Compact Appearance in Object Populations Using Quantile Function Based Distribution Families

Robert Elijah Broadhurst

A dissertation submitted to the faculty of the University of North Carolina at Chapel Hill in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Computer Science.

Chapel Hill
2008

Approved by:

Stephen M. Pizer, Advisor

Edward Chaney, Reader

Leonard McMillan, Reader

Carlo Tomasi, Reader

Andrew Nobel, Reader

ABSTRACT

ROBERT ELIJAH BROADHURST: Compact Appearance in Object Populations Using Quantile Function Based Distribution Families
(Under the direction of Stephen M. Pizer)

Statistical measurements of the variability of probability distributions are important in many image analysis applications. For instance, let the appearance of a material in a picture be represented by the distribution of its pixel values. It is necessary to model the variability of these distributions to understand how the appearance of the material is affected by viewpoint, lighting, or scale changes. In medical imaging, an organ's appearance varies not only due to the parameters of the imaging device but also due to changes in the organ, either within a patient day to day or between patients. Classical statistical techniques can be used to study distribution variability, given a distribution representation for which variation forms linear subspaces. For many distributions relevant to image analysis, standard representations are either too constrained or have nonlinear variation, in which case classical linear multivariate statistics are not applicable. This dissertation presents general, non-parametric representations of a variety of distribution types, based on the quantile function, for which a useful class of variability forms linear subspaces. A key consequence is that principal component analysis can be used to efficiently parameterize their variability, *i.e.*, construct a distribution family.

The quantile function framework is applied to two driving problems in this dissertation: (1) the statistical characterization of the texture properties of materials for classification, and (2) the statistical characterization of the appearance of objects in images for deformable model based segmentation. It is shown that in both applications the observed variability forms appropriately linear subspaces, allowing efficient modeling. State of the art results are achieved for both the classification of materials in the Columbia-Utrecht database and the segmentation of the kidney, bladder, and prostate in 3D CT images. While the applications presented in this dissertation use image-based appearance observations in the field of image analysis, the

methods and theory should be widely applicable to the variety of observations found in the many scientific fields, and, more specifically, to shape observations in the field of computer vision.

ACKNOWLEDGMENTS

First and foremost, this document could not have been completed without the immense amount of time happily given by my advisor, Stephen M. Pizer. The quality of this document reflects his motivating influence. My other committee members, Edward Chaney, Leonard McMillan, Carlo Tomasi, and Andrew Nobel, were also very helpful in putting the final touches on this document. The below quote compactly expresses by gratitude to them.

> "I have made this letter longer than usual, only because I have not had the time to make it shorter." $\sim$ Blaise Pascal

The research contained in this dissertation reflects how Steve and Ed picked me up when I first arrived in Chapel Hill in 2002. They stuck with me to the end, and I am thankful. This work could also not have been completed without the collaborations with all of the past and present members of the Medical Image Display and Analysis Group. Funding for this work was provided by National Institutes of Health grant P01 EB02779.

When I moved to Chapel Hill 6 years ago, I moved far from my family and friends on the west coast. Both the University of North Carolina and the department of Computer Science have supplied a wonderful, supportive environment; you all have been a great family to me during my time here. Thank you.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

BTF	bidirectional texture function
CDF	cumulative distribution function
CT	computed tomography
CUReT	Columbia-Utrecht reflectance and texture database
EMD	Earth Mover's distance
FLD	Fisher linear discrimination
GLCM	gray level co-occurrence matrix
HDLSS	high dimension low sample size
LBP	local binary pattern
QF	quantile function
MLE	maximum likelihood estimate
NN	nearest neighbor
MRF	Markov random field
PCA	principal component analysis
PDF	probability density function
SVM	support vector machine
QDA	quadratic discriminant analysis

Chapter 1

Introduction

1.1 Motivation

The variability of probability distributions of image features plays an important role in understanding the ever increasing number of observations of the world around us. Modeling the variation of an observation by estimating its probability distribution density is a fundamental technique in the sciences. Understanding the variation of more complex objects requires a hierarchy of distribution estimates, when each object is itself a distribution estimate of a collection of finer scale observations. In image analysis, observations take the form of many pixel values in several images. A hierarchy can be formed by modeling the variation across images of an object itself described by the variation of its pixel values across each image.

The goal of image analysis is to understand an image, which involves answering questions similar to the following:

1. What is this a picture of?

2. What object is in this image? Where is it?

Such questions are usually asked in a supervised context where there is prior information about the possible objects of interest. Prior information encapsulates such notions as what objects to expect, their shape, or their appearance in an image. For instance, a picture of a material, such as cork or sponge, can be identified after learning its appearance from pictures under different viewing and illumination conditions. For the second question, the location and shape

of the object in the image also plays an important role. The task of locating specific organs, such as the bladder or prostate, from 3D CT images is an example where there is strong location, shape, and appearance prior information. This dissertation focuses on appearance information.

These examples benefit from a statistical characterization of the available prior knowledge, which comes in the form of a population of examples. To encode this information, a representation of the location, shape, or appearance of the object must be chosen. Then a probability distribution of the representation's variability is estimated from the examples. A key challenge in this process is to find an appropriate representation of appearance, where one desired property is compactness. Compact representations have variation that is linear in their parameters, which allows them to be estimated using efficient, classical statistical methods, such as principal component analysis. This dissertation is concerned with representations of probability distributions that naturally describe object appearance and with understanding their variation so that they can be compactly and linearly modeled.

Previous approaches to modeling the variability of probability distributions have been based on two types of distribution representations. In the first approach, a probability distribution is represented as a member of a parametric distribution family. The family is chosen for an application specifically so that the variation is linear in its parameters. Families, however, are constrained models of distributions, which means they can only represent certain distributions. For example, the distributions arising from pictures of materials or from regions near boundaries of organs in CT images, are often too complex to lend themselves to standard distribution families.

In the second approach, a probability distribution is represented non-parametrically as a histogram. This allows any arbitrary distribution to be represented, but their variation for most applications forms nonlinear manifolds. Therefore, computing statistics of histogram variation is difficult, so most work focuses on defining application-specific nonlinear distance metrics. In this dissertation, the focus is instead on finding a representation for which the distance metric is Euclidean.

This dissertation presents representations of several types of probability distributions that

are a generalization of the quantile function (QF). The quantile function is a description of univariate distributions that, when estimated discretely, allows general, non-parametric representations for which a useful class of variability forms linear subspaces. This dissertation extends these concepts to multivariate and conditional distributions, and distributions consisting of a mixture of multiple underlying distributions.

The driving problems of this dissertation are two: (1) the statistical characterization of the texture properties of materials for classification, and (2) the statistical characterization of the appearance of objects in images for deformable model based segmentation. While the applications presented in this dissertation use image-based appearance observations in the field of image analysis, the presented representations and underlying theory should be more widely applicable. Within image analysis and computer vision, descriptions of object shape may lend themselves to particularly well suited probability distributions due to their complex shape and variation. Beyond computer vision, the representation of observations as probability distributions is a common technique in many scientific fields. The theory presented here should help in understanding, and understanding the importance of, linear variation of distribution representations in any application. Given this understanding, the specific representations presented in this dissertation could also be directly applicable.

Sections 1.1.1 and 1.1.2 continue the motivation for the two driving applications of this work: texture analysis and modeling object appearance.

1.1.1 Texture Analysis

Texture is a broad concept that describes the characteristic visual and tactile properties of objects. Characteristic properties are distinctive as judged by human perception, making it difficult to precisely define texture despite its use in computer science for decades. In this dissertation, texture refers to the characteristic visual patterns of the surface of an object, or that the object itself consists of, when observed through an imaging device. Such patterns describe the spatially repetitive layout of many small pieces across the surface or interior of the object, so is often described statistically, rather than attempting to explicitly model each element of the texture.

Texture analysis encapsulates the information in such patterns for (1) discrimination, (2) synthesis, and (3) object inference. Discrimination seeks descriptions of texture classes in order to differentiate them. Discrimination is used for classification tasks, where an entire image or a prelabeled object is identified, and for segmentation tasks, where an object is located within an image. Examples include the labeling of terrain type from arial photographs, retrieval from a database of an image similar to a reference image, and the identification of pictures of materials such as sponge, cork, and wood.

Texture synthesis is the process of generating an image of a texture with the same characteristic properties as, but is not necessarily identical to, a given texture. Examples include image restoration, where a damaged, textured portion of an image is replaced using a similarly textured image region, and computer games, where textures are synthesized using a compact description instead of storing large texture images.

Object inference is the process of inferring, from a given property such as texture, additional object properties such as pose or shape. An example is the recovery of the parameters, such as viewing and illumination directions, used to take a picture of a planar material.

Statistical descriptions, and more specifically, linear statistical descriptions such as the ones presented in this dissertation, are useful in all of these tasks. For example, consider all of these tasks in the context of a database of materials imaged under different viewing and illumination directions. Chapter 3 presents the texture discrimination task of identification on such a database. Future work section 5.2.2 discuses a synthesis task facilitated by a linear statistical description: the generation of textures from arbitrary viewing and illumination directions, given examples at specific directions. Section 5.2.2 also discusses object inference, where, for example, the discrimination task above could be made more difficult by also estimating the viewing and illumination directions used to capture each image.

1.1.2 Modeling Object Appearance

Object appearance is a general description of the appearance of an object with respect to an imaging device; it is a function of both the object and the imaging device. Chapter 3 considers in depth one aspect of object appearance, texture, in the constrained situation of

(1) having a homogeneous appearance across the object, and (2) modeling variation due only to changes in the imaging device. Chapter 4 focuses on descriptions of organs in 3D medical images, which requires building models of object appearance without such constraints.

The appearance of objects in 3D medical images is captured for a variety of tasks, such as (1) segmentation, (2) identification, and (3) validation. Chapter 4 describes two segmentation tasks in detail: the segmentation of the left kidney in 3D CT images using an across-patient data set and the segmentation of the bladder and prostate in 3D CT images using several independent, within-patient data sets. The segmentation of the bladder and prostate is required, for example, for planning external beam treatment for prostate cancer. Automatic segmentation methods reduce the time of medical professionals, increase reproducibility, and hopefully maintain a comparable level of precision. Identification and validation tasks both ask hypotheses about an existing object. Example identification tasks include determining if a tumor is present and distinguishing between a healthy and a diseased organ. Validation can, for example, be combined with an automatic segmentation method to facilitate manual editing of the segmented object by determining which portions of the object boundary are invalid.

The object appearance models used for these tasks are composed of representations of observations (image region summaries) made at specific locations and scales. The construction of appearance models of organs in medical images is driven by several factors. First, the model must distinguish between the interior and exterior of the object. Second, the irrelevance both of variation far from the organ boundary and of per voxel texture variation must be taken into account. Third, the observations must be specific enough and the representation rich enough to capture complex grey level appearances near the organ boundary. Finally, the expected variation of the representation due to such factors as imaging device normalization and tissue movement should form linear subspaces. Chapter 4 presents both an object appearance model that addresses all of these issues and a learned likelihood of the model for use in several segmentation tasks.

1.2 Thesis and Claims

Thesis: *Quantile functions provide a general framework for learning compact representations of probability distributions. This allows accurate and efficient Bayesian methods for texture classification and image segmentation using distributions of image-based appearance features.*

The contributions of this dissertation are the following:

1. A geometric interpretation of the space of discrete quantile functions has been developed and described. A key analysis linked the non-parametric representation of the quantile function to several common parametric distribution families.

2. A novel framework has been developed for representing the variability of multivariate and conditional distributions, and distributions consisting of a mixture of multiple underlying distributions. These quantile function based representations are natural in the sense that their Euclidean distance is an efficient approximation of the Mallows distance. Their variation is parametrically estimated, which results in the learning of task-specific distribution families.

3. Texture models using the QF based multivariate and conditional distribution representations have been demonstrated. Both filter bank texture models and Markov random field texture models have been developed and expressed in a common framework, allowing their strong similarities and specific differences to be described.

4. A method for the texture based classification of pictures of materials has been developed and demonstrated. It leverages the demonstrated linearity of the proposed texture models to viewpoint and lighting variation to produce the best reported classification accuracy to date on a standard CUReT database classification task. It is also at least an order of magnitude more compact and computationally efficient than existing methods.

5. A multi-scale appearance model for objects in images has been developed. It leverages surface correspondences supplied by a shape model to generate region descriptions at scales as coarse as the entire inside or outside of the object, as fine as individual boundary points, or in between at one of several novel, local scales.

6. A likelihood term for the Bayesian segmentation of organs in 3D CT images has been proposed and tested. It has been shown that between-patient variation and day-to-day variation of object-relative image regions are efficiently modeled by the quantile function mixture representation. State of the art segmentation results have been achieved in left kidney, bladder, and prostate segmentation experiments.

1.3 Overview of Chapters

This dissertation is organized in five chapters. This chapter motivated the application of quantile function based distribution representations to image analysis tasks, and it summarized the contributions of this dissertation.

Chapter 2 presents several quantile function based distribution representations, the core methodology of this dissertation. A basic review of univariate probability distributions is given, and their various representations, including the quantile function, are compared. See Chapters 3 and 4 for more detailed background material specific to texture classification and medical image segmentation. In Chapter 2 the quantile function based representations are presented and their linear subspaces, Euclidean distance, and likelihood estimation are discussed.

Chapter 3 applies the statistical methods presented in Chapter 2 to texture classification. Background material including related work, the CUReT database, and the MR8 filter bank are presented. Filter bank and Markov random field based texture models are constructed using the multivariate and conditional distribution representations. A likelihood is estimated for classification that models viewpoint and illumination variation of pictures of materials.

Chapter 4 applies the statistical methods presented in Chapter 2 to the segmentation of organs in CT images. Background material on medical image analysis and deformable shape models is presented. A multi-scale appearance model of objects in images is developed and used to describe the left kidney, bladder and prostate. A likelihood is estimated for segmentation that models the day-to-day and between patient appearance variation of these organs.

Chapter 5 discusses the contributions of this dissertation and concludes with future possible extensions and applications.

Appendix A is a user guide that discusses in detail the basic algorithms developed in this dissertation for computing and displaying QFs.

Chapter 2

Quantile Function Based Distribution Representations

This chapter lays out the properties of quantile functions for representing probability distributions and their variation. It then presents several generalizations of the quantile function for representing probability distributions beyond standard, univariate distributions. The construction of each representation is driven by the goal of understanding its linear subspaces, Euclidean distance, and appropriateness for various estimation tasks. These representations represent the core methodology of this dissertation, and they are used to build models of texture and object appearance in the driving problems presented in Chapters 3 and 4.

First, Section 2.1 reviews the quantile function and other univariate distribution representations, discusses their linear subspaces, and explores quantile functions as a geometric space. Section 2.2 presents representations based on the quantile function of multivariate and conditional distributions, and distributions consisting of a mixture of multiple underlying distributions. Section 2.3 presents a method for estimating the likelihood of these representations given an example set. This likelihood is used for classification in Chapter 3 and segmentation in Chapter 4.

2.1 Univariate Probability Distributions

Univariate probability distributions, long studied in statistics [Ros02], describe the likelihood of a random variable attaining a specific value. The allowed values are either discrete,

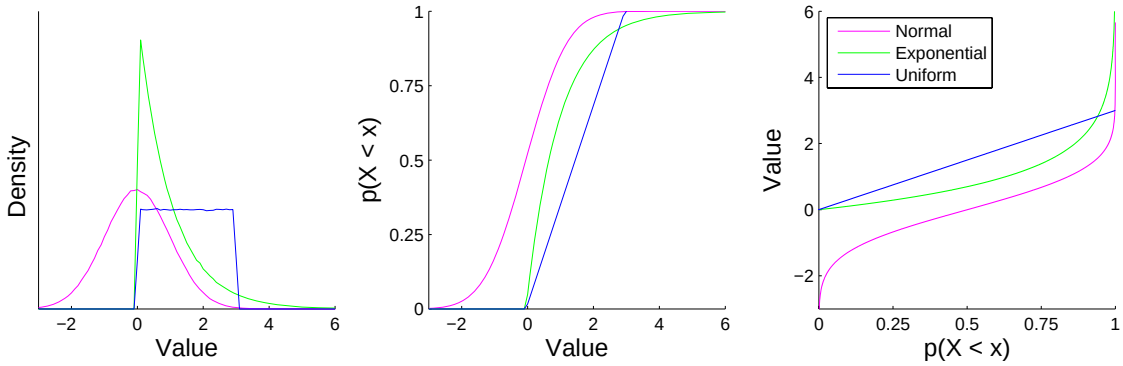


Figure 2.1: The probability distribution function (left), cumulative distribution function (center) and quantile function (right) of several common distributions.

such as the integers $\mathcal{Z}$, or continuous, such as the real line $\mathcal{R}$. The remainder of this section discusses continuous random variables; the treatment of discrete random variables is similar. Let X be a continuous random variable with probability density function (PDF) f. f has the constraints

$$f(x) \geq 0, \qquad x \epsilon X$$

$$\int_{x \epsilon X} f(x)dx = 1.$$

Most probability distributions can be equivalently described by their PDF, cumulative density function (CDF) F, or quantile function (QF) Q. The CDF describes the probability of attaining a value less than or equal to x and is defined as

$$F(x) = \int_{-\infty}^{x} f(u)du.$$

The QF is the inverse of the CDF, and it can be carefully defined as

$$Q(x) = \inf\{u : F(u) \geq x\}$$

when $F(x)$ is not strictly increasing. Both the CDF and QF are non-decreasing functions. Figure 2.1 shows the PDF, CDF, and QF for several common distributions. These are examples of parametric distributions, where the PDF or CDF is given analytically and is expressed in terms of a small number of parameters. For example, the PDF of the Gaussian distribution

$\mathcal{N}(\mu, \sigma)$ is $f(x) = \frac{1}{\sigma\sqrt{2\pi}} \mathrm{e}^{\frac{(x-\mu)^2}{-2\sigma^2}}$. Distributions can also be described non-parametrically, where the domain of f, F, or Q is divided into subsets and for each a value is specified.

2.1.1 Distribution Families

Example parametric distributions include the Gaussian, exponential, uniform, gamma, and beta distributions. Each is considered a distribution family because they express a set of related probability distributions. Families can also be related, by the type of their parameters or by other shared properties. The above examples are two-parameter families. The Gaussian, exponential, and uniform distributions are examples composed of location and scale parameters. So called location-scale families are common and easy to understand since they change the mean and standard deviation of a distribution, respectively, without affecting the shape of the PDF. Location and scale play an important role in understanding quantile functions and are discussed more in Sections 2.1.3 and 2.1.4.

More general families are constructed using parameters beyond location and scale. These additional parameters describe either mixture or shape changes. Mixture parameters construct distributions using the PDFs of several existing distributions. Let $f_1, f_2, \ldots, f_n$ be the PDFs of n independent distributions. A mixture distribution with PDF f is defined as $f = \sum_{i=1}^{n} w_i f_i$, where $\sum_{i=1}^{n} w_i = 1$ and $0 \leq w_i \leq 1$, $i = 1, 2, \ldots, n$. Mixture distributions are usually constructed using distributions from the same family, most commonly the Gaussian family. Mixture parameters are important in understanding non-parametric distributions and are discussed more in Section 2.1.3.

Shape parameters affect the characteristic shape of a distribution's PDF. Several distributions, including the Weibull and the gamma, include a single extra shape parameter. These distributions often generalize more specific location-scale families. For example, the Weibull distribution generalizes the Rayleigh and exponential distributions, and the the gamma distribution generalizes the chi-squared and exponential distributions. Jensen's family is an example that contains 2 shape parameters. This family generalizes many common distribution families, including the Gaussian and Weibull families, which exist as a point and a line, respectively, in Jensen's two-dimensional space of shape parameters.

Many parametric distributions are also part of the general exponential family. The exponential family has been extensively studied because the common form of its distributions leads to desirable properties related to sufficient statistics, estimation, and conjugate distributions. The exponential family includes the Gaussian, gamma, chi-square, beta, Dirichlet, Bernoulli, binomial, multinomial, Poisson, negative binomial, and geometric distributions. The relationship between the exponential family and other parametric distributions has been studied using differential geometry [Ama85]. In this approach each parametric distribution family describes a submanifold in the infinite-dimensional space of log-likelihoods. A key property is the curvature of the submanifold, measured as changes in the submanifold's tangent space. Exponential families form linear submanifolds in the log-likelihood space.

Throughout Section 2.1 I use the same parametric families for demonstration. Some of these families are chosen because they are standard. These include the Gaussian, uniform, exponential, gamma, and beta distributions. Other distributions are chosen because they are related to the application chapters, Chapters 3 and 4. These include the Weibull distribution, which is related to stochastic textures in Chapter 3, and the Rayleigh and Fisher-Tippett distributions, which are related to ultrasound images.

The above methods describe the relationships between parametric distribution families. Non-parametric distribution representations do not construct families in the same manner as parametric representations, since they are unconstrained. However, a notion of a distribution family can be developed for non-parametric representations by considering submanifolds in their space. In particular, this dissertation examines linear subspaces of quantile function based representations. First, Section 2.1.2 defines the non-parametric distribution representations and Section 2.1.3 describes and compares their Euclidean distances and their linear subspaces. Section 2.1.4 describes the space of quantile functions in detail and concludes 2.1 by discussing additional properties of the quantile function.

2.1.2 Estimation and Non-parametric Distributions

Estimation tasks seek parametric or non-parametric representations of a probability distribution derived from a set of samples from that distribution. Parametric estimation consists

of first choosing a distribution family and then estimating the parameters of the distribution. Many methods have been developed in statistics to accurately estimate parameters according to a metric and to measure the resulting estimation error. However, in many applications, such as the image analysis applications considered in Chapters 3 and 4, the samples are from complex distributions that do not fit existing parametric distribution families. It is in this context, the estimation of complex distributions, that non-parametric distributions are typically studied.

Non-parametric distributions are discrete representations of a distribution's (1) probability density function (PDF), (2) cumulative density function (CDF), or (3) quantile function (QF), the focus of this dissertation. Non-parametric PDF estimates are the most popular; in this dissertation these are referred to as histograms. To construct a histogram, the real line is divided into subsets x_i called bins whose frequencies are estimated. The location of the bins are normally defined by their boundaries with $b-1$ bin boundaries defining b bins. For univariate distributions it is typical to use equally spaced bins. Section 2.2.1 discusses multivariate distributions, where more complex binning strategies are often required. A histogram $\underline{h}$ with b bins x_i is defined as

$$h_i = \int_{x_i} f(u)du, \qquad i = 1, \ldots, b, \tag{2.1}$$

where $\sum_{i=1}^{b} h_i = 1$ and $0 \leq h_i \leq 1, i = 1, \ldots, b$.

Given a set of s samples, a histogram is easily constructed in $O(s \log b)$ time, or $O(s)$ time for equally spaced bins, by comparing each sample to the bin boundaries. The count in each bin is then normalized into a frequency by dividing by s. Figure 2.3 shows a Gaussian distribution estimated from 1024 samples for different values of b. Histograms are sensitive to b; this is discussed more at the end of this section and in the next section.

Non-parametric representations based on the CDF are constructed using histograms. A discrete CDF $\underline{H}$ is defined as

$$H_i = \int_{x_1,\ldots,x_i} f(u)du, \qquad i = 1, \ldots, b, \tag{2.2}$$

where $0 \leq H_1 \leq \ldots \leq H_b \leq 1$. $\underline{H}$ can be constructed from $\underline{h}$ by computing $H_i = \sum_{j=1}^{i} h_j$.

The construction of a discrete QF differs from that of PDFs and CDFs. Given a quantile c and a random variable X, a QF computes the value x for which $p(X < x) = c$. The domain of a QF is therefore between 0 and 1 and represents the cumulative probability of the distribution. PDFs and CDFs, on the other hand, have domains based on the values the random variable achieves; this is the range of QFs. A discrete QF is computed for regularly spaced values of c between 0 and 1. Let $\underline{Q}$ be a discrete QF with b values. Each element, Q_i, is called a quantile and represents $1/b$ of the distribution. Similar to $\underline{h}$, each element of $\underline{Q}$ actually represents a piecewise integration of Q,

$$Q_i = b \int_{\frac{i-1}{b}}^{\frac{i}{b}} Q(x)dx, \qquad i = 1, \ldots, b, \tag{2.3}$$

where $Q_1 \leq \ldots \leq Q_b$. Each quantile is multiplied by b so that it is the average value of the quantile function over the quantile's domain.

In this dissertation, $\underline{h}$, $\underline{H}$, and $\underline{Q}$ are typically considered as estimates of f, F, and Q, even though they are in fact piecewise integrations of these functions. Since integration is a linear operation, this distinction is not crucial.

Given a set of s samples from a distribution, and if $b = s$, $\underline{Q}$ is constructed by simply sorting the samples. To construct a lower dimensional representation with $b < s$, adjacent, sorted samples are averaged together. In this case, complete sorting is not required, allowing the QF to be computed in $O(s \log b)$ time. For continuous distributions this would require a complex median search algorithm, so in this case I use a simple $O(s \log s)$ sorting algorithm. For discrete distributions with v possible values, a $O(s + v)$ algorithm can be constructed without a loss in accuracy by first computing a v bin histogram. Also, some applications, such as the image segmentation task in Chapter 4, supply weighted samples, which requires a more complicated averaging step. Section A.1 gives MATLAB code for computing QFs from unweighted samples, weighted samples, and weighted samples from a discrete distribution.

The s sorted samples are estimates of order statistics, so the b quantiles are averages of order statistics [Dav70]. Also, $\underline{Q}$ can be understood by considering it as an adaptive bin histogram, where each bin has the same frequency and the average of each bin is stored. Figure 2.2 shows

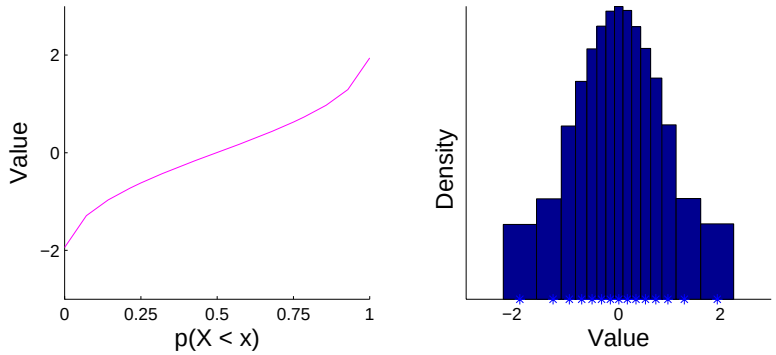


Figure 2.2: The Gaussian distribution represented as (left) a discrete quantile function with 25 values and (right) the QF's corresponding adaptive bin histogram.

an example QF and its corresponding adaptive bin histogram, whose estimation is described in Section A.2. The resulting adaptive bin histogram demonstrates two desirable properties of quantile functions: (1) the bin locations are automatically set so that arbitrary bin boundaries need not be defined, and (2) the bins automatically focus on the more likely portions of the distribution, as shown by the variable width and location of the bins.

An intuitive understanding of QFs can be achieved by considering what a discrete QF represents as its size is varied. Single value QFs represent a distribution's mean, two values are linearly equivalent to the mean and the standard deviation, and more values further describe a distribution's shape. QFs, therefore, gradually provide a detailed description of distribution shape as its size is increased, after first capturing location and scale. The mean and standard deviation equivalence of $\underline{Q}$ is based on $\underline{Q}$ being a piecewise integration of Q.

All three non-parametric representations have a single common parameter b, the number of bins. The different representations, however, are sensitive to b in different ways, which often depend on the relationship of b with s, the number of samples. These sensitivities are also confounded by the need, often for comparison, to estimate multiple distributions using the same bins. First, consider the case of a small b, for which PDF and CDF representations have a large discretization error. This error depends upon how tight the domain can be restricted, which is a function of the number, similarity, and tightness of the distributions. For example, consider 10 Gaussian distributions with unit standard deviations and means that vary equally spaced from 1 to 10. To estimate from samples the mean of these distributions using the same

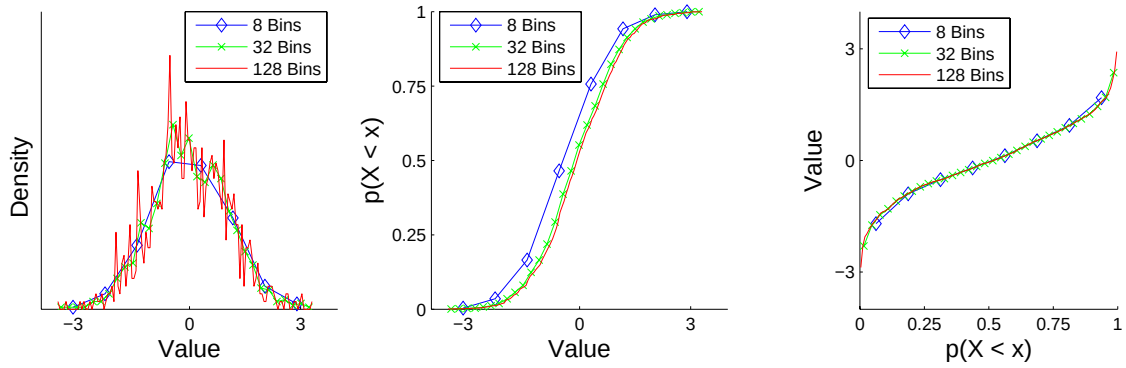


Figure 2.3: A discrete PDF, CDF, and QF of a Gaussian distribution estimated from 1024 samples. Notice the stability of the CDF and QF estimates.

bins for all the distributions, several 10s of bins are required to avoid large and misleading errors. To accurately estimate their standard deviations, even more bins would be required. QFs, on the other hand, do not suffer from this form of discretization error. In this example, QFs exactly capture all 10 distributions using two bins, which is discussed in the next section.

Another case to consider is the so called over-binning situation. When b is large, possibly larger than s, PDF estimates become unstable. Consider two sets of samples from the same distribution. It is likely that many of the samples from the two sets will be in nearby but different bins. Therefore, the histograms corresponding to these two sets of samples will be incorrectly considered as dissimilar. Distance measures between histograms and the other non-parametric representations are discussed more in the next section. For CDFs and QFs, this is not an issue. Since they both consider the integration of the PDF, corresponding bins correctly reflect the sampling error without introducing additional discretization errors. Additionally, QFs capture all information in the samples once $b = s$, including the sampling error, so increasing b beyond s has no effect.

Figure 2.3 shows PDF, CDF, and QF estimates of a Gaussian distribution from 1024 samples. The number of bins is varied from 8 to 128 to demonstrate the effects of changing the number of bins on each of these representations. The PDF estimate is sensitive to the number of bins while the CDF and QF estimates are stable. The CDF has a consistent shift to the right as the number of bins is increased. This discretization error is a display artifact caused by the fact that some samples get rounded down to the bin center, causing an overestimation

in the integration. This can be fixed by displaying CDFs with respect to the right edge of the bins instead of the bin centers.

This section described how to construct non-parametric distribution representations and compared them with respect to their common parameter, b. CDFs and QFs were shown to be less sensitive than PDFs, and QFs were shown to be more compact than PDFs or CDFs. These desirable properties of QFs are well expressed by considering their construction. Only two operations are performed during their estimation, sorting and averaging. Both operations decrease noise and neither introduce artifacts.

Now that the non-parametric representations have been introduced and their construction discussed, the next section discusses the linked properties of distance and interpolation.

2.1.3 Distance Measures and Interpolation

Representations are often analyzed through the linked ideas of distance and interpolation, where desired interpolations correspond to paths of minimal distance. In general, a submanifold of the representation's feature space is of interest. This possibly nonlinear submanifold can be specific to the data in a particular application; it can also be a general restricted submanifold of interest. For instance, a representation's feature space is often restricted to the submanifold that corresponds to valid, or legal, representations of the object. Examples include a histogram $\underline{h}$, which has the linear constraints $0 \leq h_i \leq 1$, $i = 1, \ldots, b$ and $\sum_{i=1}^{b} h_i = 1$, and a discrete QF $\underline{Q}$, which has the linear constraints that $Q_1 \leq Q_2 \leq \ldots \leq Q_b$. Desired interpolations stay on the submanifold of interest and follow paths of minimal distance called geodesics. The distance measure defines the geodesic paths and penalizes points for being off of the submanifold.

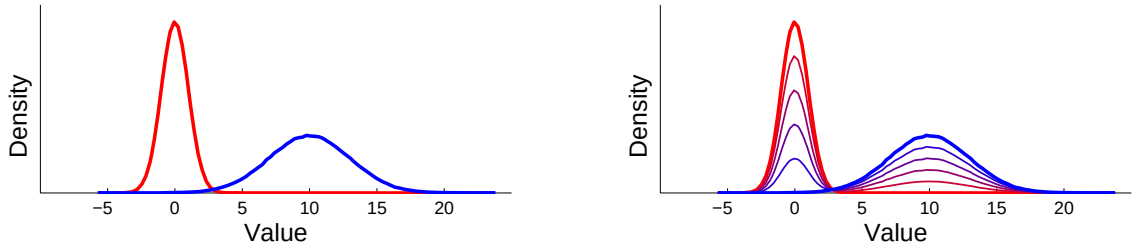
In this dissertation I am particularly interested in analyzing variation. Depending on the properties of the submanifold, this can be both theoretically and computationally challenging [FLPJ04]. Thus representations are often sought for which the submanifolds of interest are linear, *i.e.*, that have Euclidean distance as their distance metric, so interpolation follows straight line paths. Therefore, both interpolations and distances can be computed efficiently using linear operators. Also, notions such as hyperplanes and linear projection are well established.

There is also a large set of well developed statistical tools for linear submanifolds that leverage the notions above. Section 2.3 uses Principal Component Analysis (PCA) in this setting for covariance estimation. The usefulness of linear representations and their likelihood estimation is further discussed in Section 2.3 for QF based representations of probability distributions.

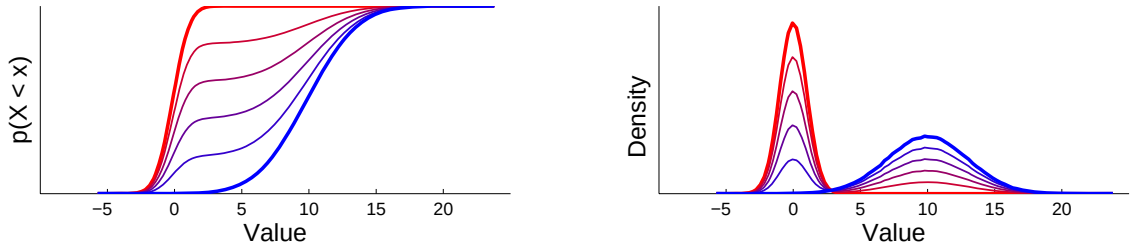
For probability distributions, distance and interpolation can be considered for both parametric and non-parametric distribution representations. A parametric representation is chosen for a particular application because the distributions of interest can be modeled by the parametric representation. Additionally, all distributions modeled by the representation typically match those of interest. Therefore, distributions linearly interpolated by the representation are valid for the application, and Euclidean distance is reasonable. For a particular application, the existence of such a parametric representation is ideal.

For many applications, however, the distributions of interest do not fit any of the existing parametric representations. In this case, non-parametric representations are used since all non-parametric representations can accurately estimate any distribution. Given a set of distributions, however, a non-parametric representation should be sought that is close to ideal, *i.e.*, a representation that describes the variation in the sample set as a linear subspace. This dissertation focuses on the usefulness of QFs for this task and how the variation in a particular sample set can be learned and expressed in a few parameters, in effect learning an ideal application-specific parametric representation. Towards this end, the remainder of this subsection examines distance measures between probability distributions and the linear subspaces of PDFs, CDFs, and QFs.

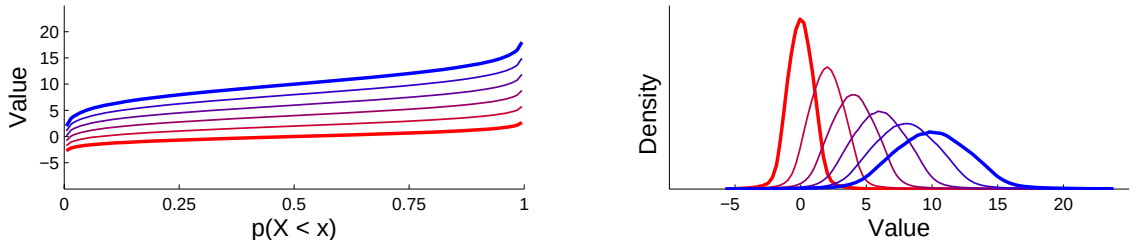
A large body of literature has explored many different distance measures between non-parametric representations of probability distributions, including the Earth Mover's distance (EMD) [RTG00], diffusion distance [LO06], CDF L_p norm, χ^2 distance, histogram intersection, quadratic form, and Kullback-Leibler divergence (see [PRTB99] for a survey). The appropriateness of a distance measure for a particular application depends on the type of variation of the distributions of interest. To examine the properties of distance measures in the general case, however, it is interesting to consider the distance measured with respect to the parameters of the various parametric representations. Since a non-parametric representation is



(a) Two Gaussian distributions, $\mathcal{N}(0,1)$ and $\mathcal{N}(10,3)$, (left) and four PDFs linearly interpolated between them (right).



(b) Interpolation of CDFs displayed as CDFs (left) and PDFs (right).



(c) Interpolation of QFs displayed as QFs (left) and PDFs (right).

Figure 2.4: Linear interpolation between two Gaussian distributions represented as PDFs, CDFs, and QFs. PDF and CDF interpolation identically describe mixtures while QF interpolation describe mean and standard deviation differences.

often used in place of a parametric representation, it is important to know the behavior of the parametric representation in the non-parametric setting.

Interpolation of PDFs, CDFs, and QFs

In order to understand distance measures and the behavior of parametric distributions in the various non-parametric settings, the linear subspaces of the non-parametric representations must first be understood. To explore linear interpolation of PDFs, CDFs, and QFs, consider Figure 2.4. Figure 2.4(a) shows two probability distributions; Gaussian distributions with means of 0 and 10 and standard deviations of 1 and 3, respectively. A tempting question to ask is "What is the correct interpolation between these two distributions"? However, given only

two example distributions, there is inadequate information to correctly answer this question. For an application the desired interpolation, or equivalently the desired submanifold, can be given by more examples. Information can also be gleaned by knowing a particular parametric family that approximately captures the distributions and variation of interest; the parametric family corresponds to an approximately correct submanifold in the non-parametric spaces.

In Figure 2.4, the two Gaussian distributions are represented as PDFs, CDFs, and QFs. MATLAB code to generate smoothed histograms from QFs is given in Section A.2; MATLAB code to generate Figure 2.4.(c) is given in Section A.4. In Figure 2.4, linear interpolation at each argument value for each representation is given, and on the right side of Figure 2.4 they are displayed for comparison as PDFs. The interpolation given by the PDF and CDF representations is identical. As mentioned in 2.1.2, the CDF is a cumulative integration of the PDF. Cumulative integration is a linear operation and it corresponds to the following linear skew. If $\underline{h}$ is a b bin discrete PDF, the corresponding discrete CDF $\underline{H}$ is computed by $H_i = \sum_{j=1}^{i} h_j, i = 1, \ldots, b$. This can also be expressed using a $b \times b$ matrix as

$$\underline{H} = \begin{bmatrix} 1 & 0 & 0 & 0 & \ldots & 0 \\ 1 & 1 & 0 & 0 & \ldots & 0 \\ 1 & 1 & 1 & 0 & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & & \vdots \\ 1 & 1 & 1 & \ldots & 1 & 0 \\ 1 & 1 & 1 & 1 & \ldots & 1 \end{bmatrix} \underline{h}. \tag{2.4}$$

This linear skew changes Euclidean distance but not linear interpolation.

In general, interpolation of PDFs and CDFs can be understood as mixture interpolation. For the two Gaussian distributions considered in figure 2.4, with random variables $X \sim \mathcal{N}(0, 1)$ and $Y \sim \mathcal{N}(10, 3)$, the PDF and CDF interpolations can be parametrically expressed as $(1 - w) * X + w * Y$. In this example $w = 0.0, 0.2, 0.4, 0.6, 0.8, 1.0$.

The linear interpolation given by the QF representation is quite different from the interpolations given by PDFs and CDFs. In general, location and scale changes are linear for QFs. Given any b bin QF $\underline{Q}$, changing a distribution's mean and standard deviation corresponds to

a simple affine transformation, defined as

$$\underline{Q}' = \alpha I \underline{Q} + c\underline{1}, \tag{2.5}$$

where I is the $b \times b$ identity matrix and $\underline{1}$ is the $b \times 1$ vector of ones. When $\underline{Q}$ corresponds to a zero mean distribution, $\underline{Q}$ and $\underline{1}$ are orthogonal vectors, α only affects the standard deviation of the distribution, and c only affects the mean. For the two Gaussian distributions considered in Figure 2.4, $\mathcal{N}(0,1)$ and $\mathcal{N}(10,3)$, the QF interpolations directly interpolate μ and σ. The interpolations correspond to Gaussian distributions $\mathcal{N}(0,1)$, $\mathcal{N}(2,0.6)$, $\mathcal{N}(4,1.2)$, $\mathcal{N}(6,1.8)$, $\mathcal{N}(8,2.4)$, and $\mathcal{N}(10,3)$. This example highlights the fact that linear interpolation of QFs from a location-scale family produces QFs that are also in the family.

The equivalent simple affine transformations can also be considered in the PDF and CDF spaces. Unfortunately, for both PDFs and CDFs, both scaling and addition lead to illegal representations. As mentioned in Section 2.1.2, a PDF $\underline{h}$ with b bins has the linear constraints $\sum_{i=1}^{b} h_i = 1$ and $0 \leq h_i \leq 1, i = 1, \ldots, b$. Addition is orthogonal to the hyperplane of legal histograms formed by the constraint that the histogram sum to one. Multiplication also does not respect either the hyperplane or the boundary constraints. A CDF $\underline{H}$ with b bins has the linear constraints $0 \leq H_1 \leq \ldots \leq H_b \leq 1$. The full domain of a distribution is captured by $\underline{H}$ if and only if $H_1 = 0$ and $H_b = 1$. Therefore, both addition and multiplication lead to either an invalid CDF or to an incompletely captured CDF.

As discussed above, the affine transformations of PDFs and CDFs include mixture changes, and the affine transformations of QFs include location and scale changes. However, it is difficult to understand the opposite cases of location and scale changes for PDFs and CDFs, and mixture changes for QFs. Distances along the nonlinear manifolds that correspond to these types of variation are also hard to interpret. Section 2.1.1 discussed how parametric distributions are composed of location, scale, and shape parameters and how mixtures of these distributions can be constructed. Since location and scale parameters, and often shape parameters, are nonlinear in the PDF and CDF spaces, many parametric distributions form hard to understand, strongly nonlinear submanifolds. The space of QFs, on the other hand,

Rep.	Location Interpolation	Mixture Interpolation
PDF	$\begin{bmatrix}1\\0\\0\\0\\0\end{bmatrix}\begin{bmatrix}0\\1\\0\\0\\0\end{bmatrix}\begin{bmatrix}0\\0\\1\\0\\0\end{bmatrix}\begin{bmatrix}0\\0\\0\\1\\0\end{bmatrix}\begin{bmatrix}0\\0\\0\\0\\1\end{bmatrix}$	$\begin{bmatrix}1\\0\\0\\0\\0\end{bmatrix}\begin{bmatrix}0.75\\0\\0\\0\\0.25\end{bmatrix}\begin{bmatrix}0.5\\0\\0\\0\\0.5\end{bmatrix}\begin{bmatrix}0.25\\0\\0\\0\\0.75\end{bmatrix}\begin{bmatrix}0\\0\\0\\0\\1\end{bmatrix}$
QF	$\begin{bmatrix}0\\0\\0\\0\\0\end{bmatrix}\begin{bmatrix}0.25\\0.25\\0.25\\0.25\\0.25\end{bmatrix}\begin{bmatrix}0.5\\0.5\\0.5\\0.5\\0.5\end{bmatrix}\begin{bmatrix}0.75\\0.75\\0.75\\0.75\\0.75\end{bmatrix}\begin{bmatrix}1\\1\\1\\1\\1\end{bmatrix}$	$\begin{bmatrix}0\\0\\0\\0\\0\end{bmatrix}\begin{bmatrix}0\\0\\0\\0\\1\end{bmatrix}\begin{bmatrix}0\\0\\0\\1\\1\end{bmatrix}\begin{bmatrix}0\\0\\1\\1\\1\end{bmatrix}\begin{bmatrix}0\\1\\1\\1\\1\end{bmatrix}\begin{bmatrix}1\\1\\1\\1\\1\end{bmatrix}$

Figure 2.5: PDF and QF representations of distributions constructed by location or mixture interpolation of delta distributions $\delta(0)$ and $\delta(1)$. The PDF representation is a histogram with bin centers at 0, 0.25, 0.5, 0.75, 1. Mixture interpolation is linear for PDFs and location interpolation is linear for QFs, while the opposite cases form strongly nonlinear paths.

is linear in location and scale parameters and some shape parameters have known forms. Parametric distributions, therefore, are better understood in the space of QFs; Section 2.1.4 discusses their corresponding manifolds in more detail.

To acquire some intuition about what interpolation of location parameters looks like in the PDF and CDF spaces and what interpolation of mixture parameters looks like in the QF space, consider the delta distribution. Let $D_0 \sim \delta(0)$ and $D_1 \sim \delta(1)$ be two delta distributions with nonzero probabilities at 0 and 1, respectively. A histogram $\underline{\mathrm{h}}$ that captures both distributions can be constructed with bin centers at 0.0, 0.25, 0.5, 0.75 and 1. Using $\underline{\mathrm{h}}$ and a 5 bin QF, Figure 2.5 shows the two delta distributions and two types of interpolation between them. For $\underline{\mathrm{h}}$, mixture interpolation is linear, as previously mentioned. Location interpolation for the five steps shown for $\underline{\mathrm{h}}$, however, is nonlinear. The path iteratively moves along four orthogonal paths, each a line segment with a slope of -1 defined in the plane of the corresponding, adjacent dimensions. For the QF, location interpolation is linear, and mixture interpolation forms a nonlinear path. Similar to the nonlinear path for $\underline{\mathrm{h}}$ location interpolation, QF mixture interpolation is composed of a series of orthogonal, linear segments. The path in Figure 2.5 is a particular L_1 path, where the dimensions are traversed from last to first (and is, in fact, the only legal 5 segment L_1 path).

Both types of nonlinear paths discussed above are more complicated when considering

distributions other than the delta. Several of these nonlinear submanifolds are considered numerically in Figures 2.6 - 2.10 and analytically in Section 2.1.4. We now turn our attention to interpreting distance measures in the PDF, CDF, and QF spaces.

Distance Measures

Most of this section has discussed linear interpolation and manifolds formed by considering particular types of variation. I now consider Euclidean distance, distance along these manifolds, and existing distance measures. Distances between QFs are considered first because the linearity of some of the submanifolds discussed above gives its Euclidean distance the most intuitive definition.

The examples above define Euclidean distance in the QF space for location-scale parametric families, and motivates and provides intuition for its use between arbitrary distributions. For example, between delta distributions $\delta(t_1)$ and $\delta(t_2)$ and Gaussian distributions $\mathcal{N}(\mu_1, \sigma_1^2)$ and $\mathcal{N}(\mu_2, \sigma_2^2)$, Euclidean distance between their QFs using b bins is $\sqrt{b}|t_1 - t_2|$ and $\sqrt{b}\sqrt{(\mu_1 - \mu_2)^2 + (\sigma_1 - \sigma_2)^2}$, respectively. Euclidean QF distance corresponds, up to a scale factor of $\sqrt{b}$, to a distance metric that has been studied in more general situations; it is most often called the Earth Mover's distance (EMD) or Mallows distance [LB01]. Intuitively, the EMD measures the work required to change one distribution into another by moving probability mass. Each element of probability mass in one distribution is matched with mass in the second distribution. The total work required (mass $\times$ distance) is the computed metric [RTG00]. The EMD is a metric that accounts for both the frequency and position of probability mass, making it a highly nonlinear, cross-bin distance for histogram representations. Section 2.2.1 further discusses the EMD and its definitions for multivariate distributions.

The Euclidean distances in the PDF and CDF spaces do not have the same intuitive definitions in terms of the parameters of parametric distributions, except for the linear mixture parameters. These distances are only Euclidean when their bins stay in correspondence, *i.e.*, when there are mixture changes in each bin's frequency but their locations do not shift. This does not hold for several types of variation, including location and scale, so the research into so called cross-bin distance measures such as the EMD and the diffusion distance. To examine the

Euclidean PDF and CDF distances, consider delta distributions $\delta(t_1)$ and $\delta(t_2)$. As mentioned above, the QF Euclidean distance is $\sqrt{b}|t_1 - t_2|$. The PDF distance is 0 when $t_1 = t_2$, and is its maximum, $\sqrt{2}$, otherwise. The CDF distance, given a bin width of w, is $\sqrt{\frac{|t_1 - t_2|}{w}}$, the square root of the number of bins between t_1 and t_2.

Two common distance measures based on PDFs and CDFs are the χ^2 distance and the two–sample Kolmogorov–Smirnov goodness–of–fit test statistic. Between two histograms $\underline{\mathrm{h}}$ and $\underline{\mathrm{g}}$ with b common bin locations and CDFs $\underline{\mathrm{H}}$ and $\underline{\mathrm{G}}$,

$$\begin{aligned} \chi^2(\underline{h}, \underline{g}) &= \sum_{i=1}^{b} \frac{(h_i - g_i)^2}{h_i + g_i}, \text{ and} \\ KS(\underline{H}, \underline{G}) &= \sup_{i=1}^{b}(|H_i - G_i|). \end{aligned}$$

The Kolmogorov–Smirnov test statistic is therefore the L_∞ CDF norm. The χ^2 distance is a simple linear scaling of the Euclidean PDF distance, similar to the CDF transformation, except it is specific to $\underline{\mathrm{h}}$ and $\underline{\mathrm{g}}$. While the CDF scaling does cumulative integration, which passes information (horizontally) between bins of the same distribution, the χ^2 distance normalizes bin differences by their frequency, which passes information (vertically) between the two distributions.

Analyzing such distance measures, or the Euclidean PDF, CDF, and QF distances, between distributions is difficult. Therefore, Figures 2.6 - 2.10 numerically consider the Gaussian, a mixture of two Gaussians, the gamma, the beta, and the Weibull distributions, respectively. For each, two parameters of the distribution are varied. In the top left of each figure, the four corners of this sampled parameter space, a - d, are shown as PDFs. The first parameter is varied from a to b and c to d. The second parameter is varied from a to c and b to d. The Euclidean and manifold distances in the PDF, CDF, and QF spaces are given along with the χ^2 distance and Kolmogorov–Smirnov test statistic. The manifold distances follow the geodesic paths determined by interpolating the parameters. All of the distances are computed from a, one corner of the sampled parameter space, to the rest of the sampled space. Each sampled parameter space is displayed as a two-dimensional submanifold in the PDF, CDF, and QF spaces using principal component analysis (PCA), which is discussed more in Sections 2.2.1

and 2.3. Each submanifold is displayed in the first three principal directions; this supplies the most possible information about the shape of the submanifold. To give a notion of the linearity of the submanifolds, the relative cumulative eigenvalues are also displayed for each space.

Figure 2.6 shows the Gaussian distribution. The Gaussian distribution is a location-scale family so it forms a linear submanifold in the QF space. Its linearity is shown by the submanifold being flat, by the cumulative eigenvalues reaching 1 at 2 modes, and by the Euclidean and manifold QF distances being identical. The nonlinearity in the PDF space is also evident. The PDF Euclidean and manifold distances differ. Specifically, when interpolating from Gaussian a to Gaussian b the Euclidean distance levels off while the manifold distance does not. This effect is shown in the manifold by the curved arc formed by that path. The manifold shows that larger sigmas make all Gaussians relatively similar while sigmas that are small relative to the mean difference makes all Gaussians equally dissimilar.

Figures 2.6 - 2.10 show that the χ^2 distance and Kolmogorov–Smirnov test statistic are usually similar to Euclidean PDF distance. For all five distributions, the figures also show that feature space for the PDF is more nonlinear than for the CDF or QF; the sum of the relative cumulative eigenvalues is always the lowest for PDFs. Also, for four of the five distributions, the exception being the case where a mixture parameter was modeled, the QF supplies the most compact, and hence linear, representation. This linearity has been discussed for location-scale distributions such as the Gaussian. In Figure 2.7 the mixture of two Gaussians distribution has a location parameter; in Figures 2.8 and 2.10 the gamma and Weibull distributions have a scale parameter. The Weibull distribution is discussed more in the next section. Figure 2.7 shows, as expected, that the mixture parameter in the mixture of two Gaussians distribution is linear for PDFs and CDFs but not QFs. Figure 2.8 shows that in the QF space, the scale and shape parameters of the gamma distribution form an approximately flat and convex, though skewed, submanifold. Figure 2.9 shows that the beta distribution forms similar, nonconvex submanifolds in all three spaces as viewed in their corresponding first 2 principal directions. While similarly shaped, the submanifold is convex near d in the PDF and CDF spaces and near a in the QF space.

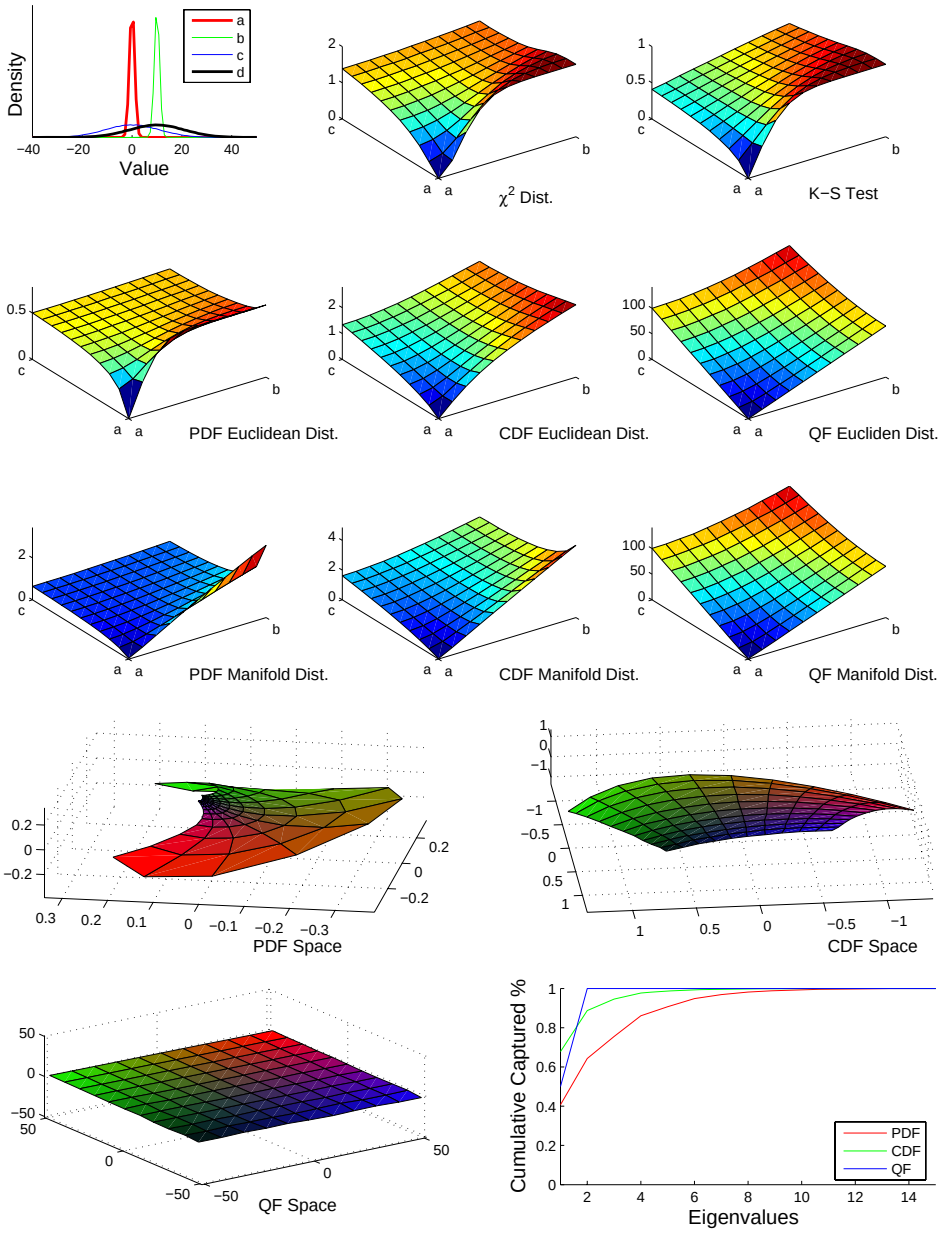


Figure 2.6: Gaussian distributions $\mathcal{N}(\mu, \sigma^2)$. The parameter space samples μ from 0 to 10 in the first dimension and σ from 1 to 11 in the second dimension. The manifold in the QF space is flat and Euclidean QF distance equally penalizes mean and standard deviation change.

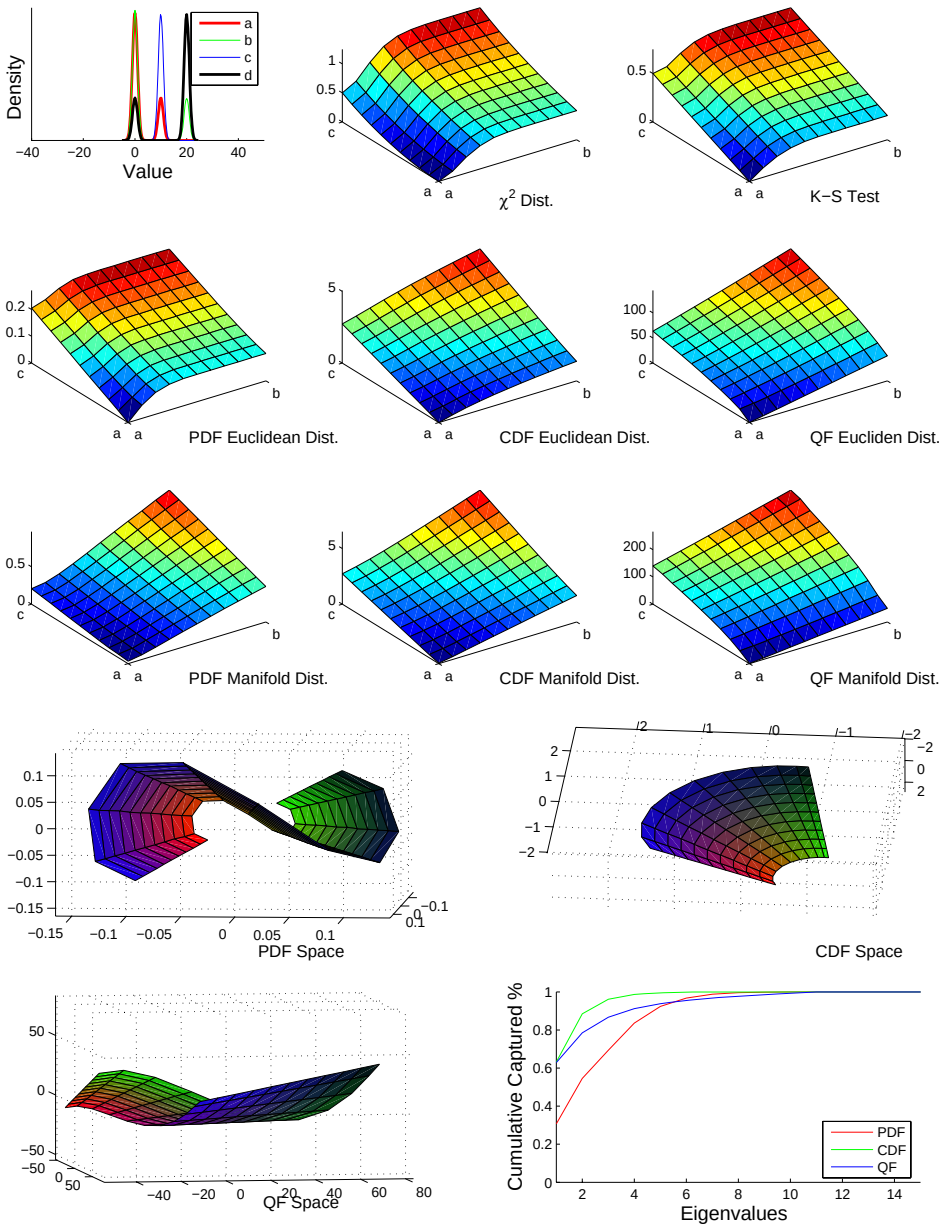


Figure 2.7: Two Gaussian mixture distributions $w*\mathcal{N}(0,1)+(1-w)*\mathcal{N}(\mu,1)$. The parameter space samples μ from 10 to 20 in the first dimension and w from 0.75 to 0.25 in the second dimension. The mixture parameter is nonlinear for QFs. The CDF is the most efficient representation for this sampling.

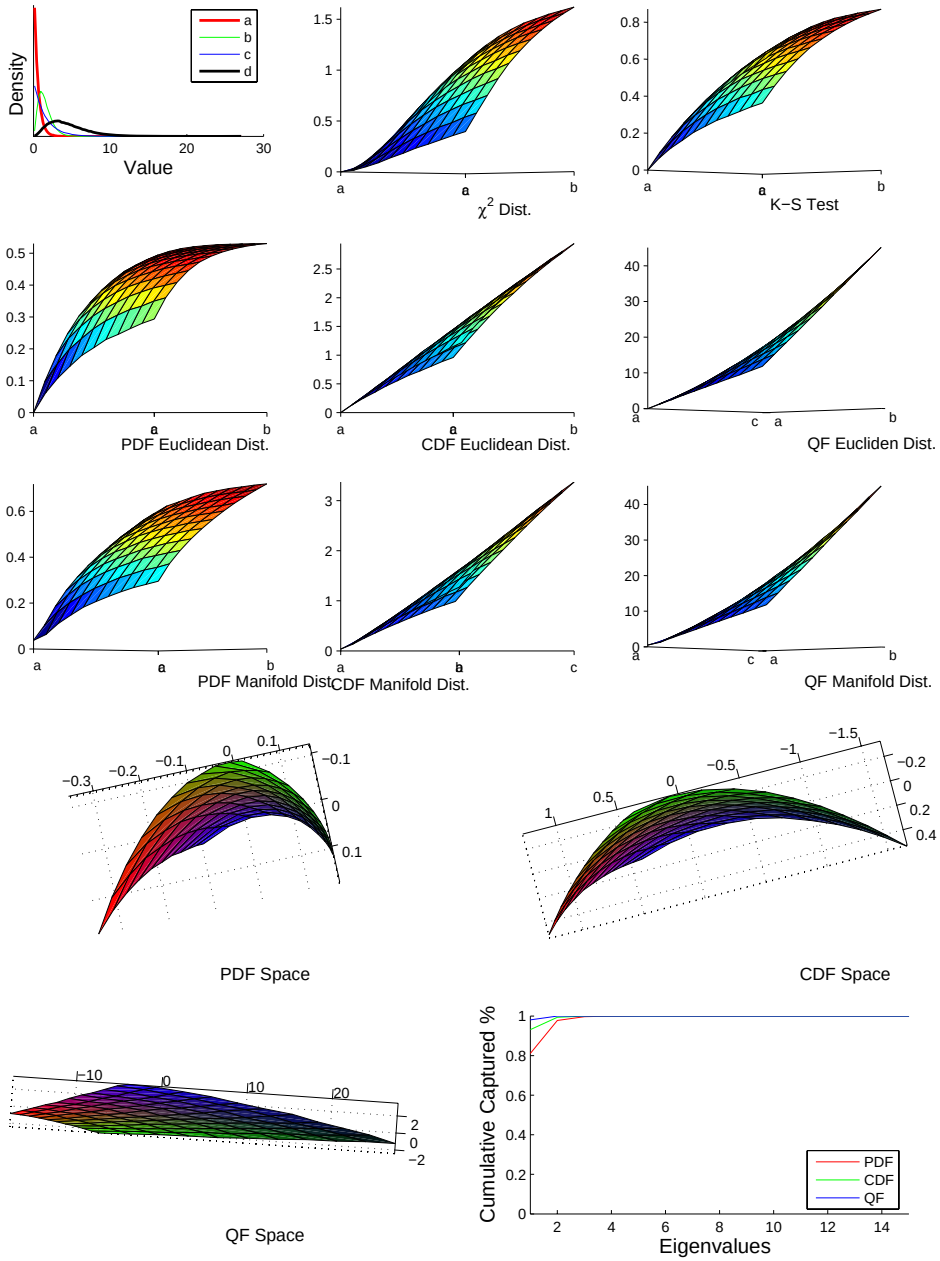


Figure 2.8: Gamma distributions $\Gamma(k, \theta)$. The parameter space samples k from 1 to 3 in the first dimension and θ from 0.5 to 1.5 in the second dimension. θ is a scale parameter so is linear for QFs. Even though k is shape parameter, the manifold is approximately flat and convex in the QF space.

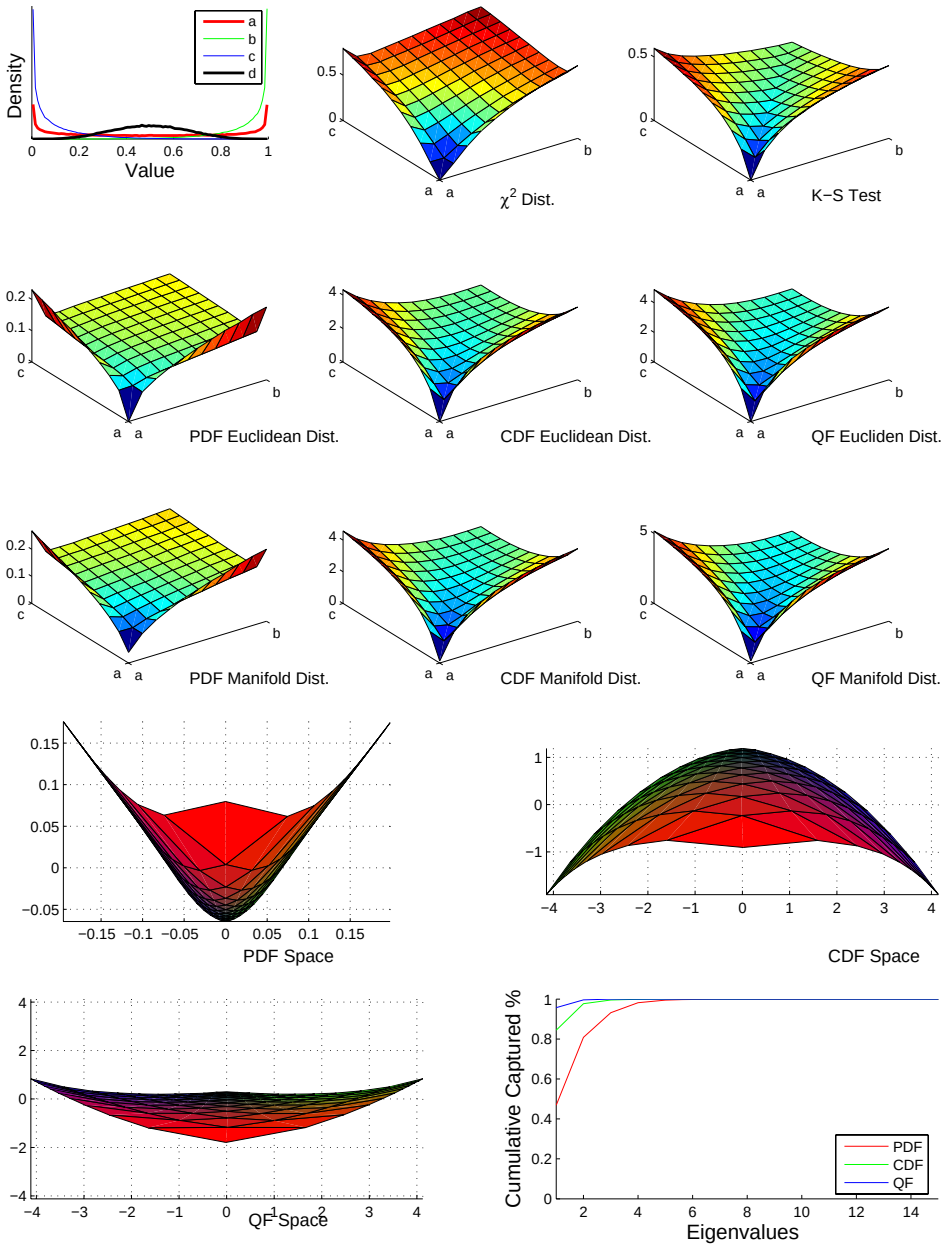


Figure 2.9: Beta distributions $\mathcal{B}(\alpha, \beta)$. The parameter space samples α from 0.5 to 5 in the first dimension and β from 0.5 to 5 in the second dimension. The three manifolds are similar as displayed in their first two principal directions, though the PDF and CDF spaces have additional twisting in the third direction.

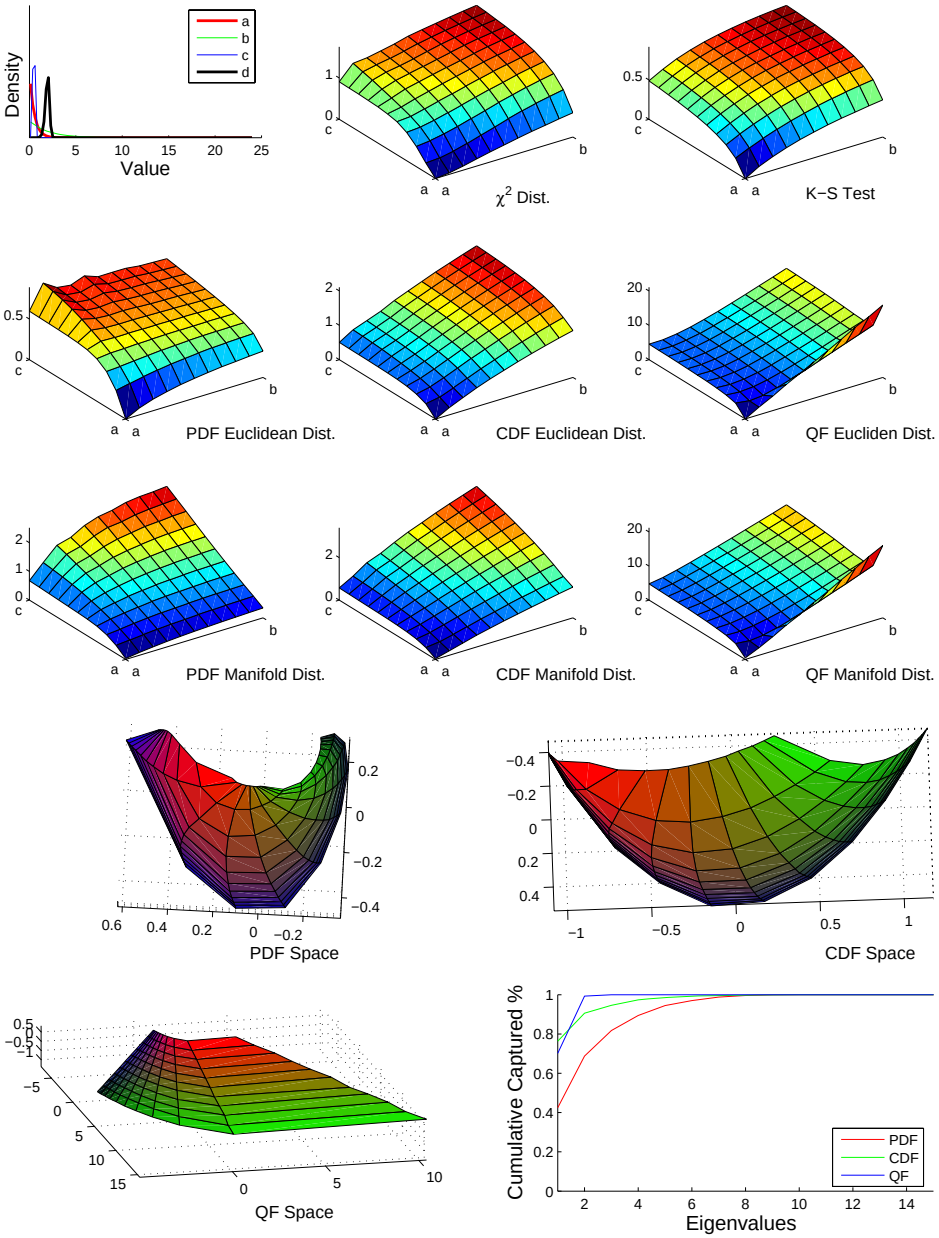


Figure 2.10: Weibull distributions $\mathcal{W}(\lambda, k)$. The parameter space samples λ from 0.5 to 2 in the first dimension and k from 1 to 11 in the second dimension. The PDF graph shows the long right tail for large λ; the QF space is sensitive to this. The QF is linear in λ, a scale parameter, and exponential in k, a shape parameter, as is discussed in Section 2.1.4.

This section discussed the analytic properties of PDFs, CDFs, and QFs and numerically considered some of the submanifolds of common parametric distributions in these spaces. QFs were shown to more compactly represent both a single distribution and a variety of common distribution families. The next section analytically considers the construction of submanifolds corresponding to common parametric families for QFs. No further analysis of PDF and CDF representations is given in this dissertation.

2.1.4 The Space of Quantile Functions

The space of quantile functions can be understood geometrically in several ways. This section builds this geometric intuition by considering several additional properties of QFs, including the space's constraints, a small number of QF bins, the various L_p norms, and the construction of submanifolds corresponding to several common parametric families.

Discrete quantile functions are constrained to be nondecreasing through its dimensions, *i.e.*, a b bin QF $\underline{Q}$ has the constraint $Q_1 \leq Q_2 \leq \ldots \leq Q_b$. Since this constraint is linear, the valid submanifold is convex. Convexity implies that QF averages and interpolation will always be valid but that extrapolation can lead outside the valid submanifold. The submanifold is not, however, a subspace of $\mathcal{R}^b$ nor is it a vector space. Valid QFs do not form a subspace because it is not closed under multiplication; multiplication by negative numbers produce invalid QFs. However, the addition of any two QFs produces valid QFs and an additive identity exists, though additive inverses in general do not. Other representations of probability distributions also do not form vector spaces, including most parametric families and discrete PDF and CDF representations.

The valid submanifold of QFs has a sharp boundary at $Q_1 = Q_2 = \ldots = Q_b$. All points that satisfy this constraint exist on the $1 \times b$ vector of ones, $\underline{1}$, which corresponds to the submanifold of delta distributions. In particular, the delta distribution $\delta(t)$ has the b bin quantile function $t\underline{1}$. As mentioned in Section 2.1.3, changing the mean and standard deviation of a distribution forms a linear submanifold that corresponds to the affine transformation $\alpha\underline{Q} + c\underline{1}$. The delta distribution consists simply of a location, or mean, change.

Location-scale distribution families include the Gaussian, exponential, uniform, double

exponential, Rayleigh, and Fisher-Tippett. Each location-scale family exists on a linear submanifold that intersects and ends at the delta distribution as the scale parameter goes to zero. A basis of each submanifold can be analytically specified by two orthogonal vectors. The first vector, $\underline{1}$, which corresponds to the mean of the distribution, is common to all of the families. Moving along this vector changes the distribution's mean, where $c\underline{1}$ corresponds to a mean of c. The second vector corresponds to the shape of the distribution and is specific to each family. It often corresponds to a zero mean and unit standard deviation distribution, to make it orthogonal to $\underline{1}$ and of unit length, respectively. Moving along this vector changes the distribution's standard deviation, where $\alpha\underline{Q}$ corresponds to a distribution with a standard deviation of α, when $\underline{Q}$ is zero mean and unit standard deviation. The standard deviation of general QFs is discussed later in this section.

Figure 2.11 gives such an orthogonal basis for six distribution families mentioned above that have location, scale, or location and scale parameters. For each distribution, the figure defines the PDF f (if convenient), the CDF F, the QF Q, and the discrete QF $\underline{Q}$. $\underline{Q}$ is given in terms of $\underline{1}$ and the distribution family's base distribution, and in terms of $\underline{1}$ and an orthogonal, unit vector. The orthogonal, unit vector is constructed by either converting the family's base distribution to a zero mean, unit standard deviation distribution or by directly choosing such a distribution from the family. For example, the two-dimensional, linear submanifold of Gaussian distributions can be constructed from the orthogonal vectors $\underline{1}$ and the QF corresponding to $\mathcal{N}(0,1)$.

Several other common distributions contain a scale parameter, a shape parameter, and an optional location parameter. One such distribution is the Weibull. The QF of the Weibull function has a closed analytic form; it is given in Figure 2.12. Figure 2.12 shows that the Weibull's QF is exponential in the shape parameter. As shown in Figure 2.10, this leads to a smooth and fairly flat submanifold. The gamma distribution also contains a scale and a shape parameter. The QF for the gamma distribution is not easy to express. However, Figure 2.8 shows the extremely smooth and flat submanifold numerically found for a portion of the parameter space. The beta distribution also does not have an easily expressed QF. Figure 2.9 shows the fairly linear, though distorted, submanifold numerically found for a portion of the

Delta distributions, $\delta(t)$		
	$F(x)$	$=$ 0 if $x < t$, 1 if $x \geq t$
	$Q(y)$	$=$ t
	$\underline{Q}_{\delta(t)}$	$=$ $t\underline{1} = t\underline{Q}_{\delta(1)}$
Gaussian distributions, $\mathcal{N}(\mu, \sigma)$		
	$f(x)$	$=$ $\frac{1}{\sigma\sqrt{2\pi}}\mathrm{e}^{\frac{(x-\mu)^2}{-2\sigma^2}}$
	$F(x)$	$=$ $\frac{1}{2}(1 + \mathrm{erf}(\frac{x-\mu}{\sigma\sqrt{2}})$
	$Q(y)$	$=$ $\mu + \sigma\sqrt{2}\mathrm{erf}^{-1}(2y-1)$
	$\underline{Q}_{\mathcal{N}(\mu,\sigma)}$	$=$ $\mu\underline{1} + \sigma\underline{Q}_{\mathcal{N}(0,1)}$
Uniform distributions, $\mathcal{U}(a, b)$		
	$f(x)$	$=$ $\frac{1}{b-a}$ if $a \leq x \leq b$, 0 otherwise
	$F(x)$	$=$ 0 if $x < a$, $\frac{x-a}{b-a}$ if $a \leq x \leq b$, 1 if $x > b$
	$Q(y)$	$=$ $(1-y)a + yb = a + (b-a)y$
	$\underline{Q}_{\mathcal{U}(a,b)}$	$=$ $a\underline{1} + (b-a)\underline{Q}_{\mathcal{U}(0,1)}$ (nonunit, nonorthogonal)
	$\underline{Q}_{\mathcal{U}(a,b)}$	$=$ $\frac{1}{2}(a+b)\underline{1} + \frac{1}{\sqrt{12}}(b-a)\underline{Q}_{\mathcal{U}(-\sqrt{3},\sqrt{3})}$
Exponential distributions, $\mathrm{Exp}(\lambda)$		
	$f(x)$	$=$ $\frac{1}{\lambda}\mathrm{e}^{-x/\lambda}$ if $x \geq 0$, 0 otherwise
	$F(x)$	$=$ $1 - \mathrm{e}^{-x/\lambda}$ if $x \geq 0$, 0 otherwise
	$Q(y)$	$=$ $-\lambda\ln(1-y)$
	$\underline{Q}_{\mathrm{Exp}(\lambda)}$	$=$ $\lambda\underline{Q}_{\mathrm{Exp}(1)} = \lambda\underline{1} + \lambda(\underline{Q}_{\mathrm{Exp}(1)} - \underline{1})$
Fisher-Tippett distributions, $\mathcal{FT}(\mu, \beta)$		
	$f(x)$	$=$ $\mathrm{e}^{-\frac{x-\mu}{\beta}}\mathrm{e}^{-\mathrm{e}^{-\frac{x-\mu}{\beta}}}$
	$F(x)$	$=$ $\mathrm{e}^{-\mathrm{e}^{-\frac{x-\mu}{\beta}}}$
	$Q(y)$	$=$ $\mu - \beta\ln(-\ln(y))$
	$\underline{Q}_{\mathcal{FT}(\mu,\beta)}$	$=$ $\mu\underline{1} + \beta\underline{Q}_{\mathcal{FT}(0,1)}$ (nonunit, nonorthogonal)
	$\underline{Q}_{\mathcal{FT}(\mu,\beta)}$	$=$ $(\mu + \beta\gamma)\underline{1} + \frac{\pi}{\sqrt{6}}\beta\underline{Q}_{\mathcal{FT}(-\gamma\sqrt{6}/\pi,\sqrt{6}/\pi)}$
Rayleigh distributions, $\mathcal{R}(a)$		
	$f(x)$	$=$ $\frac{x}{\sigma^2}\mathrm{e}^{-x^2/2\sigma^2}$
	$F(x)$	$=$ $1 - \mathrm{e}^{-x^2/2\sigma^2}$
	$Q(y)$	$=$ $\sigma\sqrt{-2\log(1-y)}$
	$\underline{Q}_{\mathcal{R}(\sigma)}$	$=$ $\sigma\underline{Q}_{\mathcal{R}(1)}$
	$\underline{Q}_{\mathcal{R}(\sigma)}$	$=$ $\sigma\sqrt{\pi/2}\underline{1} + \sqrt{\frac{4-\pi}{2}}\sigma\underline{Q}_{\mathcal{R}(\sqrt{\frac{2}{4-\pi}})}$

Figure 2.11: The PDF f, CDF F, QF Q, and discrete QF $\underline{Q}$ of several distribution families with location, scale, or location and scale parameters. $\underline{Q}$ is given in its most convenient form and in terms of an orthogonal basis composed of $\underline{1}$ and a unit vector. The second vector corresponds to a zero mean, unit standard deviation distribution. The scalars in front of the orthogonal vectors represent mean and standard deviation.

Weibull distributions, $\mathcal{W}(\lambda, k)$		
$f(x)$	$=$	$\frac{k}{\lambda}(\frac{x}{\lambda})^{k-1}\mathrm{e}^{-(x/\lambda)^k}$
$F(x)$	$=$	$1-\mathrm{e}^{-(x/\lambda)^k}$
$Q(x)$	$=$	$\lambda(-\ln(1-y))^{1/k}$
$\underline{Q}_{\mathcal{W}(\lambda,k)}$	$=$	$\lambda\underline{Q}^{1/k}_{\mathcal{W}(1,1)}$

Figure 2.12: The PDF f, CDF F, QF Q, and discrete QF $\underline{Q}$ of the Weibull distribution. In the QF space, the scale parameter is linear and the shape parameter is exponential.

parameter space. The general Jensen and exponential distribution families also do not have simple forms to their QFs. This dissertation supplies no intuition and reports no further on these distribution families.

There are also standard, known relations among distribution families that can be considered geometrically in the QF space. For example, as mentioned in Section 2.1.1, the Weibull distribution generalizes the exponential and Rayleigh distributions. Both the exponential and Rayleigh have a scale parameter and have different fixed values for the Weibull's shape parameter. Specifically, $\mathrm{Exp}(\lambda) \sim \mathcal{W}(\lambda, 1)$ and $\mathcal{R}(\beta) \sim \mathcal{W}(\sqrt{2}\beta, 2)$. Because the shape parameter is fixed and the scale parameter is varied, the exponential and Rayleigh are both straight lines on different parts of the Weibull's curved submanifold. The gamma distribution generalizes the exponential and chi-squared distributions. Specifically, $\mathrm{Exp}(\lambda) \sim \gamma(1, \lambda)$ and $\chi^2(k) \sim \gamma(k/2, 2)$. Therefore, on the submanifold of gamma distributions, the exponential distribution follows a line and the chi-squared distribution follows a curved path (which intersect at $\gamma(1, 2)$). Since both the gamma and the Weibull include the exponential distribution, these two two-dimensional, curved manifolds intersect along the line of exponentials. There are several other relationships between distributions, such as the beta distribution including the unit uniform distribution, that are given in basic statistics sources [Ros02].

There is a strong relationship in the QF space between L_p vector norms and moments of the corresponding probability distributions. The first moment, or mean, of a distribution with a b bin QF $\underline{Q}$ is the L_1 vector distance between the origin and $\underline{Q}$, divided by b. This is identical

to projecting $\underline{Q}$ onto the vector of ones and dividing by b: $\underline{Q} \cdot \underline{1}/b$. In general,

$$\begin{aligned} L_p(\underline{Q}, \underline{R}) &= (\sum_{i=1}^{b} |Q_i - R_i|^p)^{1/p}, \text{ and} \\ \mu'_p(\underline{Q}) &= \frac{1}{b} \sum_{i=1}^{b} Q_i^p, \end{aligned}$$

where μ'_p is the p^{th} raw moment of Q. Therefore, $\mu'_p = \frac{1}{b}(L_p(\underline{0}, \underline{Q}))^p$. Central moments, μ_p, can also be easily computed, where

$$\mu_p = \frac{1}{b}(L_p(\mu_1 \underline{1}, \underline{Q}))^p.$$

Since central moments are computed with respect to the mean of the distribution, it is convenient to consider only zero mean distributions. The QF space of zero mean distributions can be constructed by projecting out the dimension corresponding to $\underline{1}$. Let $\underline{Q'}$ be the zero mean distribution corresponding to $\underline{Q}$. Then $\underline{Q'} = \underline{Q} - \mu_1 \underline{1} = \underline{Q} - \frac{\underline{Q} \cdot \underline{1}}{b} \underline{1}$. The standard deviation of $\underline{Q'}$ is now equivalent, up to a constant scale factor, to the Euclidean distance between $\underline{0}$ and $\underline{Q'}$:

$$\sqrt{\mu_2} = \sqrt{1/b} L_2(\underline{0}, \underline{Q'}) = \sqrt{\underline{Q'} \cdot \underline{Q'}/b}.$$

For nonzero mean distributions,

$$\sqrt{\mu_2} = \frac{1}{b} \sqrt{b(\underline{Q} \cdot \underline{Q}) - (\underline{Q} \cdot \underline{1})^2} = \sqrt{(\underline{Q} \cdot \underline{Q})/b - (\mu'_1/b)^2}.$$

In the QF space of zero mean distributions, the origin represents the $\delta(0)$ distribution. Concentric hyperspheres about the origin correspond to distributions of the same standard deviation, where a radius of r corresponds to a standard deviation of $r/\sqrt{b}$. Location-scale families exist solely on the vectors orthogonal to $\underline{1}$ given in Figure 2.11. Each vector correspond to the shape of the distribution family and is unchanged in this space. Therefore, location-scale families exist on vectors that radiate out from the origin. Normalized central moments can be computed in this space by first changing the distribution to have a standard deviation of 1, by scaling $\underline{Q'}$ to the hypersphere of radius $\sqrt{b}$.

Section 2.1.2 mentioned how $\underline{Q}$ is actually the piecewise integration of Q multiplied by b. Not including the multiplication by b would simplify some of the distance equations. For example, the L_1 distance would then be exactly equal to the distribution's mean. This definition of $\underline{Q}$ was also used in Section 2.1.2 to understand what $\underline{Q}$ represents when $b = 1$ and $b = 2$. If $\underline{Q}$ was actually an estimate of Q, one bin would represent the distribution's median. Since $\underline{Q}$ is the average of the bin, which in this case is the whole distribution, it is instead the mean. When $b = 2$, $\underline{Q}$ is linearly equivalent to mean and standard deviation, with $\mu = \frac{1}{2}(Q_1 + Q_2), \sigma = \sqrt{\frac{1}{2}(Q_1 - \mu)^2 + \frac{1}{2}(Q_2 - \mu)^2} = \frac{1}{2}(Q_2 - Q_1)$. μ and σ correspond to the vectors $\underline{1}$ and $[-1\ 1]^T$. When $b = 3$, symmetric distributions follow the linear constraint that $Q_2 = \frac{1}{2}(Q_1 + Q_3)$.

Additional QF Properties and Relations with Random Variables

Many operations on a distribution have known effects on both the distribution's QF and on a random variable that follows the distribution. Let X follow a distribution with QF Q. If f is a nondecreasing, deterministic function, then $f(X)$ has the QF of f composed with Q: $f(Q(y))$ or $f \circ Q$. If f is a decreasing, deterministic function, then $f(X)$ has the QF $f(Q(1 - y))$.

The addition of two independent random variables X and Y corresponds to the convolution of their PDFs. The equivalent operation to their corresponding discrete QFs, $\underline{Q}$ and $\underline{R}$, is slightly more complicated. Let $\underline{Q}$ and $\underline{R}$ be b bin QFs. Then the b bin QF of $X + Y$ can be constructed by first considering the set of points formed by taking $Q_i + \underline{R}$, $i = 1, \ldots, b$. The resulting set of points are simulated samples from the distribution corresponding to $X + Y$. Its discrete QF can now be constructed identically to the QF estimate used in Section 2.1.2, which involves sorting the b^2 samples and then averaging every b adjacent values.

Some operations on discrete QFs are based on the notion of the maximal correlation between two independent distributions. For example, given a discrete QF $\underline{Q}$, one might want to know the Gaussian distribution that $\underline{Q}$ has the minimum Euclidean distance to. This is accomplished by projecting $\underline{Q}$ onto the submanifold of Gaussian distributions. If $\underline{Q}$ has a mean μ and standard deviation σ, its projection will correspond the the Gaussian $\mathcal{N}(\mu', \sigma')$,

where $\mu' = \mu$, $\sigma' = \sigma \cdot \text{max\_correlation}(\underline{Q}, \underline{Q}_{\mathcal{N}(0,1)})$, and max_correlation is defined between two independent probability distributions q and r as

$$\text{max\_correlation}(q, r) = \max_f \{\text{correlation}_f(X, Y) : (X, Y) \sim f, X \sim q, Y \sim r\}.$$

The notion of maximal correlation is also related to the addition of quantile functions. Let μ_i and σ_i be the respective means and standard deviations of $\underline{Q}_i, i = 1, 2, 3$. If $\underline{Q}_3 = \underline{Q}_1 + \underline{Q}_2$, $\mu_3 = \mu_1 + \mu_2$. If $\underline{Q}_1$ and $\underline{Q}_2$ are in the same location-scale family, $\sigma_3^2 = (\sigma_1 + \sigma_2)^2, \sigma_3 = \sigma_1 + \sigma_2$. In general,

$$\sigma_3^2 = \sigma_1^2 + \sigma_2^2 + \frac{2}{b}\sum_{i=1}^{b}(\underline{Q}_{1,i} - \mu_1)(\underline{Q}_{2,i} - \mu_2) = \sigma_1^2 + \sigma_2^2 + 2\sigma_1\sigma_2 \cdot \text{max\_correlation}(\underline{Q}_1, \underline{Q}_2).$$

In the QF space of zero mean distributions, the maximal correlation between two QFs $\underline{Q'}$ and $\underline{R'}$ is the cosine of the angle between the two points at the origin. If θ is this angle, $\text{max\_correlation}(\underline{Q'}, \underline{R'}) = \cos(\theta) = (\underline{Q'} \cdot \underline{R'})/(\sqrt{\underline{Q'} \cdot \underline{Q'}}\sqrt{\underline{R'} \cdot \underline{R'}})$. If $\underline{Q'}$ and $\underline{R'}$ have unit standard deviations, this simplifies to $\frac{1}{b}(\underline{Q'} \cdot \underline{R'})$.

2.1.5 Summary

Section 2.1 discussed representations of univariate probability distributions in the context of finding a compact representation of a given population of distributions. Compactness was defined in terms of the linearity of the submanifold formed by the population and the resulting low number of parameters needed to express the variability in the population. In this context, parametric families are ideal. However, Section 2.1 primarily discussed the options when an appropriate family does not exist. Then one must choose between the three non-parametric options, discrete PDFs, CDFs, or QFs. Section 2.1.3 discussed how the compactness of non-parametric representations can be studied in general in terms of the various parametric families. Sections 2.1.3 - 2.1.4 used the relationship between QFs and parametric families to provide an intuition to the types of populations for which QFs will be compact. This led to a geometric intuition of the space of QFs, the first contribution of this dissertation mentioned in Section 1.2.

Specifically, the relationship between QFs and parametric families were expressed in several ways. Two common parameters of distribution families, mean and standard deviation, were shown for QFs to correspond to linear variation. Euclidean QF distance was shown to correspond to a known metric, the EMD, which has a simplified form for location-scale families. Submanifolds formed by parametric families were graphed numerically in Figures 2.6 - 2.10 and analytically constructed in Section 2.1.4. A geometric intuition of the space of QFs was given by considering low dimensional QF spaces and the interpretation of location, scale, and the other distribution moments as L_p vector norms.

2.2 Quantile Function Generalizations

Section 2.1 gave a detailed analysis of quantile functions, which are only defined for univariate distributions. This section considers methods based on quantile functions for representing multivariate, conditional, and mixture distributions. The goal is to produce representations that allow easy estimation of their likelihood from population samples, which is discussed in Section 2.3. This is insured by producing QF based representations that are natural in two main senses. First, Euclidean distance is maintained as an invariant metric that is an efficient approximation of the Earth Mover's distance (EMD). The EMD is one possible generalization of Euclidean QF distance and is discussed in Section 2.2.1. Second, I wish to understand the types of variation that form linear subspaces of the representation and, in particular, to have the representation maintain the linear subspaces of QFs.

Section 2.2.1 considers multivariate distributions. Section 2.2.2 considers conditional distributions for use in representing multivariate distributions. Section 2.2.3 considers univariate distributions composed of a mixture of underlying distributions. Section 2.3 then constructs models that estimate the probability of these generalized distributions for use in classification and segmentation.

2.2.1 Multivariate Distributions

Many interesting applications, such as the texture classification tasks considered in Chapter 3, require the representation of multivariate probability distributions. Therefore, this section discusses a representation of multivariate distributions composed of several quantile functions. For univariate distributions, the QF provides a representation in which Euclidean distance and linear interpolation are understood. Also, given a population of distributions, the QF estimates distributions accurately and efficiently with respect to the number of QF bins. For multivariate distributions, it is difficult but still crucial to construct representations with these desirable properties.

I represent a multivariate distribution using multiple one-dimensional projections of the distribution. A single vector representation is obtained for a multivariate distribution by concatenating the QFs of each projection. A key issue, discussed later in this section, is the choice of projection directions. First, the Earth Mover's distance (EMD) is defined, the relationship between the EMD and the Euclidean distance of this representation is discussed, and linear interpolation of this representation is discussed. Briefly, Euclidean distance between two such vectors is the L_2 sum of each projection's Euclidean QF distance. This can be understood as a fast approximation and lower bound to the EMD between the original multivariate distributions.

The Earth Mover's and Mallows Distances

In Section 2.1.3 the EMD was only defined for univariate distributions. As in the univariate case, for multivariate distributions the EMD is equivalent to the Mallows, L_p-Wasserstein, and Kantorovich distances [Lev02]. The Mallows distance between independent probability distributions q and r is

$$M_p(q,r) = \min_f \{(\mathrm{E}_f \|X - Y\|^p)^{1/p} : (X,Y) \sim f, X \sim q, Y \sim r\},$$

the expected L_p distance between random variables $X \sim q$ and $Y \sim r$ assuming q and r are maximally related. Throughout this dissertation I choose to use the L_2 distance. The

motivation given here for using the Mallows distance and the EMD leaves as arbitrary the choice in the underlying distance measure. However, Section 2.1 focused on the advantages of forming a Euclidean space, as made possible by the relationship of the quantile function to the L_2 EMD.

The EMD is computed between two point sets $\underline{x}$ and $\underline{y}$ with m and n points and corresponding weights $\underline{w}^x$ and $\underline{w}^y$. The EMD measures the minimum total work (mass $\times$ distance) required to move the set with a smaller total mass so that it completely overlaps with the larger set [RTG00]. Between each pair of points i in $\underline{x}$ and j in $\underline{y}$, the EMD requires a distance, d_{ij}, and the optimal correspondence, or flow, f_{ij}. The EMD is a metric when d_{ij} is a metric and the two sets have equal mass. When the total weight of each set is normalized such that $\sum_i w_i^x = 1$ and $\sum_i w_i^y = 1$, these point sets can be viewed as discrete estimates of probability distributions. Given normalized point sets and an underlying distance d_{ij} of $\|x_i - y_j\|^p$, the EMD is equivalent to the Mallows distance, except that the EMD does not take the pth root of the distance [LB01]. In this case the EMD is defined as

$$\mathrm{EMD}(\underline{x}, \underline{y}) = \min_f \sum_{i=1}^{m} \sum_{j=i}^{n} f_{ij} d_{ij}$$

subject to the constraints:

$$\begin{aligned} f_{ij} &\geq 0 \qquad 1 \leq i \leq m, 1 \leq j \leq n \\ \sum_{j=1}^{n} f_{ij} &= w_i^x \qquad 1 \leq i \leq m \\ \sum_{i=1}^{m} f_{ij} &= w_j^y \qquad 1 \leq j \leq n \\ \sum_{i=1}^{m} \sum_{j=i}^{n} f_{ij} &= 1 \end{aligned}$$

The EMD is the solution to the well-known transportation, or Monge–Kantorovich, problem [Hit41, Rac84]. There are two situations which simplify the EMD. First, if all of the points have equal mass, *i.e.*, if $m = n$ and $w_i^x = w_i^y = 1/n, i = 1, \ldots, n$, a 1-1 correspondence will be found. Second, if the points are in a one-dimensional space, the correspondence has a solution

given through sorting. The quantile function framework discussed in Section 2.1 has both of these simplifications, yielding the reduction of the L_2 EMD to Euclidean distance.

The EMD has been successfully used to compare multivariate distributions represented as histograms [RTG00]. While successful, this approach has weaknesses in three respects. First, it is limited computationally by the optimization required to compute the EMD. Second, understanding the variability in a population is limited by only having a distance metric, instead of a Euclidean space. Third, histogram representations tend to be noncompact for object populations. There is work into EMD approximations that addresses the first weakness, but, similar to the EMD, these approximations are nonlinear distance measures for histograms [GD04].

The multivariate distribution representation presented in this section addresses all three of these issues. The first and second weaknesses are solved by having a Euclidean space, which is discussed below in terms of Euclidean distance and linear interpolation. Euclidean distance resolves the first weakness because it is a fast approximation and lower bound to the L_2 EMD. Thus, it can be used instead of the EMD itself. The ability of a Euclidean space to resolve the second weakness is the topic of Section 2.3. The third weakness is addressed by using a QF based representation, which tends to be compact, as discussed in Section 2.1 and again briefly below.

Euclidean Distance

As mentioned above, I represent a multivariate distribution using QFs estimated from multiple one-dimensional projections. To choose the projection directions, I use principal component analysis (PCA) [Jol86]. PCA is a standard technique for modeling linear variation in data with origins dating back to Hotelling [Hot33] and Pearson [Pea01]. PCA has the desirable property that for an orthogonal basis, the directions maximally capture correlations among the distribution variables. This implies that the projection coefficients are minimally correlated. Therefore, when independent QFs are built for each projection, the least amount of information about the joint distribution is lost. For separable distributions, such as the multivariate Gaussian, being uncorrelated implies independence, and no information is lost.

This is the most accurate representation of the multivariate distribution possible based on orthogonal projections. Thus, for a population of multivariate distribution estimates given as samples, I find common projection directions for the distributions using PCA on samples pooled across the distributions.

The following argument shows that the Euclidean distance of this representation is an approximation to and a lower bound of the L_2 EMD between the multivariate distributions. Let X and Y be two d-variate distributions with n samples q_j^i and r_j^i, $i = 1, \ldots, n, j = 1 \ldots, d$. Let Q_j^i and R_j^i be d n-bin quantile functions corresponding to d projections of q and r. Each $\underline{Q_j}$ and $\underline{R_j}$ corresponds to a sorted version of $\underline{q_j}$ and $\underline{r_j}$, where $\underline{Q_j}$, $\underline{R_j}$, $\underline{q_j}$, and $\underline{r_j}$ are the respective tuples over i. Let L_2 denote Euclidean distance. The EMD between X and Y, or more appropriately, EMD(q, r), is equal to $L_2(q, r')$, where r' is an optimal reordering of the samples r^i. $L_2(Q, R)$ will always be less than or equal to $L_2(q, r')$, since Q and R are computed using d optimal reorderings for each projection while r' is computed using a single reordering.

$L_2(Q, R)$ can be considered in two equivalent ways. First, each Q^i and R^i can be considered as samples from X and Y. Q^i and R^i correspond to samples when the d sortings are equivalent to the single reordering of r'. There are several correlation structures of X and Y when this would hold. One example is when the d projections are maximally correlated for both X and Y. Second, $L_2(Q, R)$ can be considered as a more accurate estimate of EMD(q, r) when the projections are independent. This is exactly the assumption made when constructing this representation. When the d projections are independent, any mismatch between the d sortings and the single reordering of r' can be viewed as sampling error. $L_2(Q, R)$ is less than $L_2(q, r')$ because the two sets of d univariate distributions are estimated more accurately than their corresponding multivariate distribution. Their difference is sampling error that is removed by making the assumption X and Y are independent.

Linear Interpolation

Linear interpolation also has some understood properties. Changing a distribution's mean or scaling along the projection directions forms a linear path. Variation in the population that corresponds to a rotation of the projection directions also has a known effect. However,

rotational population variation forms a nonlinear manifold. Assume that PCA finds directions for a distribution such that it is appropriate to consider the projected coefficients independent. The effect of a rotation can then be considered using independent random variables X and Y. Let X' be a random variable corresponding to a vector in the joint space (X, Y) at an angle θ to X. Since the projection directions are fixed in the population, a rotation by θ will cause the distribution to be projected onto X' instead of X. Therefore, X' is distributed as $\cos(\theta)X + \sin(\theta)Y$. Rotation of a distribution in the population corresponds to a weighted convolution of the univariate marginal distributions. As mentioned in Section 2.1.4, convolution typically produces nonlinear paths in the space of QFs. One exception is the multivariate Gaussian distribution. Rotation of this distribution corresponds to scalings of its projected marginal distributions. This idea is often expressed by stating that any projection of a multivariate Gaussian is a Gaussian. With this understanding of linear interpolation and Euclidean distance, it is appropriate to consider this QF based multivariate distribution representation as belonging in a Euclidean space.

Compactness

The efficiency of the QF based multivariate distribution representation can be considered in terms of its length (number of bins). First, for a distribution on d variables, d univariate distributions are estimated. Thus the curse of dimensionality is avoided and the representation can be accurately estimated given few samples. Second, similar to the univariate case, d ($2d$) bins, represent the distribution's mean (mean and standard deviations). This allows the number of bins to be set based on the accuracy needed in the application. Compactness can also be discussed in the context of a population of objects. It is easy to construct a compact representation of a population when the population variation forms linear subspaces. The linear subspaces of this representation were discussed above and the methods for building a compact representation in this case are discussed in Section 2.3.

Additional Details of the Multivariate Distribution Representation

As mentioned earlier, a key issue is the choice of projection directions. The PCA approach is used in Section 3.3 for representing the joint probability of pixel intensities for texture classification. Section 3.2 uses simple marginal distributions, the projections onto the coordinate axes, of a distribution of discriminative features (filter responses). The significant effort required to choose and calculate these features is strongly related to the PCA approach. This relationship, which leverages a property of the PCA approach when done on variables with spatial relationships, is discussed more in Section 3.3.2.

Multiple nonorthogonal projections can also be considered for distributions on a low number of variables to capture relationships among the variables beyond correlation. This could be applicable, for example, for shape descriptions based on the probability that an (x, y) position in an image is part of an object's contour in a population of such contours. This is considered in future work in Section 5.2.1. Projection directions that maximize discrimination instead of minimizing information loss could also be found using methods such as independent component analysis (ICA).

The PCA approach is a supervised method for finding projection directions. Using marginals is unsupervised, so a training set is unnecessary. While supervised methods are usually more accurate than unsupervised methods, sometimes for generality or computational reasons unsupervised methods are preferred. The next section presents an additional unsupervised approach based on conditional distributions. This approach is particularly useful when the marginal distributions of a multivariate distribution are identically distributed. In this case the d marginal distributions supply no more information than any one of the marginals, except in reducing sampling error. Hence, it strongly benefits from capturing the correlations among the distribution variables.

Representations of multivariate distributions based on univariate projections cannot capture all of the joint information in the multivariate distribution. The goal, however, is to describe a population of multivariate distributions. The representations presented in this section allow a computationally efficient two stage approach for this task. First, the representation of each multivariate distribution captures the correlations among the distribution variables.

Second, since this representation forms a space in which the variation of a population forms approximately linear subspaces, the correlations among the one-dimensional projections across the population can also be captured. This leads to compact representations of the population that greatly simplify classification and likelihood estimation tasks on the population. This is discussed more in Section 2.3.

2.2.2 Conditional Distributions

This section constructs an unsupervised representation of conditional probability distributions. The representation is primarily discussed in terms of its applicability for representing multivariate distributions with identically distributed marginal distributions. The representation is used for this purpose in Section 3.3.1, for Markov random field (MRF) based texture classification.

Let $p(x|y)$ be the conditional probability distribution on the scalar variables x and y. Similar to the multivariate representations presented in Section 2.2.1, I represent $p(x|y)$ as the concatenation of several quantile functions representing $p(x|y_i)$, where $y_1, \ldots, y_n$ are n subsets that partition y. This is depicted in Figure 2.13. I partition y based on the n bin quantile function of y, where y_i equals the domain of bin i. To construct this representation given samples from $p(x|y)$, first separate the samples into their correct y quantile y_i by partially sorting y. Then, for each y_i estimate an m bin QF using the corresponding set of x values. The concatenation of these n QFs forms the final $n \times m$ dimensional vector representation. The computation of this representation is detailed in Section A.3. When $n = 1$, $p(x|y)$ nicely simplifies to the marginal $p(x)$.

Once again, this representation should be considered in terms of its Euclidean distance, its linear interpolation, its ability to be accurately estimated, and its efficiency with respect to its length. It should also be noted that this representation does not fully capture $p(x|y)$; it is invariant to monotonic functions of the conditioning variable y. This is because the values of y are not directly represented but rather are represented through their relationship to x. However, in two specific situations the representation is fully invariant. The first is when the two random variables are identically distributed, *i.e.*, when $x \sim y$. Any change in y must have

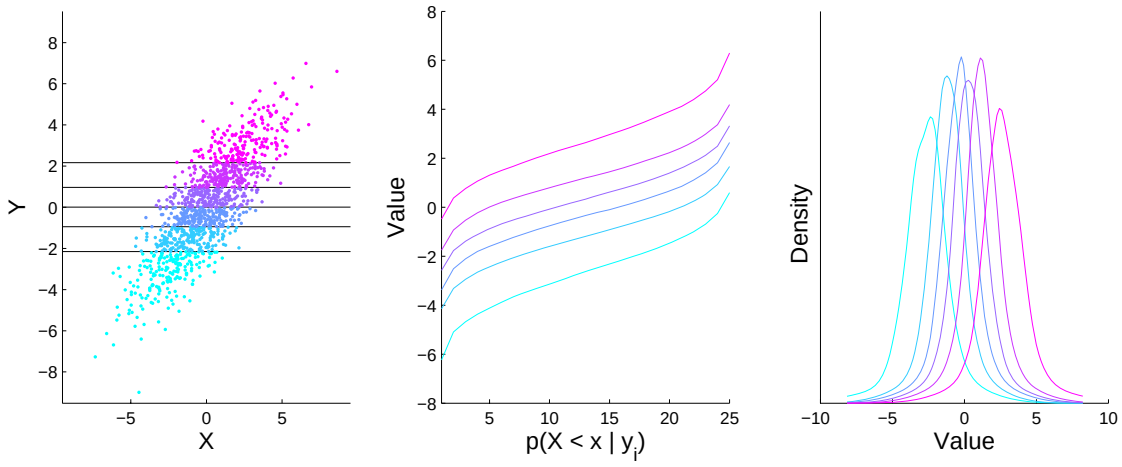


Figure 2.13: A depiction of the QF based representation of conditional distributions. Left: Samples from the joint distribution $p(x, y)$ are partitioned in y. Center: A QF on x is estimated for each y_i. Right: Each $p(x|y_i)$ is displayed as a histogram.

a corresponding change in x. The second case is when $p(x|y)$ is being used as an intermediary to represent a multivariate distribution. For example, $p(x, y)$ can be represented as $p(y)p(x|y)$. The combined representation of $p(y)p(x|y)$ will be fully invariant because changes in y are accounted for in $p(y)$. Both of these cases apply in the application of this representation in Section 3.3.1.

The EMD is not defined for conditional probability distributions. Therefore, the Euclidean distance and linear interpolation of this representation will be analyzed through the distribution $p(x, y)$. The representation of $p(x, y)$ as $p(y)p(x|y)$ can be directly compared to the EMD between distributions on two variables. First, consider changes to x and y that do not affect which partition i that each sample corresponds to. Such changes to y only affect the QF representing $p(y)$. Similarly, such changes to x only affect the QFs representing $p(x|y_i)$. Therefore, any such changes that are linear for QFs will also be linear for this representation. However, variation that changes a sample's partition is more complicated, and it can easily form non-linear manifolds in the feature space. One exception, once again, is variation consisting of the rotation of a multivariate Gaussian distribution, which is linear.

Unfortunately, the Euclidean distance of this representation has a weak relationship to the EMD. Any type of variation can make Euclidean distance differ from the EMD. As mentioned in Section 2.2.1, the EMD between two distributions represented by samples computes a 1-1

correspondence between the two sets of samples. The Euclidean distance of this representation is comparable to the EMD only when it computes the same correspondence as the EMD. This representation assumes that samples remain in the same partition of the conditioning variable. Further, by computing independent QFs for $p(y)$ and each $p(x|y_i)$, the representation assumes the maximal correlation of x and y within each subset y_i. Therefore, when the optimal correspondence is computed, Euclidean distance is a lower bound on the EMD. In general, however, any mismatch between the correspondence assumed in this representation and the optimal correspondence will increase Euclidean distance as compared to the EMD.

The ability to estimate this representation from samples is also conveniently discussed in the context of the joint distribution $p(x, y)$. The $n \times m$-vector representing $p(x|y)$ can be viewed as an adaptive binning of the space (x, y). Since each bin represents $1/(nm)$ of the distribution, each bin is estimated with the same accuracy. However, this representation requires $n \times m$ values instead of the $2n$ values required by the representation given in the previous section. A generalization of this conditional representation for distributions on d variables that would require n^d values would be both computationally infeasible and impossible to accurately estimate. Therefore, I now present a generalization requiring dn^2 values.

Let $p(x_j)$ be a probability distribution on $d+1$ random variables. Partition x_j into the random variable y and d random variables x_i, where y is chosen so that it is most correlated to the other d random variables in x_j. $p(x_j)$ can now be rewritten as $p(y, x_i) = p(y, x_1, \ldots, x_d)$. Using Bayes rule, $p(y, x_1, \ldots, x_d) = p(y)p(x_1, \ldots, x_d|y)$. Then, assuming the conditional independence of each x_i given y, this simplifies to $p(y) \prod_{i=1}^{d} p(x_i|y)$. This assumption corresponds to a known assumption in the MRF literature and is discussed in Section 3.3.1. When y and the d x_i marginal distributions are identically distributed, this can be rewritten as $p(y) \prod_{i=1}^{d} p(y|x_i)$. A representation of $p(y, x_1, \ldots, x_d)$ can now be constructed by concatenating together the representations for each of the reduced $d + 1$ independent distributions.

Sections 2.2.1 and 2.2.2 discussed represents of multivariate distributions. Next, Section 2.2.3 considers populations of standard, univariate distributions that contain mixture variation.

2.2.3 Quantile Function Mixtures

Section 2.1 showed that the quantile function is appropriate for modeling distributions with variation similar to location and scale change. Mixture variation, however, was shown to form nonlinear paths in the space of QFs. One approach to modeling mixture variation is to explicitly compute the underlying distributions and their mixture weights. This is the approach taken in this section, where quantile function mixtures are computed by estimating each underlying distribution by a quantile function. Let Y and $X_1, \ldots, X_n$ be univariate random variables, and let $w_1, \ldots, w_n$ be mixture weights such that $Y \sim w_1 X_1 + \ldots + w_n X_n$. A QF mixture $\underline{Q} = [w_1, \underline{Q_1}, w_2, \underline{Q_2}, \ldots, w_n, \underline{Q_n}]$ is defined where the QF $\underline{Q}$ defines the distribution followed by Y, and the QF $\underline{Q_i}$ defines the distribution followed by X_i, for $i = 1, \ldots, n$.

This section focuses on understanding the linear subspaces of QF mixtures and on how to relate their Euclidean distance to the EMD. The estimation of the QF mixture parameters for a given distribution or for a given set of distribution samples is only briefly discussed. However, this can be a complicated task that is the focus of a large body of literature in mixture modeling [Has66, TSM85, MP00]. Parametric mixture models, such as a mixture of Gaussian distributions, are constrained, *i.e.*, they cannot exactly represent all distributions. This leads to a parameter estimation task that must trade off between the accuracy of the mixture model and the likelihood of the model parameters as given by a prior. QF mixtures are able to exactly represent all distributions, so do not face this tradeoff.

This flexibility of QF mixtures, however, leads to the disadvantage that there is ambiguity in the representation: a given distribution can be exactly represented by a variety of QF mixtures. A prior on the model parameters resolves this ambiguity by allowing the most likely QF mixture to be selected. In the medical imaging applications discussed in Chapter 4, a QF mixture is estimated using thresholding, where $n - 1$ threshold values separate n underlying distributions that correspond to different tissue types. When the underlying distributions are widely separated, this approach is accurate as well as computationally simple, and it resolves any ambiguities in the representation. This is further discussed in Section 4.2.

The linear subspaces of QF mixtures are easily understood. By construction, QF mixtures have the linear subspaces of QFs plus the linearity given by the mixture parameters. Therefore,

mixture changes between the $\underline{Q_i}$'s are linear, but mixture variation within each $\underline{Q_i}$ remains nonlinear. The additional mixture linearity, however, comes at a cost. As discussed above, estimating a QF mixture can be more difficult than estimating the distribution's QF. Also, QF mixtures are less general than QFs; QFs are a purely nonparametric representation while QF mixtures introduce specific model parameters that must be chosen in advance for an application.

While linear interpolation of QF mixtures is appropriate, unfortunately Euclidean distance is not. One simple requirement of a sensible Euclidean distance is for the dimensions to be in commensurate units; the mixture weights and quantiles in a QF mixture are incommensurate. I will now linearly scale the space of QF mixtures to make Euclidean distance appropriate while leaving linear interpolation unchanged. Depending on the number of underlying distributions in the QF mixture and the assumptions made, Euclidean distance can be made to be locally equivalent to the EMD or to an upper or lower bound of the EMD.

First, consider mixtures of 2 quantile functions. Let $\underline{Q} = [w_1, \underline{Q_1}, w_2, \underline{Q_2}]$, where $\underline{Q_i}$ has b_i bins. The EMD is measured in units of work, mass $\times$ distance. For a QF, each dimension has a fixed mass and a variable location. A change in a variable is a change in location, which is distance and it can be converted to work by multiplying by its mass, $1/b$. In a QF mixture, the weight of the quantiles in $\underline{Q_i}$ are w_i/b_i. The weights, w_1 and w_2, can also be put into units of work. A change to w_i corresponds to moving mass from one of the underlying distributions to the other. A change in mass can be converted to work by multiplying by the fixed distance the mass must travel. The distance between the underlying distributions is their EMD, EMD$(\underline{Q_1}, \underline{Q_2})$. I include both w_1 and w_2 in this representation, which counts both the positive and negative movement of the distribution mass. Therefore, I instead multiply w_i by half the distribution distance, $\frac{1}{2}$EMD$(\underline{Q_1}, \underline{Q_2})$. Alternatively, only w_1 could be included in the representation, but this approach does not generalize well for $n > 2$. Let $\underline{Q_{ave}} = \frac{1}{2}(\underline{Q_1} + \underline{Q_2})$. To summarize, $\underline{Q}$ can be scaled to $\underline{Q'}$ by setting $\underline{Q'_i} = w_i\underline{Q_i}/b_i$ and $w'_i = \frac{1}{2}\text{EMD}(\underline{Q_1}, \underline{Q_2})w_i = \text{EMD}(\underline{Q_i}, \underline{Q_{ave}})w_i$. In practice, a lower bound of the EMD between $\underline{Q_1}$ and $\underline{Q_2}$ can often be used. If the two distributions are well separated, the difference of their means is an accurate lower bound on their EMD. In this case, $w'_i = |\mu_i - \mu_{ave}|w_i$.

The scaling computed above is specific to the QF mixture $\underline{Q}$. Euclidean distance is equal to the EMD only when comparing QF mixtures close to $\underline{Q}$. Otherwise the assumption that the weights and quantiles can be independently scaled is false. It is also inappropriate to compare QF mixtures that have been scaled with respect to different distributions. Therefore, I define a metric between two QF mixtures using their average to determine the scaling. Similarly, for a population of QF mixtures, distances can be computed with respect to the average QF mixture of the population. For a population, this results in a Euclidean distance near the population's mean that is approximately equal to the EMD.

Currently, a distance metric has been defined for QF mixtures consisting of two underlying distributions. When $n > 2$, a similar metric can also be constructed. However, the exact EMD is difficult to express in terms of the parameters of a QF mixture. Therefore, an upper bound of the EMD is used instead. Let $\underline{Q} = [w_1, \underline{Q_1}, w_2, \underline{Q_2}, \ldots, w_n, \underline{Q_n}]$ be a QF mixture with n underlying distributions. The scaling computed for the quantiles in the $n = 2$ case are still appropriate, where $\underline{Q_i}$ is scaled to $w_i \underline{Q_i}/b_i$. The scaling on the weights, however, does need to be reconsidered. For $n > 2$, when mass moves from an underlying distribution, it is not straightforward which underlying distribution it moves to. This problem is equivalent to the underlying optimal matching done in the EMD itself. Therefore, I use an upper bound on this distance that leverages the triangle inequality. For $n = 2$, $w_i' = \text{EMD}(\underline{Q_i}, \underline{Q_{ave}})w_i$. This distance is exactly the EMD because as the mass moves from $\underline{Q_1}$ to $\underline{Q_2}$, or vice versa, it must pass through $\underline{Q_{ave}}$. For $n > 2$, the mass does not need to pass through $\underline{Q_{ave}}$. However, due to the triangle inequality, forcing the mass to go through $\underline{Q_{ave}}$ makes the distance an upper bound of the EMD. I use this scaling for the $n > 2$ case. Intuitively, the w_i's that increased move extra mass to the mean distribution, and the w_i's that decreased grab needed mass from the mean distribution. Therefore, knowing the matching between the w_i's is not needed. Thus, the same scaling of $\underline{Q}$ is computed for all values of n, just its interpretation changes.

The space of QF mixtures forms a convex space. The constraint $\sum_{i=1}^{n} w_i = 1$ is linear, which implies that averaging and interpolation will be valid. Also, since the scalings computed above are linear these properties also hold for the scaled QF mixtures.

2.2.4 Summary

Section 2.2 presented three representations of probability distributions that generalize the quantile function. The generalizations for multivariate distributions presented in Sections 2.2.1 and 2.2.2 are used in Chapter 3 to represent textured materials. The generalization for univariate distributions containing mixture variation presented in Section 2.2.3 is used in Chapter 4 to represent the appearance of organs in CT images. Next, Section 2.3 discusses how to estimate likelihoods in all of these spaces from a population.

2.3 Population Likelihood Estimation

This section provides accurate and efficient descriptions of populations of objects represented using one of the quantile function based representations discussed in Section 2.2. Each population is described by its mean and covariance using principal component analysis, under the assumptions that the population is Gaussian, that only valid objects are of interest, and that objects dissimilar to the population are expected and must be identified. The resulting models are appropriate for the two related tasks considered in Chapters 3 and 4, classification and segmentation. The segmentation task models a single population of objects. Then, the object most likely to come from that population is sought, under certain constraints. The classification task models multiple populations labeled into different classes. Then, new objects are presented, and the most likely, existing class that it belongs to is found.

This section presents an identical methodology for use in both classification and segmentation. Section 2.3.1 describes likelihood estimation of a single population. Chapter 4 discusses how these likelihood estimates can be used in the posterior segmentation of organs in CT images. Section 2.3.2 describes a classifier based on the decision boundaries implied by each population's estimated likelihoods. This classifier is used and discussed further in Chapter 3. Section 2.3.3 discusses other interpretations of this method; Section 2.3.4 discusses how to select the parameters of the method.

2.3.1 Modeling a Population's Variability

Let $\{\underline{Q_i}\}$ be a population of n objects described by one of the quantile function based representations presented in Section 2.2. This section assumes that $\{\underline{Q_i}\}$ are samples from an underlying probability distribution P that must be estimated. The resulting estimate, $\hat{P}$, is used to define the likelihood that a new object is from P. I parametrically estimate P by assuming it is Gaussian distributed as $\mathcal{N}(\mu, \Sigma)$, *i.e.*, I estimate its first and second order statistics. The details of this model will now be discussed along with the factors that determine its appropriateness and its ability to be accurately estimated using principal component analysis (PCA).

First, μ can be simply computed as the linear average of $\{\underline{Q_i}\}$. Recall that μ will always be a valid quantile function since the space of the representation is convex. Further, μ will be representative of the population when it exists on a linear subspace. Section 2.1 described in detail the convexity and linear subspaces of QF functions; Section 2.2 discussed how these properties are conserved for the generalized QF representations. Chapters 3 and 4 discuss the linearity of their particular populations.

Σ can be estimated using PCA. In order to understand if it is appropriate to use PCA in this situation, it is important to remember that PCA is typically used for two different tasks with different requirements. One task is to generate points of interest, which requires, in increasing order of stringency, convexity, linearity, and a vector space. The other task is to estimate the likelihood of points in a space, which requires convexity, linearity, and Gaussianity (in increasing order of stringency). The first task is generative while the second is discriminative. The generative task requires a vector space so that only valid points are generated. The discriminative task, however, is typically not concerned with invalid points. Both tasks considered in Chapters 3 and 4 are discriminative, and only the probability of valid points are of interest. The likelihood of invalid points is never asked for, so the fact that they get assigned a nonzero probability is of little concern. Σ is being estimated in this section for such a discriminative task. The convexity of the space and the linearity of the population's variation have already been discussed. Approximate Gaussianity is assumed in this section and for the populations considered in Chapters 3 and 4.

Now, I consider how well Σ can be estimated using PCA. Assuming that the population $\{\underline{Q_i}\}$ is appropriately Gaussian, there are three main factors that determine how well Σ can be estimated: the number of points in the population, n, the dimension of the space, d, and the inherent dimensionality of the population, D. The inherent dimensionality of a population is the dimensionality of the subspace ($\mathcal{R}^D$) that the population is restricted to in the full space ($\mathcal{R}^d$), disregarding any noise present in the population samples.

PCA is typically considered only in terms of n and d. The populations in chapters 3 and 4 typically have $d > n$, with n in the 10's and d in the 100's. This is known as a high dimension low sample size (HDLSS) situation [MCAM]. A direct application of PCA can only estimate a singular covariance matrix, Σ', in HDLSS situations. Σ' only estimates the likelihood of points in a subspace of $\mathcal{R}^d$. The likelihood of a point x in $\mathcal{R}^d$ is computed by first projecting x into the computed subspace as x' and then computing the likelihood of x'. Σ', however, is inappropriate for the tasks considered in Chapters 3 and 4. Both tasks need to estimate how likely x is from P, when you expect to see points not from P. Thus, the likelihood of points far from the estimated subspace need to be computed. Σ' discards the difference between x and x', the information that in some situations is the most informative for determining if x is from P.

In order to estimate a non-singular covariance matrix, I consider Σ in terms of the population's variation in $\mathcal{R}^D$ and an isotropic variation, or noise, in $\mathcal{R}^d$. The covariance matrix can then be thought of as $\Sigma = \Sigma' + \sigma' I$, the sum of a singular covariance matrix and an isotropic variance. The populations considered in Chapters 3 and 4 are shown to exhibit a low inherent dimensionality, allowing this formulation to be effective and efficient despite the high dimensionality of the space (large d) and the limited sample sizes (small n). Therefore, for the remainder of this section I will assume that $D < n < d$.

Before discussing how to estimate Σ using PCA, first I first express Σ in a different form. Any non-singular covariance matrix in $\mathcal{R}^d$ can be written as $\Sigma = U\Lambda U^{-1}$, where U is a rotation matrix composed of orthogonal unit vectors and Λ is a diagonal matrix. PCA expresses Σ in such a form where the columns of U are eigenvectors of Σ and the diagonal entries of Λ are eigenvalues of Σ. The above Σ can be expressed in this form using eigenvalues

$[\lambda_1, \ldots, \lambda_D, \sigma, \ldots, \sigma]$, where there are $d - D$ σ's. The maximum likelihood estimate (MLE) of covariance matrices of this form can be estimated as follows. Use PCA to compute, in decreasing order by eigenvalue, the n non-zero eigenvalues, λ_i, with corresponding eigenvectors, U_i, in the d dimensional space. The columns of U are $U_1, \ldots, U_D$ and an arbitrary orthogonal basis in the remaining $d - D$ dimensional subspace. Λ is composed of $\lambda_1, \ldots, \lambda_D$ and $\sigma = \sum_{i=D+1}^{n} \lambda_i/(n - D)$, the sum of the remaining eigenvalues normalized appropriately for the HDLSS situation.

In my experiments in Chapters 3 and 4, I have found the above formulation to be overly sensitive to σ. This is due to the fact that often $d >> d - D$, making σ much more important in the likelihood estimate than the D eigenvalues. Therefore, in my model I do not normalize σ by dividing by $n - D$, instead I set it as the simple sum of the remaining eigenvalues. This formulation can be viewed as measuring the expected projection error onto the measured $\mathcal{R}^D$ subspace. The resulting Gaussian likelihood estimate contains $D + 1$ Mahalanobis distances rather than the d in the original formulation, which seems sensible since it is based on the inherent variability of the population instead of the arbitrary dimension of the space.

This section described an approach to estimating the likelihood of a population of objects described by a QF based distribution representation. Next, Sections 2.3.2 and 2.3.3 discuss other interpretations of μ and Σ. Section 2.3.4 then considers how to select the new parameter this approach introduces, the number of kept eigenvalues.

2.3.2 Classification

This section describes how the population likelihoods presented in the previous section can be used for supervised classification. Classification has 2 phases. First, the classifier is trained using samples from each of the c classes. Second, novel objects are presented and identified by the classifier into 1 of the c classes learned during training.

I train the classifier by learning a mean, μ_i, and a covariance, Σ_i, for each class. Novel target objects are identified by assigning them to the class for which it has the maximum likelihood. This classification strategy is known as quadratic discriminant analysis (QDA) [DHS01]. QDA finds optimal class decision boundaries under the assumption of Gaussian class

variation. When a common covariance structure is learned for all the classes, linear decision boundaries are defined, and the classification strategy is known as Fisher linear discrimination (FLD). When per class covariances are learned, as is the case here, QDA defines quadratic decision boundaries [DHS01].

I use standard QDA with two exceptions. The first is in the estimation of each Σ_i, as was discussed in the previous section. The second is in the selection of the number of eigenvalues for each class. This is discussed in Section 2.3.4.

2.3.3 Other Interpretations

Learning a mean and covariance matrix from samples of a population has other interpretations. First, instead of the likelihood that μ and Σ define, consider the corresponding Mahalanobis distance. The Mahalanobis distance is a linear scaling of the underlying Euclidean distance metric (by Σ) to account for the variability in the population samples. In a classification context, QDA can then be thought of as a nearest neighbor (NN) classifier with one prototype, μ_i, and class specific scalings of the distance metric. The appropriateness of PCA can also be reconsidered in this context. In Section 2.3.1 I claimed that a vector space was unnecessary for discriminative uses of μ and Σ. Since the Mahalanobis distance is a linear scaling of Euclidean distance, it is appropriate when two things hold. The first is when Euclidean distance is appropriate, which has been discussed in detail. The second is when Σ is appropriate. Σ exactly describes multivariate Gaussian distributions, but it can be descriptive and useful in many situations when the distribution is simply unimodal.

The model can also be interpreted as a distribution family. Recall that μ and Σ measure the variation of a population in a D dimensional subspace. The D eigenvectors λ_i measure the expected variation in each of the directions given by U_i. Using different coefficients on the eigenvectors generates points of interest that are similar to the samples. Let $F(\alpha_1, \ldots, \alpha_D) = \mu + \sum_{i=1}^{D} \alpha_i U_i$. F is a function of D parameters that generates points of interest. Recall that every point in this space is a representation of a probability distribution. Therefore, F defines a parametric distribution family with D parameters. Similar to standard distribution families, only certain combinations of the parameters will produce valid probability distributions. Also,

F cannot describe all distribution families. While any single distribution can be represented, F is restricted to the distribution families that linearly vary in the underlying space of QF based representations. This includes the location–scale families detailed in Section 2.1. Chapters 3 and 4 give examples of the learned, population specific, distribution families.

2.3.4 Determining the Number of Principal Components

This approach introduces a parameter for each population that must be determined, the dimensionality of the estimated subspace. Both the classification and segmentation tasks considered in Chapters 3 and 4 contain multiple populations. The classification task contains a population for each class. The segmentation task often splits a higher dimensional probability distribution into multiple, independent distributions with their own likelihood estimates.

Typically, it is too difficult to manually select, or automatically tune, this parameter separately for each population. Therefore, the parameters are constrained so that there is only 1 free variable. One standard approach is to constrain each population to have the same number of components. However, this makes the assumption that the covariance of the populations are similarly distributed, which is unfounded. Another standard approach is to constrain each population to have similar relative remaining variances. Thus, the same percentage of the covariance is captured for each population. This approach is standard for generative, or compression, purposes, since error is explicitly tracked. However, in a discriminative context this approach assumes each population has the same total variance.

In this work, I choose to constrain each population to have the same absolute remaining variance, which is the computed expected projection error, σ_i. This constraint leverages the fact that Euclidean distance is sensible, so it should allow a consistent measure across the populations of which variation is important. Such a consistent measure assumes that each population has the same level of noise in their variable estimates. This is often a reasonable assumption and it allows the chosen projection error to be considered as the common noise level across classes. Additionally, one rationale for choosing the same absolute remaining variance across classes, is that the likelihood model is built with the goal of having the same level of sensitivity when considering objects from different populations. Allowing different σ_is might

introduce a bias or sensitivity to the estimated likelihoods.

I have found it convenient to parameterize the desired σ by the minimum number of components across the population. This sets σ as follows. I compute σ_i for each population given the input number of components. Then, I set the target σ to the minimum σ_i. Next, each population determines the number of components it needs so that $\sigma_i \leq \sigma$. In chapter 3, I use cross-validation on the training populations to automatically determine this parameter. I show the sensitivity of the model to this parameter, and I compare this constraint on the number of components to constraining each population to have an equal number of components. In Chapter 4, I manually tune the number of components. However, the range of number of components considered is quite small, typically between 1 and 3.

2.4 Summary and Conclusions

Chapter 2 presented a methodology for estimating the likelihood of several types of probability distributions from populations. These likelihoods are used to build models of texture and object appearance in the driving problems presented in Chapters 3 and 4.

The underlying theme throughout Chapter 2 is compactness. Section 2.1 introduced this theme by describing the general context of this dissertation: finding compact representations of a population of distributions. Next, methods of analyzing compactness were introduced, which was in terms of linearity and the resulting low number of needed parameters. Representations of univariate probability distributions where discussed in this context, for which parametric families are ideal. However, Section 2.1 primarily discussed the non-parametric options when an appropriate parametric family does not exist. Quantile functions were shown to have several advantages over other non-parametric representations. First, they compactly describe a single distribution using notions such as mean and standard deviation. Second, this advantage is not lost when describing a population of distributions. Third, several general forms of distribution variation are linear, including mean change and standard deviation change. Lastly, several parametric families form known submanifolds in the space of quantile functions, which gives a geometric understanding to the space.

Section 2.2 considered methods based on quantile functions for representing multivariate, conditional, and mixture distributions. The compactness of the representations were discussed by showing they had linear properties similar to quantile functions. Specifically, Euclidean distance and linear interpolation were discussed for each. Euclidean distance was shown in each situation to be an approximation to the Earth Mover's distance. The multivariate distribution representations presented in Sections 2.2.1 and 2.2.2 are used in Chapter 3 to represent textured materials. The univariate distribution representation that models mixture variation and that is presented in Section 2.2.3 is used in Chapter 4 to represent the appearance of organs in CT images.

Section 2.3 discussed how to estimate likelihoods from a population in such spaces. The presented methods leverage the fact that the population is expected to have linear variation and to be compact. Linearity and compactness greatly simplify estimation tasks and allow for fairly standard techniques to be used. In particular, Section 2.3 discussed the appropriateness of principal component analysis for estimation in these spaces, methods for handling high dimension low sample size situations, and methods for selecting the introduced parameters.

Chapter 3

Quantile Function Based Texture Classification

This chapter applies the methodology described in Chapter 2 to texture classification. Specifically, photographed materials are represented using probability distributions of texture features and identified from examples. Texture classification consists of three main elements: 1) texture features that are typically dense and per pixel, 2) the representation of these features for an entire texture (image), and 3) the classifier or comparison measure between texture representations. A driving concern throughout this chapter is the relationship between texture features and their representation. I consider these two elements together with the goal of compactly and discriminatively describing a population of textures. A compact population representation greatly simplifies the actual classification task, allowing the accurate and efficient classification techniques discussed in Section 2.3.

Section 3.1 discusses previous texture classification work. First, Section 3.1.1 describes texture and existing data sets in the texture community. Particular interest is given to the Columbia-Utrecht reflectance and texture database (CUReT) [DvGNK99]. All experiments reported in Sections 3.2 and 3.3 are on CUReT. Then Section 3.1.2 describes existing methods for texture classification. Existing approaches are stressed that use histograms with bin locations defined through clustering. This approach to representing multiple features is common across many applications, so it is of particular interest to this work, which spans multiple applications itself. Particular interest is also given to the MR8 filter bank [VZ02]. This fil-

ter bank was introduced by Varma & Zisserman for CUReT, and I use it for my classifier in Section 3.2.

Section 3.2 describes my filter bank based classification methodology and gives results. Section 3.3 describes my Markov random field (MRF) based classification methodology and gives results. The similarity, strengths, and weakness between these two approaches as well as between my approaches and previous work are discussed. An early version of this work appears in [Bro05].

3.1 Texture Classification Background

Texture analysis has a rich history dating back three decades. Due to the breadth of the texture analysis literature, this section focuses on the statistical texture classification techniques most related to this work. Also, the challenges and techniques in texture analysis are not unique to this task. This section describes some related methods that are not from the texture analysis community.

As mentioned in Section 1.1.1, texture refers to the characteristic visual patterns exhibited by particular types of objects in images. Objects that can be described well in terms of texture have surfaces that exhibit some degree of approximate spatial periodicity, either deterministic or stochastic. Texture analysis seeks to represent images of such objects in a way that captures the characteristic patterns in the texture while being invariant to other information specific to the image, such as viewing and illumination angle. In other words, a representation of the object is sought that is more compact than the image and that only captures information that helps distinguish between different sets of textures.

Section 3.1.1 further describes texture and existing data sets in the community. Section 3.1.2 discusses existing methods to analyze such databases.

3.1.1 Texture and Existing Databases

Below are two early definitions of texture compiled by Coggins [Cog82] and also given in [TJ98].

1. "We may regard texture as what constitutes a macroscopic region. Its structure is simply attributed to the repetitive patterns in which elements or primitives are arranged according to a placement rule." [TMY78]

2. "A region in an image has a constant texture if a set of local statistics or other local properties of the picture function are constant, slowly varying, or approximately periodic." [Skl78]

The above definitions correspond to structural and stochastic views of texture. Structural models decompose texture into two elements, underlying texture elements and their arrangement [Har79, BA88, VP88]. Stochastic models focus on the random properties instead of the deterministic properties of the texture, and they describe local properties of the texture sufficient for characterization [HB95, GS05]. Both definitions leverage the important texture property of spatial homogeneity at a particular scale, which allows texture to be identified by local statistics. Julesz analyzed the set of local statistics used in preattentive human texture perception, the process that makes textures "effortlessly distinguishable" [Jul81, BJ83]. Julesz introduced the term "texton" to describe the basic texture primitives recognized in preattentive perception. Textons are analogous to phonemes in speech recognition. Textons are described as "elongated blobs (of given orientation, width and aspect ratios) and their terminators" [Jul81]. Julesz hypothesized that the human preattentive system analyzes the frequency of textons and does not perform any higher order statistical analysis of spatial interactions in the texture. Both the structural and stochastic models can be decomposed into and thought of in terms of such textons.

There is a large variety in the types of textures considered in the texture analysis community. Textures can be generated or imaged through various devices such as cameras, aerial satellites, microscopes, sonar, computed tomography (CT), magnetic resonance (MR), and ultrasound (US). There are purely structural and stochastic textures, as well as textures with both structural and stochastic aspects, such as natural textures and imaged materials. This chapter focuses on natural and material textures imaged using standard cameras. Imaged textures are divided according to their surface properties into either two-dimensional (2D)

or three-dimensional (3D) textures [DvGNK99, LM01, CHM05]. 2D textures have smooth, locally planar surfaces whose primary, physical cause is local variation in surface spectral reflectance. 3D textures have rough surfaces whose texture is related to local height variations. Even in the restricted domain of imaged 2D and 3D textures, the texture databases used in the community have varied over the years.

As mentioned in the beginning of Section 3.1, the relevant information and desired invariances in a texture description depend on the specific set of textures being examined. This points out one difficulty in texture analysis: the generalizability of methods beyond the examined database. However, specific types of texture variation are of interest in the community, and methods and databases can be generally discussed in terms of the types of variation they handle or express.

Early texture classification work used databases such as the Brodatz collection [Bro66]. Typical experiments acquired multiple training and target images per class by partitioning the single per class image supplied by the database. Therefore, the texture variation within a class is limited and only includes sampling variation caused by large scale features or deformations in the physical material being imaged. Later texture databases, such as the MIT Vision [vis] and MeasTex Image [mea] texture databases, introduced variability that would be expected in less constrained, "real world" situations. Such situations include variation due to lighting and viewpoint angle. However, the variation included in these databases is not comprehensive because only a small number of lighting conditions or viewpoint angles are given.

More recently, the CUReT [DvGNK99] and the KTH-TIPS2 [CHM05] databases have been introduced, which supply a much more comprehensive collection of images. CUReT, which is described in detail below, supplies 205 images of 61 materials taken in a controlled environment under varying viewing and illumination angles. Such a database allows texture models to be constructed and analyzed in terms of these specific variations. KTH-TIPS2 supplies images similar to CUReT but that contain two additional forms of variation. First, changes in scale, *i.e.*, zoom, in the camera are included for each material. Second, multiple example materials for each class are defined. Multiple examples allows the classification of true texture categories, instead of identification of specific examples in a category.

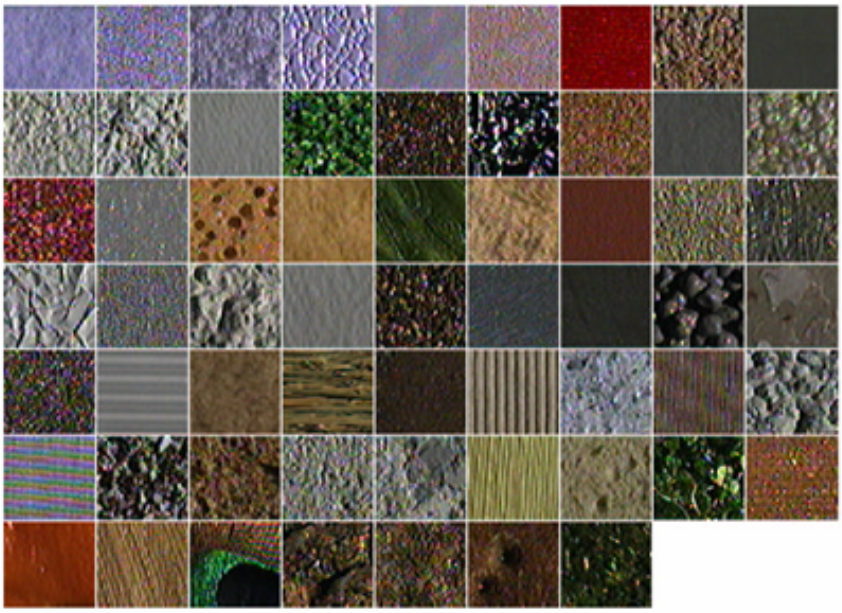


Figure 3.1: The 61 materials in the CUReT database. Image taken from [VZ02].

The Columbia-Utrecht Reflectance and Texture Database

CUReT was collected by researchers at Columbia University and Utrecht University [DvGNK99]. Figure 3.1 shows each of the 61 materials at a frontal viewing angle. Each class contains images from one material that exhibit 3D effects such as specularities, inter-reflections, and shadowing, as shown in Figure 3.2. This large intra-class variability makes correct classification of the database a challenging task. The limitations of CUReT include a lack of significant scale change, limited in-plane rotation, and small-scale texture features. Small-scale features tend to simplify classification tasks by allowing more compact and better sampled texture measurements.

In Sections 3.2 and 3.3, I follow an experiment on CUReT designed by Varma & Zisserman [VZ02] and followed in [VZ03, HCFE04] (and also, roughly, in [PNMT04]). The experiment uses 92 of the 205 per class images, those with the largest minimum number of valid pixels across the samples. Each of these $61 \times 92 = 5612$ images are cropped to a resolution of 200×200, converted to grey scale, and processed to have zero mean and unit variance. In Sections 3.2 and 3.3, I use exactly the same images as Varma & Zisserman, which Varma supplied [VZ02]. In Section 3.3, I discuss the necessity of the grey-scale intensity normalization, which is done

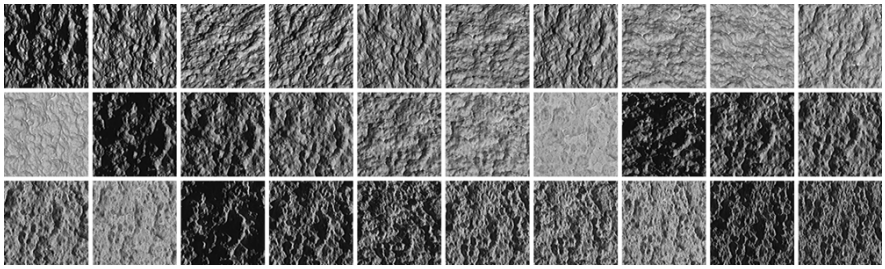


Figure 3.2: Thirty images from the "Zoomed Plaster B" material (number 30) illustrating the large intra-class variability present in CUReT.

to achieve partial invariance to linear intensity variation across images.

An experiment must also split the 92 per class images into disjoint training and test sets. Varma & Zisserman typically reported results for two cases, each with 46 training and 46 target images per class. The first case alternates training and target assignment in the order the images are given. The second case gives results averaged over a small number of random splits. These splits yield a total of $61 \times 46 = 2806$ training images and 2806 test images for each split. In Sections 3.2 and 3.3, unless otherwise specified, I report results averaged over 100 random splits with equally sized training and test sets. For consistency, the test set is not modified when smaller training sets are examined.

Of the 5612 images considered in CUReT, one of the images has a corrupted file. It is image 60-101 (sample 60, view 02-62). The effects of this corrupted image are not discussed further, beyond noting that parametric classifiers, such as QDA, are more sensitive to outliers than non-parametric classifiers, such as nearest neighbor.

3.1.2 Existing Methods

As mentioned at the beginning of this chapter, texture classification consists of three main elements: 1) texture features that are typically dense and per pixel, 2) the representation of these features for an entire texture (image), and 3) the classifier or comparison measure between texture representations. The choice of these elements depends on the textures of interest. Of concern are the types of expected textures and their variation, which influence the effectiveness and appropriateness of the various texture analysis elements.

This section continues with a discussion of texture features and their representations with

an emphasis on dense features that are statistically represented. Then, invariances are discussed in terms of the 2D and 3D texture databases described in Section 3.1.1. Finally, texture classification is discussed.

Texture Features

One category of texture features is based on the spatial statistics of pixel intensities. Features include co-occurrence matrices (or gray level co-occurrence matrices (GLCMs)) [HSD73] and Markov random fields (MRFs) [Pag04]. Spatial statistics are characterized by their order and have the following definitions. First-order statistics measure the distribution of gray values of a single pixel, which is simply the marginal PDF of intensity in the image. Second-order statistics measure the intensity distribution of two pixels at a fixed spatial relationship. These 2-variate distributions can be thought of as measuring the intensity at the ends of a dipole at a fixed orientation and length [Jul81]. There are many possible second-order statistics in an image. Higher-order statistics, n-order with $n > 2$, can be similarly defined as the n-variate distributions of gray values of n pixels at a fixed spatial relationship. A GLCM is a second-order statistic represented, if there are G gray values, by the full $G \times G$ two-dimensional histogram. Since this histogram is large and several spatial relationships may want to be modeled, summary statistics of GLCMs are often computed as texture features, such as autocorrelation, entropy, and homogeneity (see [TJ98] for their definition). Such summary statistics are similar to scale or orientation summaries of the fourier transform, which all form non-dense texture features that will not be discussed further.

MRF models measure the distribution of intensities in compact pixel neighborhoods. Given a 3×3 (5×5) neighborhood, MRFs estimate its 9(25)-variate joint distribution or, equivalently, its 8(24)-variate distribution conditioned on the center pixel's intensity. This 9(25)-variate distribution is a specific 9(25)-order statistic. MRFs are further related to spatial order statistics by the Hammersley-Clifford theorem [HC71], which allows MRFs to be decomposed into distributions on a set of cliques. The cliques of a 3×3 neighborhood are all of the first, second, third, and fourth order statistics with the spatial constraint that all pixels in the statistic are neighbors. Some MRF models, such as the Derin-Elliot [DE87] and auto-binomial [Bes74], are

constrained to first and second order cliques, capturing equivalent information to GLCMs.

Another category of texture features is based on spatial filtering. Spatial filters are a generalization of global measures of frequency like the Fourier transform to local measures of orientation and scale. Features are the response of a bank of linear spatial filters convolved with the image. Many such approaches have been proposed since the 1980s [KG83, KvD87, MP90, HB95]. Filters include Gabor functions, Gaussian derivatives, Laplacians, differences of offset Gaussians, order moments, wavelets, and image pyramids. Invariant filters and the representation of distributions of filters are discussed later in this section. Spatial filters are linked to MRF models since a linear filter is a one-dimensional projection in the joint space of intensities modeled by MRFs. This is discussed more in Section 3.3.2.

Spatial filtering was inspired by models of processing in the early stages of the primate visual system, which suggested that a retinal image is transformed into a local spatial/frequency representation [VV88, BA88, VP88, MP90, Hee93]. Several methods have attempted to model this visual system, which has resulted in filter banks with particular properties. For example, Malik & Perona used only even-symmetric, second-order filters [MP90]. After filtering, Malik & Perona and Heeger perform half-wave rectification, where the positive and negative responses of each filter are separated into their own features [MP90, Hee93]. After rectification, these methods perform additional nonlinear normalization based on inhibition. In later work, Malik also argues that the PDF of filter responses captures information equivalent to the first order statistics of textons [MBLS01], similar to the human preattentive system [Jul81]. Continuous filter responses measure translation and rotation variate responses to discrete textons. Hence, clusters in the filter response PDF space represent textons [MBLS01].

Local binary patterns (LBPs) are texture features similar to a discrete valued filter response [OPH96]. For each pixel, LBPs construct a binary mask (code) that describes the local neighborhood by thresholding neighboring pixels by the gray value of the center pixel. An entire texture is represented by the frequency of each binary mask. This is similar to the frequency of clusters in the joint space of filter responses [MBLS01].

There are other texture features that do not easily fit into one of the categories above. Geometric features were not discussed since they can be principally applied only to black and

white structural textures. Also of note are dense fractal features such as the fractal dimension [CS95]. Roughly speaking, fractals appear similar at all scales, which means a fractal is composed of n copies of itself scaled by a factor s. Fractal dimension, d, assumes n and s are related by a power law, *i.e.*, $n \propto s^{-d}$. Local fractal dimension can be computed using the sum of intensities at a radius r from the center pixel as the "number" of copies and r as their "scale" [CS95].

Feature Representations

Dense, per pixel texture features were stressed above. Such features are represented across an entire texture by their probability distribution. Distributions leverage the large scale homogeneity in textures to form a more compact representation than the image. The types of distributions that represent a texture depend on the texture features, the texture, and the variation within the set of textures. For specific combinations of these three variables, appropriate parametric representations, *i.e.*, families, have been found. Further, if an appropriate family exists, as was discussed in Chapter 2, the family is an ideal representation of the texture in terms of compactness and linearity. If an appropriate family does not exist, non-parametric representations are used instead.

MRFs have a long history of parametric estimation. The corresponding Gibbs distribution is broken down into distributions on several cliques each with a set of model parameters. Examples include the Derin-Elliot [DE87] and auto-binomial [Bes74]. For stochastic textures, Geusebroek has shown that spatial statistics of intensity differences, including those assessed by Gaussian derivative filters, are well modeled by Weibull distributions [GS05]. Grenander presents a simplified parametric model for clutter in natural images [GS01]. However, the textures in the CUReT and KTH-TIPS2 databases do not fit into any one of these categories, which highlights the difficulty in finding an appropriate parametric model for texture analysis.

Non-parametric, histogram-based probability distribution representations are widely used and are effective because they provide a rich, unconstrained estimate of the distribution of texture features [MBLS01, CD01, VZ02, VZ03, LW03, Blu04, VG07]. Many features have been represented using standard, uniformly binned marginal or two-dimension histograms, includ-

ing GLCMs [Har79], LBPs [OPH96], and filter responses [LW03]. For filter based methods, recent approaches have estimated the joint PDF of the filter responses. Some methods have argued that this is essential [MBLS01, VZ02] while others argue that the marginal distributions are sufficient and preferred [LW03, GS05]. Joint PDF estimates are more expressive than marginals, but they require more data for estimation and are therefore prone to overfitting. Section 3.2 presents a texture classification algorithm based on marginal distributions. Malik *et al.* proposed a method for estimating the joint distribution of filter responses using clustering [MBLS01]. Representative cluster centers define a texton dictionary, yielding a texton histogram representation for each image. Such an approach is common in computer vision, where it is referred to as a bag of visual words model. Similar joint distribution estimates have since been used for filter responses [CD01, VZ02, CD04], MRFs [VZ03, Blu04], and fractal features [VG07].

Two major drawbacks of histogram based representations is that they tend to be noncompact (high dimensional) and texture sets tend to form nonlinear submanifolds. These properties lead to more difficult classification tasks, as is discussed later in this section. Despite this drawback, a trend in the community has been to use such histogram based representations for generality and to compensate for their drawbacks using elaborate classification methods. Additionally, methods based on histogram estimates of joint PDFs tend to have many parameters and to be computationally complex. This is partially due to the need to learn and select a representative texton library using k-means clustering. However, it is also due to the representation being sensitive to certain variations in the texture features. For example, filter bank features require both intensity and filter response normalization. Sections 3.2 and 3.3 demonstrate the effectiveness of several alternative non-parametric distribution representations based on the quantile function. This work extends that of Levina, who represented textures by a set of independent filter response empirical distributions [Lev02].

Invariance in 2D and 3D Textures

Natural textures, such as those in the Brodatz collection [Bro66], are often only weakly-homogeneous due to large scale changes in the texture or deformations in the imaged material.

Thus, though many are 2D textures, they exhibit local effects from being imaged in our 3D world. Such local effects generated a large interest in local texture features that are invariant to the expected imaging effects. Some local invariances of interest are related to viewing angle, including in-plane rotation, out of plane rotation (projection), scale, or full affine invariance. Other desired invariances are to lighting conditions, such as lighting angle and intensity.

By design, some of the features already discussed have certain invariances. LBPs are invariant to scaling of intensity [OPH96]. Varma & Garg showed that two fractal features, local fractal dimension and local fractal length, are affine and intensity shift invariant, and in-plane rotation and intensity shift invariant, respectively [VG07]. Particular spatial filters, such as the Gaussian and Laplacian of Gaussian, are also rotationally invariant.

One common approach to developing texture features with particular, desired local invariances is to modify existing variant features. Several types of features have been modified to have in-plane rotational invariance, including generalized co-occurrence matrices [DJA79], LBPs [PNMT04], filters [VZ02], and MRFs [VZ03]. Varma & Zisserman developed a rotationally invariant filter bank called MR8 that is discussed in detail later in this section [VZ02]. Local scale invariance has been discussed in [HCFE04, Blu04].

More recently, 3D textures, such as those in CUReT [DvGNK99] and KTH-TIPS2 [HCFE04], have been extensively studied [LM01, CD01, VZ02]. Such textures are typically planar and contain explicitly introduced global variation, due to in and out of plane rotation, scale, or illumination angle. Such variation is global because a set of homogeneously textured images are produced. The dependence of 3D textures on viewing and illumination directions has been referred to as the bidirectional texture function (BTF) [DvGNK99]. Early work analytically modeled the BTF of textures restricted to Lambertian, isotropic, and randomly rough surfaces [DN98]. Other methods have estimated properties of the BTF to quantify, for example, the magnitude of the 3D effects [SH98]. However, for texture classification direct learning approaches have been the most popular. Such approaches typically modeled 3D textures using either the locally variant or invariant features discussed above. Methods using both types of features have been successful, often without either being clearly superior. Neither set of features are ideal in terms of invariances. Variant features maintain full discriminative power, but

they model more variation than desired. Locally invariant features introduce more invariance than desired, reducing discriminative power. Therefore, the preferred features depend on the parameters of the specific experiment, such as training set size.

A texture representation with global invariances would be ideal. Global invariances need to be modeled by the representation rather than at the local scale of the features. Sections 3.2 and 3.3 describe a method that learns, so is approximately invariant to, the types of global variation expected in a class.

Leung and Malik presented one of the few approaches that explicitly learns a 3D representation of the texture [LM01]. The local appearance of the surface at several viewing and illumination conditions was learned using registered images at known viewing and illumination configurations. The resulting distribution of so called 3D textons was used to classify a texture given between 1 and 4 registered images also at known viewing and illumination configurations. However, compared to more recent, similar 2D approaches [VZ02], the 3D approach is computationally more complex, it requires registered images at known viewing and illumination configurations, and it is less accurate. Recently, methods have focused on the classification of single, unregistered images acquired under unknown viewpoint and illumination configurations. Such methods use experiments similar to the one described in Section 3.1.1 on CUReT.

The MR8 Filter Bank

As defined by Varma & Zisserman, the MR8 filter bank consists of 38 linear filters but only 8 filter responses [VZ02]. There are two isotropic filters, a Gaussian and a Laplacian of a Gaussian (LOG), both at scale $\sigma = 10$. The 36 other filters are first derivative (edge) and second derivative (bar) Gaussian filters at 6 orientations and 3 scales $(\sigma_x, \sigma_y) = \{(1,3),(2,6),(4,12)\}$. Figure 3.3 displays these 38 filters. Rotational invariance is achieved by storing only the maximum response over all orientations of a given filter type and scale. The filters are L_1 normalized, and the filter responses are nonlinearly normalized based on Weber's Law [VZ02]. I refer to this as the MR8-1W filter bank.

The 38 filters that comprise MR8 are a subset of a filter bank designed by Leung & Malik

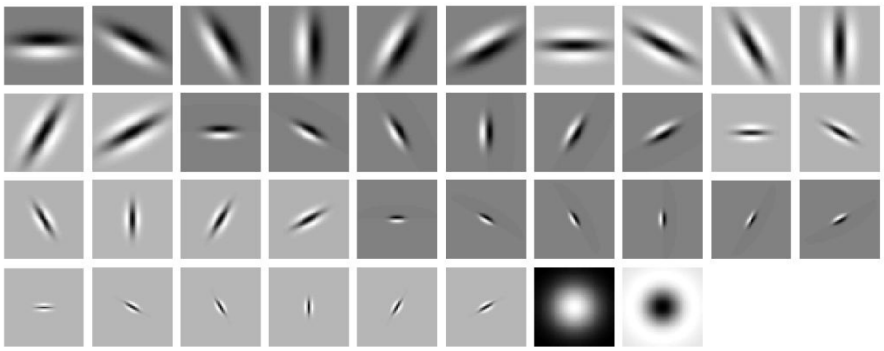


Figure 3.3: The linear filters in the MR8 filter bank. Image taken from [VZ02].

[LM01]. These filters were not designed to mimic the human visual system. Therefore, they contain many differences with filter banks that were. For example, compared to the filter bank used by Malik & Perona [MP90], the MR8 filter bank 1) contains first derivative filters as well as second derivative filters, 2) performs full-wave rectification, in which the absolute value of the filter responses are taken, instead of half-wave rectification, and 3) performs response normalization based on contrast instead of inhibition of spurious responses. Also, the scale and orientation of the MR8 filters where chosen for discrimination, unlike the filters used to define some other types of spatial filters. For example, steerable image pyramids chose filter scale and orientation based on the efficient representation and computation of the image description [HB95].

Hayman made a slight modification in his implementation of the MR8 filter bank [HCFE04]. He used a filter support of 41×41 instead of 49×49. I refer to this as the MR8-2W filter bank. In Section 3.2, I use two additional, slight modifications of the MR8. First, I change the scale of the two isotropic filters to $\sigma = 2$ and the three scales of the other filters to $(\sigma_x, \sigma_y) = \{(0.5, 1.5), (1, 3), (2, 6)\}$. The resulting more local filter bank I refer to as MR8-3W. Second, I normalize each filter response by the maximum attained over the training set instead of the normalization based on Weber's Law. This change can be made with all of the versions above, making the MR8-1M, MR8-2M, and MR8-3M versions of the filter bank.

Classification

So far Section 3.1.2 has discussed general texture analysis techniques for the representation of textures. Now, their actual classification is discussed. The results of these classification schemes are discussed in Sections 3.2 and 3.3.

Many of the methods discussed above use histogram based distribution estimates of dense local features [MBLS01, LM01, CD01, VZ02, HCFE04, PNMT04]. As mentioned earlier, histogram based representations tend to be high dimensional and tend to have nonlinear variation in the set of textures that comprise each class. Parametric classification is very difficult under these conditions [DHS01]. High dimensional data is prone to overfitting; standard parametric classifiers assume linear class variation. The approaches above recognize the inappropriateness of parametric classification for histogram based representations and instead use non-parametric, distance based classifiers, such as nearest neighbor (NN) and support vector machines (SVMs), with a nonlinear distance measure.

The NN classifier is a well known and often used classification method [DHS01]. NN classifiers use a specified distance measure, which allows methods to account for the specific nonlinear variations in their application. The χ^2 distance measure combined with a 1-NN classifier is prevalent in recent texture classification work [MBLS01, LM01, VZ02, VZ03]. Pietikäinen *et al.* use a log likelihood measure [PNMT04]. Many different distribution distance measures have been used in computer vision. Several were discussed in Section 2.1.3; see [PRTB99] for a survey. Sections 3.2 and 3.3 perform classification based on the EMD.

Several modifications to NN classification have been proposed. NN can be computationally expensive since the classification of each target texture requires commuting a distance to every training texture. One approach to alleviate this is to remove unnecessary training samples that do not significantly contribute to classification [VZ02, PNMT04]. Levina estimated the marginal distributions of several features and argued that each should be considered independently. This led Levina to estimate a target texture's distance to a class as the product of several independent NN computations [Lev02]. Recently, Varma & Ray described a method to enhance NN classification with a per class, isotropic variance for a given distance measure [VR07]. Such a method that learns a class's second-order statistics could be combined with a

distance based method to learn the class's first-order statistics, *i.e.*, its intrinsic, or Fréchet, mean [Fre48, Fle04]. The classification methods discussed in Sections 3.2 and 3.3 estimate for each class a mean and a rich, non-isotropic covariance structure.

Moving beyond NN classification given nonlinear variation, however, is both computationally and theoretically difficult. One popular approach is to find nonlinear decision boundaries using kernel support vector machines (kernel SVMs), which map the input space to a higher dimensional feature space in which class-separating hyperplanes are found. Hayman *et al.* used kernel SVMs to extend Varma & Zisserman's filter bank classifier, improving results and reducing the number of stored vectors [HCFE04]. However, kernel SVMs are computationally intensive and lack a principled approach to kernel selection. Recently, class-specific kernel and feature selection methods have been shown to improve kernel SVM results on KTH-TIPS2 [CHM05].

Other approaches to handling nonlinear variation in the computer vision community try to find a linear parameterization of the space. Data specific nonlinear manifolds can be learned using methods such as Isomap [TdSL00] or local linear embedding [RS00], but this is a difficult and computationally intensive task. For texture classification, Cula & Dana represent a class by a one-dimensional nonlinear manifold in a histogram based space. Classification is performed by finding the minimum Euclidean distance to the one-dimensional curves [CD04].

3.2 Filter Bank Based Classification

Section 3.1 reduced the representation and classification of homogeneously textured images to the representation and classification of multivariate distributions of local texture features. This allows the techniques discussed in Chapter 2 for the representation and classification of multivariate distributions to be directly applied. In this section, the local texture features of interest are the responses of the MR8 filter bank described in Section 3.1.2; Section 3.3 uses local features composed of simple pixel intensities. Both Sections 3.2 and 3.3 present results on the experiment on CUReT discussed in Section 3.1.1.

3.2.1 Implementation

The implementation of the methods presented in Chapter 2 to this task starts by applying the MR8 filter bank to 92 normalized, CUReT images for each of the 61 classes. Each image is then a multivariate distribution estimate composed of an empirical distribution of $200 \times 200 = 40,000$ points in an 8-dimensional space. Each empirical distribution is then reduced to 8 quantile functions from each filter response marginal distribution. Typically 32 bins per QF are used, which creates a 256-dimensional vector representation of each image.

The marginal distributions use the simplest strategy for choosing projection directions, which was discussed in Section 2.2.1. The results in Section 3.2.2 show that filter marginals are adequate and efficient on CUReT. In Section 3.3, more complicated projection strategies are required since the marginal distributions of interest are identically distributed. However, Section 3.3.2 discusses the equivalence of the approaches taken here and those taken in Section 3.3.2.

The 92 image in each class are split into equally sized training and test sets. PCA is used on the training set to estimate a per class mean and covariance matrix. Examples of this representation and its first and second order statistics are given in Figure 3.4. These examples, unlike those used for classification, are per filter and use all 92 images. The first mode captures a high percentage of each class's variation, and its linear path matches the samples. For the classes in CUReT, QFs form approximately linear spaces with low inherit dimensionalities, which can be compactly represented using PCA.

3.2.2 Results

Quadratic discriminant analysis (QDA), as discussed in Section 2.3.2, is now used to classify the distribution estimates given in Section 3.2.1. QDA classification using these QF estimated filter marginals is termed QF-QDA. QF-QDA is first compared to previous classification results on CUReT. Next, the sensitivity of QF-QDA to the number of training samples, QF bins, and principal modes is examined. Unless otherwise stated, all results use 32 values per QF, use leave-one-out cross-validation on the training set to determine a common expected projection error across classes, and use 46 training images per class. All results are averaged over 100

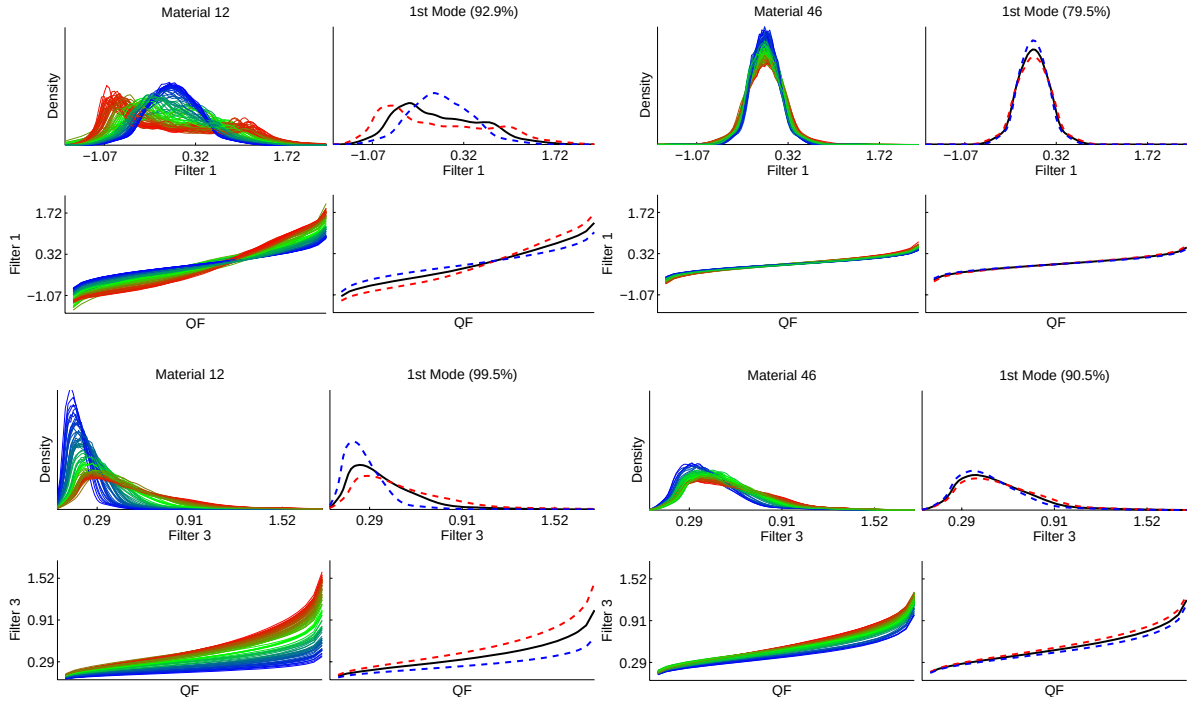


Figure 3.4: Filter responses from all 92 images in materials 12 (left) and 46 (right). Responses are shown for two filters in MR8-3M. On top is the Gaussian filter. On bottom is the edge filter with $(\sigma_x \sigma_y) = (0.5, 1.5)$. QFs and histograms are given for both the 92 responses and their mean and ± 1 standard deviation along the first eigenvector. The coloring of the histograms is set by each QF's projection coefficient onto the first mode. In parenthesis is the relative variance of each class in the first mode.

random training and test splits.

Comparative Results

Table 3.1 shows the accuracy of QF-QDA using the original and modified MR8 filter banks described in Section 3.1.2. The primary interest of Table 3.1 is in its comparison with previous methods, though the filter bank modifications are also discussed. Two of the results in Table 3.1 are directly comparable with previous methods. Using MR8-1M, QF-QDA achieves an accuracy of 99.00% versus the 97.43% achieved by Varma & Zisserman [VZ02]. Using MR8-2M, QF-QDA achieves an accuracy of 99.12% versus the 98.46% achieved by Hayman *et al.* [HCFE04]. These comparisons use the QF-QDA results with maximum based response normalization instead of Weber's Law based response normalization as in [VZ02, HCFE04]. However, I feel this comparison is fair because the maximum based normalization is simpler

Table 3.1: QF-QDA classification accuracy on CUReT for three versions of the MR8 filter bank and two normalization schemes. MR8-1 and MR8-2 allow a direct comparison with Varma & Zisserman (97.43%) [VZ02] and Hayman (98.36%) [HCFE04], respectively.

	Normalization scheme	
Filter Bank	Weber's Law (W)	Maximum in Training (M)
MR8-1 (Varma & Zisserman)	98.73% ± 0.52	99.00% ± 0.43
MR8-2 (Hayman)	98.92% ± 0.41	99.12% ± 0.39
MR8-3 (proposed)	99.48% ± 0.30	99.60% ± 0.27

and the Weber's Law based normalization seems detrimental to the QF representation. A comparison of the two normalization schemes for QF-QDA is given in Table 3.1.

Using MR8-3M improves accuracy to 99.60%. QF-QDA achieves a near perfect accuracy with all three versions of MR8, which outperforms all other known existing methods. Pietikainen *et al.* perform a similar experiment on CUReT using multi-scale, rotationally invariant local binary patterns (LBPs) with nearest neighbor (NN) classification to achieve an accuracy of 96.55% [PNMT04]. The results of Cula & Dana are not easily compared since their experiments did not use all 61 material classes [CD04]. Later in this section the methods of [VZ02, CD04, HCFE04, PNMT04] are also compared to the QF-QDA approach in terms of compactness and computational complexity.

The MR8 filter bank is a modification of a filter bank designed by Leung & Malik [LM01]. Both were designed and first applied to CUReT. Despite these efforts, QF-QDA achieves better results using MR8-3 than using MR8-1. While this result could be specific to QF representations, I believe it would also hold true for the histogram based representations used by the designers of MR8. If this is true, the MR8-1 filter bank is not ideal for CUReT. Specifically, more local features like those introduced in MR8-3 are useful. These results correspond to findings in Section 3.3, which show that extremely local features are all that is required to distinguish the materials in CUReT.

Figure 3.5 gives classification results for the smaller training set sizes of 12 and 23. Hayman *et al.* gave these results for their proposed SVM classifier and for their implementation of Varma & Zisserman's NN classifier [HCFE04]. A single training and target split was used, where the images were chosen in an alternating order to provide an even distribution of the viewing and

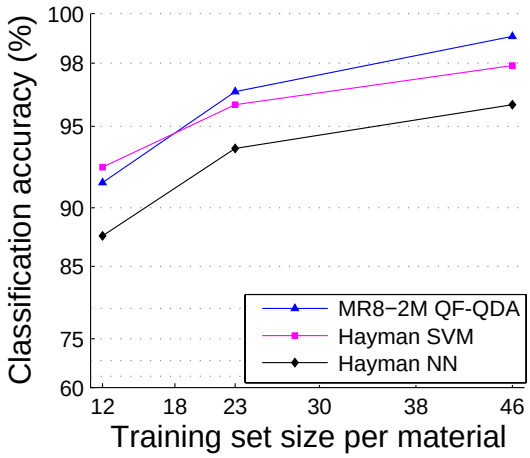


Figure 3.5: A comparison of QF-QDA using MR8-2M with Hayman *et al.*'s NN and SVM classifiers [HCFE04]. Training set size is varied from 12, 23, to 46 and results are reported for a single training and target split.

illumination directions between the training and target splits. Equivalent results using QF-QDA with MR8-2M are given in Figure 3.5 to allow a direct comparison with Hayman *et al.*'s results. QF-QDA consistently outperforms Hayman's NN classifier. However, Hayman *et al.*'s SVM classifier appears to be more efficient for extremely small training sizes. For training set sizes of 12 and 23, the differences between QF-QDA and the SVM classifier are roughly 1% and 0.6%, respectively. These results are given for a single training and test split. Given random splits, QF-QDA's performance varies with standard deviations of roughly 2% and 1% for these training set sizes. Therefore, the statistical significance of the comparison between QF-QDA and Hayman *et al.*'s SVM classifier is difficult to judge. To further analyze the behavior of QF-QDA for various training sizes, I next consider NN, SVM, and QDA classification using the QF based distribution estimates.

Training Set Size

I now explore NN, SVM, and QDA classification using the QF based distribution estimates. The NN classifier is a 1-NN classifier and is termed QF-NN. The SVM classifier uses linear decision boundaries and is termed QF-SVM. Figure 3.6 (left) shows the sensitivity of QF-NN, QF-SVM, and QF-QDA to training set size. The error bars in Figure 3.6 represent one standard deviation in the results from the 100 random training and target splits. QF-QDA

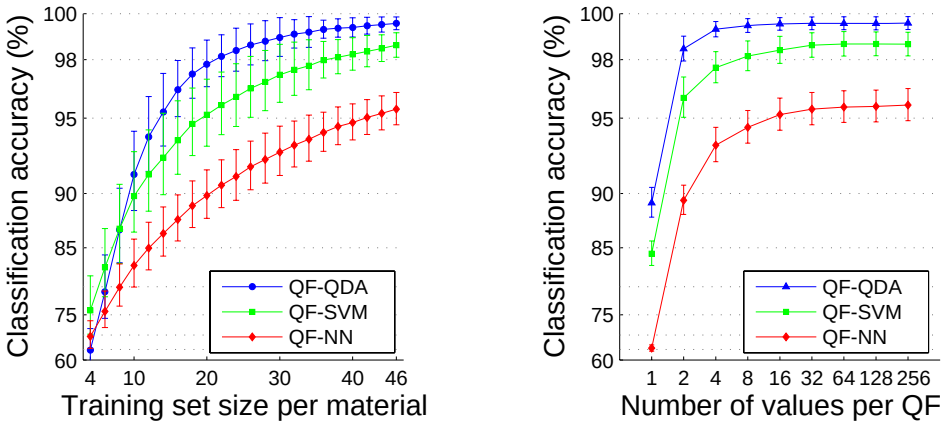


Figure 3.6: Varying the number of images available during training (left). Varying the size of each marginal's QF (right). The accuracy of QF-QDA increases faster than both QF-NN and QF-SVM as more of both types of information are introduced. The MR8-3M filter bank is used.

quickly becomes the most accurate method as the size of the training set is increased. Given 4 training samples, QF-QDA performs the worst; given 8, it is equivalent to QF-SVM. Cross-validation limits QF-QDA for small training sets. For example, given 4 training samples, cross-validation limits QDA to using 1 of the 2 available principal components.

QF-SVM used the *libSVM* library to perform SVM training and classification [CLvm]. The parameter C is set through cross-validation. I also tried a nonlinear SVM kernel, radial basis functions, but accuracy remained within 0.1% of linear SVM. While a more appropriate kernel may exist, a fundamental difference between SVM and QDA is that SVM places class boundaries with an equal margin on each side while QDA scales the distance metric for each class.

Quantile Function Size and Compactness

Figure 3.6 (right) shows the sensitivity of the three QF based classifiers to QF size. In all the other experiments this has been set to 32. As QF size is varied from 4 to 256, QF-QDA achieves a consistent accuracy of over 99%. QF-QDA also achieves the good classification results of 89.25% and 98.50% using simple 1 and 2 bin QFs, respectively. This shows that most of the information needed for classification is encoded in 2 QF values, which are linearly equivalent to mean and standard deviation. This finding is in contrast with [LM01, VZ02, CD04], which

have focused on rich, histogram based models of the joint variation of the filter responses. Here, 4-bin QFs for each marginal are shown to almost completely characterize the materials. For the MR8-2W and MR8-3M filter banks, 4 bins have an accuracy of 98.61% and 99.35%, and 32 bins have an accuracy of 98.92% and 99.60%, respectively. In both cases, using more than 4 bins only increases accuracy by approximately 0.3%. This consistent and gradual improvement of accuracy with QF size demonstrates that larger QFs gradually incorporate useful information about the marginal distributions. The accuracy of QF-QDA using a small number of bins shows that QFs are a compact, descriptive representation. The accuracy of QF-QDA using a large number of bins shows that QFs do not dramatically suffer from "over-binning". Section 3.3.2 gives similar results using simpler linear filters, which demonstrates that these properties are not specific to the MR8 filter bank.

Using 4 and 32 bin QFs for each of the 8 feature marginals yields 32 and 256 dimensional representations per image. QF-QDA learns an even more compact representation for the images in each class. On average, cross-validation keeps 18 and 32 principal modes for the 4 and 32 bin QF cases. For the 32 bin case, accuracies over 98% are achieved for MR8-3M and MR8-2W using 6 modes minimum, 15 on average, and 12 modes minimum, 23 on average, respectively. Recall that the classes are typically constrained to have an equal projection error across classes, parameterized by the minimum number of modes.

These QF based representations are more compact as well as more accurate than previous methods. Varma & Zisserman [VZ02] and Hayman *et al.* [HCFE04] report their best results of 97.43% and 98.46% using 2440 bin texton histograms. The NN classifier achieves accuracies of 96.93% [VZ02] and 96.1% [HCFE04] with 610 and 200 textons, respectively. The SVM classifier achieves an accuracy of 97.9% with 200 textons [HCFE04]. The QF based representation is approximately two orders of magnitude more compact. Pietikainen *et al.*'s multi-resolution, rotationally invariant LPBs form a compact set of 54 binary masks [PNMT04]. Pietikainen *et al.* achieve compactness by allowing only a few discrete feature values while QFs compactly describe a set of continuous feature values.

Hayman *et al.* report that either SVM classifier reduces the size of the histogram based training model by 10% - 20% [HCFE04]. Cula and Dana [CD04] used a joint PDF representa-

tion similar to [LM01, VZ02, HCFE04], but they additionally performed dimension reduction using PCA. They report good performance given at least 300 textons and 70 principal modes, where PCA reduces the model by approximately 75%. As mentioned above for the QF representation, QDA reduces the 256 and 32 dimensional representation for each image to approximately 15 - 30 and 15 values, respectively. This is a reduction of about 90% - 95% for the 32 bin QFs and 50% for the already compact 4 bin QFs. The better PCA based compression of the QF distribution representation over the histogram representation reinforces the believed strong linearity and low inherent dimensionality of the QF representation.

The success of low bin count and PCA compression with QFs can also be considered in terms of known parametric representations of the local texture features. As mentioned in Section 3.1.2, Geusebroek has shown that stochastic textures have marginal filter responses that follow the Weibull distribution [GS05]. The variation of stochastic textures affect the scale and shape parameters of the Weibull while the scale parameter is fixed at zero. If QFs were equally ideal for representing such distributions, 2 bins (values) would also be all that is required. As mentioned in Chapter 2, scale is a linear QF parameter, and the Weibull shape parameter is exponential (see Figures 2.10 and 2.12 on pages 30 and 34). With 4 QF bins the scale and shape parameters of the Weibull can be well estimated. The connection of marginal filter responses with the Weibull distribution is consistent with the success achieved when using a small number of QF bins. The Weibull distribution also gives insight to the success of using PCA. In the space of QFs, Weibull distributions live on a two-dimensional submanifold that is linear in one dimension and exponential in the second. PCA estimates a locally linear subspace around each class's average distribution. The success of the approach shows that this is an adequate approximation. Recall, however, that many of the textures in CUReT have a mix of stochastic and structural properties. Therefore, distributions other than the Weibull are of interest. QFs nicely handle this situation since they can compactly represent any set of distributions with variation similar to the Weibull.

Computational Complexity

The slowest part of the MR8 QF-QDA classifier is in the two preprocessing steps. First, computing the filter bank responses for all 5612 images takes approximately 7 hours for MR8-2 and 2 hours for MR8-3. This step requires convolving each image with 38 filters. MR8-2 is more complex than MR8-3 since MR8-2 uses a fixed filter support of 41×41 while MR8-3 uses a maximum filter support of 25×25. For each of the $200 \times 200 = 40,000$ pixels in each image, the MR8-2 convolutions require $8 * 41^2 = 13,448$ operations. This step could be more efficiently done in the Fourier domain. Second, computing the quantile function representations for all 5612 images takes approximately 10 minutes. This step requires sorting the 8 sets of filter responses for each image . This corresponds to roughly $8 * \log_2(40,000) = 122$ per-pixel operations, since sorting is a $O(n \log n)$ algorithm.

For each training and test split, training the QDA classifier takes approximately 0.3 seconds, and QDA classification takes approximately 15 seconds. For all 100 splits, this adds up to training and classification times of about 30 seconds and 30 minutes, respectively. Cross-validation adds approximately 5 minutes to the training time. Timing is given on a 3.4 GHz Pentium® 4 with 2 GB of RAM using MATLAB®.

QF-QDA training is computationally inexpensive. Previous histogram methods performed k-means clustering in at least an 8-dimensional space [LM01, CD04, VZ02, HCFE04]. In contrast, QF-QDA computes 8 independent partial sortings. Additionally, the PCA computations required for QDA are less expensive than the computations required for SVM training, though neither is overly burdensome due to the compact representations.

Classification time is greatly impacted by compactness. NN and QDA classifiers have a classification time that is linear in the size of the training set. For NN this is clear since a distance must be computed to every training model. For QDA, even though only a single Mahalanobis distance is computed to the class average, a Mahalanobis distance requires projection onto each principal component. Computationally, each projection is comparable to a Euclidean distance computation between two models. Since QF-QDA learned a texture model that is at least two orders of magnitude more compact than those in [VZ02, HCFE04], classification speed is two orders of magnitude faster. Each classification result given in this

section, which is averaged across 100 splits, would have taken approximately 50 hours, instead of 30 minutes, for the histogram based approaches in [VZ02, HCFE04].

Parameterization of the Number of Principal Components

As mentioned in Section 2.3.4, a parameter of the QDA model is the minimum allowed number of principal components, which determines the common expected projection error across classes. I now compare this parameter to an alternative global constraint parameterized by the number of common modes across classes. Table 3.2 gives classification results comparing the two approaches. Equal projection error improves classification results over equal component number by 0.5% when using MR8-1W and MR8-2W. For MR8-3M there is not much room for improvement.

To ensure that this improvement is not an artifact due to cross-validation, Figure 3.7 shows the difference between equal projection error and equal component number for several values of their parameters. Figure 3.7 (left) gives equal projection error results in terms of its parameter, the minimum number of components across classes. Figure 3.7 (right) compares the two approaches in terms of the average number of per class components. The bigger dot on each curve is the approximate position found through cross-validation. Figure 3.7 shows that cross-validation works equally well for the two approaches. Equal component number is more efficient given a small of the number of modes, but equal projection error has a higher maximum accuracy. I hypothesize that this increased accuracy is a result of the equal projection error method assigning more components to the classes with more variability, which avoids overfitting for as long as possible. Figure 3.7 also shows that good performance is achieved for a wide range of values, which demonstrates that QF-QDA is relatively insensitive to either parameter.

Conclusions

Section 3.2 presented classification results using the QF-QDA classifier with the MR8 filter bank. The results were shown to be accurate, compact, and insensitive to its two parameters, the number of QF bins and the common projection error across classes. QDA was also shown

Table 3.2: Classification accuracy of QF-QDA for two schemes of selecting the number of components across the classes. In parenthesis is the parameter found during cross-validation, which corresponds to the average and minimum number of per class modes, for equal number and equal projection error, respectively.

	Selection scheme for number of components	
Filter Bank	Equal Number	Equal Projection Error
MR8-1W	98.27% ± 0.68 (31.5)	98.73% ± 0.52 (30.7)
MR8-2W	98.47% ± 0.61 (32.6)	98.92% ± 0.41 (31.6)
MR8-3M	99.54% ± 0.32 (33.9)	99.60% ± 0.27 (30.4)

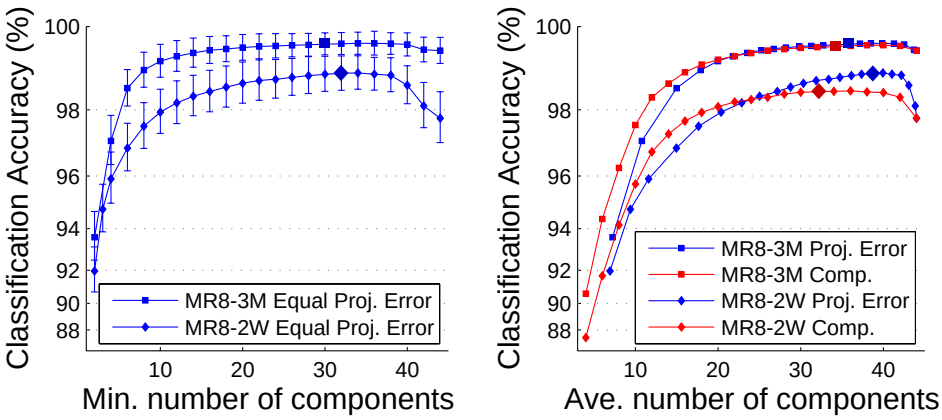


Figure 3.7: Left: Varying the minimum number of principal components learned by QF-QDA when the expected projection error is kept constant across the classes. Right: A comparison of keeping projection error (blue) versus component number (red) constant across classes. The larger dot on each curve is the point found through cross-validation.

to perform better than SVM and NN for all but extremely small training sets.

The MR8 filter bank, the current top-of-the-line for CUReT, was demonstrated to be less than ideal; the materials in CUReT are better distinguished by more local features. This highlights the difficulty in constructing an ideal filter bank. This finding, combined with the results presented in Section 3.3, shows that the reliance of QF-QDA on a particular filter bank is a limiting factor. Also, the computation of the MR8 filter responses is expensive compared to the rest of the method.

The difficulties of filter bank selection and computation are addressed in Section 3.3 using MRF based texture models, which use the simpler distributions of intensities in a pixel neighborhood. Section 3.3.1 presents a texture method related to GLCMs and the Strong-MRF assumption. This model, like the filter bank based model presented in this section, has the

advantage of using unsupervised local features. The filter bank model uses features preselected for a specific task while the MRF model in Section 3.3.1 uses general features based on pairwise pixel interactions. Section 3.3.2 also presents a texture model based on non-preselected features that is able to capture multiple pixel interactions. It produces features by learning projection directions in the joint space of a pixel neighborhood. This approach is shown to be equivalent to supervised filter learning. Both MRF approaches are compared to the filter bank approach in terms of accuracy, compactness, and sensitivity to training set size.

3.3 Markov Random Field Based Classification

Markov random fields (MRFs) estimate probabilities of complete local neighborhoods of pixel intensities. Filter banks are bypassed by directly modeling pixel intensities. Small, compact neighborhoods are examined, of sizes 1×1, 3×3, 5×5, and 7×7. I give results using the 1×1 neighborhood, which is simply the image's marginal distribution of pixel intensities, for comparison when no texture features are available. Each image gives an empirical distribution estimate for each of these 1-, 9-, 25-, or 49-variate probability distributions. As in Section 3.2, each image is a multivariate distribution estimate. Therefore, the methods presented in Chapter 2 can once again be directly applied for their representation and classification. This section reports classification results on the task evaluated in Section 3.2.

In Section 3.2, the multivariate distribution of filter responses was represented by its marginal distributions. This simple strategy will not work for a multivariate distribution of pixel intensities since their marginals are identically distributed. Two alternative representations are considered in this section. First, Section 3.3.1 uses the multivariate distribution representation presented in Section 2.2.2 based on conditional distributions. Section 3.3.1 discusses the equivalence of this approach to GLCMs and second-order Strong-MRF models [Pag04]. Second, Section 3.3.2 uses the multivariate distribution representation presented in Section 2.2.1 based on projection directions learned through PCA. Section 3.3.2 discusses the equivalence of this approach to learning a set of linear filters. The accuracy of the model presented in Section 3.3.1 is shown to be limited by its pairwise pixel features, while the model presented

in Section 3.3.2 is shown to be as accurate as the hand-tuned MR8-3M filter bank.

3.3.1 The Conditional Distribution Representation: Second-Order Strong-MRFs

The multivariate distribution representation given in Section 2.2.2 assumes the conditional independence of intensities in a pixel neighborhood given the center pixel's intensity. Let x be the intensity of a pixel and let $y_1, \ldots, y_n$ be the intensities of x's neighboring pixels. In Section 2.2.2, it was shown that $p(x, y_1, \ldots, y_n)$ can be rewritten as $p(x) \cdot \prod_{i=1}^{n} p(y_i|x)$ using Bayes rule and conditional independence. This is exactly the equation used by Paget to describe second-order Strong-MRF models [Pag04]. Paget describes Strong-MRF models in detail and demonstrates that they capture sufficient information for the synthesis of some natural textures. The Strong-MRF model simplifies the MRF model by additionally assuming that a neighborhood's cliques are independent. This allows a neighborhood's joint intensity distribution to be reduced to the product of the clique probabilities. The equation above corresponds to a second-order Strong-MRF model since it is constructed from the neighborhood's first-order (single pixel) and second-order (two pixel) cliques.

Second-order Strong-MRF models are also related to gray level co-occurrence matrices; both only capture pairwise pixel information. Specifically, $p(x) \cdot \prod_{i=1}^{n} p(y_i|x)$ consists of the first-order distribution $p(x)$ and n GLCMs $p(y_i|x)$ at specific pairwise spatial relationships. The representation of each $p(y|x)$ given in Section 2.2.2 can be viewed as an alternative, QF based representation of GLCMs. This representation is typically much more compressed than the full histogram of pairwise intensities, and it has a Euclidean distance related to the EMD.

Figure 2.13 on page 46 depicts the representation of each $p(y_i|x)$. Each conditional distribution is represented by $j \times k$ bins, where x is divided into k quantiles. For each of these k conditions, j-bin QFs of y_i are estimated. Typically, $j = 4 \times k = 4$ bins are used for each conditional distribution. These are well estimated from the approximately 40,000 sample empirical distributions estimated from each image. Each conditional distribution is an independently estimated local feature, much like each filter response used in Section 3.2.

Next, the construction of multi-scale neighborhoods are discussed before classification re-

sults using this texture model are presented.

Multi-scale Neighborhoods

This section describes an MRF texture model that uses compact pixel neighborhoods of sizes 3×3, 5×5, and 7×7, with 9, 25, and 49 local texture features, respectively. While this model has a much smaller spatial extent than filter bank methods, it quickly produces more features as neighborhood size is increased. This quadratic increase in the number of local features can become computationally prohibitive. Therefore, I also explore multi-scale neighborhoods to more compactly increase spatial extent. Pioneering work on multi-scale image representations was done in [HB95].

I define a pixel's multi-scale neighborhood to include the original 3×3 local neighborhood. Then, I use a Gaussian filter with $\sigma = \sqrt{2}$ to generate successively blurred images with pixels that summarize progressively larger spatial extents. At each scale, a 3×3 neighborhood is defined by doubling the distance between each of these 9 pixels and the center pixel.

Results

Classification results are now presented using this Strong-MRF model on the experiment discussed in Sections 3.1 and 3.2. As in Section 3.2, unless otherwise specified all results are averaged over 100 random training and target splits, cross-validation is performed to estimate the common projection error across classes, and training uses 46 images per class.

Table 3.3 summarizes the classification accuracy of QF-NN and QF-QDA for different neighborhood sizes [1]. As found in Section 3.2, QF-QDA consistently outperforms QF-NN. QF-QDA achieves the high accuracy of 98.24% using a 3×3 neighborhood and accuracies $> 99\%$ for larger neighborhoods. These results use 4×4 bins for each conditional distribution, which constructs 16, 144, 400, 784, 288, and 432 dimensional vector representations of each image for 1×1, 3×3, 5×5, 7×7, 2 scale 3×3, and 3 scale 3×3 neighborhoods, respectively.

Compared with the MR8 QF-QDA results from Section 3.2, the Strong-MRF results using neighborhoods larger than 3×3 surpass the results using the original MR8-1 and MR8-2 filters

[1] An early version of these results where presented in [Bro05], where $n \times 1$, instead of $n \times n$, neighborhoods were computed due to a programming error.

Table 3.3: Classification accuracy of QF-NN and QF-QDA using second-order Strong-MRF texture features. The results use 4×4 bins for each conditional distribution, 4 QF bins for each of 4 conditions.

Neighborhood Size	Strong-MRF QF-NN	Strong-MRF QF-QDA
1x1	63.05 ± 1.72	65.66 ± 1.57
3x3	89.19 ± 1.16	98.24 ± 0.58
5x5	93.12 ± 1.01	99.31 ± 0.41
7x7	94.58 ± 0.87	99.43 ± 0.34
3x3, 2 Scales	93.33 ± 1.01	99.33 ± 0.38
3x3, 3 Scales	95.04 ± 0.93	99.55 ± 0.35

and are equivalent to results using the hand-tuned MR8-3 filters. These results also surpass the previous best MR8 filter bank based results, the 98.46% achieved by Hayman's SVM classifier [HCFE04].

The results summarized in Table 3.3 can also be compared to previous MRF based texture models. Varma & Zisserman constructed several MRF models that, like their method based on the MR8 filter bank, estimate the joint distribution of a neighborhood's intensities through clustering [VZ03]. Their best model clusters in the joint space of a pixel neighborhood without the center pixel. Then for each cluster the univariate distribution of the center pixel's intensity is modeled with a histogram. This model with a NN classifier achieves an accuracy of 95.87%, 97.22%, and 97.47%, using 3×3, 5×5, and 7×7 neighborhoods, respectively. These results use 610 textons and 90 bins for the center histogram, for a 54,900 dimensional representation. Their best classification result of 98.03% uses 2440 textons (219,600 values) and a 7×7 neighborhood. The results presented in Table 3.3 are more accurate and compact.

Good classification results are also achieved by the Strong-MRF QF-QDA classifier when modeling each conditional distribution with fewer than 4×4 bins. For example, the 3-scale, 3×3 strong-MRF QF-QDA classifier achieves an accuracy of 99.47% using 1×4 bins. This model, which only computes the mean of 4 conditions for each conditional distribution, constructs a more compact, 108 dimensional vector representation for each image.

Figure 3.8 shows the accuracy of the 3-scale, 3×3 Strong-MRF QF-NN and QF-QDA classifiers as training set size and the size of each conditional distribution is varied. The right graph shows that for smaller training set sizes, the Strong-MRF QF-QDA classifier performs

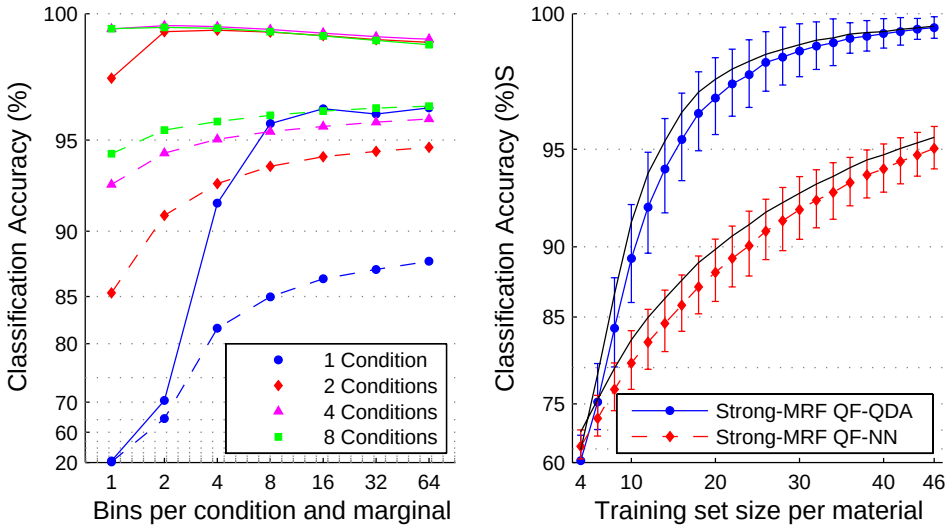


Figure 3.8: Left: Varying the number of QF and conditional bins for the 3 scale, 3x3 Strong-MRF QF-QDA (solid lines) and QF-NN (dashed lines) classifiers. Right: Varying the number of images available during training, using 4×4 bins. The MR8-3M QF-QDA and QF-NN results are in black.

similarly to but not as accurately as the MR8-3M QF-QDA classifier. The left graph shows that for the 3-scale, 3×3 conditional distributions, only a small number of conditions and QF bins are required. QF-QDA achieves an accuracy of 96.4% using 1 condition per distribution, which only models the three multi-scale, marginal intensity distributions.

The Strong-MRF QF-QDA model achieves excellent classification results for small neighborhood sizes. However, compared to the MR8 QF-QDA classifier, it is not as compact, and it does not perform as well for smaller training set sizes. The Strong-MRF model also requires 2 parameters for each conditional distribution to be specified. These issues are addressed by the second MRF model presented next in Section 3.3.2. The accuracy and compactness of these MRF models are discussed further at the end of Section 3.3.2.

3.3.2 The PCA Based Projections Representation: Learning a Linear Filter Bank

Section 2.2.1 described a multivariate distribution representation that uses PCA to select a set of orthogonal vectors within that distribution. The representation is a concatenation of QFs estimated from samples projected onto each vector. This section uses this multivariate

distribution representation to represent pixel intensities in compact neighborhoods. In this context, the learned vectors correspond to linear filters, as discussed in more detail below. I term the model built using these local features to be the PCA-MRF texture model.

PCA-MRF can be compared to the MR8 filter bank. PCA-MRF is computationally more complex but more straightforward. It is more complex since it must learn its filters during training. Also, PCA-MRF learns many filters, n^2 filters for an $n \times n$ neighborhood. PCA-MRF combined with QF-QDA leads to a straightforward classification algorithm compared to classification using the MR8 filter bank for several reasons. First, PCA-MRF uses small 3×3 and 5×5 pixel neighborhoods. This is in contrast to MR8, which uses filters computed from 49×49, 41×41, and 25×25 neighborhoods for the MR8-1, MR8-2, and MR8-3 filter banks, respectively. Second, PCA-MRF does not require a preselected filter bank. Third, PCA-MRF requires few normalization steps. The MR8 model uses images with a normalized intensity distribution, filters that are L_1 normalized, and responses normalized by either Weber's Law or the max. achieved in training. PCA-MRF uses L_2 normalized filters and unmodified responses. At the end of this section, image normalization is discussed and QF-QDA classification using PCA-MRF is shown to not require such normalization. Later, the results section compares classification accuracy using both models.

PCA-MRF can also be compared to the Strong-MRF model. PCA-MRF is not restricted to pairwise pixel features like the Strong-MRF model. In the results section, the PCA-MRF features are shown to be more discriminative than the Strong-MRF features. This allows smaller neighborhoods to be used. However, learning these features increases training complexity. Also, PCA-MRF has a single parameter for QF bin count while the Strong-MRF model has two parameters.

The Learned Linear Filters

Section 2.2.1 presented a multivariate distribution representation based on multiple projections. For the representation, it was shown that PCA computes an ideal set of projection directions. The directions produce maximally uncorrelated coefficients, which minimizes information loss when QFs are independently built for each projection.

Linear projections have special meaning for distributions of intensities in a pixel neighborhood. Projection onto a vector is a linear function of a pixel neighborhood, an identical operation to a linear image filter. Therefore, a vector is a L_2 normalized linear filter and projecting all the samples of a distribution onto a vector is equivalent to image convolution. In this context, PCA can be considered as an approach to learning a task-specific filter bank composed of minimally correlated, orthogonal linear filters.

In this section, a common set of filters is learned across classes. They are computed using PCA on a random sample of 400 pixels from every training image pooled across classes. Figure 3.9 shows all 9, 25, and 49 filters learned for 3×3, 5×5, and 7×7 neighborhoods, in order of decreasing eigenvalues across the rows. PCA seeks directions that best represent the samples. Such generative directions have similar goals as lossy image compression techniques. There is a strong resemblance between the learned directions and the discrete cosine transform (DCT), which is used in jpeg image compression. Figure 3.10 shows the DCT for an 8×8 image patch. Both methods generate orthogonal, nonlocal vectors, where local vectors are constrained to a portion of the image patch.

The filter banks used for texture classification have distinctly different properties from the vectors found through PCA. Many of these differences arise from filter bank design being focused on discrimination while PCA focuses on generative vectors. Filter banks are typically spatially smooth and have locality about the center pixel. They are often selected to have certain invariances, such as rotational invariance. Filter banks are also often not constrained to have linear responses. The MR8 filter bank, for example, takes as a response the maximum of several filters. These properties of preselected filter banks generally tend to increase their discriminative power, especially when training sets are limited. Limited training sets could also prevent PCA from learning sufficiently general directions. The sensitivity of PCA-MRF to training set size is examined later in this section. Many of these desirable properties of preselected filter banks could be incorporated into a more compex, PCA based learning process.

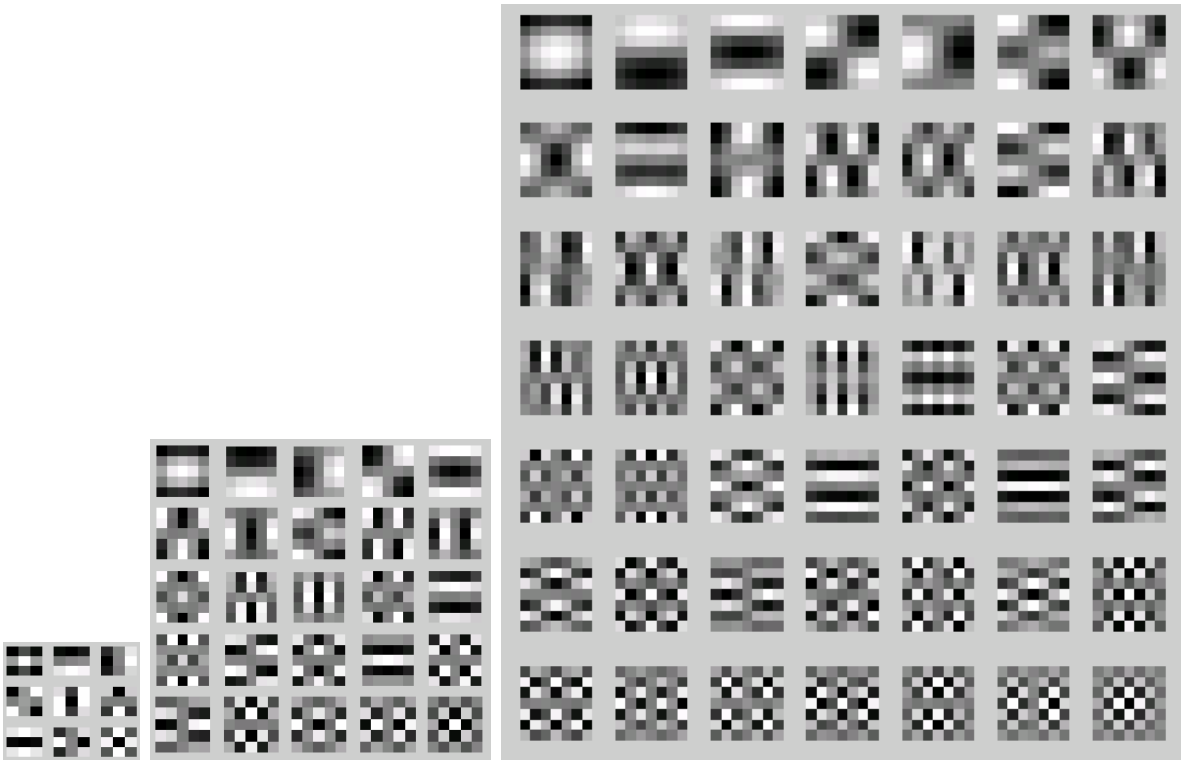


Figure 3.9: All 9, 25, and 49 principal directions found by applying PCA to 3×3 (left), 5×5 (center), 7×7 (right) image patches pooled across classes. Each set represents both a learned filter bank and an uncorrelated, orthogonal basis.

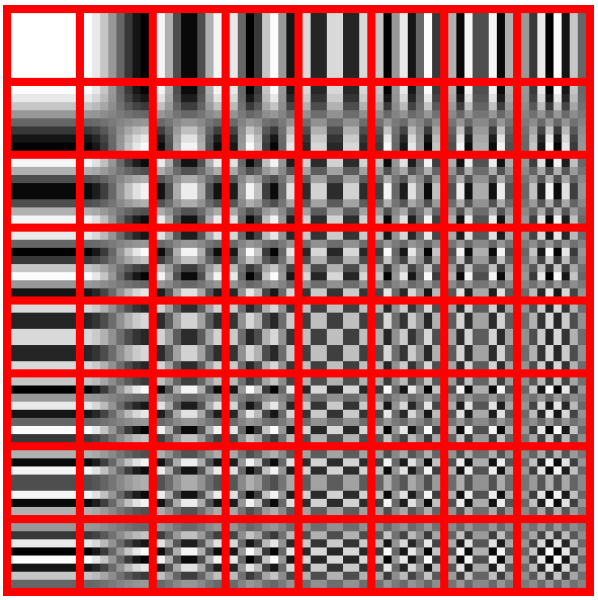


Figure 3.10: The discrete cosine transform for an 8×8 image patch. Image taken from [dct].

Table 3.4: Classification accuracy of QF-NN and QF-QDA using PCA-MRF with and without image normalization. The QF-QDA results demonstrate that 3×3 pixel neighborhoods are sufficient to discriminate the CUReT materials. QF-QDA is also insensitive to image normalization compared to QF-NN.

Neighborhood Size	QF-NN		QF-QDA	
	Raw	Norm.	Raw	Norm.
1x1	51.65	63.04	69.31	65.60
3x3	74.95	93.09	98.76	99.62
5x5	78.92	94.75	99.28	99.72

Results

Classification results using PCA-MRF are given in Table 3.4 and Figure 3.11. Unless otherwise specified all results use 32 values per QF. I first discuss the results using normalized images in Table 3.4; a discussion of the other results is given later in this section. QF-QDA achieves an accuracy of 99.62% and 99.72% using 3×3 and 5×5 neighborhoods, respectively. Previous methods have pointed out that small, compact neighborhoods specify the CUReT materials. However, these 3×3 results allow the stronger statement that the CUReT materials can be completely distinguished by simple 3×3 neighborhoods. As mentioned in Section 3.3.1, Varma & Zisserman report results of 95.87% and 97.22% using their MRF model with 3×3 and 5×5 neighborhoods, respectively [VZ03]. Pietikainen *et al.* report results of 87% using LBPs constrained to a 3×3 neighborhood [PNMT04].

QF-QDA using PCA-MRF outperforms QF-QDA using MR8-1, MR8-2, and Strong-MRF, and it is equivalent to QF-QDA using MR8-3. This finding holds not only for the results in Table 3.4 based on a training set of size 46, but also for all training set sizes, as shown in Figure 3.11 (top).

Similar to the MR8 QF-QDA classifier, the PCA-MRF QF-QDA classifier is also compact. PCA-MRF QF-QDA achieves an accuracy of 99.04% using just 4 values per QF and a 3×3 neighborhood, for a compact 36 dimensional representation. Figure 3.11 (bottom) shows the sensitivity of PCA-MRF to QF size. QF-QDA results using PCA-MRF are very similar to MR8-3M results. The only exception is the failure of PCA-MRF when only one QF value is

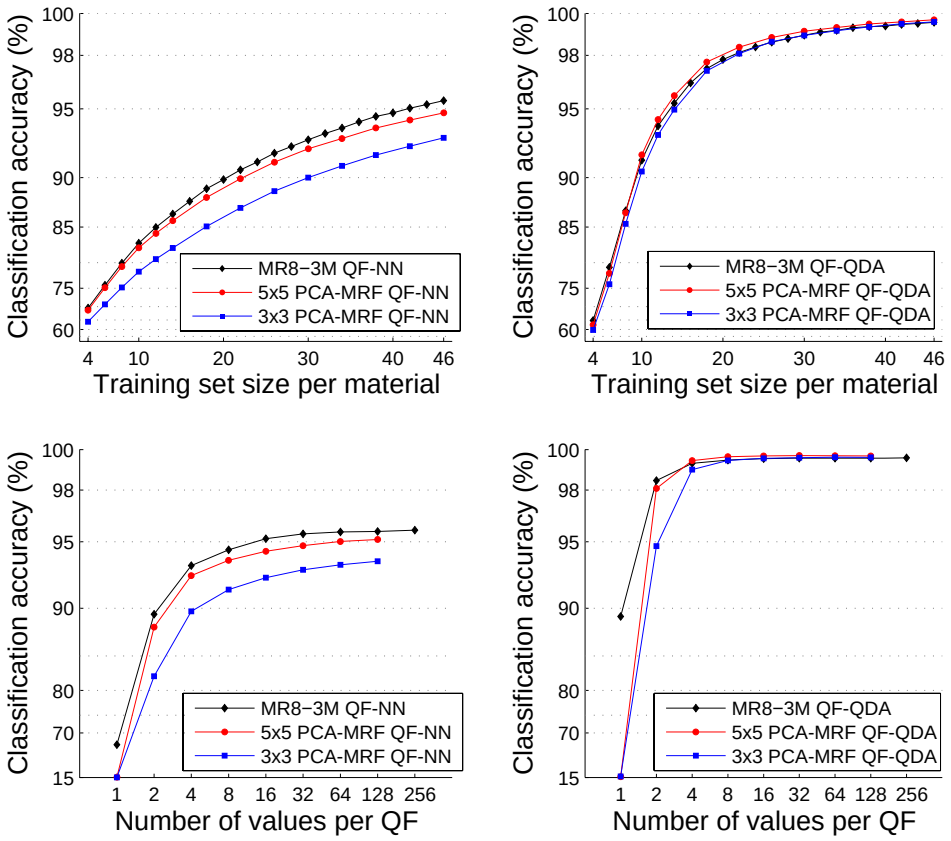


Figure 3.11: Accuracy of PCA-MRF compared to the MR8-3M filter bank for varying training set and QF sizes. The 3×3 and 5×5 PCA-MRF QF-QDA results are very similar to those using the hand-tuned, non-linear, MR8-3M filter bank.

used. One QF bin is equivalent to the mean of each projection, which has been effectively normalized to zero for every image. Therefore, this failure is expected. The MR8 filter bank model avoids this problem by taking the absolute value of several of the filter responses (before taking their max.).

Image Normalization

I also examine the dependence of classification using PCA-MRF to image normalization. All results discussed so far have used preprocessed images with zero mean, unit standard deviation marginal intensity distributions. Table 3.4 gives classification results with and without this normalization. Results show that the normalization is crucial for the QF-NN classifier. In contrast, with no normalization QF-QDA performs nearly as well for 3×3 and 5×5 neighborhoods and actually better for 1×1 neighborhoods. Since mean and standard deviation

are linear changes for QFs, this information can be useful for classification, as it is for 1×1 neighborhoods, or easily down-weighted in the covariance matrix when more discriminating texture information is available.

The effect of this normalization can also be considered geometrically in the space of QFs. As mentioned in Section 2.1.4, all zero mean, unit standard deviation distributions exist on a hypersphere. Since all of the marginals in the various neighborhoods are approximately identically distributed, this normalization makes the concatenated QF vectors live in an approximately spherical space. This could confound the linear estimation performed by QDA. However, the results in Table 3.4 show that this normalization is useful, if possibly not ideal, for QF based representations.

3.3.3 Conclusions on the Strong-MRF and PCA-MRF Texture Models

Section 3.3 presented two MRF based texture models. Both models demonstrated that accurate and efficient classification is possible without preselected, nonlinear filter banks. The models are both more accurate and spatially more compact than in other non-filter bank approaches [VZ03, PNMT04].

Section 3.3.1 presented the Strong-MRF classifier, which uses local features based on pairwise pixel interactions. Section 3.3.2 presented the PCA-MRF classifier, which uses local features equivalent to learned, linear filters. The PCA-MRF model outperformed the Strong-MRF model. The restriction of the Strong-MRF model to pairwise pixel features limits its discrimination power, which forces the model to use larger pixel neighborhoods.

The PCA-MRF model achieved an accuracy equivalent to MR8-3M for various training set sizes and QF sizes. Thus, the hand-tuned MR8-3M features have no benefit over the linear filters learned by PCA-MRF for the evaluated experiment on CUReT.

3.4 Summary and Conclusions

This chapter presented three texture classification algorithms and gave results for a standard experiment on the CUReT database. The three algorithms use different local texture

features, including the rotationally invariant, nonlinear MR8 filter bank, pairwise intensities in a pixel neighborhood, and linear filters learned using PCA on intensities in a pixel neighborhood. For each set of local features, one of the quantile function based multivariate distribution representations developed in Chapter 2 was shown to be appropriate. One focus of the chapter was to compare these classifiers to previous histogram based algorithms. QF based representations were shown to be an accurate and compact alternative to histogram based representations.

The success of the presented classifiers on CUReT is due in large part to two properties of the database, its controlled variation and the small-scale features present in the materials. The variation between images in the same class is due to controlled and well sampled changes in viewing and lighting angles. This type of variation was shown to be approximately linear for QF based distribution representations, an important finding that should generalize beyond the CUReT database. The controlled, linear variation within each class also made possible the use of the QDA classifier, which was shown to be more accurate and efficient than SVM and NN. The covariances learned by QDA were also shown to be effective for reducing the amount of required normalization. Specifically, QDA was shown to not require image normalization.

The materials in CUReT were shown to be completely distinguishable by extremely local, 3×3 pixel neighborhoods. Previous works demonstrated the small-scale nature of the CUReT materials, but they did not demonstrate that such features completely characterize the materials. The success of simple linear filters and pairwise features using 3×3 neighborhoods demonstrate that CUReT supplies poor experiments for the analysis of different local texture features. This finding was only possible, however, due to the success of QDA and QF based representations for a variety of local texture features. These findings even held for smaller training set sizes, where a greater benefit was expected from features with invariances, such as the MR8 responses. The most striking results presented in this chapter were achieved by the QDA classifier using PCA-MRF features. An accuracy of 99% was achieved using 3×3 neighborhoods with a compact 36 dimensional representation for each image.

Additional experiments should be performed on the classifiers presented in this chapter. Two challenges will come from the two key properties of CUReT mentioned above being

removed. First, textures that can only be distinguished by larger scale features could be considered, which would require larger pixel neighborhoods. This would be computationally expensive for the MRF based features presented in this chapter since n^2 local features are found for $n \times n$ neighborhoods. Multi-scale neighborhoods or filter selection techniques could be used to help alleviate these issues. However, large scale features should not affect the appropriateness of QDA or QF based representations.

Second, the more fundamental issue of additional types of variation within each material class could be considered. Variation due to scale, *i.e.*, camera zoom, should be examined. Since the KTH-TIPS2 database measures such variation in a controlled, well sampled, manner, I believe it should be possible to extend the classifiers presented in this chapter to this type of variation. It is more difficult to extend QDA to uncontrolled or undersampled sources of variation. The inclusion in KTH-TIPS2 of multiple but few physical materials for each category is one such challenging addition.

The success of the presented texture models may not be limited to texture classification. These models could be applied to other areas of texture analysis. In particular, the linear properties of the QF representations for differing viewing and illumination angles could be very beneficial for the synthesis of texture onto arbitrary surfaces or for object shape inference. These are briefly discussed in the future work proposed in Section 5.2.2.

Chapter 4

Quantile Function Based Image Segmentation

Segmentation seeks to identify and label image regions. Often a specific object is sought, and segmentation finds where the object is in the image. Segmentation is a complex task that in a Bayesian-like framework integrates shape and appearance information with a search for the most likely location of the object. This chapter focuses on the application of quantile function based distribution representations for describing object appearance.

The amount of available prior information widely varies among segmentation tasks. Tasks with little prior information may not know which objects are possibly in the image. A shape prior may not exist, which leads to each pixel being considered independently. These tasks use a limited appearance prior characterized by a homogeneous boundary feature such as edginess. This chapter focuses on the segmentation of 3D medical images in which there is strong prior information available from manually segmented training images. For these tasks, a known, specific set of objects are segmented. Expected object shapes are given by a strong shape prior that supplies an explicit 3D volume or 2D surface model. Expected object appearances are given by a strong appearance prior that measures nonhomogeneous boundary and regional features in object-relative locations.

The appearance model presented in this chapter is demonstrated on organs in 3D computed tomography (CT) images. Segmentation is performed by deforming a 3D volume shape model. An objective function is optimized over the parameters of the model until the object in the

image is located. In this context, Section 4.1 discusses related work in the entire segmentation pipeline. Particular interest is given to components used by the segmentation algorithm in Section 4.3. This includes the m-rep shape model and its training and segmentation in a Bayesian framework, which are described in Sections 4.1.1 and 4.1.2. Section 4.1.3 discusses the properties and requirements of appearance models and previous work in this area.

Section 4.2 presents an appearance model for use in deformable model segmentation. The model uses the QF based distribution representations presented in Chapter 2. Section 4.2.2 presents a function learned from training examples that expresses expected object appearances. Section 4.3 demonstrates the appearance model and its learned appearance function on several CT data sets. The appearance model is shown to adequately describe the appearance of the left-kidney, bladder, and prostate in CT images. The learned appearance function is shown to adequately describe the variation in a population of such organs. Populations with both between-patient variation and day-to-day variation are examined. Successful segmentation results are reported on a data set of left kidneys from different patients and on multiple data sets each of the bladder and prostate in the same patient on multiple days.

Earlier versions of this work were presented in [BSPC05, BSPC06, BPC$^+$06] and used in [PBJ$^+$06, SBPC07b, PBL$^+$07, SBPC07a, LBR$^+$07, LGL$^+$07, LBJ$^+$07].

4.1 Image Segmentation Background

This chapter presents a novel appearance model for use in segmentation. In order to understand this appearance model, this section first discusses the entire segmentation pipeline. Section 4.3 presents segmentation results for a segmentation methodology based on deformable models and a Bayesian point of view. Deformable models have a rich history in 3D medical imaging. The image segmentation background presented in this section focuses on the previous work in this area.

Deformable models are defined by a set of parameters $\underline{m}$ that determine which image pixels get labeled as belonging to the object. Deformable models segment an image $\underline{I}$ by optimizing an objective function $f(\underline{m}, \underline{I})$ over $\underline{m}$. Typically, f is decomposed into two functions $f_{shape}(\underline{m})$

and $f_{appear}(\underline{m}, \underline{I})$, which capture prior knowledge about the model's shape and appearance, respectively. $f_{appear}(\underline{m}, \underline{I})$ is often expressed as $f_{appear}(\underline{a})$, where $\underline{a}(\underline{m}, \underline{I})$ is a model of the appearance of the object, a representation of $\underline{I}$ relative to $\underline{m}$. $f_{appear}(\underline{a})$ is discussed in depth in Section 4.1.3.

Prior information about the likely shape of an object can be expressed in three ways. The first is by the choice of deformable model. The model must be able to represent the objects of interest and to be able to deform from one to the other. An ideal model meets this requirement using few parameters that linearly describe the variation in the objects of interest. These properties greatly simplify components of the segmentation pipeline discussed below.

Deformable models are either geometric or voxel (3D pixel) based. Geometric models directly represent the object's shape. Typically the boundary is represented, using local parameters, such as the vertices of a mesh [MD97, CTCG95], or global parameters of a specific shape model, such as spherical harmonics [GSS02]. Section 4.1.1 discusses the m-rep shape model, which represents a medial structure of the object and implies its boundary [PFF+03]. Voxel based methods define the object as a function of voxel values. Two examples are deformable atlases [CRM94, Jos97], which supply object labels for each voxel, and level sets [LFGW00, TYW+03], which define an object's boundary as a particular level set of a function whose value is given at each voxel. The deformable atlases approach deforms the entire underlying space using diffeomorphisms. Level sets directly modify the voxel values, an approach which allows arbitrary topologies but does not compute voxel correspondences between deformations. All of these models except level sets supply pixel correspondences across deformations of an object in the volume near the object boundary. This is the sole requirement of the appearance model presented in Section 4.2.

The second method to encode prior shape information is to limit the models that are optimized over. Recall that segmentation requires an optimization over $\underline{m}$ to find the object in the image. Hard shape constraints can be imposed by limiting the optimization to a portion of the search space. Cootes & Taylor were the first to optimize in a bounded, linear shape subspace learned using PCA, instead of optimizing directly on the model primitives [CT01]. PCA estimates the ideal, low parameter shape model mentioned above. The segmentation

framework used in section 4.3 uses a variant of PCA termed PGA that is appropriate for the m-rep shape representation [Fle04]. Optimizing in this reduced subspace also has a large computational advantage. The learned subspace typically has 10's of parameters or fewer while $\underline{m}$ typically has 100's.

The third method to encode prior shape information is to use $f_{shape}(\underline{m})$. f_{shape} places a soft constraint on $\underline{m}$ that penalizes unexpected models. Some f_{shape} functions measure local geometric features, such as curvature. The snakes model is an example that uses only local features [KWT88]. Other f_{shape} functions measure global features, such as a distance measure from a target model to one of the training examples. Both local and global features are either specified in advance or learned from training examples. Local features are easy to specify and can be stably estimated, but they cannot capture rich shape descriptions so tend to be underconstrained. Global features tend to not be specific enough, since they are both hard to adequately specify and hard to learn. Multi-scale approaches, which use global then local features and optimization parameters, address both shortcomings. Section 4.3 uses such an approach. Appearance models have similar scale issues; these are discussed in Sections 4.1.3 and 4.2.

Both shape and appearance prior information benefit from a statistical characterization. A Bayesian segmentation framework nicely incorporates prior information in two phases. First, task-specific prior knowledge is used to select the shape model $\underline{m}$ and appearance model $\underline{a}$. Then the variation of these models is statistically estimated from training examples to specify $f_{shape}(\underline{m})$ and $f_{appear}(\underline{a})$. Bayesian frameworks specify the desired segmentation of image $\underline{I}$ as $\max_{\underline{m}} p(\underline{m}|\underline{I})$; the most probable model is sought for the image. This objective function is called a posterior, and its optimum, the desired segmentation, is the maximum posterior. Using Bayes Rule, $\max_{\underline{m}} p(\underline{m}|\underline{I}) = \max_{\underline{m}}(\log p(\underline{m}) + \log p(\underline{I}|\underline{m}))$ for a fixed $\underline{I}$. $\log p(\underline{m})$ corresponds to f_{shape}, and $\log p(\underline{I}|\underline{m})$ corresponds to f_{appear}. $\log p(\underline{m})$ is referred to as the log (shape) prior. $\log p(\underline{I}|\underline{m})$ is referred to as the log (image) likelihood. Both the prior and the likelihood need to be estimated from training examples. Details of this process are given in Section 4.1.2. Section 4.2 presents an image likelihood for a quantile function based appearance model.

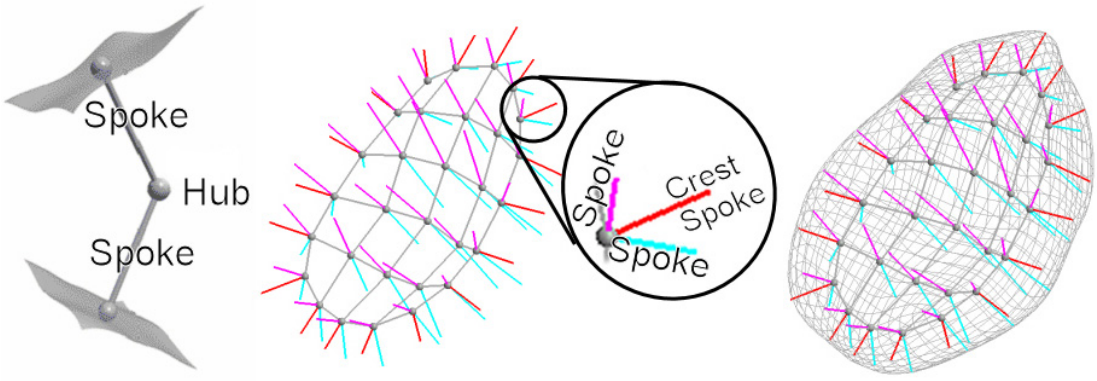


Figure 4.1: The m-rep shape model is a grid of medial atoms (left). The figure shows an m-rep of a bladder (center) with its implied surface (right). This image is taken from [PBJ+06].

4.1.1 M-Reps

The segmentation framework used in Section 4.3 is based on the m-rep shape model [PFF+03]. For simple shapes, such as the ones segmented in Section 4.3, the object representation is a sampled sheet of medial atoms. Each atom in the interior of the sheet consists of a hub and two equal-length spokes. Atoms along the edge of the sheet additionally need to control the boundary crest, so they have one additional parameter that controls the length of a "crest" spoke, which bisects the other two spokes. See Figure 4.1. The representation implies a boundary that passes orthogonally through the spoke ends. Medial atoms are sampled in a discrete grid and properties, such as spoke length and orientation, are interpolated between grid vertices. The model defines a coordinate system which dictates surface normals and a correspondence between deformations of the same m-rep model and the 3D volume in the object boundary region.

4.1.2 Training and Segmentation for Bayesian Methods

Bayesian, deformable model segmentation frameworks must learn the two components of their objective function, the shape prior $p(\underline{m})$ and the image likelihood $p(\underline{I}|\underline{m})$. This section focuses on the shape prior; an in-depth discussion of the image likelihood is given in the next section. Training the shape prior requires three steps: fitting, alignment, and statistical learning. Segmentation requires two main steps: initialization and optimization. These steps

of the Bayesian segmentation pipeline are now discussed.

The shape prior is estimated from training images that have been manually segmented by a human expert. This task is typically challenging, and different experts produce different manual segmentations. This effect is called rater bias. The challenges with accounting for this variability are not discussed in this chapter. Instead, a single expert is used, and automatic segmentations are sought that mimic this specific expert.

Manual segmentation typically supplies a segmentation in a format equivalent to voxel labels. To train the shape prior, parameters of the shape model must be found that match the voxel labels. This is itself a segmentation task. I term "fitting" to be the segmentation of a label image by a shape model. Fitting has the same requirements as segmentation, namely an objective function and an optimization. However, fitting is simpler than segmentation since the appearance of the object is well defined. For fitting, $f_{appear}(\underline{m}, \underline{I})$ compares $\underline{m}$'s implied voxel labeling to the voxel labeling of $\underline{I}$. Comparison measures are either boundary based or region based. Region based comparisons typically use a volume overlap measure between the two voxel labelings. A popular boundary based comparison measure computes the sum of squared distances from many points on the object boundary given by $\underline{m}$ to the closest point on the object boundary implied by $\underline{I}$, and vice-versa. However, computing distances from the boundary given by $\underline{m}$ is computationally expensive for an $\underline{m}$ that is varying, so this is often either not computed or approximated. Such an approximation is used for the f_{appear} function used for fitting in Section 4.3 [MTS$^+$08]. The f_{shape} functions used for fitting are identical to those used for non-Bayesian segmentation. f_{shape} is composed of soft geometric constraints that are designed to obtain non-self-interpenetrating shapes and good model-to-model correspondences.

The above first step of training produces a set of models $\{\underline{m}_i\}$ fit to each training image. As Section 4.2 discusses in more detail, $\{\underline{m}_i\}$ and their corresponding training images are all that are required to train the image likelihood. Computing the shape prior, however, often additionally requires alignment. Here, I consider alignment to include any variation within $\{\underline{m}_i\}$ that one does not wish to statistically model. For example, there is often a change in the global coordinate system of each image which one does not wish to model. Alignment produces

a modified set of models $\{\underline{m}_i'\}$ that are either expressed in a new coordinate system or have had some of its variation subtracted out. The segmentation tasks examined in Section 4.3 use organ specific alignments that are further discussed in Section 4.3. Alignment is also linked to the initialization and optimization performed during segmentation, which is discussed below.

Given the aligned training models $\{\underline{m}_i'\}$, the shape prior can finally be estimated. This is typically done using PCA. The segmentation framework used in Section 4.3 uses the PGA generalization of PCA to compute a multi-scale f_{shape} function on m-reps [Fle04]. First the Fréchet mean m-rep model $\underline{m}_\mu$ of $\{\underline{m}_i'\}$ is computed with respect to a distance metric. Then an appropriately scaled, linear tangent plane in the m-rep shape space is computed at $\underline{m}_\mu$. The training models are projected onto the tangent plane. PCA is used on the projections to compute several global modes of variation and several local, per-atom residual modes of variation. These modes are used for optimization, and their corresponding Mahalanobis distance functions define f_{shape}.

Segmentation begins by placing $\underline{m}_\mu$ at an initial position in the target image. This starting object is the most likely object as determined solely by the shape prior. Then the maximum of the posterior is found by optimizing over the coefficients of the model's learned modes of variation. The initial position or deformation of $\underline{m}_\mu$ for each target image is termed its initialization. The initialization used for each target image should be identical to the alignment used for each training image. Otherwise, the learned variation that is optimized over will not match the variation needed to segment the target images, *i.e.*, the prior will be inappropriate. Section 4.3 further discusses specific initializations and alignments. The segmentation framework used in Section 4.3 performs a multi-scale, conjugate-gradient optimization. It is multi-scale since the optimization is first constrained to the learned global models of variation. Then the local residues are independently optimized with f_{shape} functions that are independent of each other and the global prior. Conjugate-gradient optimization finds the local maximum of the objective function [PFF$^+$03]. It proceeds by first numerically sampling the derivative of the objective function. Then it computes the gradient direction and the gradient's first conjugate direction. Next, for each direction in series, the optimum of the objective function along each line is found using a Brent linear search. This process is repeated until convergence.

The entire training and segmentation pipeline for Bayesian, deformable model frameworks has now been described except for appearance models. Next Section 4.1.3 discusses this remaining piece of the segmentation pipeline. Then Section 4.2 presents a novel quantile function based appearance model.

4.1.3 Object Appearance

Prior information about an object's appearance is encoded into the model of its appearance $\underline{a}$ and its corresponding f_{appear} function, which determines if a particular appearance is expected. Appearance models are composed of several measurements that summarize object relative image regions at specific locations and scales. In medical images, each region experiences intensity variations due to five factors:

1. Imaging device settings
2. Random noise
3. Texture due to the physical properties of a tissue
4. The amount of each tissue in the region
5. Imaging artifacts

Different data sets have different amounts of each type of variation. For example, CT data sets are considered in this chapter, which have little type 1 variation because each image is absolutely calibrated so that values are in Hounsfield units. Typical errors in this calibration lead to variation that appears similar to scaling, a linear form of variation for QF based representations. Data sets of the same patient day-to-day have less type 4 variation than across-patient data sets. This is discussed in more detail in Sections 4.2 and 4.3. Variation due to imaging artifacts can be difficult for appearance models to account for. In this chapter, images with this type of variation are generally excluded.

The primary aim of an appearance model is to allow an f_{appear} function that can identify and distinguish the object interior and exterior. This requires rich and spatially specific measurements that can capture complex gray level appearances near the object boundary. Also,

an ideal f_{appear} function should penalize only relevant, unexpected variation. For example, a candidate prostate model with rectal gas in its interior should be penalized. However, variation far from the object boundary is irrelevant and should not be modeled. Bayesian appearance models must also linearly represent the expected variation so that it can be modeled via PCA. This is examined for the Bayesian, quantile function based appearance model presented in Section 4.2

Appearance Models

Existing appearance models can be characterized by the scale of the regions they model. Local appearance models have many parameters at the scale of a voxel while global models have few parameters in entire object interior and exterior regions. The simplest local models measure edginess as given by the gradient magnitude at the object boundary. However, the objects considered in Section 4.3 do not have boundaries characterized by uniformly strong edges. The appearance of such objects must be specified using more complex features. One category of more complex local models uses tri-linearly interpolated voxel values acquired along profiles normal to the object boundary [SPCR04]; such models include active shape models [CHTH93, CTCG95]. These models capture a rich description of the image in the object boundary region. Another category of local models uses voxel values from entire object-relative image regions; such models include active appearance models [CET98, CT01] and deformable atlas methods [CRM94, JDJG04, RBR06]. These regions are typically rectangular bounding boxes around the objects of interest. The voxel values used by both categories of models are typically intensity as given by the image. The local models can also use image filters to generate per-voxel features that summarize information at larger spatial scales and that measure image structures such as texture or gradients [SCT03, ZS06].

Global appearance models measure the distribution of voxel values in object interior and exterior regions. Typically the univariate distribution of pixel intensity is modeled using standard parametric or non-parametric distribution representations. Some parametric representations measure simple region statistics, such as mean and variance [CV01, TYW$^+$03]. However, these statistics capture limited information and have not been shown to be successful for the

segmentation tasks considered in Section 4.3. Other parametric representations use more complex families, such as a mixture of Gaussians [PD99], but no families have been shown to be able to model the complex intensity patterns that exist in the boundary regions of the objects considered in Section 4.3. Such complex boundaries are typically modeled non-parametrically using histograms [FRZ+05, CDA07].

Appearance models at the fixed local and global scales described above are not ideal for the medical imaging tasks considered in Section 4.3, though both could be appropriate in a multi-scale framework. Global models do not capture all of the relevant information for segmentation. Inhomogeneity in the boundary region cannot be modeled due to the lack of spatial locality. On the other hand, local models capture too much information. Given exact voxel correspondences, there is still expected variation due to types 2 and 3 above, noise and tissue texture. Also, errors in correspondence produce more type 4 variation, *i.e.*, changes in tissue type. These difficulties with both local and global appearance models indicate the promise of an appearance model at an in-between, regional scale. Recently, Costa et. al. presented a regional model that allows some nonhomogeneity by dividing the object interior into 1-3 large regions [CDA07]. Section 4.2 presents a regional appearance model that estimates the distribution of intensities in multiple regions. Each region is at a scale large enough to stably estimate tissue texture and tissue type but small enough to provide spatial locality.

Appearance Functions

Appearance models support and use different types of f_{appear} functions. Existing f_{appear} functions can be divided into three categories: functions that do not require training, functions that learn a template appearance model, and Bayesian functions that learn an average model and its expected variation. These types of f_{appear} functions will be compared to the ideal function, which only penalizes unexpected variation of object-relative appearance.

The first category of f_{appear} functions do not require training. That is, they are solely a function of the target appearance model. For example, a local appearance model that measures edginess at each boundary point can use an f_{appear} function that simply sums these values. Such f_{appear} functions rely on the voxel feature to measure unexpected variation while

being invariant to expected types of variation. However, such a feature does not exist for the complex, inhomogeneous object boundaries that need to be described for the segmentation tasks examined in Section 4.3. For global models, an example untrained f_{appear} function estimates an interior region distribution from the target image. Then the current interior and exterior distributions are compared to this interior estimate, with the exterior region desired to be dissimilar to the interior estimate [CDA07]. Another global model example defines an f_{appear} function based on the mutual information between the voxel values and the model's implied voxel labeling [TWT+03]. Such global f_{appear} functions also tend to be inadequate for the inhomogeneous regions considered in Section 4.3. For example, in the work of [CDA07], their segmentations of bladders in CT images "leak" into the prostate due to their similar appearance being unexpected.

The second category of f_{appear} functions learn their optimal appearance, *i.e.*, a golden template, from training examples. Local models are able to learn an arbitrary, nonhomogeneous appearance while global models are restricted to rich estimates without locality. Such f_{appear} functions first specify a dissimilarity measure between appearance models. Then f_{appear} is set as the dissimilarity between the target appearance model and a reference model learned from training. Local models have used dissimilarity measures such as normalized correlation [SPCR04], mutual information [RBR06], and Euclidean distance [CRM94, JDJG04]. Local models use a reference model that is either a single training example or a mean appearance model computed from a training set. Global models with parametric distribution estimates use dissimilarity measures that are simple expressions of their parameters [CV01, TYW+03]. For a reference model they can use a single training example or averaged parameters from a training set. The histogram based representation of Freedman *et al.* uses a CDF L_p norm distance [FRZ+05]. However, histogram based models are restricted in their choice of reference model. A mean appearance cannot be computed because the variation of histogram based models is nonlinear, which results in the linear mean not being representative of the training examples. Freedman *et al.* address this issue by computing the minimum distance between the target appearance model and all training examples [FRZ+05]. This category of f_{appear} functions can richly describe optimal object appearance, but they cannot distinguish between expected and

unexpected variation. For this a Bayesian based f_{appear} function is required.

The third category of f_{appear} functions is Bayesian based. Existing Bayesian methods model f_{appear} as the image likelihood $p(\underline{I}|\underline{m})$ and assume that it is Gaussian distributed, which allows the model's average appearance and its expected variation to be linearly modeled using PCA. Two Bayesian local appearance models have been proposed by Cootes *et al.*, associated with active shape models (ASMs) [CHTH93, CTCG95] and active appearance models (AAMs) [CET98, CT01]. AAMs define a global f_{appear} function via PCA on the entire tuple of voxel values. ASMs define independent f_{appear} functions for each profile. For each profile, PCA is computed on the tuple of voxel measurements along the profile, which have been converted to normalized derivative values. AAMs and ASMs highlight a difficulty with estimating Bayesian f_{appear} functions for local appearance models. Local appearance models contain a large number values, so they correspond to high dimensional tuples. Also, as mentioned above, each measurement captures more variation than required. Therefore, globally estimating f_{appear} as is done by AAMs is difficult. This issue is prominent for the segmentation tasks considered in Section 4.3 due to the images being 3D, which dramatically increases the number of voxel measurements, and the limited training examples. ASMs address this issue by independently estimating f_{appear} for each profile. These are much easier to estimate, but the interrelations among the profiles are lost.

Distribution based regional and global appearance models do not suffer from the estimation difficulties of local models. Distributions model less variation than per-voxel measurements, so they have a lower inherent dimensionality. Regional and global models are also much more compact: they define lower dimensional tuples. Therefore, they can be stably estimated while still capturing pixel interrelationships. However, existing histogram based distribution models have nonlinear variation, so they cannot be modeled using PCA. Next, Section 4.2 presents a quantile function based regional appearance model without this difficulty.

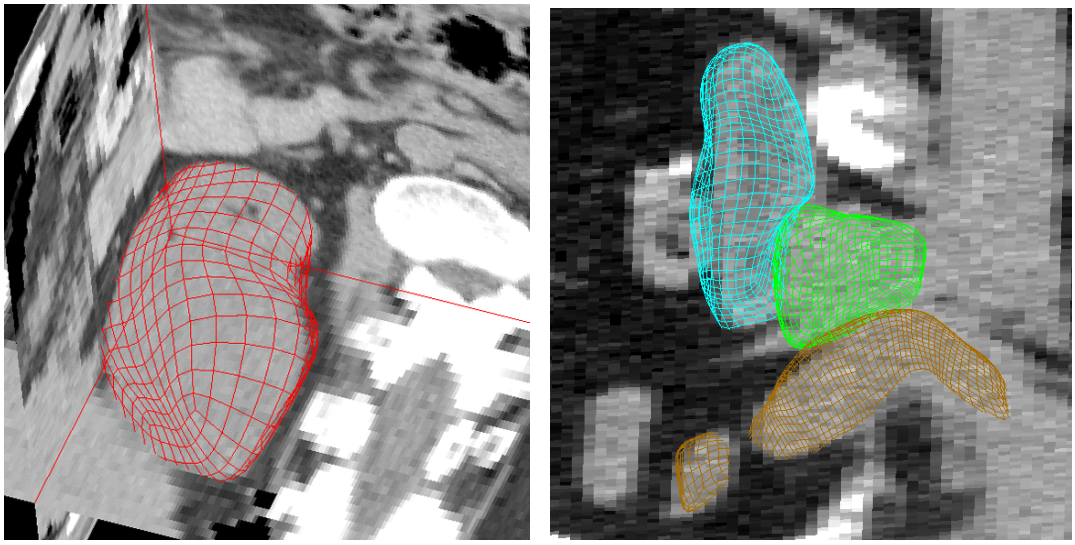


Figure 4.2: The appearance of objects in CT images. Left: the left kidney (red mesh) displayed in a tri-orthogonal display of the 3D image. Right: bladder (blue mesh), prostate (green mesh), and rectum (brown mesh) displayed with a single 2D slice of the CT image.

4.2 The QF Based Regional Appearance Model

This section presents an appearance model for use with objects in 3D CT images, which have boundary regions characterized by complex intensity patterns. Figure 4.2 displays four objects: the left kidney, bladder, prostate, and rectum. Organs, such as the kidney, bladder, and prostate, typically have fairly homogeneous interior boundary regions. Exterior to these organs there is often fatty tissue around a portion of the boundary. Such regions have the characteristic light-to-dark transitions sought by simple, edge-based appearance models. However, other portions of the object boundary may be adjacent to other tissues with similar intensities. The bones might be nearby, which would generate strong dark-to-light edges. Only a narrow strip of fatty tissue may be present, which would generate two strong edges in the region. Therefore, a richer description of the object boundary region is needed beyond edginess.

Modeling these exterior intensity patterns would be extremely difficult if they occurred randomly. Fortunately, they are far from random and instead correspond to objects with spatial relationships given by human anatomy. Therefore, it is possible to learn the likely intensity patterns in each object relative boundary region. As mentioned in Section 4.1.3, measurements in regions the scale of a voxel are highly variable and therefore difficult to learn.

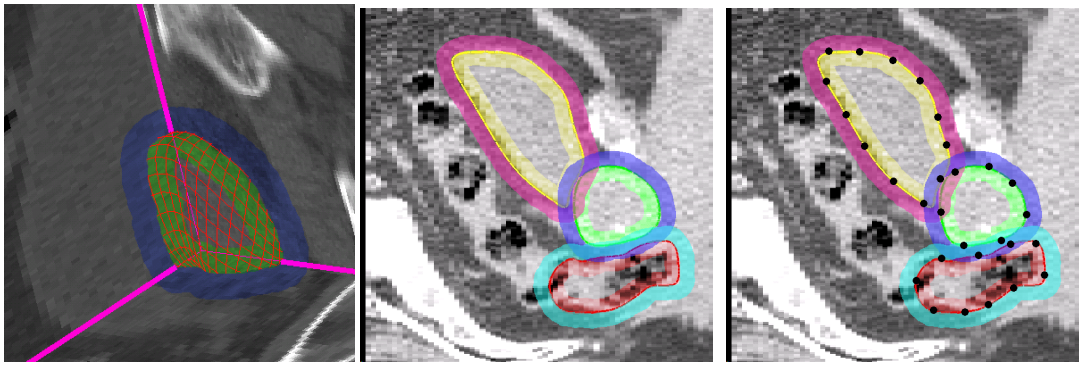


Figure 4.3: An illustration of global image regions (left and center) and the centers of local image regions (right). Left: A prostate displayed as a red mesh with a tri-orthogonal display of the 3D CT image. Center and right: A bladder, prostate, and rectum displayed as contours on a single slice of a 3D CT image.

Measurements in entire interior and exterior regions cannot measure where these different, expected intensity patterns are located. Such information may be required to correctly identify the object boundary. Therefore, this section proposes an appearance model that can be defined at any scale at or between these extremes.

This appearance model describes spatially localized image regions near the object boundary using QF based distribution representations. Section 4.2.1 discusses the details of this appearance model and its computation. A Bayesian image likelihood is estimated for this model from training data. Section 4.2.2 discusses this learned image likelihood function and its training.

4.2.1 The Appearance Model

I define this appearance model at two fixed scales. The definition of these image regions are discussed next. Then the QF based representation of each regional distribution of intensities is discussed. Finally, the computation of this appearance model is discussed when using the m-rep shape model.

Region Definition

I examine two region definitions in detail. The first definition I refer to as my global appearance model; it is depicted in Figure 4.3. For each object being modeled, two regions are

defined, the near object interior and exterior. The contribution of each voxel to its distribution is Gaussian weighted by its distance to the object boundary. Therefore, each region has a hard cutoff at the object boundary and a soft cutoff that gradually falls off away from the boundary. The Gaussian weighting allows narrow regions to be defined that have larger capture ranges and smoother likelihood functions during segmentation than equivalent non-weighted regions. This model has a single free parameter, the common scale $\sigma_{boundary}$ of the Gaussian weighting used for both the interior and exterior region. This parameter is only set in common for the interior and exterior regions to reduce the amount of required parameter tuning. For computational simplicity during segmentation, only voxels within a certain distance from the boundary are found, creating a hard cutoff. However, this distance is typically set to $2\sigma_{boundary}$, so that the affect of the hard cutoff is minimal and so that an additional free parameter is not introduced.

This global appearance model is local to the object boundary, but it does not have any locality along the object boundary. Models with more locality have the flexibility to choose the scale, location, and number of image regions. I now give a region definition for what I refer to as my local appearance model. This model sets the scale, location, and number of image regions based on the choices of these three parameters made by the shape model. This choice may not be ideal since the optimal description of an object's appearance may be at a different scale than the optimal description of its shape. However, much of the inhomogeneity along the object boundary is due to changes in the location and shape of surrounding anatomic objects. Therefore, the scale of the m-rep atom, which has been shown to be a useful scale at which measure the shape of some of these surrounding objects, is used to guide these parameters of the appearance model. In future work, Section 5.2.5 proposes a method to estimate the ideal scale of the appearance model at each stage of the multi-scale segmentation pipeline.

Recall that the m-rep shape model is composed of a grid of medial atoms. Each atom implies a boundary point at the end of each of its 2 or 3 spokes. My local appearance model defines two local image regions centered at each spoke end, interior and exterior regions near the spoke end. Section 4.3 uses m-rep models with atom grids that are approximately 5×6, which define 78 spoke ends with 156 image regions. Therefore, this model is fairly local with a dense set of regional estimates spread along the boundary. Each region has a hard cutoff

at the object boundary, a soft cutoff like the global model based on $\sigma_{boundary}$, and a hard cutoff based on the Euclidean distance between the spoke end and the voxel's corresponding boundary point. This distance, d_{spoke}, is set in common for all regions. This parameter is typically set so that every voxel near the boundary belongs to at least one region, which leads to significant overlap between the regions.

A third region definition at a scale between these presented global and local appearance models was also examined in my early work [BSPC05]. This model was composed of a small number of manually defined, non-overlapping regions that partitioned the object boundary. However, this model was not pursued further due to the success and simpler interface of the local appearance model.

Section 4.3 presents segmentation results using both region definitions. The local appearance model is shown to give more accurate segmentation results than the global model, which demonstrates the benefit of adding locality to the regional estimates.

The QF Based Representation of each Regional Distribution

The probability distribution of voxel intensities is modeled for each image region in the appearance model. Each distribution is represented by the quantile function mixture representation presented in Section 2.2.3. Additional voxel features could be modeled using one of the multivariate distribution representations discussed in Section 2.2.1. This is discussed as future work in Section 5.2.4.

In choosing a distribution representation, the types of distributions that need to be modeled and their variation across images should be considered. The ideal distribution representation would be rich enough to describe the distribution of interest while still being compact. It would be able to be stably estimated given few samples. Also, since a Bayesian image likelihood is desired, its variation would be linear across a set of distributions.

For distributions measured in object boundary regions in CT images, quantile functions have many of these desirable properties. QFs are a non-parametric representation, so they can model the complex distributions measured in these regions. Also, as discussed in Section 2.1, they are more compact than other non-parametric representations. The distributions estimated

from the object relative image regions discussed earlier in this section could theoretically be from extremely local regions consisting of a single voxel or global regions with a hundred thousand samples. The QF is a natural representation across these scales. Recall that the QF representation is basically a sorted list of the available samples. Given extremely local regions containing a single voxel, the QF representation is exactly that voxel value. Therefore, the QF representation reduces to existing local appearance models as region size is reduced to the scale of a voxel.

Section 4.1.3 discusses how each regional distribution experiences variation across images due to CT normalization, noise, tissue texture, and tissue frequency. QFs are approximately linear in the first 3 types of variation, but they are nonlinear in the last type. To partially alleviate this issue, the QF mixture representation is used. The distributions measured in these image regions can be roughly characterized as a mixture of four tissue types, where each tissue is roughly a Gaussian distribution. Additionally, some samples are a linear combination of more than one of the Gaussian distributions due to partial voluming. The four types of intensities correspond to gas (air), fatty tissue, other tissue, and bone.

The ideal QF mixture representation would be composed of four components, one for each intensity type. Such a QF mixture would be defined as $[w_g, \underline{Q_g}, w_f, \underline{Q_f}, w_t, \underline{Q_t}, w_b, \underline{Q_b}]$, where w_i corresponds to the frequency of QF $\underline{Q_i}$. Section 5.2.3 discusses in future work such an ideal representation. Here, however, a simpler representation is proposed. CT intensities are in Hounsfield units, which are normalized such that gas, fat, tissue, and bone have typical values of -1,000, -100 – -50, 10 – 60, and 1,000, respectively. Therefore, the underlying gas and bone distributions are easy to identify. However, the underlying fat and tissue distributions have significant overlap due to partial voluming, noise, and texture.

I propose using the simpler QF mixture representation $[w_g, \underline{Q_{ft}}, w_b]$. Since the underlying gas and bone distributions are well separated from the other distributions, I use thresholding to separate them. Typically threshold values of -224 and 176 are used for gas and bone, respectively. Additionally, I chose not to model $\underline{Q_g}$ or $\underline{Q_b}$. Both QFs characterize only limited information related to partial voluming. Also, the underlying fat and tissue distributions are not estimated. Instead their pooled distribution $\underline{Q_{ft}}$ is modeled for computational simplicity.

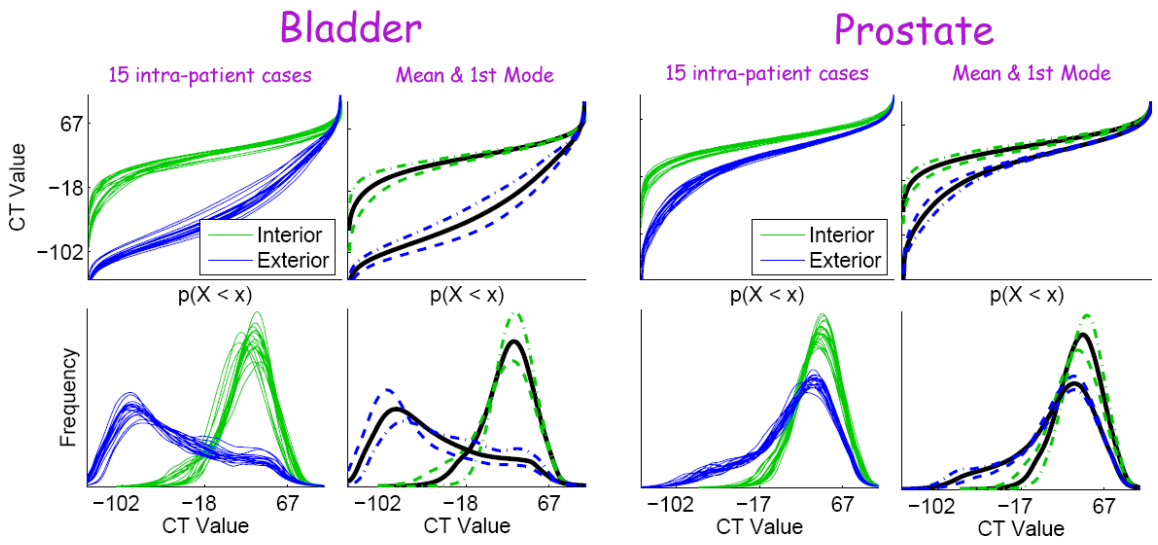


Figure 4.4: QFs estimated from global image regions of the bladder and prostate from a single patient over 15 days. For each region, the mean QF and $\pm 2\sigma$ along the first principal is given, where the modes for region are computed independently. Histogram estimates of these QFs are also displayed.

Their pooled frequency w_{ft} is not modeled, since changes in w_{ft} are already modeled by w_g and w_b. Figure 4.4 shows this pooled fat and tissue distribution for global bladder and prostate regions from a single patient day-to-day. The figure also shows the result of applying PCA to each region's $\underline{Q_{ft}}$, which is discussed more in Section 4.2.2.

The expected variation of the regional distributions represented as QF mixtures $[w_g, \underline{Q_{ft}}, w_b]$ is approximately linear except for mixture changes in the frequency of fat and tissue. Fortunately, the amount of fat and tissue mixture variation is limited, particularly for day-to-day variation within the same patient. Interior regions for objects such as the left kidney, bladder, and prostate are expected to have little tissue mixture variation. Exterior object-relative image regions are expected to have fairly constant tissue frequencies at appropriately large scales. This is true within a patient day-to-day because exterior fat and tissue are physically associated with the organs. Thus, their locations are stable relative to the organ. This fact holds less strongly across patients, where consistency in fat and tissue locations is based only on the consistency of human anatomy.

Additionally, the degree to which fat and tissue mixture variation is nonlinear can be examined. As mentioned in Section 2.1.3, the degree of nonlinearity of QFs undergoing mixture

changes is a function of the distance between the underlying distributions. Based on their mean CT values, the underlying gas and bone distributions are very dissimilar to each other and the fat and tissue distributions. Therefore, changes in their frequencies are extremely nonlinear in the space of QFs; this is why the QF mixture representation is being used. However, the underlying fat and tissue distributions are much more similar to each other. Hence, their mixture variation is much more linear. In Section 4.2.2, an image likelihood using this QF mixture representation is estimated assuming its variation is linear.

Computation of the Appearance Model

The appearance model is a tuple of QF mixtures $[w_g, \underline{Q_{ft}}, w_b]$ for each image region concatenated together. Given an m-rep shape model and an image, I now discuss how to compute this tuple.

First, every voxel near the object boundary must be assigned object coordinates. These are used to assign a voxel to one or more regions and to compute their contribution (weight) to each distribution. However, I do not start at every possible voxel of interest and compute its object coordinates. Instead, an inverse algorithm based on following boundary normals is used that starts with many points in the boundary region with known object coordinates. Then the voxels that the points belong to are computed. This approach is computationally less expensive though it is not guaranteed to find all of the voxels near the boundary. However, for the largely convex objects that are modeled, the voxels most likely lost will be exterior and far from the boundary.

For the m-rep shape model, many points with known object coordinates are generated as follows. Recall that m-reps define a boundary point with a normal at every spoke end. A detailed boundary is defined using a surface subdivision algorithm that generates both point positions and normals. Every level of subdivision increases the number of points by a factor of 4. Typically 4 levels of subdivision are needed for the objects and images examined in Section 4.3. A dense set of points in the boundary region are generated by sampling each normal in this detailed boundary representation. For each paired interior and exterior image region, each voxel is assigned object coordinates based on the first point to find the voxel. However, care

is taken to guarantee the following properties: 1) the voxel is correctly identified as interior or exterior based on the location of its center, 2) the first point to find a voxel has the highest weight of all points that will find the voxel, by sampling from the boundary out and from the region center out, and 3) every voxel is used no more than once per paired interior and exterior region.

The computation above generates a list of weighted samples for every image region. When each distribution is represented using a single QF, this QF is computed by sorting the samples and averaging adjacent values to compute the specified number of equally weighted bins. When each distribution is represented using the QF mixture representation discussed in the previous section, a computationally more efficient approach is used. The use of the gas and bone thresholds leaves only 400 possible unique CT values. Therefore, the list of weighted samples is converted to a 400 bin histogram with additional gas and bone counts. Then a QF is computed from the 400 bin histogram without a loss in accuracy.

4.2.2 The Image Likelihood

A Bayesian appearance function is now defined for the appearance model presented in the previous section. Such appearance functions learn from training examples the probable appearance models of objects segmented using the shape model. This is often characterized by a Gaussian model, which learns both the expected appearance model and its expected variation across correctly segmented images.

The Bayesian framework defines the appearance function $f_{appear}(\underline{m}, \underline{I})$ as the log likelihood $p(\underline{I}|\underline{m})$. The previous section defined an appearance model $\underline{a}$ that is assumed to capture all relevant information in image $\underline{I}$. This allows $p(\underline{I}|\underline{m})$ to be simplified to $p(\underline{a}|\underline{m})$. Recall that $\underline{a}$ is computed relative to $\underline{m}$. Here, I additionally assume that $\underline{a}$ is conditionally independent of $\underline{m}$ beyond this, which simplifies $p(\underline{a}|\underline{m})$ to $p(\underline{a})$. This is a common assumption made by appearance functions [CRM94, CTCG95, CV01, SPCR04, JDJG04, FRZ$^+$05, CDA07], with the exception of active appearance models [CET98, CT01]. This assumption is sensible for medical imaging when modeling variation across patients, which is dominated by anatomy differences that are not known to correspond to specific appearance changes. However, variation

of the same patient day-to-day often does correspond to specific changes in appearance. For example, in the pelvic region 1) increased bladder size is due to an increase in the amount of urine, which affects the intensities in its interior since urine has a slightly different appearance than the bladder wall, 2) the rectum is often distended due to gas, in which case more gas intensities are expected in its interior, and 3) the above changes move or squish the prostate possibly towards the pelvic bones, which affects its exterior appearance. In this work, these effects are ignored to reduce the number of training examples needed to adequately train the image likelihood.

Bayesian methods typically assume both the shape prior and the image likelihood are Gaussian distributed, and they are estimated using PCA. The image likelihood defined in this section is similar. PCA is only appropriate when the variation of the model is linear in the training set and when its parameters are in commensurate units. Great care was taken in Section 4.2.1 to construct an appearance model $\underline{a}$ with such properties. Recall that $\underline{a}$ is a tuple of concatenated quantile mixtures. Section 2.2.3 defined quantile mixtures and a method to scale its elements into commensurate units so that PCA could be used. This scaling allows PCA to be used on the entire $\underline{a}$ tuple to jointly estimate the appearance of the image regions in $\underline{a}$. However, this is typically not done. Instead, different levels of independence are assumed, which allows each distribution to be more stably estimated given a limited training set.

Both the local and global appearance models defined in the previous section can be described as a concatenation of n pairs of interior and exterior image regions with quantile mixtures $[w_g^i, \underline{Q_{ft}^i}, w_b^i]$, where $i = 1, \ldots, 2n$ and the n interior regions are indexed before the n exterior regions. I typically assume that the quantile mixtures are independent. This simplifies the image likelihood $p(\underline{a})$ to $\prod_{i=1}^{2n} p(w_g^i, \underline{Q_{ft}^i}, w_b^i)$. Additionally, for each quantile mixture I also typically assume that the frequency of gas and bone is independent of the distribution of fat and tissue. This simplifies the likelihood to $\prod_{i=1}^{2n} p(w_g^i)p(\underline{Q_{ft}^i})p(w_b^i)$. Most of the results presented in Section 4.3 use image likelihoods that make all of these assumptions. However, in Section 4.3 it is demonstrated that for the examined data sets, the segmentation results are insensitive to this choice.

Appropriate density estimates of these distributions must address one major concern. Dur-

ing segmentation the likelihood of incorrectly segmented objects must be computed. Recall that $p(\underline{a})$ only captures the expected variation of correct segmentations. Therefore, an incorrect notion of variability is applied to the sequence of segmentations that ideally are successively less incorrect as the optimization proceeds. Such a likelihood term is overly sensitive in the shape space. An objective function that uses such a likelihood tends not to be smooth in the shape space, so it is difficult to optimize. However, the optimum of the objective function is still correct, since the optimum of $p(\underline{a})$ is correct and it is correctly weighted against the shape prior for correct segmentations. A method for resolving this issue is discussed as future work in Section 5.2.5.

Standard PCA-based estimates of the aforementioned inappropriate likelihood function $p(\underline{a})$ will not reliably penalize incorrect segmentations. Therefore, the following likelihood function from Chapter 2 is used to give a more appropriate penalty. $p(\underline{a})$ is estimated by applying PCA to appearance models computed from m-reps fit to training images. The subspace learned by PCA is the subspace of correct appearances. Therefore, the incorrectly segmented objects evaluated during segmentation will have appearances far from the learned subspace. This makes measuring the projection distance of such appearance models onto the learned subspace crucial. This is done using the estimation techniques described in Section 2.3. Moreover, this projection distance can be the primary penalty in the objective function for incorrect segmentations. Therefore, the sensitivity of $p(\underline{a})$ to deformations away from the correct segmentation depends on its estimated expected projection distance. When $p(\underline{a})$ is estimated as $\prod_{i=1}^{2n} p(w_g^i, \underline{Q_{ft}^i}, w_b^i)$, it is important that each distribution estimate have the same expected projection distance. Otherwise, some measurements will be more sensitive than others. This is particularly important for the distributions corresponding to paired interior and exterior regions. If the distributions in each pair do not have similar expected projection distances, an interior or exterior bias could be introduced into the segmentations. Therefore, the number of principal components in each is set so that the estimated projection distances are similar. This is either manually done or, as is described in Chapter 2 and used in Chapter 3, the number of components estimated for one distribution is manually set and the others are automatically set to best equalize their projection distances. This not an issue when the appearance in these

regions are jointly estimated.

Additionally, when estimating $p(\underline{a})$ there is often additional prior knowledge that should be taken into account. Specifically, it is usually known in advance if gas or bone are expected in the object interior or exterior. For example, the bladder and prostate should have neither gas nor bone in its interior, and they may have gas or bone in their exterior. I incorporate this prior knowledge into the estimation of $p(\underline{a})$ by introducing *ad hoc* weights α_g^{int}, α_b^{int}, α_g^{ext} and α_b^{ext}, which specify the interior and exterior importance of gas and bone variation. During segmentation each estimated w_g^i and w_b^i is scaled by its corresponding α. Since this scaling is not done on the training data, an $\alpha < 1$ reduces the importance of the variable while an $\alpha > 1$ increases its importance. When $p(w_g^i)$ and $p(w_b^i)$ are independently estimated, this scaling is equivalent to artificially modifying their estimated variances by $1/\alpha^2$. Typically, α is set to 0.1 or 0 where the gas or bone is expected, and it is set to 1 where it is unexpected. For example, neither gas nor bone should be interior to the bladder and prostate. Therefore, α_g^{int} and α_b^{int} are set to 1. In their near exterior, the location of gas is very variable, which makes its expected locations difficult to learn. Therefore, α_g^{ext} is often set to 0. α_b^{ext} is often set to 0.1 so that some information about expected bone position is preserved.

Training the Image Likelihood

The image likelihood $p(\underline{a})$ is estimated from m-rep models fit to training images. The estimation of each independent distribution in $p(\underline{a})$ proceeds as described in Section 2.3, except for one complication. Each fit m-rep model does not perfectly describe the training objects, *i.e.*, there is tolerance in the fitting. Such error is common to all shape models, which segment the training objects at a particular spatial scale. The optimum of the likelihood function should not include these errors. Otherwise, during segmentation, models that happen to segment the object better than expected will be penalized. Additionally, global appearance models cannot localize where these errors occur. Not correcting the optimum would allow global models to accumulate these appearance errors into a localized portion of the object instead of spreading them out along the boundary as desired.

To correct the optimum of the likelihood function, a modified set of appearance models are

estimated from each training image that takes into account the label image. Let $\{\underline{a}_j^T\}$ be the set of m original training appearance models. I additionally measure $\{\underline{a}_j^{T,L}\}$ that uses image regions with interior/exterior correction computed by its label image. I define the optimum of the likelihood function to be $\mu^L = \frac{1}{m}\sum_{j=1}^{m} \underline{a}_j^{T,L}$, the mean of the corrected training set.

However, it is insufficient to use $\{\underline{a}_j^{T,L} - \mu^L\}$ as the input for PCA to estimate the expected covariance. This variation does not include the expected segmentation errors, which will cause the likelihood function to be overweighted in the objective function. Instead, covariance is estimated by applying PCA to $\{\underline{a}_j^T - \mu^L\}$. This solution pools the variation of the correct segmentations and the variation due to fitting error. A more elaborate solution is proposed in Section 5.2.5, where these sources of variation are separately modeled. That section discusses the appearance variation due to the deformations expected during segmentation. The fitting error considered here is the smallest expected deformation error.

Figure 4.4 on page 114 shows $\{\underline{a}_j^{T,L}\}$ for global bladder and prostate regions taken from the same patient on 15 different days. Also displayed is μ^L and $\pm 2\sigma$ along the first principal direction learned from $\{\underline{a}_j^{T,L} - \mu^L\}$. The high degree to which μ^L and the first component match $\{\underline{a}_j^{T,L}\}$ demonstrates that the variation in this set is indeed approximately linear for QFs.

Typical results in Section 4.3 assume each region is independent. These results learn 2 principal components for every interior region and 3 principal components for every exterior region. These numbers were chosen so that roughly 95% of the total variance is captured and so that their expected projection distances are roughly equal. These numbers are consistent with the expectation for organs that exterior regions are typically more variable than interior regions.

One additional small issue is in the independent estimation of each $p(w_g^i)$ and $p(w_b^i)$. In regions where gas and bone are not expected, such as the interior of the bladder and prostate, it is probable that no gas or bone will be measured in the training set. In this case, a variance cannot be estimated. Therefore, a common minimum variance of 0.0001 is defined for all gas and bone frequencies.

Normalizing the Image Likelihood

During optimization, the importance of the log likelihood and the log shape prior is based on the magnitude of their variation. The variations of the log likelihoods defined in this section tend to be very large compared to the variation of the log shape prior. This mismatch leads to objective functions whose optimization is dominated by the likelihood term, which I have empirically found to degrade segmentation results.

The high degree of variation of the log likelihood during segmentation is caused by two factors. First, when the log likelihood is composed of many independent estimates, its expected variation is large. Second, as mentioned previously, the likelihood function does not model the expected appearance changes due to shape deformations from correct segmentations. Therefore, the actual variation in the log likelihood function during optimization will be greater than expected. I propose the following somewhat *ad hoc* solution to downweight the log likelihood term in the objective function. I modify both the log likelihood and the log shape prior so that their expected variances are 1. A more principled solution would instead estimate the actual variation of the log likelihood during optimization; this is discussed in Section 5.2.5.

Recall that segmentation finds $\max_{\underline{m}}(f_{shape}(\underline{m}) + f_{appear}(\underline{a}))$. For the Bayesian model used in this section, $f_{shape}(\underline{m}) = \log p(\underline{m})$ and $f_{appear}(\underline{a}) = \log p(\underline{a})$. Further, since both the prior and the likelihood are Gaussian distributed, their logarithms are characterized by the Mahalanobis distances defined by each distribution up to an additive constant, which can be ignored since it does not affect the optimum. Let $\mathrm{MD}_{shape}(\underline{m})$ and $\mathrm{MD}_{appear}(\underline{a})$ be the Mahalanobis distances corresponding to $p(\underline{m})$ and $p(\underline{a})$. This allows segmentation to be equivalently defined as finding $\min_{\underline{m}}(\mathrm{MD}_{shape}(\underline{m}) + \mathrm{MD}_{appear}(\underline{a}))$.

Both MD_{shape} and MD_{appear} follow chi-squared distributions. The degrees of freedom in each distribution equals its number of estimated components. Let n_{shape} and n_{appear} be the number of components in each. The expected variances of MD_{shape} and MD_{appear} are $2n_{shape}$ and $2n_{appear}$. Therefore, an objective function that has equally weighted likelihoods and shape priors in the sense of expected variance is $\left(\frac{1}{\sqrt{2n_{shape}}}\mathrm{MD}_{shape}(\underline{m}) + \frac{1}{\sqrt{2n_{appear}}}\mathrm{MD}_{appear}(\underline{a})\right)$.

The effect of this normalization is large when local image regions are used and when the likelihood function is estimated assuming that the image regions, gas frequencies, and bone

frequencies are independent. In this case, n_{appear} can be as high as 1,000 (see page 136) while n_{shape} is typically no more than 10. This results in a normalization that downweights the log likelihood by a factor of 10 as compared to the Bayesian model.

4.3 Segmentation Results

Several segmentation results are presented in this section. For each specific segmentation task, the appropriateness of the appearance model and its corresponding likelihood function is first discussed. Then the actual segmentation results are examined. Section 4.3.1 discusses the segmentation of the left kidney and the learning of its across-patient variation. It also compares the appearance model to a voxel-match-based appearance function. Section 4.3.2 discusses the segmentation of the bladder and prostate and the learning of their day-to-day variation within the same patient. Section 4.3.3 discusses a clinically relevant variant of the bladder and prostate segmentation pipeline that pools day-to-day variations across patients.

4.3.1 Across Patient Left Kidney Segmentation: A Comparison of Appearance Models

This section examines the appearance of the human left kidney and its variation across patients. Segmentation results using QF mixtures with the global appearance model are presented and compared with three other segmentation results. First, the benefit of specially handling gas and bone using the QF mixture representation is examined. Second, these segmentation results are compared to a voxel-scale appearance model. Third, the effectiveness of the optimization performed during segmentation is examined by comparing these results to the approximate global maximum of the objective function. These results are from an early study presented in [BSPC06]. As is mentioned in the conclusions of this section, there have been many recent improvements to the entire segmentation pipeline.

The data set consists of 39 slice-by-slice scanned CT images from different patients. Each image captures a completely imaged kidney without pharmaceutical contrast. Each image was acquired at an in-plane resolution of 512×512 with voxel dimensions of $0.98\ mm \times 0.98\ mm$,

and an inter-slice distance between 3 mm and 5 mm. For training, the left kidney in each image was carefully segmented by a human expert slice-by-slice using an interactive contouring tool. 6 landmarks were also identified for each kidney: 2 at the north and south poles, and 4 on a belt around the midsection of the kidney. These landmarks are used for two purposes. First, the landmarks are used to enforce an anatomic correspondence between m-reps fit to each training image. This is accomplished by forcing, via a penalty in the objective function, the ends of 6 pre-identified spokes to correspond to the 6 landmarks. Second, a similarity transform is computed from the landmarks for both alignment and initialization. This initialization uses manually defined information. Therefore, this segmentation pipeline is semi-automatic.

A leave-one-out segmentation experiment was performed. The parameters of the global appearance model were set as follows. 200 bins were used for each quantile function. A scale factor of 0.1 was used for exterior gas and bone frequencies, *i.e.*, $\alpha_g^{int} = \alpha_b^{int} = 1$ and $\alpha_g^{ext} = \alpha_b^{ext} = 0.1$. Two principal components were learned for the interior QF and three components were learned for the exterior QF. Voxel weights were determined using a Gaussian with a standard deviation of 3 mm, *i.e.*, $\sigma_{boundary} = 3\ mm$.

Evaluation of Appearance Model Variation

Figure 4.5 shows the $\underline{Q_{ft}^i}$ QFs estimated from all 39 images in the data set. Also displayed is the mean QF and ± 1.0 standard deviation along the first two principal directions of variation for each region. Two principal components capture 94.8% of the variation in the interior region and 97.4% of the variation in the exterior region.

For the interior region, the QF mean and principal modes visually appear to characterize the 39 input QFs. Therefore, the variation of these QFs is approximately linear. For the exterior region, the 39 QFs contain mixture variation in the amount of fat versus other tissue in the distributions. This mixture variation generates slightly inappropriate QF means and principal modes. The principal modes appear to adequately model the variation of the QFs that roughly correspond to fat. However, it has difficulty capturing the intensities that correspond to other tissue. Instead, a higher than desired probability is assigned to intensities between the desired fat and tissue intensities.

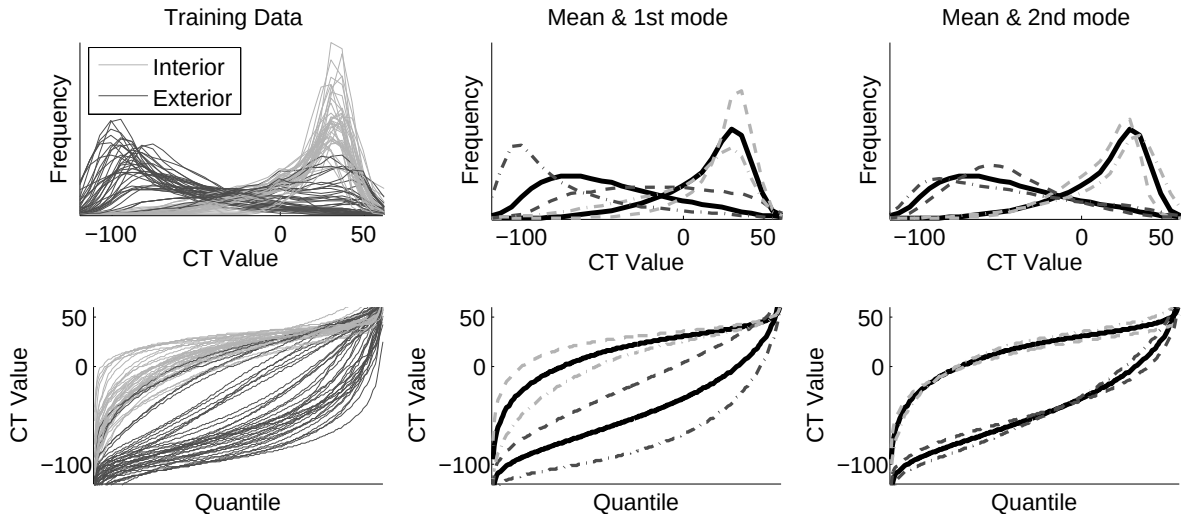


Figure 4.5: Quantile functions estimated from global image regions of the left kidney from 39 patients. For each region the Q_{ft} QFs are displayed. The mean QF and ± 1.0 standard deviation along the first two principal components of variation are displayed. Histogram estimates corresponding to these QFs are also displayed.

Evaluation of Segmentation Results

The global appearance model and its likelihood function are now evaluated by its success on this segmentation task. Success is quantified in two ways. First, an expert can decide if the results are clinically acceptable. Second, the segmentations can be compared to manual segmentations using a performance measure. This section reports both a volume based measure and a boundary based measure. Volume overlap is reported, defined as the volume of the intersection of the two objects divided by the average volume of the two objects. This measure is known as the Dice coefficient. The average closest-point surface distance is also reported. This average surface distance is computed by first computing the minimum distance from many points on both object's boundaries to the other object's boundary. The average of all of these minimum distances is then computed. Improvements in these performance measures often have clinical significance. For example, improvements in average distance often corresponds to large improvements in the most inaccurate portion of the object, instead of many small improvements across the object. This notion of improvement corresponds to clinical improvement. However, other performance measures such as maximum surface distance or 90th percentile surface distance may be more clinically relevant.

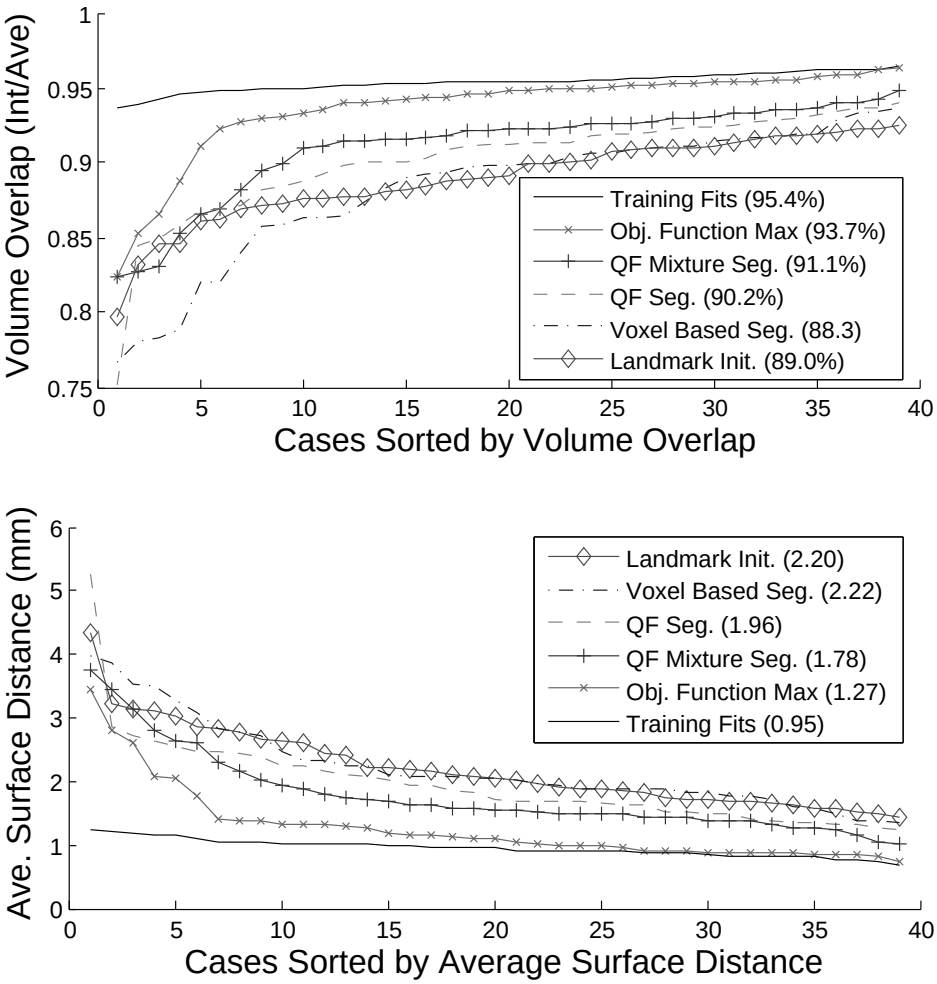


Figure 4.6: Left kidney segmentation results on 39 cases. The legends gives the average performance of each set of segmentations. An objective function based on the QF mixture global appearance model has both an accurate approximate global optimum and an accurate local optimum found via semi-automatic segmentation.

Figure 4.6 reports both of these performance measures for 6 different segmentation results discussed next. First, the accuracy of the models fit during training is reported. The training results represent the best possible results expected during segmentation. They are a good measure of the scale at which the m-rep shape model can represent the left kidney. Second, the quality of the landmark based initialization is reported. Recall that initialization places the mean m-rep model learned from training into the image by applying a similarity transform to the model. The relatively high accuracy of this initialization demonstrates the quality of both the mean m-rep model and the similarity transform. Ideally, segmentation will only improve the results.

Third, Figure 4.6 reports results for the QF mixture based global appearance model presented in Section 4.2.1. Approximately 30 of these 39 segmentations were deemed clinically acceptable by an expert. These results are largely high in quality and they tend to be a significant improvement over the initialization. Average surface distance is improved 0.4 mm from initialization on average across the 39 segmentations.

Fourth, the importance of specially handling gas and bone intensities using QF mixtures is tested. To test this, segmentation is performed using only a QF in each region for all of the intensities. The accuracy of these QF based segmentations is reported. While these results still improve upon the initialization, the QF mixture representation is clearly beneficial.

Fifth, segmentation results are reported for a voxel-scale appearance model that estimates intensities at several positions on many boundary normals [SPCR04]. Its appearance function computes normalized correlation to a carefully constructed template. This model is typically unable to improve the segmentations beyond the initialization, which highlights the difficulty of this segmentation task.

While the QF mixture based segmentation results are largely acceptable, they fail to approach training accuracy. This degradation in performance could be due to many factors related to the shape space, the optimization, and the appearance model. To determine the magnitude that the appearance model is responsible for this issue, the quality of the global optimum of the objective function was estimated. In order to estimate this global optimum, each image was segmented using as initialization the m-rep fit to the label image. These segmentations find the local optimum closest to the ideal training segmentation. The segmentations used as the approximate global optimum are the segmentations with the better objective function values, chosen between these segmentations and the QF mixture based segmentations with the landmark based initialization. In 35 of the 39 cases this optimization found a better estimate of the global optimum of the objective function. Figure 4.6 reports this sixth set of segmentation results. Assuming these results are representative of the true optimum of the objective function, they show the high quality segmentations defined by the QF mixture based global appearance model. 35 of these segmentations are clinically acceptable. Figure 4.7 shows three clinically unacceptable segmentations and three typical segmentations. The first poor

segmentation in Figure 4.7 is due to contrast in the bowel, which is atypical in the data set. The second poor segmentation is due to reconstruction artifacts in the CT image.

Conclusions

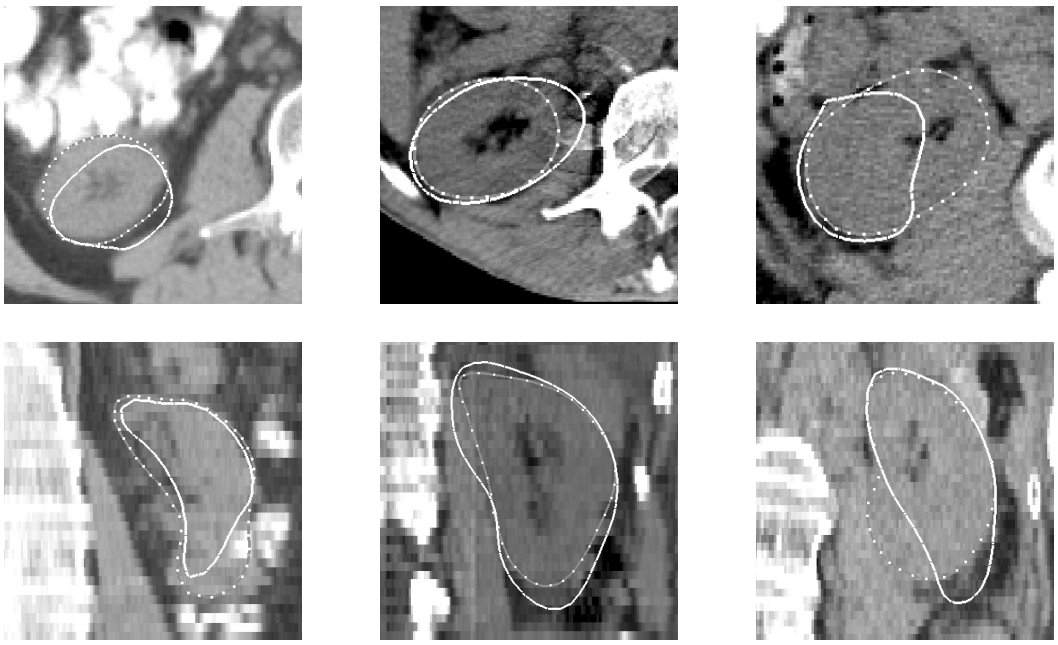
In this experiment, the QF mixture based global appearance model was examined. Its estimated likelihoods showed that interior region distributions were well modeled while exterior region distributions contained some nonlinear artifacts. Despite these artifacts, high quality segmentation results were achieved. In future work, these results should be further compared to existing methods.

This experiment used an early version of the segmentation pipeline. Several improvements and bug fixes have been made to the pipeline since this experiment was performed. Most notably for the appearance model, local image regions have been defined. Also, an atom scale shape prior is now available to refine the results of the object scale segmentation. These features should improve the segmentation results presented in this section and give a clearer view of the comparisons described above.

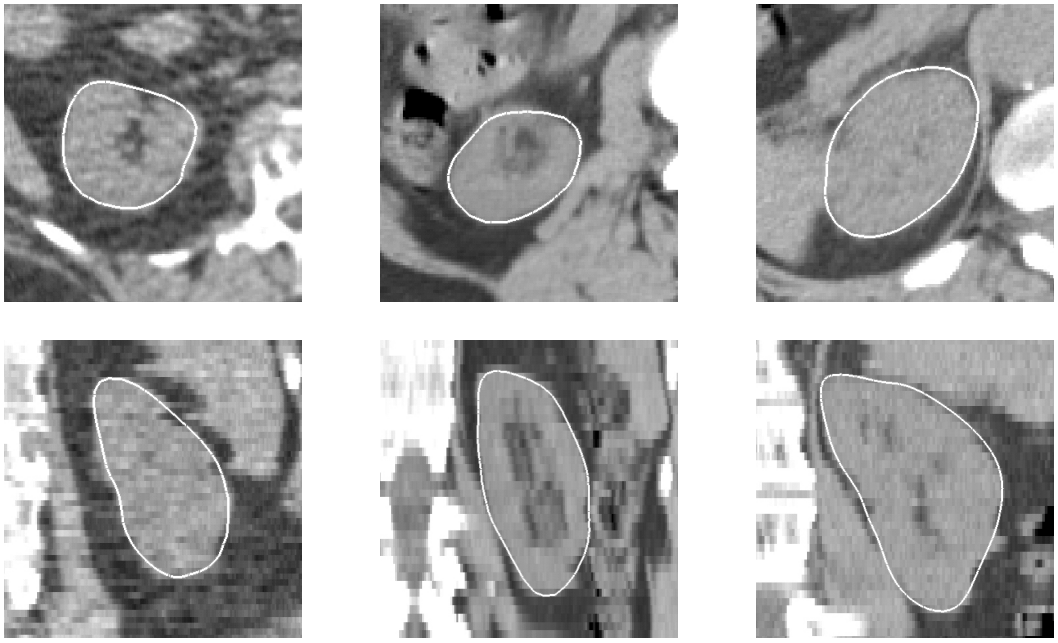
4.3.2 Day-to-Day Bladder and Prostate Segmentation: Evaluating Appearance Model Scale and Statistical Choices

This section examines the appearance of the bladder and prostate in CT images, and the variation in their appearance day-to-day in the same patient. Three experiments are performed that analyze different properties of the appearance model and its learned likelihood function. First, the benefit of appearance functions that estimate a mean and covariance from training examples is examined. Second, global and local image regions are compared. Third, when estimating the likelihood function, different levels of independence are examined.

Each experiment uses a data set of 5 patients each with 13 to 18 daily CT scans of the male pelvic area. The images have an in-plane resolution of 512×512 with voxel dimensions of $0.98\ mm \times 0.98\ mm$ and an inter-slice distance of $3\ mm$. Four of the patients were acquired at University of North Carolina, and one was acquired at William Beaumont Hospital. Expert manual segmentations of the bladder and prostate are supplied for each image. The manual



(a) 3 of the 4 segmentations deemed clinically unacceptable.



(b) 3 typical segmentations from the remaining 35.

Figure 4.7: Segmentation results using the estimated global optimum of the objective function defined using the QF mixture based global appearance model. Each column is a single patient viewed in an axial and coronal slice. The solid contours are the segmentation results and the dotted contours in (a) are the training fits. Note the contrast enhanced bowel in the left column of (a) and the imaging artifacts in the center column of (a).

segmentations are used to produce training m-rep fits for each image. An m-rep shape model with a 5×6 atom grid is used for the bladder; a 4×7 atom grid is used for the prostate. Figure 4.8 gives an example of the day-to-day variation in the manual segmentations and m-rep fits. For alignment and initialization, two additional pieces of information are supplied for each image. First, a similarity transform is supplied that was automatically computed from the bones in each image. Second, a similarity transform is supplied from two prostate landmarks. Each experiment in this section uses an alignment and initialization based on one of these similarity transforms.

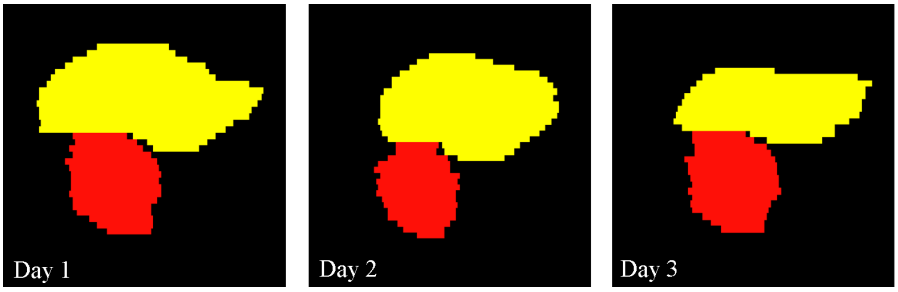
The experiments in this section consider each patient separately. Each patient is segmented using a leave-one-day-out strategy, where training is based on all the images for the patient except the target image. This strategy is clinically unrealistic since the images for each patient are sequentially acquired. A more clinically applicable strategy is discussed in Section 4.3.3.

Figure 4.9 displays global regions of the bladder for all days of 15 patients. The interior and exterior distributions contain little mixture variation within a patient day-to-day. Therefore, day-to-day variation should be accurately modeled via PCA on their QFs. This is supported by the example learned principal components given in Figure 4.4 on page 114.

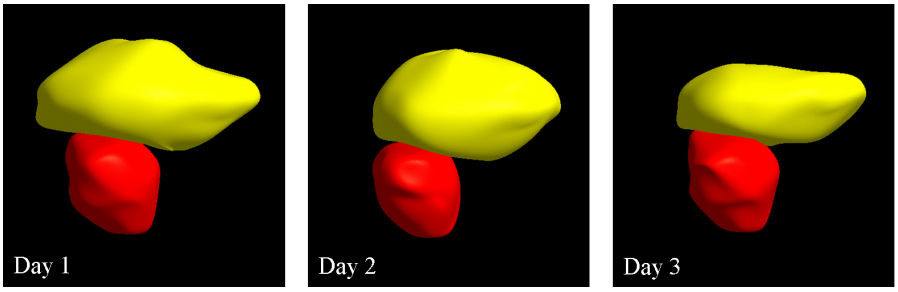
Expected Appearance and Expected Appearance Variation

The appearance function presented in Section 4.2.2 estimates both an object's expected appearance and its expected appearance variation. The expected appearance of the model is its mean in the training set; its expected variation is its covariance in the training set estimated using PCA. With both, the appearance function is a true image likelihood. The benefits of estimating both is examined in this experiment.

This experiment was an initial examination of the appearance model. Its results used a preliminary version of the appearance model and the rest of the segmentation pipeline [BSPC05]. The experimental setup was as follows. A single patient of the 5 described above was used. Alignment and initialization was done via the bone-based similarity transform. Shape training learned 6 modes of variation for optimization. The shape prior was not used, *i.e.*, maximum likelihood segmentation was performed. The appearance model used global



(a) Manual segmentations.

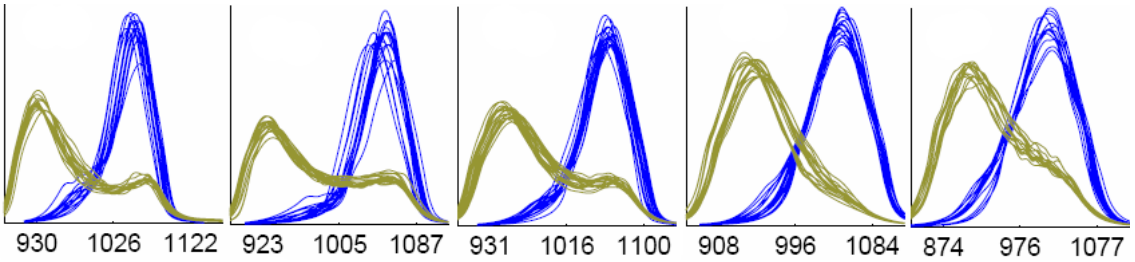


(b) Training M-rep fits.

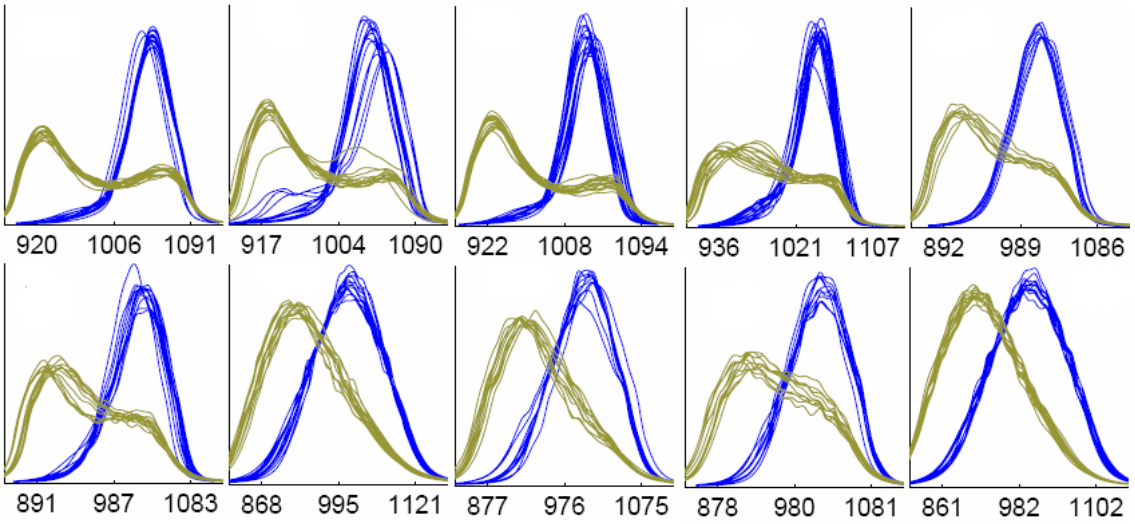
Figure 4.8: The bladder (yellow) and prostate (red) in 3 days of the same patient.

image regions with a sharp 1 *cm* boundary. Each region was represented by a 25 bin QF; the QF mixture representation was not used to specially handle gas and bone intensities.

To measure the impact of the computed mean and covariance, three appearance functions for the global appearance model are examined. The remainder of the segmentation pipeline is kept unmodified. The three appearance functions described next learn increasingly more information about the appearance of the object from training. First, the *EMD-to-day-1* function creates a reference appearance model from the first image. The function is defined as the earth mover's distance to the reference model, which is Euclidean distance for this appearance model. Second, the *EMD-to-mean* appearance function is defined as the EMD to the average appearance model, which is computed from all the other images. Third, the *Mahalanobis-to-mean* appearance function is the image likelihood function presented in Section 4.2.2, which learns both the mean and covariance of appearance models from training. The interior and exterior regions are considered as independent, and two principal components are learned for each.



(a) Global image regions of the bladder for the 5 patients considered in Sections 4.3.2 and 4.3.3.



(b) Global image regions of the bladder for 10 other patients.

Figure 4.9: Interior (blue) and exterior (yellow) global regions of the bladder for all days of 15 patents displayed as histograms.

Segmentation results are given in Table 4.1[1]. Segmentation accuracy improves with increased statistical training, and there is a clear benefit to estimating the mean and covariance of this appearance model. These results highlight the appropriateness of the QF for these linear estimation tasks. The global appearance model is also compared to a voxel based appearance model (see Section 4.3.1 and [SPCR04]). The *EMD-to-day-1* function allows a direct comparison to the voxel method, since both use only the first image and neither are statistically trained. The global appearance model with *EMD-to-day-1* outperforms the voxel based method. The clinical appropriateness of these results is discussed in Section 4.3.2.

This experiment examined the likelihood function proposed in Section 4.2.2. Next, global

[1]These results are reported using the more stringent volume overlap measure defined as intersection volume divided by union volume.

Table 4.1: Bladder and prostate segmentation results using an early version of the QF based global appearance model. The results indicate the usefulness of estimating the mean and covariance of the appearance model. The average result over a single patient's 17 images is given below.

	Volume Overlap (Int./Union %)		Ave. Surface Dist. (mm)	
Appearance Model	Bladder	Prostate	Bladder	Prostate
Training	88.6	87.8	1.11	1.05
Voxel, correlation-to-day-1	79.8	76.0	2.07	2.20
Global, EMD-to-day-1	80.7	78.4	1.97	1.94
Global, EMD-to-mean	81.8	79.4	1.84	1.86
Global, Mahalanobis-to-mean	84.8	79.6	1.53	1.86

image regions are compared to local image regions.

A Comparison of Global and Local Image Regions

Both the previous experiment and the experiment on the left kidney described in Section 4.3.1 used the appearance model with global image regions. This experiment presents a set of segmentation results using local image regions. A similar experiment was reported in [SBPC07b].

Recall that local image regions are defined such that there is an interior and exterior region centered on and localized to every spoke end of the m-rep shape model. The m-rep shape models used for the bladder and prostate in this experiment have 78 and 74 spokes, respectively. As defined in Section 4.2.1, each region has a soft boundary as it falls away from the object boundary and a hard boundary based on the distance to the spoke end. Specifically, $\sigma_{boundary}$ is set to 5 mm and d_{spoke} is set to 1 cm. The QF mixture representation was used with $\alpha_g^{int} = \alpha_b^{int} = 1$ and $\alpha_g^{ext} = \alpha_b^{ext} = 0$. 128 bins were used for each QF.

This experiment used all 5 patient data sets described at the beginning of Section 4.3.2. The bladder and prostate were segmented independently. During shape training, 8 and 4 principal components were learned, respectively. These components were used both for optimization and to define the shape prior. Alignment and initialization were performed using the prostate landmark based similarity transform.

Segmentation results are given in Tables 4.2, 4.3, and 4.4. Using local image regions consistently improved segmentation results for both the bladder and the prostate. This demonstrates the benefit of modeling the appearance inhomogeneity across the boundaries of these objects, which was discussed in Section 4.2. Local image regions capture distinguishing features near the boundary, which is useful for segmentation. The local appearance model depends on a correspondence in the boundary region supplied by the m-rep shape model. The success of the local appearance model suggests that the correspondence provided by m-reps is useful for describing a patient's day-to-day anatomic variation in the pelvic region.

The clinical acceptability of these segmentation results can also be discussed. However, clinical acceptability is application specific, and careful observer studies are necessary to rate the segmentation results discussed here. In the informal studies for radiation oncology that we have conducted so far, clinically acceptable bladder segmentations have had 90% or greater volume overlap, and clinically acceptable prostate segmentations have had 1.5 *mm* or less average surface distance. There are 80 total images for the 5 patients. Initialization, global region segmentation, and local region segmentation produce 9, 62, and 72 such segmentations for the bladder and 54, 67, and 70 such segmentations for the prostate, respectively. The prostate experiences mostly rigid day-to-day variation, which is well captured by the landmark based initialization. However, many of the prostates that were not adequately captured by the initialization were still segmented acceptably.

Joint Versus Independent Image Region Estimation

All of the segmentation results presented so far have used likelihood functions estimated assuming that the image regions, gas frequencies, and bone frequencies are independent. The main benefit of their assumed independence is increased stability in their estimation. However, the resulting model has four undesirable properties. First, these assumptions are not valid because the intensities in the image regions are highly correlated. Therefore, information is discarded that could be useful for segmentation. Second, the minimum expected variance of gas and bone frequencies must be defined. This variance effectively defines an *ad hoc* penalty in the likelihood function. Third, when combined with local image regions, the expected variance

Table 4.2: Bladder segmentation results. Median per patient results are given below. Local regions are more accurate than global regions for all 5 patients.

Patient	Volume Overlap (Int./Ave. %)		Ave. Surface Distance (mm)	
	Global	Local	Global	Local
1	91.5	92.7	1.38	1.07
2	93.4	94.3	1.26	1.09
3	91.2	92.1	1.56	1.32
4	93.7	95.2	1.15	0.94
5	90.1	91.6	1.90	1.53

Table 4.3: Prostate segmentation results. Median per patient results are given below. Local regions are more accurate than global regions for 4 of the 5 patients. The other patient has excellent results for both methods.

Patient	Volume Overlap (Int./Ave. %)		Ave. Surface Distance (mm)	
	Global	Local	Global	Local
1	90.8	91.1	0.93	0.89
2	92.5	94.3	1.30	0.97
3	92.3	92.5	0.96	0.87
4	94.4	94.4	0.90	0.89
5	90.5	92.1	1.70	1.38

Table 4.4: Bladder and prostate segmentation results comparing global and local image regions. Average and standard deviation results for the 5 patients pooled together are given below.

Method	Volume Overlap (Int./Ave. %)		Ave. Surface Distance (mm)	
	Bladder	Prostate	Bladder	Prostate
Training Fits	95.3 ± 0.8	95.4 ± 0.8	0.83 ± 0.08	0.65 ± 0.07
Initialization	79.9 ± 8.9	90.4 ± 4.4	3.62 ± 1.61	1.38 ± 0.75
Global	91.6 ± 3.3	91.7 ± 3.2	1.51 ± 0.54	1.22 ± 0.60
Local	92.7 ± 3.2	92.1 ± 3.5	1.31 ± 0.55	1.14 ± 0.61

of the likelihood function is very large compared to the shape prior. As discussed in Section 4.2.2, this could be detrimental to segmentation performance. Fourth, the estimated interior and exterior likelihood functions may have different levels of sensitivity, which could lead to segmentations biased towards either the object interior or object exterior.

This experiment examines learned likelihood functions that relax these independence assumptions and address the undesirable properties above. Recall that the appearance model is a tuple of concatenated quantile mixtures. Section 2.2.3 defined quantile mixtures and a method to scale its elements into commensurate units so that PCA could be applied. This scaling allows PCA to be used both across image regions and within an image region to jointly estimate gas frequencies, bone frequencies, and the fat and tissue QF.

Two levels of joint estimation are examined. First, the *In/Out Joint* likelihood function jointly estimates each paired interior and exterior region. For local image regions, this jointly models all measurements for each spoke end. For global regions, all measurements made by the appearance model are jointly estimated. Using the notation from Section 4.2.2 (see page 117), the *In/Out Joint* likelihood function estimates $\prod_{i=1}^{n} p(w_g^i, \underline{Q_{ft}^i}, w_b^i, w_g^{n+i}, \underline{Q_{ft}^{n+i}}, w_b^{n+i})$. This likelihood function directly addresses the second and fourth properties above while only lessening the first and third properties. Second, the *All Joint* likelihood function jointly estimates the entire appearance model. For global regions, this likelihood is the same as the *In/Out Joint* likelihood. The *All Joint* likelihood function addresses all four concerns above. However, it may be difficult to adequately estimate.

This experiment used an identical setup to the previous experiment that examined global and local image regions. Table 4.5 presents segmentation results. The results of the three likelihood functions are comparable. That is, segmentation does not appear to be sensitive to the assumed level of independence in the appearance model. However, an experiment with a less accurate initialization might highlight differences in their segmentation results. Beyond accuracy, as discussed above, more effort was needed to define the parameters of the independent likelihood function. Also, the joint likelihood functions estimate a more compact statistical representation of the appearance variability. When independently estimated, the interior and exterior QFs learned 2 and 3 principal components, respectively. Including the

Table 4.5: Bladder and prostate segmentation results under different assumptions of independence of the image regions. The median result is given over all 80 images of the 5 patients. Results suggest that segmentation is not sensitive to this choice.

Method	Volume Overlap (Int./Ave. %)		Ave. Surface Distance (mm)	
	Bladder	Prostate	Bladder	Prostate
Training Fits	95.5	95.6	0.8	0.7
Initialization	81.6	91.4	3.4	1.1
Global Independent	92.3	92.5	1.39	1.05
Global Joint	92.7	92.2	1.32	1.07
Local Independent	93.3	92.9	1.19	0.97
Local In/Out Joint	93.5	93.0	1.15	0.99
Local All Joint	93.2	93.1	1.22	0.98

expected projection distances, gas frequencies, and bone frequencies, this leads to 11 estimated Mahalanobis distances (coefficients) for each paired interior and exterior region. For global image regions, both the bladder and the prostate are represented by 11 coefficients when using the independent likelihood function. Joint estimation learned only a single principal component and its projection distance, which simplifies to 2 coefficients. For local image regions, the number of components depends on the number of spokes in the m-rep shape model. The bladder m-rep has 78 spokes and the prostate m-rep has 74 spokes. Also, the *In/Out Joint* likelihood learned 3 principal components for each region pair, and *All Joint* learned 10 global components. For the independent, *In/Out Joint*, and *All Joint* likelihood functions, the bladder has 858, 312, and 10 coefficients, and the prostate has 814, 296, and 10 coefficients, respectively.

The *All Joint* likelihood function with local regions achieved its best segmentation results when it learned 10 principal components. However, 10 components is the maximum that can be learned across patients. One of the patients has a total of 13 images. This means 12 training images are available for each target image, and computing the mean and the expected projection distance each require one of these degrees of freedom. Therefore, it is difficult to adequately estimate the *All Joint* likelihood function, given the training sets that are available in this experiment.

I conclude that the *In/Out Joint* likelihood function estimates appearance at the most

useful scale for the segmentation of the bladder and prostate in CT images. Compared to the independent likelihood function, it has a lower expected variance and it has more easily defined parameters. Compared to the *All Joint* likelihood function, it is easier to estimate. Also, in each local portion of the object, all of the appearance measurements are jointly modeled. This is a natural scale that will model many of the correlations that are possibly useful for segmentation.

Conclusions

This section focused on the appearance variation of the bladder and prostate within a patient day-to-day. A series of three experiments were discussed that evaluated different aspects of the proposed appearance model and likelihood function. First, the estimated mean and covariance of the appearance model were examined and found to be appropriate. Second, the local appearance model was shown to be at a novel, useful scale for segmentation. Also, the success of the local appearance model suggests that the correspondence provided by the m-rep shape model is useful for describing a patient's day-to-day anatomic variation in the pelvic region. Third, the joint estimation of the appearance model parameters was discussed. The joint estimation of local, paired image regions was shown to be useful for segmentation.

Most of the results presented in this section used an initialization based on two prostate landmarks. In future work, fully-automatic segmentation using a bone based initialization should be examined. I believe that modeling the expected appearance changes due to the shape deformations expected during segmentation will be essential for this task. This is discussed as future work in Section 5.2.5. Also, the shape model uses a multi-scale approach to jointly estimate its parameters. A similar approach could be useful for estimating the appearance model. Such an approach would learn several global principal components and several independent, local residue components. This should allow more correlations in the appearance measurements to be stably estimated.

4.3.3 Bladder and Prostate Segmentation Using Pooled Day-to-Day Variations Across Patients

This section considers the same task as the previous section: the day-to-day segmentation of the bladder and prostate in CT images. The proposed approach estimates a mean shape and appearance model from the previous days of the current patient, and it estimates their expected day-to-day variation from other patients [PBJ+06, BPC+06]. This approach has three main differences with the experiments presented in the previous section. First, day-to-day variation is estimated from other patients instead of the current patient. Second, this approach has more clinical relevance. Each patient's images are acquired in series day-to-day, which limits clinically relevant segmentation approaches to use only previously acquired images for training. This is violated by the leave-one-day-out approach used in Section 4.3.2. Third, this approach cannot be used to segment the first daily image of a patient. Such a task must estimate the variation between patients since no previous images of the patient are available. However, the shape and appearance variation of the bladder and prostate is typically much greater between patients than within a patient day-to-day; this is depicted for bladder appearance in Figure 4.9 on page 131. This more difficult task is left for future work.

Instead, this section presents an approach for segmenting the other $i = 2, \ldots, n$ daily images of a patient. To segment the day i image of a patient, the mean and covariance of both the shape model and appearance model must be estimated. The available training images for this task include the previous daily images of the current patient and all the images from several other patients.

Because between patient variation tends to be much larger than day-to-day variation, it is assumed that the previous days of the current patient will provide the best estimate of the object's mean shape and appearance, *i.e.*, other patient information is ignored. To formalize this, let there be n_{pats} patients each with n^p_{days} daily images. Also let $\underline{a}^p_i$ be the training appearance model corresponding to image $\underline{I}^p_i$, where $p = 1, \ldots, n_{pats}$ and $i = 1, \ldots, n^p_{days}$. Then, the mean appearance model used in segmenting image $\underline{I}^p_i$ is $\mu^p_{i-1} = \frac{1}{i-1} \sum_{j=1}^{i-1} \underline{a}^p_j$. The mean of the shape model is similarly estimated and can be equivalently defined.

The variation of μ^p_i could also be examined. However, it can be shown that modeling the

variation of μ_i^p corresponds to scaling the learned covariance. Since the same scale factor would also be learned when modeling the variation of expected shape, capturing it would only scale the objective function, and thus would have no effect on segmentation.

Learning a patient's day-to-day variation from only previous daily images of that patient is impractical for earlier days. The training sets are too small to adequately train the shape prior and image likelihood. Therefore, information from other patients needs to be incorporated into their training. As is depicted in Figure 4.9 for the bladder, it is unclear if the day-to-day appearance variation of each patient about his distinctive mean is substantially different. For example, the amount of fat versus tissue in the bladder exterior region varies from between patients. Within each patient, however, this mixture appears to vary by a similar amount. Therefore, in this experiment I assume that the day-to-day variation of each patient about their mean is identically distributed. This assumption is made for both the appearance model and the shape model. The usefulness of this assumption will be tested by the quality of the segmentations based on it.

This assumption allows day-to-day appearance covariance to be estimated via pooling across patients. Since the mean of each patient is different, across-patient pooling is done on the residues of the patient's models after its mean model is subtracted. The expected covariance used to segment image $\underline{I}_i^p$ can be estimated from the residues of its own previous days, $\bigcup_{j=1}^{i-1} \underline{a}_j^p - \mu_{i-1}^p$, pooled with the other patient's residues, $\bigcup_{k\neq p,k=1,\ldots,n_{pats},i=1,\ldots,n_{days}^k} \underline{a}_i^k - \mu_{n_{days}^k}^k$. However, the previous days of the current patient are ignored for computational simplicity. Therefore, all days of the current patient are segmented using a common covariance Σ^p estimated from other patients.

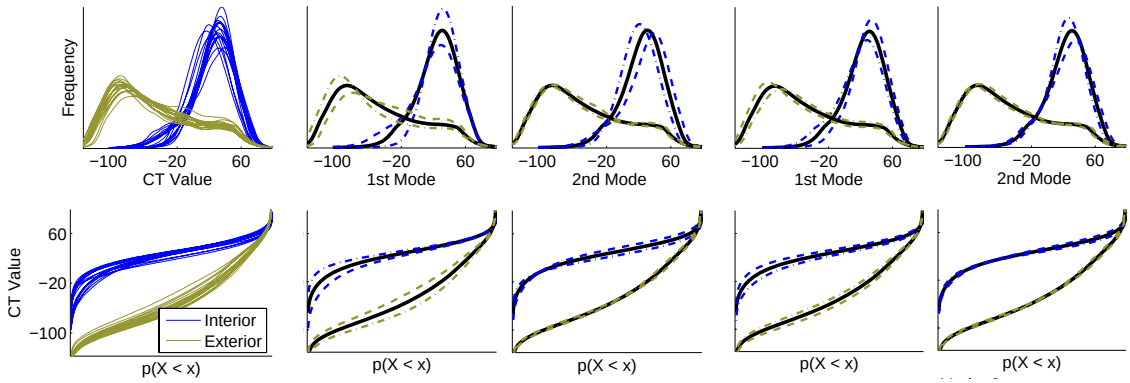
There are three important differences between the learned likelihood function proposed in this section, based on μ_i^p and Σ^p, and the likelihood function trained using the leave-one-day-out approach presented in the previous section. First, μ_i^p will be less accurate since it is estimated from fewer days of patient p. This degrades the quality of the appearance model optimum, the quality of the shape prior optimum, and the quality of the initialization. Second, the assumption that each patient's day-to-day variation is identically distributed has not been carefully examined. Therefore, Σ^p may not accurately describe the day-to-day variation

of patient p. Third, Σ^p is estimated using many more samples than the leave-one-day-out covariance. Assuming that each patient's day-to-day variation is identically distributed, this should increase the accuracy of its estimation.

This experiment used global image regions. Day-to-day appearance variation across patients is not identically distributed for local image regions due to the lack of exterior region correspondence between patients. Local image regions assume that the geometric correspondence defined by the m-reps is appropriate both interior and exterior to the object. Fortunately, within a patient it is reasonable to assume that interior and exterior correspondences are identical since day-to-day variation is physically constrained to be diffeomorphic [JDJG04]. That is, shearing across the object boundary is not physically possible, which is the source of mismatches between interior and exterior correspondences. However, between patients there is no such constraint and the lack of such shearing is an unreasonable assumption. Therefore, global image regions are used in this experiment.

The details of the experimental setup are as follows. The 5 patient data sets are used from the previous section (see page 129). A leave-one-patient-out study is performed, where Σ^p for each patient is trained using the other 4 patients. Global image regions are used with independent QF mixtures, 200 bin QFs, $\sigma_{boundary} = 5\ mm$, $\alpha_g^{int} = \alpha_b^{int} = 1$, and $\alpha_g^{ext} = \alpha_b^{ext} = 0.1$. Alignment and initialization within a patient day-to-day is done using a similarity transform computed from two landmarks for the prostate and 6 landmarks for the bladder. Across-patient alignment of patient mean m-reps is also required to allow the pooling of residues in the estimation of the day-to-day shape variation. However, since this is only required during training, a highly accurate alignment is performed based on geodesic distance [Fle04].

Figure 4.10 displays a patient's global bladder regions and its mean interior QF and mean exterior QF. About these mean QFs, Figures 4.10.b and 4.10.c compare the principal components estimated from the same patient to the components estimated from the other patients. The first two principal components trained from the same patient estimate 96.7% and 97.4% of the patient's interior and exterior variability, respectively. The components trained from the other patients estimate 95.2% and 90.0%. Therefore, the QF space spanned by the two sets of interior components is very similar. The two sets of exterior components have more



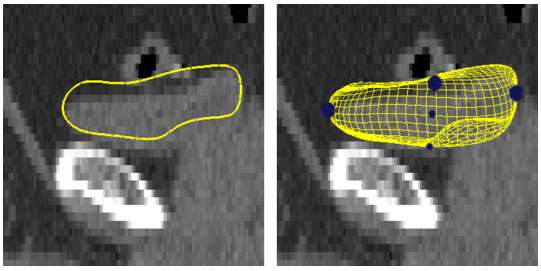
(a) All days of a patient. (b) Day-to-day variation estimated from the current patient. (c) Day-to-day variation estimated from other patients.

Figure 4.10: A comparison of day-to-day variation estimates of global bladder regions. $\pm 2\sigma$ along the first and second principal components is given.

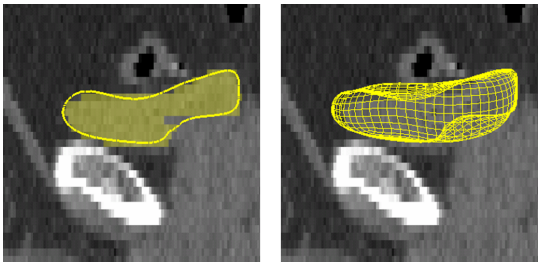
differences. If it is a valid assumption that day-to-day variation across patients is identically distributed, these two sets of percentages would be identical. Their similarity suggests this assumption is roughly correct, especially in the object interior.

Figures 4.11, 4.12, and 4.13 give segmentation results. Figure 4.11 shows an example of the high quality of the landmark based bladder initialization and the further, successful refinement performed during segmentation. Figure 4.12 shows an example of the best and typical segmentation results from both the leave-one-day-out and the leave-one-patient-out approaches. Figure 4.13 compares the leave-one-patient-out segmentation results to the leave-one-day-out segmentation results in terms of volume overlap and average surface distance. The leave-one-day-out segmentation results are consistently but typically slightly more accurate. As mentioned earlier in this section, leave-one-patient-out training is affected by two factors compared to leave-one-day-out training. First, its estimated mean is less accurate since it is estimated from fewer examples. Second, the assumption that day-to-day variation is identically distributed across patients could be invalid. It appears that one or both of these factors significantly effects the estimation of the shape prior or the image likelihood, or both.

While the leave-one-patient-out results are less accurate than the leave-one-day-out results, they use a training approach that can be applied to segment clinical images as they are acquired day-to-day. The main benefit of the leave-one-patient-out approach is that more samples are

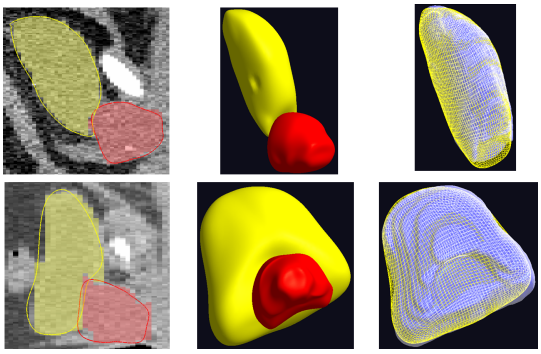


(a) Landmark based bladder initialization.

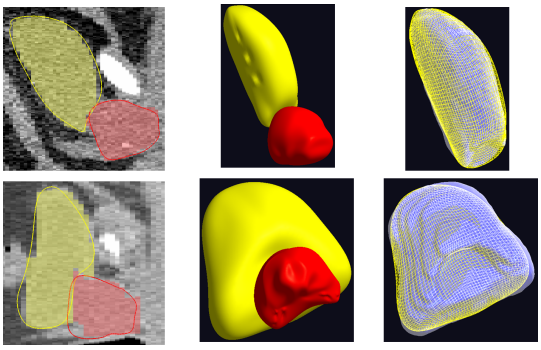


(b) Bladder segmentation using other patient training.

Figure 4.11: Example of bladder initialization and segmentation.



(a) Leave-one-day-out



(b) Leave-one-patient-out

Figure 4.12: Best (top) and typical (bottom) segmentation results.

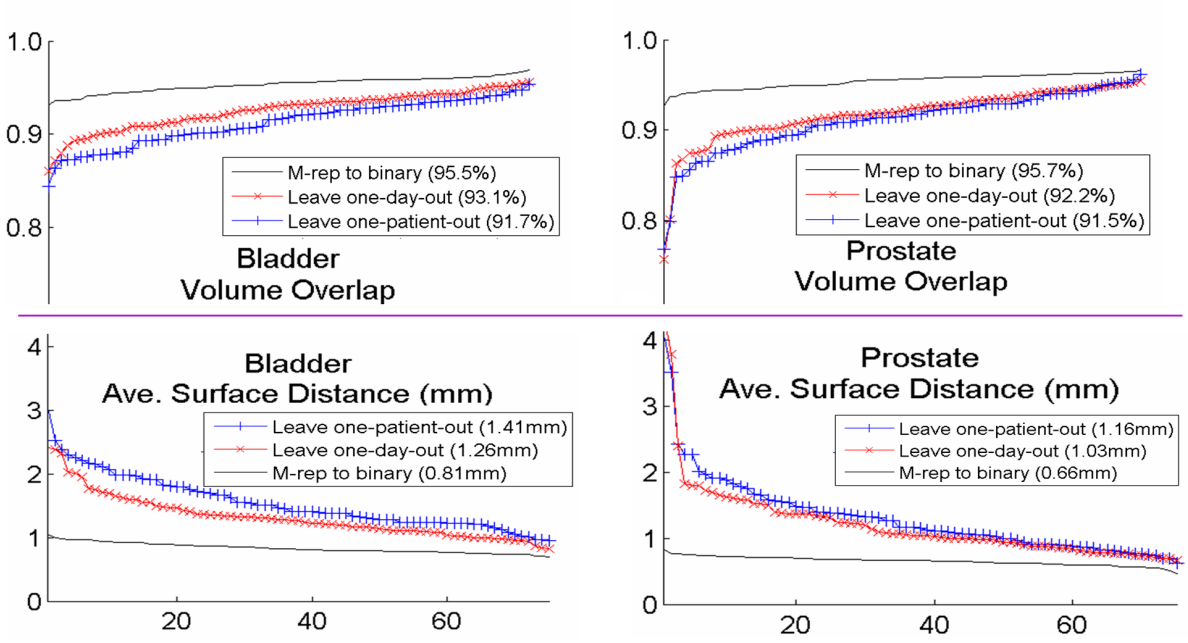


Figure 4.13: Sorted measures comparing segmentation results to manual segmentations in 75 images. The legends give mean performance in parentheses.

available to estimate day-to-day variation. In this experiment the image likelihood was estimated assuming the independence of the image regions, gas frequencies, and bone frequencies. However, the larger training sets supplied by the other patients might be particularly useful for estimating the joint variation of the appearance model. One of the sources of error in the leave-one-patient-out approach is its assumption that day-to-day variation is independently distributed across patients. Methods that relax this assumption could be explored.

4.4 Summary and Conclusions

This chapter presented an appearance model and image likelihood function based on the quantile function methodology discussed in Chapter 2. The appearance model was shown to adequately describe the appearance of the left-kidney, bladder, and prostate in CT images. The learned likelihood function was shown to efficiently describe the variation in a population of such organs.

The presented appearance model estimates object appearance at a novel scale defined

using local image regions. It describes the distribution of intensities in each region using a QF mixture. This representation efficiently represents image regions at any scale, and it simplifies to existing local appearance models at the voxel-scale. The QF mixture representation was shown to have linear variation across the training populations considered in this chapter. This allowed an efficient, Bayesian image likelihood function to be defined. Further, the scale of the appearance model allowed this likelihood function to be stably estimated while still capturing local correlations in the variation of object appearance.

This chapter reported several specific segmentation results. A variety of additional segmentation experiments have been performed. This appearance model has been used to segment the rectum in a setting similar to the reported bladder and prostate experiments. Also, the caudate has been segmented in MR images [LGL+07]. The applicability of the appearance model to these additional objects highlights its generalizability. Additional left kidney, bladder, and prostate segmentation experiments have also been performed using other types of alignment and initialization. The need for these experiments highlights the sensitivity of the segmentation pipeline to initialization. This sensitivity is largely due to the difficult optimization task imposed by the likelihood function. Sections 5.2.4 and 5.2.5 discuss possible appearance models and likelihood functions with less sensitivity to initialization.

Chapter 5

Discussion and Future Work

This chapter reviews and discusses the contributions of this dissertation. This is followed by a discussion of future work. Some of these future projects are developed in some detail.

5.1 Summary of Contributions

This section revisits the thesis and claims laid out in Chapter 1 and presented in Chapters 2, 3, and 4. Each contribution is restated along with a discussion of how it was accomplished in this dissertation.

1. *A geometric interpretation of the space of discrete quantile functions has been developed and described. A key analysis linked the non-parametric representation of the quantile function to several common parametric distribution families.*

   The space of quantile functions was discussed in Section 2.1. Its geometric properties were explored so that the compactness of a population of points in the space could be examined. Compactness was defined in terms of the linearity of the submanifold formed by the population and the resulting low number of parameters needed to express the variability in a population. Compactness in the QF space was studied in general by considering the submanifolds formed by common parametric distribution families. Specifically, location-scale families where shown to form linear submanifolds, and the Weibull distribution was shown to form an exponential submanifold. Other families were analyzed, and their estimated submanifolds in the QF space were examined. The

result of this analysis was an understanding of what types of distribution variation, or equivalently, what types of parameters of distribution families, can be compactly modeled using QFs. Specifically, mean and standard deviation parameters are linearly modeled in the QF space while mixture parameters are strongly nonlinear. This understanding was used to construct appropriate task-specific, QF-based distribution representations.

2. *A novel framework has been developed for representing the variability of multivariate and conditional distributions, and distributions consisting of a mixture of multiple underlying distributions. These quantile function based representations are natural in the sense that their Euclidean distance is an efficient approximation of the Mallows distance. Their variation is parametrically estimated, which results in the learning of task-specific distribution families.*

   Several QF based distribution representations were presented in Section 2.2. These representations were carefully constructed to conserve many of the known linear properties of QFs. Specifically, Section 2.2.1 described a multivariate distribution representation based on QF estimates of multiple, orthogonal distribution projections. Section 2.2.2 described a conditional distribution representation based on a QF partitioning of the conditioned variable. Section 2.2.3 described a representation composed of a mixture of QFs. These more complex distribution representations were needed to represent the appearance of the objects of interest in Chapters 3 and 4, namely, pictures of materials and organs in 3D CT images.

   Section 2.3 discussed the linear estimation of the variation in a set of these representations using principal component analysis. The resulting principal components define a learned parametric distribution family that is ideal for the set.

3. *Texture models using the QF based multivariate and conditional distribution representations have been demonstrated. Both filter bank texture models and Markov random field texture models have been developed and expressed in a common framework, allowing their strong similarities and specific differences to be described.*

   A filter bank texture model was presented in Section 3.2 based on filter response mar-

ginal distributions represented as QFs. Two Markov random field texture models were presented in Section 3.3. First, an MRF texture model was described in Section 3.3.1 that uses the QF based conditional distribution representation presented in Section 2.2.2. This model was shown to be equivalent to second-order Strong-MRF models and gray level co-occurrence matrices. Second, the PCA-MRF texture model was described in Section 3.3.2, which uses the QF based multivariate distribution representation presented in Section 2.2.1. This texture model uses PCA based projection directions in the joint space of pixel intensities in a neighborhood. Each projection direction was shown to be equivalent to a linear filter, which makes the PCA-MRF model equivalent to the proposed filter bank texture model except for the differences in their filters. The PCA-MRF model learns linear filters that accurately estimate the joint distribution of neighboring pixel intensities. The filter bank model uses a bank of nonlinear filters that were preselected based on their discriminative power.

4. *A method for the texture based classification of pictures of materials has been developed and demonstrated. It leverages the demonstrated linearity of the proposed texture models to viewpoint and lighting variation to produce the best reported classification accuracy to date on a standard CUReT database classification task. It is also at least an order of magnitude more compact and computationally efficient than existing methods.*

   Chapter 3 examined the CUReT database, which contains pictures of materials with variation due to controlled and well sampled changes in viewing and lighting angles. This type of variation was shown to be approximately linear for the three proposed QF based texture models. This finding justified the use of the QDA classifier proposed in Section 2.3 for classification tasks on CUReT. The QDA classifier was shown to be more accurate and efficient than SVM and NN when using the proposed texture models.

   The accuracies of the three proposed texture models were compared using the QDA classifier. The PCA-MRF model outperformed the Strong-MRF model, which demonstrated that the restriction of the Strong-MRF model to pairwise pixel features limits its discrimination power. The PCA-MRF model achieved an accuracy equivalent to the MR8-3M

filter bank texture model for various training set sizes and QF sizes. This showed that the hand-tuned MR8-3M features have no benefit over the linear filters learned by the PCA-MRF model for the evaluated experiment on CUReT. All three proposed texture models when combined with the QDA classifier outperformed all equivalent existing texture models that have been applied to the same experiment on the CUReT database.

5. *A multi-scale appearance model for objects in images has been developed. It leverages quantile functions and geometric correspondences supplied by a shape model to generate region descriptions at scales as coarse as the entire inside or outside of the object, as fine as individual boundary points, or in between at the novel scale of a local region.*

   Section 4.2.1 presented an appearance model for objects in 3D CT images that can model the inhomogeneous intensity patterns expected in these object boundary regions. It represents the distribution of intensities in object-relative image regions using the QF mixture representation presented in Section 2.2.3. It was argued that QFs can be used to efficiently represent the distribution of intensities in image regions at any scale and that QFs simplify to existing local appearance models at the voxel-scale. However, since voxel-scale models cannot be stably estimated, two larger scale appearance models were proposed. First, the *global appearance model* was described, which models two image regions: the near boundary object interior and the near boundary exterior. This model can be stably estimated, but it is unable to express inhomogeneity in the boundary region. Second, the *local appearance model* was described, which models many local interior and exterior image regions. This model is at a scale that can be stably estimated and that captures local correlations in the variation of the object appearance. This model requires interior and exterior correspondences near the object boundary. The construction of this model was described when using the m-rep shape model, and the appropriateness of its geometric correspondence for day-to-day segmentation tasks was discussed.

6. *A likelihood term for the Bayesian segmentation of organs in 3D CT images has been proposed and tested. It has been shown that between-patient variation and day-to-day variation of object-relative image regions are efficiently modeled by the quantile function*

mixture representation. State of the art segmentation results have been achieved in left kidney, bladder, and prostate segmentation experiments.

Section 4.2.2 presented an image likelihood function for use in deformable model segmentation. The likelihood function was used with both appearance models presented in Section 4.2.1, which were shown to vary linearly in the training sets considered in Chapter 4. This allowed an efficient, Bayesian image likelihood function to be defined using the techniques discussed in Section 2.3.

In Chapter 4, the usefulness of the likelihood function was evaluated for three segmentation tasks. First, in Section 4.3.1, the likelihood function was combined with the global appearance model to describe the appearance of the left kidney and its variation between patients. Second, in Section 4.3.2, the day-to-day appearance variation of the bladder and prostate was examined. The appropriateness of the local appearance model was demonstrated for this task. Third, in Section 4.3.3, pooling day-to-day variations of the bladder and prostate across patients was examined. All of these segmentation results were evaluated based on both clinical acceptability and performance measures. The results were typically clinically acceptable and they compared favorably to results based on a voxel-scale appearance model.

Finally, the thesis statement presented in Chapter 1 is revisited.

Thesis: *Quantile functions a provide general framework for learning compact representations of probability distributions. This allows accurate and efficient Bayesian methods for texture classification and image segmentation using distributions of image-based appearance features.*

In order to efficiently model a set of probability distributions, their variation must be understood. Quantile functions compactly represent a set of univariate probability distributions in many applications. The first claim in the list above supplies a framework to analyze when QFs will be appropriate. Claim 2 says QFs can be used to represent other types of probability distributions, which could be useful for representing the more complex distributions of interest in computer vision. Specifically, claims 3 and 4 show that QF based representations of both

filter bank and MRF features can be used to efficiently and accurately model textures for classification. Claims 5 and 6 show that QFs in local, object-relative image regions can be used to efficiently and accurately model the appearance of organs in CT images for segmentation.

5.2 Future Work

This dissertation studied the variability of probability distributions. Probability distributions are typically described by a set of constrained functions such as the PDF, CDF, or QF. The statistical analysis of these functions and their relationships could be considered in the more general context of functional data analysis. Work in this field might shed additional light on the trade-offs among these representations, or it might provide additional avenues of research. Inversely, this dissertation might provide insights into techniques used in that field. Specifically, inverses of cumulative functions beyond CDFs could be considered.

5.2.1 Object Recognition

I am most excited about applying this work to object recognition. Here, object recognition is considered to be the supervised classification of a pre-segmented image region. Three factors combine to make object recognition tasks interesting and applicable to this work. First, it integrates shape, color, and texture information. Second, since the goal is discrimination, these features are naturally described by probability distributions. Third, there are available data sets in which the sets of objects within each class undergo known, controlled variation. Therefore, analyzing the variation of probability distributions is both of significant interest and possible using the techniques discussed in this dissertation. One motivating data set for this work is the ETH-80 database of color, segmented images of fruits and toys taken under varying viewing and illumination conditions.

The objects in such images can be described by a variety of texture, color and shape features. Different features need to be examined for the ETH-80 database to see if their distributions undergo linear variation when represented via QFs. Applicable texture features were discussed in detail in Chapter 3. One possible description of object color is the distribution

of its pixels in the RGB or CIE LAB color spaces. This distribution can be represented by the QF based multivariate distribution representation presented in Section 2.2.1. One possible description of object shape is the distribution of distances between its boundary points. This univariate distribution could be represented by a QF. A similar distance based distribution is discussed in Section 5.2.4. The linear variation of such distributions needs to be examined. For example, the above distribution describing shape is linear in scale changes of the object.

Two particular shape features of interest are the local and global versions of the shape context, which estimates $p(d, \theta)$: the distribution of distances and angles between all the object's contour points and a reference point [MBLS01, BMP02]. The local version estimates the shape context at every contour point, which would require modeling the distribution of $p(d, \theta)$ estimates across the object. The global version estimates a single shape context at a reference point such as the object's center. However, modeling $p(d, \theta)$ is not straightforward since θ is cyclic. Therefore, I propose modeling distributions on cyclic random variables.

For modeling a univariate distribution on a cyclic random variable, I first propose defining the Mallows distance for such distributions. Between two distributions estimated by n bin QFs $\underline{x}$ and $\underline{y}$, I define $M_2(\underline{x}, \underline{y})$ as the minimum Mallows distance over all possible $2n - 1$ alignments, or cuts, between $\underline{x}$ and $\underline{y}$. That is,

$$M_2(\underline{x}, \underline{y}) = \min_{j=1,\ldots,n-1} \left(\|x - y\|, \left\| x - \begin{bmatrix} y_{(j+1):n} \\ y_{1:j} + 2\pi \end{bmatrix} \right\|, \left\| x - \begin{bmatrix} y_{(n-j+1):n} - 2\pi \\ y_{1:(n-j)} \end{bmatrix} \right\| \right),$$

where MATLAB notation has been used to specify the reorderings of $\underline{y}$. This distance is the Euclidean distance between properly reordered QFs. Therefore, the mean and covariance of a set of such QFs can be estimated as follows. First the set's Frechet mean is estimated given this distance metric and using an arbitrary cutoff of $[0, 2\pi)$ for the mean. Given this mean, a reordering of each QF can be computed so that the defined M_2 distance is Euclidean distance. PCA can then be performed to estimate the set's covariance.

5.2.2 Texture Synthesis and Object Inference from Texture

In Chapter 3 texture discrimination was performed on CUReT, a database of materials imaged under varying viewing and illumination angles. This database is also an excellent resource for two additional texture analysis tasks: synthesis and object inference. Further, in Chapter 3 it was shown that the variation of the textures in CUReT is approximately linear for QFs. Both of these tasks can greatly benefit from such a representation.

One interesting application of texture synthesis is the synthesis of a texture onto an arbitrary, smooth surface. This task requires an estimate of the appearance of the texture at arbitrary viewing angles, illumination angles, and viewing distances. Ignoring viewing distance, CUReT supplies the appearance of several textures at a sampling of these parameters. In Chapter 3 it was shown that QF representations vary linearly with respect to these parameters. Therefore, accurate estimates of the appearance of the texture at arbitrary values of these parameters can be computed using linear interpolation. Specifically, I propose estimating a texture at an arbitrary viewing angle and illumination angle using a 3-mode tensor decomposition of the training QFs [MV04]. Then, I will use these models to synthesize a texture onto any smooth object.

Similarly, object inference tasks benefit from a representation with linear variation. For example, consider the classification task performed in Chapter 3. This task could be extended to also estimate the viewing and illumination angles of the target image. In order to estimate these angles, a piecewise linear 2D manifold could be estimated during training that represents the variation of each material to these two parameters. Then, a target texture could be projected to the point on the manifold it is closest to, and its estimated angle could be computed by linear interpolation.

5.2.3 More Accurate Mixture Distribution Representations

Chapter 4 examined the distribution of intensities in CT images near organ boundaries. Such distributions can be characterized as being a mixture of 4 underlying unimodal distributions corresponding to gas, fat, other tissue, and bone. I propose two approaches to representing such distributions beyond the QF mixture representation used in Section 4.2.1.

First, I propose an enhancement to the simplified QF mixture representation used in 4.2.1. In Section 4.2.1, the ideal QF mixture representation $[w_g, \underline{Q_g}, w_f, \underline{Q_f}, w_t, \underline{Q_t}, w_b, \underline{Q_b}]$ was simplified to $[w_g, \underline{Q_{ft}}, w_b]$, which does not separate the underlying fat and tissue distributions. Separating these distributions requires more than the simple thresholding used to separate the gas and bone intensities. The following general approach is proposed. To estimate the parameters w_f, $\underline{Q_f}$, w_t, and $\underline{Q_t}$ for a distribution given by QF $\underline{Q}$, perform an optimization that minimizes a prior on the parameters while forcing the defined distribution to exactly match $\underline{Q}$. During training, use a non-Gaussianity penalty for the prior on $\underline{Q_f}$ and $\underline{Q_t}$, such as the projection distance in the space of QFs of the estimated distribution to the Gaussian submanifold. Then, use these training mixture QFs to estimate a PCA based prior for use in estimating the parameters of target distributions.

Second, I propose using a continuous Gaussian mixture model to represent the distribution of intensities in these object-relative image regions. As described below, this model ideally represents a finite Gaussian mixture model that additionally has variation due to partial voluming, an effect in images due to pixels being a linear combination of the underlying distributions. Generalizing a finite Gaussian mixture model is of interest because the four underlying unimodal distributions mentioned above each roughly follows a Gaussian distribution in the absence of partial voluming. First, I will consider the case of two underlying Gaussian distributions $\mathcal{N}_1 = \mathcal{N}(\mu_1, \sigma_1^2)$ and $\mathcal{N}_2 = \mathcal{N}(\mu_2, \sigma_2^2)$. A generalization to four underlying Gaussian distributions is mentioned at the end of the section.

Let X be a standard Gaussian mixture random variable that models mixtures of $\mathcal{N}_1$ and $\mathcal{N}_2$ without partial voluming. Then $f_X(x) = w f_{\mathcal{N}_1}(x) + (1-w) f_{\mathcal{N}_2}(x)$, where w is a scalar mixture parameter. Equivalently, $X \sim \pi\mathcal{N}_1 + (1-\pi)\mathcal{N}_2$, where π is Bernoulli with probability w, *i.e.*, $f_\pi(x) = \{w \text{ if } x = 1, (1-w) \text{ if } x = 0, 0 \text{ otherwise}\}$. In order to generalize this model to allow for partial voluming, recall that this effect corresponds to samples being generated that are a linear combination of $\mathcal{N}_1$ and $\mathcal{N}_2$. A random variable Y that allows partial voluming can be defined as $Y \sim \alpha\mathcal{N}_1 + (1-\alpha)\mathcal{N}_2$, where α is a [0,1] continuous random variable that defines how often samples are generated with α percent of their volume from $\mathcal{N}_1$. In order to express the PDF of this distribution, first I rewrite Y as $Y \sim \mathcal{N}_\alpha = \mathcal{N}(\alpha\mu_1 + (1-\alpha)\mu_2, \alpha^2\sigma_2^2 + (1-\alpha)^2\sigma_2^2)$.

Then, $f_Y(y) = \int_0^1 f_\alpha(y) f_{\mathcal{N}_\alpha}(y) d\alpha$, a continuous Gaussian mixture.

$f_Y(y)$ depends on the form that the random variable α takes. I believe α will be able to be modeled as a beta distribution, *i.e.*, $\alpha \sim \beta(a, b)$, for two reasons. First, the beta distribution converges to the delta distribution as a and b go to 0, which allows the case of no partial voluming to be modeled. Second, more samples are expected with small amounts of partial voluming than large amounts, which the beta distribution models when $a < 1$ and $b < 1$. This defines Y as a parametric distribution family with parameters $\underline{\theta} = [a, b, \mu_1, \sigma_1, \mu_2, \sigma_2]$. Alternatively, I believe a more natural parameterization of this distribution would have the same parameters as a standard Gaussian mixture with the addition of a single parameter that controls the amount of partial voluming, which defines a specific relationship between the number of pixels with partial voluming and the degree of mixture in each pixel. Such a parameterization $\underline{\theta} = [w, v, \mu_1, \sigma_1, \mu_2, \sigma_2]$ could be defined as follows, where $0 \leq w \leq 1$ is the mixture parameter and $0 \leq v \leq 1$ is the amount of partial voluming. I define $f_\alpha(y)$ as the linearly skewed symmetric beta distribution $f_\alpha(y) = \{f_{\beta(v,v)}(y)((2-4w)x+2w)$ if $0 \leq y \leq 0.5$, $f_\alpha(1-y)$ if $0.5 < y \leq 1$, 0 otherwise$\}$.

I believe that the distributions of object appearance modeled in Chapter 4 can be accurately estimated by this parametric family, that its parameters can be stably estimated, and that its parameters will linear vary across the training populations. For this application I propose modeling a mixture of four Gaussian distributions with partial voluming allowed only between adjacent Gaussians. The resulting parametric family will have four mixture parameters and three partial voluming parameters, in addition to the mean and variance parameters of the Gaussians. These parameters should be constrained so that the weights assigned to the corresponding continuous Gaussian mixture are continuous.

5.2.4 Additional Appearance Models

I propose four alternative appearance models to the model presented in Section 4.2.1. A related proposal about the likelihood function is discussed in 5.2.5.

Multiple Appearance Features

One simple improvement to the existing appearance model is to use image features beyond intensity. Segmentation in ultrasound images has been shown to benefit from incorporating texture features. MR images typically have three available features: T1, T2, and proton density. The methods presented in Section 2.2.1 for modeling multivariate probability distributions could be used to estimate the appearance of such features in each model-relative image region.

The Object-Scale Appearance Model

I propose an appearance model that uses object-scale exterior correspondences defined by other objects being (simultaneously) segmented in the image. The global appearance model assumes that no correspondences are known between the objects or along the surface of each object boundary. The local appearance model assumes that both interior and exterior correspondences are implied by the geometric correspondence given by the m-rep shape model. Here, I propose a third appearance model, called the object appearance model. This model does not change interior correspondences, so it could use either local or global interior regions. Exterior to the object, it modifies the global exterior region to be exterior to all of the objects being segmented. Therefore, the current segmentations of the other objects are taken into account. For example, the bladder would use an exterior region that would not include intensities from the interior of the current prostate segmentation. The object appearance model more accurately separates shape and appearance variations by leveraging more of the information supplied by the shape model. For example, if the prostate were to slide along the surface boundary of the bladder, this would cause nonlinear mixture variation in local, external bladder image regions. If the prostate were to move away from the bladder, this would cause variation in the global, external bladder image region. However, this variation should only be modeled by the shape prior. The object appearance model correctly does not model such variation.

For the bladder and prostate segmentation experiments described in Section 4.3.2, this model should be more accurate than the global appearance model. However, since not all

of the objects of interest are modeled and segmented, it may be less accurate than the local appearance model. Therefore, the biggest benefit of the object appearance model may be for the segmentation experiment discussed in Section 4.3.3 for which local image regions are inappropriate. Additionally, in both experiments the pelvic bones and the rectum could also be segmented, which would increase the benefit of this approach over the global appearance model.

Pre-Computing an Approximatated Local Appearance Model

I propose an approximation to the local appearance model that has a likelihood function that could be pre-computed. One drawback of the existing appearance model is its computational complexity. I believe an approximation based on locally linear image regions (see below) could dramatically speedup the optimization performed during segmentation.

The local appearance model presented in Section 4.2.1 uses local image regions centered at every spoke that are defined using the local, curved surface of the object boundary. Instead, I propose defining each local image region using the tangent plane defined by the spoke. Each of these more local image regions is only a function of a spoke's 3 position and 2 orientation parameters. I believe this 5D space could be adequately sampled and used for segmentation as follows. First, for each sample point in this space, compute QFs for its paired interior and exterior regions. Then, for each spoke end, compute and store their Mahalanobis distances to each sample's estimated QFs. A straightforward implementation as described above combined with an m-rep with 75 spoke ends will require too much memory to store. In this case, a sampling at every pixel with 100 orientations would result in a file 7500 times larger than the image. However, I believe this could be made manageable using a bounding box in the image and a multi-scale sampling scheme.

An Appearance Model based on Distance Distributions

I propose an appearance model based on distributions on distance variables instead of distributions on intensity variables. This proposed appearance model computes the spatial relationship of many boundary points to gas, fat, tissue, and bone. It is natural to combine

this model with the shape prior since they both model spatial relationships. The shape prior estimates the probability of explicitly modeled objects while this appearance model estimates the probability of objects implied by image intensities.

Local image regions can be viewed as defining $p(d, i)$, the joint distribution of distance and intensity with respect to each spoke end. The local appearance model computes two weighted marginal distributions $p^{int}(i)$ and $p^{ext}(i)$ from $p(d, i)$ using signed distance to compute the contribution of each sample to $p^{int}(i)$ and $p^{ext}(i)$. Here, I instead propose modeling $p_\alpha(d|i)$, the distribution of unsigned distances of the closest α samples at fixed intensities corresponding to gas, fat, tissue, and bone. Specifically, I propose defining 4 intensities $\{i_g, i_f, i_t, i_b\}$ corresponding to gas, fat, tissue, and bone. The appearance model will consist of 4 QF-represented distance distributions for each spoke end, $p_\alpha(d|i_g)p_\alpha(d|i_f)p_\alpha(d|i_t)p_\alpha(d|i_b)$. To compute each distribution, I propose using a piecewise linear weighting scheme that assumes that all intensity variation from $\{i_g, i_f, i_t, i_b\}$ is caused by partial voluming. Additional image normalization may be required to insure the image intensities correctly correspond to $\{i_g, i_f, i_t, i_b\}$.

This model has 4 desirable properties. First, it can describe object boundaries that are described by any stable spatial relationship of fat, tissue, and bone. Therefore, it is not constrained to object boundaries at transitions between them, like gradient based methods. Second, it is not constrained to predefined local image regions. This model examines the image as far from the boundary as required in order to find the amount of gas, fat, tissue, and bone specified by α, *i.e.*, it does not have a limited capture range. Third, it can be pre-computed. When each spoke end is considered independently, it has a 3 parameter input space that could be sampled at every pixel position. A straightforward implementation with an m-rep model with 75 spoke ends will have storage requirements of 75 times the input image, which can easily be made manageable. Fourth, I believe it will be fairly invariant to day-to-day variation of correct segmentations while varying linearly with increased deformation from correct segmentations. Roughly, local movement of gas, fat, tissue, and bone relative to a spoke end should be linear while local changes in the amount of each one is nonlinear. For the segmentation of the bladder and prostate I believe day-to-day variation of fat, tissue, and bone can be locally characterized as movement. Therefore this variation should be linear in

distance. Gas, however, needs special handling because its position is highly variable day-to-day and its amount changes day-to-day. For the bladder and the prostate, this can be resolved by pooling fat and gas intensities together. Since they are both always exterior to the bladder and prostate, this will decrease their variability while not effecting accuracy.

An early version of this idea has been implemented for distances to bone by Joshua Stough at UNC Chapel Hill. Limited experiments showed no advantages in segmentation accuracy of the prostate over the QF mixture approach described in Section 4.2 based on thresholding bone intensities and estimating their frequency.

5.2.5 Incorporation of Segmentation Variability: The Ideal Image Likelihood Function

In Chapter 4, a likelihood function was designed for use in deformable model based image segmentation. Fenster & Kender [FK01] made two key observations about the requirements of such likelihood functions:

1. For optimization to succeed, the function must be optimal for the correct segmentation. Contrary to intuition, the distribution of a function's values for correct segmentations gives no information about its goodness for segmentation. Incorrect segmentations must have different values, *i.e.*, the function must be specific.

2. If a gradient-descent type optimization is performed, the function must meet the more stringent condition that it become closer to optimal as a shape gets closer to correct.

Many likelihood functions do not meet this second requirement, so they suffer from local minima or capture range issues. Additionally, many likelihood functions are only evaluated by the quality of their segmentation results. However, the cause of a poor segmentation result is difficult to determine. It could be due to one of the issues above or to other portions of the segmentation pipeline such as the shape model, initialization, shape prior, or optimization. Fenster & Kender recognize the inadequacy of relying on segmentation results to evaluate the likelihood function, so they introduce a different evaluation metric based on the likelihood function's behavior as a segmentation gets further from correct [FK01].

In this section I expand upon this idea. First, a more principled evaluation metric is proposed when a shape prior is available. This metric is based on a definition of the ideal likelihood function for a given shape prior. Second, a training strategy is proposed to learn likelihood functions that are closer to ideal. Current learned likelihood functions are trained only on correct segmentations, which was argued in Section 4.2.2 to be inadequate.

Evaluating the Quality of a Likelihood Function

Whereas Section 4.2 argued for equally penalized expected variations of correct segmentations, I believe that the ideal objective function used for segmentation will have a shape prior and image likelihood that equally penalize *expected deviations* from correct segmentations, as well. Also, to meet the two requirements above, both components must define a smooth function such as a multivariate Gaussian distribution in the space of expected deviations. Here I focus on defining a likelihood function given a fixed shape prior.

The quality of a likelihood function can be defined for a given training image and fit shape model as follows. The shape prior is typically defined to be a multivariate Gaussian distribution on the parameters of the shape model. Further, since the prior is used to deform the object during segmentation, this prior centered on the correct segmentation defines the expected deviations. The ideal likelihood function would define the same Gaussian distribution in the shape space as this recentered shape prior. Equivalently, in the parameter space of the shape prior, the ideal likelihood function would be a centered, unit multivariate Gaussian.

To evaluate how close a likelihood function is to this ideal for the given image and fit shape model, the expected deviations from the correct segmentation can be simulated by sampling the recentered shape prior. Then, the likelihood function can be computed for each shape sample. To evaluate the quality of the likelihood function, a dissimilarity measure must be defined that measures how close these sampled likelihood function values are to the centered, unit multivariate Gaussian. For each sampled value the ideal value is known. Therefore, I propose using the sum of squared differences as the penalty measure.

This approach is similar to that taken by Fenster & Kender, who defined the correlation between the sampled values and a 1D variable that described the degree of deformation. The

proposed approach simply leverages the shape prior to do a more principled sampling and to define a more accurate penalty.

An additional idea to explore is the notion of the scale of the expected deformations. As segmentation proceeds, the current segmentation should get closer to the correct segmentation. A multi-scale shape prior somewhat captures this notion; the likelihood function could be estimated at each of these fixed scales. Alternatively, a scale parameter $0 \leq \alpha \leq 1$ could be defined that scales the expected deformations generated by the prior. The likelihood function could be evaluated at different values of α. An α of 0 corresponds to only expecting segmentation deformations that are on the scale of the fitting error of the shape model during training, an idea discussed in Section 4.2.2.

A straightforward use of this evaluation framework is for parameter selection. The parameters of the appearance model and the likelihood function could be tuned to make the likelihood function closest to ideal. Such a principled approach to parameter tuning makes reasonable the introduction of appearance models with many more parameters. For example, a multi-scale appearance model could be defined, where for each scale of the shape prior an appearance model is found that is closest to ideal. Such a multi-scale appearance model could explore parameters such as the degree of Gaussian smoothing of the image and region size.

Learning a Closer to Ideal Likelihood Function

The notion of expected segmentation variability can also be incorporated into the training of the likelihood function. Since the likelihood function is evaluated by its performance on such deformations, it makes sense to train on them. The current likelihood function estimates $p_{corr}(\underline{a_{corr}})$, the likelihood of a correct segmentation. I propose additionally modeling $p_{def,\alpha}(\underline{\Delta a_{def}})$, the changes in the appearance model due to expected segmentation deformations at scale α. This formulation assumes $\underline{\Delta a_{def}}$ is i.i.d. for different $\underline{a_{corr}}$, which allows $p_{def,\alpha}$ to be trained by pooling estimates across the training images. During segmentation, the likelihood of a segmentation with appearance $\underline{a}$ is now $p(\underline{a}) = \min_{\underline{a_{corr}}} p_{corr}(\underline{a_{corr}}) p_{def,\alpha}(\underline{\Delta a_{def}})$ subject to $\underline{a} = \underline{a_{corr}} + \underline{\Delta a_{def}}$. I additionally assume that $p_{def,\alpha}$ is Gaussian distributed in the space of the appearance parameters. This assumption combined with the i.i.d. assumption

above allows a simple closed form solution to this equation. Let $p_{corr} \sim \mathcal{N}(\mu_{corr}, \Sigma_{corr})$ and $p_{def,\alpha} \sim \mathcal{N}(\underline{0}, \Sigma_{def,\alpha})$. If Σ_{corr} and $\Sigma_{def,\alpha}$ are estimated using the same principal directions, p can be simply expressed as $p \sim \mathcal{N}(\mu_{corr}, \Sigma_{corr} + \Sigma_{def,\alpha})$.

One possible issue with this formulation that needs to be examined is the appropriateness of the QF based representations in this context. The variation measured by $p_{def,\alpha}$ should primarily be mixture changes in the amount of each tissue in the object-relative region. Therefore, one of the appearance models proposed in Sections 5.2.3 or 5.2.4 may be more appropriate in this context.

This likelihood function more accurately estimates the spatial accuracy of a given segmentation. This could be useful to automatically signal failures by defining a likelihood function at the scale of acceptable segmentations [LBR$^+$07]. Another possible use of this likelihood function is that it could be used to guide the optimizer, using an approach similar to [CET98]. Jingdan Zhang (a Ph.D. student at UNC) has explored an idea similar to the ones presented here to learn an ideal likelihood function in a kernel framework directly from the images.

Appendix A

Users Guide

This appendix presents a guide to the basic algorithms developed in this dissertation for the computation and display of quantile functions, for converting between QFs and PDFs, and for representing conditional distributions using QFs. The guide concludes with an example that uses some of these functions to generate Figure 2.4(c) on page 19. All code is given in MATLAB.

A.1 QF Computation

This section provides three functions for computing the discrete quantile function representation from samples. These algorithms were mentioned in Section 2.1.2 on page 14. The first function is used to quickly compute a quantile function from samples when many samples are available. This algorithm was used in Chapter 3 by both the MR8 and PCA-MRF texture models. The second function is slower, more accurate, and also allows weighted samples. The third function assumes the samples are from a discrete distribution that takes on only integer values. This is leveraged to avoid sorting by instead computing a fine histogram. This third function was used in the appearance model described in Section 4.2.

```
function qfs = getQFs(features, numBins);
  % Input
  %   features: a numFeatures by numSamples matrix
  %   numBins: the number of QF bins to use per feature
  % Output
  %   qfs: a numFeatures by numBins matrix that is
  %   a discrete QF for each feature
  % Approach
  %   1. Compute an integer number of samples per QF bin
  %      by randomly discarding some of the samples
```

```
  %   2. Sort the samples for each feature
  %   3. Average adjacent samples to compute each bin value

  [numFeatures, numSamples] = size(features);
  ind = randperm(numSamples);
  numSamplesPerBin = floor(numSamples/numBins);
  numSamples = numSamplesPerBin * numBins;

  qfs = reshape(mean(reshape(sort(features(:,ind(1:numSamples))'), ...
                             [numSamplesPerBin numBins numFeatures]...
                                         )), [numBins, numFeatures])';
end



function qfs = getQFsFromWeightedSamples(features, weights, numBins);
  % Input
  %   features: a numFeatures by numSamples matrix of samples
  %   weights: a 1 by numSamples vector that gives the weight, or
  %     contribution, of each sample to the distribution
  %   numBins: the number of QF bins to use per feature
  % Output
  %   qfs: a numFeatures by numBins matrix that is
  %     a discrete QF for each feature
  % Approach
  %   1. For each feature sort the samples
  %   2. Linearly go through the samples to find the QF bin
  %     boundaries, which generally split a sample into two.
  %   3. Sum the samples in each bin as you go through the
  %     samples so that their average can be computed

  [numFeatures, numSamples] = size(features);
  qfs = zeros(numFeatures, numBins);

  for f = 1:numFeatures,
    [orderedFeatures indices] = sort(features(f,:));
    orderedWeights = weights(indices);
    totalWeight = sum(orderedWeights);
    wpb = totalWeight / numBins; %weight per bin
    qf = zeros(1, numBins);
    currentBinWeight = 0;
    currentBin = 1;
    i = 1;
    while(i <= numSamples)
      if(orderedWeights(i)+currentBinWeight <= wpb)
        %all of sample is in bin
        currentBinWeight = currentBinWeight + orderedWeights(i);
```

```
        qf(currentBin) = qf(currentBin) + orderedWeights(i)...
                         * orderedFeatures(i);
        i = i + 1;
      else
        %part of sample is in bin
        partial = wpb - currentBinWeight;
        qf(currentBin) = qf(currentBin) + partial * orderedFeatures(i);
        orderedWeights(i) = orderedWeights(i) - partial;
        currentBinWeight = 0;
        currentBin = currentBin + 1;
        if(currentBin == numBins + 1)
          break;
        end
      end
    end
    qf = qf / wpb;
    qfs(f, :) = qf;
  end
end



function qfs = getQFsFromDiscreteDiscribution(features, weights, numBins)
  % Input
  %   features: a numFeatures by numSamples matrix of samples from
  %     a distribution with integer values
  %   weights: a 1 by numSamples vector that gives the weight, or
  %     contribution, of each sample to the distribution
  %   numBins: the number of QF bins to use per feature
  % Output
  %   qfs: a numFeatures by numBins matrix that is
  %     a discrete QF for each feature
  % Approach
  %   1. For each feature compute a histogram with a bin for
  %     every possible discrete value
  %   2. Use the bin locations and frequencies as weighted
  %     samples for input into getQFsFromWeightedSamples()
  [numFeatures, numSamples] = size(features);
  qfs = zeros(numFeatures, numBins);
  for f = 1:numFeatures,
    samples = features(f,:);
    %Compute histogram
    binCenters = min(samples):max(samples);
    frequencies = zeros(1, size(binCenters, 2));
    for i = 1:numSamples,
      index = samples(i)-binCenters(1)+1;
      frequencies(index) = frequencies(index) + weights(i);
```

```
    end
    %Compute QF
    qfs(f,:) = getQFsFromWeightedSamples(binCenters, frequencies, numBins);
  end
end
```

A.2 Displaying an Estimated Smooth PDF From a QF

Throughout Chapters 2, 3 and 4 smoothed histograms are estimated from discrete quantile functions. This section gives two functions for converting between discrete QFs and PDFs. First, a function is given that directly estimates the adaptive bin histogram with equal frequency bins implied by the QF. Second, a function is given that smooths this histogram using a Gaussian kernel.

```
function [frequencies, binEdges, binWidths] = QFtoPDF(qf)
  % Input
  %   qf: a matrix where each column represents a discrete quantile function
  %     from a CONTINUOUS probability distribution. This code does not
  %     enforce a minimum bin width as required for discrete distributions
  % Output
  %   frequencies: the height of each histogram bin
  %   binEdges: the estimated edges between the bins (same # as bins)
  %   binWidths: the width of the histogram bins
  % Approach
  %   1. Pad the QF to facilitate the vectorized math
  %   2. Compute bin widths defined as half the distance to
  %     the quantiles on either side

  numBins = size(qf, 1);
  paddedQF = [2 * qf(1, :) - qf(2, :); qf; 2 * qf(end, :) - qf(end-1,:)];
  binWidths = 0.5 * (paddedQF(3:end, :) - paddedQF(1:end-2, :));
  frequencies = 1.0 / numBins ./ binWidths;
  binEdges = 0.5 * (paddedQF(2:end, :) + paddedQF(1:end-1, :));
end



function frequencies = QFtoSmoothPDF(qfs, commonBins);
  % Input
  %   qfs: a matrix where each row represents a discrete quantile function
```

```
  %     from a CONTINUOUS probability distribution. This code does not
  %     enforce a minimum bin width as required for discrete distributions
  %   commonBins: row vector of common bins for the histograms
  % Output
  %   frequencies: the height of each histogram bin
  % Approach
  %   Use QFtoPDF() to estimate the width of each adaptive histogram bin.
  %   Assume each quantile is a Gaussian with location given by the
  %   quantile and sigma given by the half width of its bin multiplied
  %   by a smoothing factor.

  smoothingFactor = 5;
  numQFs = size(qfs, 1);
  numBins = size(commonBins, 2);

  [dummy1, dummy2, binWidths] = QFtoPDF(qfs');
  frequencies = zeros(numQFs, numBins);

  for i = 1:numQFs,
    means = qfs(i, :);
    sigmas = smoothingFactor * binWidths(:, i)' / 2;
    for ii = 1:numBins,
      % Gaussian weights without 1/2pi constant that is normalized out
      frequencies(i, ii) = sum(exp(-0.5 * (commonBins(ii) - means) .^ 2 ...
                                   ./ (sigmas .^ 2)) ./ sigmas);
    end
      frequencies(i, :) = frequencies(i, :) / sum(frequencies(i, :));
    end
end
```

A.3 Computation of the QF Based Conditional Distribution Representation

This section gives a function for computing the QF based representation of conditional distributions presented in Section 2.2.2 on page 45. This function was used to compute the features of the Strong-MRF texture model presented in Section 3.3.1.

```
function qfs = getConditionalQFs(features, numCondBins, numProbBins);
  % Input
  %   features: a numFeatures by numSamples matrix of samples
  %             The first feature is assumed to be the conditioning variable
```

```
  %   numCondBins: number of partitions the conditioning var. is split into
  %   numProbBins: number of QF bins for each partition
  % Output
  %   qfs: a matrix that for row i is p(feature i | feature 1) represented
  %      as numCondBins numProbBins-bin QFs, where
  %      p(f1 | f1) is just the QF of f1 with numCondBins*numProbBins bins

  % Make integer number of points per bin
  [numFeatures, numSamples] = size(features);
  ind = randperm(numSamples);
  numCondSamplesPerBin = floor(numSamples/numCondBins/numProbBins)...
                         * numProbBins;
  numProbSamplesPerBin = numCondSamplesPerBin / numProbBins;
  numSamples = numCondSamplesPerBin * numCondBins;
  features = features(:, ind(1:numSamples));

  % Sort based on the first feature then average adjacent values in each bin
  qfs = zeros(numFeatures, numCondBins * numProbBins);
  [dummy, ordering] = sort(features(1, :));
  for f = 1:numFeatures,
    qfs(f, :) = reshape(mean(reshape(sort(reshape(features(f, ordering), ...
                                    numCondSamplesPerBin, numCondBins)), ...
                        [numProbSamplesPerBin, numProbBins, numCondBins])),...
                              1, numProbBins * numCondBins);
  end
end
```

A.4 Example: Displaying Figure 2.4(c)

This section supplies a function that puts together several of the functions given in this guide, demonstrating their use. The below function generates Figure 2.4(c) on page 19. This figure interpolates the QFs corresponding to two Gaussian distributions and displays the interpolated results as QFs and smoothed histograms. A similar pipeline was followed when producing many of the other figures in the dissertation, particularly Figures 2.13 (page 46), 3.4 (page 75), 4.4 (page 114), 4.5 (page 124), and 4.10 (page 141). These figures were key in understanding the QF subspaces estimated using PCA.

```
function figureExample()
  % Interpolate Gaussians N(0,1) and N(10, 3) in QF space
  % Display as QFs and smoothed histograms
```

```
  numSamples = 400000;
  numBins = 100;
  I = 5; % I-1 is the number of Interpolated distributions

  % Get Gaussian Samples
  a = randn(1, numSamples);
  b = randn(1, numSamples) * 3 + 10;

  % Get common bins for histograms using full domain of samples
  [dummy commonBins] = hist([a b], numBins);

  a = getQFs(a, numBins);
  b = getQFs(b, numBins);

  % This is how you could create QFs if the samples were weighted
  %a = getQFsFromWeightedSamples(a, ones(1, numSamples), numBins);
  %b = getQFsFromWeightedSamples(b, ones(1, numSamples), numBins);

  % Interpolate Gaussians in QF space and their colors
  for i = 0:I
    interpQFs(i+1,:) = (I-i)/I * a       + i/I * b;
    colors(i+1,:)    = (I-i)/I * [1 0 0] + i/I * [0 0 1];
  end

  % Display QFs
  figure; subplot(1, 2, 1); hold on;
  xlabel('p(X < x)');
  ylabel('Value');
  set(gca,'ColorOrder', colors);
  quantile = ((1:numBins)- 0.5)/numBins; % Domain of the QFs
  plot(quantile, interpQFs);

  % Compute and display smoothed histogram
  subplot(1, 2, 2); hold on;
  xlabel('Value');
  ylabel('Density');
  set(gca,'ColorOrder', colors);
  frequencies = QFtoSmoothPDF(interpQFs, commonBins);
  plot(commonBins, frequencies);
  % This is how to display the unsmoothed histograms
  %frequencies = QFtoPDF(interpQFs');
  %plot(interpQFs', frequencies);
end
```

BIBLIOGRAPHY

[Ama85] S Amari. *Differential-geometrical methods in statistics*. Springer-Verlag, New York, 1985.

[BA88] J R Bergen and E H Adelson. Early vision and texture perception. *Nature*, 333:363–364, May 1988.

[Bes74] J Besag. Spatial interaction and the statistical analysis of lattice systems. *Journal of Royal Statistical Society*, B-36:344–348, 1974.

[BJ83] J R Bergen and B Julesz. Parallel versus serial processing in rapid pattern discrimination. *Nature*, 303:696–698, June 1983.

[Blu04] S Blunsden. Texture classification using non-parametric markov random fields. Master's thesis, The University of EdinBurgh, 2004.

[BMP02] S Belongie, J Malik, and Jan Puzicha. Shape matching and object recognition using shape contexts. *PAMI*, 24(24):509–522, 2002.

[BPC+06] R E Broadhurst, S M Pizer, E L Chaney, J Levy, J Stough, G Tracton, and J Jeong. Automatic segmentation via a novel likelihood on regional intensity patterns. In *American Society for Therapeutic Radiology and Oncology (ASTRO)*, 2006.

[Bro66] P Brodatz. *Textures: A Photographic Album for Artists and Designers*. Dover, New York, 1966.

[Bro05] R E Broadhurst. Statistical estimation of histogram variation for texture classification. In *Texture*, 2005.

[BSPC05] R E Broadhurst, J Stough, S M Pizer, and E L Chaney. Histogram statistics of local model-relative image regions. In *DSSCV*, 2005.

[BSPC06] R. E. Broadhurst, J. Stough, S. M. Pizer, and E. L. Chaney. A statistical appearance model based on intensity quantile histograms. In *Proc. of IEEE Int. Symposium on Biomedical Imaging*, pages 422–425, 2006.

[CD01] O G Cula and K J Dana. Compact representation of bidirectional texture functions. In *CVPR*, 2001.

[CD04] O G Cula and K J Dana. 3d texture recognition using bidirectional feature histograms. *IJCV*, 59(1):33–60, 2004.

[CDA07] M Costa, H Delingette, and N Ayache. Automatic segmentation of the bladder using deformable models. In *IEEE Int. Symposium on Biomedical Imaging (ISBI)*, 2007.

[CET98] T F Cootes, G J Edwards, and C J Taylor. Active appearance models. In *ECCV*, 1998.

[CHM05] B Caputo, E Hayman, and P Mallikarjuna. Class-specific material categorisation. In *International Conference on Computer Vision*, 2005.

[CHTH93] T. F. Cootes, A. Hill, C. J. Taylor, and J. Haslam. The use of active shape models for locating structures in medical images. In *Proc. of Information Processing in Medical Imaging*, volume 687 of *Lecture Notes in Computer Science*, pages 33–47, 1993.

[CLvm] Chih-Chung Chang and Chih-Jen Lin. Libsvm : a library for support vector machines, 2001. Software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm.

[Cog82] J M Coggins. *A Framework for Texture Analysis Based on Spatial Filtering*. PhD thesis, 1982.

[CRM94] G. E. Christensen, R. D. Rabbit, and M. I. Miller. 3d brain mapping using a deformable neuroanatomy. *Phys. Med. Biol.*, 39:609–618, 1994.

[CS95] B B Chaudhuri and N Sarkar. Texture segmentation using fractal dimension. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 17(1):72–77, 1995.

[CT01] T. F. Cootes and C. J. Taylor. Statistical models of appearance for medical image analysis and computer vision. In *Proc. of SPIE Medical Imaging*, volume 4322, pages 236–248, 2001.

[CTCG95] T F Cootes, C J Taylor, D H Cooper, and J Graham. Active shape models their training and application. In *Computer Vision and Image Understanding*, volume 61, pages 38–59, 1995.

[CV01] T Chan and L Vese. Active contours without edges. In *IEEE Trans. Image Processing*, volume 10, pages 266–277, Feb. 2001.

[Dav70] H A David. *Order statistics*. John Wiley & Sons, New York, 1970.

[dct] Discrete cosine transform image from wikipedia, http://en.wikipedia.org/wiki/image:dctjpeg.png.

[DE87] H Derin and H Elliott. Modeling and segmentation of noisy and textured images using gibbs random fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9:39–55, 1987.

[DHS01] R Duda, P Hart, and D Stork. *Pattern classification*. John Wiley & Sons, New York, 2001.

[DJA79] L S Davis, S A Johns, and J K Aggarwal. Texture analysis using generalized cooccurrence matrices. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 1:251–259, 1979.

[DN98] K J Dana and S K Nayar. Histogram model for 3d textures. In *CVPR*, pages 618–624, 1998.

[DvGNK99] K Dana, B van Ginneken, S Nayar, and J Koenderink. Reflectance and texture of real world surfaces. *ACM Trans. on Graphics*, 18(1):1–34, 1999.

[FK01] S D Fenster and J R Kender. Sectored snakes: Evaluating learned-energy segmentations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(9):1028–1034, Sept. 2001.

[Fle04] P T Fletcher. *Statistical Variability in Nonlinear Spaces: Application to Shape Analysis and DT-MRI*. PhD thesis, 2004.

[FLPJ04] P T Fletcher, C Lu, S M Pizer, and S Joshi. Principal geodesic analysis for the study of nonlinear statistics of shape. In *IEEE Trans. on Medical Imaging*, volume 23, pages 995–1105, Aug. 2004.

[Fre48] M. Frechet. Les éléments aléatoires de nature quelconque dans un espace distancié. *Ann. Inst. H. Poincaré*, 10:215–310, 1948.

[FRZ+05] D. Freedman, R. J. Radke, T. Zhang, Y. Jeong, D. M. Lovelock, and G. T. Y. Chen. Model-based segmentation of medical imagery by matching distributions. *IEEE Trans. Medical Imaging*, 24(3):281–292, 2005.

[GD04] K Grauman and T Darrell. Fast contour matching using approximate earth movers distance. In *CVPR*, 2004.

[GS01] U Grenander and A Srivastava. Probability models for clutter in natural images. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 23(4):424–429, April 2001.

[GS05] J-M Geusebroek and A Smeulders. A six-stimulus theory for stochastic texture. *IJCV*, 62(1):7–16, 2005.

[GSS02] G Gerig, M Styner, and G Szekely. Statistical shape models for segmentation and structural analysis. In *IEEE Int. Symposium on Biomedical Imaging (ISBI)*, 2002.

[Har79] R M Haralick. Statistical and structural approaches to texture. *Proceedings of the IEEE*, 67:786–804, 1979.

[Has66] Victor Hasselblad. Estimation of parameters for a mixture of normal distributions. *Technometrics*, 8(3):431–444, August 1966.

[HB95] D J Heeger and J R Bergen. Pyramid-based texture analysis/synthesis. In *ACM SIGGRAPH*, 1995.

[HC71] J M Hammersley and P Clifford. Markov field on finite graphs and lattices. *Unpublished*, 1971.

[HCFE04] A Hayman, B Caputo, M Fritz, and J Eklundh. On the significance of real-world conditions for material classification. In *ECCV*, 2004.

[Hee93] D J Heeger. Modeling simple-cell direction selectivity with normalized, half-squared, linear operators. *Neurophysiology*, 70(5):1885–1898, 1993.

[Hit41] F L Hitchcock. The distribution of a product from several sources to numerous localities. *J. Math. Phys.*, 20:224–230, 1941.

[Hot33] H Hotelling. Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, 24:417441, 498520, 1933.

[HSD73] R M Haralick, K Shanmugam, and I Dinstein. Textural features for image classification. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-3(6):610–621, 1973.

[JDJG04] S Joshi, B Davis, M Jomier, and G Gerig. Unbiased diffeomorphic atlas construction for computational anatomy. *NeuroImage*, 23(Suppl. 1):S151S160, 2004.

[Jol86] I T Jolliffe. *Principal Component Analysis*. Springer-Verlag, 1986.

[Jos97] S Joshi. *Large Deformation Diffeomorphisms and Gaussian Random Fields for Statistical Characterization of Brain Submanifolds*. PhD thesis, 1997.

[Jul81] B Julesz. Textons, the elements of texture perception, and their interactions. *Nature*, 290:91–97, 1981.

[KG83] H Knutsson and G H Granlund. Texture analysis using twodimensional quadrature filters. In *IEEE Computer Society Workshop on Computer Architecture for Pattern Analysis and Image Database Management*, 1983.

[KvD87] J J Koenderink and A J van Doom. Representation of local geometry in the visual system. *Biological Cybernetics*, 55(6):367–375, 1987.

[KWT88] M Kass, A Witkin, and D Terzopoulos. Snakes: Active contour models. *Int. Journal of Computer Vision*, 1(4):321–331, 1988.

[LB01] E Levina and P Bickel. The earth movers distance is the mallows distance: Some insights from statistics. In *ICCV*, 2001.

[LBJ+07] J H Levy, R E Broadhurst, J Jeong, X Liu, J Stough, G Tracton, S M Pizer, and E L Chaney. Prostate and bladder segmentation using a statistically trainable model. In *American Society for Therapeutic Radiology and Oncology (ASTRO)*, 2007.

[LBR+07] J Levy, R E Broadhurst, S Ray, E L Chaney, and S M Pizer. Signaling local non-credibility in an automatic segmentation pipeline. *Medical Imaging 2007: Image Processing, (Josien P. W. Pluim and Joseph M. Reinhardt, eds.), published as Procedures of SPIE*, 6512, 2007.

[Lev02] E Levina. *Statistical Issues in Texture Analysis*. PhD thesis, University of California, Berkley, 2002.

[LFGW00] M. Leventon, O. Faugeras, E. Grimson, and W. Wells. Level set based segmentation with intensity and curvature priors. In *Proc. of Mathematical Methods in Biomedical Image Analysis (MMBIA)*, pages 4–11, 2000.

[LGL+07] J Levy, K Gorczowski, X Liu, S M Pizer, and M Styner. Caudate segmentation using deformable m-reps. In *MICCAI Workshop: 3D Segmentation in the Clinic: A grand challenge*, 2007.

[LM01] T Leung and J Malik. Representing and recognizing the visual appearance of materials using three-dimensional textons. *IJCV*, 43(1):29–44, 2001.

[LO06] H Ling and K Okada. Diffusion distance for histogram comparison. In *CVPR*, 2006.

[LW03] X Liu and D Wang. Texture classification using spectral histograms. *IEEE Trans. on Image Processing*, 12(6):661–670, June 2003.

[MBLS01] J Malik, S Belongie, T Leung, and J Shi. Contour and texture analysis for image segmentation. *International Journal of Computer Vision*, 43(1):7–27, 2001.

[MCAM] K Muller, Y-Y Chi, J Ahn, and J S Marron. High dimension, low sample size principal components; estimating eigenvalues of a singular wishart. *In preparation for resubmission to J. Amer. Stat. Assoc.*

[MD97] J Montagnat and H Delingette. Volumetric medical images segmentation using shape constrained deformable models. *CVRMed*, 1205:12–22, 1997.

[mea] The meastex image texture database, http://www.texturesynthesis.com/meastex/meastex.html.

[MP90] J Malik and P Perona. Preattentive texture discrimination with early vision mechanisms. *Journal of the Optical Society of America*, A-7:923–932, 1990.

[MP00] G McLachlan and D Peel. *Finite Mixture Models*. John Wiley & Sons, 2000.

[MTS+08] D Merck, G Tracton, R Saboo, J Levy, E L Chaney, S M Pizer, and S Joshi. Training models of anatomic shape variability. *in submission*, 2008.

[MV04] D. Terzopoulos M.A.O. Vasilescu. Tensortextures: Multilinear image-based rendering. In *ACM SIGGRAPH*, 2004.

[OPH96] T Ojala, M Pietikainen, and D Harwood. A comparative study of texture measures with classification based on featured distribution. *Pattern Recognition*, 29(1):51–59, 1996.

[Pag04] R Paget. Strong markov random field model. *PAMI*, 26(3):408–413, 2004.

[PBJ+06] S M Pizer, R E Broadhurst, J Jeong, Q Han, R Saboo, J Stough, G Tracton, and E L Chaney. Intra-patient anatomic statistical models for adaptive radiotherapy. In *MICCAI Workshop From Statistical Atlases to Personalized Models: Understanding Complex Diseases in Populations and Individuals*, 2006.

[PBL+07] S M Pizer, R E Broadhurst, J Levy, X Liu, J Jeong, J Stough, G Tracton, and E L Chaney. Segmentation by posterior optimization of m-reps: Strategy and results. In *submitted for journal review*, 2007.

[PD99] N Paragios and R Deriche. Coupled geodesic active regions for image segmentation: A level set approach. In *ECCV*, 1999.

[Pea01] K Pearson. On lines and planes of closest fit to points in space. *Philosophical Magazine*, 2:609629, 1901.

[PFF+03] S M Pizer, T Fletcher, Y Fridman, D S Fritsch, A G Gash, J M Glotzer, S Joshi, A Thall, G Tracton, P Yushkevich, and E L Chaney. Deformable m-reps for 3d medical image segmentation. *IJCV*, 55(2):85–106, 2003.

[PNMT04] M Pietikainen, T Nurmela, T Maenpaa, and M Turtinen. View-based recognition of real-world textures. *Pattern Recognition*, 37(2):313–323, February 2004.

[PRTB99] J Puzicha, Y Rubner, C Tomasi, and J M Buhmann. Empirical evaluation of dissimilarity measures for color and texture. In *CVPR*, 1999.

[Rac84] S T Rachev. The monge-kantorovich mass transference problem and its stochastic applications. *Theory of Probability and its Applications*, 29(4):647–676, 1984.

[RBR06] A Rajwade, A Banerjee, and A Rangarajan. Continuous image representations avoid the histogram binning problem in mutual information based image registration. In *IEEE Int. Symposium on Biomedical Imaging (ISBI)*, 2006.

[Ros02] S Ross. *A First Course in Probability*. Prentice-Hall, Inc., Upper Saddle River, New Jersey, 2002.

[RS00] S Roweis and L Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, December 2000.

[RTG00] Y Rubner, C Tomasi, and L J Guibas. The earth mover's distance as a metric for image retrieval. *IJCV*, 40(2):99–121, 2000.

[SBPC07a] J Stough, R E Broadhurst, S M Pizer, and E L Chaney. Clustering on local appearance for deformable model segmentation. In *International Symposium on Biomedical Imaging: From Nano to Macro (ISBI)*, 2007.

[SBPC07b] J Stough, R E Broadhurst, S M Pizer, and E L Chaney. Regional appearance in deformable model segmentation. In *Image Processing in Medical Imaging (IPMI)*, 2007.

[SCT03] I M Scott, T F Cootes, and C J Taylor. Improving appearance model matching using local image structure. In *IPMI*, 2003.

[SH98] P Suen and G Healey. Analyzing the bidirectional texture function. In *CVPR*, pages 753–758, 1998.

[Skl78] J Sklansky. Image segmentation and feature extraction. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-8:237–247, 1978.

[SPCR04] J Stough, S M Pizer, E L Chaney, and M Rao. Clustering on image boundary regions for deformable model segmentation. In *ISBI*, pages 436–439, Apr. 2004.

[TdSL00] J Tenenbaum, V de Silva, and J Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, December 2000.

[TJ98] M Tuceryan and A Jain. *The Handbook of Pattern Recognition and Computer Vision (2nd Edition)*, chapter 2.1: Texture Analysis. World Scientific Publishing Co., 1998.

[TMY78] H Tamura, S Mori, and Y Yamawaki. Textural features corresponding to visual perception. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-8:460–473, 1978.

[TSM85] D. M. Titterington, A. F. Smith, and U. E. Makov. *Statistical Analysis of Finite Mixture Distributions*. John Wiley & Sons, 1985.

[TWT+03] A Tasi, W Wells, C Tempany, E Grimson, and A Willsky. Coupled multi-shape model and mutual information for medical image segmentation. In *Information Processing in Medical Imaging (IPMI)*, 2003.

[TYW+03] A Tsai, A Yezzi, W Wells, C Tempany, D Tucker, A Fan, W E Grimson, and A Willsky. A shape-based approach to the segmentation of medical imagery using level sets. *IEEE Trans. Medical Imaging*, 22(2), Feb. 2003.

[VG07] M Varma and R Garg. Locally invariant fractal features for statistical texture classification. In *IEEE International Conference on Computer Vision*, 2007.

[vis] The mit vision texture database, http://vismod.media.mit.edu/vismod/imagery/visiontexture/vistex.html.

[VP88] H Voorhees and T Poggio. Computing texture boundaries from images. *Nature*, 333:364–367, May 1988.

[VR07] M Varma and D Ray. Learning the discriminative power-invariance trade-off. In *IEEE International Conference on Computer Vision*, 2007.

[VV88] R L De Valois and K K De Valois. *Spatial Vision*. Oxford Univ. Press, New York, 1988.

[VZ02] M Varma and A Zisserman. Classifying images of materials: achieving viewpoint and illumination independence. In *ECCV*, 2002.

[VZ03] M Varma and A Zisserman. Texture classification: are filter banks necessary? In *CVPR*, 2003.

[ZS06] Y Zhan and D Shen. Deformable segmentation of 3-d ultrasound prostate images using statistical texture matching method. *IEEE Transactions on Medical Imaging*, 25(3):256–272, 2006.