Journal of Machine Learning Research 24 (2023) 1-36 Submitted 2/21 Revised 2/23; Published 6/23

Variational Inference for Deblending Crowded Starfields

Runjing Liu RUNJING_LIU@BERKELEY.EDU
Department of Statistics
University of California, Berkeley
Berkeley, CA 94720, USA

Jon D. McAuliffe JON@STAT.BERKELEY.EDU
The Voleon Group
Berkeley, CA 94704, USA
and
Department of Statistics
University of California, Berkeley
Berkeley, CA 94720, USA

Jeffrey Regier REGIER@UMICH.EDU
Department of Statistics
University of Michigan
Ann Arbor, MI 48109, USA

The LSST Dark Energy Science Collaboration

Editor: Shakir Mohamed

Abstract

In images collected by astronomical surveys, stars and galaxies often overlap visually. Deblending is the task of distinguishing and characterizing individual light sources in survey images. We propose StarNet, a Bayesian method to deblend sources in astronomical images of crowded star fields. StarNet leverages recent advances in variational inference, including amortized variational distributions and an optimization objective targeting an expectation of the forward KL divergence. In our experiments with SDSS images of the M2 globular cluster, StarNet is substantially more accurate than two competing methods: Probabilistic Cataloging (PCAT), a method that uses MCMC for inference, and DAOPHOT, a software pipeline employed by SDSS for deblending. In addition, the amortized approach to inference gives StarNet the scaling characteristics necessary to perform Bayesian inference on modern astronomical surveys.

Keywords: Bayesian methods, amortized inference, approximate inference, cataloging astronomical surveys

1. Introduction

Astronomical images record the arrival of photons from distant light sources. Astronomical catalogs are constructed from these images. Catalogs label light sources as stars, galaxies, or other objects; they also list the physical characteristics of light sources such as flux, color, and morphology. These catalogs are the starting point for many downstream analyses. For example, Bayestar used a catalog of stellar fluxes and colors to infer the 3D distribution of

interstellar dust (Green et al., 2019). Catalogs of galaxy morphologies have also been used to validate theoretical models of dark matter and dark energy (Abbott et al., 2018).

A light source, be it a star or a galaxy, produces a peak brightness intensity at some location in an image. When light sources are well separated, catalog construction is relatively straightforward: characteristics of each light source, such as flux, can be estimated by analyzing intensities at the peak and surrounding pixels. However, in images crowded with many light sources, observed pixel intensities may result from the combined light of multiple sources. Source separation, or *deblending*, is the task of differentiating and characterizing individual light sources from a mixture of intensities in an image. A key challenge in deblending is inferring whether an observed intensity is blended, that is, whether it is composed of a single source or multiple (dimmer) sources.

Deblending is challenging for several reasons. First, it is an unsupervised problem without ground truth labeled data. Second, it is a problem with a sample size of one: there is only one night sky, which is imaged many times, and the collected survey images capture overlapping regions of it. Third, for blended fields, the properties of light sources are ambiguous; therefore, providing calibrated uncertainties for catalog construction is as important as making accurate predictions. Finally, the scale of the data is immense. The upcoming Rubin Observatory Legacy Survey of Space and Time (LSST), scheduled to begin full operations by 2024, is expected to produce 60 petabytes of astronomical images over its lifetime (LSST, 2023).

As more powerful telescopes are developed, and their ability to detect more distant light sources improves, the density of the imaged light sources will increase. For instance, Bosch et al. (2018) estimates that 58% of light sources are blended in images captured by the Subaru Telescope's Hyper Suprime-Cam, and that percentage is expected to increase for LSST (Sanchez et al., 2021). Therefore, developing a method that reliably characterizes light sources, even in cases of significant blending, advances astronomical research that derives conclusions about the physical universe from estimated catalogs.

We focus on cataloging applications in which all light sources are well modeled as points without spatial extent. Point-source-only models are applicable to surveys such as the Dark Energy Camera (DECam) plane survey, which imaged the center of the Milky Way (Schlafly et al., 2018), and the Wide-field Infrared Survey Explorer, which has a telescope resolution that does not allow for differentiation between stars and galaxies (Wright et al., 2010). We use images from the Sloan Digital Sky Survey (SDSS) of the globular cluster M2, which is a region that is densely populated with stars, as a test bed for assessing the accuracy of our approach. We also demonstrate the ability of our method to scale to large, modern astronomical surveys by cataloging a subregion of the DECam survey.

1.1 From Software Pipelines to Probabilistic Cataloging

Traditionally, most cataloging has been performed using software pipelines. These pipelines are algorithms that usually involve the following stages: locating the brightest peaks, estimating fluxes, and subtracting the estimated light sources. These stages may be performed iteratively. Pipelines do not normally produce statistically calibrated error estimates that propagate the uncertainty accumulating in each of the steps. Failure to properly accumulate error at each step results in unreliable point estimates for images in which ambiguity exists

in identifying sources and estimating their characteristics. For example, PHOTO (Lupton et al., 2001), the default cataloging pipeline used by SDSS, failed to produce a catalog of the Messier 2 (M2), a globular cluster (Portillo et al., 2017).

In contrast, *probabilistic* cataloging posits a statistical model consisting of a likelihood for the observed image given a catalog and a prior distribution over possible catalogs (Portillo et al., 2017; Brewer et al., 2013; Feder et al., 2020). Instead of deriving a single catalog, probabilistic cataloging produces a posterior distribution over the set of all possible catalogs. Uncertainties are quantified by the posterior distribution. For example, in an image with an ambiguously blended bright peak, some catalogs sampled from the posterior would contain multiple dim light sources while others would contain one bright source. The relative density that the posterior distribution places on one explanation relative to others another represents the statistical confidence in that explanation. Moreover, a distribution over the set of all catalogs induces a distribution on any estimate derived from a catalog. Therefore, calibrated uncertainties can be propagated to downstream analyses.

Previous work on probabilistic cataloging employed Markov chain Monte Carlo (MCMC) to sample from the posterior distribution. The MCMC procedure in Portillo et al. (2017) and Feder et al. (2020) is called PCAT, short for "Probabilistic CATaloging."[1] A difficulty in any probabilistic cataloging approach is that the number of sources in a catalog is unknown and random, so the latent variable space is transdimensional. PCAT sampled transdimensional catalogs with reversible jump MCMC (Green, 1995), in which auxiliary variables are added to encode the "birth" and "death" of light sources in the Markov chain.

The computational cost of MCMC for this model is problematic for large-scale astronomical surveys. Early implementations of PCAT required a day to process a 100×100-pixel SDSS image of M2 (Portillo et al., 2017). More recent implementations running inexact MCMC brought the runtime down to 30 minutes (Feder et al., 2020). However, a 100×100 pixel image covers only a 0.66×0.66 arcminute patch of the sky. For comparison, in one night, SDSS scans a region on the order of 100×1000 arcminutes. Extrapolating the 30-minute runtime suggests that PCAT would take on the order of ten years to process a nightly SDSS run. The LSST survey will be even larger, collecting five trillion pixels nightly (LSST, 2023b), which would require $28,000$ years to catalog using PCAT.

As an alternative to MCMC, Regier et al. (2019) proposed to use variational inference (VI) to approximate the posterior. VI considers a family of candidate approximate posteriors and employs numerical optimization to find the distribution in the family closest in KL divergence to the exact posterior (Jordan et al., 1999; Wainwright and Jordan, 2008; Blei et al., 2017). With a sufficiently constrained family of distributions, the VI optimization problem can be solved orders of magnitude faster than MCMC runs.

However, Regier et al. (2019) is limited in that the number of light sources in a given image is treated as known and fixed—it had to be set using a preprocessing routine. The authors avoided the transdimensional latent variable space induced by the unknown number of sources in order to have a tractable objective for numerical optimization.

1. We use "probabilistic cataloging" to refer to any method that produces a posterior over possible catalogs, whereas "PCAT" refers specifically to the MCMC procedure in Portillo et al. (2017) and Feder et al. (2020).

1.2 Our Contribution

We propose *StarNet*, an approach to deblending that employs several recent VI innovations (Zhang et al., 2019; Le et al., 2020). Unlike Regier et al. (2019), our VI approach is able to handle arbitrary probabilistic models, including a transdimensional model with an unknown number of sources. Section 2 introduces the statistical model, which is similar to the model used in PCAT.

Secondly, again unlike Regier et al. (2019), we employ amortization, which enables StarNet to scale inference to large astronomical surveys. In amortized variational inference, a neural network maps input images to an approximate posterior over catalogs. Following a one-time cost to fit the neural network, inference on new images requires just a single neural-network evaluation. Rapid inference is possible without the need to re-run MCMC or numerically optimize VI for each new image. For StarNet, a network evaluation, or "forward pass," on a 100×100 pixel image takes less than a second (vs. 30 minutes for inference using PCAT). Section 3 details the variational distribution and neural network architecture in StarNet.

Finally, and critically, StarNet is fit using an expected "forward" Kullback–Leibler (KL) divergence between the approximate posterior q and the exact posterior p, where the expectation is taken over the data distribution defined by the statistical model. In contrast, traditional variational inference minimizes the "reverse" KL divergence (Bishop, 2006), which uses an expectation with respect to the variational distribution. The forward KL is minimized using stochastic gradient descent (SGD), which involves sampling complete data—images and their corresponding catalogs—from their joint likelihood and fitting the network in a supervised fashion. Section 4 details our inference procedure.

In this application, optimizing the forward KL produces more reliable approximate posteriors than optimizing the traditional reverse KL (Section 5): taking advantage of complete data allows the network to better avoid shallow local minima where the approximate posterior is far from the exact posterior in terms of KL divergence.

The forward KL has been used in previous research to train deep generative models (Ambrogioni et al., 2019; Le et al., 2020), and appears in the sleep phase of the wake-sleep algorithm (Hinton et al., 1995; Bornschein and Bengio, 2014; Le et al., 2020). Variational inference using the forward KL is an example of simulation-based inference, where approximate posteriors are constructed for likelihoods from which sampling is easy, but are unavailable analytically (Papamakarios and Murray, 2016; Greenberg et al., 2019). Simulation-based inference has found applications in physics where theory can provide realistic simulations (Cranmer et al., 2020). For example, Baydin et al. (2019) use simulation-based inference to model time-series data of particle paths at the Large Hadron Collider, and they use the forward KL objective to fit a recurrent neural network, whose output are proposals to an MCMC sampling scheme. To the best of our knowledge, our work is the first to combine amortized inference with the forward KL divergence to perform Bayesian inference over a transdimensional latent space, producing an approximate posterior distribution over sets.

We applied StarNet to an SDSS image of M2, a globular cluster (Section 6.1) and show that StarNet was more accurate than the MCMC-based cataloger PCAT: though MCMC is asymptotically exact, it often suffers from incomplete mixing on practical timescales. StarNet was also more accurate that traditional deterministic cataloging approaches in

several metrics. We then demonstrate the scalability of StarNet by cataloging a DECam image of the Milky Way (Section 6.2). Our approximate Bayesian method can produce scientifically relevant results on the order of minutes, while running PCAT would take on the order of days.

Code to reproduce our results is publicly available in a GitHub repository (BLISS, 2023).

2. The Generative Model

In crowded starfields such as globular clusters and the galactic plane of the Milky Way, the vast majority of light sources are stars. An astronomical image records the number of photons that reached a telescope and arrived at each pixel. Typically, photons must pass through one of several filters, each selecting photons from a specified band of wavelengths, before being recorded.

For a given $H \times W$ pixel image with B filter bands, our goal is to infer a catalog of stars. The catalog specifies the number of stars in an image; for each such star, the catalog records its location and its flux (brightness) in each band. The space of latent variables $\mathcal{Z}$ is the collection of all possible catalogs of the form

$$z := \{N, (\ell_i, f_{i,1}, ..., f_{i,B})_{i=1}^{N}\},$$

where the number of stars in the catalog is $N \in \mathbb{N}$; the location of star i is $\ell_i \in \mathbb{R}^2$; and the flux of the star i in band b is $f_{i,b} \in \mathbb{R}^+$.

A Bayesian approach requires specification of a prior over catalog space $\mathcal{Z}$ and a likelihood for the observed images. Our likelihood and prior, detailed below, are similar to previous approaches (Brewer et al., 2013; Portillo et al., 2017; Feder et al., 2020), which facilitates the comparisons of inference algorithms in isolation of model differences.

2.1 The Prior

The prior over $\mathcal{Z}$ is a marked spatial Poisson process. To sample the prior, first draw the number of stars contained in the $H \times W$ image as

$$N \sim \text{Poisson}(\mu HW), \tag{1}$$

where μ is a hyperparameter specifying the average number of sources per pixel. Next, draw locations

$$\ell_1, ..., \ell_N \mid N \overset{iid}{\sim} \text{Uniform}([0, H] \times [0, W]).$$

The fluxes in the first band are from a power law distribution with slope α:

$$f_{1,1}, ..., f_{N,1} \mid N \overset{iid}{\sim} \text{Pareto}(f_{min}, \alpha). \tag{2}$$

Fluxes in other bands are described relative to the first band. Like Feder et al. (2020), we define the log-ratio of flux relative to the first band as "color." Colors are drawn from a Gaussian distribution

$$c_{1,b}, ..., c_{N,b} \mid N \overset{iid}{\sim} \mathcal{N}(\mu_c, \sigma_c^2), \quad b = 2, ..., B.$$

Given the flux in the first band $f_{i,1}$ and color $c_{i,b}$, the flux in band b is $f_{i,b} = f_{i,1} \times 10^{c_{i,b}/2.5}$.

We set the power law slope $\alpha = 0.5$ and use a standard Gaussian for the color prior ($\mu_c = 0$, $\sigma_c^2 = 1$), as in Feder et al. (2020).

Rather than having a hierarchical structure, the prior parameters are fixed in this model: our goal is to produce a posterior on catalogs for a specific image, not to model the population over many images. Appendix E evaluates the sensitivity of the resulting catalog to choices of the prior parameters.

2.2 The Likelihood

Let x_{hw}^b denote the observed number of photoelectrons at pixel (h, w) in band b. For each band, at every pixel, the expected number of photoelectron arrivals is $\lambda_{hw}^b(z)$, a deterministic function of the catalog z. Motivated by the Poissonian nature of photon arrivals and the large photon arrival rate in SDSS and LSST images, observed pixel intensities are drawn as

$$x_{hw}^b \mid z \overset{ind}{\sim} \mathcal{N}(\lambda_{hw}^b, \lambda_{hw}^b), \quad b = 1, ..., B;\ h = 1, ..., H;\ w = 1, ..., W,$$
$$\text{where} \quad \lambda_{hw}^b = I_b + \sum_{i=1}^{N} f_{i,b} \mathcal{P}_b\big(h - \ell_{i,1}, w - \ell_{i,2}\big).$$

Here, $\mathcal{P}_b$ is the point spread function (PSF) for band b and I_b is the background intensity. The PSF is a function $\mathcal{P}_b : \mathbb{R} \times \mathbb{R} \mapsto \mathbb{R}^+$, describing the appearance of a stellar point source at any 2D position of the image. Our PSF model is a weighted average between a Gaussian "core" and a power-law "wing" as described in Xin et al. (2018). For each band, the PSF has the form

$$\mathcal{P}(u, v) = \frac{\exp(\frac{-(u^2+v^2)}{2\sigma_1^2}) + \zeta \exp(\frac{-(u^2+v^2)}{2\sigma_2^2}) + \rho(1 + \frac{v^2+u^2}{\gamma\sigma_P^2})^{-\gamma/2}}{1 + \zeta + \rho}.$$

The PSF parameters are allowed to vary by band. In our applications to SDSS and DECam data, we use estimates of the background and PSF obtained from a pre-processing pipeline that are distributed by these surveys along with the images.

3. The Variational Distribution

The central quantity in Bayesian statistics is the posterior distribution $p(z \mid x)$. However, in many nontrivial probabilistic models, including our own, the posterior distribution is intractable to calculate—it requires us to compute the marginal likelihood, $p(x)$, which involves integrating over the latent variable z. In our model, the latent variable space is high dimensional: it is the set of all catalogs. Approximate methods such as MCMC and variational inference are therefore required.

Variational inference (Jordan et al., 1999; Wainwright and Jordan, 2008; Blei et al., 2017) posits a family of distributions $\mathcal{Q}$ and seeks the distribution $q^* \in \mathcal{Q}$ that is "closest" to the exact posterior in KL divergence. The defined divergence and the family $\mathcal{Q}$ are chosen such that minimizer q^* will not be too difficult to find via optimization. We index

the distributions in $\mathcal{Q}$ using a real-valued vector η, in which case solving for the optimal variational distribution q_{η^*} becomes a numerical optimization problem.

Most commonly, variational inference minimizes the "reverse" KL divergence between q and p:

$$\eta^* = \arg\min_{\eta} \mathrm{KL}\Big[\, q_\eta(z \mid x) \,\|\, p(z \mid x) \,\Big]. \tag{3}$$

Minimizing the KL divergence in (3) is equivalent to maximizing the evidence lower bound (ELBO):

$$\mathcal{L}_{elbo}(\eta) = \mathbb{E}_{q_\eta(z|x)}\Big[\log p(x, z) - \log q_\eta(z \mid x)\Big]. \tag{4}$$

This equality is shown in Blei et al. (2017). Computing the ELBO does not require computing the marginal distribution $p(x)$, which is intractable, or the posterior distribution $p(z \mid x)$, which would be circular. In Section 4, we consider an alternative objective to (3), where we instead minimize an *expectation* of the KL divergence with its arguments q and p reversed.

3.1 Amortized Variational Inference

We describe the construction of the family $\mathcal{Q}$. Traditionally in variational inference, the posterior approximation q_η depends on the data x only implicitly, in that η^* is chosen according to (3). In this case, $q_\eta(z \mid x)$ is usually written $q_\eta(z)$, suppressing the dependence on x. When a new data point x^{new} arrives, finding a variational approximation to the posterior $p(z^{new} \mid x^{new})$ requires solving (3) with $x = x^{new}$ through an iterative optimization procedure, which may be computationally expensive.

On the other hand, in *amortized* variational inference (Kingma and Welling, 2013; Rezende et al., 2014), q_η is an explicit function of the data. In our case, this means a flexible, parameterized function maps input x, an observed image, to a real-valued vector characterizing a distribution on the latent space $\mathcal{Z}$. Typically, the function is a neural network, in which case the variational parameters η are the neural network weights. After the neural network is fitted using a collection of observed x's, the approximate posterior $q_\eta(z^{new} \mid x^{new})$ for a new data point x^{new} can be evaluated with a single forward pass through the neural network. No additional run of an optimization routine is needed for a new data point x^{new}.

The following subsections detail the construction of our variational distribution, which will be fitted in an amortized fashion.

3.2 The Factorization

To make optimization tractable, the family $\mathcal{Q}$ is normally restricted to probability distributions without conditional dependencies between some latent variables. In the most extreme case, known as mean-field variational inference, the variational distribution completely factorizes across all latent variables.

Our factorization has a spatial structure. First, we partition the full $H \times W$-pixel image into disjoint $R \times R$-pixel tiles. R is chosen such that the probability of having three or more

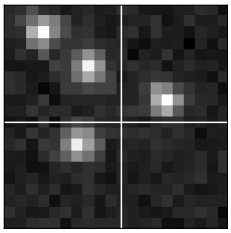

Figure 1: Tiling a 20×20 pixel image into four 10×10 tiles.

stars in one tile is small. In this way, the cataloging problem decomposes to inferring only a few stars at a time (Section 3.4).

Let $S = {}^{H}/_{R}$ and $T = {}^{W}/_{R}$ and assume without loss of generality that H and W are multiples of R. For $s = 1, ..., S$ and $t = 1, ..., T$, the tile $\tilde{x}_{st}$ is composed of the pixels

$$\tilde{x}_{st} = \{x_{hw} : Rs \leq h < R(s+1) \text{ and } Rt \leq w < R(t+1)\}.$$

Figure 1 gives an example with $R = 2$.

Let $\tilde{N}^{(s,t)}$ be the number of stars in tile (s, t). Because $\tilde{N}^{(s,t)}$ is random, the cardinality of the set of locations and fluxes in each tile is also random. To handle the trans-dimensional parameter space, we consider *triangular arrays* of latent variables for each tile:

$$\tilde{\ell}^{(s,t)} = (\tilde{\ell}^{(s,t)}_{N,i} : i = 1, ..., N; N = 1, 2, ...),$$
$$\text{and } \tilde{f}^{(s,t)} = (\tilde{f}^{(s,t)}_{N,i} : i = 1, ..., N; N = 1, 2, ...),$$

where $\tilde{\ell}^{(s,t)}_{N,i}$ and $\tilde{f}^{(s,t)}_{N,i}$ are the elements of the triangular array corresponding to location and fluxes, respectively. Tile locations $\tilde{\ell}^{(s,t)}_{N,i} \in [0, R] \times [0, R]$ give the location of stars within a tile. The fluxes $\tilde{f}^{(s,t)}_{N,i}$ are vectors in $\mathbb{R}^B_+$ (one flux for each band).

We refer to $(\tilde{N}^{(s,t)}, \tilde{\ell}^{(s,t)}, \tilde{f}^{(s,t)})^{S,T}_{s=1,t=1}$ as the *tile latent variables*. The distribution on tile latent variables factorize over image tiles:

$$\tilde{q}_\eta\big((\tilde{N}^{(s,t)}, \tilde{\ell}^{(s,t)}, \tilde{f}^{(s,t)})^{S,T}_{s=1,t=1} \mid x\big) = \prod_{s=1}^{S}\prod_{t=1}^{T} \tilde{q}_\eta\big(\tilde{N}^{(s,t)}, \tilde{\ell}^{(s,t)}, \tilde{f}^{(s,t)} \mid x\big).$$

We denote tile latent variables as $\tilde{z}$. The ultimate latent variable of interest is $z = \{N, (\ell_i, f_{i,1}, ..., f_{i,B})^N_{i=1}\}$, the catalog for the full image. There is a mapping from $\tilde{z}$ to z. First, the number of stars in the full catalog is given by the sum of the stars in each tile, $N = \sum_{s,t} \tilde{N}^{(s,t)}$. Then, for every tile (s, t), we index into the $\tilde{N}^{(s,t)}$-th row of the triangular array of tile latent variables $\tilde{f}^{(s,t)}$ and $\tilde{\ell}^{(s,t)}$. The union of these fluxes and locations over all tiles form the full catalog (tile locations are shifted by the position of the tile in the full image to obtain locations in the full image). See Figure 2 for a schematic.

If τ is the mapping from $\tilde{z}$ to z, then the variational distribution on catalogs z is

$$q_\eta(z \mid x) := \tilde{q}_\eta(\tau^{-1}(z) \mid x), \tag{5}$$

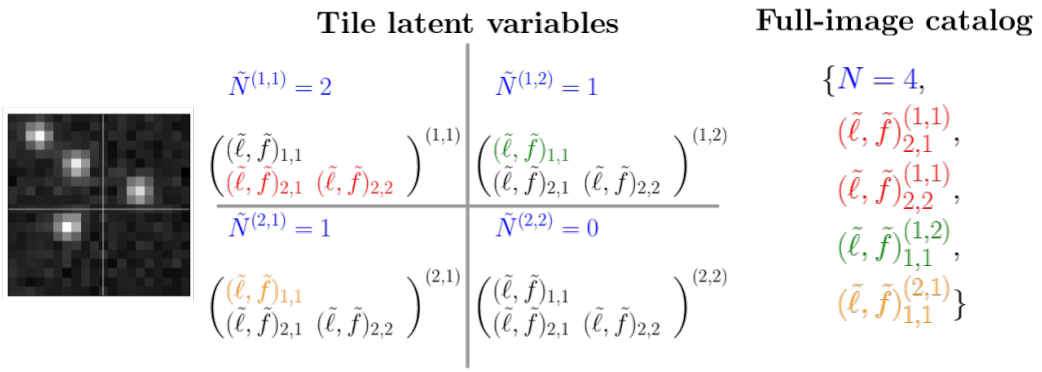

Figure 2: An example image with four tiles and four stars illustrating the relationship between the tile latent variables and the full-image catalog. To construct the full-image catalog, we index into the appropriate row of the triangular array for each tile.

where $\tau^{-1}(z)$ is the pre-image of z under τ. See Appendix A for details on evaluating $q_\eta(z \mid x)$ for any given catalog z, which by (5) requires finding the pre-image $\tau^{-1}(z)$.

3.3 Variational Distributions on Image Tiles

We describe the variational distribution for each tile, $\tilde{q}_\eta\big(\tilde{N}^{(s,t)}, \tilde{\ell}^{(s,t)}, \tilde{f}^{(s,t)} \mid x\big)$. The latent variables fully factorize within each tile. Dropping the index (s,t) in this subsection,

$$\tilde{N} \sim \text{Categorical}(\omega; 0, ..., N_{max}); \tag{6}$$

$$\tilde{\ell}_{j,i}/R \sim \text{LogitNormal}(\mu_{\ell_{j,i}}, \text{diag}(\nu_{\ell_{j,i}})); \tag{7}$$

$$\tilde{f}^b_{j,i} \sim \text{LogNormal}(\mu_{f^b_{j,i}}, \sigma^2_{f^b_{j,i}}), \tag{8}$$

independently for $i = 1, ..., j$; $j = 1, ..., N_{max}$. Here ω is a $(\tilde{N}_{max} + 1)$-dimensional vector on the simplex. $\mu_{\ell_{j,i}}$ and $\nu_{\ell_{j,}}$ are two-dimensional vectors—the covariance on locations is diagonal. Note that in the exact posterior, $\tilde{N}$ has support on the nonnegative integers, whereas in the variational distribution $\tilde{N}$ is truncated at some large N_{max}.

These distributions were taken to match the constraints of the latent variables: fluxes are positive and right skewed, suggesting a log-normal; locations are between zero and R, suggesting a scaled logit-normal.

3.4 Neural Network Architecture

In each tile, the distributional parameters in (6), (7), and (8) are the output of a neural network. The input to the neural network is an $R \times R$ tile, padded with surrounding pixels. Padding enables the neural network to produce better predictions inside the tile. For example, a bright source outside but in the vicinity of the tile affects the pixel values inside the tile. Padding the tiles allows the neural network access to this information. Thus,

while the distribution on tile latent variables factorize over tiles, the neural network is able to use information from neighboring tiles in producing the distributional parameters.

The appropriate amount of padding will depend on the PSF width in the analyzed image. To catalog the crowded starfield M2 (Section 6.1), we set $R = 2$ and padded the tile with a three-pixel-wide boundary. In cataloging a DECam image, we use larger tiles with more padding because the width of the PSF is larger in these images. There, we set $R = 10$ and used a five-pixel-wide boundary.

In amortized inference, the variational parameters η are neural network weights. The architecture consists of a convolutional layer followed by several residual network layers, which themselves contain convolutions, before ending with several fully connected layers (Figure 3). This architecture has been successful on image classification challenges such as ImageNet (Russakovsky et al., 2015). We tuned the architecture using Optuna, an automatic hyper-parameter optimization package (Akiba et al., 2019). Our search included the number of convolution layers, the number of fully connected layers, the number of channels in the convolution layers, and the size of the fully connected layers.

An input to the network is a padded tile, which consists of B color bands. We also append an additional "band" to the input, which is a one-hot encoding with ones for pixels inside the tile, and zeros outside. We do this because the network is only responsible for inferring sources inside each tile, and this additional band gives the network access to a feature which encodes the tile interior.

See Appendix G for further details concerning the parameters of our neural network architecture.

Note that the output dimension of the neural network is quadratic in N_{max}: the outputs are parameters for a triangular array consisting of $\frac{1}{2}(N_{max}^2 + N_{max})$ sources. Factorizing the variational distribution spatially keeps the output dimension manageable. While the full image may contain many stars (on the images that we catalog, the number of stars is on the order of thousands), we set $N_{max} = 3$ for each tile. Thus, the network is responsible for inferring only a few stars at once—a much easier task than inferring all stars simultaneously.

We emphasize that while the variational distribution factorizes over tiles, our method does not break the inference problem for the full image into isolated subproblems. The likelihood of the full image does not factorize over tiles. Light from a star within a tile spills over into neighboring tiles, so the likelihood should not and does not decouple across image tiles.

4. The Expected Forward Kullback-Leibler Divergence

Procedures such as black-box variational inference (BBVI) (Ranganath et al., 2014) and automatic-differentiation variational inference (ADVI) (Kucukelbir et al., 2017) optimize the ELBO without the need for deriving analytic expressions for the expectation over q_η. These approaches employ stochastic gradient descent (SGD); they sample latent variables from q_η and produce an unbiased estimate for the gradient of the ELBO by taking advantage of modern automatic differentiation tools. ADVI is closely related to the reparameterization trick (Spall, 2003; Kingma and Welling, 2013; Rezende et al., 2014), which is often used to fit variational autoencoders and applies when the latent variables are continuous.

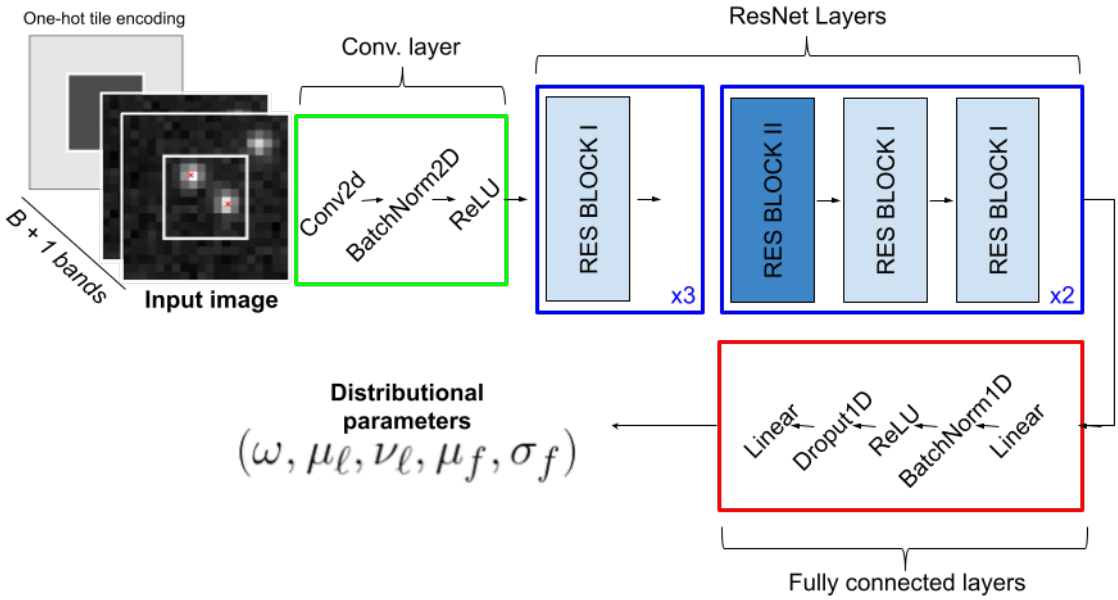

Figure 3: The neural network architecture. The DECam input is a two-band 20×20 padded tile, and the network returns distributional parameters corresponding to sources located in the 10×10 tile outlined in white. In this example, there are two sources located in the 10×10 tile. The additional input band is a one-hot encoding with ones for pixels inside the tile, and zeros outside. For further details concerning the residual network blocks (in blue), see Appendix G.

In our model, the number of stars N is discrete. The REINFORCE estimator (Williams, 1992) is one way to produce an unbiased stochastic gradient for both continuous and discrete latent variables. However, REINFORCE gradients often suffer from high variance in practice, even with the introduction of control variates, resulting in slow convergence of stochastic optimization. We find this to be true in our empirical study (Section 5).

The key difficulty in constructing stochastic gradients of the ELBO is that the integrating distribution depends on the optimization parameter η. We instead maximize a negated expectation of the "forward" KL divergence:

$$\mathcal{L}_{fwd}(\eta) := -\mathbb{E}_{x \sim p(x)}\Big[\mathrm{KL}(p(z \mid x) \| q_\eta(z \mid x))\Big], \tag{9}$$

an objective that appears in the "sleep phase" of the wake-sleep algorithm (Hinton et al., 1995; Bornschein and Bengio, 2014; Le et al., 2020) and was re-introduced by Ambrogioni et al. (2019), who called this approach "Forward Amortized Variational Inference."

Section 4.1 details a simple gradient estimator for (9) that does not require reparameterization or REINFORCE.

There are two key differences between the expected forward KL objective (9) and the ELBO (4). First, recall that maximizing the ELBO is equivalent to minimizing $\mathrm{KL}(q\|p)$; the KL in $\mathcal{L}_{fwd}$ transposes the arguments. Second, the outer expectation over $p(x)$ in $\mathcal{L}_{fwd}$ gives a different meaning to the objective. The ELBO objective seeks η to minimize the KL between $q_\eta(z \mid x)$ and $p(z \mid x)$ for fixed, observed data x, in this case the $H \times W$ image. In contrast, minimizing $\mathcal{L}_{fwd}$ minimizes the KL on average over all possible data x, as weighted by $p(x)$. The target is no longer an approximate posterior for the observed data, but rather an approximate posterior that is "good on average" over all possible data under the model $p(x)$.

4.1 Decomposing the Expected Forward KL

In this subsection, we decompose the expected forward KL objective $\mathcal{L}_{fwd}$ to obtain an unbiased stochastic gradient for stochastic gradient descent.

First, observe that optimizing $\mathcal{L}_{fwd}$ does not require computing the intractable term $p(x)$:

$$\begin{aligned}
\arg\max_\eta \ \mathcal{L}_{fwd}(\eta) &= \arg\min_\eta \ \mathbb{E}_{x\sim p(x)}\Big[\mathrm{KL}(p(z \mid x)\|q_\eta(z \mid x)\Big] \\
&= \arg\min_\eta \ \mathbb{E}_{p(x)}\Big[\mathbb{E}_{p(z|x)}\Big(\log p(z \mid x) - \log q_\eta(z \mid x)\Big)\Big] \\
&= \arg\min_\eta \ \mathbb{E}_{p(x,z)}\Big[-\log q_\eta(z \mid x)\Big].
\end{aligned}$$

Notice the integrating distribution $p(x, z)$ does not depend on the optimization parameter η. Thus, unbiased stochastic gradients can be obtained as

$$g = -\nabla_\eta \log q_\eta(z \mid x) \quad \text{for } (x, z) \sim p(x, z).$$

In other words, at each iteration of SGD, we simulate complete data (x, z) from the generative model and evaluate the loss $-\log q_\eta(z \mid x)$. Here, "complete data" refers to the image along with its catalog. This loss encourages the neural network to map an image x to a distribution $q_\eta(\cdot \mid x)$ that places large density on the image's catalog z.

We decompose the loss $-\log q_\eta(z \mid x)$ further. Recall that q_η fully factorizes over tile latent variables, and thus $-\log q_\eta(z \mid x)$ is a summation over all tile latent variables. To evaluate $-\log q_\eta(z \mid x)$ for some $(x, z) \sim p$, first convert z to its tile parameterization $(\tilde{N}^{(s,t)}, \tilde{\ell}^{(s,t)}, \tilde{f}^{(s,t)})_{s=1,t=1}^{(S,T)}$, as detailed in Appendix A. For each tile (s, t), the variational distribution on the number of stars $\tilde{N}^{(s,t)}$ is categorical with probability vector $\omega^{(s,t)} \in \Delta^{N_{max}}$ (recall Section 3.3). The loss function for the number of stars becomes

$$-\log q_\eta(\tilde{N}^{(s,t)} \mid x) = -\sum_{n=0}^{\tilde{N}_{max}} 1\{\tilde{N}^{(s,t)} = n\} \log \omega_n^{(s,t)}. \tag{10}$$

The vector $\omega^{(s,t)}$ is the output of the neural network, and (10) is the usual cross-entropy loss for a multi-class classification problem.

Next, recall that in the variational distribution location coordinates are logit-normal and fluxes are log-normal. Let y generically denote either the logit-location or log-flux for

a star in the sampled catalog z; let $(\hat{\mu}, \hat{\sigma}^2)$ be the Gaussian mean and variance returned by the neural network. Then the loss for these latent variables is,

$$-\log q_\eta(y \mid x) = \frac{1}{2\hat{\sigma}^2}(y - \hat{\mu})^2 + \frac{1}{2}\log(2\pi\hat{\sigma}^2). \tag{11}$$

The first term encourages network predictions $\hat{\mu}$ to be close to the sampled latent variable y, while $\hat{\sigma}^2$ encodes the uncertainty of the network: the second term encourages small uncertainties, but is balanced by the scaling of the error $(y - \hat{\mu})^2$ in the first term.

The losses in (10) and (11) show that the expected forward KL objective results in a supervised learning problem on complete data sampled from our generative model: the objective function for the number of stars is the usual cross-entropy loss for classification, while the objective function for log-fluxes and logit-locations are L_2 losses in the mean parameters.

5. Empirical Comparison of KL Objectives

A simple example demonstrates that there exist shallow local optima in the ELBO where the fitted approximate posterior is far in KL divergence from the exact posterior. These local optima result in unreliable catalogs. The expected forward KL, by taking advantage of complete data, appears to have a more favorable optimization landscape. The simulated 20×20 single-band image x_{test} is shown in Figure 4(d).

We compare three approaches to deblending. The first two approaches directly optimize the ELBO,

$$\mathcal{L}_{elbo}(\eta; x) = \mathbb{E}_{q_\eta(z|x)}\Big[\log p(x, z) - \log q_\eta(z \mid x)\Big], \tag{12}$$

evaluated at $x = x_{test}$. The third approach minimizes the expected forward KL (9). In each case, q_η is the inference network from Section 3.4. The input to the network is a 10×10 tile with no padding.

Note that the expected forward KL does not depend on x_{test}. Optimizing $\mathcal{L}_{fwd}$ only requires sampling catalogs from the prior and simulating images conditional on each catalog. The prior on the number of stars per image was set to be Poisson with mean $\mu = 4$.

Figure 4 (top row) charts the test ELBO (12) as optimization proceeds in our three approaches. The first approach optimizes the ELBO with SGD and a REINFORCE plus control-variate gradient estimator (Ranganath et al., 2014). The path of the ELBO objective in this first approach is irregular, likely due to the high variance of the REINFORCE gradient estimator, and the optimization does not appear to converge (Figure 4a). For a lower-variance gradient estimator, the second approach employed the reparameterized gradient. To employ this gradient estimator, we analytically integrated the ELBO with respect to the number of stars N to remove the discrete random variable. See Appendix B for details about the gradient estimators. Using reparameterized gradients instead of REINFORCE gradients enabled the optimization to converge to stationary points (Figure 4b). However, for two randomly initialized restarts, the optimization found local optima where the negative ELBO is notably higher than other restarts.

These shallow local optima in the ELBO result in unreliable catalogs. The bottom row of Figure 4 displays the estimated locations, defined as the mode of the fitted variational

distribution. Figure 4(e) shows these locations after converging to a shallow local optimum. Here, the upper left tile was correctly estimated to have two stars, but both estimated stars were incorrectly placed at the same location. (One of the locations should be placed on the second star.) To move one of the estimated locations to the second star, the optimization path must traverse a region where the log-likelihood is lower than the current configuration (Figure 5). The displayed configuration is a local optimum where the gradient with respect to its locations is approximately zero.

On the other hand, using the expected forward KL does not directly optimize the test ELBO. However, the test ELBO increases nonetheless, because the variational posterior better approximates the exact posterior as the optimization proceeds. Optimizing $\mathcal{L}_{fwd}$ consistently converged to a similar ELBO across all restarts and avoided shallow local optima (Figure 4c). At each iteration of SGD, the evaluated loss is quadratic in the logit-location estimate μ_ℓ (11), and the gradient does not vanish. By avoiding shallow local optima, the variational distribution fit with the forward KL always placed its mode on the four true stars in our trials. An example of successful detection by fitting with $\mathcal{L}_{fwd}$ is shown in Figure 4(f).

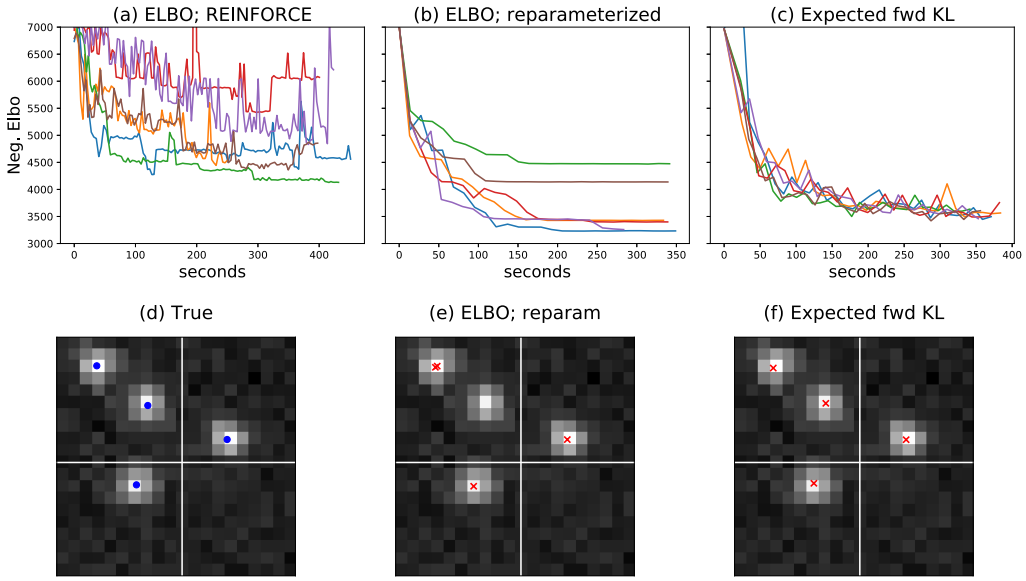

Figure 4: (Top row) The negative ELBO as the optimization progresses for six random restarts. (Bottom row) In red, modal locations from ELBO-optimized and $\mathcal{L}_{fwd}$-optimized variational posteriors, for one of the six restarts. In blue, the true locations.

Finally, note that low-variance gradients of the ELBO for this simple example were constructed by analytically integrating out N, and this was only possible because the image consisted of only four tiles. For each tile, the variational distribution has support over only 0, 1, or 2 stars. Since the variational distribution factorizes over the four tiles, integrating N is a summation of $3^4 = 81$ terms. On larger images with more tiles, analytically integrating

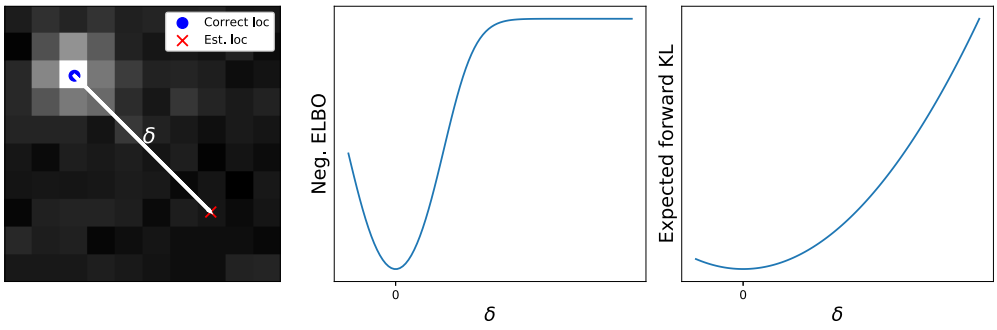

Figure 5: An illustration of local optima in the ELBO objective. To move an estimated location to a correct location, the optimization path must traverse a region where the negative ELBO is flat, with near-zero gradients. In contrast, when optimizing the expected forward KL with SGD, each iteration evaluates a quadratic loss between a true and estimated location, and the gradient does not vanish.

N would be computationally infeasible, and the standard reparameterization trick would not apply as it does in this small illustrative example.

6. Results on Astronomical Surveys

We evaluate StarNet on two distinct surveys. First, we catalog an SDSS image of the Messier 2 (M2) globular cluster. We evaluate the catalog quality by validating against data collected from the Hubble Space telescope, which we use as a ground truth.

Then, we run StarNet on a high-resolution DECam image of the galactic plane of the Milky Way. We demonstrate the ability of StarNet to scale to larger astronomical surveys.

6.1 Results on the M2 Globular Cluster

The M2 globular cluster is a crowded starfield found in field 136 of camera column 2 in run 2583 of the SDSS survey. M2 was also imaged in the ACS Globular Cluster Survey (Sarajedini et al., 2007) using the Hubble Space Telescope (HST), which has greater resolution than the SDSS telescope. The resolution of the HST wide-field channel is 0.05 arcseconds per pixel versus 0.40 arcseconds per pixel in SDSS (ESAHubble, 2021; SDSS, 2020). For this image, the catalog from the HST survey (henceforth the "HST catalog") serves as ground truth for validating our results.

We first analyze the 100×100 pixel subimage of M2 that Portillo et al. (2017) and Feder et al. (2020) analyzed with their MCMC-based approach, PCAT. This subimage shows a region located outside the heavily saturated core of the cluster (Figure 6). Nonetheless, in this subimage the HST catalog contains 1114 stars with F606W-band magnitudes less than 22. We include two bands in our model, the SDSS r-band and i-band. The SDSS r-band and the Hubble F606W band are centered at roughly the same wavelength, while the wavelength range of the Hubble F606W band is slightly broader.

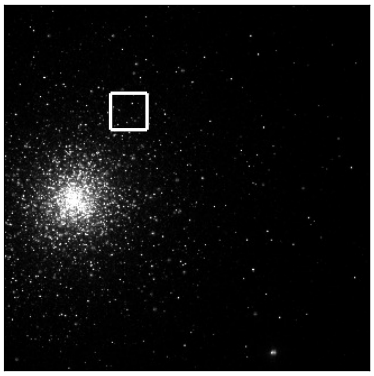

Figure 6: The M2 globular cluster as imaged by SDSS. In white is the 100×100 subregion cataloged by PCAT in Feder et al. (2020).

We compare the cataloging accuracy of StarNet against PCAT, the aforementioned MCMC-based approach that uses the same generative model as StarNet; and DAOPHOT, an algorithmic routine for detecting stars in crowded starfields which does not use a probabilistic model (Stetson, 1987). DAOPHOT convolves the observed image with a Gaussian kernel and scans for peaks above a given threshold. The DAOPHOT catalog of M2 was reported in An et al. (2008).

To evaluate the three methods, we filtered the ground truth HST catalog to stars with magnitudes smaller than 22.5 in the Hubble F606W band (note that smaller magnitude corresponds to brighter stars), because none of the three methods were able to detect stars with lower apparent brightness in the SDSS image.

Estimated catalogs are evaluated on three metrics: the true positive rate (TPR), or recall; the positive predictive value (PPV), or precision; and the F1 score. The TPR is the proportion of true stars in the HST catalog matched with a predicted star in the estimated catalog. The PPV is the proportion of predicted stars in the estimated catalog matched with a true star in the HST catalog. The F1 score summarizes the two metrics as the harmonic mean of the PPV and the TPR.

Like Portillo et al. (2017) and Feder et al. (2020), we define a "match" between an estimated star and an HST star as follows: (1) the estimated location and the HST location are within 0.5 SDSS pixels, and (2) the estimated SDSS r-band flux and the HST F606W band flux are within half a magnitude.

In probabilistic cataloging (PCAT and StarNet), the posterior defines a distribution over catalogs. For StarNet, the TPR, PPV, and F1 score were computed for the catalog corresponding to the mode of the variational distribution (henceforth, the StarNet catalog). For PCAT, 300 catalogs were sampled using MCMC; the metrics were computed for each sampled catalog and averaged.

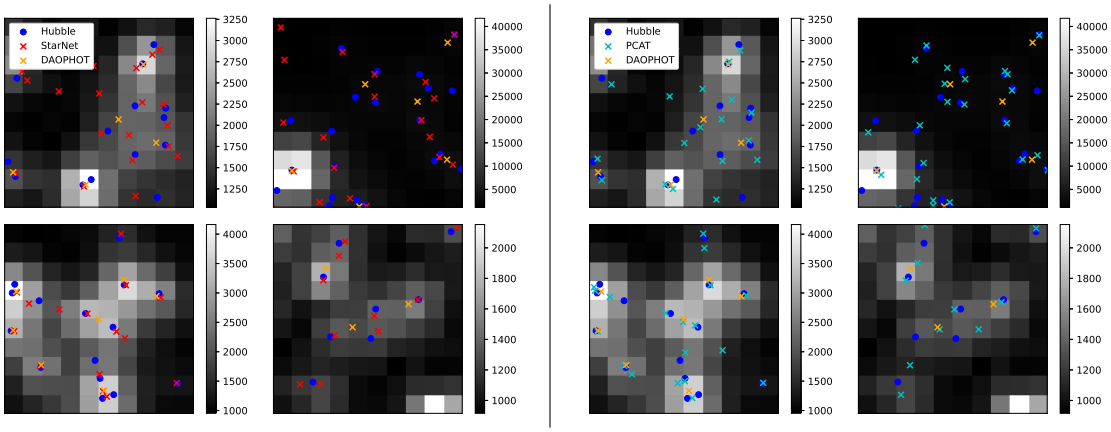

Figure 7: Estimated catalogs on four 10×10 subimages from M2. Blue dots are stars from the HST catalog used as ground truth. StarNet, PCAT, and DAOPHOT estimated stars are shown as red, cyan, and orange crosses, respectively.

StarNet produced a catalog that outperforms DAOPHOT and PCAT in F1 score (Table 1). Figure 7 compares the StarNet catalog to the PCAT, DAOPHOT, and HST catalogs. DAOPHOT estimated less than half the number of stars when compared to the other methods. It therefore had a large PPV but a small TPR. The StarNet catalog had similar TPR as PCAT while having an 11% higher PPV.

The improvement of StarNet over PCAT in PPV was most pronounced for the brightest stars (Figure 8), suggesting that some of the brightest stars in the PCAT catalog may have in truth been collections of blended stars. The TPR for StarNet was uniformly better than DAOPHOT across all magnitudes. Of all methods, StarNet best approximated the HST flux distribution (Figure 9).

Table 1 also shows the number of stars inferred by each method. There are 1114 stars in the HST catalog. For probabilistic methods (StarNet and PCAT), we display the mean number of stars under the approximate posterior, along with the 5th and 95th percentiles. We compute the StarNet posterior mean and quantiles by sampling from the variational posterior. Recall that on each tile, the variational posterior on the number of stars is a categorical random variable; to construct a distribution for the number of stars on the whole image, we first sample from the per-tile categorical distribution, then sum over all tiles. StarNet posterior intervals were three times wider than the PCAT intervals. The small PCAT intervals may indicate that the MCMC sampler failed to mix well. While neither the StarNet intervals nor the PCAT intervals cover the ground truth, though the StarNet intervals come closer to doing so. For StarNet, we attribute the over-estimated number of stars by StarNet to the tiling structure of the approximate posterior (Appendix C.2).

In a subsequent experiment, we go beyond the 100×100 subimage cataloged by Feder et al. (2020) and catalog the entire M2 globular cluster contained in a 1000×1000-pixel image (Figure 6). We produce a color-magnitude diagram on this entire region (Figure 10). On the entire region, a second distinct cluster, shifted to the right in color, appears in addition to the main sequence of stars. The second cluster becomes more apparent after

Method	TPR	PPV	F1 score	#Stars mean	#Stars (q-5%, q-95%)
DAOPHOT	0.20	0.65	0.31	357	–
PCAT	0.55	0.37	0.44	1672	(1664, 1680)
StarNet (our)	0.53	0.48	**0.50**	1462	(1430, 1497)

Table 1: Performance metrics on M2. For probabilistic methods (StarNet and PCAT) the "#stars" columns provide the posterior mean along with the 5th and 95th posterior percentiles for the number of stars. The number of stars in the ground-truth Hubble catalog is 1114.

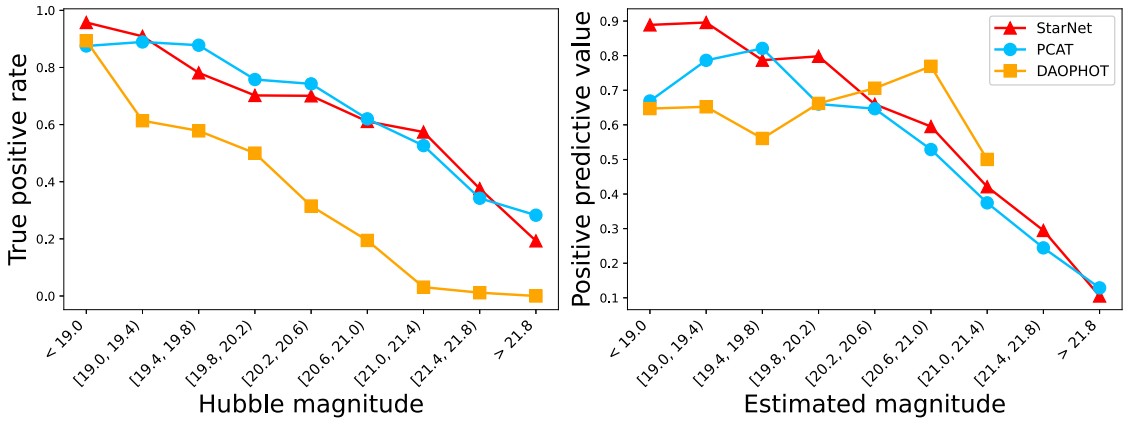

Figure 8: True positive rate (left) and positive predicted value (right) of various cataloging procedures on M2, plotted against r-band magnitude. Smaller magnitudes correspond to brighter stars.

we filter to high-confidence stars in the StarNet catalog, defined as stars with flux posterior standard deviation less than one. This second cluster, corresponding to a collection of red giants, is undetectable without the ability of StarNet to scale to larger images.

However, the patterns are less definite in the StarNet color-magnitude diagram than in the Hubble color magnitude diagram. There is more spread in the StarNet color estimates, particularly at faint magnitudes. This is due to the superior resolution of the Hubble telescope; near the heavily saturated core of the M2 cluster, stars are near impossible to deblend in the SDSS image, and our performance suffers (Appendix F).

6.2 Results on the DECam Survey

We next demostrate StarNet on a larger region of the sky. The DECam survey imaged stars in our own Milky Way, and we chose a 4000×2000 frame centered at coordinates RA $= 266.044°$ and DEC $= -28.88111°$. See Figure 11 for an example image.

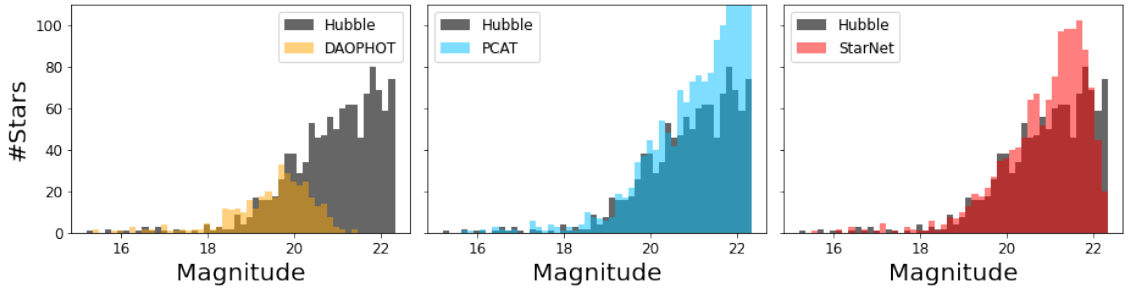

Figure 9: Flux distributions for the r-band observations of M2. The flux distribution of the HST catalog is in grey. Estimated distributions from DAOPHOT, PCAT, and StarNet catalogs are overlaid. For PCAT, the flux distribution is from a single catalog sample.

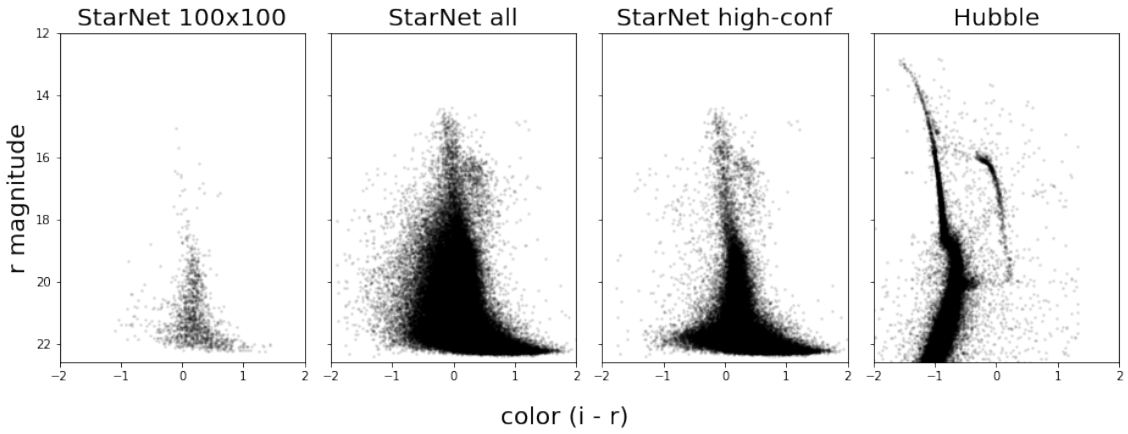

Figure 10: Color-magnitude diagrams from StarNet and Hubble. From left to right, color-magnitude diagrams constructed from: the same 100×100 subimage as was cataloged in Feder et al. (2020); the StarNet catalog derived from the entire 1000×1000 image of M2; that StarNet catalog, filtered to stars with posterior SD(flux) < 1; the Hubble catalog.

The DECam image is somewhat sparser than M2. Thus, we set the Poisson mean parameter of the star density lower smaller than on M2 to fifty stars per 100×100-pixel image. This allowed us to use larger 10×10-pixel tiles with 20×20-pixel padded tiles. We produced a catalog for the full 4000×1000 frame, consisting of $9,000$ stars. The color-magnitude diagram shows a sequence of blue stars that are reddened at fainter magnitudes.

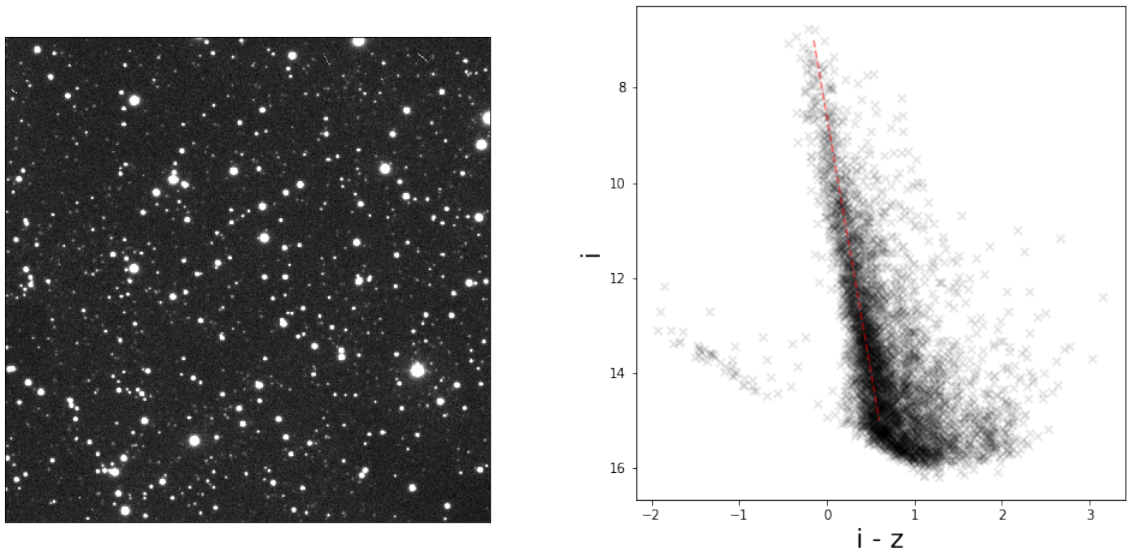

Figure 11: (Left) A 1000 x 1000 pixel subregion of the DECam survey. (Right) Color magnitude diagram for the DECam image. Red dashed line highlights the inferred blue main-sequence stars

6.3 Runtime

We ran SGD to minimize the expected forward KL for 400 epochs; at each epoch, 200 images of size 100×100 pixels were sampled from the generative model. We performed optimization with Adam (Kingma and Ba, 2014). On a single NVIDIA GeForce RTX 2080 Ti GPU, this fitting procedure took one hour.

After fitting the variational posterior, computing the approximate posterior (that is, producing the distributional parameters of the variational approximation) given either the 1000×1000 M2 image or the 4000×2000 DECam image takes less than a second. By comparison, the reported runtime of PCAT, which uses MCMC, is 30 minutes on a 100×100 pixel image (Feder et al., 2020).

The speed at inference time (which excludes training time) gives StarNet the scaling characteristics necessary for processing large astronomical surveys. A single SDSS image is 1489×2048 pixels. Based on the reported 30-minute runtime of PCAT for a 100×100 pixel subimage, we project that the runtime to process the full image would be $30 \text{ min} \times 14 \times 20 = 8400$ minutes, or almost six days. The SDSS survey consists of nearly one million images, and thus scaling PCAT to the entire SDSS survey would be infeasible. The upcoming LSST survey will be 300 times larger than SDSS.

On the other hand, StarNet incurs a one-time cost to fit the variational distribution with synthetic data; this cost is then amortized over a potentially large region of the sky. Re-fitting StarNet may nonetheless be necessary when the model parameters such as the background or PSF change—which is the case for large ground-based astronomical surveys, where data is collected over many nights. The SDSS data processing pipeline, for instance,

estimates a new PSF and background for each new frame. Even assuming a new StarNet refit for each SDSS frame, StarNet is still 100× faster than PCAT.

We can further push the scalability of StarNet by amortizing over a range of model parameters such as the background and PSF. With appropriate priors on these model parameters, fitting StarNet using the expected forward KL enables it to generalize across a diverse set of images and further reduce the need for retraining.

7. Conclusion

StarNet employs forward variational inference and is more accurate than both a recently published MCMC-based probabilistic cataloger and a widely used non-model-based procedure. In the framework of probabilistic modeling, StarNet produces catalog uncertainties captured by a posterior over the set of all catalogs. Importantly, unlike current MCMC approaches, StarNet also has the capacity to scale probabilistic cataloging to process large astronomical surveys.

The quality of StarNet detections is the result of optimizing the forward KL, a different objective than the one traditionally used in variational inference. Optimizing the forward KL allows the variational posterior to be fit on large amounts of complete data (i.e., images along with their latent catalogs) generated from StarNet's statistical model.

While this work focuses on stars, our methodology can be extended to include more general light sources, such as galaxies. One promising direction is to incorporate a highly accurate deep generative model of galaxies (Regier et al., 2015; Reiman and Göhre, 2019; Lanusse et al., 2021; Arcelin et al., 2021) into the StarNet model. The statistical framework in this research lays the foundation for building flexible models to incorporate the cataloging of other celestial objects.

Future astronomical surveys will produce far more data than past surveys. As telescopes peer deeper into space, fields will reveal more sources and images will become more crowded. The uncertainties in crowded fields necessitate a probabilistic approach. Our method holds the promise of providing a scalable inference tool that can meet the challenges of future surveys.

Acknowledgments

RL acknowledges support from the NSF Graduate Research Fellowship Program. JR acknowledges support for this work from the National Science Foundation (OAC-2209720) and the Department of Energy (DE-SC0023714). This paper has been approved by the LSST Dark Energy Science Collaboration following an internal review. The internal reviewers were Bastien Arcelin, François Lanusse, and Peter Melchior. The authors also thank Derek Hansen, Ismael Mendoza, and Zhe Zhao for providing thoughtful feedback about this manuscript.

Appendix A. Evaluating the Variational Distribution

Optimizing the expected forward KL requires evaluating $q_\eta(z \mid x)$ for a given catalog

$$z = \{N, (\ell_i, f_{i,1}, ..., f_{i,B})_{i=1}^N\}.$$

By (5), it suffices to evaluate $\tilde{q}_\eta(\tau^{-1}(z) \mid x)$, where τ is the mapping from tile latent variables $\tilde{z}$ to the catalog z as described in Section 3.2, and $\tilde{q}_\eta$ is the distribution on tile latent variables.

Here, $\tau^{-1}(z)$ is a *set* of tile latent variables because the mapping from tile latent variables $\tilde{z}$ to catalogs z is not injective, as we now explain.

Locations in the catalog $\{\ell_i\}_{i=1}^N$ determine the number of stars on tile (s,t). The number of stars $\tilde{N}^{(s,t)}$ is simply the count of the locations that reside within that tile:

$$\tilde{N}^{(s,t)} = \sum_{i=1}^{N} \mathbf{1}\Big\{\ell_i \in [Rs, R(s+1)] \times [Rt, R(t+1)]\Big\},$$

where $\mathbf{1}\{\cdot\}$ is the indicator function, equal to one its predicate is true if true and zero otherwise.

Now, consider $\tilde{\ell}^{(s,t)}$ and $\tilde{f}^{(s,t)}$, the triangular array of locations and fluxes on tile (s,t). For each (s,t), the $\tilde{N}^{(s,t)}$-th row of the triangular array of fluxes and locations is determined by the locations and fluxes of stars imaged in tile (s,t); they are determined by the catalog z. However, the other rows of the triangular arrays are not determined by the catalog z; they are free to take any value in their domain. Therefore, the mapping τ is not injective.

Thus, evaluating the probability of $\tau^{-1}(z)$ under $\tilde{q}_\eta$ requires marginalizing over the rows of the triangular arrays $\ell^{(s,t)}$ and $\tilde{f}^{(s,t)}$ that are not determined by z. However, because $\tilde{q}_\eta$ fully factorizes, the terms where $n \neq \tilde{N}^{(s,t)}$ do not enter the product after marginalization. On each tile (s,t),

$$\begin{aligned}\tilde{q}_\eta\big(\tilde{N}^{(s,t)}, \tilde{\ell}^{(s,t)}, \tilde{f}^{(s,t)} \mid x\big) &= \tilde{q}_\eta(\tilde{N}^{(s,t)} \mid x) \prod_{n=1}^{N_{max}} \tilde{q}_\eta\big((\tilde{\ell}_{n,i}^{(s,t)}, \tilde{f}_{n,i}^{(s,t)})_{i=1}^n \mid x\big) \\ &= \tilde{q}_\eta(\tilde{N}^{(s,t)} \mid x)\tilde{q}_\eta\big((\tilde{\ell}_{n,i}^{(s,t)}, \tilde{f}_{n,i}^{(s,t)})_{i=1}^n \mid x\big)\Big|_{n=\tilde{N}^{(s,t)}}.\end{aligned}$$

In words, given a catalog z, first convert z to tile latent variables; then on each tile, it suffices to evaluate $\tilde{q}_\eta$ only at the rows of triangular arrays determined by the number of stars falling in each tile.

The last technical detail is computing the probability for a given row of a triangular array. Let $(\tilde{\ell}_i, \tilde{f}_i)_{i=1}^n$ generically denote the tile latent variables in the n-th row of a triangular array, on some tile (s,t). Because catalogs are sets, each entry $(\tilde{\ell}_i, \tilde{f}_i)$ in the catalog must be matched with corresponding variational parameters, and the probability of the set $(\tilde{\ell}_i, \tilde{f}_i)_{i=1}^n$ under q is given by the sum over the permutations of the possible matches:

$$q((\tilde{\ell}_i, \tilde{f}_i)_{i=1}^n | x) = \sum_{\pi} \Big\{ \prod_{i=1}^{n} \text{LogitNormal}(\tilde{\ell}_{\pi(i)}; \mu_{\ell_i}, \nu_{\ell_i}) \times \text{LogNormal}(\tilde{f}_{\pi(i)}; \mu_{f_i}, \sigma_{f_i}^2) \Big\}$$

where the sum is taken over all permutations on $\{1, ..., n\}$. This is feasible because on each tile $N_{max} = 3$, so we only need to enumerate $3! = 6$ possibilities.

Appendix B. Reparameterized and REINFORCE Gradients

The ELBO objective (4) is of the form

$$\mathcal{L}(\eta) = \mathbb{E}_{q_\eta(z)}[f_\eta(z)]. \tag{13}$$

The parameter η is to be optimized, and z is the latent variable. The integrating distribution q and the function f depend on η.

The REINFORCE estimator (Williams, 1992) is a general-purpose unbiased estimate for the gradient of (13). It is given by

$$g_{\text{rf}}(z) = \nabla_\eta f_\eta(z) + f_\eta(z)\nabla_\eta \log q_\eta(z) \quad \text{for } z \sim q_\eta(z).$$

The REINFORCE estimate is unbiased for the true gradient:

$$\begin{aligned}
\mathbb{E}_{q_\eta(z)}[g_{\text{rf}}(z)] &= \int q_\eta(z)\nabla_\eta f_\eta(z)\, dz + \int q_\eta(z) f_\eta(z)\nabla_\eta \log q_\eta(z)\, dz \\
&= \int q_\eta(z)\nabla_\eta f_\eta(z)\, dz + \int f_\eta(z)\nabla_\eta q_\eta(z)\, dz \\
&= \int \nabla_\eta [q_\eta(z) f_\eta(z)]\, dz \\
&= \nabla_\eta \int q_\eta(z) f_\eta(z)\, dz = \nabla_\eta \mathbb{E}_{q_\eta(z)}[f_\eta(z)],
\end{aligned}$$

assuming that f is well-behaved so that integration and differentiation can be interchanged.

In many applications, the REINFORCE estimator has too high variance to be useful. One way to lower the variance is to introduce a control variate C (Ranganath et al., 2014), and estimate the gradient as

$$g_{\text{cv}}(z) = \nabla_\eta f_\eta(z) + (f_\eta(z) - C)\nabla_\eta \log q_\eta(z) \quad \text{for } z \sim q_\eta(z).$$

This estimate remains unbiased because the score function $\nabla_\eta \log q_\eta(z)$ is zero mean under q.

A simple but often effective choice of control variate is to let C be a second evaluation of f_η at an independently drawn $z' \sim q$:

$$g_{\text{cv}}(z) = \nabla_\eta f_\eta(z) + (f_\eta(z) - f_\eta(z')\nabla_\eta \log q_\eta(z) \quad \text{for } z, z' \overset{\text{iid}}{\sim} q_\eta. \tag{14}$$

This estimate is unbiased conditional on z' and hence unconditionally unbiased as well. We use this control variate for our experiments involving the REINFORCE estimator in Section 5.

Alternatively, the reparameterized gradient (Rezende et al., 2014; Kingma and Welling, 2013) can be used when there exists some distribution F not involving η and a differentiable mapping h_η such that

$$w \sim F \implies h_\eta(w) \sim q_\eta.$$

For example, if $q_\eta(z) = \mathcal{N}(z; 0, \eta)$ that is, a Gaussian with zero mean and variance η, one possibility is to let F be the standard Gaussian and $h_\eta(w) = w\sqrt{\eta}$. The gradient of $\mathcal{L}(\eta)$ can then be written as

$$\nabla_\eta \mathbb{E}_{q_\eta(z)}[f_\eta(z)] = \nabla_\eta \mathbb{E}_{w\sim F}[f_\eta(h_\eta(w)] = \mathbb{E}_{w\sim F}[\nabla_\eta f_\eta(h_\eta(w)],$$

again assuming the interchangability of integrals and derivatives. Unbiased gradients arise from the chain rule:

$$g_{\text{rp}} = \nabla_\eta f_\eta(h_\eta(w)) = \nabla_z f_\eta(z)\Big|_{z=h_\eta(w)} \nabla_\eta h_\eta(w) \quad \text{for } w \sim F.$$

The reparameterized gradient includes gradient information $\nabla_z f_\eta(z)$, while the REINFORCE gradient does not. Taking into account the structure of f through its gradient lowers the variance of reparameterized gradient in comparison to the REINFORCE gradient. However, if z contains discrete components, there cannot be a differentiable mapping h_η, and the reparameterization trick will not apply.

B.1 Gradients for the Empirical Comparison of KL Objectives

In experiments of Section 5, we used a combination of reparameterized and REINFORCE plus control variate gradients. Let $\tilde{N}$ be the vector of per-tile number of stars (a discrete component) and y be the locations and fluxes (continuous components) Our variational distribution factorizes, so we write the expectation as

$$\mathcal{L}(\eta) = \mathbb{E}_{q_\eta(\tilde{N})}\mathbb{E}_{q_\eta(y)}[f_\eta(\tilde{N}, y)]. \tag{15}$$

We use the REINFORCE estimator with control variate (14) for the outer expectation and the reparameterization trick for the inner expectation. We first apply REINFORCE to the outer expectation:

$$\begin{aligned}
&\nabla_\eta \mathbb{E}_{q_\eta(\tilde{N})}\mathbb{E}_{q_\eta(y)}\Big[f_\eta(\tilde{N}, y)\Big] \\
&\quad = \mathbb{E}_{q_\eta(\tilde{N})}\Big[\nabla_\eta \log q_\eta(\tilde{N})\mathbb{E}_{q_\eta(y)}\big[f_\eta(\tilde{N}, y) - \mathbb{E}_{q_\eta(\tilde{N})}[f_\eta(\tilde{N}, y)]\big] + \nabla_\eta \mathbb{E}_{q_\eta(y)}[f_\eta(\tilde{N}, y)]\Big] \\
&\quad \approx \nabla_\eta \log q_\eta(\tilde{N})\mathbb{E}_{q_\eta(y)}[f_\eta(\tilde{N}, y) - f_\eta(\tilde{M}, y)] + \nabla_\eta \mathbb{E}_{q_\eta(y)}[f_\eta(\tilde{N}, y)]
\end{aligned} \tag{16}$$

for $\tilde{N}, \tilde{M} \overset{iid}{\sim} q_\eta$. Then we use the reparameterization trick for y, so

$$\begin{aligned}
\mathbb{E}_{q_\eta(y)}[f_\eta(\tilde{N}, y) - f_\eta(\tilde{M}, y)] &\approx f_\eta(\tilde{N}, h_\eta(w)) - f_\eta(\tilde{M}, h_\eta(w)) \\
\nabla_\eta \mathbb{E}_{q_\eta(y)}[f_\eta(\tilde{N}, y)] &\approx \nabla_y f_\eta(\tilde{N}, y)\Big|_{y=h_\eta(w)} \nabla_\eta h_\eta(w)
\end{aligned} \tag{17}$$

for $w \sim F$, where h_η and F are chosen appropriately. Combining (16) and (17), our gradient estimator is

$$g(z) = \nabla_\eta \log q_\eta(\tilde{N})[f_\eta(\tilde{N}, h_\eta(w)) - f_\eta(\tilde{M}, h_\eta(w))] + \nabla_y f_\eta(\tilde{N}, y)\Big|_{y=h_\eta(w)} \nabla_\eta h_\eta(w). \tag{18}$$

Equation (18) is what our main text called the "REINFORCE gradient." These gradients produced the optimization path in Figure 4(a).

The "reparameterized" gradient in Section 5 requires integrating out $\tilde{N}$. Here, we write the outer expectation in (15) as a summation of $4^{N_{max}+1}$ terms (recall our experiments have four tiles, and at most N_{max} stars per tile), with each term representing a different possible assignment of the number of stars to each tile:

$$\mathcal{L}(\eta) = \sum_{\tilde{n}} \mathbb{E}_{q_\eta(y)}[f_\eta(\tilde{n}, y)].$$

Then, the reparameterization trick is applied to each term of the summation. Stochastic gradients are computed as,

$$g(z) = \sum_{\tilde{n}=1} \nabla_y f_\eta(\tilde{n}, y)\Big|_{y=h_\eta(w)} \nabla_\eta h_\eta(w) \quad \text{for } w \sim F.$$

Gradients of this form produced the optimization path in Figure 4(b). No REINFORCE estimates were required.

Notice that to compute these re-parameterized gradients we require a summation of $(S \times T)^{N_{max}+1}$ terms (recall from the main text that $S \times T$ is the total number of tiles in an image). For large images, i.e. those requiring many tiles, computing re-parameterized gradients would be infeasible.

Appendix C. Experiments on Synthetic Data

We present results on a set of synthetic data experiments. We first demonstrate the ability of StarNet to deblend two simulated stars. Next, we revisit the coverage of StarNet credible intervals. Finally, we empirically demonstrate the effect of padded tiles in our architecture.

C.1 Deblending Two Stars

We set up an experiment to study the ability of StarNet to deblend two simulated stars. On a 20×20 pixel image, we simulate two stars of equal flux at distance δ pixels apart, and examine the approximate posterior produced by StarNet (Figure A.1). We generate the stars with the DECam PSF, which has a full width at half maximum (defined as the diameter at which the PSF is half its peak brightness) of 4.2 pixels. The threshold of near-perfect deblending is a distance of $\delta = 1.5$ pixels, which is less than half the PSF full width at half maximum.

C.2 Coverage of Credible Intervals

We have seen that on M2, the StarNet 95% posterior interval did not contain the ground truth number of stars (Section 6.1). On M2, we attribute this to model mis-specification, specifically due to an imperfectly estimated background.

We check the coverage of StarNet posterior intervals on synthetic data to reveal any issues other than model mis-specification that may explain these results. We sample a single 100×100 pixel image from the generative model. On this sampled image, the true number of stars, $N = 1195$, is still considerably smaller than the 0.01-th percentile of the approximate posterior distribution (Figure A.2).

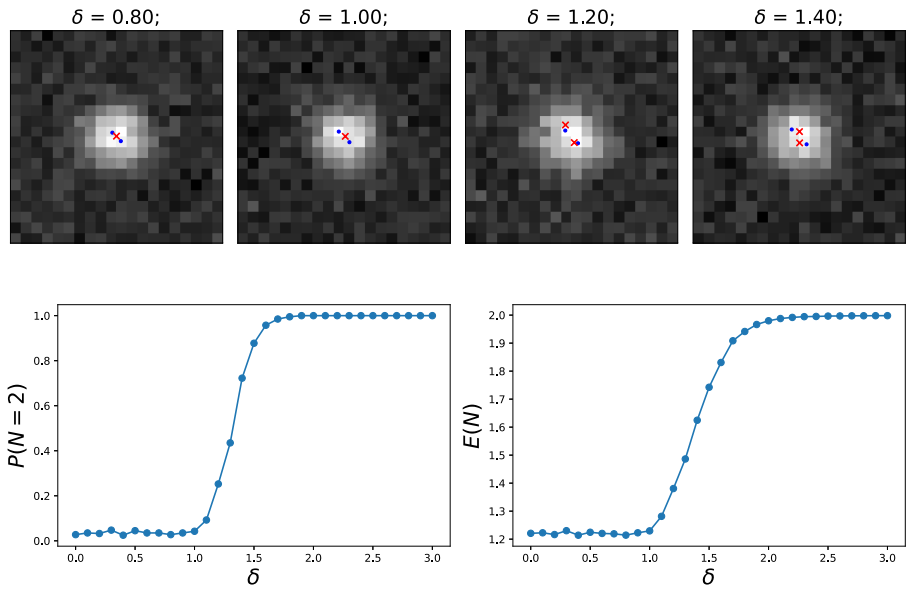

Figure A.1: (Top row) Simulated images with two stars separated by distance δ in pixels. True locations are in blue. StarNet MAP locations in red. In these examples, the StarNet MAP catalog correctly contains two stars when separated by $\delta \geq 1.2$, but only estimated one star when $\delta \leq 1$. (Bottom left) The probability that N, the number of sources in the image, equals two under the variational posterior, as δ varies. (Bottom right) The expectation of N under the variational posterior as δ varies.

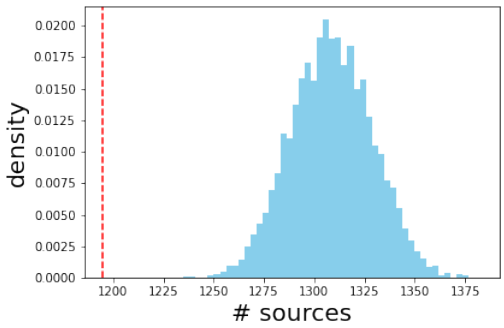

Figure A.2: Distribution of 5000 samples of the number of sources from the StarNet approximate posterior. The true number of sources demarcated in red.

We attribute this over-estimation to the spatial independence of the approximate posterior. Specifically, StarNet *overestimates* the number of sources close to tile boundaries. Heuristically, for a a source located in the interior of the tile within ϵ of a boundary (but

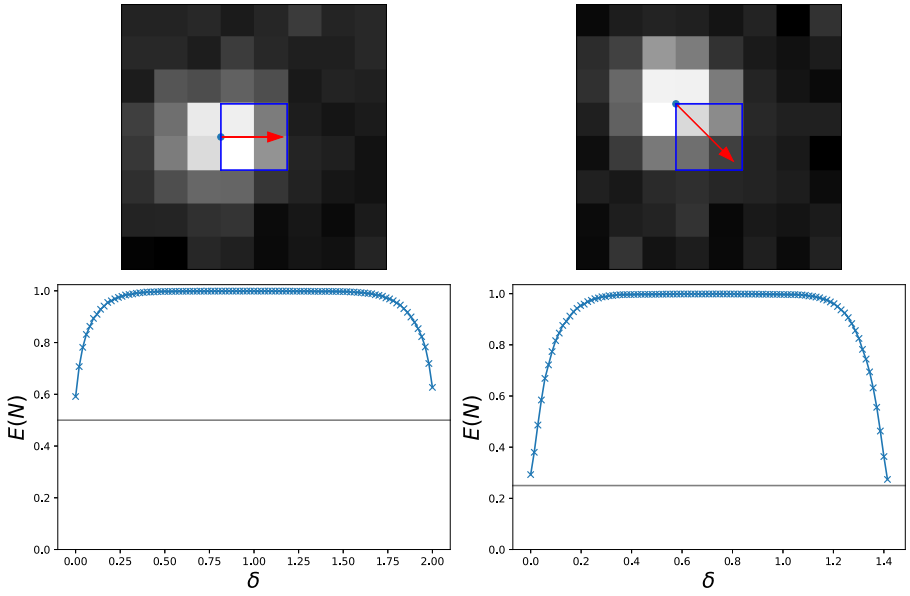

Figure A.3: The expected number of sources under the StarNet approximate posterior as a function of distance from the tile edge (in pixels). On the left, we place a star on the left-most edge, and move its location δ pixels to the right. On the right, we place a star on the top-left corner, and move its location δ pixels towards the bottom-right corner.

still far from a corner), StarNet should assign a probability of $\frac{1}{2}$ for having one source, and a probability $\frac{1}{2}$ for having none, as $\epsilon \to 0$. Should this be the case, then over the entire image, which consist of many tiles, a source on a tile boundary is correctly accounted for: it has a 50-50 chance of being assigned to one tile or another in the approximate posterior, and this source contributes a count of one to the posterior expectation on the number of stars.

However, we observe empirically that as a source approaches the edge of a tile, the posterior probability that $N = 1$ approaches a number slightly *larger* than $\frac{1}{2}$ (Figure A.3). Therefore, the approximate posterior overestimates the expected number of sources in the full image.

To illustrate the effect of tiles, we simulate images with the constraint that all sources are at least 0.1-pixels from all tile edges. In this case, then the StarNet approximate posterior has much closer to correct coverage (Figure A.4)

C.3 Effects of Tile Padding

We study the effect of padding the input tiles to the StarNet neural network architecture. We re-visit the investigation of tile boundary effects described in the subsection above, where we simulate a source closer and closer to the tile boundary. The experiments above

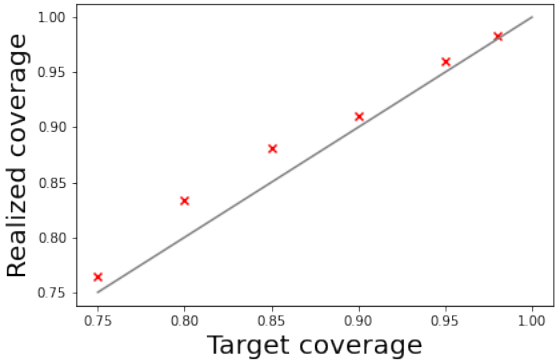

Figure A.4: Coverage test in simulated 100×100 images where all true stars are constrained to be 0.1-pixels away from any tile boundary. We simulate 1000 images from the generative model, and for each image, we compute a $(1-\alpha)$-level posterior interval for the number of stars by taking the $\frac{\alpha}{2}$ and $(1-\frac{\alpha}{2})$-th percentiles of 5000 StarNet posterior samples. For each α, we plot the observed coverage against the target $(1-\alpha)$-level coverage.

	TPR	PPV	F1
0	0.53	0.48	0.50
1	0.52	0.52	0.52
2	0.53	0.57	0.55
3	0.53	0.49	0.51
4	0.53	0.52	0.52
5	0.53	0.53	0.53
6	0.55	0.54	0.55
7	0.51	0.54	0.52
8	0.53	0.52	0.53
9	0.51	0.55	0.53

Table A.1: Performance metrics on M2 for ten random refits of StarNet.

used the M2 network, which uses 2×2 tiles and three pixels of padding. With only one pixel of padding, sources become even more over-estimated on tile edges (Figure A.5).

Appendix D. Sensitivity to Refits

Our optimization procedure uses stochastic optimization. We evaluate the sensitivity of our StarNet M2 performance metrics to ten random refits (Table A.1). The F1 score over the refits range between 50% and 55%. Our comparisons to the performance of other methods, PCAT and DAOPHOT, are unaffected by random reruns.

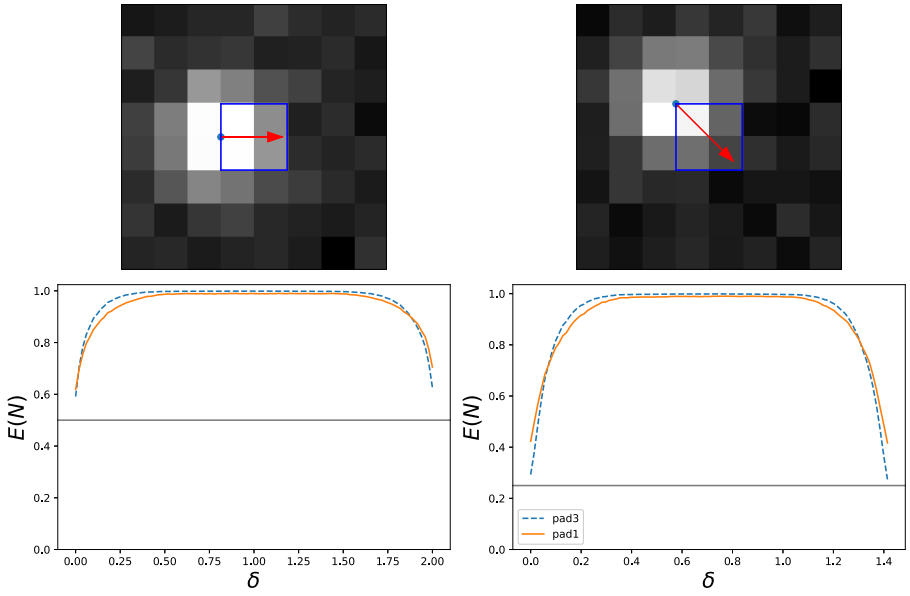

Figure A.5: The same experimental setup as Figure A.3, but we study the effect of padding the neural network input tiles. In dashed blue, the posterior when StarNet uses three pixels of padding. In solid orange, the posterior when only one pixel of padding is used.

Appendix E. Sensitivity to Prior Parameters

We examine the sensitivity of StarNet to prior parameters μ and α on the image M2. Recall μ is the prior mean number of stars per pixel (1); α is the power law slope on the r-band fluxes (2). In the results of Section 6.1, $\mu = 0.12$ and $\alpha = 0.5$.

The model is robust to these prior choices. In fact, the variation in performance metrics due to prior choices is about the same as the variation due to random refits. Thus, for these prior sensitivity experiments, we initialize the neural network weights using the network fit at the original prior.

Over a range of μ between 0.08 (corresponding to an prior average of 800 stars on a 100×100 image) and 0.20 (corresponding to 2000 stars) the F1-score remains steady between 0.49 and 0.51 (Table A.2). Similar robustness in F1 hold when α varies between 0.25 and 1.0 (Table A.3).

Appendix F. Other M2 Subregions

The initial 100×100 subregion of M2 considered in our main paper was located at pixel coordinates (630, 310) in SDSS run 2583, field 136, camera column 6. We evaluate StarNet on two other subregions of M2. The first is another 100×100 pixel subregion of similar density as the original; the second is in the center of globular cluster (Figure A.6).

μ	TPR	PPV	F1 score
0.08	0.47	0.53	0.50
0.10	0.49	0.53	0.51
0.12	0.53	0.48	0.50
0.14	0.50	0.46	0.48
0.16	0.51	0.48	0.50
0.20	0.51	0.48	0.49

Table A.2: Performance metrics on M2 as a function of the prior parameter μ. The smallest μ, $\mu = 0.08$ corresponds to an prior average of 800 stars on a 100×100 image, while the largest, $\mu = 0.2$, corresponds to 2000 stars.

α	TPR	PPV	F1 score
0.25	0.47	0.54	0.51
0.50	0.53	0.48	0.50
0.75	0.56	0.45	0.50
1.00	0.55	0.47	0.51

Table A.3: Performance metrics on M2 as a function of the prior parameter α.

The performance metrics on the center of the globular cluster suggest that deblending in this region is nearly impossible — there are $15,000$ stars in this 100×100 subregion, averaging to more than one star per pixel and is ten times as dense as the region considered in the main text.

In all the considered regions, our comparison with DAOPHOT is unchanged, and we continue to outperform DAOPHOT in F1 score (Table A.4).

Region	Method	TPR	PPV	F1 score	#stars	(q-5%, q-95%)	True #stars
(A)	DAOPHOT	0.20	0.65	0.31	357	–	1114
(A)	StarNet	0.53	0.48	0.50	1462	(1430, 1497)	1114
(B)	DAOPHOT	0.21	0.65	0.32	310	–	941
(B)	StarNet	0.56	0.44	0.49	1384	(1352, 1416)	941
(C)	DAOPHOT	0.003	0.19	0.007	293	–	15094
(C)	StarNet	0.08	0.31	0.13	3306	(3258, 3355)	15094

Table A.4: Performance metrics of StarNet and DAOPHOT on the three regions labeled "A", "B", and "C" in Figure A.6.

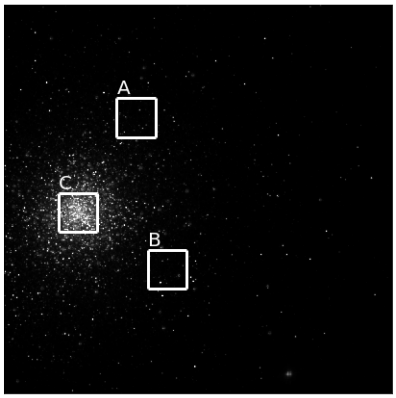

Figure A.6: Subregions of M2 for the performance metrics in Table A.4. The subregion labeled "A" was cataloged in the main text.

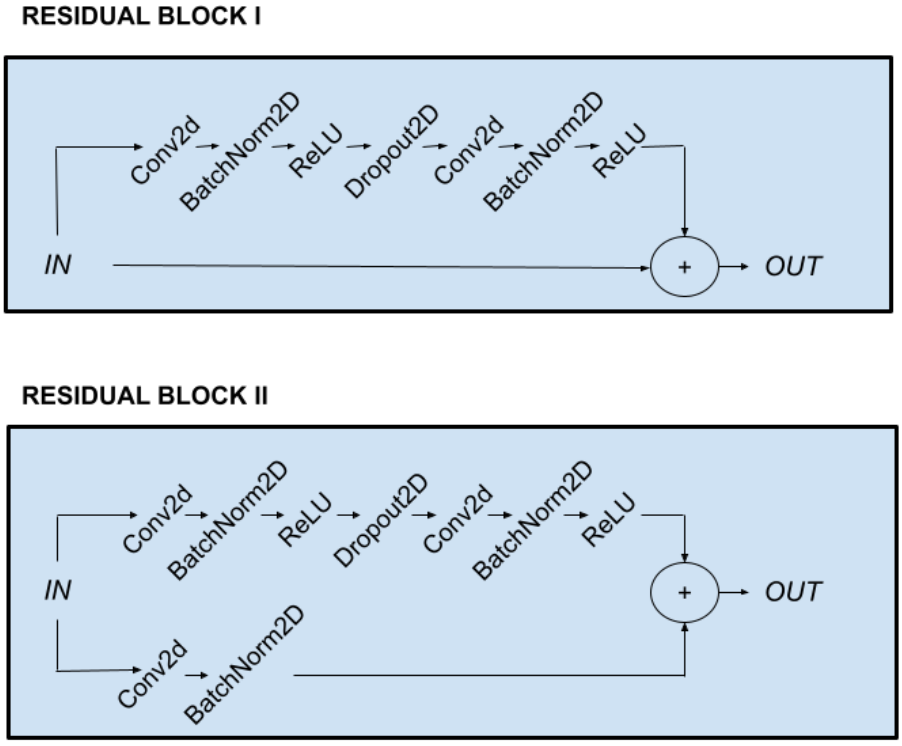

Figure A.7: Details of the residual network blocks from Figure 3.

Appendix G. Neural Network Architecture Details

We detail the neural network architecture. Figure 3 shows a schematic of the architecture, and Figure A.7 depicts specifically the residual network blocks.

The first convolutional layer (green block, Figure 3) has 17 output-channels, a kernel size of three, a stride of one, and one pixel of padding. All convolutional layers inside residual block 1, as well as the convolutional layers on the top row of residual block 2 (Figure A.7) also have the same parameters. Only the convolutional layers on the bottom row of residual block 2 are different: they still have output channels of dimension 17, but down-sample using a kernel size of one, and a stride of 2. Inside the residual blocks, the dropout layers have dropout probability of 0.11399.

The final fully connected block (red block, Figure 3) has latent dimension 185, and a dropout probability of 0.013123.

References

Timothy M.C. Abbott, Filipe B. Abdalla, Alex Alarcon, et al. Dark energy survey year 1 results: cosmological constraints from galaxy clustering and weak lensing. *Physical Review D*, 98(4), 2018.

Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework. In *International Conference on Knowledge Discovery and Data Mining*, page 2623–2631, 2019.

Luca Ambrogioni, Umut Güçlü, Julia Berezutskaya, Eva van den Borne, Yağmur Güçlütürk, Max Hinne, Eric Maris, and Marcel van Gerven. Forward amortized inference for likelihood-free variational marginalization. In *International Conference on Artificial Intelligence and Statistics*, 2019.

Deokkeun An, Jennifer A. Johnson, James L. Clem, et al. Galactic globular and open clusters in the Sloan Digital Sky Survey: Crowded-field photometry and cluster fiducial sequences in ugriz. *The Astrophysical Journal Supplement Series*, 179(2):326–354, 2008.

Bastien Arcelin, Cyrille Doux, Eric Aubourg, Cécile Roucelle, and LSST Dark Energy Science Collaboration. Deblending galaxies with variational autoencoders: a joint multi-band, multi-instrument approach. *Monthly Notices of the Royal Astronomical Society*, 500(1):531–547, 2021.

Atilim Gunes Baydin, Lei Shao, Wahid Bhimji, et al. Efficient probabilistic inference in the quest for physics beyond the standard model. In *Advances in Neural Information Processing Systems*, 2019.

Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer, New York, New York, 2006.

David M. Blei, Alp Kucukelbir, and Jon D. McAuliffe. Variational inference: a review for statisticians. *Journal of the American Statistical Association*, 112(518):859–877, 2017.

BLISS, 2023. Bayesian light source separator. `https://github.com/prob-ml/bliss`, 2023. [Accessed: 2023-1-27].

Jörg Bornschein and Yoshua Bengio. Reweighted wake-sleep. *arXiv preprint arXiv:1406.2751*, 2014.

James Bosch, Robert Armstrong, Steven Bickerton, et al. The hyper suprime-cam software pipeline. *Publications of the Astronomical Society of Japan*, 70(SP1):1–39, 2018.

Brendon J. Brewer, Daniel Foreman-Mackey, and David W. Hogg. Probabilistic catalogs for crowded stellar fields. *The Astronomical Journal*, 146(1):7–15, 2013.

Kyle Cranmer, Johann Brehmer, and Gilles Louppe. The frontier of simulation-based inference. *Proceedings of the National Academy of Sciences*, 117(48):30055–30062, 2020.

ESAHubble, 2021. Hubble's instruments: ACS – advanced camera for surveys. `https://esahubble.org/about/general/instruments/acs/`, 2021. [Accessed: 2021-02-21].

Richard M Feder, Stephen K. N. Portillo, Tansu Daylan, and Douglas Finkbeiner. Multi-band probabilistic cataloging: a joint fitting approach to point-source detection and deblending. *The Astronomical Journal*, 159(4):163–188, 2020.

Gregory M. Green, Edward Schlafly, Catherine Zucker, Joshua S. Speagle, and Douglas Finkbeiner. A 3D dust map based on Gaia, Pan-STARRS 1, and 2MASS. *The Astrophysical Journal*, 887(1):93–120, 2019.

Peter J. Green. Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, 82:711–732, 1995.

David Greenberg, Marcel Nonnenmacher, and Jakob Macke. Automatic posterior transformation for likelihood-free inference. In *International Conference on Machine Learning*, 2019.

Geoffrey E. Hinton, Peter Dayan, Brendan J. Frey, and Radford M. Neal. The wake-sleep algorithm for unsupervised neural networks. *Science*, 268(5214):1158–1161, 1995.

Michael I. Jordan, Zoubin Ghahramani, Tommi I. Jaakkola, and Lawrence K. Saul. An introduction to variational methods for graphical models. *Machine Learning*, 37:183–233, 1999.

Diederik P Kingma and Jimmy Ba. Adam: a method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Diederik P Kingma and Max Welling. Auto-encoding variational Bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Alp Kucukelbir, Dustin Tran, Rajesh Ranganath, Andrew Gelman, and David M. Blei. Automatic differentiation variational inference. *Journal of Machine Learning Research*, 18(14):1–45, 2017.

François Lanusse, Rachel Mandelbaum, Siamak Ravanbakhsh, Chun-Liang Li, Peter Freeman, and Barnabás Póczos. Deep generative models for galaxy image simulations. *Monthly Notices of the Royal Astronomical Society*, 504(4):5543–5555, 2021.

Tuan Anh Le, Adam R. Kosiorek, N. Siddharth, Yee Whye Teh, and Frank Wood. Revisiting reweighted wake-sleep for models with stochastic control flow. In *Uncertainty in Artificial Intelligence*, pages 1039–1049, 2020.

LSST, 2023. About LSST. `https://www.lsst.org/about/dm`, 2023. [Accessed: 2023-1-27].

LSST, 2023b. Key numbers. `https://lsst.org/scientists/keynumbers`, 2023. [Accessed: 2023-02-12].

Robert Lupton, James E. Gunn, Zeljko Ivezic, Gillian R. Knapp, Stephen Kent, and Naoki Yasuda. The SDSS imaging pipelines. *arXiv preprint astro-ph/0101420*, 2001.

George Papamakarios and Iain Murray. Fast ϵ-free inference of simulation models with Bayesian conditional density estimation. In *International Conference on Neural Information Processing Systems*, page 1036–1044, 2016.

Stephen K. N. Portillo, Benjamin C. G. Lee, Tansu Daylan, and Douglas P. Finkbeiner. Improved point-source detection in crowded fields using probabilistic cataloging. *The Astronomical Journal*, 154(4):132–156, 2017.

Rajesh Ranganath, Sean Gerrish, and David Blei. Black box variational inference. In *Artificial Intelligence and Statistics*, 2014.

Jeffrey Regier, Jon D. McAuliffe, and Prabhat. A deep generative model for astronomical images of galaxies. In *NIPS Workshop on Advances in Approximate Bayesian Inference*, 2015.

Jeffrey Regier, Andrew C. Miller, David Schlegel, Ryan P. Adams, Jon D. McAuliffe, and Prabhat. Approximate inference for constructing astronomical catalogs from images. *The Annals of Applied Statistics*, 13(3):1884–1926, 2019.

David M. Reiman and Brett E. Göhre. Deblending galaxy superpositions with branched generative adversarial networks. *Monthly Notices of the Royal Astronomical Society*, 485 (2):2617–2627, 2019.

Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *International Conference on Machine Learning*, 2014.

Olga Russakovsky, Jia Deng, Hao Su, et al. ImageNet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.

Javier Sanchez, Ismael Mendoza, David P. Kirkby, and Patricia R. Burchat. Effects of overlapping sources on cosmic shear estimation: statistical sensitivity and pixel-noise bias. *Journal of Cosmology and Astroparticle Physics*, 2021(07):043, 2021.

Ata Sarajedini, Luigi R. Bedin, Brian Chaboyer, et al. The ACS survey of galactic globular clusters. *The Astronomical Journal*, 133(4):1658–1672, 2007.

Edward F. Schlafly, Gregory M. Green, Dustin Lang, et al. The DECam plane survey: optical photometry of two billion objects in the southern galactic plane. *The Astrophysical Journal Supplement Series*, 234(2):39–58, 2018.

SDSS, 2020. Scope. `https://www.sdss.org/dr16/scope/`, 2020. [Accessed: 2020-12-23].

James C. Spall. *Introduction to Stochastic Search and Optimization*. John Wiley & Sons, Inc., New York, New York, 2003.

Peter B. Stetson. DAOPHOT: a computer program for crowded-field stellar photometry. *Astronomical Society of the Pacific*, 99:191–222, 1987.

Martin J. Wainwright and Michael I. Jordan. Graphical models, exponential families, and variational inference. *Foundations and Trends in Machine Learning*, 1(1–2):1–305, 2008.

Ronald J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8(3–4):229–256, 1992.

Edward L. Wright, Peter R. M. Eisenhardt, Amy K. Mainzer, et al. The wide-field infrared survey explorer (WISE): mission description and initial on-orbit performance. *The Astronomical Journal*, 140(6):1868–1881, 2010.

Bo Xin, Zeljko Ivezič, Robert H. Lupton, et al. A study of the point-spread function in SDSS images. *The Astronomical Journal*, 156(5):222–232, 2018.

Cheng Zhang, Judith Bütepage, Hedvig Kjellström, and Stephan Mandt. Advances in variational inference. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(8):2008–2026, 2019.