Picture: a probabilistic programming language for scene perception

Tejas D Kulkarni[1], Pushmeet Kohli[2], Joshua B Tenenbaum[1], Vikash Mansinghka[1]
[1]Brain and Cognitive Science, Computer Science and Artificial Intelligence Lab, Massachusetts Institute of Technology. [2]Microsoft Research Cambridge.

Probabilistic scene understanding systems aim to produce high-probability descriptions of scenes conditioned on observed images or videos, typically either via discriminatively trained models or generative models in an "analysis by synthesis" framework. Discriminative approaches lend themselves to fast, bottom-up inference methods and relatively knowledge-free, data-intensive training regimes, and have been remarkably successful on many recognition problems. Generative approaches hold out the promise of analyzing complex scenes more richly and flexibly, but have been less widely embraced for two main reasons: Inference typically depends on slower forms of approximate inference, and both model-building and inference can involve considerable problem-specific engineering to obtain robust and reliable results. These factors make it difficult to develop simple variations on state-of-the-art models, to thoroughly explore the many possible combinations of modeling, representation, and inference strategies, or to richly integrate complementary discriminative and generative modeling approaches to the same problem. More generally, to handle increasingly realistic scenes, generative approaches will have to scale not just with respect to data size but also with respect to model and scene complexity. This scaling will arguably require general-purpose frameworks to compose, extend and automatically perform inference in complex structured generative models – tools that for the most part do not yet exist.

Here we present *Picture*, a probabilistic programming language [1] that aims to provide a common representation language and inference engine suitable for a broad class of generative scene perception problems. We see probabilistic programming as key to realizing the promise of "vision as inverse graphics". Generative models can be represented via stochastic code that samples hypothesized scenes and generates images given those scenes. Rich deterministic and stochastic data structures can express complex 3D scenes that are difficult to manually specify. Multiple representation and inference strategies are specifically designed to address the main perceived limitations of generative approaches to vision. Instead of requiring photorealistic generative models with pixel-level matching to images, we can compare hypothesized scenes to observations using a hierarchy of more abstract image representations such as contours, discriminatively trained part-based skeletons, or deep neural network features. Top-down Monte Carlo inference algorithms include not only traditional Metropolis-Hastings, but also more advanced techniques for inference in high-dimensional continuous spaces, such as elliptical slice sampling, and Hamiltonian Monte Carlo which can exploit the gradients of automatically differentiable renderers. These top-down inference approaches are integrated with bottom-up and automatically constructed data-driven proposals, which can dramatically accelerate inference by eliminating most of the "burn in" time of traditional samplers and enabling rapid mode-switching.

We demonstrate *Picture* on three challenging vision problems: inferring the 3D shape and detailed appearance of faces, the 3D pose of articulated human bodies, and the 3D shape of medially-symmetric objects. The vast majority of code for image modeling and inference is reusable across these and many other tasks. We shows that *Picture* yields competitive performance with optimized baselines on each of these benchmark tasks.

[1] Noah Goodman, Vikash Mansinghka, Daniel Roy, Keith Bonawitz, and Daniel Tarlow. Church: a language for generative models. *arXiv preprint arXiv:1206.3255*, 2012.

[2] Richard David Wilkinson. Approximate bayesian computation (abc) gives exact results under the assumption of model error. *Statistical applications in genetics and molecular biology*, 12 (2):129–141, 2013.

[3] David Wingate, Noah D Goodman, A Stuhlmueller, and J Siskind. Nonstandard interpretations of probabilistic programs for efficient inference. *NIPS*, 23, 2011.

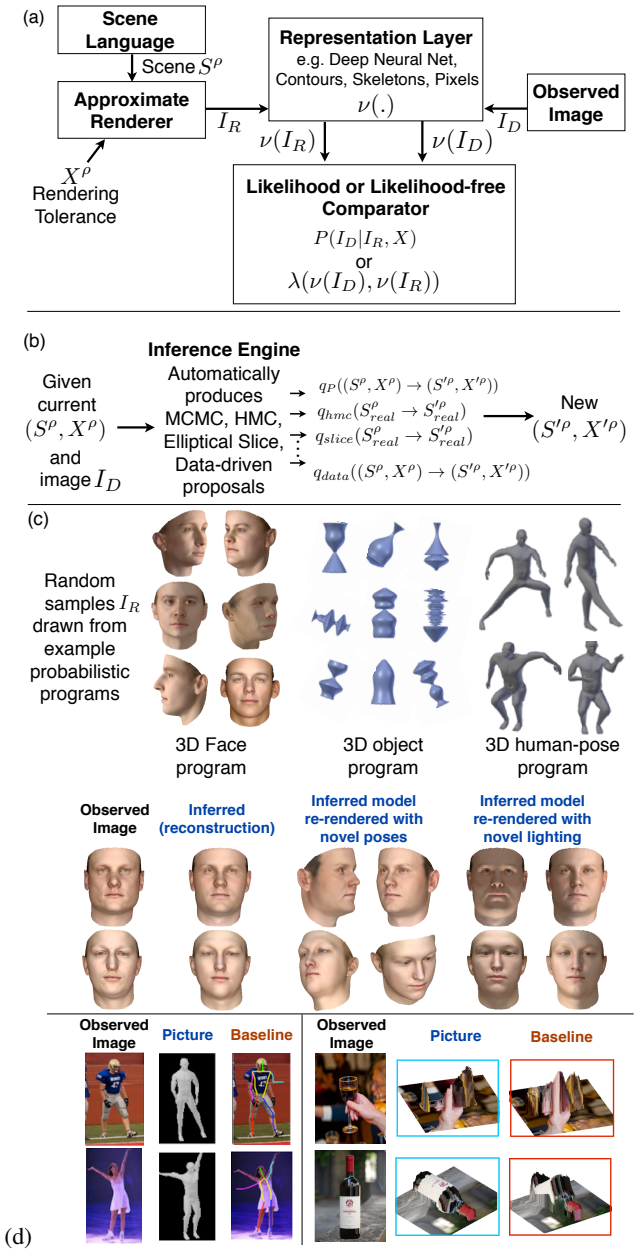

Figure 1: **Overview: (a)** All models share a common template; only the scene description S and image I_D changes across problems. Every probabilistic graphics program f defines a stochastic procedure that generates both a scene description and all the other information needed to render an approximation I_R of a given observed image I_D. The program f induces a joint probability distribution on these program traces ρ. Every *Picture* program has the following components. **Scene Language:** Describes 2D/3D scenes and generates particular scene related trace variables $S^\rho \in \rho$ during execution. **Approximate Renderer:** Produces graphics rendering I_R given S^ρ and latents X^ρ for controlling the fidelity or tolerance of rendering. **Representation Layer:** Transforms I_D or I_R into a hierarchy of coarse-to-fine image representations $\nu(I_D)$ and $\nu(I_R)$ (deep neural networks, contours and pixels). **Comparator:** During inference, I_R and I_D can be compared using a likelihood function or a distance metric λ (as in Approximate Bayesian Computation [2]). **(b) Inference Engine:** Automatically produces a variety of proposals and iteratively evolves the scene hypothesis S to reach a high probability state given I_D. **(c):** Representative random scenes drawn from probabilistic graphics programs for faces, objects, and bodies. **(d) Illustrative results:** We demonstrate *Picture* on a variety of 3D computer vision problems and check their validity with respect to ground truth annotations and task-specific baselines.

This is an extended abstract. The full paper is available at the Computer Vision Foundation webpage.