# Advancing the Search for dark matter with Deep Convolutional Neural Networks

**Ben Sorscher**
Department of Applied Physics
Stanford University
bsorscher@stanford.edu

**Sebastian Wagner-Carena**
Department of Physics
Stanford University
swagnerc@stanford.edu

## Abstract

Gravitational lenses provide a unique observational tool for probing the structure of dark matter in the universe. By studying the features of galactic images that have been distorted by gravitational lenses, physicists can perform calculations to infer the underlying composition of matter that generated this distortion. These calculations are involved and costly. Here, we use deep CNNs to perform the same inference. We achieve highly accurate predictions $\sim$1 million times faster than standard physics computations, and we generate a full covariance matrix of uncertainties on our predictions. This presents a valuable rapid inference technique for dark matter research on next-generation sky surveys.

## 1 Introduction

One of the profound predictions of Einstein's general theory of relativity is the bending of light's path by sufficiently massive objects. When the mass of the object is on the scale of a galaxy, light from a distant source can be bent to produce what is known as an Einstein ring (see Figure 1). This effect - known as strong gravitational lensing - has become a powerful tool in modern observational cosmology. Images of gravitational lenses enable us to infer features about the matter distribution that generated them, even if we cannot see the matter itself. This makes lensing measurements uniquely suited to probing the structure of dark matter, a form of matter that is believed to interact only weakly with photons but compose around 85% of the matter in the universe [15]. Better understanding the distribution of dark matter will help set constraint on the universe's early evolution, allow us to better understand dark matter's role in the formation of galaxies, and help us probe particle physics beyond the standard model [1].

Unfortunately, the calculations required to infer these parameters are computationally costly [12]. Before next-generation sky surveys such as LSST, Euclid, and WFIRST come online in the next 2-3 years, we need efficient, accurate tools to predict the parameters of strong gravitational lenses. Additionally, for these prediction tools to be scientifically viable, we must be able to present meaningful constraints on their uncertainties. In this paper, we use deep Bayesian convolutional neural networks to achieve state-of-the-art predictions on the salient features of gravitational lenses, along with a full covariance matrix of uncertainties. The model is trained on strong gravitational lensing images and outputs a series of six parameters that describe the Singular Isothermal Ellipsoid density profile of the underlying mass distribution. [4].

## 2 Related Work

### 2.1 Graviational Lensing Models

There is a large collection of modeling software currently used for gravitational lensing [12]. The majority requires some level of expertise in the subject to operate, and can take on the order of

Preprint. Work in progress.

days or weeks to generate accurate predictions for a lens. The techniques often use either a Markov Chain Monte Carlo or a downhill optimization strategy to fit the parameters to the lensing image. More recent work has shown that convolutional neural networks can accurately identify images containing gravitational lenses [11], and can even infer the underlying parameters of the these networks [7]. Efforts have also been made to obtain uncertainties on these predictions [13], although they are limited to the diagonal entries of the covariance matrix. Work by Hezaveh et al. has found similar performance across various model architectures including Inception-v4, Alexnet, and Overfeat [7]. Their experiments have shown that convolutional networks can achieve prediction accuracy competitive with other sophisticated modeling techniques while taking fractions of a second for a prediction.

### 2.2 Bayesian Deep Learning Models

In the past two years, Bayesian deep learning models have been proposed as a mean of modeling the full uncertainty in computer vision tasks [9]. These are heteroscdastic models that seek to capture the two main sources of modeling uncertainty: aleatoric uncertainty - the uncertainty stemming from inherent noise in the observation - and epistemic uncertainty - the uncertainty arising from our choice of the model and its parameters. Extending these models to capture the full covariance matrix of the output parameters requires finding an efficient mapping between the outputs of the network and the space of positive definite matrices. This issue has been explored beyond the framework of deep learning, and a number of useful parametrizations have been developed [8]. Additionally, some recent work in convolutional neural networks has also explored how to efficiently learn the covariance of pixels in an image [3].

## 3 Dataset and Features

We have gathered a training set of 100,000 high-quality, labeled images of gravitational lenses (Figure 1). These images are generated using background galaxy images from GalaxyZoo [2] and GREAT3 [14], along with simulations of gravitational lensing events developed at Stanford/SLAC [7]. In addition, we have a held-out test set of 15,000 images.

In order to produce realistic images that more accurately represent those captured by a telescope, and to generate a larger set of images for training, we implement a data augmentation pipeline involving the following augmentations: (1) adding Gaussian noise to the images associated with our detector, (2) adding masked pixels to the images, (3) applying a point spread function associated to the detector, (4) adding Poisson shot noise associated with the intrinsic statistics of measuring a finite number of photons, (5) translating the center of the lens. The difference between the original images and the modified images can be seen in Figure 1. Finally, we whitened the outputs such that they had mean zero and variance one on the training set. Before reporting our results we transformed the outputs of our model back into their original scale.

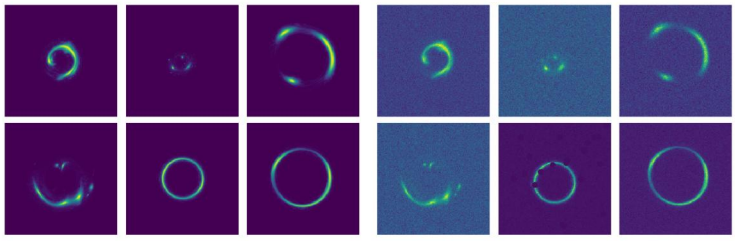

Figure 1: A picture of the strong lensing dataset before (left) and after (right) the data augmentation process.

## 4 Methods

The model architecture used in this work is a modified version of Alexnet [10] that includes both batch normalization and dropout after each convolutional layer (see Figure 2). Building off of the

Bayesian modeling techniques developed by Kendall et al. [9], we design a heteroscedastic model built to capture the full covariance of both the aleatoric and epistemic uncertanties. The aleatoric uncertanty is independent of our choice of model, so we treat it as a prediction of our network and incorporate it into our loss function as follows:

$$\mathcal{L}(y_n, \hat{y}_n, \Sigma_n) = \frac{1}{2}(y_n - \hat{y}_n)^T \Sigma_n^{-1}(y_n - \hat{y}_n) + \frac{1}{2}\log|\Sigma_n| \tag{1}$$

where $y_n$ is the true value of the $n^{\text{th}}$ example, $\hat{y}_n$ is the predicted value, $\Sigma_n$ is the predicted covariance matrix, and $|\Sigma_n|$ denotes the determinant of the covariance matrix. We are modeling six parameters of the Singular Isothermal Ellipsoid density profile; therefore our model outputs the six values of $\hat{y}_n$ and the twenty-one values required to describe the degrees of freedom of $\Sigma_n$ for each image. In order to map from the twenty-one covariance matrix outputs of our model to the space of positive definite matrices we use a log-Cholesky decomposition. In this framework, our twenty-one parameters map to the twenty-one nonzero values of the six by six lower traingular matrix $L_n$. Six of the twenty-one parameters are considered to be log predictions of the diagonal values of $L_n$, therefore forcing $L_n$ to have positive diagonal values. The remaining fifteen parameters give the off-diagonal values. Given $L$ we construct the matrix $\Pi_n = L_n L_n^T$. It can be shown that $\Pi$ is guaranteed to be positive definite and the mapping between $L_n$ and $\Pi_n$ is unique [8]. If a matrix is positive definite then not only is it invertible, but its inverse is also positive definite. This allows us to take $\Pi_n = \Sigma_n^{-1}$ and define a more numerically stable loss function:

$$\mathcal{L}(y_n, \hat{y}_n, L_n) = \frac{1}{2}(y_n - \hat{y}_n)^T \Pi_n(y_n - \hat{y}_n) - \frac{1}{2}\log|\Pi_n| \tag{2}$$

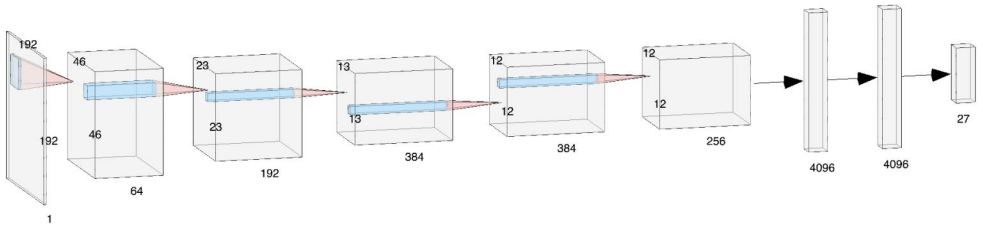


Figure 2: A graphic describing our modified implementation of Alexnet [10]. Dropout occurs before each convolutional layer, and batch normalization is conducted after each convolutional layer.

We are also interested in capturing the epistemic uncertainties - the uncertainty associated with errors from our choice of network and its parameters. To do so we will make use of previous work in the development of Bayesian Neural Networks [5] and take an expectation maximization approach to marginalizing over the weights. If we choose a variational distribution $q(w)$ for our weights such that:

$$q(w_i) = w_i \cdot \text{Bernoulli}(p) \tag{3}$$

we can then sample from this marginalized distribution by using Monte Carlo integration (that is, setting weights to 0 in accordance with the Bernoulli distribution). The advantage of this choice of varirational distribution is that it exactly mimics dropout, and therefore training our network to estimate epistemic uncertainties is equivalent to simply training our network with dropout. At test time we can then estimate the epistemic uncertainties by calculating the prediction on a single input multiple times and conducting dropout each time. This is equivalent to sampling from the marginalized distribution of our test example. In this framework, the dropout rate must be tuned so that the prediction of the epistemic uncertainty matches what is seen in the data.

## 5 Results

We trained the models with an initial learning rate of $10^{-5}$, manually adjusting the learning rate as the training progressed. We found that a mini-batch size of 64 allowed the network to learn quickly

while allowing for relatively a smooth loss function. We trained using the Adam optimizer with the standard choice $\beta_1 = 0.9$, $\beta_2 = 0.999$. The momentum for our batch norm was also the standard value of $0.99$. In order to understand the effect of the dropout rate we tested three separate rates: 0.02, 0.03, and 0.1. The best results were achieved with a dropout rate of 0.03 where after training for 40 epochs, we obtained the 68% error interval shown in table 1. We chose this error metric to be able to compare our results to work by Hezaveh et al. [7]. Training for longer should allow us to achieve even lower test error.

Table 1: 68% errors on Singular Ellipsoid Density profile parameters

| | $\theta_R$ | $\epsilon_x$ | $\epsilon_y$ | $x$ | $y$ | $\mu_F$ |
|---|---|---|---|---|---|---|
| Hezaveh et al. [7] | 0.03 | 0.04 | 0.05 | 0.06 | 0.06 | - |
| **Ours** | 0.07 | 0.11 | 0.12 | 0.26 | 0.29 | 0.37 |

Comparing the error intervals show that our model has similar accuracy in its predictions both to previous deep convolutional strong lensing models and downhill optimization models. In terms of performance, our model takes 2 seconds to make predictions on 1000 images. This represents a $\sim 1,000,000\times$ speed-up over standard computations [12]. No significant overfitting to the training set seems to have occurred: the performance on the training set is identical to what is shown in Table 1.

We now consider metrics to quantify the validity of our predicted covariance matrix. Figure 3 plots the confidence intervals on each parameter estimate (given by the diagonal elements of our covariance matrix), for two different dropout rates. Ideally, exactly 68% of the parameters would be within one predicted standard deviation (where $\sigma = \sqrt{\text{var}}$) of the true value, 95% would be within two, and 99.7% would be within 3. We find that the confidence intervals match the expected distribution more closely for a dropout rate of 0.03. At this dropout rate, an average of 61.9%, 85.8% and 94.9% of the data are within 1, 2, and 3 standard deviations respectively. Though these averages are close to matching the desired distribution, it is worth noting that the distribution for each individual parameter deviates from the mean. There is also evidence that more training is required for the the 0.1 dropout model. We would expect a larger dropout rate to increase the epistemic uncertainty, causing more of the predictions to fall within one standard deviation of the true value. The opposite is true here, mainly because the larger uncertainties are washed out by poorer predictions.

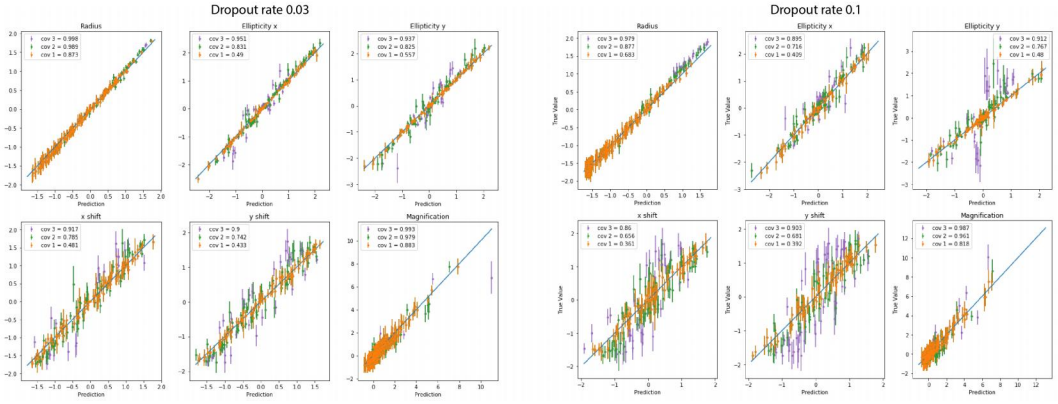


Figure 3: Uncertainties on the six parameter estimates (equivalently, diagonal elements of the covariance matrix). Predicted parameter values are on the x-axis, true parameter values on the y-axis, with error bars given by the network's predicted uncertainties. Orange represents points within one predicted standard deviation of the true value, green within two, and purple within three.

We now turn to uncertainties on the full covariance matrix. Assuming that the set of parameters $\mathbf{x}$ is drawn i.i.d. for each lens from a multivariate Gaussian distribution specified by our covariance matrix $\Sigma_n$, then the quantity

$$(\mathbf{x} - \mu)^T \Sigma^{-1} (\mathbf{x} - \mu) \sim \chi^2_p \tag{4}$$

is distributed as a Chi-squared statistic in $p = 6$ dimensions. Therefore, we can plot the predicted values of equation 4 for a set of 1000 images from our test set and see how closely they match the desired statistic. Figure 5 shows the results of this test for a dropout rate of 0.03 and a dropout rate of 0.1. Both histograms capture the correct shape of a chi-squared distribution, but both models appear to be overconfident in their predictions. As before, the dropout rate of 0.1 should have larger uncertainties that produce better results on the confidence intervals, but the models inaccuracy in predictions outweighs that benefit. It's likely that training a model with a larger dropout rate more extensively would yield a closer fit to the desired chi-squared statistic.
Finally, we can also visually confirm that our models' predictions of the uncertainty match what we expect qualitatively. Figure 4 shows examples of images the model thinks are difficult to predict, and those it thinks are easy –quantified by the determinant of the covariance matrix. Qualitatively, the outputs are what we expect. The images for which the model is uncertain are those where there are few, small copies of the source to constrain the matter field. The images for which the model is confident are those where the lens traces out a well defined ring or there are several copies of the source that are spread out.

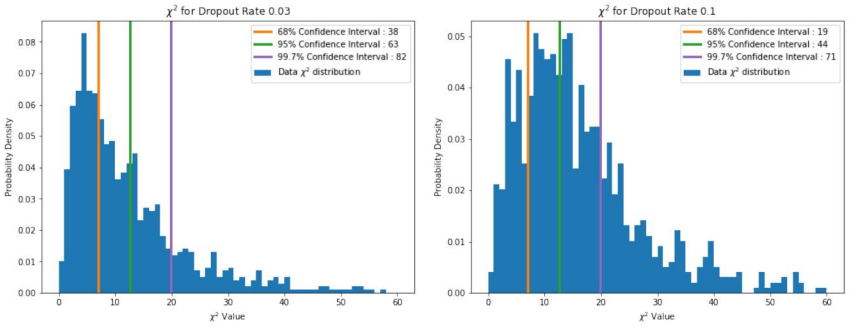


Figure 4: The chi-squared distribution for a dropout rate of 0.03 (left) and a dropout rate of 0.1 (right). Both models are biased towards larger chi-squared values compared to the expected distribution but fit the expected shape.

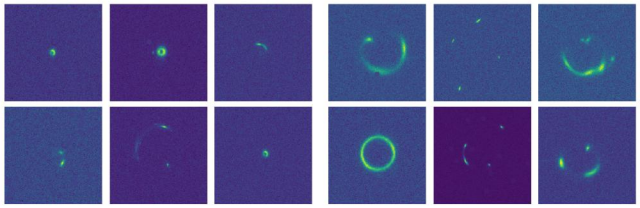

Figure 5: Images on which the model is uncertain (left) and certain (right). The characteristics of the lensing images match what would be intuitively labeled as a difficult or easy image to model.

## 6 Conclusion

We have presented a technique for rapidly estimating the parameters specifying the underyling distribution of matter in a gravitational lens, along with a full covariance matrix of associated uncertainties. Our parameter estimates are very accurate, and our predicted uncertainties approximately capture both the epistemic uncertainty in our choice of model, and the aleatoric uncertainty inherent in our data. Tests on the full covariance matrix show that it is capturing the desired distribution, but that further tuning of the dropout rate and training of the models may be required. It may also be interesting to explore frameworks where the dropout rate is more flexible. In particular, we intend to implement Concrete Dropout [6], which will allow us to treat the dropout rate as a learnable parameter.

## 7 Contributions

Sebastian worked on fleshing out the data augmentation pipeline and building the loss function for the models. Ben worked on building the model architecture and training each individual model. Both partners contributed equally to testing code and to the production of this paper.

## 8 Code availability

All code is available at the following github repository `https://github.com/bsorsch/cov_grav_lenses`.

We used some code from this repo as a starting point: `https://github.com/yasharhezaveh/Ensai`

## References


[1] Gianfranco Bertone, Dan Hooper, and Joseph Silk. *Particle Dark Matter: Evidence, Candidates and Constraints*. Tech. rep. 2004. URL: https://arxiv.org/pdf/hep-ph/0404175.pdf.

[2] Hugh Dickinson et al. "Galaxy Zoo: Morphological Classification of Galaxy Images from the Illustris Simulation". In: *The Astrophysical Journal* 853.2 (Feb. 2018), p. 194. ISSN: 1538-4357. DOI: 10.3847/1538-4357/aaa250. URL: http://stacks.iop.org/0004-637X/853/i=2/a=194?key=crossref.a199c1a481a4f242455d61e41c7207cd.

[3] Garoe Dorta et al. "Structured Uncertainty Prediction Networks". In: (Feb. 2018). URL: http://arxiv.org/abs/1802.07079.

[4] R. European Southern Observatory., P. Schneider, and M. Bartelmann. *Astronomy and astrophysics*. Vol. 284. EDP Sciences [etc.], 1969, pp. 285–299. URL: http://adsabs.harvard.edu/abs/1994A%26A...284..285K.

[5] Yarin Gal and Zoubin Ghahramani. "Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning". In: 48 (2015). ISSN: 10414347. DOI: 10.1109/TKDE.2015.2507132. URL: http://arxiv.org/abs/1506.02142.

[6] Yarin Gal and Alex Kendall. *Concrete Dropout*. Tech. rep. URL: https://arxiv.org/pdf/1705.07832.pdf.

[7] Yashar D. Hezaveh, Laurence Perreault Levasseur, and Philip J. Marshall. "Fast automated analysis of strong gravitational lenses with convolutional neural networks". In: *Nature* 548.7669 (2017), pp. 555–557. ISSN: 14764687. DOI: 10.1038/nature23463. URL: http://dx.doi.org/10.1038/nature23463.

[8] José Jos, José C Pinheiro, and Douglas M Bates. *Unconstrained Parameterizations for Variance-Covariance Matrices*. Tech. rep. URL: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.31.494&rep=rep1&type=pdf.

[9] Alex Kendall and Yarin Gal. "What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision?" In: (Mar. 2017). URL: http://arxiv.org/abs/1703.04977.

[10] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. *ImageNet Classification with Deep Convolutional Neural Networks*. Tech. rep. URL: http://code.google.com/p/cuda-convnet/.

[11] François Lanusse et al. "CMU DeepLens: Deep learning for automatic image-based galaxy-galaxy strong lens finding". In: *Monthly Notices of the Royal Astronomical Society* 473.3 (2018), pp. 3895–3906. ISSN: 13652966. DOI: 10.1093/mnras/stx1665.

[12] Alan T. Lefor, Toshifumi Futamase, and Mohammad Akhlaghi. "A systematic review of strong gravitational lens modeling software". In: *New Astronomy Reviews* 57.1-2 (July 2013), pp. 1–13. ISSN: 1387-6473. DOI: 10.1016/J.NEWAR.2013.05.001. URL: https://www.sciencedirect.com/science/article/pii/S138764731300002X.

[13] Laurence Perreault Levasseur, Yashar D. Hezaveh, and Risa H. Wechsler. "Uncertainties in Parameters Estimated with Neural Networks: Application to Strong Gravitational Lensing". In: *The Astrophysical Journal Letters* 850.1 (2017), p. L7. ISSN: 20418213. DOI: 10.3847/2041-8213/aa9704. URL: http://arxiv.org/abs/1708.08843%0Ahttp://dx.doi.org/10.3847/2041-8213/aa9704.

[14] Rachel Mandelbaum et al. “THE THIRD GRAVITATIONAL LENSING ACCURACY TESTING (GREAT3) CHALLENGE HANDBOOK”. In: *The Astrophysical Journal Supplement Series* 212.1 (Apr. 2014), p. 5. ISSN: 0067-0049. DOI: 10.1088/0067-0049/212/1/5. URL: http://stacks.iop.org/0067-0049/212/i=1/a=5?key=crossref.10e5da7465374e0b76a1096531ec9b42.

[15] K.A. Olive and Particle Data Group. “Review of Particle Physics”. In: *Chinese Physics C* 40.10 (Oct. 2016), p. 100001. ISSN: 1674-1137. DOI: 10.1088/1674-1137/40/10/100001. URL: http://stacks.iop.org/1674-1137/40/i=10/a=100001?key=crossref.5ca5f9c166adc59f41e03a802b715c19.