



Published in final edited form as:

J Opt Soc Am A Opt Image Sci Vis. 2003 March ; 20(3): 421–429.

Experimental determination of object statistics from noisy images

Matthew A. Kupinski and **Eric Clarkson**

Optical Sciences Center and Department of Radiology, University of Arizona, Tucson, Arizona 85721

John W. Hoppin

Program in Applied Mathematics, University of Arizona, Tucson, Arizona 85721

Liying Chen

Optical Sciences Center, University of Arizona, Tucson, Arizona 85721

Harrison H. Barrett

Optical Sciences Center and Department of Radiology, University of Arizona, Tucson, Arizona 85721

Abstract

Modern imaging systems rely on complicated hardware and sophisticated image-processing methods to produce images. Owing to this complexity in the imaging chain, there are numerous variables in both the hardware and the software that need to be determined. We advocate a task-based approach to measuring and optimizing image quality in which one analyzes the ability of an observer to perform a task. Ideally, a task-based measure of image quality would account for all sources of variation in the imaging system, including object variability. Often, researchers ignore object variability even though it is known to have a large effect on task performance. The more accurate the statistical description of the objects, the more believable the task-based results will be. We have developed methods to fit statistical models of objects, using only noisy image data and a well-characterized imaging system. The results of these techniques could eventually be used to optimize both the hardware and the software components of imaging systems.

1. INTRODUCTION

Modern imaging systems rely on complicated hardware and sophisticated image-processing methods to produce images for evaluation. Owing to the complexity of the imaging chain, there are numerous variables in both the hardware and software that need to be determined. We and other groups advocate a task-based approach to measuring and optimizing image quality, i.e., optimizing imaging systems on the basis of the performance of an observer performing a specific task.^{1–3} This type of analysis is especially important in medical imaging, where systems are inherently task driven, e.g., detection of tumors in images.

Tasks in medical imaging are either detection and classification tasks where the observer is classifying an image into two or more classes, or estimation tasks where the observer is attempting to estimate a particular quantity of interest.^{1,4} In general, the types of observers that researchers can employ are ideal observers,³ human observers, or anthropomorphic observers (human-model observers).^{5,6} The class of ideal observers includes the Bayesian observer,^{3,7} which sets a theoretical upper limit on observer performance, and the Hotelling or ideal linear observer,¹ which sets an upper limit on the performance of an observer who

Address correspondence to Matthew A. Kupinski, Optical Sciences Center, University of Arizona, 1630 East University Boulevard, Tucson, Arizona 85721; e-mail, mkupinski@optics.arizona.edu..

performs only linear manipulations of the image data. Anthropomorphic observers attempt to predict the performance of a human observer reading the images for the specified task.

Numerous images are often required for computation of task-based measures of image quality and are thus infeasible with real patients. Researchers are forced to employ statistical models from which they can produce as many images as required. The stochastic nature of images is due to both detector noise and the randomness in the objects being imaged; e.g., patient anatomy differs greatly from person to person. Much of the current literature in this field has focused on generating *image* models that simulate a type of image produced with a particular imaging system.^{8,9} Image models cannot be used to optimize imaging hardware, because they are specific to one imaging system.

It is desirable to have a model that takes both the object variability and the detector noise into account yet is independent of the imaging system. A statistical description of the objects being imaged is required, though we can observe objects only through a particular imaging system. We have developed methods to fit statistical models of *objects* that use only noisy image data, a well-characterized imaging system, and knowledge of the statistics of the noise. In this paper we will describe these methods and test them on a lumpy object model. In the future, we plan to use the results of these techniques to optimize both the hardware and the software components of imaging systems.

2. BACKGROUND

Imaging can be mathematically represented by

$$\mathbf{g} = H \mathbf{f} + \mathbf{n}, \quad (1)$$

where \mathbf{g} is the $M \times 1$ data vector, \mathbf{f} is the continuous object being imaged, H is an operator representing the method in which an imaging system maps a continuous object to discrete data, and \mathbf{n} is the measurement noise. Vectors are denoted by boldface type. (A note to the reader: Although \mathbf{f} is a function, it still can be thought of as a vector in a Hilbert space, thus the vector notation.) The object \mathbf{f} can represent, for example, the three-dimensional distribution of activity within an organ or the two-dimensional fluence of radiation impinging on a detector system. The vector \mathbf{g} represents the data obtained from an imaging system that may be an image ready for viewing such as in projection x-ray imaging, or it may need processing before viewing, such as sinogram data from tomographic imaging systems.

The continuous-to-discrete operator H is often assumed to be linear and can be measured or modeled for a particular imaging system. We also assume that \mathbf{n} is characterized by a known distribution—typically Poisson for gamma-ray imaging. We observe image data \mathbf{g} , and thus it may be possible to experimentally determine the statistical properties governing \mathbf{g} . However, little is known about the statistical properties of the objects \mathbf{f} being imaged.

Because it is a random vector, the image data vector is governed by a probability density function (PDF), denoted $\text{pr}(\mathbf{g})$. We could equivalently characterize the randomness in image data and objects by using the Fourier transform of the PDF or the characteristic function (CF). It is advantageous to represent the Fourier transform as an expectation over certain complex exponentials, defined for \mathbf{g} as,

$$\Psi_{\mathbf{g}}(\boldsymbol{\rho}) = \left\langle \exp(-2\pi i \mathbf{g}^\dagger \boldsymbol{\rho}) \right\rangle_{\mathbf{g}}. \quad (2)$$

Here $\boldsymbol{\rho}$ is the Fourier conjugate of \mathbf{g} , $\mathbf{g}^\dagger \boldsymbol{\rho}$ is the inner product of \mathbf{g} and $\boldsymbol{\rho}$, and $\langle \cdot \rangle_{\mathbf{g}}$ represents an expectation over $\text{pr}(\mathbf{g})$. The CF of the random process \mathbf{f} has a similar definition,

$$\Psi_{\mathbf{f}}(\xi) = \left\langle \exp(-2\pi i \xi^\dagger \mathbf{f}) \right\rangle_{\mathbf{f}}, \quad (3)$$

where ξ is the infinite-dimensional Fourier conjugate of \mathbf{f} . Because \mathbf{f} is a random function, we will call $\Psi_{\mathbf{f}}(\xi)$ its characteristic *functional* (CFI).

Channels are often used in signal-detection tasks^{5,10} to reduce the dimensionality of the image data vector \mathbf{g} for computational purposes while retaining the salient information about the statistics of the image data. Specifically, Laguerre–Gauss channels have been employed successfully to approximate the ideal linear observer for complicated signal-detection tasks.¹⁰ A channel is a set of filters that are applied to an image to produce a vector of outputs \mathbf{v} , i.e.,

$$\mathbf{v} = T\mathbf{g} = T(H\mathbf{f} + \mathbf{n}), \quad (4)$$

where T is the $N \times M$ linear channel matrix. The dimension (N) of \mathbf{v} is much less than the dimension (M) of \mathbf{g} ; typically N is 6 to 15. Although we are not performing signal-detection tasks in this paper, the use of channels will aid in many of our computations.

3. OBJECT MODELS

In the past, researchers have focused on determining the statistical properties of the images^{8,9} or assumed very simple nonrandom objects such as flat backgrounds. Both approaches essentially ignore the statistics of the objects that generate the images. Object variability, however, is known to have a substantial effect on observer performance and thus should not be ignored.^{5,11} Characterizing the randomness of a continuous object function is difficult. In order to make the problem of determining the statistics of objects tractable, we will employ models of objects. We study two object models: lumpy objects¹¹ and clustered lumpy objects.¹²

A. Lumpy Objects

The first object model that we consider is the lumpy object model developed by Rolland and Barrett.¹¹ The lumpy object model is a continuous mathematical phantom designed to synthesize realistic objects such as those obtained in imaging biological tissue. For example, it is not uncommon to observe relatively few hot spots when one is imaging with nuclear medicine. The lumpy object model may be a good representation of such objects. Lumpy objects are generated by summing a random number of lump functions placed randomly in the field of view (FOV). Typically Gaussians are used as the lump functions, but other lump profiles could be used. Mathematically, a lumpy object is represented as

$$\mathbf{f} = f(\mathbf{r}) = \sum_{n=1}^N \Lambda(\mathbf{r} - \mathbf{c}_n | a, s), \quad (5)$$

where \mathbf{r} is a spatial coordinate in two or three dimensions, $\Lambda(\cdot)$ is the lump function, N is the number of lumps, \mathbf{c}_n is the center of the n th lump function, a is the magnitude of the lump function, and s is the width parameter for the lump function.¹¹ The number of lumps N is a Poisson-distributed random variable with mean \bar{N} , \mathbf{c}_n is uniformly distributed within the FOV, and both a and s are held constant. If we assume that the FOV and the lump function $\Lambda(\cdot)$ are given, then lumpy objects are characterized by the parameters \bar{N} , a , and s , which we will represent by the vector θ . Thus the CFI of \mathbf{f} is conditional upon the parameter θ . Two examples of two-dimensional lumpy functions are shown in Fig. 1. The lumpy object model is fairly simple yet is an important starting point for this work. Note that although the lumps are Gaussian, the statistics governing this object model as a stochastic process are not Gaussian.

The random variables N and $\{\mathbf{c}_n\}$ fully encompass the randomness of lumpy objects. Furthermore, as shown by Barrett and Myers,¹³ we can calculate the CFI for lumpy objects conditioned on the parameters characterizing the lumpy object model θ by substituting the definition of a lumpy object [Eq. (5)] into the CFI defined in Eq. (3):

$$\begin{aligned} \Psi_{\mathbf{f}}(\xi | \theta) &= \left\langle \exp(-2\pi i \xi^\dagger \mathbf{f}) \right\rangle_{\mathbf{f} | \theta} \end{aligned} \quad (6)$$

$$= \left\langle \exp \left[-2\pi i \int \xi(\mathbf{r}) \sum_{n=1}^N \Lambda(\mathbf{r} - \mathbf{c}_n | a, s) d\mathbf{r} \right] \right\rangle_{\{\mathbf{c}_n\}, N} \quad (7)$$

Here we have written out the L_2 inner product $\xi^\dagger \mathbf{f}$ explicitly as $\int \xi(\mathbf{r}) f(\mathbf{r}) d\mathbf{r}$ for a lumpy object. The expectation is taken with respect to $\{\mathbf{c}_n\}$ and N , the random variables that determine a lumpy object. The centers of the lumps, $\{\mathbf{c}_n\}$, are uniformly distributed, and the number of lumps N is Poisson distributed. Simplifying the above expression, we have

$$\begin{aligned} \Psi_{\mathbf{f}}(\xi | \theta) &= \left\langle \prod_{n=1}^N \left\langle \exp \left[-2\pi i \int \xi(\mathbf{r}) \Lambda(\mathbf{r} - \mathbf{c}_n | a, s) d\mathbf{r} \right] \right\rangle_{\mathbf{c}_n} \right\rangle_N \end{aligned} \quad (8)$$

$$= \left\langle \Psi_{\Lambda}(\xi | \theta)^N \right\rangle_N \quad (9)$$

$$= \exp(-N) \sum_{N=0}^{\infty} \frac{N^N}{N!} \Psi_{\Lambda}(\xi | \theta)^N \quad (10)$$

$$= \exp \left\{ -N [1 - \Psi_{\Lambda}(\xi | \theta)] \right\}, \quad (11)$$

where N is the mean of N and $\Psi_{\Lambda}(\xi | \theta)$ is the characteristic functional of one randomly located lump given by

$$\Psi_{\Lambda}(\xi | \theta) = \left\langle \exp \left[-2\pi i \int \xi(\mathbf{r}) \Lambda(\mathbf{r} - \mathbf{c} | a, s) d\mathbf{r} \right] \right\rangle_{\mathbf{c}}. \quad (12)$$

The computation of this function will be discussed in Section 6.

B. Clustered Lumpy Objects

In 1999, Bochud *et al.*¹² extended the lumpy object model to synthesize more complicated and realistic objects. Their model, known as clustered lumpy objects, clusters the lumps around certain cluster centers that are uniformly distributed within the FOV. This is represented mathematically for two-dimensional \mathbf{r} as

$$\mathbf{f} = f(\mathbf{r}) = \sum_{n=1}^N \sum_{k=1}^{K_n} \Lambda([R_{\varphi_n}] (\mathbf{r} - \mathbf{c}_n - \Delta_{nk}) | a), \quad (13)$$

where N is the number of clusters, K_n is the number of lump functions in the n th cluster, \mathbf{c}_n is the center of the n th cluster, Δ_{nk} is the center of the k th lump function in the n th cluster, φ_n is the orientation of the lumps in the n th cluster, and a are the parameters characterizing the shape of the lumps. The rotation of the lumps is represented by the rotation matrix R_{φ_n} . Both N and K_n are Poisson distributed with means N and K , respectively. The center of a cluster \mathbf{c}_n is uniformly distributed over the FOV, and the lumps within a cluster are Gaussian distributed

around \mathbf{c}_n . That is, Δ_{nk} is sampled from a radially symmetric Gaussian distribution with width σ_Δ . The angle φ_n is, in general, uniformly distributed between 0 and 2π but may be distributed differently if desired. Again, assuming that the lump function $\Lambda(\cdot)$ is known, the parameters characterizing a clustered lumpy object are \mathcal{N} , \mathcal{K} , σ_Δ , and α , which we combine to form the vector of parameters θ .

The function $\Lambda(\cdot)$ need not be a Gaussian function. Bochud used a two-dimensional asymmetric lump¹² given in polar coordinates as

$$\begin{aligned} \Lambda(r, \varphi | \gamma, \beta, l_x, l_y) \\ = \exp \left\{ -\gamma r^\beta \left[\frac{l_y^2 \cos^2(\varphi) + l_x^2 \sin^2(\varphi)}{l_x^2 l_y^2} \right]^{1/2} \right\}. \end{aligned} \quad (14)$$

This type of lump function was chosen because the resultant images approximated the texture of mammograms. Using this lump function results in seven parameters that characterize this object model, i.e., $\theta = \{ \mathcal{N}, \mathcal{K}, \sigma_\Delta, \gamma, \beta, l_x, l_y \}$. Examples of clustered lumpy objects are shown in Fig. 2.

To calculate the CFI for the clustered lumpy object model, we rewrite the definition of a clustered lumpy object [Eq. (13)] as

$$\mathbf{f} = f(\mathbf{r}) = \sum_{n=1}^{\mathcal{N}} \Omega(\mathbf{r} - \mathbf{c}_n, \{ \Delta_{nk} \}, \varphi_n, K_n | \mathbf{a}), \quad (15)$$

where

$$\begin{aligned} \Omega(\mathbf{r} - \mathbf{c}_n, \{ \Delta_{nk} \}, \varphi_n, K_n | \mathbf{a}) \\ = \sum_{k=1}^{K_n} \Lambda([R_{\varphi_n}] (\mathbf{r} - \mathbf{c}_n - \Delta_{nk}) | \mathbf{a}), \end{aligned} \quad (16)$$

and $\{ \Delta_{nk} \}$ is a list of K_n lump centers relative to the cluster center \mathbf{c}_n . Equation (15) is similar to a lumpy object model with a more complicated lump function. Thus we will calculate the CFI of the clustered lumpy model in similar fashion. Beginning with

$$\Psi_{\mathbf{f}}(\xi | \theta) = \exp \left\{ -\mathcal{N} [1 - \Psi_{\Omega}(\xi | \theta)] \right\}, \quad (17)$$

all that is left to compute is $\Psi_{\Omega}(\cdot)$. The CFI for Ω can be expressed as

$$\Psi_{\Omega}(\xi | \theta) = \left\langle \exp \left\{ -\mathcal{K} [1 - \Psi_{\Lambda}(\xi | \mathbf{c}, \varphi, \theta)] \right\} \right\rangle_{\mathbf{c}, \varphi}, \quad (18)$$

where $\Psi_{\Lambda}(\xi | \mathbf{c}, \varphi, \theta)$ is the characteristic functional of a single randomly located lump with cluster center \mathbf{c} and angle φ . The form of $\Psi_{\Lambda}(\cdot)$ is similar to Eq. (12) except that we now take the expectation over Δ instead of \mathbf{c} . We are now able to write the final expression for the CFI of a clustered lumpy object model as

$$\begin{aligned} \Psi_{\mathbf{f}}(\xi | \theta) = \exp \left[-\mathcal{N} (1 - \left\langle \exp \left\{ -\mathcal{K} [1 \right. \right. \right. \\ \left. \left. \left. - \Psi_{\Lambda}(\xi | \mathbf{c}, \varphi, \theta) \right\} \right\rangle_{\mathbf{c}, \varphi}) \right]. \end{aligned} \quad (19)$$

We will discuss the computation of $\Psi_{\mathbf{f}}(\xi | \theta)$ in Section 6.

4. TRANSFORMATION THROUGH IMAGING SYSTEMS

Both lumpy and clustered lumpy objects are characterized by the parameters θ . Our goal is to determine the parameters θ of the object model that best fit a set of image data $\{\mathbf{g}_l\}$. The difficulty of this problem lies in that we can only observe \mathbf{g}_l , and yet we wish to determine parameters that characterize the statistics of the objects \mathbf{f} . We are further impeded because we do not know the PDF of \mathbf{g} even though we have a model for \mathbf{f} and a known imaging operator H . Maximum-likelihood estimation is thus not an immediate possibility. A different estimation approach is required that makes use of the knowledge we do have, namely, the CFI of the objects.

A complete description of the transformation of an object CFI through an imaging system is described by Clarkson *et al.*¹⁴ We summarize those results in Subsection 4.A.

A. Noiseless Imaging System

Let us assume that we know the CFI of the object distribution $\Psi_{\mathbf{f}}(\xi | \theta)$. Let us, for now, envision a noiseless imaging system

$$\bar{\mathbf{g}} = H \mathbf{f} \quad (20)$$

and analyze the characteristic function of $\bar{\mathbf{g}}$. (The over-bar notation is used because $\bar{\mathbf{g}}$ can be thought of as the noise-averaged image for a particular object \mathbf{f} .) The characteristic function of $\bar{\mathbf{g}}$ is

$$\Psi_{\bar{\mathbf{g}}}(\boldsymbol{\rho} | \theta) = \left\langle \exp(-2\pi i \boldsymbol{\rho}^\dagger \bar{\mathbf{g}}) \right\rangle_{\bar{\mathbf{g}} | \theta} \quad (21)$$

$$= \int p(\bar{\mathbf{g}} | \theta) \prod_{m=1}^M \exp(-2\pi i \rho_m \bar{g}_m) d\bar{\mathbf{g}}, \quad (22)$$

which, by use of our noiseless imaging system [Eq. (20)] and using properties of the adjoint, can be rewritten as^{13,14}

$$\Psi_{\bar{\mathbf{g}}}(\boldsymbol{\rho} | \theta) = \left\langle \exp(-2\pi i (H^\dagger \boldsymbol{\rho})^\dagger \mathbf{f}) \right\rangle_{\mathbf{f} | \theta} \quad (23)$$

$$= \Psi_{\mathbf{f}}(H^\dagger \boldsymbol{\rho} | \theta). \quad (24)$$

Thus, if the CFI of \mathbf{f} is known, then we also know the CF of any linear mapping of \mathbf{f} by simply using the adjoint of the linear operator. Although we do not know the PDF of $\bar{\mathbf{g}}$, we do know $\Psi_{\bar{\mathbf{g}}}(\boldsymbol{\rho} | \theta)$ for either of our object models.

B. Noise

Thus far, we have dealt with noise-free imaging systems that are of little practical value. In order for the methods developed in this paper to be of practical interest, we must be able to compute the characteristic function of the image data $\mathbf{g} = \bar{\mathbf{g}} + \mathbf{n}$ accounting for both object variability and noise. Two common noise models that researchers employ are Gaussian noise and quantum or Poisson noise.

The characteristic function for Gaussian noise with covariance K is known to be Gaussian as well, with the form

$$\Psi_{\mathbf{n}}(\boldsymbol{\rho} | K) = \exp(-2\pi^2 \boldsymbol{\rho}^\dagger K \boldsymbol{\rho}). \quad (25)$$

Because the noise is independent of the object being imaged, we know that the PDF of \mathbf{g} is a convolution of the PDF of $\bar{\mathbf{g}}$ and the PDF of \mathbf{n} , which, using the Fourier-convolution theorem, yields

$$\Psi_{\mathbf{g}}(\boldsymbol{\rho} | \boldsymbol{\theta}, K) = \Psi_{\mathbf{f}}(H^\dagger \boldsymbol{\rho} | \boldsymbol{\theta}) \Psi_{\mathbf{n}}(\boldsymbol{\rho} | K). \quad (26)$$

Poisson detector noise can also be incorporated into this framework. Poisson noise is conditional upon the mean image, i.e.,

$$\text{pr}(\mathbf{g} | \bar{\mathbf{g}}) = \prod_{m=1}^M \exp(-\bar{g}_m) \frac{\bar{g}_m^{g_m}}{g_m!}, \quad (27)$$

where \bar{g}_m denotes the m th component of the M vector $\bar{\mathbf{g}}$. The PDF of \mathbf{g} can then be obtained by marginalizing over the object $\bar{\mathbf{g}}$,

$$\text{pr}(\mathbf{g} | \boldsymbol{\theta}) = \int \text{pr}(\mathbf{g} | \bar{\mathbf{g}}) \text{pr}(\bar{\mathbf{g}} | \boldsymbol{\theta}) d\bar{\mathbf{g}} \quad (28)$$

$$= \int \text{pr}(\bar{\mathbf{g}} | \boldsymbol{\theta}) \prod_{m=1}^M \exp(-\bar{g}_m) \frac{\bar{g}_m^{g_m}}{g_m!} d\bar{\mathbf{g}}. \quad (29)$$

By taking the expectation of the complex exponentials over the PDF given above, we arrive (after some manipulations described by Clarkson *et al.*¹⁴) at

$$\begin{aligned} \Psi_{\mathbf{g}}(\boldsymbol{\rho} | \boldsymbol{\theta}) \\ = \int \text{pr}(\bar{\mathbf{g}} | \boldsymbol{\theta}) \prod_{m=1}^M \exp[-\bar{g}_m + \bar{g}_m \exp(-2\pi i \rho_m)] d\bar{\mathbf{g}}, \end{aligned} \quad (30)$$

which is close to the CF of $\bar{\mathbf{g}}$ [Eq. (22)] except that the term in the exponent is not the same. We can relate the above expression to the CF of $\bar{\mathbf{g}}$ by defining a nonlinear operator $\Gamma(\cdot)$ that maps an M vector to another M vector, by using the following equation for each component m ,

$$[\Gamma(\boldsymbol{\rho})]_m = \frac{-1 + \exp(-2\pi i \rho_m)}{-2\pi i}. \quad (31)$$

Thus we can relate the CF of \mathbf{g} to that of the noiseless CF of $\bar{\mathbf{g}}$ that we previously related to the CFI of \mathbf{f} , i.e.,

$$\Psi_{\mathbf{g}}(\boldsymbol{\rho} | \boldsymbol{\theta}) = \Psi_{\bar{\mathbf{g}}}(\Gamma(\boldsymbol{\rho}) | \boldsymbol{\theta}) = \Psi_{\mathbf{f}}(H^\dagger \Gamma(\boldsymbol{\rho}) | \boldsymbol{\theta}). \quad (32)$$

In other words, because we know the CFI for our object models, we are able to use H^\dagger and a known nonlinear operator to determine the CF for our noisy image data.

C. Channels

The channel operator, like the imaging system itself, is a linear operator, and thus if one wants to know the CF for the channel outputs $\mathbf{v} = T\mathbf{g} = T(H\mathbf{f} + \mathbf{n})$, one needs only the adjoint T^\dagger . That is,

$$\Psi_{\mathbf{v}}(\boldsymbol{\omega} | \boldsymbol{\theta}) = \Psi_{\mathbf{f}}(H^{\dagger} \Gamma(T^{\dagger} \boldsymbol{\omega}) | \boldsymbol{\theta}), \quad (33)$$

where $\boldsymbol{\omega}$ is the Fourier conjugate of the channel outputs \mathbf{v} .

5. ESTIMATION

One can determine exactly the CF of channel outputs $\Psi_{\mathbf{v}}(\boldsymbol{\omega} | \boldsymbol{\theta})$ by using either of the two object models discussed. Given a database of L images $\{\mathbf{g}_l\}$, we would like to estimate the parameters $\boldsymbol{\theta}$ that characterize the object model that best fits the data, using channel outputs from these images $\mathbf{v}_l = T\mathbf{g}_l$.

A standard maximum-likelihood or Bayesian approach to solving this problem is not computationally feasible. The likelihood of $\boldsymbol{\theta}$ for an ensemble of images $\{\mathbf{g}_l\}$ is given by

$$\text{pr}(\{\mathbf{g}_l\} | \boldsymbol{\theta}) = \prod_{l=1}^L \int d\mathbf{f} \text{pr}(\mathbf{g}_l | \mathbf{f}) \text{pr}(\mathbf{f} | \boldsymbol{\theta}). \quad (34)$$

We would have to perform L high-dimensional integrals for a given $\boldsymbol{\theta}$. To optimize $\boldsymbol{\theta}$, we would have to compute these L integrals repeatedly as part of an iterative search. We present an estimation technique that relies on characteristic-function analysis to simplify the estimation of $\boldsymbol{\theta}$.

A. Empirical Characteristic Function

The characteristic function of a random variable or vector is an expectation value of the complex exponentials, i.e.,

$$\Psi_{\mathbf{v}}(\boldsymbol{\omega}) = \left\langle \exp(-2\pi i \boldsymbol{\omega}^{\dagger} \mathbf{v}) \right\rangle_{\mathbf{v}}. \quad (35)$$

Hence, given independent data sampled from $\text{pr}(\mathbf{v})$ and denoted by \mathbf{v}_l with $l = 1 \dots L$, the characteristic function of \mathbf{v} can be approximated by the sample average of the complex exponentials. That is, the *empirical* characteristic function is given by

$$\hat{\Psi}_{\mathbf{v}}(\boldsymbol{\omega}) = \frac{1}{L} \sum_{l=1}^L \exp(-2\pi i \boldsymbol{\omega}^{\dagger} \mathbf{v}_l). \quad (36)$$

The empirical characteristic function is an unbiased estimate of the true characteristic function. We can further derive an expression for the covariance $C(\boldsymbol{\omega}, \boldsymbol{\omega}')$ of this estimate:

$$\begin{aligned} C(\boldsymbol{\omega}, \boldsymbol{\omega}') &= \left\langle \hat{\Psi}_{\mathbf{v}}(\boldsymbol{\omega}) \hat{\Psi}_{\mathbf{v}}^*(\boldsymbol{\omega}') \right\rangle_{\{\mathbf{v}_l\}} - \Psi_{\mathbf{v}}(\boldsymbol{\omega}) \Psi_{\mathbf{v}}^*(\boldsymbol{\omega}') \\ &= \frac{1}{L^2} \sum_{l=1}^L \sum_{l'=1}^L \left\langle \exp(-2\pi i \boldsymbol{\omega}^{\dagger} \mathbf{v}_l) \right. \\ &\quad \times \left. \exp(2\pi i \boldsymbol{\omega}'^{\dagger} \mathbf{v}_{l'}) \right\rangle_{\mathbf{v}_l, \mathbf{v}_{l'}} \\ &\quad - \Psi_{\mathbf{v}}(\boldsymbol{\omega}) \Psi_{\mathbf{v}}^*(\boldsymbol{\omega}'). \end{aligned} \quad (37)$$

In the above double summation, there are L terms where $l = l'$ in which the inner expectation values become

$$\left\langle \exp[-2\pi i(\boldsymbol{\omega} - \boldsymbol{\omega}')^\dagger \mathbf{v}_l] \right\rangle_{\mathbf{v}_l} = \Psi_{\mathbf{v}}(\boldsymbol{\omega} - \boldsymbol{\omega}') \quad (39)$$

and $L^2 - L$ terms where $l \neq l'$ in which the inner expectation values become

$$\left\langle \exp(-2\pi i \boldsymbol{\omega}^\dagger \mathbf{v}_l) \right\rangle_{\mathbf{v}_l} \left\langle \exp(2\pi i \boldsymbol{\omega}'^\dagger \mathbf{v}_{l'}) \right\rangle_{\mathbf{v}_{l'}} = \Psi_{\mathbf{v}}(\boldsymbol{\omega}) \Psi_{\mathbf{v}}^*(\boldsymbol{\omega}'). \quad (40)$$

Equation (40) makes use of the independence of the two distinct samples \mathbf{v}_l and $\mathbf{v}_{l'}$. Thus our final expression for the covariance is

$$C(\boldsymbol{\omega}, \boldsymbol{\omega}') = \frac{1}{L} [\Psi_{\mathbf{v}}(\boldsymbol{\omega} - \boldsymbol{\omega}') - \Psi_{\mathbf{v}}(\boldsymbol{\omega}) \Psi_{\mathbf{v}}^*(\boldsymbol{\omega}')]. \quad (41)$$

By setting $\boldsymbol{\omega} = \boldsymbol{\omega}'$ and noting that $\Psi(0) = 1$ by definition, we arrive at an expression for the variance $V(\boldsymbol{\omega})$ of the empirical estimate of the characteristic function,

$$V(\boldsymbol{\omega}) = \frac{1}{L} [1 - |\Psi_{\mathbf{v}}(\boldsymbol{\omega})|^2]. \quad (42)$$

Note that the variance of the empirical characteristic function goes to 0 as $\boldsymbol{\omega}$ approaches $\mathbf{0}$, so for most smoothly varying functions, the variance of the empirical characteristic function is small near $\boldsymbol{\omega} = \mathbf{0}$.

B. Cost Functions

We have yet to discuss potential cost function that could be employed to determine the object model parameters $\boldsymbol{\theta}$ that best fit the available data. One obvious choice would be to apply weighted least-squares fitting of the characteristic functions. We are further aided by our knowledge of the variance of the empirical characteristic function as given in Eq. (42). The weighted least-squares solution is one that minimizes the weighted mean squared error,

$$W_{\text{mse}}(\boldsymbol{\theta}) = \int \frac{|\hat{\Psi}_{\mathbf{v}}(\boldsymbol{\omega}) - \Psi_{\mathbf{v}}(\boldsymbol{\omega} | \boldsymbol{\theta})|^2}{\frac{1}{L} [1 - |\Psi_{\mathbf{v}}(\boldsymbol{\omega})|^2]} d\boldsymbol{\omega}, \quad (43)$$

where the denominator is an estimate of the variance of the empirical characteristic function [Eq. (42)]. Dividing by the variance of the empirical characteristic function has the effect of weighting regions with small variance more than those with large variance. We actually have a choice in Eq. (43) of using either $\Psi_{\mathbf{v}}(\boldsymbol{\omega} | \boldsymbol{\theta})$ or $\hat{\Psi}_{\mathbf{v}}(\boldsymbol{\omega})$ in the denominator for the variance expression.

The aforementioned cost function involves an integral over $\boldsymbol{\omega}$. Another approach to generating a cost function would be to approximate the inverse Fourier transform of $\Psi_{\mathbf{v}}(\boldsymbol{\omega} | \boldsymbol{\theta})$ to obtain $\hat{\text{pr}}(\mathbf{v} | \boldsymbol{\theta})$ and then use maximum-likelihood estimation to choose the $\boldsymbol{\theta}$ that maximizes

$$\lambda(\boldsymbol{\theta} | \{\mathbf{v}^{(l)}\}) = \sum_{l=1}^L \log [\hat{\text{pr}}(\mathbf{v}^{(l)} | \boldsymbol{\theta})]. \quad (44)$$

This also involves an integral over $\boldsymbol{\omega}$, but this integral turns out to be difficult to approximate.

6. COMPUTATIONAL ISSUES

Each of the cost functions [Eqs. (43) and (44)] requires an integral over $\boldsymbol{\omega}$ that we calculate with Monte Carlo techniques. Monte Carlo techniques require a PDF in the integrand,

something our cost functions lack. Importance sampling^{15,16} provides the means to avert this lack of a PDF by multiplying and dividing by a user-defined PDF. It is advantageous to choose a PDF that would sample points around the *important* region of the integrand, i.e., nonzero values of the integrand. For example, Eq. (43) could be approximated by

$$W_{\text{mse}}(\boldsymbol{\theta}) = \int \frac{|\Psi_{\mathbf{v}}(\boldsymbol{\omega} | \boldsymbol{\theta}) - \hat{\Psi}_{\mathbf{v}}(\boldsymbol{\omega})|^2}{\frac{1}{L} [1 - |\Psi_{\mathbf{v}}(\boldsymbol{\omega} | \boldsymbol{\theta})|^2]} d\boldsymbol{\omega} \quad (45)$$

$$= \int \frac{|\Psi_{\mathbf{v}}(\boldsymbol{\omega} | \boldsymbol{\theta}) - \hat{\Psi}_{\mathbf{v}}(\boldsymbol{\omega})|^2}{\frac{1}{L} [1 - |\Psi_{\mathbf{v}}(\boldsymbol{\omega} | \boldsymbol{\theta})|^2]} \frac{\text{pr}(\boldsymbol{\omega})}{\text{pr}(\boldsymbol{\omega})} d\boldsymbol{\omega} \quad (46)$$

$$\approx \frac{1}{J} \sum_{j=1}^J \frac{|\Psi_{\mathbf{v}}(\boldsymbol{\omega}_j | \boldsymbol{\theta}) - \hat{\Psi}_{\mathbf{v}}(\boldsymbol{\omega}_j)|^2}{\frac{1}{L} [1 - |\Psi_{\mathbf{v}}(\boldsymbol{\omega}_j | \boldsymbol{\theta})|^2]} \frac{1}{\text{pr}(\boldsymbol{\omega}_j)}, \quad (47)$$

where $\boldsymbol{\omega}_j$ is sampled from $\text{pr}(\boldsymbol{\omega})$.

The accuracy of the expression in Eq. (47) depends on J and the importance sampler $\text{pr}(\boldsymbol{\omega}_j)$. Therefore a good choice of $\text{pr}(\boldsymbol{\omega}_j)$ is needed. If we were to assume that the \mathbf{v}_l were Gaussian distributed with covariance matrix Σ , then $|\Psi_{\mathbf{v}}(\boldsymbol{\omega})|$ would be an unnormalized Gaussian with covariance matrix Σ^{-1} . Furthermore, the maximum of a characteristic function is 1 and occurs at $\mathbf{v} = 0$. Thus a reasonable choice for an importance sampler would be a Gaussian with zero mean and covariance Σ^{-1} . We measure the sample covariance $\hat{\Sigma}$ for the channel outputs \mathbf{v}_l and use $\hat{\Sigma}^{-1}$ as the covariance of our Gaussian importance sampler. The \mathbf{v}_l are not Gaussian distributed, but this method provides a successful Gaussian sampler; i.e., the $\boldsymbol{\omega}_j$'s sampled lie near the peak of the integrand.

The term $\hat{\Psi}(\boldsymbol{\omega})$ in Eq. (47) is a straightforward computation. However, we have yet to describe how one calculates $\Psi_{\mathbf{v}}(\boldsymbol{\omega}_j | \boldsymbol{\theta})$. Specifically, the term $\Psi_{\Lambda}(\cdot)$ shown in Eqs. (11) and (18) is computationally burdensome. For the lumpy object model, the characteristic functional of a single randomly located lump is given by

$$\Psi_{\Lambda}(\boldsymbol{\xi} | \boldsymbol{\theta}) = \int p(\mathbf{c}) \exp\left(-2\pi i \int d\mathbf{r} \boldsymbol{\xi}(\mathbf{r}) \Lambda(\mathbf{r} - \mathbf{c})\right) d\mathbf{c}, \quad (48)$$

where $p(\mathbf{c})$ is the PDF of the lump centers, which, as we described earlier, are uniformly distributed within the FOV for lumpy objects. Now, replacing $\boldsymbol{\xi}(\mathbf{r})$ with $H_{\dagger} \Gamma(T^{\dagger} \boldsymbol{\omega})$, we arrive at

$$\Psi_{\Lambda}(\boldsymbol{\omega} | \boldsymbol{\theta}) = \int p(\mathbf{c}) \exp\left(-2\pi i \int \sum_{m=1}^M h_m(\mathbf{r}) \Gamma(T^{\dagger} \boldsymbol{\omega})_m \times \Lambda(\mathbf{r} - \mathbf{c}) d\mathbf{r}\right) d\mathbf{c} \quad (49)$$

$$= \int p(\mathbf{c}) \exp\left(-2\pi i \sum_{m=1}^M \Gamma(T^{\dagger} \boldsymbol{\omega})_m \times \int h_m(\mathbf{r}) \Lambda(\mathbf{r} - \mathbf{c}) d\mathbf{r}\right) d\mathbf{c}. \quad (50)$$

This equation makes use of the adjoint of a continuous-to-discrete operator, i.e.,

$$H^\dagger \mathbf{x} = \sum_{m=1}^M h_m(\mathbf{r}) x_m. \quad (51)$$

Also note that the inner integral in Eq. (50) is equivalent to transferring a lump centered at position \mathbf{c} through the imaging system H and taking the inner product with the M vector $\Gamma(T^\dagger \omega)$. We are aided in this computation because the M vector $\Gamma(T^\dagger \omega_j)$ is fixed for a given importance sample ω_j . Thus, once we have our importance samples ω_j , we can precompute and store every element of this vector. In practice, we store this information in pieces because of computer memory limitations. We are still left with computing the outer integral over \mathbf{c} . We again resort to Monte Carlo techniques to approximate this integral by sampling \mathbf{c}_n uniformly in the FOV and averaging the integrand at each sampled point.

Similar techniques are applied to compute $\Psi_\Lambda(\cdot)$ for the clustered lumpy object model. The key difference is the increase in the number of random variables to integrate over: the angle of the lump, the cluster center, and the lump center. Again, Monte Carlo techniques can readily be employed for this computation because the densities are known for each of these parameters.

7. SIMULATIONS

The goal of our work is to determine parameters that characterize an object model by using only noisy images of the objects. To validate the methods presented in Section 3, we performed simulation studies using objects that match the model employed. We ran our simulations in parallel on five Pentium-based computers running Linux. The computers communicated by using the Parallel Virtual Machine (PVM) package. The optimization was performed with APPSPACK, a freely available optimization routine from Sandia National Laboratories.

We began by generating 100 64×64 images \mathbf{g}_i , using the lumpy object model and a simulated imaging system. We then calculated the channel outputs \mathbf{v}_i by using a channel operator T . We calculated $\Psi_{\mathbf{v}}(\omega | \theta)$ from our knowledge of $\Psi_{\mathbf{f}}(\xi | \theta)$ for the lumpy object model and our knowledge of H , T , and the noise. We varied the parameters θ to match the data \mathbf{v}_i by using the cost function shown in Eq. (43). Note that this cost function requires the empirical characteristic function determined from the data $\hat{\Psi}_{\mathbf{v}}(\omega)$ [Eq. (36)].

The imaging system that we simulated was an idealized parallel-hole collimator system represented by

$$\bar{g}_m = \gamma \int h_m(\mathbf{r}) f(\mathbf{r}) d\mathbf{r}, \quad (52)$$

where $h_m(\cdot)$ is a Gaussian blur function centered at the m th pixel with a standard deviation of 1.5 pixels and γ is the amplitude of the system, which we set to 40. The images were generated by sampling a Poisson distribution for the m th pixel with parameter \bar{g}_m . For the lumpy object model with Gaussian lumps, there exists a closed-form expression for the integral in the imaging equation [Eq. (52)]. Example images used in the simulation are shown in Figs. 3(a)–3(c).

The filter T that we used consisted of the first three Laguerre–Gauss functions.¹⁰ Laguerre–Gauss functions are circularly symmetric products of a Gaussian with Laguerre polynomials centered within the FOV. Because the lumps that we used in the lumpy object model are circularly symmetric and the statistical distribution of their positions is circularly symmetric and stationary (ignoring boundaries), the Laguerre–Gauss functions are a reasonable choice of channels to extract the salient statistical information about the object model. As we will show, we are able to get good estimates of our three object model parameters by using just three

channels. It should be noted that computing the cost function becomes more and more computationally burdensome as the number of channels increases.

Note that we have run these simulations with the data-matching model in an attempt to verify our method. In this situation, we therefore have a great deal of prior knowledge about our objects that we exploit to limit the number of channels used. A more realistic situation is one in which the model does not perfectly match the data. Of course, it is difficult to verify that the method works properly if the true parameters θ are not known.

The parameters θ used to generate the 100 images \mathbf{g}_i were $\mathcal{N} = 30$, $a = 1$, and $s = 5$. The method returned an estimate of θ with parameter $\mathcal{N} = 26$, $a = 1.06$, and $s = 5.23$, which are close to the true values. Using these fitted parameters, we then regenerated random images, using our simulated imaging system [shown in Figs. 3(d)–3(f)] that are statistically similar to the original images shown in Figs. 3(a)–3(c). Thus the method was able to determine the parameters that characterize the stochastic object model by using only channel outputs from 100 noisy and blurry images.

We reperformed this experiment ten times with different sets of images (100 each time) to obtain the average estimate of θ as well as the standard deviations of our estimates. The results of these ten runs in terms of the mean \pm the standard deviation were $\mathcal{N} = 32.0 \pm 6.2$, $a = 1.16 \pm 0.28$, $s = 4.67 \pm 0.77$. Thus the method performs consistently over different sets of images.

8. DISCUSSION AND CONCLUSIONS

We have presented a method for fitting statistical object models to real image data produced by a well-characterized imaging system. With a statistical object model, one can readily simulate imaging systems and produce image data to be used in computing task performance. The method that we have presented addresses directly the variability in the objects being imaged and accounts for any blur or any degradation that may occur during imaging as well as the noise in the image data produced.

We presented and discussed our technique as it relates to two specific object models: the lumpy object model and the clustered lumpy object model. Our method, however, is general. All it requires is that the characteristic function of the object be computable. We showed that this condition is met for the lumpy and clustered lumpy object models. There are other object models based on filter-bank or wavelet analysis that might also be applicable to this technique.

The simulation study performed in this paper was a consistency test. That is, given that the model matches the data, does the method return parameters that are close to the true parameters? If one were to apply this technique to real images, then it would be impossible to say what the correct parameters were. The method for applying this technique to real images is exactly as presented in this paper. We will address the issues involved with matching statistical object models to real images in a future publication.

To ease computational complexity, we employed channel operators to reduce the dimensionality of the image data. Channels have been successfully employed in many signal-detection^{5,17} and texture-analysis methods.⁸ However, we have not addressed the issue of the ideal or best filters to use for our method. This issue became particularly important when we applied our method to the clustered lumpy object model, because the lump functions for that model are asymmetric. It was difficult to determine the lump parameters by using symmetric Laguerre–Gauss channels. In particular, the parameters l_x and l_y that characterize the asymmetry in the lumps were never properly estimated with the Laguerre–Gauss channels. Channel selection is also an issue with sinogram image data, in which locally supported filters are not such an obvious choice.

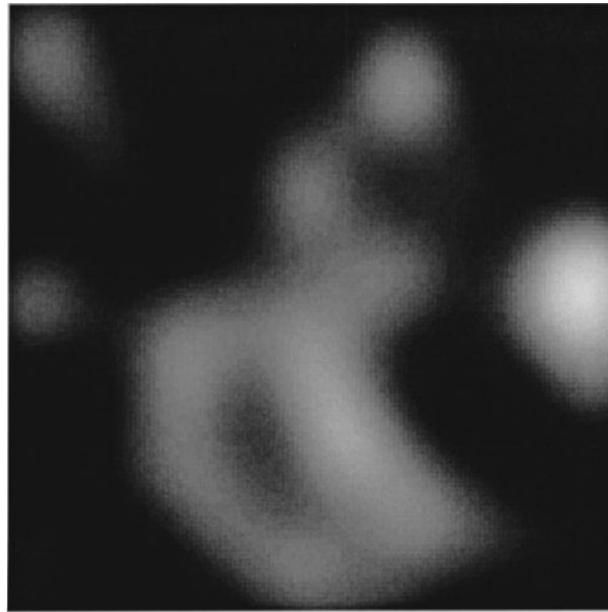
We would like to compare different object models by measuring the performances of the corresponding ideal observers on real images of the real objects. The eventual goal of studying this technique is to perform hardware optimizations by using task-based measures of image quality. Specifically, we would like to compute ideal-observer performance for an imaging system by using one of our fitted object models. These topics will be addressed in future publications.

Acknowledgements

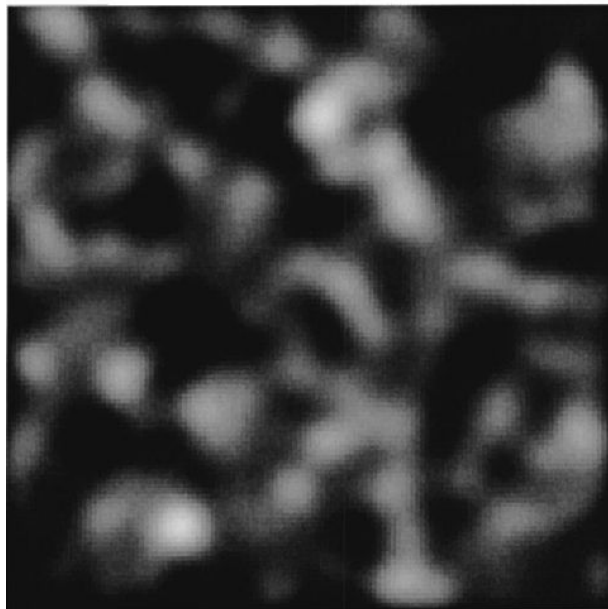
This work was supported by National Science Foundation grant 9977116 and National Institutes of Health grants P41 RR14304, KO1 CA87017, and RO1 CA52643.

References

1. Barrett HH. "Objective assessment of image quality: effects of quantum noise and object variability," *J Opt Soc Am A* 1990;7:1266–1278. [PubMed: 2370589]
2. Barrett HH, Denny JL, Wagner RF, Myers KJ. "Objective assessment of image quality. II. Fisher information, Fourier crosstalk, and figures of merit for task performance," *J Opt Soc Am A* 1995;12:834–852.
3. Barrett HH, Abbey CK, Clarkson E. "Objective assessment of image quality. III. ROC metrics, ideal observers, and likelihood-generating functions," *J Opt Soc Am A* 1998;15:1520–1535.
4. Liu Z, Knill DC, Kersten D. "Object classification for human and ideal observers," *Vision Res* 1995;35:549–568. [PubMed: 7900295]
5. Abbey CK, Barrett HH. "Human- and model-observer performance in ramp-spectrum noise: effects of regularization and object variability," *J Opt Soc Am A* 2001;18:473–488.
6. Schwartz O, Simoncelli EP. "Natural signal statistics and sensory gain control," *Nat Neuroscience* 2001;4:819–825.
7. H. L. Van Trees, *Detection, Estimation, and Modulation Theory, Part I* (Academic, New York, 1968).
8. Zhu SC, Wu Y, Mumford D. "Filters, random fields and maximum entropy (FRAME)," *Int J Comput Vision* 1998;27:1–20.
9. Simoncelli EP, Olshausen B. "Natural image statistics and neural representation," *Annu Rev Neurosci* 2001;24:1193–1217. [PubMed: 11520932]
10. H. H. Barrett, C. Abbey, B. Gallas, and M. Eckstein, "Stabilized estimates of Hotelling-observer detection performance in patient-structured noise," in *Medical Imaging: Image Perception*, H. L. Kundel, ed., Proc. SPIE 3340, 27–43 (1998).
11. Rolland JP, Barrett HH. "Effect of random background inhomogeneity on observer detection performance," *J Opt Soc Am A* 1992;9:649–658. [PubMed: 1588452]
12. Bochud F, Abbey CK, Eckstein M. "Statistical texture synthesis of mammographic images with clustered lumpy backgrounds," *Opt Express* 1999;4:33–43. <http://www.opticsexpress.org>.
13. H. H. Barrett and K. J. Myers, *Foundations of Image Science* (Wiley, New York, to be published).
14. Clarkson E, Kupinski MA, Barrett HH. "Transformation of characteristic functionals through imaging systems," *Opt Express* 2002;10:536–539. <http://www.opticsexpress.org>.
15. C. P. Robert and G. Casella, *Monte Carlo Statistical Methods* (Springer-Verlag, New York, 1999).
16. W. R. Gilks, S. Richardson, and D. J. Spiegelhalter, eds., *Markov Chain Monte Carlo in Practice* (Chapman & Hall, Boca Raton, Fla., 1996).
17. Myers KJ, Barrett HH. "Addition of a channel mechanism to the ideal-observer model," *J Opt Soc Am A* 1987;4:2447–2457. [PubMed: 3430229]

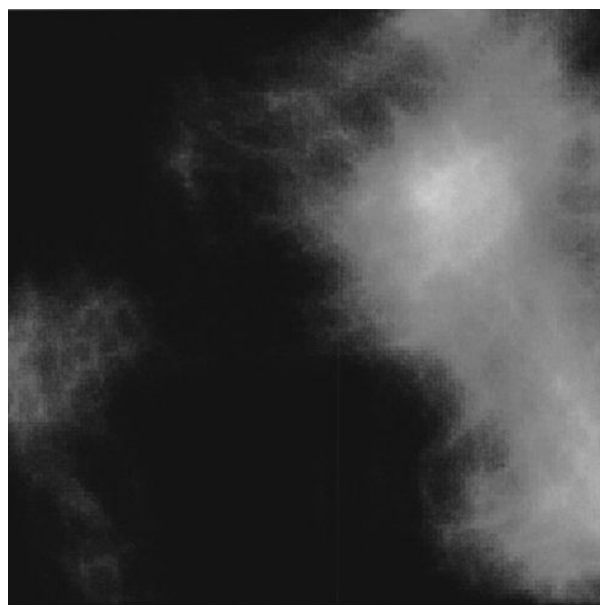


(a)

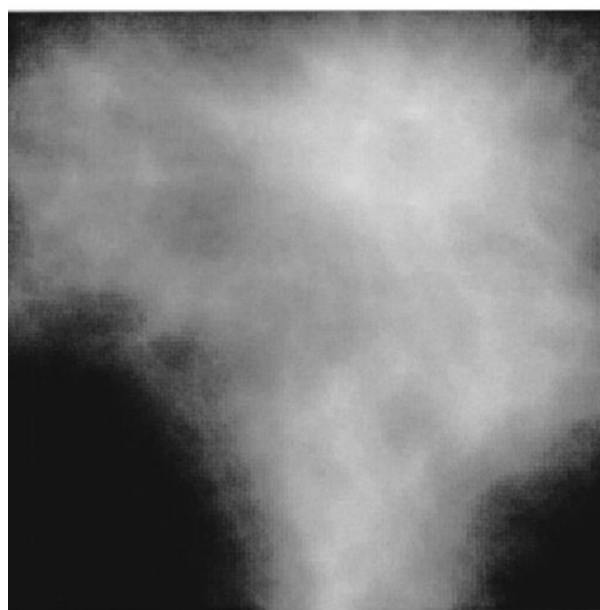


(b)

Fig. 1.
Examples of two-dimensional lumpy objects with different parameters θ .



(a)



(b)

Fig. 2.
Examples of two-dimensional clustered lumpy objects with different parameters θ .

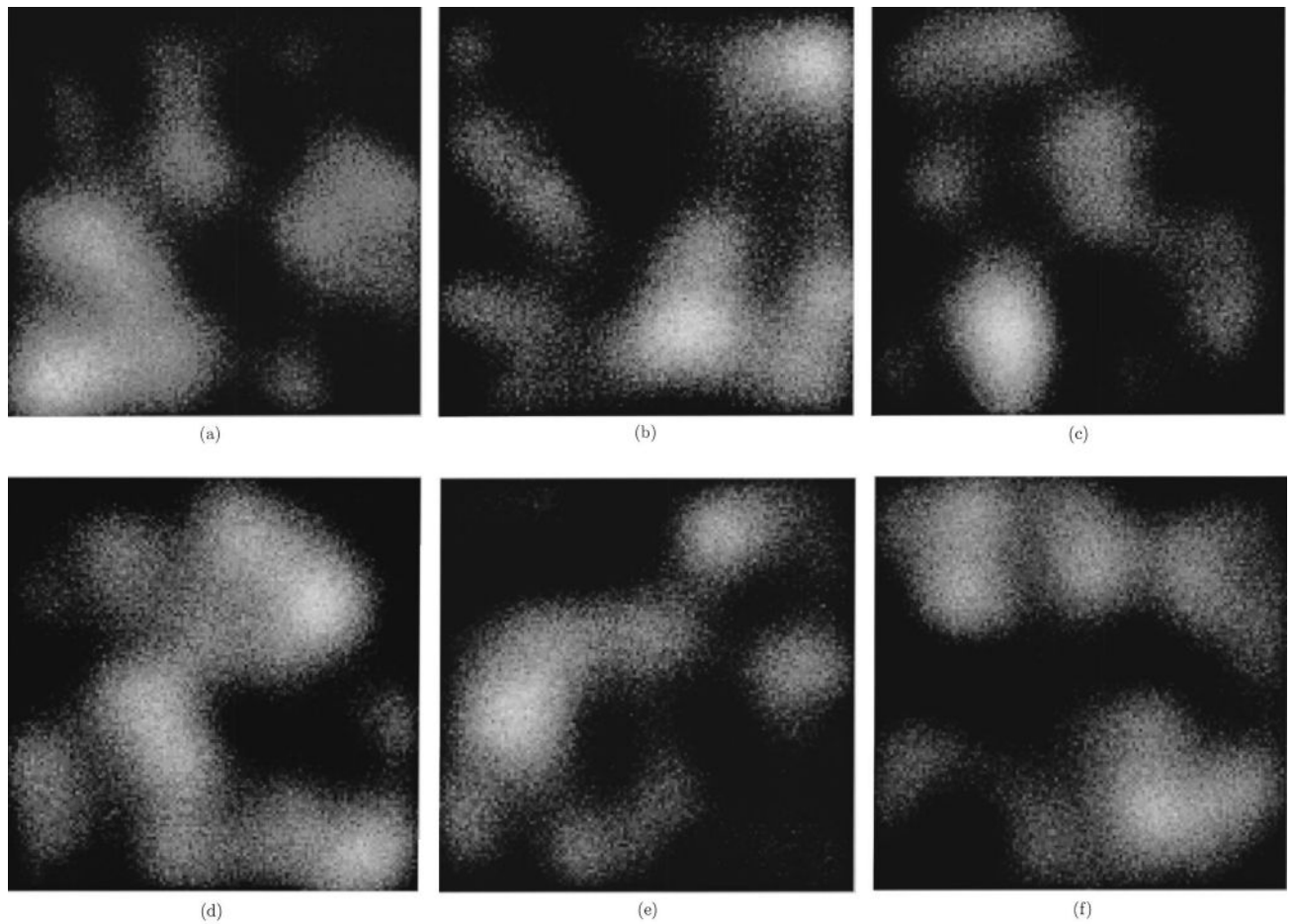


Fig. 3. Simulation results for lumpy objects. (a)–(c) Three of the images used to fit the lumpy object model parameters. (d)–(f) Three images generated by using the fitted model parameters. The top row of images are not meant to look exactly like the bottom row of images; they are simply meant to appear statistically similar.