as Amazon Mechanical Turk.

Active learning [7] is a natural candidate for reducing annotation efforts by seeking labels only on the most *informative* images, rather than the annotator passively labeling all images, many of which may be uninformative. Unfortunately, active learning for structured-output models is challenging. Perhaps even the simplest definition of "informative" involves computing the entropy of the learnt model over the output space:

$$H(\mathrm{P}) = -\mathbb{E}_{\mathrm{P}(\mathbf{y}|\mathbf{x})}[\log(\mathrm{P}(\mathbf{y}|\mathbf{x}))] \tag{1a}$$

$$= -\sum_{\mathbf{y}\in\mathcal{Y}} \mathrm{P}(\mathbf{y}|\mathbf{x}) \log \mathrm{P}(\mathbf{y}|\mathbf{x}), \tag{1b}$$

which is intractable due to the summation over an exponentially-large output space $\mathcal{Y}$.

Overview and Contributions. In this paper, we study active learning for probabilistic models such as Conditional Random Fields (CRFs) that encode probability distributions over an exponentially-large structured output space.

Our main technical contribution is a variational approach [8] for approximate entropy computation in such models. Specifically, we present a crude yet surprisingly effective *histogram approximation* to the Gibbs distribution, which replaces the exponentially-large support with a *coarsened* distribution that may be viewed a histogram over M bins. As illustrated in Fig. 1, each bin in the histogram corresponds to a subset of solutions – for instance, all segmentations where size of foreground (number of ON pixels) is in a specific range $[L\ U]$. Computing the entropy of this coarse distribution is simple since M is a small constant ($\sim$10). Importantly, we prove that the *optimal histogram*, *i.e.* one that minimizes the KL-divergence to the Gibbs distribution, is composed of the mass of the Gibbs distribution in each bin, *i.e.* $\sum_{\mathbf{y}\in\text{bin}} P(\mathbf{y}|\mathbf{x})$. Unfortunately, the problem of estimating sums of the Gibbs distribution under general hamming-ball constraints continues to be #P-

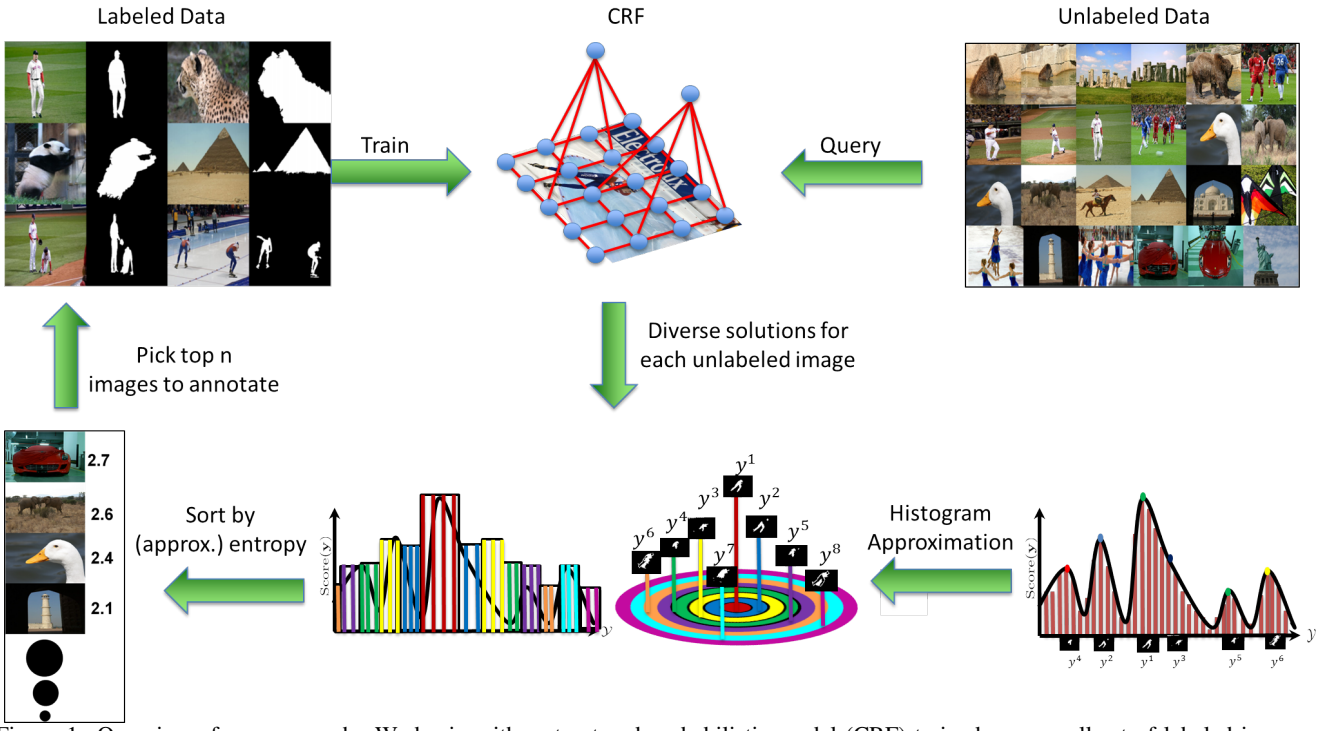


Figure 1: Overview of our approach. We begin with a structured probabilistic model (CRF) trained on a small set of labeled images; then search the large unlabeled pool for a set of informative images to annotate where our current model is most uncertain, *i.e.* has highest entropy. Since computing the exact entropy is NP-hard for loopy models, we approximate the Gibbs distribution with a *coarsened histogram* over M bins. The bins we use are 'circular rings' of varying hamming-ball radii around the highest scoring solution. This leads to a novel variational approximation of entropy in structured models, and an efficient active learning algorithm.

complete [9]. Thus, we upper bound the mass of the distribution in a bin with the maximum entry in a bin multiplied by the size of the bin. Fortunately, finding the most probable configuration in a hamming ball has been recently studied in the graphical models literature [10–12], and efficient algorithms have been developed, which we use in this work.

We perform experiments on figure-ground image segmentation and coarse 3D geometric labeling [13]. Our proposed algorithm significantly outperforms a large number of baselines and can help save hours of human annotation effort.

2. Related Work

Large-scale data annotation efforts in computer vision have typically involved thousands of hours of human effort, either for pay (Mechanical Turk) or motivated by the task being fun [14] or a game [15].

Learning from weak annotations. One theme in reducing annotation effort in recognition tasks is to learn from weak annotations – where the annotation provides only the name of the object in the image [16–18], or partial labelings where the annotations for some pixels are missing [19, 20], or learning in an interactive setting where the annotator repeatedly provides scribbles [21–24], or propagating labels from annotated images to unannotated images [25]. In contrast, we focus on the fully-supervised active learning setting where the goal is to identify *which* images to label; once an image is chosen, we receive full annotations.

Active learning is a vast sub-field of machine learning, with a number of approaches for quantifying the informativeness of an as yet unlabeled example – based on disagreement among a committee of classifiers [26], version space of an SVM [27], and expected informativeness [28] for probabilistic models. We focus on entropy-based active learning, which is a natural definition of informativeness, but is intractable to compute for structured models such as CRFs.

In computer vision, active learning has been used for scene classification [29], object/image categorization [30, 31], and annotating large image and video datasets [32–36]. Notice that these are all instances of *unstructured* prediction – binary or multi-class classification. We focus on structured prediction where the space of possible outcomes and thus the support of the distribution of our model is exponentially large. Two previous works address the problem of exact computation of entropy in such models [37, 38]. However, both these works assume chain/tree-structured graphical models, which is understandable in natural language processing, but is an unreasonable assumption for computer vision problems. We make no such assumptions.

The closest to our goal is the recent work of Luo *et al*. [39], on active learning in latent structured models. There are a number of subtle but important differences w.r.t. to our work. The algorithm in [39] estimates the *local entropy* of the marginal distribution of each variable via convex belief propagation [40], and asks the user to annotate the single variable/pixel that is most marginally uncertain. In comparison, the focus of our work is to estimate the entropy of the joint distribution not the entropy of the marginal. Thus, we are able to find an *image* where the model is most uncertain, rather than a pixel. This matches the natural annotation modality, where annotators are shown full images and asked to provide polygonal annotations [5, 6, 14]. In our experiments, we compare against an adapted version of the algorithm from [39], which estimates the entropy of the joint distribution by summing entropies of the marginals (thereby assuming that pixels are independent). To its credit, [39] studies a more general setting (learning under partial supervision) than the one studied in this paper (full supervision). However, within this narrower but important domain, we find that our approach outperforms all baselines, including our adaption of [39].

The work of Maji *et al*. [41] defines a novel uncertainty measure for structured models, called 'MAP perturbation uncertainty,' which upper-bounds the true entropy of the Gibbs distribution via MAP perturbations [42, 43] under Gumbel noise. Note that for entropy-based active learning, ideally we require *lower bounds* on entropy, not upper bounds. Inspired by the MAP perturbation literature [42, 43], we compare to (and outperform) a baseline that directly approximates the entropy by treating the MAP perturbation solutions as samples.

Finally, a number of recent works have looked into active learning with multiple modalities of annotator feedback and rich learner-supervisor interactions beyond simply asking for class-labels [44–46]. Combining such rich interactions with our approach is an interesting direction for future.

3. Preliminaries and Notation

We begin by establishing the notation used in the paper.

Notation. For any positive integer n, let $[n]$ be shorthand for the set $\{1, 2, \ldots, n\}$. Given an input image $\mathbf{x} \in \mathcal{X}$, our goal is to make a prediction about $\mathbf{y} \in \mathcal{Y}$, where $\mathbf{y}$ may be a figure-ground segmentation, or a category-level semantic segmentation. Specifically, let $\mathbf{y} = \{y_1 \ldots y_n\}$ be a set of discrete random variables, each taking value in a finite label set, $y_u \in Y_u$. In semantic segmentation, u indexes over the (super-)pixels in the image, and these variables are the labels assigned to each (super-)pixel, *i.e.* $y_u \in Y_u = \{\text{sky, building, road, car, } \ldots\}$.

CRF Model. Let $G = (\mathcal{V}, \mathcal{E})$ be a graph defined over the output variables $\mathbf{y}$, *i.e.* $\mathcal{V} = [n]$, $\mathcal{E} \subseteq \binom{[n]}{2}$. Let $\theta_u(y_u)$ be the unary term expressing the local confidence at site u for the label y_u, and $\theta_{uv}(y_u, y_v)$ be the pairwise term expressing compatibility of label y_u and y_v at adjacent vertices. The *score* for any configuration $\mathbf{y}$ is given by the sum $S(\mathbf{y}) = \sum_{u \in \mathcal{V}} \theta_u(y_u) + \sum_{(u,v) \in \mathcal{E}} \theta_{uv}(y_u, y_v)$, and its probability is given by the Gibbs distribution: $P(\mathbf{y}|\mathbf{x}) = \frac{1}{\mathcal{Z}} e^{S(\mathbf{y})}$, where $\mathcal{Z}$ is the partition function or normalization constant. The techniques proposed in this paper are naturally applicable to higher-order CRFs. However, to simplify the exposition we only consider pairwise energies.

These unary and pairwise terms are derived from a weighted combination of features extracted at vertices and edges, *i.e.*, $\theta_u(y_u) = w_u^\intercal \phi(\mathbf{x}, y_u)$ and $\theta_{uv}(y_u, y_v) = w_{uv}^\intercal \phi(\mathbf{x}, y_u, y_v)$. Thus, this is a log-linear model, with $S(\mathbf{y}) = \mathbf{w}^\intercal \boldsymbol{\phi}(\mathbf{x}, \mathbf{y})$, where $\mathbf{w}$ are all the model parameters concatenated into a long vector, and $\boldsymbol{\phi}(\mathbf{x}, \mathbf{y})$ are all the features concatenated.

4. Approach: Approximate Entropy for Gibbs

We now describe our proposed active learning approach. We begin with a small number of labelled images from which an initial estimate of $\mathbf{w}$ is trained. Given a pool of unlabeled images, our goal is to find and seek annotation for the image where our current model is most uncertain.

Exact Entropy. For each unlabeled image $\mathbf{x}$, we need to compute the entropy of the conditional distribution $\mathrm{P}(\mathbf{y}|\mathbf{x})$:

$$H(\mathrm{P}) = -\mathbb{E}_{\mathrm{P}(\mathbf{y}|\mathbf{x})}[\log(\mathrm{P}(\mathbf{y}|\mathbf{x}))] \tag{2a}$$

$$= -\sum_{\mathbf{y} \in \mathcal{Y}} \mathrm{P}(\mathbf{y}|\mathbf{x}) \log \mathrm{P}(\mathbf{y}|\mathbf{x}) \tag{2b}$$

Computing this entropy exactly is intractable due to the summation over an exponentially-large output space $\mathcal{Y}$.

Variational Inference for Approximate Entropy. At a high level, the goal of any variational method is to construct a surrogate distribution $Q(\mathbf{y})$, and measure its entropy as an approximation to the entropy of $\mathrm{P}(\mathbf{y}|\mathbf{x})$. There are two desiderata for constructing a good surrogate:

- **Efficiency:** We should be able to quickly construct the surrogate distribution $Q(\mathbf{y})$ and compute its entropy since this computation needs to be repeatedly performed as the model learns, and the unlabeled pool of images may be very large. Thus, $Q(\mathbf{y})$ should be *compact*, and allow computation of entropy in a small number of (say $O(M)$) operations:

$$H(Q) = -\sum_{m=1}^{M} Q(\mathbf{y}^m) \log Q(\mathbf{y}^m) \tag{3}$$

- **Approximation Quality:** The surrogate $Q(\mathbf{y})$ should faithfully approximate the Gibbs distribution $\mathrm{P}(\mathbf{y}|\mathbf{x})$ and lead to an accurate entropy approximation, even for high level of compactness.

In the next few subsections, we look at a few different notations of compactness – first considering a standard technique and then proposing our own notion of compactness.

4.1. Surrogate with Stochastic Samples

Classical techniques such as Monte Carlo methods for numerically approximating integrals involve replacing the exponentially large summation with a finite sum over a small set of M solutions $\mathbf{Y} = \{\mathbf{y}^1, \mathbf{y}^2, \ldots, \mathbf{y}^M\}$, which corresponds to *sum-of-weighted-delta (SOWD) approximation* to the Gibbs distribution:

$$H(P) \approx -\sum_{m=1}^{M} \frac{e^{S(\mathbf{y}^m)}}{\mathcal{Z}_\delta} \log \frac{e^{S(\mathbf{y}^m)}}{\mathcal{Z}_\delta}, \tag{4}$$

where $\mathcal{Z}_\delta = \sum_{i=1}^{M} e^{S(\mathbf{y}^i)}$ is the normalizing constant of the delta-approximation.

Broadly speaking, there are two main families of methods for constructing $\mathbf{Y}$:

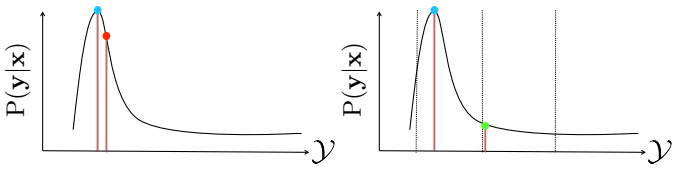
- **Classical Monte Carlo:** where $\mathbf{y}^i$ are samples from the distribution $\mathrm{P}(\mathbf{y}|\mathbf{x})$. Since direct sampling from undirected graphical models is hard [47], typically Markov Chain Monte Carlo (MCMC) methods such as Gibbs sampling are used, which sample from a Markov Chain whose stationary distribution is $\mathrm{P}(\mathbf{y}|\mathbf{x})$.
- **Quasi Monte Carlo:** where $\mathbf{y}^i$ are stochastic *low-discrepancy* points that try to cover the space as uniformly as possible without creating any regions with high or low density.

However, both these methods fall short. MCMC sampling based methods are slow, often requiring a long *burn-in* period before the Markov Chain converges to the stationary distribution, and even after that a large number of samples may be needed before they transition out of one mode of the distribution $\mathrm{P}(\mathbf{y}|\mathbf{x})$ to another mode. Since our goal is to estimate entropy, it is crucial that we see samples from as many modes of the distribution as possible. In our experiments, we compare to Gibbs sampling and confirm that it performs poorly. On the other hand, Quasi Monte Carlo methods completely ignore the function being summed, and may end up summing terms with insignificant effect, especially if the distribution $\mathrm{P}(\mathbf{y}|\mathbf{x})$ is non-uniform.

4.2. Surrogate with Deterministic Samples

Instead of running a long Markov Chain to convergence, can we efficiently find *deterministic* samples that are representative of $\mathrm{P}(\mathbf{y}|\mathbf{x})$? Specifically, can we construct a surrogate $Q(\mathbf{y})$ with support on exactly M solutions $\mathbf{Y} = \{\mathbf{y}^1, \mathbf{y}^2, \ldots, \mathbf{y}^M\}$ that *optimally* approximates $\mathrm{P}(\mathbf{y}|\mathbf{x})$?

Let $Q(\mathbf{y}) = \sum_{m=1}^{M} q_m [\![\mathbf{y} = \mathbf{y}^m]\!]$, where $[\![\cdot]\!]$ is the Iverson bracket, which is 1 when the input argument is true, and 0 otherwise. Thus, $Q(\mathbf{y})$ is a sum of weighted delta functions. This surrogate is parameterized by (i) the location of the support $\mathbf{Y}$, and (ii) the weights $\mathbf{q} = \{q_1, \ldots, q_m\}$ that must clearly sum to 1. Lemma 1 shows that optimal support location and weights correspond to the top M highest scoring configurations in P.



(a) Delta approximation. (b) Histogram approximation.

Figure 2: Delta vs Histogram Approximation.

Lemma 1. *Let* $Q(\mathbf{y}; \mathbf{Y}, \mathbf{q}) = \sum_{m=1}^{M} q_m [\![\mathbf{y} = \mathbf{y}^m]\!]$ *be a SOWD-approximation parameterized by* $\mathbf{Y}$ *and* $\mathbf{q}$*. Let* $KL(Q||\mathrm{P})$[1] $= \sum_{\mathbf{y}\in\mathcal{Y}} Q(\mathbf{y}) \log \frac{Q(\mathbf{y})}{\mathrm{P}(\mathbf{y}|\mathbf{x})}$ *denote the KL-divergence between the two distributions. The parameters* $\hat{\mathbf{Y}}, \hat{\mathbf{q}}$ *that minimize* $KL(Q||\mathrm{P})$ *are:*

$$\hat{\mathbf{y}}^m = \operatorname*{argmax}_{\mathbf{y}\in\mathcal{Y}} \ \mathrm{P}(\mathbf{y}|\mathbf{x}) \tag{5a}$$

$$s.t. \ \mathbf{y} \neq \hat{\mathbf{y}}^{m'} \quad \forall m' < m \tag{5b}$$

$$\hat{q}_m = \frac{e^{S(\hat{\mathbf{y}}^m)}}{\sum_{m'=1}^{M} e^{S(\hat{\mathbf{y}}^{m'})}} \tag{6}$$

Proof. Using the method of Lagrangian multipliers and solving the system of partial derivates of the Lagrangian. More details in the supplement. □

Lemma 1 matches what we would intuitively expect – that if we need to approximate P with a set of points, these should be placed at the top M most probable locations in P. Eqn. (5) corresponds to a problem known in the graphical models literature as the M-Best MAP [48–50]. Eqn. (6) corresponds to normalizing the unnormalized probabilities of the M-Best MAP points to form a valid distribution.

Unfortunately, while this delta-surrogate may be intuitive, it suffer from some pathological behaviors. One such counter-intuitive behavior is illustrated in Fig. 2. Consider $M = 2$. Lemma 1 asks us to pick the two points shown in Fig. 2a since they have the highest probability under P. The scores of these two configurations are very similar and thus Q seems nearly uniform, with $H(Q) \simeq \log_2 2 = 1$. However, P is extremely peaky, and this will lead to wasted effort in annotating this image. The reason for this discrepence is that $\min_Q KL(Q||\mathrm{P})$ attempts to approximate the entire distribution P, while our primary interest is in approximating the entropy. Even if we changed the objective function, there is a second pathology. Recall that the entropy of any discrete distribution with support on M points is upper-bounded by $\log_2 M$ bits. This is a *significantly* smaller quantity than $\log_2 |\mathcal{Y}|$. For instance, in a

[1] We work with $KL(Q||\mathrm{P})$ because $KL(\mathrm{P}||Q)$ is not defined when Q has more restrictive support than P.

binary image segmentation problem, the size of the state space is $|\mathcal{Y}| = 2^n$, and the maximum entropy possible is $\log_2 2^n = n$ bits. Here n is the number of super-pixels and typically around $200-2000$, while the number of points M is typically around 10 ($\log_2 10 = 3.29$). While we do not expect distributions over real image segmentation instances to be nearly uniform, it is clear that as soon as the entropy of P becomes larger than 3.29 bits, any uniform distribution Q that places support on *any* M points is an *optimal* entropy approximator. Clearly, such an approximation cannot be used to perform active learning.

4.3. Surrogate with Histogram Bins

Based on these intuitions, we propose a different notion of compactness of Q – one that still requires the same number of M parameters to represent Q, but is more representative of P globally. As illustrated in Fig. 2b, we partition $\mathcal{Y}$ into M non-overlapping bins and make Q a normalized histogram over these bins. Specifically, let $\{\bar{\mathbf{y}}^1, \bar{\mathbf{y}}^2, \ldots, \bar{\mathbf{y}}^M\}$ denote the bin centers, $\Delta(\mathbf{y}^1, \mathbf{y}^2)$ denote the Hamming distance between $\mathbf{y}^1$ and $\mathbf{y}^2$, and $\mathcal{Y}^m = \{\mathbf{y} \mid \Delta(\mathbf{y}, \bar{\mathbf{y}}^m) \leq r\}$ be the set of configurations that lie within bin m, which is to say that they lie within an appropriately defined r-radius distance ball of $\bar{\mathbf{y}}^m$. With this notation, we define the surrogate Q to be:

$$Q(\mathbf{y}) = \sum_{m=1}^{M} q_m [\![\mathbf{y} \in \mathcal{Y}^m]\!] = \sum_{m=1}^{M} q_m [\![\Delta(\mathbf{y}, \bar{\mathbf{y}}^m) \leq r]\!] \tag{7}$$

i.e. if $\mathbf{y}$ lies in the the m^{th} bin, it is assigned a probability of q_m. Note that this formulation can contain the delta-approximation as a special case with $r = 0$, if we allow for the bins to not be a complete cover of $\mathcal{Y}$ (*i.e.* $\mathcal{Y} \neq \cup_m \mathcal{Y}^m$).

We can now ask a similar question as in the previous section – what is the *optimal* histogram approximation?

Lemma 2. *Let* $Q(\mathbf{y}; \{\mathcal{Y}^m\}, \mathbf{q}) = \sum_{m=1}^{M} \frac{q_m}{|\mathcal{Y}^m|} [\![\mathbf{y} \in \mathcal{Y}^m]\!]$ *be a histogram-approximation parameterized by bins* $\{\mathcal{Y}^m\}$ *and weights* $\mathbf{q}$. *Let* $KL(\mathrm{P}||Q) = \sum_{\mathbf{y} \in \mathcal{Y}} \mathrm{P}(\mathbf{y}|\mathbf{x}) \log \frac{\mathrm{P}(\mathbf{y}|\mathbf{x})}{Q(\mathbf{y})}$ *denote the KL-divergence between the two distributions. For any fixed set of non-overlapping (potentially unequally sized) bins* $\{\mathcal{Y}^m\}$, *such that* $\mathcal{Y} = \cup_m \mathcal{Y}^m$, *the weights* $\hat{\mathbf{q}}$ *that minimize* $KL(\mathrm{P}||Q)$ *are:*

$$\hat{q}_m = \sum_{\mathbf{y} \in \mathcal{Y}^m} \mathrm{P}(\mathbf{y}|\mathbf{x}) = \frac{1}{\mathcal{Z}} \sum_{\mathbf{y} \in \mathcal{Y}^m} e^{S(\mathbf{y})} \tag{8}$$

Proof. Using the method of Lagrangian multipliers and solving the system of partial derivates of the Lagrangian. More details in the supplement. □

Lemma 2 is also intuitive in its prescription – the optimal histogram is one that represents the *mass of the Gibbs distribution* over each bin. Note that the partition function $\mathcal{Z}$ is a constant and does not depend on the bin, thus we can equivalently compute the mass of the *unnormalized* Gibbs distribution, *i.e.* $\tilde{q}_m = \sum_{\mathbf{y} \in \mathcal{Y}^m} e^{S(\mathbf{y})}$, and then simply normalize these to compute $q_m = \frac{\tilde{q}_m}{\sum_{m'} \tilde{q}_{m'}}$.

Unfortunately, the problem of estimating sums of the Gibbs distribution under general hamming-ball constraints continues to be #P-complete. Thus, we proposed to compute a simple upper-bound on the unnormalized mass:

$$\tilde{q}_m \leq |\mathcal{Y}_m| \cdot \max_{\mathbf{y} \in \mathcal{Y}^m} e^{S(\mathbf{y})}, \tag{9}$$

i.e. we upper-bound the mass of a bin with the maximum entry in a bin multiplied by the size of the bin. This upper bound is a good approximation if P is nearly flat over the bin, and a poor approximation if P is very peaky in the bin.

In order to compute this upper-bound, we build on recent advances in the graphical models literature for producing a set of diverse high-scoring solutions in CRFs [10, 11, 51], specifically the Parallel Diverse MAPs (PDivMAP) formulation of Meier *et al.* [11]. Let $\mathbf{y}^1 = \max_{\mathbf{y} \in \mathcal{Y}} e^{S(\mathbf{y})}$ be the MAP solution. We define M *circular bins or rings* around the MAP solution, with inner and outer radii of the rings given by L_m and U_m. This allows our histogram approximation to be distribution-specific and be "centered" around the MAP solution, which is where P places most mass. Formally, we search for the the highest scoring configuration in a bin via the following optimization problem:

$$\mathbf{y}^m = \operatorname*{argmax}_{\mathbf{y} \in \mathcal{Y}} \sum_{u \in \mathcal{V}} \theta_u(y_u) + \sum_{(u,v) \in \mathcal{E}} \theta_{uv}(y_u, y_v) \tag{10a}$$

$$s.t. \quad L_m \leq \Delta(\mathbf{y}, \mathbf{y}^1) \leq U_m. \tag{10b}$$

We set $U_m = L_{m+1}$, and chose L_m's by evenly dividing the range $[0 \quad \max \Delta(\cdot, \cdot)]$, so that the rings cover the entire output space $\mathcal{Y}$. Meier *et al.* [11] showed that the partial Lagrangian dual of this modified formulation is easily optimizable:

$$f(\alpha_m, \beta_m) = \max_{\mathbf{y} \in \mathcal{Y}} \; S_{\alpha,\beta}(\mathbf{y}) \doteq \sum_{u \in \mathcal{V}} \theta_u(y_u) + \sum_{(u,v) \in \mathcal{E}} \theta_{uv}(y_u, y_v)$$
$$+ \alpha_m \left(\Delta(\mathbf{y}, \mathbf{y}^1) - L_m\right) - \beta_m \left(\Delta(\mathbf{y}, \mathbf{y}^1) - U_m\right) \tag{11}$$

where α_m, β_m are the two Lagrangian multipliers for the inner and outer radius constraints respectively. This Lagrangian dual function is easy to evaluate (and consequently minimize) since the Hamming distance function is absorbed into the node terms:

$$S_{\alpha,\beta}(\mathbf{y}) = \sum_{u \in \mathcal{V}} \underbrace{\left(\boldsymbol{\theta}_u(y_u) + (\alpha_m - \beta_m)[\![y_u \neq y_u^1]\!]\right)}_{\text{Perturbed Unary Score}}$$
$$+ \sum_{(u,v) \in \mathcal{E}} \boldsymbol{\theta}_{uv}(y_{uv}). \tag{12}$$

Thus, this maximization can be performed simply by feeding a perturbed unary term to the algorithm used for MAP inference (*e.g.* α-expansion or TRW-S). Lagrangian multipliers α_m, β_m can be optimized via subgradient descent, and the update rules are described in [11].

4.4. Summary of the algorithm

To summarize the entire algorithm, we initialize the weights of the CRF $\mathbf{w}$ by training on a small set of labeled images. Then these weights are used to compute the node and edge potentials for each image in the unlabeled pool. For each unlabeled image, we produce the highest scoring configurations in the M circular bins $\{\mathcal{Y}^m\}$, use these to estimate the entropy and pick the unlabeled image with the highest estimated entropy. The parameters of the model $\mathbf{w}$ are then retrained and this process is repeated.

5. Experiments

We evaluate our approach on one synthetic and two real problems. The goal of the synthetic experiments is to perform sanity-checks and compare the performance of our algorithm when the entropy of the Gibbs distribution can be exactly computed. The goal of the real experiments is to show broad applicability and performance gains relative to other approximate inference techniques that may be used to estimate entropy.

Practical Considerations. For all experiments, we used the PDivMAP algorithm to find the highest scoring solutions in the bins, with L_m, U_m set by breaking the diversity range $[0 \;\; \max \Delta(\cdot,\cdot)]$ evenly into $M = 10$ bins and α_m, β_m optimized via subgradient descent. Naïvely computing $\sum_{i=1}^{M} e^{S(\mathbf{y}^i)}$ results in loss of numerical precision (underflow or overflow depending on whether $S(\mathbf{y}^i)$ were positive or negative). We used the "log-sum-exp trick", where the re-normalized distribution is computed as:

$$\tilde{q}_i = \frac{e^{\frac{S(\mathbf{y}^i)-S_{\min}}{T}}}{\sum_{j=1}^{M} e^{\frac{S(\mathbf{y}^j)-S_{\min}}{T}}} \tag{13}$$

where $S_{\min} = \min_{j\in[M]} S(\mathbf{y}^j)$ is the smallest score, and T is a "temperature" parameter. When $T = 1$, the role of $S_{\min}$ is simply to avoid numerical underflow/overflow and otherwise does not change the entropy approximation. When $T \leq 1$, the delta-approximate distribution is sharpened around the MAP (thus decreasing the estimated entropy), and when $T \geq 1$ the approximate distribution is flatted towards uniform (thus increasing the estimated entropy). We tried two different approaches to set T: (i) cross-validation (where we pick T to maximize performance of active learning on a fully-annotated held-out set), and (ii) scaling by the score of the MAP solution, *i.e.* $T = \left|S(\mathbf{y}^1)\right|$. Interestingly, they both performed similar, suggesting that only a normalization of the scores was needed. All results reported in the paper are from (ii).

Parameter learning: We learn $\mathbf{w}$ via Maximum (Conditional) Likelihood Estimation, optimized via Stochastic Gradient Ascent. In order to compute gradients of the likelihood, we computed marginals via sum-product loopy Belief Propagation (without damping) from Mark Schmit's UGM package [52]. We observed that BP converged in all our experiments. We also tried MCMC for computing gradients, but did not find any significant differences in the results.

In our preliminary experiments, we also tried parameter learning with max-margin objectives such as N/1-slack Structured SVMs [53–55], however the performance was not as good as MLE. We believe this is due to the fact that SSVMs are not probabilistic and lead to weight vectors and scores that are not "calibrated" probabilities. Similar observations [56, 57] have been made in the context of SVMs and Logistic Regression (the unstructured analogues of SSVM and CRFs). We believe this uncalibrated nature of scores/probabilities leads to a model whose peak (1-best) is generally accurate, but the entropy is unreliable.

Baselines. We compare our approach `Active-PDivMAP` against 7 baselines:

- `Gibbs`: we run a Gibbs sampler to produce 500 samples, and then use the delta-approximation over these samples. The burn-in period was 1000 samples.
- `Perturb-and-MAP`: we inject Gumbel noise into the node potentials, followed by MAP inference, as proposed by [42] to produce approximate samples, and then perform delta-approximation over these samples.
- `Mean-Field`: we perform variational mean-field approximation to find the fully-factorized distribution $Q_{mf}(\mathbf{y}|\mathbf{x}) = \prod_i Q_{mf}(y_i|\mathbf{x})$, which is closest to $\mathrm{P}(\mathbf{y}|\mathbf{x})$ in terms of KL-divergence. Then we compute exact entropy in this mean-field approximation.
- `Min-Marginals`: we use the ideas from interactive segmentation literature – we compute min-marginals [58] for each super-pixel, treat this min-marginal as a measure of uncertainty at this super-pixel, and use the entropies of (normalized) min-marginals averaged over all super-pixels.
- `Marginals`: we approximate the approach of Luo *et al.* [39] by calculate the marginal probabilities at each variable, and then summing these entropies to estimate the entropy for an image. The key difference is that [39] uses convex BP and we use loopy BP (we observed that BP always converged in our experiments). To be precise, this is only an approximation of the "separate" algorithm from [39]. Unfortunately, a direct comparison against all algorithms proposed by [39] is not possible because their code is not available.

- `Margin-based` [59]: we calculate the margin (difference in scores) between the best solution and the second-best solution, and select those unlabeled images with the smallest margin.
- `Rand`: we pick an unlabeled image uniformly at random to annotate.

5.1. Synthetic Experiment

Setup. We generated random spanning trees on 100 nodes. All variables took two states. The node and edge potentials were sampled from Gaussians such that the true entropy lied in the range of $[5\ \ 20]$, which represents low- and mid-level of entropy (maximum entropy possible in this setting is 100 bits). Since the graph is a tree, exact entropy can be computed via sum-product message passing. Table 1 compares the three sampling-based approaches – `PDivMAP`, `Gibbs`, and `Perturb-and-MAP` – in terms of:

(a) *Rank correlation:* correlation between their predicted ordering of trees and the correct ordering according to true entropy (higher is better)

(b) *True-Rank-of-Pred:* the average rank of the tree picked by the methods to be annotated, according to true entropy (lower is better);

(c) *Pred-Rank-of-True:* average rank of the tree with the true highest entropy in the lists generated by the methods (lower is better).

Table 1

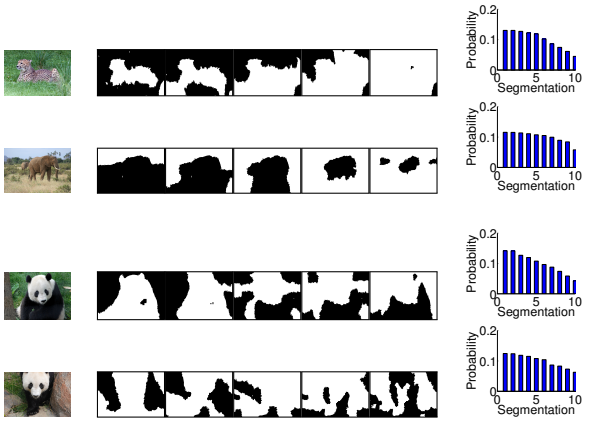
	Rank correlation (↑)	True-Rank-of-Pred (↓)	Pred-Rank-of-True (↓)
`PDivMAP`	$\mathbf{0.47 \pm 0.03}$	$\mathbf{1.9 \pm 0.18}$	$\mathbf{1.7 \pm 0.13}$
`Gibbs`	0.32 ± 0.04	6.6 ± 0.37	4.2 ± 0.18
`Perturb-and-MAP`	0.17 ± 0.04	9.5 ± 0.26	6.1 ± 0.25

In all three metrics, we can see that `PDivMAP` significantly outperforms the baselines.

5.2. Foreground-Background Segmentation

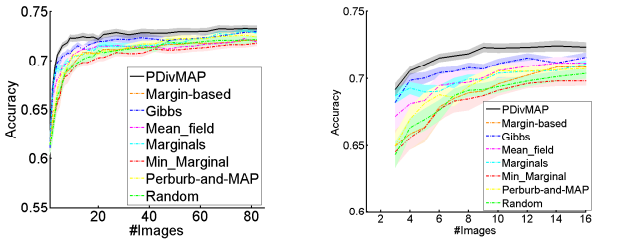
Setup. We tested our algorithm on the problem of binary (foreground-background) image segmentation. We replicated the experimental setup of [60, 61]. We used the co-segmentation dataset iCoseg [24], which consists of 37 groups of related images mimicking typical consumer photograph collections. Each group may be thought of as an "event" (*e.g.*, images from a baseball game, a safari, *etc.*). The dataset provides pixel-level ground-truth foreground-background segmentations for each image. We used 9 difficult groups containing 166 images in total. These images were then split into `train` and `test` sets of equal size. We initialize with 1 annotated image, perform active learning on the `train` set, and use the `test` to report accuracies. See Fig. 3 for some example images and segmentations.

Model and Features. The segmentation task is modeled as a binary pairwise CRF where each node corresponds to a superpixel [62] in the image. We extracted 12-dim color features at each superpixel (mean RGB; mean HSV; 5 bin Hue histogram; Hue histogram entropy). The edge features, computed for each pair of adjacent superpixels, correspond to a standard Potts model and a contrast sensitive Potts model. The weights at each edge were constrained to be positive so that the resulting supermodular potentials could be maximized via graph-cuts [63, 64].

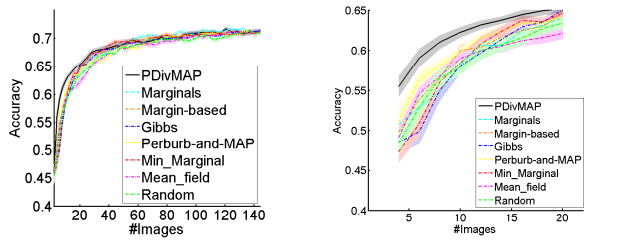


(a) $\mathbf{x}$ (b) $\mathbf{y}^1$ (c) $\mathbf{y}^3$ (d) $\mathbf{y}^5$ (e) $\mathbf{y}^7$ (f) $\mathbf{y}^9$ (g) $\mathrm{P}(\mathbf{y}|\mathbf{x})$

Figure 3: Qualitative Results: Each row shows the image, highest scoring segmentation in different bins, and the estimated distribution. Each row illustrates a different scenario: (High Entropy, Accurate MAP), (High Entropy, Inaccurate MAP), (Low Entropy, Accurate MAP), (Low Entropy, Inaccurate MAP). See Section 5 for details.



(a) Binary Segmentation (b) Binary Segmentation: Zoomed



(c) Geometric Labeling (d) Geometric Labeling: Zoomed

Figure 4: Accuracy vs number of images annotated. Shaded regions indicate confidence intervals, achieved from 20 (top) and 30 runs (bottom). We can see that our approach `Active-PDivMAP` outperforms all baselines and is very quickly able to reach the same performance as annotating the entire dataset.

5.3. Geometric Labeling

Dataset. We used CMU Geometric Context dataset of Hoiem *et al*. [13], where every region is categorized into one of three main classes: "ground", "sky", and "vertical". The "vertical" class is further divided into 5 subclasses: "left", "center", "right", "porous", and "solid". These images were then split into `train` and `test` sets with 150 and 50 images respectively. We initialized with 2 annotated images, performed active learning on the `train` set, and use the `test` to report accuracies. The segmentation task is modeled as a pairwise CRF where each node corresponds to a superpixel in the image that can take 7 states.

5.4. Results and Analysis

Qualitative Results. Fig. 3 shows example images, diverse solutions under the current model, and entropy estimated by our approach. We generally observed four basic situations:

- **High Entropy, Accurate MAP**: The MAP is accurate but the model also places similar mass on other solutions (often poorer than MAP). This is the case when the model needs to become "sharper" towards MAP.
- **High Entropy, Inaccurate MAP**: The MAP is inaccurate and the model is also highly uncertain, typically placing mass on many inaccurate solutions. This typically happens when the model is either overfitting, or some test image is particularly difficult.
- **Low Entropy, Accurate MAP**: The MAP is accurate and the model is confident in the sense that other solutions are generally inaccurate and low-probability. These are the least informative images, and we are able to avoid selecting them for annotation.
- **Low Entropy, Inaccurate MAP**: The MAP is inaccurate but the diverse solutions have a lower score than the MAP. In some sense, these images are highly informative since after annotation they have the ability to cause a significant change in the parameters. However, we do not have the ability to distinguish these cases from the previous one without seeking annotations, and thus entropy-based methods are not able to exploit such images.

Quantitative Results and Take-Home Message. For both experiments, we ran multiple runs with different initial images (30 runs for binary segmentation and 20 runs for geometric labeling). Fig. 4 shows the accuracy of various methods vs the number of images annotated for both datasets (shaded regions indicate confidence intervals). Note that the performance of a "fully supervised" approach is the rightmost point on the curve. We can see that our approach `Active-PDivMAP` significantly outperforms all the baselines, with no overlap in confidence intervals. Moreover, `Active-PDivMAP` is able to reach within 1%-points of the final accuracy (where all images have been annotated) with less than 9% of the data annotated. Based on Mechanical Turk annotation statistics reported in previous work [65], a simple back-of-the-envelope calculation – assuming 3-minutes per image, 10-cents per image $\times$ 5 MTurk annotations per image – show that our approach saved approximately 45 hours of human-effort and $35 – even for these medium-sized dataset.

Overall, outperforming `Rand` shows that even though it may be crude, the histogram approximation does capture enough information about the entropy to be useful. Outperforming `Gibbs` and `Perturb-and-Map` shows the power of using non-overlapping bins as opposed to IID samples, and outperforming `Mean-Field`, `Min-Marginals` and `Marginals` shows that it is better to approximate the entropy computation with a histogram approximation than with a fully factorized model.

Efficiency and Runtime. Due to reliance on efficient MAP solvers (e.g dynamic graph-cuts in binary segmentation), our implementation has fairly low overhead. Specifically, 60 subgradient iterations $\times$ 10 solutions takes 1.8s, which is much less than Gibbs (40s for 500 samples), comparable to LoopyBP (1.2s), MeanField (1.4s), and slower than Margin-based (0.12s), and MinMarginals (0.08s).

6. Conclusions

We investigated active learning in structured probabilistic models such as CRFs. The key challenge in such models is that computing entropy of the model on unlabeled images is intractable, since the distribution has an exponentially-large support. We proposed a variational "histogram approximation" approach for estimating entropy that replaces the exponentially-large support with a *coarsened* distribution that may be viewed as a histogram over M bins.

Generalizations. The assumption that bins are non-overlapping simplifies the theoretical analysis, but our approach can be easily generalized to non-overlapping bins. At the heart of our approach is the idea of deterministic sampling inside an appropriately defined bin, which can be generalized to other (Diverse) M-Best MAP methods.

We found that our approximation outperforms techniques such as Gibbs sampling (which come with strong asymptotic convergence guarantees), presumably because such a deterministic sampling procedure is quickly able to determine if the distribution under consideration is flat or peaky (which is our primary consideration from an active learning perspective), while Gibbs samplers have a difficult time transitioning out of the biggest mode of the distribution.

Overall, our proposed solution is theoretically well-motivated, computationally efficient, easy to implement, and practical.

Acknowledgements. AL contributed to this work while he was an intern at Virginia Tech. This work was partially supported by the National Science Foundation under grants IIS-1353694 and IIS-1350553, the Army Research Office YIP Award W911NF-14-1-0180, and the Office of Naval Research grant N00014-14-1-0679. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the U.S. Government or any sponsor.

References

[1] J. Hays and A. A. Efros, "im2gps: estimating geographic information from a single image," in *CVPR*, 2008. 1

[2] D. Parikh and C. Zitnick, "The role of features, algorithms and data in visual recognition," in *CVPR*, 2010. 1

[3] X. Zhu, C. Vondrick, D. Ramanan, and C. Fowlkes, "Do we need more training data or better models for object detection?," in *BMVC*, 2012. 1

[4] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *IJCV*, vol. 88, pp. 303–338, June 2010. 1

[5] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A Large-Scale Hierarchical Image Database," in *CVPR*, 2009. 1, 3

[6] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common objects in context," 2014. 1, 3

[7] B. Settles, *Active Learning*. Synthesis Lectures on Artificial Intelligence and Machine Learning, Morgan & Claypool, 2012. 1

[8] M. J. Wainwright and M. I. Jordan, "Graphical models, exponential families, and variational inference," *Foundations and Trends in Machine Learning*, vol. 1, no. 1-2, pp. 1–305, 2008. 1

[9] L. G. Valiant, "The complexity of computing the permanent," *Theoretical Computer Science*, vol. 8, no. 2, 1979. 2

[10] D. Batra, P. Yadollahpour, A. Guzman-Rivera, and G. Shakhnarovich, "Diverse M-Best Solutions in Markov Random Fields," in *ECCV*, 2012. 2, 5

[11] F. Meier, A. Globerson, and F. Sha, "The More the Merrier: Parameter Learning for Graphical Models with Multiple MAPs," in *ICML Workshop on Inferning: Interactions between Inference and Learning*, 2013. 2, 5, 6

[12] A. Prasad, S. Jegelka, and D. Batra, "Submodular meets structured: Finding diverse subsets in exponentially-large structured item sets," in *NIPS*, 2014. 2

[13] D. Hoiem, A. A. Efros, and M. Hebert, "Recovering surface layout from an image," *IJCV*, vol. 75, no. 1, 2007. 2, 8

[14] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman, "LabelMe: A database and web-based tool for image annotation," *IJCV*, vol. 77, pp. 157–173, May 2008. 2, 3

[15] L. von Ahn and L. Dabbish, "Labeling images with a computer game," in *CHI*, CHI '04, 2004. 2

[16] J. Winn and N. Jojic, "LOCUS: learning object classes with unsupervised segmentation," in *CVPR*, 2005. 2

[17] H. Arora, N. Loeff, D. Forsyth, and N. Ahuja, "Unsupervised segmentation of objects using efficient learning," in *CVPR*, 2007. 2

[18] J. Xu, A. G. Schwing, and R. Urtasun, "Tell me what you see and I will show you where it is," in *CVPR*, 2014. 2

[19] X. He and R. S. Zemel, "Learning hybrid models for image annotation with partially labeled data.," in *NIPS*, 2008. 2

[20] J. Verbeek and W. Triggs, "Scene Segmentation with CRFs Learned from Partially Labeled Images," in *NIPS*, 2008. 2

[21] Y. Boykov and M.-P. Jolly, "Interactive graph cuts for optimal boundary and region segmentation of objects in n-d images," *ICCV*, 2001. 2

[22] C. Rother, V. Kolmogorov, and A. Blake, ""GrabCut": interactive foreground extraction using iterated graph cuts," *SIGGRAPH*, 2004. 2

[23] D. Batra, R. Sukthankar, and T. Chen, "Semi-supervised clustering via learnt codeword distances," in *BMVC*, 2008. 2

[24] D. Batra, A. Kowdle, D. Parikh, J. Luo, and T. Chen, "iCoseg: Interactive Co-segmentation with Intelligent Scribble Guidance," in *CVPR*, 2010. 2, 7

[25] D. Küttel, M. Guillaumin, and V. Ferrari, "Segmentation propagation in imagenet," in *ECCV*, 2012. 2

[26] Y. Freund, H. Seung, E. Shamir, and N. Tishby, "Selective sampling using the query by committee algorithm," *Machine Learning*, vol. 28, no. 2-3, pp. 133–168, 1997. 2

[27] S. Tong and D. Koller, "Support vector machine active learning with applications to text classification," *JMLR*, vol. 2, pp. 45–66, Mar. 2002. 2

[28] D. MacKay, "Information-based objective functions for active data selection," *Neural Computation*, vol. 4, no. 4, pp. 590–604, 1992. 2

[29] G.-J. Qi, X.-S. Hua, Y. Rui, J. Tang, and H.-J. Zhang, "Two-dimensional active learning for image classification," in *CVPR*, pp. 1–8, 2008. 2

[30] A. Kapoor, K. Grauman, R. Urtasun, and T. Darrell, "Active learning with gaussian processes for object categorization," in *ICCV*, 2007. 2

[31] P. Jain and A. Kapoor, "Active learning for large multi-class problems," in *CVPR*, 2009. 2

[32] R. Yan, J. Yang, and A. Hauptmann, "Automatically labeling video data using multi-class active learning," in *ICCV*, 2003. 2

[33] B. Collins, J. Deng, K. Li, and L. Fei-Fei, "Towards scalable dataset construction: An active learning approach," in *ECCV*, 2008. 2

[34] A. Fathi, M. F. Balcan, X. Ren, and J. M. Rehg, "Combining self training and active learning for video segmentation," in *BMVC*, 2011. http://dx.doi.org/10.5244/C.25.78. 2

[35] S. Vijayanarasimhan and K. Grauman, "Active frame selection for label propagation in videos," in *ECCV*, 2012. 2

[36] C. Vondrick, D. Patterson, and D. Ramanan, "Efficiently scaling up crowdsourced video annotation," *IJCV*, vol. 101, pp. 184–204, Jan. 2013. 2

[37] B. Settles and M. Craven, "An analysis of active learning strategies for sequence labeling tasks," in *EMNLP*, 2008. 2

[38] A. Culotta and A. McCallum, "Reducing labeling effort for structured prediction tasks," in *AAAI*, 2005. 2

[39] W. Luo, A. G. Schwing, and R. Urtasun, "Latent Structured Active Learning," in *NIPS*, 2013. 3, 6

[40] T. Hazan and A. Shashua, "Norm-product belief propagation: Primal-dual message-passing for approximate inference," *Information Theory, IEEE Trans. on*, vol. 56, pp. 6294–6316, Dec 2010. 3

[41] S. Maji, T. Hazan, and T. Jaakkola, "Active boundary annotation using random map perturbations," in *AISTATS*, 2014. 3

[42] G. Papandreou and A. L. Yuille, "Perturb-and-map random fields: Using discrete optimization to learn and sample from energy models," in *ICCV*, 2011. 3, 6

[43] D. Tarlow, R. P. Adams, and R. S. Zemel, "Randomized optimum models for structured prediction," in *Fifteenth International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2012. 3

[44] S. Vijayanarasimhan and K. Grauman, "What's it going to cost you?: Predicting effort vs. informativeness for multi-label image annotations," in *CVPR*, 2009. 3

[45] B. Siddiquie and A. Gupta, "Beyond active noun tagging: Modeling contextual interactions for multi-class active learning.," in *CVPR*, 2010. 3

[46] A. Parkash and D. Parikh, "Attributes for classifier feedback," in *ECCV*, 2012. 3

[47] D. Koller and N. Friedman, *Probabilistic Graphical Models: Principles and Techniques*. MIT Press, 2009. 4

[48] D. Nilsson, "An efficient algorithm for finding the m most probable configurations in probabilistic expert systems," *Statistics and Computing*, vol. 8, pp. 159–173, 1998. 10.1023/A:1008990218483. 4

[49] C. Yanover and Y. Weiss, "Finding the m most probable configurations using loopy belief propagation," in *NIPS*, 2003. 4

[50] D. Batra, "An Efficient Message-Passing Algorithm for the M-Best MAP Problem," in *Uncertainty in Artificial Intelligence*, 2012. 4

[51] C. Chen, V. Kolmogorov, Y. Zhu, D. Metaxas, and C. H. Lampert, "Computing the m most probable modes of a graphical model," in *AISTATS*, 2013. 5

[52] M. Schmidt, "http://www.cs.ubc.ca/~schmidtm/Software/UGM.html." UGM: A Matlab toolbox for probabilistic undirected graphical models. 6

[53] I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun, "Large margin methods for structured and interdependent output variables," *JMLR*, vol. 6, pp. 1453–1484, 2005. 6

[54] T. Joachims, T. Finley, and C.-N. Yu, "Cutting-plane training of structural svms," *Machine Learning*, vol. 77, no. 1, pp. 27–59, 2009. 6

[55] A. Guzman-Rivera, P. Kohli, and D. Batra, "Divmcuts: Faster training of structural svms with diverse m-best cutting-planes.," in *AISTATS*, 2013. 6

[56] J. C. Platt, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods," in *Advances in Large Margin Classifiers*, 1999. 6

[57] A. Niculescu-Mizil and R. Caruana, "Predicting good probabilities with supervised learning," in *ICML*, 2005. 6

[58] P. Kohli and P. H. S. Torr, "Measuring uncertainty in graph cut solutions," *CVIU*, vol. 112, no. 1, pp. 30–38, 2008. 6

[59] D. Roth and K. Small, "Margin-based active learning for structured output spaces," in *ECML*, 2006. 7

[60] A. Guzman-Rivera, D. Batra, and P. Kohli, "Multiple Choice Learning: Learning to Produce Multiple Structured Outputs," in *Proc. NIPS*, 2012. 7

[61] A. Guzman-Rivera, P. Kohli, D. Batra, and R. Rutenbar, "Efficiently enforcing diversity in multi-output structured prediction," in *AISTATS*, 2014. 7

[62] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "SLIC superpixels compared to state-of-the-art superpixel methods," *PAMI*, vol. 34, no. 11, 2012. 7

[63] Y. Boykov, O. Veksler, and R. Zabih, "Efficient approximate energy minimization via graph cuts," *PAMI*, vol. 20, no. 12, pp. 1222–1239, 2001. 7

[64] V. Kolmogorov and R. Zabih, "What energy functions can be minimized via graph cuts?," *PAMI*, vol. 26, no. 2, pp. 147–159, 2004. 7

[65] A. Sorokin and D. Forsyth, "Utility data annotation with amazon mechanical turk," in *Workshop on Internet Vision, CVPR.*, pp. 1–8, 2008. 8