Fast Nonlinear Regression via Eigenimages Applied to Galactic Morphology

Brigham Anderson and Andrew Moore
Robotics Department
Carnegie Mellon University
Pittsburgh, PA 15217
brigham@cmu.edu
awm@cs.cmu.edu

Andrew Connolly
Astrophysics Department
University of Pittsburgh
Pittsburgh, PA 15213
ajc@phyast.pitt.edu

Robert Nichol
Astrophysics Department
Carnegie Mellon University
Pittsburgh, PA 15217
nichol@cmu.edu

ABSTRACT

Astronomy increasingly faces the issue of massive datasets. For instance, the Sloan Digital Sky Survey (SDSS) has so far generated tens of millions of images of distant galaxies, of which only a tiny fraction have been morphologically classified. Our aim is to reduce each dataset image to a small set of informative features, in this case by using a known parameterized model of the image contents, and replacing each image with its best-fit parameters. This is a standard nonlinear regression problem, whose challenges are fourfold, 1) the atmospheric and mirror-based distortion suffered by each image, 2) large numbers of local minima, 3) large amounts of noise, and 4) the speed required to cope with the massiveness of the datasets.

Our strategy is to use the known model's eigenimages to form a new basis, then to map both the target images and the model parameters into this eigenspace, and finally to find the best image-to-parameter matches within the space. To do this, we create a database of many random sets of parameters and their locations in eigenspace, thereby making the fitting process a nearest-neighbor search. Complications arise in the form of missing data and heteroskedasticity, both of which are addressed with weighted linear regression. Compared to existing techniques, speedups achieved are between 2 and 3 orders of magnitude. This enables the analysis of the entire SDSS dataset, itself a scientific wealth.

1. INTRODUCTION

In order to understand the formation of large scale structures in the universe, a necessary step is the understanding of the varied galaxy morphologies. This is still an open area of research in astronomy; it is not known how galaxy shapes arise. The distribution of shapes and their correlation with other measured properties of galaxies is important to generating and testing hypotheses about the basic nature of the universe. This endeavor requires extracting various types of information from large numbers of faint and noisy images of galaxies, e.,g., whether the galaxy is spherical, elliptical, or disk-shaped, the size of the central bulge relative to the size of the disk, etc. Some examples of such images are in Figure 1.

There are significant obstacles, however, to fitting these models. The most challenging is the Point Spread Function (PSF.) Taking images of galaxies from ground-based telescopes involves having the image smeared by a turbulent atmosphere, and distortion also results from lens imperfections (telescope, filters, mirrors, lenses, etc.) All these effects can be summarized in a PSF, which in this study takes the form of an image, e.g., a two-dimensional Gaussian. As its name implies, the PSF is the spread that a point source of light would display had it been centered on the central detector/pixel. The PSF can also be viewed as a probability mass function for the arrival of a single photon at a given pixel, given that the photon was initially aimed at the center pixel.

Large numbers of local minima are another feature of this problem. The noise of the images and a sometimes too-flexible model make finding the correct fit difficult. Some form of global search is generally necessary, and this is quite time-consuming. The most trusted of the current 2-d morphology techniques is a simulated annealing algorithm, which is robust to local minima, but is slower due to its caution.

The state of the art is to use standard nonlinear regression techniques to fit these images, such as simulated annealing [6] and Levenberg-Marquardt [4]. These approaches are all effective, but time-consuming, e.g., roughly 1-3 minutes per 64×64 image on a 1.4 GHz pentium desktop, so 10 million galaxies would require about 20 years of CPU time. For higher resolution images, performance rapidly degrades. The code introduced in this paper performs the same fits in less than a second, and is robust to changes in resolution of the target image. Other machine learning approaches to similar problems have analyzed the use of EM in classifying certain "bent-double" galaxies [2] with good success.

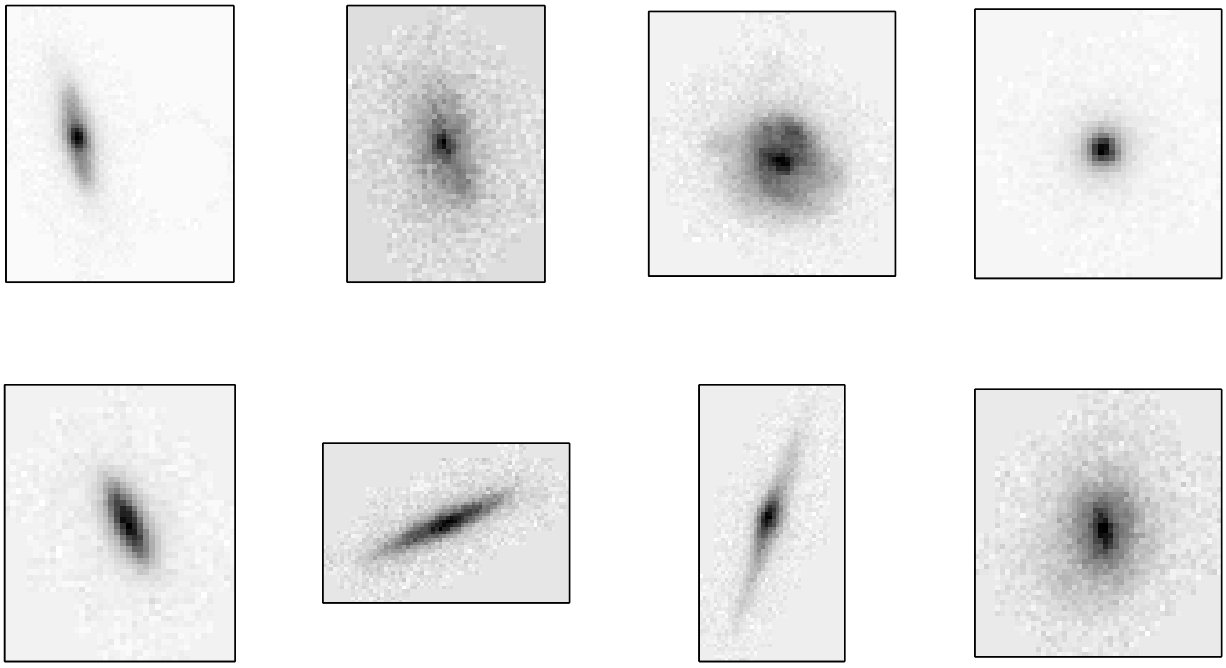

Figure 1: Galaxy images taken from the Sloan Digital Sky Survey

2. PROBLEM

The general task is to fit a model to data, in this case the data is in the form of images of galaxies. In the galactic morphology task (herein referred to as the GM task), the model is a function whose 12 free parameters are the morphological characteristics of the galaxy, e.g., shape, size, and location. This task is described in detail in [6]. We assume all images are square with N pixels. Notationally, all images are represented as vectors, e.g., the columns of the image are vertically concatenated. All vectors are boldfaced or denoted with an overhead arrow ($\vec{\cdot}$).

2.1 Generative Model

The most basic assumption that we make is that the target image can be modelled. Here we use a model consisting of set of four main elements: $\{\boldsymbol{\psi}, \Theta, \Pi, \boldsymbol{\varepsilon}\}$:

- $\boldsymbol{\psi}$ is a sum of c component nonlinear functions: $\boldsymbol{\psi}(\boldsymbol{\theta}; \boldsymbol{\pi}) = \sum_i^c \boldsymbol{\psi}_i(\boldsymbol{\theta}_i; \boldsymbol{\pi})$. In the galactic morphology task, $\boldsymbol{\psi}$ is a standard surface brightness function which has three components: a disk, a bulge, and constant background.
- Θ is the space of possible values for the fittable parameters of $\boldsymbol{\psi}$. In the GM task, these parameters are disk flux, disk angle, disk inclination, bulge flux, bulge xy location, etc. (see Appendix.)
- Π is the space of possible values for the *fixed* parameters of $\boldsymbol{\psi}$. These parameters are those that vary from regression to regression, but are not controllable or fittable. For the galactic morphology problem, this is the PSF, $\boldsymbol{\pi}$, since the PSF varies from image to image. The Π space contains all the PSFs that could possibly occur.
- $\boldsymbol{\varepsilon}$ is a noise model which also varies from regression to regression. For this galactic morphology problem, we assume additive, zero-mean gaussian noise, where $\boldsymbol{\varepsilon}_i$ is the variance of the noise at i-th pixel of the model image. This allows for heteroscedastic noise. For simplicity, the $\boldsymbol{\varepsilon}$ component will be omitted in the text where it is unnecessary.

The elements of the model above are illustrated in Figure 2. The relationships between the elements are also described in the following equation:

$$\boldsymbol{\psi}(\boldsymbol{\theta}, \boldsymbol{\pi}) = \sum_i^c \boldsymbol{\psi}_i(\boldsymbol{\theta}, \boldsymbol{\pi}) + \boldsymbol{\varepsilon} \tag{1}$$

where $\boldsymbol{\theta} \in \Theta$. The model describes the probability distribution function for the image; each pixel has an independent Gaussian distribution: pixel i is distributed as $N\left(\psi(\boldsymbol{\theta}, \boldsymbol{\pi})_i, \varepsilon_i\right)$. This is a distribution over model's output space. In the GM task, the output space is equal to $\mathbb{R}^N$.

2.1.1 The Function ψ

The model $\boldsymbol{\psi}$ is a function on a 2-dimensional plane which indicates the amount of flux received at a given point on the plane. The general shape of the function is a sharp peak at the center of the galaxy, tapering off with distance. The disk tapers off exponentially with respect to distance from the center of the galaxy, and the bulge tapers off exponentially w.r.t. the cube root of the distance. The appendix contains the details of the function and its motivation. Importantly, the model is not smooth and has no derivatives at $(0, 0)$.

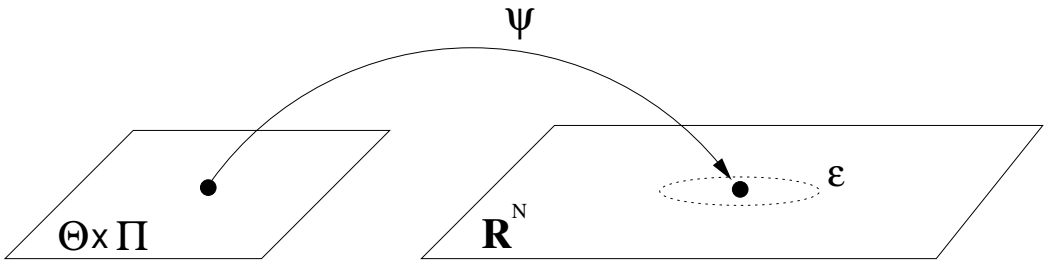

Figure 2: The generative model. Parameters from the space $\Theta \times \Pi$ are mapped into image space, $\mathbb{R}^N$. The error model ε describes the variance in each of the N dimensions of $\mathbb{R}^N$.

There is a single sharp spike at this location, which creates difficulties later on.

2.1.2 The Fixed Parameter(s) π

In the GM task, the relationship between the fixed parameter $\boldsymbol{\pi}$ (the PSF) and the model $\boldsymbol{\psi}$ is one of convolution[1]:

$$\boldsymbol{\psi}(\boldsymbol{\theta}, \boldsymbol{\pi}) = \boldsymbol{\pi} \star \boldsymbol{\psi}(\boldsymbol{\theta}) \tag{2}$$

Since $\boldsymbol{\pi}$ is an image of the PSF, convolving an image with a particular $\boldsymbol{\pi}$ is equivalent to blurring with a particular atmospheric condition and/or mirror imperfection. Fortunately, convolution is a linear operation, since each resulting pixel is a linear combination of all other pixels. Hence, Equation 2 can be rewritten as

$$\boldsymbol{\psi}(\boldsymbol{\theta}, \boldsymbol{\pi}) = \boldsymbol{\pi} \star \boldsymbol{\psi}_{disk}(\boldsymbol{\theta}) + \boldsymbol{\pi} \star \boldsymbol{\psi}_{bulge}(\boldsymbol{\theta}) \tag{3}$$

Due to the physical interpretation of $\boldsymbol{\pi}$, all the pixels must be nonnegative and sum to one. Since light passes through the atmosphere and mirror before hitting the detector, the PSF is convolved with the image *before* the Poisson noise is added, as is reflected in Equation 1.

2.1.3 The Noise Model ε

The noise model $\boldsymbol{\varepsilon}$ in general will be different for every target image. The noise in the image comes primarily from a Poisson distribution. The brightness function, $\boldsymbol{\psi}(\boldsymbol{\theta})$, only describes the *expected* number of photons or counts at the detector/pixel during the exposure. The actual number of photons or "counts" that arrive at pixel i will follow a Poisson distribution whose mean and variance are equal to the value of $\psi(\boldsymbol{\theta})_i$. This results in an image whose error variance at a pixel is proportional to the amount of signal at that pixel. This heteroscedasticity must be accounted for in the regression. We consider the Poisson to be well approximated in this case with a Gaussian distribution, since the number of photons is usually greater than 30 in the more influential central pixels.

The noise model can also be used to effect a "mask" for the input image. If the galaxy of interest is close to another galaxy, or artifacts are present in the image, then a mask can be used to specify which pixels are to be fit. By setting a pixel's ε value to be infinite, bad pixels can be masked out entirely.

[1]The convolution operator is denoted as a $\star$. Additionally, since the representation of all images are as vectors, any convolution between two vectors is assumed to take place with each vector's *image* equivalent.

2.2 Objective

The task is to take a given target image $\boldsymbol{y}$, PSF $\boldsymbol{\pi}$, and error model $\boldsymbol{\varepsilon}$, and to find a parameter vector $\boldsymbol{\theta}^* \in \Theta$ that, when fed to the generative model of Equation 1, produces an image close to $\boldsymbol{y}$. By 'close' we mean to minimize the most likely distance between the two images, taking into account the noise model. This distance function will be denoted by χ^2.

$$\chi^2(\boldsymbol{y}, \boldsymbol{\theta}, \boldsymbol{\pi}, \boldsymbol{\varepsilon}) = \sum_i^N \frac{(\boldsymbol{y}_i - \boldsymbol{\psi}(\boldsymbol{\theta}, \boldsymbol{\pi})_i)^2}{\boldsymbol{\varepsilon}_i^2} \tag{4}$$

This χ^2 is the distance, between the target image $\boldsymbol{y}$ and the image $\boldsymbol{\psi}(\boldsymbol{\theta}, \boldsymbol{\pi})$, taking into account the known noise properties of the target image.

3. NONPARAMETRIC REGRESSION

The algorithm must be able to invert the galaxy image model and to deconvolve images that have been blurred by a PSF, and it must be able to do so quickly. The most successful algorithm type we found has been instance-based, the nearest neighbor algorithm. This approach creates a mapping from galaxy image space $\mathbb{R}^N$ to parameter space Θ by remembering and generalizing from many previous Θ-to-$\mathbb{R}^N$ mappings (via prototypes).

We are thus performing a parametric regression using a nonparametric method. Several characteristics of the galaxy morphology problem motivated this choice, though primarily it is due to two reasons: 1) the model is expensive to evaluate because the PSF convolution requires a Fourier transform of the model (Equation 2) for every convolution. Iterative techniques need to evaluate the generative model at every step, so search becomes costly, and 2) the number of local minima is large, so most descent-based methods are inappropriate.

3.1 Brute Force

For purposes of exposition, we will start with the most direct approach. The prototypes that will be used to map from images to parameters are the members of the set of

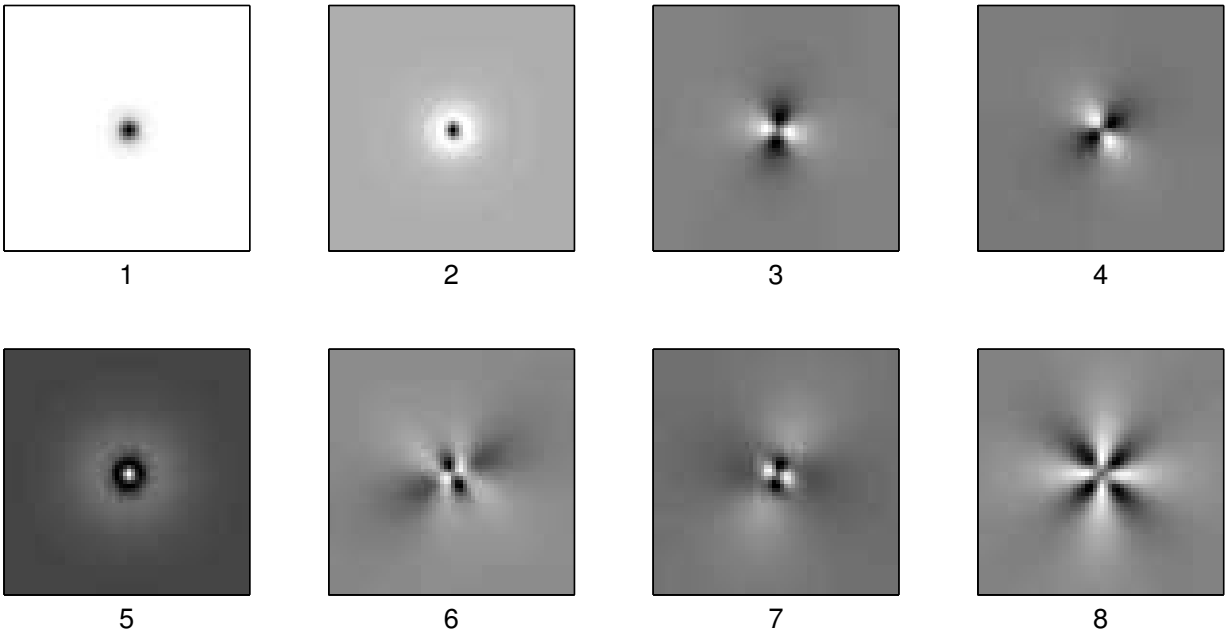

Figure 3: The first 8 eigengalaxies obtained from galaxies which have been convolved with a Gaussian PSF with 2-pixel standard deviation.

prototypes $\{(\boldsymbol{x}_i, \boldsymbol{\theta}_i, \boldsymbol{\pi}_i)\}_{i=1}^{p}$, wherein $\boldsymbol{x}_i = \boldsymbol{\psi}(\boldsymbol{\theta}_i, \boldsymbol{\pi}_i)$ and p is the number of prototypes. We generate this set by sampling uniformly from Θ and Π. This set of prototypes can more conveniently be discussed later if we line up the image vectors as the columns of a single matrix, X, so that the i-th column of X is the i-th prototype image.

The regression task in this context is to find θ^* by finding the smallest distance between the target image and each of the prototypes,

$$i^* = \mathsf{argmin}_{i=1..p} \left[\chi^2(\boldsymbol{x}_i, \boldsymbol{y}, \boldsymbol{\varepsilon})\right] \quad (5)$$

$$\boldsymbol{\theta}^* = \boldsymbol{\theta}_{i^*} \quad (6)$$

This is the core strategy of this regression algorithm. However, there are clear barriers to overcome. First, the dimensionality of the prototypes/input space is very high; one dimension per pixel for a 32×32 image is equal to 1024 dimensions. Comparing prototypes to incoming queries will thus be expensive. Second, the presence of the uncontrollable fixed parameters $\boldsymbol{\pi}$ means that a large number of prototypes will be required to adequately sample the parameter space $\Theta \times \Pi$. However, nearest neighbor search can exploit two strategies, which account for most of the speed gains made, 1) Principal Components Analysis, 2) PSF-local Principal Component Analysis, and 3) prototype decomposition.

4. EIGENSPACE CREATION

One problem with working in image space $\mathbb{R}^N$ is the computational load of so many dimensions, one per pixel. However, ignoring noise, the model $\{\boldsymbol{\psi}, \Theta, \Pi\}$ creates images that can *at most* occupy a manifold whose dimension is equal to the number of model parameters. Ideally, one should focus one's efforts in the most relevant subspace and ignore the rest. Principal Components Analysis (PCA) is often used to do just this, by determining the linear subspace in which most of the variance resides.

To determine the best subspace of the model's manifold, $\{\boldsymbol{\psi}(\boldsymbol{\theta}, \boldsymbol{\pi}) \mid \boldsymbol{\theta} \in \Theta, \boldsymbol{\pi} \in \Pi\}$ in $\mathbb{R}^N$, we would like to know the shape of the manifold. One measure of this shape is the covariance between the values of each dimension (pixel) of the images that the model produces.

$$UU^T = \int_{\Theta} \int_{\Pi} \left[\boldsymbol{\psi}(\boldsymbol{\theta}, \boldsymbol{\pi}) - \mu\right] \left[\boldsymbol{\pi} \star \boldsymbol{\psi}(\boldsymbol{\theta}, \boldsymbol{\pi}) - \mu\right]^T d\boldsymbol{\pi} d\boldsymbol{\theta} \quad (7)$$

where μ is the mean model image over all θ and π. This is the pixel covariance matrix, UU^T, whose ij-th element is the covariance between pixel i and pixel j in the model. Once the pixel covariance is known, we can use PCA to determine an "optimal" linear subspace of $\mathbb{R}^N$ that captures the most model variance per subspace dimension.

Before commencing with PCA, we must first estimate UU^T. This can be done efficiently by sampling the model manifold, i.e., by randomly choosing $\boldsymbol{\theta}$s and $\boldsymbol{\pi}$s from the parameter space $\Theta \times \Pi$ and generating images from them using the model $\{\boldsymbol{\psi}, \Theta, \Pi\}$. These sample images are then mean-normalized[2], lined up as column vectors into the matrix U, and multiplied to produce UU^T.

In the PCA paradigm, the first $K \ll N$ eigenvectors of

[2]The mean of all sample images is subtracted from each image.

UU^T form the basis for a new space, a space which optimally captures the most model variance. The basis, Φ, is an $N \times K$ matrix, whose orthonormal columns are the first K eigenvectors of UU^T. The span of the columns of Φ will be referred to as an eigenspace and the individual columns as eigenimages. See Figure 3 for the first eight eigengalaxies. Note that Φ^T is a projection matrix such that $\tilde{\boldsymbol{y}} = \Phi^T \boldsymbol{y}$. Vectors in eigenspace will be denoted by $\tilde{\bullet}$.

4.1 Projection into Eigenspace

The entire nearest neighbor search should now take place within the eigenspace Φ. This requires projecting all of the prototype images and all incoming input images into Φ. Since the eigenvectors are orthogonal, projection of either type of image into the eigenspace should be straightforward:

$$\tilde{\boldsymbol{y}} = \Phi^T \boldsymbol{y} \tag{8}$$

However, if the noise is heteroscedastic, the projection requires more care. The pixels with less noise should receive relatively more weight than pixels with high noise. Given our noise model, the optimal projection is a weighted linear regression. The diagonal matrix Σ_y is the given covariance of $\boldsymbol{y}$, which is merely a reorganization of $\boldsymbol{\varepsilon}$:

$$\mathsf{diag}(\Sigma_y) = \boldsymbol{\varepsilon} \tag{9}$$

The projection of $\boldsymbol{y}$ is

$$\tilde{\boldsymbol{y}} = (\Phi^T \Sigma_y^{-1} \Phi)^{-1} \Sigma_y^{-1} \Phi^T \boldsymbol{y} \tag{10}$$

and the resulting error covariance matrix of $\tilde{\boldsymbol{y}}$ is

$$\Sigma_{\tilde{y}} = (\Phi^T \Sigma_y^{-1} \Phi)^{-1} \tag{11}$$

The matrix $\Sigma_{\tilde{y}}$ is the covariance matrix of $\tilde{\boldsymbol{y}}$, reflecting any uncertainty in the eigenspace projection of $\boldsymbol{y}$.

4.2 Nearest Neighbor in Eigenspace

Now we can attack the regression problem while inside the eigenspace, where we enjoy a much reduced dimensionality. Now the algorithm uses only the eigencoordinates of the prototypes, denoted $\tilde{X}$. The algorithm is slightly more complicated since the χ^2 distance function must now account for any correlated errors in $\tilde{\boldsymbol{y}}$ introduced in Equation 4. The new algorithm is as follows:

$$i^* = \mathsf{argmin}_{i=1..p} \left[(\tilde{\boldsymbol{x}}_i - \tilde{\boldsymbol{y}})^T \Sigma_{\tilde{y}}^{-1} (\tilde{\boldsymbol{x}}_i - \tilde{\boldsymbol{y}}) \right] \tag{12}$$

$$\boldsymbol{\theta}^* = \boldsymbol{\theta}_{i^*} \tag{13}$$

where $\tilde{\boldsymbol{x}}$ and $\tilde{\boldsymbol{y}}$ are the eigencoordinates of $\boldsymbol{x}$ and $\boldsymbol{y}$ respectively.

5. PSF-LOCAL PCA

Up to this point we have included all possible PSFs in our PCA. The model manifold $\{\boldsymbol{\psi}(\boldsymbol{\theta}, \boldsymbol{\pi}) \mid \boldsymbol{\theta} \in \Theta, \boldsymbol{\pi} \in \Pi\}$ is quite large, as is the number of eigenspace dimensions required to represent it adequately. There are at least two approaches to this problem: attempting to remove the effect of $\boldsymbol{\pi}$ and PSF-local PCA.

The most direct approach is to modify $\boldsymbol{y}$ to remove the effect of the fixed parameters $\boldsymbol{\pi}$. This can be accomplished via deconvolution. Unfortunately, deconvolution has difficulty with regions of the image with sudden changes in intensity, which is the part of the image with the most information relevent to our model. The central spike (which is the galaxy center) is a discontinuity that is very difficult for deconvolution to reconstruct. Also, deconvolution suffers from instability in the presence of noise.

The next approach is PSF-local PCA. This maintains a different Φ for every PSF. Each eigenspace is obtained by fixing $\boldsymbol{\pi}$ and repeating the steps of PCA in Section 4. Each eigenspace is optimal for its $\boldsymbol{\pi}$. The number of dimensions required is therefore quite small, approximately 20 dimensions.

This is feasible because most PSFs in a given run of galaxy images are similar, and because small differences between PSFs generally produce small differences in resulting images.

The algorithm stores a relatively small number n_s of PSF-specific eigenspaces $\{(\boldsymbol{\pi}_i, \Phi_i)\}_{i=1}^{n_s}$. The initial PSF population consists of a few Gaussians of varying standard deviation, to which PSFs are added during on-line operation. The decision to add a PSF to the database is made, somewhat arbitrarily, when an incoming PSF differs by more than 0.02 in variance explained from its closest match in the database. Variance explained here is $1 - \Sigma_i^N (\pi_i - \pi_i')^2 / \Sigma_i \pi_i'^2$. Where $\boldsymbol{\pi'}$ is a PSF from the existing database.

6. PROTOTYPE DECOMPOSITION

The fact that the model is of the form $\boldsymbol{\psi}(\boldsymbol{\theta}, \boldsymbol{\pi}) = \sum_i^c \boldsymbol{\psi}_i(\boldsymbol{\theta}, \boldsymbol{\pi})$ can be used to good advantage; only prototypes for the component $\boldsymbol{\psi}_i(\boldsymbol{\theta}, \boldsymbol{\pi})$ functions need to be generated and stored.

For example, in the GM task the component functions are the disks, bulges, and background. Instead of generating and storing large numbers of individual combinations of disks and bulges (the prototype set X) we can instead store two much smaller prototype sets, a disk set and a bulge set (X_d and X_b). Far fewer prototypes will be needed to represent essentially the same information as before. The size of the representable number of prototypes is now $|X_d| \times |X_b|$.

Typically, nearest neighbor has only to look for neighbors of a target which are single points. Since we are using component prototypes, we must define a distance metric between a particular combination of component prototypes $\{\tilde{\boldsymbol{x}}^{\boldsymbol{0}}, \tilde{\boldsymbol{x}}^{\boldsymbol{1}}, \tilde{\boldsymbol{x}}^{\boldsymbol{2}}, ..., \tilde{\boldsymbol{x}}^{\boldsymbol{c}}\}$ and a target image $\tilde{\boldsymbol{y}}$. In the GM task, this would be finding the distance between a galaxy image and, for instance, bulge #55 with disk #1244. Given our error definition, the appropriate metric is the shortest distance is obtained by projecting $\tilde{\boldsymbol{y}}$ onto the plane defined by the c component prototypes. This distance χ^2 is calculated via weighted linear regression:

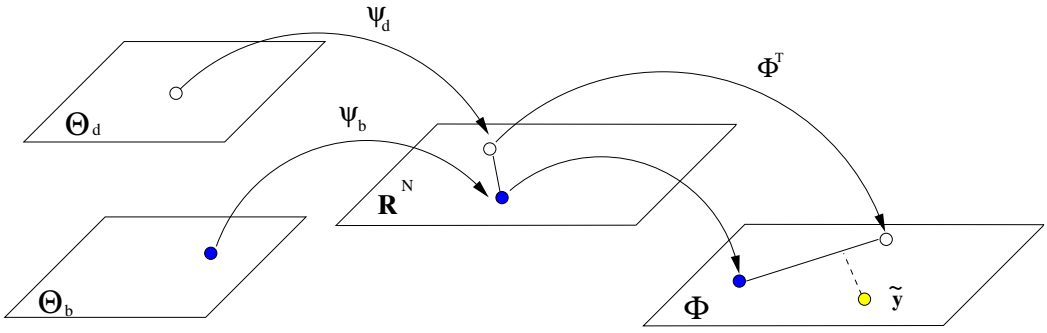

Figure 4: Fitting image $\tilde{y}$ to component prototypes. The fitting is done in eigenspace Φ. The empty circle is a disk component and the filled circle is a bulge component. The intersection of the dashed line and the solid line is the best-fitting linear combination of this particular bulge and disk to $\tilde{y}$.

$$Z = \left[\tilde{\boldsymbol{x}}^{\mathbf{0}}\ \tilde{\boldsymbol{x}}^{\mathbf{1}}\ \tilde{\boldsymbol{x}}^{\mathbf{2}}\ ...\ \tilde{\boldsymbol{x}}^{\mathbf{c}}\right] \quad (14)$$

$$\beta = (Z^T \Sigma_{\tilde{y}}^{-1} Z)^{-1} \Sigma_{\tilde{y}}^{-1} Z^T \tilde{\boldsymbol{y}} \quad (15)$$

$$\chi^2 = \tilde{\boldsymbol{y}}^T \Sigma_{\tilde{y}}^{-1} \tilde{\boldsymbol{y}} - \beta^T Z^T \Sigma_{\tilde{y}}^{-1} \tilde{\boldsymbol{y}} \quad (16)$$

which also determines the optimal linear combination coefficients, β, of the prototypes for that particular target $\tilde{\boldsymbol{y}}$. In the case of the GM problem, the coefficients β are the optimal amounts of disk, bulge, and background. Figure 4 illustrates the procedure.

At this point, we in principle have only to calculate χ^2 for all combinations of disks and bulges, and select the combination with the lowest error. Unfortunately, speed would then be unacceptably compromised, so instead we search selectively.

7. NEAREST NEIGHBOR SEARCH

After the eigenspace has been selected and the target image has been projected into the space, then the search for a nearest neighbor begins. The search could be accomplished by an exhaustive search of all bulge-disk combinations. However, we save time with the following two-part search algorithm which has global and a local search components:

1. **Global: Random Pair Sampling** starts by randomly sampling a large number of disk/bulge pairs from $\tilde{X}_d$ and $\tilde{X}_b$. Each pair is fit to $\boldsymbol{y}$ via weighted linear regression as in Equation 16. We typically are able to sample 50,000 pairs, which is an unusually large covering of the parameter space for this particular problem.

2. **Local: Iterative search** starts with the best candidate from phase 1. The bulge-related parameters are held fixed while the bulge component is then paired with all disk prototypes from $\tilde{X}_d$ and a χ^2 is calculated for each combination. The best combination becomes the new start point for another 'step'. Now the disk is fixed while $\tilde{X}_b$ is searched for a better bulge. The process continues in this manner until no improvement results. To evaluate each combination, χ^2 is obtained by weighted linear regression as in Section 6.

Algorithm	Strategy	Speed
GIM2d	Simulated Annealing	~360 sec
Galfit	Levenberg-Marquadt	~30 sec
GMORPH	Instance-Based	~1 sec
1-d approaches	Descent	<1 sec

Table 1: Comparison of the different strategies and speeds of existing algorithms for the galaxy morphology task.

The process is guaranteed to converge because the search space is finite, and the sequence of pairs must always have a decreasing χ^2. We repeat the search using the top 10 or 20 candidates from random sampling as starting points. We have found this type of search to generally converge to a better minimum than simple local (e.g., hillclimbing) search. We conjecture that this is due to the large number of local minima inherent in the problem.

8. RESULTS

Table 1 summarizes the strongest difference between this algorithm and its predecessors, its speed. GMORPH can analyze a 64×64 image in approximately 1 second. The nearest competitor can do the same image in about 30 seconds, but it is a descent method and vulnerable to local minima. The times were obtained by generating random galaxy parameters from the range $F_d \in [0,1], F_b \in [0,1], \mu_x = 0, \mu_y = 0, r_d \in (0,16], \gamma_{inc} \in [0°, 85°], \gamma_d \in [0°, 180°], r_e \in (0,16], \epsilon \in [0, 0.7]$, and $\gamma_b \in [0°, 180°]$, and were used to generate 64×64 images of galaxies. The PSFs were Gaussian with a standard deviation of 2 pixels.

Figure 5 contains the results of a comparison between GMORPH and the traditional and currently most-trusted measure of galaxy shape: human classification. We tested the agreement between GMORPH and an already-classified dataset with 300 galaxies. Each image had been classified visually by a panel of four human experts onto a scale which varies from 0 (all bulge) to 5 (all disk), with 6 being 'irregular'. The results show a clear correlation between GMORPH and expert classification.

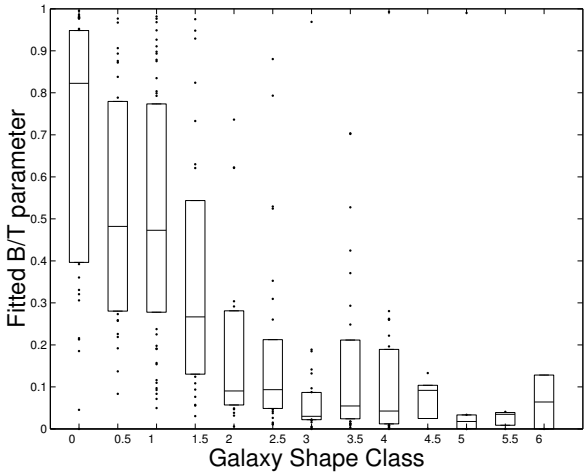

Figure 5: Comparison to human expert classification of 300 galaxies. The horizontal axis is the galaxy classification, which varies from 0 (all bulge) to 5 (all disk), with 6 being 'irregular'. The vertical axis is the bulge-to-total flux ratio returned by GMORPH. Each box indicates the 25th, 50th, and 75th quartiles.

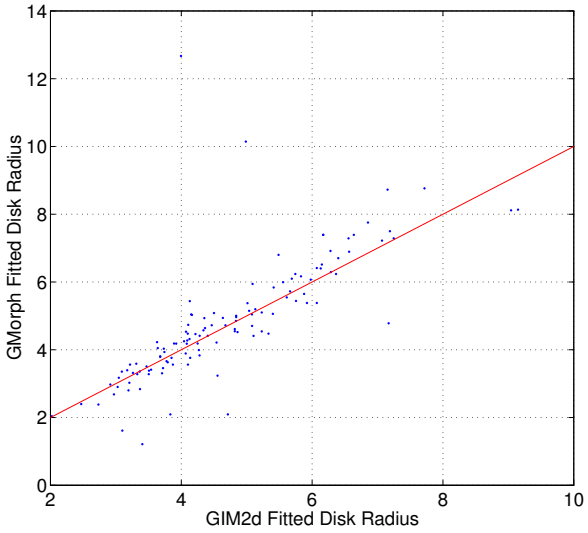

Figure 6: Noise and PSF effect on recovered disk radius error. The agreement between disk radius parameters fitted by GIM2d and those fitted by GMORPH on images from the Sloan Digital Sky Survey.

Figure 6 summarizes the agreement between GMORPH and GIM2d on the disk radius for low-noise, predominantly disk galaxy images from the Sloan Digital Sky Survey. Both algorithms were run on 100 images, each with a unique PSF. The catalog of prototypes used by GMORPH consisted of 1000 disk and 1000 bulge images. The size of the images varied, but were approximately 50×50. The agreement between the two methods is apparent here, however, in high-noise images the two methods produce different results. Although we are still investigating the source of these occasional discrepencies, there is preliminary evidence that these are cases in which either the galaxy morphology diverges from the assumed bulge/disk model, or noise is too severe to reliably fit the data.

9. CONCLUSIONS

We investigate the following nonlinear regression problem from astronomy: given a massive dataset of noisy, distorted images of unknown galaxies, rapidly fit a nonlinear model to each image in the dataset. A instance-based method for accomplishing this task has been described.

Instance-based methods allow for very fast identification of these galaxies through sampling of the parameter space, the use of eigenimages, and decomposing the prototypes into components. GMORPH can avoid the expense of calculating the PSF during the search process, and can scan through the space of galaxy images rapidly because it restricts search to the much smaller subspace determined by PCA. We have measured the performance via simulation and it should in principle allow for unprecedented analysis of astronomical datasets of galaxy images.

10. ADDITIONAL AUTHORS

11. REFERENCES

[1] K. C. Freeman. On the Disks of Spiral and s0 Galaxies. *Astrophysical Journal*, 160:811, 1970.

[2] S. Kirshner, I. Cadez, P. Smyth, and C. Kamath. Learning to classify galaxy shapes using the EM algorithm. In *Advances in Neural Information Processing Systems*, volume 15. Morgan Kaufmann, 2002.

[3] J. Kormendy. Brightness distributions in compact and normal galaxies. III - Decomposition of observed profiles into spheroid and disk components. *Astrophysical Journal*, 217:406–419, 1977.

[4] C. Y. Peng, L. C. Ho, C. D. Impey, and H. Rix. Detailed Structural Decomposition of Galaxy Images. *Astronomical Journal*, 124:266–293, 2002.

[5] K. Ratnatunga, R. Griffiths, and E. Ostrander. Disk And Bulge Morphology Of Wfpc2 Galaxies: The Hst Medium Deep Survey Database. *accepted to Astronomical Journal*, 1999.

[6] Luc Simard, Christopher N. A. Willmer, Nicole P. Vogt, Vicki L. Sarajedini, Andrew C. Phillips, Benjamin J. Weiner, David C. Koo, Myungshin Im, Garth D. Illingworth, and S. M. Faber. The Deep Groth Strip Survey II. Hubble Space Telescope Structural Parameters of Galaxies in the Groth Strip. *Astrophysical Journal Supplements*, 142(1), 2002.

APPENDIX

A. SURFACE BRIGHTNESS FUNCTION

Before being blurred by the PSF, the galaxy is created by the surface brightness function, $\boldsymbol{\psi}$, which takes as an argument a vector from Θ. Here are the 12 model parameters of Θ, a brief description, and their units:

F_b, F_d total integrated flux of bulge and disk components $(erg \cdot cm^2/sec)$

μ_x, μ_y the x and y offset of the galactic center from the center of the image $(pixels)$

r_e, r_d bulge and disk scale lengths $(pixels)$

ϵ_b apparent bulge ellipticity $(unitless)$

γ_{inc} disk inclination $(degrees)$. Rotation toward viewer

γ_b γ_d bulge and disk angle of rotation $(degrees)$. Clockwise rotation relative to viewer

sky sky background offset $(flux/cm^2)$

$Sersic$ a bulge shape parameter that is fixed to the value 4 for all experiments

The classic model of galaxies has been additive: a linear combination of a bulge image, a disk image, and a sky (background) image [6, 4, 5]. The sky image is a constant, and will omitted from the formulae for clarity. The model is

$$\boldsymbol{y}(\boldsymbol{\theta}, \boldsymbol{\pi}) = \boldsymbol{\pi} \star \sum_{i}^{n_c} \boldsymbol{\psi}_i(\boldsymbol{\theta}) \quad (17)$$

$$= \boldsymbol{\pi} \star \left[\boldsymbol{\psi}_{disk}(\boldsymbol{\theta}) + \boldsymbol{\psi}_{bulge}(\boldsymbol{\theta})\right] \quad (18)$$

the components of which we will refer to as the disk image $\boldsymbol{\psi}_{disk}(\boldsymbol{\theta})$, and the bulge image $\boldsymbol{\psi}_{bulge}(\boldsymbol{\theta})$.

The surface brightness, $\boldsymbol{\psi}_{disk}$, of a pure disk galaxy w.r.t. radius has been found to have an exponential form [1, 3]. A commonly used model consists of an infinitely thin disk with brightness in the plane of the disk tapering off exponentially away from the center. When projected onto the image plane, the brightness $\boldsymbol{\psi}_{disk}$ has the form

$$\psi_{disk}(x, y) \propto F_d exp\left(-\frac{\sqrt{x^2 + y^2 \cos^{-2}\gamma_{inc}}}{r_d}\right) \quad (19)$$

where F_d is the integrated brightness of the disk, γ_{inc} is the degree of inclination of the disk towards the viewer, and r_d is the disk "radius", or scale parameter. Both Equation **??** and Equation 20 are simplified for presentation in that they omit clockwise rotation and fix the center of the galaxy at $(0, 0)$.

The bulge is modeled with a classical de Vaucouleurs profile. Also known as the $r^{1/4}$ law, de Vaucouleurs' law is perhaps the most widely used empirical law to describe the surface brightness profile of a pure bulge galaxy. The bulge brightness is

$$\psi_{bulge}(x, y) \propto F_b exp\left(-b\left[\frac{\sqrt{x^2 + y^2(1-\epsilon_b)^{-2}}}{r_b}\right]^{\frac{1}{4}}\right) \quad (20)$$

where F_d is the integrated brightness of the bulge, ϵ_b describes the ellipticity of the bulge, and r_b is the bulge "radius".