Atmospheric Point-spread Function Estimation of Galactic Survey Images

Riley Carlson
Stanford University
rileydc@stanford.edu

Ishaan Singh
Stanford University
ishaanks@stanford.edu

Tirth Surti
Stanford University
tsurti@stanford.edu

Abstract

Ground-based observatories conduct large scale surveys of distant galaxies to map the night sky, producing millions of images of galaxies. Significant amounts of information can be learned about galaxy evolution and structure just from the photometric morphology of the galaxy. However, images produced by ground-based observatories suffer from blurring due to the turbulence of the atmosphere, which can only be corrected by deconvolution of the image with an atmospheric point-spread function (PSF). Using blurred galaxy images generated by applying randomly sampled PSFs to Hubble Space Telescope data, we experiment with end-to-end convolutional neural networks and vision transformers to reproduce the applied PSF's parameters and 2D shape. A mean-squared error and residual analysis shows strong recovery of the PSF across the explored methods.

1. Introduction

Ground-based galactic surveys conducted by programs like the Sloan Digital Sky Survey and Vera C. Rubin Observatory will have increasing capabilities to push observation of galaxies out to large redshifts ($z < 6$), corresponding to times when the universe was just several billions of years old [11, 16]. This large volume will enable these observatories to collect images of millions of galaxies to better understand their morphologies, the distribution of luminous matter throughout the universe, and the evolution of the universe itself through the precise measurement of distances.

Studying the morphology of galaxies, such as their shapes, surface brightness profiles, and unique structures, can give insight into a large number of galactic astrophysical phenomena. In particular, galaxies are often classified into one of three categories, based upon the classic Hubble tuning fork diagram: ellipticals, spirals, and irregulars. These individual galaxies each have their own proposed formation histories and characteristics. In particular, spiral galaxies contain significant gas material with bright young stars, elliptical galaxies are smooth, featureless ellipsoids containing old stars that are likely to have formed from collisions of spiral galaxies, and irregular galaxies may be somewhere in between the other two types on the galaxy evolutionary stage [3]. The overall surface brightness profile of a galaxy is also crucial for constraining the mass distribution and dynamics of these galaxies through its deprojection to a luminosity density and hence mass density [2]. Finally, sub-galactic features such as spiral arms can indicate places of ongoing star formation, extended jets can reveal the presence of an active central supermassive black hole, and tidal tails are markers of past galactic mergers.

Unfortunately with ground-based observatories, these important morphological features can become hard to distinguish and surface brightness profiles can become strongly distorted due to the effects of atmospheric turbulence. In Figure 1, we show the effect of convolving a Hubble Space Telescope image with a PSF, revealing the significant loss of morphological features. Atmospheric turbulence is typically modeled by a point-spread function (PSF), which redistributes the light of the astronomical image, modeled by the convolution of the raw image $I(x, y)$ with a kernel $K(x, y)$:

$$I'(x, y) = I(x, y) * K(x, y), \tag{1}$$

where $I'(x, y)$ is the degraded image. The kernel acts in a flux-preserving matter, ensuring that the total photon count of the original source is unchanged as long as the spreading effect is smaller than the instrument detector's field of view. This means that

$$\int K(x, y)\, dxdy = 1. \tag{2}$$

It is therefore important to be able to reconstruct the PSF accurately in order to deconvolve the degraded image and obtain science-ready data for analysis.

The PSF can usually be modeled using point sources in the astronomical image. Stars, for example, are intrinsically point source Dirac delta functions $\delta(x, y)$, that have the fortunate property where $\delta(x, y) * K(x, y) = K(x, y)$, so stars in the degraded image itself can be used to approximate the PSF directly. However, this approach is typically hindered by the absence of unsaturated resolved stars that are well

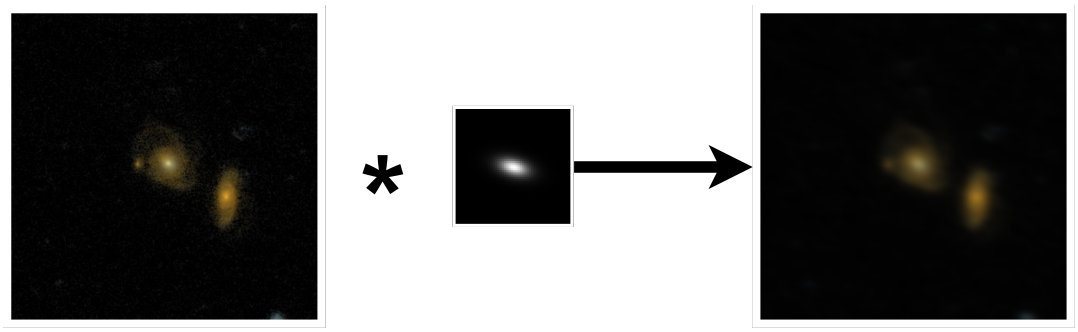

Figure 1. An unblurred multi-channel image of two galaxies (left) being convolved with a PSF (middle) yielding a blurred image (right). We see that important morphological features such as individual spiral arms are mostly lost.

distributed around the extended source. Required exposure times to image distant galaxies during surveys may generally be too long to keep nearby stars unsaturated and free from artifacts but too short to get high signal for distant stars [19], which are possibly blended into the light of other galaxies, crowded with other stars, or not even resolvable.

Another approach is to use blind deconvolution, using prior constraints to derive a maximum a posteriori method (MAP) to simultaneously estimate the PSF K and the raw image I. However, this parameter space is often still too large to achieve a suitable solution. Finally, direct modeling based upon the degraded image and atmospheric processes often require tedious parameter tuning [14]. We therefore propose to use deep neural networks to accurately infer the PSF directly from a blurred galaxy image.

2. Related Work

Existing approaches to estimate the PSFs of astronomical images focus on a much simpler class of objects. One of the earliest deep-learning approaches to address the issues of forward modeling PSFs was conducted by [9]. The authors use a basic convolutional neural network to predict the parameters of a predefined PSF using images of simulated stars. The PSF profiles are based upon the Moffat function [15], introducing additional anisotropies to the radially symmetric profile by shifting the centroid position and adding ellipticity, flexion, and kurtosis. Since the PSFs themselves represent the blurred point-source stars, the authors simulate stars by randomly sampling from the parameters of the PSF profile. Qualitatively, their neural network achieved strong recovery of the PSF parameters from their images.

Using relatively high signal-to-noise images containing only one extended object, [14] aim to estimate adaptive optics (AO) PSFs from ground-based telescopes. These PSFs arise from AO systems used to correct image degradation caused by atmospheric turbulence. The authors generate training data by randomly sampling PSF parameters from a predefined AO PSF model and convolving resolved images of satellites and planets with the simulated PSFs. The PSF model used is radially symmetric without anisotropy. The authors argue this restricted model can significantly hamper their model's generalization to real PSF-convolved images. PSF parameters are estimated from the blurred images using a deep ResNet34-like CNN architecture, modified to output a 2D PSF model and optimized over the root mean square error (RMSE) between the truth and generated PSF. The authors find that this model performs well on the simulated blurred images.

PSF parameters can also be inferred by the instrumental configuration of telescopes. The authors of [10] developed a new method to estimate the PSF for wide field small aperture telescopes (WFSATs) using a deep convolutional neural network called Tel-Net, using as input information about the optical configuration of a telescope and translating this optical configuration into a PSF. The Tel-Net architecture consists of a series of convolutional layers to encode the PSF information followed by "fixup" layers which are deep neural network blocks that can serve as an important form of regularization and speed up convergence during training.

Applying deep learning to the task of inferring PSFs from blurred galaxy images is a novel and challenging task compared to prior works. Unlike [9], we aim to estimate PSF parameters from a galaxy image convolved with a generated PSF. This is more challenging because stars are essentially PSFs themselves, as a PSF convolved with a star is the PSF itself. In our work, we do not input the generated PSF directly, making it more difficult to predict the PSF parameters from the blurred galaxy image. We are predicting the PSF with an additional layer of abstraction, requiring our model to learn more complex patterns, as illustrated in Figure 1.

In addition to the difference in the input paradigm, the distant galaxies we use have fewer resolved features in comparison to [14], making it more challenging to infer the effects of a PSF. The decrease in resolved features means that our models may struggle to capture the complexity of the data distribution. This aspect of our project adds to the

difficulty of accurately modeling and estimating PSFs from blurred galaxy images at larger distances. Similarly, the nature of the input data introduces more confounding factors, as original images may contain multiple galaxies. Our models must learn to find consistent blurring features across all the galaxies in the image and not be influenced by their different morphologies, further complicating the learning process.

Finally, unlike [10], we aim to remain agnostic to the configuration of the original instrument by implementing a full end-to-end model that can infer the PSF just from the blurred galaxy image.

3. Dataset Generation

No datasets of blurred galaxy images with their PSFs exist, so we generate our dataset using above-atmosphere galaxy images and convolving them with a randomly sampled PSF. We utilize the Galaxy Zoo Hubble Space Telescope (HST) dataset [23], which contains human-made classifications of 113,705 galaxies out to redshift $z \sim 4$ from the AEGIS, COSMOS, GEMS, and GOODS surveys. These galaxies were hand-classified by human participants of the Galaxy Zoo project [13], indicating whether the galaxy images they looked at have specific features such as spiral arms, bars, bulges, and more. Among this dataset of galaxies, we select those that have a spiral structure indicating a spiral galaxy, subject to the threshold that at least 75% of respondents could identify the structure. Among the spiral galaxies, we also select those at redshift $z < 1.5$, corresponding to the earliest time of when the universe was just around 4 billion years old. This threshold is chosen so that the galaxies are distant but still close enough to have resolvable structures. We stick with spiral galaxies because their sub-galactic spiral structure is more readily observed and easily lost under atmospheric blurring, whereas elliptical galaxies are largely smooth, featureless ellipsoids. We select 5000 galaxy images with each image $G^{(i)} \in \mathbb{Z}^{424\times424\times3}$.

For the PSF model, we use the Moffat function, which is good for modeling atmospheric blurring [15]. The most general Moffat function is given by:

$$K(r) = C\left(1 + \left(\frac{r}{\alpha}\right)^2\right)^{-\beta}, \tag{3}$$

where $r = \sqrt{x^2 + y^2}$ is the radial point over which we are evaluating the function, α controls the size of the blurring, β controls the steepness of the PSF profile, and C is a normalization constant so that the kernel integrates to 1. In order to make the PSF more realistic, we add anisotropy to the blurring; that is, we make the blurring axisymmetric rather than circularly symmetric. Concretely, we generate a 64×64 grid of coordinates (x, y) to evaluate the function over. We then rotate the coordinate system about an angle θ, to generate coordinates (x', y'):

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}, \tag{4}$$

We also introduce an ellipticity term ϵ, which dictates how much the PSF function be narrowed in the rotated y' direction. We re-evaluate the Moffat function over the rotated and scaled radial coordinate:

$$r' = \sqrt{x'^2 + \frac{y'^2}{(1-\epsilon)^2}}. \tag{5}$$

This gives us a PSF function that is narrowed along the y' direction, allowing for extra anisotropy within the parameterization.

To generate the training dataset, we downsample each image to a size 192×192, normalize by the max RGB value 255, and zero-pad by half the kernel size in each dimension so that the new image is $G' \in \mathbb{R}^{256\times256\times3}$. We randomly sample 5 Moffat parameters for each image such that $\alpha \sim \text{Uniform}(2, 15)$, $\beta \sim \text{Uniform}(2, 5)$, $\epsilon \sim \text{Uniform}(0, 0.8)$, and $\theta \sim \text{Uniform}(0, \pi)$. α and β are chosen so that the PSF falls to zero and hence fits within the given 64×64 space. We note that we do not extend up to $\epsilon = 1$ to avoid potential numerical errors in the PSF generation and to have sufficient width in both the principal directions of the PSF. The remaining axisymmetry (defined by the semi-major axis of the elliptical shape) of the PSF enables us to keep $\theta \in [0, \pi]$ rather than the full angular range. We generate the 64×64 anisotropic Moffat profile for each parameter sample $K(x, y)$ and convolve with each channel c of the padded image using a fast Fourier transform:

$$G_c^{(i)\prime\prime}(x, y) = K(x, y) * G_c^{(i)\prime}(x, y), \tag{6}$$

where $G^{(i)\prime\prime} \in \mathbb{R}^{256\times256\times3}$. This leaves us with a dataset X of the blurred images, where $|X| = 25000$, and the Moffat parameters y consisting of $\{\alpha, \beta, \epsilon, \theta\}$ that are used for each image. We perform a 80-10-10 split on the dataset to generate a training, validation, and test set, yielding 20,000 images for training. In order to improve the generalization of our models, we ensure that unblurred-galaxy images used to generate the training data are not found in the test or validation sets. This is so that our models can learn salient features that are relevant to the blurring of the galaxy rather than the galaxies themselves.

4. Methods

We experiment with two different classes of models: those that infer the parameters of the Moffat profile PSF and those that infer the 64×64 PSF image directly, given the blurred galaxy image as input. All models apart from those that are finetuned are implemented and trained from scratch using TensorFlow/Keras [1].

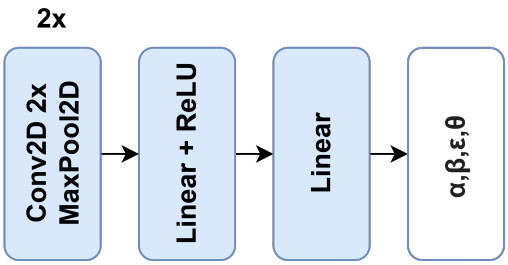

Figure 2. Baseline Model Architecture

4.1. Parameter Inference Models

4.1.1 Baseline

The baseline model consists of a shallow convolutional neural network with four convolutional layers having filter counts of {32, 32, 64, 64} and sizes 3×3. We keep the filter sizes small in comparison to the overall image so as to capture local features, where the effects of the blurring is best observed visually. To maintain spatial extent of the features, we perform "same" zero-padding with a stride of 1 at each layer and only downsample using maximum pooling after each pair of convolutions. Since we are regressing the PSF parameters, we flatten the output dimension and pass through additional linear layers. The model architecture is summarized in Figure 2.

4.1.2 Vision Transformer

Vision Transformer (ViT) has been shown to outperform deep convolutional neural networks on image classification tasks [6], so we consider using it to extract better features within the blurred galaxy images in order to predict PSF parameters. Traditionally, the ViT model takes a square input image $G^{(i)} \in \mathbb{R}^{n \times n \times 3}$ and splits it up into m non-overlapping patches of size $m \times 3p^2$, where p^2 is the number of pixels in the patch and $m = n^2/p^2$, generating a sequence of patches: $\{G_p^{(i)}\}_{p=1}^{m}$. However, in order to enhance the feature extraction of the vision transformer, we generate the input sequence using a convolutional layer rather than the input image itself. This is achieved by using h filters that have size $p \times p$ with stride that is the size of the filter, which yields a feature input sequence $F^{(i)} \in \mathbb{R}^{m \times h}$ after combining the spatial dimensions. This allows the transformer layer to use features generated by learned parameters for the input sequence, which will be specifically optimized for PSF estimation. Using images directly, on the other hand, may bias the transformer to attend less to blurring-specific features and thus make the transformer model more difficult to train. For our dataset, we proceed with $p = 16$, giving us sequences of length $m = 256$ for each blurred galaxy image given the input image shape.

ViT uses multi-headed attention to generate an encoded output sequence of features, but the attention operations make no regards to the relative positions of the patches within the original feature map. Encoding the relative position of the patches is crucial because the effect of the blurring is larger than the patch size as a result of the convolution operation. We therefore add positional embeddings to the initial feature sequence $F^{(i)}$. In particular, we generate positional embedding indices j over the length of the feature sequence m and we map each index to an embedding vector $E_j \in \mathbb{R}^h$ using a learnable weight, whose dimension matches the number of filters used in the initial convolution. We then add these embeddings to the feature sequence to obtain the input sequence for the subsequent transformer layers:

$$F_j'^{(i)} = F_j^{(i)} + E_j. \tag{7}$$

We perform multi-headed attention using 4 transformer heads and 8 layers. Each layer follows the standard encoder architecture first defined in [22]. The multilayer perceptron component consists of two linear layers with GeLU activation, commonly used in transformer encoders [5, 8]. We also incorporate dropout with probability $p = 0.1$ for regularization. We flatten the output sequence of the final encoder layer and project down to the desired output size of four for parameter estimation. We expect that the output sequence yields improved features compared to the baseline CNN over the original image for parameter estimation. The model architecture is summarized in Figure 3, showing the process of image patching, the incorporation of positional embeddings, and the transformer encoder. The transformer layer code is based upon and further modified from the Keras tutorial code [17].

4.1.3 Finetuning InceptionV3 and EfficientNet

Since the introduction of the VGG and ResNet models [18, 7], significant improvements have been made in training deep convolutional neural networks for image classification while reducing the number of required floating point operations and parameters. We seek to adapt such large models for PSF parameter regression, using the features that these networks are able to extract. Given our input image sizes are quite large, it is imperative that we choose a robust model that also has few parameters to enable for quicker training and meet the system's constraints. We therefore proceed with InceptionV3 and EfficientNetB0 [20, 21]. InceptionV3 contains $\sim$ 24 million parameters while EfficientNetB0 contains just $\sim$ 5.3 million parameters, which are less than the standard VGG and ResNet50 models. However, they still outperform on ImageNet [4] classification as a result of the larger depth. The Inception model consists of branches that simultaneously perform convolutions whose outputs are then concatenated; it also factorizes larger convolutions into smaller symmetric

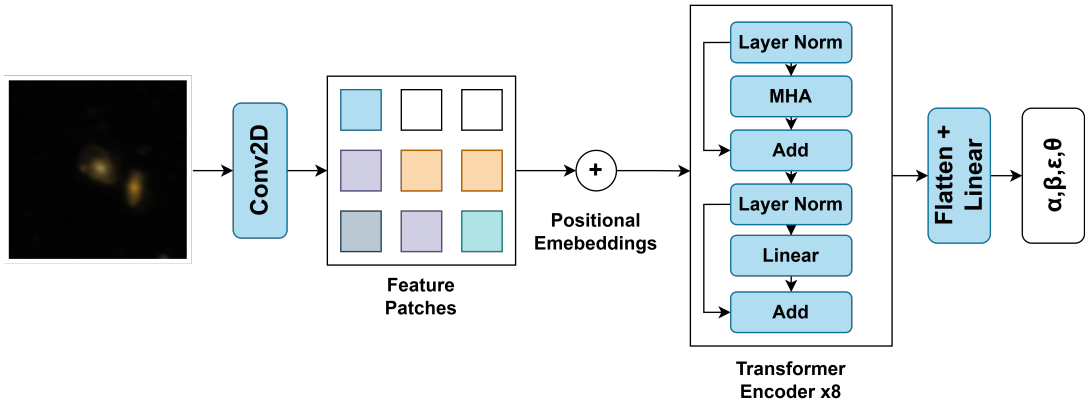

Figure 3. ViT Model Architecture, showing the procedure to turn an input image into patches using a convolutional layer, adding positional embeddings, and extracting features using multi-headed attention (MHA) transformer layers.

and asymmetric convolutions in order to reduce the parameter count. On the other hand, the EfficientNet model scales network depth, width, and resolution uniformly to achieve superior performance.

For further preprocessing with InceptionV3, we must rescale images pixels from $[0, 1]$ to $[-1, 1]$. EfficientNetB0, however, has normalization as a part of its architecture in TensorFlow, so we only rescale the blurred galaxy images back to a range $[0, 255]$, as required for the normalization layers. We test the following different configurations of the models:

1. Despite the models being pretrained over the ImageNet dataset, whose images are completely different from galaxies, we begin by freezing all the training layers and removing the classification head. This is so that we can initially determine how suitable the pretrained feature extraction is for galaxy images. We append two additional linear layers to project down to the PSF parameter dimension from the extracted features of each model.

2. We finetune all of the original layers of the pretrained models (without the classification head), with the same additional linear layers as configuration 1.

3. We finally train the models from scratch without the pretrained weights, with the expectation that the deeper models can better adapt to our specific dataset and correspondingly learn the necessary salient features to infer the PSF parameters. We incorporate the same additional linear layers as configuration 1.

4.2. Direct PSF Inference Model

4.2.1 Conv-TConv Model

Our final approach is to tackle the more challenging task of recovering the entire 2D PSF as opposed to the Moffat parameters, similar to what was done in [14]. We achieve this by using successive convolution layers to downsample to a spatial size of 8×8. We use five convolution blocks, with batch normalization, maximum pooling, and filter counts of $\{64, 64, 128, 128, 128\}$. Each convolution layer uses a 3×3 filter with stride of 1 and padding such that the output size remains the same. Downsampling is only done during the pooling layers.

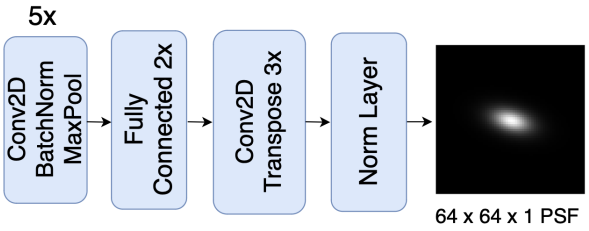

Figure 4. Model architecture for predicting entire 2D PSF. We note that the "Norm Layer" corresponds to the renormalization of the PSF pixels.

Because the features of the galaxies themselves may not directly give information about the full 2D PSF structure, we flatten the spatial features and pass them through additional linear layers. We then upsample using three transposed convolutions (hence "TConv") to get back up to a single-channel 64×64 PSF. The transposed convolutions use filters counts of $\{256, 128, 1\}$ with a stride of 2. We use "same" padding so that with the stated stride size, the resulting spatial dimensions can been doubled after each layer. We incorporate a final custom normalization layer to ensure that the PSF integrates to one. The resulting model architecture is shown in Figure 4.

4.3. Loss

As the task of PSF prediction is effectively one of regression, we optimize all models using the mean squared error

(MSE) loss:

$$\mathcal{L} = -\frac{1}{MN}\sum_{i=1}^{N}\sum_{j=1}^{M}(y_{ij} - \hat{y}_{ij})^2, \tag{8}$$

where N is the number of examples and M is either the number of PSF parameters (4) or the number of pixels in 2D PSF (64^2). For the models where we infer the PSF parameters, y_{ij} is the true i-th PSF's j-th parameter. For the Conv-TConv model, y_{ij} represents the i-th 2D PSF's j-th pixel value. $\hat{y}_{ij}$ is then the corresponding predicted parameter or pixel value.

5. Experiments and Results

5.1. Evaluation Methods

To examine the quality of the generated PSFs quantitatively, we analyze the mean squared error of different quantities of interest:

1. **All PSF Parameters**. This lets us compare which among the models that infer the PSF parameters directly perform better as a whole, testing recovery for α, β, ϵ, and θ. Among the parameter inference models, this is the primary point for direct comparison, as this is what we are directly optimizing over in Equation (8).

2. **Individual Parameters**. Again, among the models that infer the PSF parameter directly, this lets us identify which parameter experienced the best recovery. This also given indication as to which specific features the models may have focused on more or less.

3. **2D PSFs**. This allows us to do a direct comparison between the resulting PSF structures that would nominally be used for image deconvolution. Differences in the parameters may become less or more obvious in the corresponding 2D PSF image. This also lets us compare all of the parameter inference models with Conv-TConv as we can use the inferred parameters to generate a 2D PSF.

For qualitative analysis, we visually compare the inferred 2D PSFs from the models to the truth. We also examine their residuals to identify PSF orientation or shape misalignments.

5.2. Training and Quantitative Results

All models were trained using 20,000 blurred galaxy images with a hold out validation set of 2,500 images, as described in Section 3, for evaluation and hyperparameter tuning. For all the models apart from the pretrained models, we find that a learning rate of 10^{-3} yields reasonable training and validation loss convergence. Due to

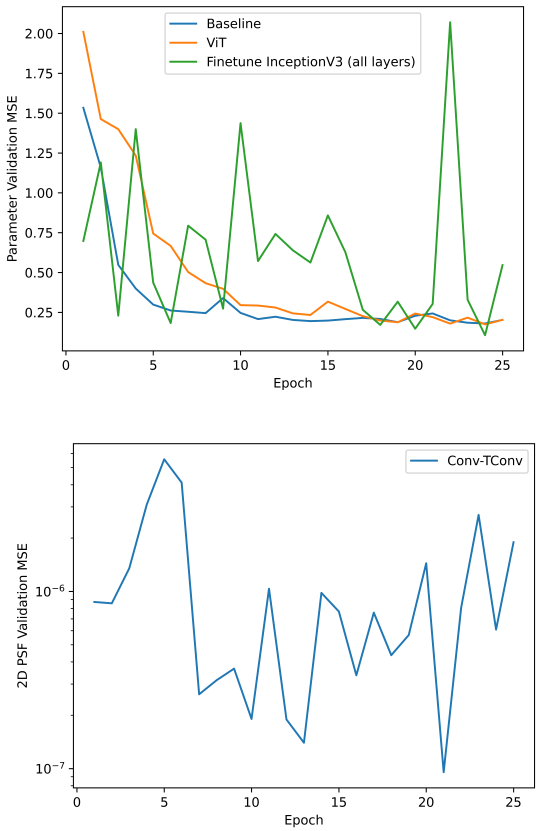

Figure 5. Validation loss histories for the parameter inference models (top) and the 2D PSF inference model (bottom).

the larger depth of the pretrained Inception and EfficientNet models, we use a smaller learning rate of 10^{-4} as a starting point (discussed in the following paragraph). We use ADAM optimization [12] with the default Keras parameters of $(\beta_1, \beta_2) = (0.9, 0.999)$ and $\epsilon = 10^{-7}$. For all the parameter estimation models, we use a batch size of 32, which should be large enough for batch normalization layers to achieve reasonable sample statistics for the overall dataset. For the Conv-TConv model, we choose a batch size of 16 as a result of having to use the 64×64 2D PSFs during training, which take up $\sim 10^3$ times the memory per training example. For each model/configuration, we train for 25 epochs over the entire training set, and we save the model with the best validation MSE. The evolution of the validation MSE for select models is shown in Figure 5.

We note that we were unable to achieve comparable results to other parameter inference models for all configurations of the EfficientNetB0 model (frozen base layers, finetuning all layers, and training from scratch). We found immediate overfitting after the first few epochs, as the validation MSE largely fluctuated around 1.0 or diverged while the training MSE monotonically decreased. We experi-

	All	α	β	ϵ	θ	2D PSF
Baseline	0.186	0.319	0.060	0.007	0.356	1.041×10^{-6}
ViT	0.182	0.359	0.056	0.018	0.295	1.857×10^{-6}
Finetune	0.110	0.165	0.072	0.003	0.200	3.623×10^{-7}
Conv-TConv	-	-	-	-	-	9.967×10^{-8}

Table 1. We report the test dataset MSE results for all the parameters (column 1), each individual parameter (column 2-5), and the reconstructed 2D PSF. We note that for Conv-TConv we do not report the parameter-wise MSE values as parameters were not inferred by the model.

mented with adding L2 regularization and a decaying learning rate; namely, starting from 10^{-4}, we reduce the learning rate by 10% after each epoch to increase stability during training. We found that these modifications did not yield significant improvements. We find a best validation MSE of ~ 0.8 by finetuning all the layers, ~ 8 times worse than that of the other approaches. For InceptionV3, the only configuration that yielded comparable validation losses to ViT and Baseline was finetuning all layers with a fixed learning rate of 10^{-4} without needing any regularization, which is what we show in Figure 5 and later use for analysis on the test set. The other model configurations of InceptionV3 exhibit similar behavior to that of EfficientNetB0. Nevertheless, we find significant fluctuation in the validation MSE over the 25 epochs.

For the Baseline and ViT models, we see a less noisy convergence to low validation MSE values at ~ 0.18 that are very close to each other by around epoch 15. These models, however, do not achieve a better validation MSE over the finetuned InceptionV3 model (which we call "Finetune" from now on), which found a minimum validation MSE of ~ 0.11 before the last epoch. All of the parameter-inference models started overfitting by around epoch 15, where the validation losses plateaued while the training loss continued to decrease. For the Conv-TConv model, we see relatively quick convergence by around 10-15 epochs, with clear signs of overfitting as the validation loss starts to increase globally by epoch 25 with significant fluctuations. However, we reach the minimum validation MSE at epoch 21 with a value $\sim 10^{-7}$, a whole order of magnitude better than the starting epoch.

Using the model weights that achieved the best validation accuracy over the 25 epochs, we evaluate on the test set of 2,500 blurred galaxy images. The MSE of the quantities listed in Section 5.1 are shown in Table 1. We see that the Finetune model performs the best for PSF parameter inference, as indicated by its overall parameter MSE of 0.110, which is $\sim 40\%$ better than the baseline and ViT. We also observe that all the parameter inference models seem to struggle the most with inferring the size of the blurring α and the orientation of the blurring θ given their comparatively larger MSE values. This is somewhat unexpected for α as its size directly impacts image quality. For θ, however, if the PSF is sufficiently circular with $\epsilon \sim 0$, then θ will be degenerate and hence difficult to infer. Since we are directly inferring from the blurred images rather than the PSF, information about α and θ may still be difficult to discern. Unsurprisingly, we see that the Conv-TConv model, which directly infers the 2D PSF, yields the best performance when reconstructing the full PSF. In particular, it yields a test MSE that is over an order of magnitude lower than the Baseline and ViT models and ~ 3 times lower than the Finetune model. This result suggests that even slight differences in the Moffat parameters among the predicted and truth can yield quite different 2D PSFs, thus explaining the higher MSE values for the parameter-estimation models.

Finally, we note that the training and test results among the pretrained models strongly indicate that they have significant potential to perform well on galaxy PSF estimation, but training them is difficult. It appears that small changes to the parameters (as indicated by small decreases in training loss over epochs) can quickly degrade generalization. Perhaps, far more data than currently used is required to achieve more reliable training with pretrained models.

5.3. Qualitative Examination of PSFs

Although the parameter estimation models yield clean symmetric PSFs, as indicated by the test MSE on the reconstructed 2D PSFs, there may still be significant disagreements with the truth. These disagreements are on average more than that of the Conv-TConv model, which is not even guaranteed to return an axisymmetric PSF. In Figure 6, we show the resulting 2D PSF output of the four models on an unseen blurred galaxy image whose PSF has noticeable anisotropy. Alongside the inferred PSFs are the residuals, plotted with the same scaling across all the models.

In this example, all the models do well visually to reproduce the overall shape and orientation of the true PSF. However, we observe that the major axis of the predicted PSF of the Baseline model is several degrees off from the truth, exemplifying the comparatively larger MSE values we saw in the test set for θ. Combined with the somewhat smaller ellipticity than the truth, this misalignment results in distinct dark/bright spots in the residual map corresponding to regions where the predicted PSF was brighter/darker than the truth, respectively. In particular, the larger minor axis of the predicted PSF yields an overestimation to the upper-left and lower-right compared to the truth, resulting in a dearth

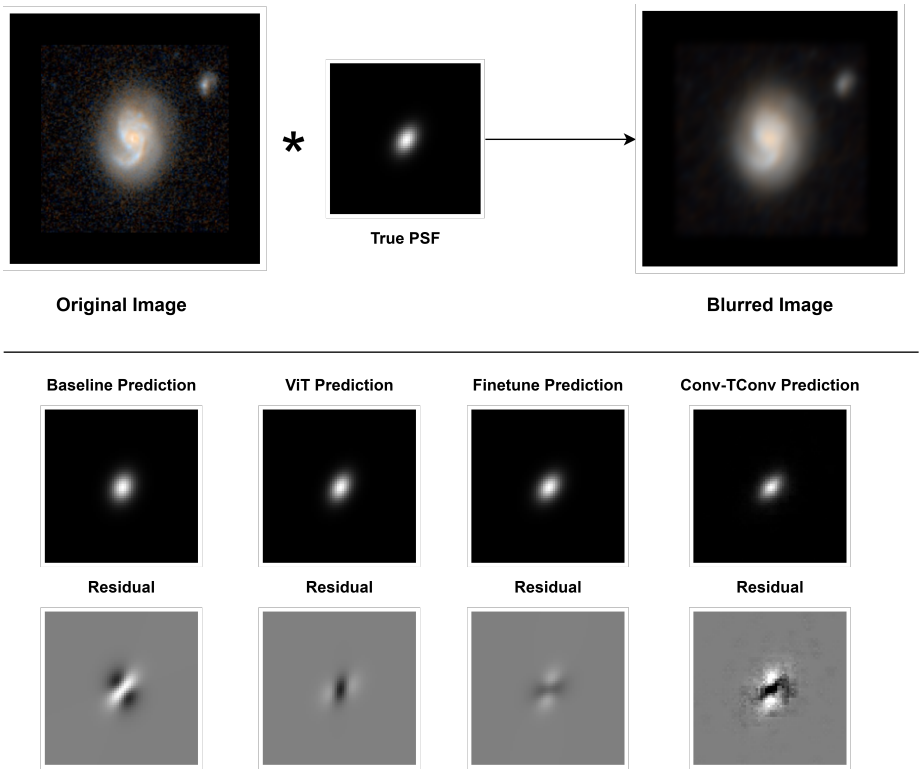

Figure 6. PSF predictions for a sample galaxy image. Above the dividing line, we show the unblurred galaxy and the effect of convolving it with the true PSF. Below, we show the resulting predictions made by select models and the corresponding residuals (truth - predicted). The scaling for the residuals are all on the same scale from a minimum of -0.01 to a maximum 0.01 in difference from the true PSF.

of signal at the center and in the major axis direction.

In this example, the ViT and Finetune models perform the best, as exemplified by their much dimmer residuals. This is because the ellipticities and the alignment agree more with the true PSF. We emphasize that this is not always the case across other examples, where both ViT and the Finetuned model can still suffer from several degrees of misalignment like the Baseline, as exemplified by their high θ MSE as well. Despite the Conv-TConv model having no constraints beyond normalization, we observe that it does well to recover the true PSF and visually maintain axisymmetry. However, the ellipticity of the predicted PSF is slightly smaller, thus yielding brighter spots in the minor-axis direction, opposite to what was seen in the Baseline. Furthermore, the outskirts of the PSF are filled with noisy pixels, which may generally not be desired in image deconvolution due to its sensitivity to noise. Further visual inspections of other unseen examples yield similar results.

6. Conclusion

We experimented with several models to estimate the PSF from blurred galaxy images whose PSFs are described by an anisotropic Moffat function, achieving strong PSF recovery both quantitatively and qualitatively. For PSF parameter estimation, we implemented a baseline shallow convolutional neural network and a vision transformer network. We also finetuned all layers of the InceptionV3 pretrained model. Despite exhibiting non-monotonic training improvements, we were able to achieve the best parameter estimation overall with the finetuned model, yielding a test MSE across all parameters of 0.110, $\sim 40\%$ lower than the other two. For 2D PSF estimation, we implemented a neural network with transposed convolutions to reconstruct the entire PSF. We find that the direct PSF estimation model yields the best PSF recovery when comparing the PSFs directly, but it may not generally be desirable for PSF deconvolution, given the tendency to produce noisy pixels.

Future work will investigate the nature of the noisy performance of finetuning pretrained models on PSF estimation, additionally exploring the effects of model depth and complexity, dataset size, and image resolution. Additional approaches that are worth exploring include using variational autoencoders for PSF parameter regression similar to what was done in [24] and investigating different upsampling techniques to mitigate noise in full PSF recovery. Finally, we might consider testing the ability of these models to perform on much more complicated PSF models such as those with an airy disk component, which contains concentric rings around a bright central region.

7. Contributions

Tirth worked on the dataset generation and implemented the baseline and finetuned models. Riley worked on the ViT model while Ishaan worked on the on the Conv-TConv model. All group members contributed to the analysis for their respective models.

References

[1] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[2] J. Binney and M. Merrifield. *Galactic Astronomy*. 1998.

[3] R. J. Buta. *Galaxy Morphology*, pages 1–89. Springer Netherlands, Dordrecht, 2013.

[4] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.

[5] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.

[6] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale, 2021.

[7] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition, 2015.

[8] D. Hendrycks and K. Gimpel. Gaussian error linear units (gelus), 2023.

[9] J. Herbel, T. Kacprzak, A. Amara, A. Refregier, and A. Lucchi. Fast point spread function modeling with deep learning. *Journal of Cosmology and Astroparticle Physics*, 2018(07):054, jul 2018.

[10] P. Jia, X. Wu, Z. Li, B. Li, W. Wang, Q. Liu, A. Popowicz, and D. Cai. Point spread function estimation for wide field small aperture telescopes with deep neural networks and calibration data. *Monthly Notices of the Royal Astronomical Society*, 505(4):4717–4725, 05 2021.

[11] L. Jiang, I. D. McGreer, X. Fan, M. A. Strauss, E. Bañados, R. H. Becker, F. Bian, K. Farnsworth, Y. Shen, F. Wang, R. Wang, S. Wang, R. L. White, J. Wu, X.-B. Wu, J. Yang, and Q. Yang. The Final SDSS High-redshift Quasar Sample of 52 Quasars at z¿5.7. , 833(2):222, Dec. 2016.

[12] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization, 2017.

[13] C. Lintott, K. Schawinski, S. Bamford, A. Slosar, K. Land, D. Thomas, E. Edmondson, K. Masters, R. C. Nichol, M. J. Raddick, A. Szalay, D. Andreescu, P. Murray, and J. Vandenberg. Galaxy zoo 1: data release of morphological classifications for nearly 900000 galaxies: Galaxy zoo. *Monthly Notices of the Royal Astronomical Society*, 410(1):166–178, Nov. 2010.

[14] M. Long, Y. Soubo, N. Weiping, X. Feng, and Y. Jun. Point-spread function estimation for adaptive optics imaging of astronomical extended objects. *The Astrophysical Journal*, 888(1):20, 2019.

[15] A. F. J. Moffat. A Theoretical Investigation of Focal Stellar Images in the Photographic Emulsion and Application to Photographic Photometry. , 3:455, Dec. 1969.

[16] V. C. R. Observatory. Vera c. rubin observatory science goals, May 2020.

[17] K. Salama. Image classification with vision transformer. Keras.io, 2021.

[18] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition, 2015.

[19] S. D. S. Survey. Instruments, 2022.

[20] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. Rethinking the inception architecture for computer vision, 2015.

[21] M. Tan and Q. V. Le. Efficientnet: Rethinking model scaling for convolutional neural networks, 2020.

[22] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need, 2023.

[23] K. W. Willett, M. A. Galloway, S. P. Bamford, C. J. Lintott, K. L. Masters, C. Scarlata, B. D. Simmons, M. Beck, C. N. Cardamone, E. Cheung, E. M. Edmondson, L. F. Fortson, R. L. Griffith, B. Häußler, A. Han, R. Hart, T. Melvin, M. Parrish, K. Schawinski, R. J. Smethurst, and A. M. Smith. Galaxy Zoo: morphological classifications for 120 000 galaxies in HST legacy imaging. , 464(4):4176–4203, Feb. 2017.

[24] Q. Zhao, E. Adeli, N. Honnorat, T. Leng, and K. M. Pohl. Variational autoencoder for regression: Application to brain aging analysis, 2019.