Universitat
Autònoma
de Barcelona

Generative Models for Video Analysis and 3D Range Data Applications

A dissertation submitted by **Xavier Orriols Majoral** at Universitat Autònoma de Barcelona to fulfil the degree of **Doctor en Informática**.

Bellaterra, January 8, 2004

Director: **Dr. Xavier Binefa**
Universitat Autònoma de Barcelona
Dept. Informàtica & Computer Vision Center

**Universitat
Autònoma
de Barcelona**

This document was typeset by the author using LaTeX 2_ε.

The research described in this book was carried out at the Computer Vision Center, Universitat Autònoma de Barcelona.

ISBN 84–932156–9–4

Printed by Ediciones Gráficas Rey, S.L.

A la memòria de Maria Falgars Orriols

Acknowledgements

Aquesta tesi ha estat realitzada amb la beca predoctoral de Formació d'Investigadors pròpia del Departament d'Informàtica de la Universitat Autònoma de Barcelona. Part de la recerca presentada en aquesta memòria ha estat finançada per la CICYT TEL 99-1206-C02-02.

En primer lloc, vull agrair la formació que he rebut tan des d'un punt de vista acadèmic com professional i personal al Dr. Xavier Binefa, director, soci i amic. Aprecio enormement la seva confiança dipositada, els seus savis consells, la seva capacitat d'il·lusionar i emprendre noves aventures, la seva atenció i predisposició al diàleg, el seu suport i la seva ajuda en tot moment.

Agraeixo al Dr. Juan José Villanueva la oportunitat d'entrar a col·laborar al Centre de Visió per Computador i al idiosincràtic món de la recerca universitària.

Part dels treballs presentats en aquesta tesi són conseqüència de dues estades al SHAPE Lab de Brown University, Providence Rhode Island. Mantinc un gran record de l'experiència professional del Professor David B. Cooper que em va guiar en tot moment oferint-me la seva experiència i saviesa en l'àmbit de la Visió per Computador i, més concretament, en l'entorn de les tècniques aplicades a dades 3D range. A Providence vaig mantenir les més encoratjadores i estimulants discussions de recerca amb el Dr. Frederic Leymarie, amb el meus col·legues Andrew Willis, Kongbin i Yan Cao i el Dr. Jean Phillipe Tarel. La meva integració a l'altre costat de l'Oceà l'agraeixo a en Daniel Acebedo, que va fer de Ciceró en terra estranya. Un munt de converses sobre models lineals les vaig mantenir amb el Dr. Fernando de la Torre, sense deixar passar per alt totes les experiències viscudes enfocades a la recerca lúdica, que també són de vital importància.

La materialització de les idees en aplicacions concretes que funcionin en entorns reals hauria de ser el desig de tot investigador. Aquesta part la dec a en Ferran Muñoz, que va fer possible que els algorismes de reconeixement basats en models locals lineals fossin aplicats a una màquina d'inspecció de llavors per a la producció d'oli dins el projecte Cèl·lula d'Inspecció Flexible del CeRTAP de la Generalitat de Catalunya. Em fa especialment feliç que estiguem col·laborant junts en una nova aventura actualment.

Agraeixo als meus companys de despatx Jordi González, David Guillamet, Juanma Sánchez, Ramon Felip i Lluís Barceló el seu suport i la seva tolerància a la meva condició de físic en un entorn d'informàtics.

El treball realitzat en aquest tesi referent a Optical Flow ha estat realitzat amb la col·laboració de'n Lluís Barceló, es per això que particularment li agraeixo les agradables hores de treball conjunt.

A en Ramon Felip per les llargues estones de xerrades i aventures on ens ha tocat jugar fort.

Especialment, agraeixo a en Juanma Sánchez els agradables i grans moments passats tan a dins l'entorn de la recerca, no sé on seria sense la seva vital ajuda, com a fora en terres estranyes (EEUU, França i Alemània) i en terres més properes on em vaig sentir com un més de la família (Cádiz). Gran amic i company de viatge!

A en Marco Bressan, un excel·lent investigador, amb qui he tingut el gust de compartir temps de diàleg, motivacions i inquietuds comunes de recerca. Moltes idees que surten en aquesta tesi han estat conseqüència de les altament productives estones de cafè amb ell. La seva arribada al Centre de Visió per Computador va contribuir a allunyar el sentiment de solitari bitxo raro teòric que tenia sobre mi mateix donat un entorn eminentment pragmàtic.

Vull dedicar un especial agraïment a el Dr. Jordi Vitrià per la seva ajuda i plena confiança fent possible que gaudís de la docència i sobretot proporcionant-me tots els coneixements sobre Intel·ligència Artificial.

A en Marc Navarro per la seva altruista paciència a l'hora d'ensenyar-me el que ara sé de programació en JAVA.

A la gent del Centre de Visió per Computador que fa possible el seu funcionament: Joan, Raquel, María Jose, Pilar, Mari Carmen, Montse, i, no menys important, neteja i manteniment. Gràcies a la resta de companys del CVC: tothom@cvc.uab.es. Especialment a en Ricardo pel treball conjunt realitzat durant els inicis d'aquesta tesi.

Dedico un especial agraïment als meus companys de Visual Century: Alba Sort (per la seva ajuda en l'anglès), Enric Carrión, Ferran Bonàs, Ferran Muñoz, Gerard Tortajada, Josep Cristià, Juanma Sánchez, Marco Bressan, Marc Navarro i Sabrina Ciarlo, per la seva paciència i suport en el transcurs de la recta final d'aquesta tesi.

Agraeixo al tribunal per la seva paciència i pel seu temps a la lectura d'aquesta memòria.

Finalment, el més important. A la meva família i amics per la seva paciència. Especialment a la meva mare Engràcia per seu suport i esforços incondicionals i per la seva paciència. Al Benet, al Miquel i al Roger, al Pere i la Núria, a la Maria, moltes gràcies. A Maria Falgars Orriols, sense el seu ajut i la seva immensa generositat no hauria pogut mai accedir a uns estudis universitaris, ni fer aquesta tesi. Espero i desitjo que, des d'allà on siguis puguis, veure el final d'aquesta tesi i el principi de les noves aventures, i que tard o d'hora ens tornem a trobar. Et portaré una copia d'aquesta memòria i la llegirem junts.

Abstract

The majority of problems in Computer Vision do not maintain a direct relation between the stimuli provided by a general-purpose sensor and its corresponding perceptual category. A complex learning task must be involved in order to provide such a connection. In fact, the basic forms of energy, and their possible combinations, are a reduced number compared to the infinite possible perceptual categories corresponding to objects, actions, relations among objects, etc. In addition, the observations provided by a sensor correspond to a set of variables, whose number is rather larger than desired. More specifically, in Computer Vision, the most used sensors are digital cameras and laser scanners. Both represent objects in terms of a set of variables that are highly correlated. For instance, given a set of different images from objects concerning the same category, the variability in each pixel value is constrained to a certain range, which is influenced by the rest of pixel value variations. In this sense, feature extraction can be seen as a dimensionality reduction or coding procedure, or, in general, as a representation in a different coordinate system.

The lines of research of this thesis are directed to the management of multiple data sources, with straightforward applications such as within the MPEG 4 and 7 framework. More specifically, this thesis presents a set of new algorithms applied to two different areas in Computer Vision: i) video analysis and summarization, and ii) 3D range data. Both areas have been approached through the Generative Models framework, where similar protocols for representing data have been employed.

Generative Models introduce a reduced number of hidden variables (Latent Variables) under the assumption of being the underlying causes that produce the observed phenomena. Latent Variables represent the intrinsic degrees of freedom that govern the essential structure of the observations. On the other hand, the accidental structure of the observed data is modelled as noise through the introduction of a probabilistic approach, which, consequently, not only provides a distance measure for posterior recognition tasks, but also a manner of including knowledge of the domain in terms of a priori information. The use of latent variables for dimensionality reduction has the purpose of combating the curse of dimensionality while retaining sufficient accuracy of representation.

The choice of an appropriate representation for the data takes a significant relevance when it comes to deal with symmetries, since these usually imply that the number of intrinsic degrees of freedom in the data distribution is lower than the co-

ordinates used to represent it. Indeed, this means that the problem can be reduced to a lower dimensional one. Therefore, the decomposition into basic units (model parameters) and the change of representation, make that a complex problem can be transformed into a manageable one. This simplification of the estimation problem has to rely on a proper mechanism of combination of those primitives in order to give an optimal description of the global complex model. This thesis shows how the process that reduces dimensionality, taking into account the internal symmetries of a problem, provides a manner of dealing with missing data and makes possible predicting new observations. The connection between Lie's group theory, the structural equation and the internal symmetries present in the observed data is also pointed out.

Two main points of discussion have an essential presence in this thesis: i) the combination of global and local information, and, ii) the complexity of a model in terms of linearity and non-linearity.

Linear models are quite useful since their simplicity, low computational cost and interpretability (from a geometrical point of view). However, there are situations where the distribution of data can be more complex than a linear model can cope with. In order to cope with global nonlinear behaviors we study the advantages of dealing with local combinations of linear sub-models. The advantage of finite mixtures of latent variables models is that they can place different latent variable models in different regions of data space, where each latent variable model models locally the data. This allows the use of simple local models (e.g. linear-normal, like Factor Analysis or Principal Component Analysis) that build a complex global model (piecewise linear-normal). In other words, finite mixtures of latent variable models combine clustering with dimensionality reduction.

The probabilistic approach given to the formulation of the models presented in this thesis permits: i) the combination of several probabilistic methods in a mixture in a natural way, ii) comparing a method with other probabilistic methods as well as constructing statistical tests, iii) the prediction of any variable(s) as a function of any other variable(s) by using conditional probabilities, and iv) the natural extension to Bayesian analysis for model comparison through the use of prior distributions as well as the inclusion of external sources of knowledge.

Contents

List of Figures

Chapter 1

Introduction

A vast majority of problems in Computer Vision and Pattern Recognition deal with the need to find a suitable manner for representing data. This pursuit has become the central problem in Computer Vision since its beginnings. For a long time, one of the main motivations has been the limitations of existing computer processors and the need to minimize computational costs. With the subsequent improvement of computer processors, a wider range of possibilities has been progressively presented to researchers, giving them the tools to face more ambitious tasks. Accuracy and reliability are also relevant factors that constrain the selection process of a specific representation.

With regards to representation selection, there are two main factors that determine the level of difficulty and solvability of a specific problem: i) the different levels of information that are employed, and ii) the complexity of the model which is intend to explain the observations.

Information can be extracted from data through either *local* or *global* measurements. For instance, in relation to images, we can perform pixel-based operations (neighboring filters, statistical measurements, mathematical morphology, etc...) to obtain local information, whereas, on the other hand, we can apply some global transformation (Principal Component Analysis, Factor Analysis, Positive Matrix Factorization, etc...) to the raw image data to retrieve some global information. Although both approaches work well to a certain extent, much work done in Computer Vision limits problems to be tackled by solely one of these two approaches.

The combination of both *global* and *local* information is the basis of the different Computer Vision applications presented in this thesis. The scope of this thesis includes two major blocks: one dedicated to analyze sequences of images, and another focussed on studying 3D range data. Both sections have in common the way the different levels of information complement each other, as well as, the manner models are built in order to solve each problem.

First, we consider the analysis of video sequences since it offers the possibility of working together with spatial and temporal information at the same time, making problems more challenging and their solutions very interesting. This interest not only concerns the research community, but also the industry as a whole. Content-based video browsing and retrieval in video databases is becoming a relevant field in Computer Vision and Multimedia. This fact goes in accordance with the increasing developments in digital storage and transmission. In addition to this, the wide range of applications in this framework, such as advertising, publishing, news and video clips, points out the necessity for more efficient and organizing techniques. We focus on three important subjects related to Multimedia: video segmentation, preview and summarization, which make feasible a quick intuition of the evolution of a sequence, (under low streaming cost), of higher-level perceptual structures, such as stories, scenes or pieces of news. Moreover, analyzing and structuring video through summarization and hyper-linking permits a continuous media to be *seekable*.

Another interesting and challenging area of Pattern Recognition involves 3D *range data*, (distributions of 3D unorganized points obtained by means of CT and laser scanners). Many interesting applications concern 3D range data: segmentation, object recognition, perceptual organization, applications to archaeology pottery reconstruction, sherds classification, the 3D puzzle problem, etc... These applications usually deal with broken pieces and patches which can be characterized a set of features such as, axis of symmetry, breaking curves (3D contour), thickness, etc...

The purpose of the following sections is to introduce the **Latent Variable** framework that provides a methodology for building models that learn patterns from observations under a probabilistic approach. *Noise*, *similarity*, and *complexity* (linearity and nonlinearity) concepts arise straightforwardly from this formulation. Considering the specific forms of such factors, the most well known models are described and located inside the latent variables' taxonomy. Particularly, two of these models (Principal Component Analysis and Mixtures of Principal Components) are detailed in order to show the fundamental role of latent variable models when it comes to reducing dimensionality. Moreover, a geometrical interpretation for these two specific models is given. The aim of this is to unveil the main motivations that lead to employing these techniques when it comes to dealing with not only **recognition** and **classification**, but also **data visualization**, and **feature extraction**.

One section is dedicated to explaining the *Expectation-Maximization* algorithm through a variational approach for estimating parameters and latent variables for a given set of observations. Such an approach not only proofs convergence under some specific requirements for the likelihood function, but also yields a manner of dealing with certain constrained problems[1] thanks to the introduction of lower bound functions.

The last section introduces the scope of this thesis, linking the previously introduced methodology with the applications and algorithms that are presented in the following chapters.

[1] This type of problems often are difficult to estimate with traditional Least Squares techniques [58].

1.1 Generative Modelling

In both *video analysis* and *3D range data* situations, the aim is to extract as much *essential* structure as possible from a data set without modelling any of its *accidental* structure (e.g. noise and sampling artifacts). In this sense, *data* refers to both video sequences and 3D points. For instance, these variabilities may refer to: i) the behavior of the curvature of an axially symmetric surface represented by a distribution of points, or to ii) video sequences regarding some specific camera movements, color, temporal linking among different camera shots in an interview, etc... Both cases have in common the fact that data must be modelled in order to explain the underlying phenomena. To this end, we study the possibilities that offer *Generative Models* for Pattern Recognition.

In this framework, modelling data has a twofold purpose: i) the *inference of the causes* that originate such characteristic variabilities, which is a task of analyzing similarities between observations that "look" similar, and ii) the estimation of a stochastic contribution associated to the term *"noise"*.

Existing methods for modelling data fall into two categories: *descriptive* and *generative* methods. Descriptive methods characterize visual patterns by extracting some feature statistics and imposing constraints at a "signal" level, such as geometrical distributions, predefined features, Markov Random Fields, Minimax Entropy Learning, Deformable Models, etc. The purposes of generative methods are instead: i) *learning features from observations* and ii) capturing high-level *semantics.*

Both generative and descriptive methods require: i) an internal *representation* for describing the observations, as well as, ii) a *distance* measure, which provides a manner of determining the ownership of a certain test sample to a specific learned pattern. Probabilistic techniques couple both representation and distance concepts under the same framework, where it is possible modelling noise and the essential structure of a specific pattern from the observations at the same time. Descriptive models are built directly on original signals yielding probability densities of very high dimensions. Operations such as inference, sampling, and estimating probability densities are difficult to perform when dealing with high dimensional feature vectors (curse of dimensionality).

1.2 Latent Variables

Generative models introduce a set of hidden variables (semantics) under the assumption of being the underlying causes that produce observed phenomena. Usually, a small number of these hidden (latent) variables are sufficient for describing the observed phenomena. The use of latent variables for dimensionality reduction has the purpose of combating the *curse of dimensionality*[2] while retaining sufficient accuracy

[2] The term curse of the dimensionality, coined by Bellman (1961)[8], refers to the fact that, in the absence of simplifying assumptions, the sample size needed to estimate a function of several variables to a given degree of accuracy (i.e., to get a reasonably low-variance estimate) grows exponentially

of representation. Operations like **prediction** and **compression** become easier and rigorously justifiable.

Four components are necessary to build a generative model based on this causal approach:

1. The **latent variables**, which represent the *intrinsic degrees of freedom* that govern the essential structure of the observations. These fall into two main categories: discrete and continuous.

2. The **model parameters**, which provides a *compact internal representation* for the observations.

3. The **structural equation**, which defines how to combine the latent variables in order to produce the observed phenomena.

4. The **noise model** that permits introducing a probabilistic approach to the model, and therefore, a *distance measure* for posterior recognition tasks.

To build a general framework, we will consider that the latent variables are *mapped* by a fixed transformation into a higher-dimension observed space (measurement procedure) and noise is added there (stochastic variation). In contrast with this *generative*, bottom-up point of view[3], statisticians often consider latent variable models from an *explanatory*, top-down point of view (Bartholomew, 1987) [7]: given the empirical correlation between the observed variables, the mission of the latent variables is to explain those correlations via the *axiom of local independence* ; i.e., given an observed distribution, find a combination of latent distribution and noise model that approximates it well. We will consider that both the observed and the latent variables are continuous. Nevertheless, the extension to discrete variables is straightforward.

Let $\mathcal{T} \subseteq \Re^d$ be the d-dimensional data or **observed space**. Consider an unknown distribution $p(\mathbf{t})$ in the data space, for $\mathbf{t} \in \mathcal{T}$, of which we only can observe a finite sample $\{\mathbf{t}_1, \ldots, \mathbf{t}_N\}$. The latent variables are represented by a lower dimensional space $\mathcal{X}$, where $\dim(\mathcal{X}) < \dim(\mathcal{T})$.

with the number of variables. A related fact, responsible for the curse of the dimensionality, is the *empty space phenomenon* (Scott and Thompson, 1983 [91]): high-dimensional spaces are inherently sparse. For example, for a one-dimensional standard normal $N(0,1)$, 70% of the mass is at points contained in a sphere of radius one standard deviation (i.e., the $[-1,1]$ interval), for a 10-dimensional $N(0,I)$, that same (hyper)sphere contains only 0.02% of the mass and one has to take a radius of more than 3 standard deviations to contain 70%.Therefore, and contrarily to our intuition, in high-dimensional distributions the tails are much more important than in one-dimensional ones. This is a difficult problem in multivariate density estimation, as regions of relatively very low density can contain a considerable part of the distribution, whereas regions of apparently high density can be completely devoid of observations in a sample of moderate size (Silverman, 1986 [94]). The curse of the dimensionality has the following consequence for density estimation: since most density estimation methods are based on some local average of the neighboring observations), in order to find enough neighbors in high-dimensional spaces, the neighborhood has to reach out farther and the locality is lost.

[3] All the observations are assumed to be caused by *latent variables*, that is, the observations are assumed to be at the end of the causal chain.

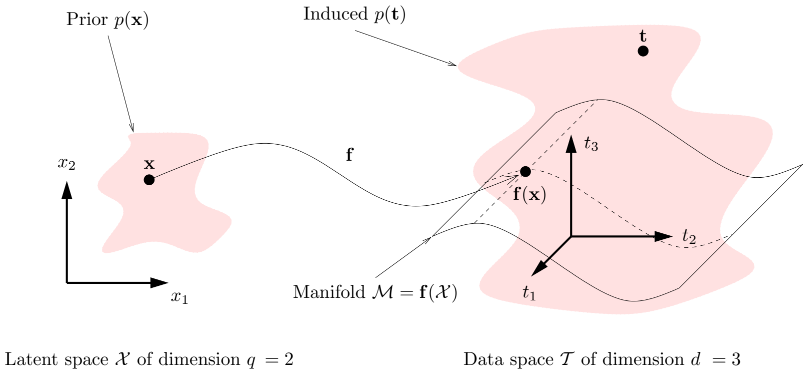

Prior $p(\mathbf{x})$
Induced $p(\mathbf{t})$
$\mathbf{t}$
x_2
$\mathbf{x}$
$\mathbf{f}$
t_3
$\mathbf{f}(\mathbf{x})$
t_2
x_1
Manifold $\mathcal{M} = \mathbf{f}(\mathcal{X})$
t_1
Latent space $\mathcal{X}$ of dimension $q = 2$
Data space $\mathcal{T}$ of dimension $d = 3$

Figure 1.1: Schematic of a continuous latent variable model with a 3-dimensional data space and a 2-dimensional space.

Therefore, a point in the **latent space** is generated according to a *prior distribution* $p(\mathbf{x})$ and it is mapped onto data space $\mathcal{T}$ by a smooth, non-singular mapping:

$$f[W] : \mathcal{X} \to \mathcal{T} \tag{1.1}$$

where W corresponds to the model's parameters. This structural equation modelling is a commonly used statistical method for quantifying the relationships among variables that cannot be observed directly. The overall model for the observed variables consists of two parts; the measurement model relating the observed indicators to the latent variables or factors, and the underlying structural model expressing a relationship among the unobservable variables (fig. 1.2). The success of the model depends on how well it can capture the structure of the phenomena underlying the observations.

Since $\mathcal{M} = f(\mathcal{X}, W)$ is a q-dimensional manifold in $\mathcal{T}$, in order to extend it to the whole d-dimensional space, a distribution $p(\mathbf{t} \mid \mathbf{x}) = p(\mathbf{t} \mid f(\mathbf{x}))$ on $\mathcal{T}$, called the **error** or **noise model**. Figure 1.1 illustrates the idea of latent variable models. Observations can be affected by many variables that may not be conveniently modelled deterministically because they are too complex or to hard to observe, and often, belong to categories of events that are irrelevant to the observations of interest. In these cases, it is necessary to treat some of them as noise. In fact, high-dimensionality arises for several reasons, including stochastic variation and the measurement process.

The joint probability density function in the product space $\mathcal{T} \times \mathcal{X}$ is $p(\mathbf{t}, \mathbf{x})$, and the integration over the latent space yields the marginal distribution in data space:

$$p(\mathbf{t}) = \int_{\mathcal{X}} p(\mathbf{t}, \mathbf{x}) d\mathbf{x} = \int_{\mathcal{X}} p(\mathbf{t} \mid \mathbf{x}) p(\mathbf{x}) d\mathbf{x} \tag{1.2}$$

This is called the **fundamental equation**[4] of latent variable models by Bartholomew

[4] Equation (1.2) can also be seen as a continuous mixture model with the latent variable $\mathbf{x}$ "indexing" continuously the mixture component. In fact, any density function can be considered as a mixture density where extra variables have been integrated over.

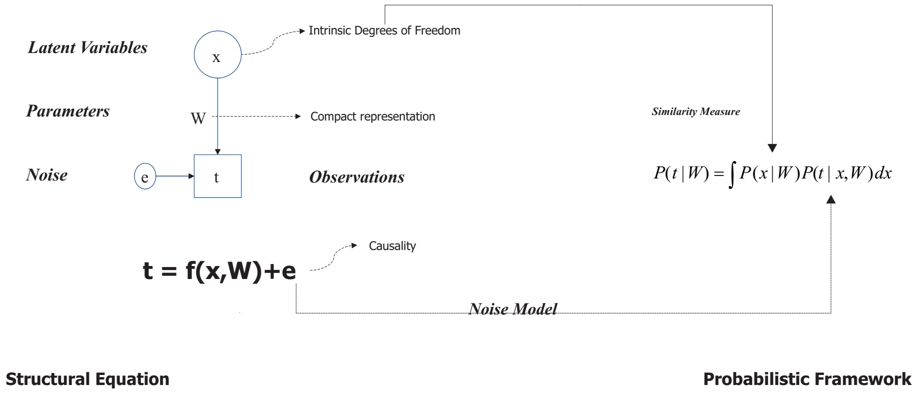


Figure 1.2: Causal interpretation of Latent Variable Models

(1984). Thus, a model is essentially a specification of $p(\mathbf{x})$ and $p(\mathbf{t} \mid \mathbf{x})$-the specification of the mapping f can be absorbed into that of $p(\mathbf{t} \mid \mathbf{x})$ (see fig. 1.2). The only empirical evidence available concerns $p(\mathbf{t})$ through the sample $\{\mathbf{t}_1, \ldots, \mathbf{t}_N\}$ and so, the only constraint on $p(\mathbf{x})$ and $p(\mathbf{t} \mid \mathbf{x})$, apart from the need to be nonnegative and integrate to 1, is given by eq. (1.2). In general, there are many combinations of $p(\mathbf{x})$ and $p(\mathbf{t} \mid \mathbf{x})$ that can satisfy (1.2) for a given $p(\mathbf{t})$.

If the latent variables are to be efficient in representing faithfully the observed variables, we should expect that, given a value for the latent variables, the values of any group of observed variables are independent of the values of any other group of observed variables. Otherwise, the chosen latent variables would not completely explain the correlations between the observed variables and further latent variables would be necessary. Thus, for all $k \in \{1, \ldots, d\}$, the observed variables conditioned on the latent variables (noise model), is factorial:

$$p(\mathbf{t} \mid \mathbf{x}) = \prod_{k=1}^{d} p(t_k \mid \mathbf{x}) \tag{1.3}$$

That is, for some $q \leq d$, the observed variables are conditionally independent given the latent variables. This is usually called the **axiom of local (or conditional) independence** (Bartholomew, 1984; Everitt, 1984). It should be noted that, rather than an assumption, the axiom of local independence is a definition of what it means to have fully explained the joint distribution of the observed variables in terms of the latent ones.

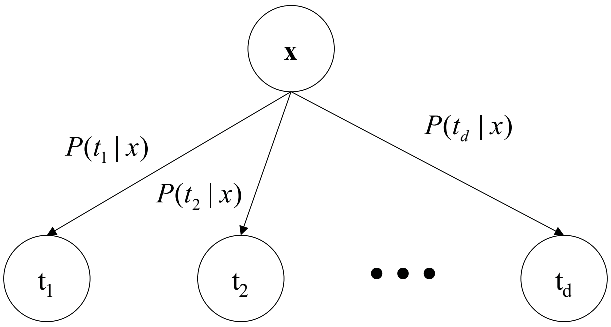


Figure 1.3: Axiom of conditional independence

1.3 Expectation-Maximization Algorithm for Parameter Estimation

A latent variable model is specified by the functional forms of:

- the prior model in latent space $p(\mathbf{x})$
- the smooth mapping $f : X \to T$ from latent space to data space, i.e. the causal relation between latent variables and observations.
- the noise model in data space $p(\mathbf{t}|\mathbf{x})$

all of which are equipped with parameters. For the sake of notation, we write them collectively through θ. The aim is to **estimate** these parameters from a finite set of observations $\{\mathbf{t_1}, \ldots, \mathbf{t_N}\}$, while **inferring** the latent variables at the same time. Under the selection of a certain model, this goal means finding those parameters and latent variables whose *likelihood* of explaining the observations is maximum. Given that generative models can be approached through probability distributions, the term *"likelihood"* is directly quantified by the joint probability in equation 1.2.

Usually, maximum likelihood maximization is carried out using a form of the **Expectation-Maximization** (EM) algorithm (Dempster et al., 1977 [31];McLachlan and Krishnan, 1997 [66]), which is usually simpler than other optimization techniques and is guaranteed to increase the log-likelihood monotonically.

In the EM approach to latent variable models, the latent variables $\{\mathbf{x}\}_{n=1}^{N}$ (one per data point) are considered missing[5]. If their values were known, estimation of the parameters θ would be straightforward by least squares. However, for a given data point tn we do not know the value of $\mathbf{x}_n$ that generated it. The EM algorithm is a fixed-point fashion procedure that operates in two steps, which are repeated alternatively until convergence:

[5] Depending on the model, additional missing variables may have to be introduced. For example, the component labels for mixture models.

Expectation Compute the sufficient statistics for the latent variables posterior distributions $p(\mathbf{x}_n|\mathbf{t}_n, \theta)$. Each observation $\mathbf{t}_n$ has a specific way of combining the latent variable values $\mathbf{x}_n$. These are **inferred** from each corresponding data point $\mathbf{t}_n$ and the parameters of the model θ. Inference occurs when computing the values for $\mathbf{x}_n$ that maximize the *a posteriori* probability of a given data point $\mathbf{t}_n$ and a specific instance for the model's parameters θ.

Maximization Maximize the joint likelihood with respect to the parameters θ. At this point, the corresponding values for the latent variables estimated in the previous step are fixed.

The standard EM algorithm has some disadvantages:

- It is a batch algorithm. However, by interpreting EM as an alternating maximization of a negative free-energy-like function (Neal and Hinton, 1998 [69]), it is possible to derive online EM algorithms, suitable for online learning (e.g. in sequential tasks, where the data come one at a time). In this thesis, we present an *online* EM algorithm for video segmentation and hyper-linking.

- Its slow convergence after the first few steps, which are usually quite effective. Also, the greater the proportion of missing information, the slower the rate of convergence of EM (Dempster et al., 1977, [31]). However, methods for accelerating it are available; see, for example, McLachlan and Krishnan (1997) [66] and references therein.

Despite these shortcomings, EM usually remains the best choice for parameter estimation thanks to its reliability. For a general choice of prior in latent space, mapping between latent and data space and noise model the log-likelihood surface can have many local maxima of varying height. In some cases, some or all of those maxima are equivalent, in the sense that the model produces the same distribution (and therefore the same log-likelihood value at all the maxima), i.e., the model is not identifiable. This is often due to symmetries of the parameter space, such as permutations (e.g. Principal Component Analysis, PCA) or general rotations of the parameters (e.g. Factor Analysis). In those cases, the procedure to follow is to find a first maximum likelihood estimate of the parameters (in general, by some suitable optimization method, e.g. EM, although sometimes an analytical solution is available, as for PCA) and then possibly apply a transformation to them to take them to a canonical form satisfying a certain criterion.

1.3.1 Variational Approach of the EM Algorithm

In this section, a variational approach of the EM algorithm is described in order to show the mathematical origin of these "two steps", as well as, to demonstrate monotonically increasing of the log-likelihood while iterating the process. Variational methods has been used in the past few years to approximate untractable integrals [52, 58, 16]. This framework involves introducing a set of distributions $Q_n(\mathbf{x}_n)$ that provide an approximation to the true posterior distributions $p(\mathbf{x}_n|\mathbf{t}_n, \theta)$. In this algorithm, two main hypotheses govern the nature of the joint probability distribution $p(\mathbf{t}_1, \ldots, \mathbf{t}_N|\theta)$ for the data set $\{\mathbf{t}_1, \ldots, \mathbf{t}_N\}$:

- The joint probability $p(\mathbf{t}_1, \ldots, \mathbf{t}_N|\theta)$ is log-concave.
- The data set $D = \{\mathbf{t}_1, \ldots, \mathbf{t}_N\}$ is independent and identically distributed, i.e.:

$$p(\mathbf{t}_1, \ldots, \mathbf{t}_N|\theta) = \prod_{n=1}^{N} p(\mathbf{t}_n|\theta) \tag{1.4}$$

Using the second hypothesis, the logarithm of the likelihood function in equation (1.2) can be written as follows:

$$\mathcal{L} \equiv \log\left[p(\mathbf{t_1}, \ldots, \mathbf{t_N}|\theta)\right] = \sum_{n=1}^{N} \log p(\mathbf{t}_n|\theta) \tag{1.5}$$

Taking into account that the marginal distribution for the observations comes from integrating the joint distribution $p(\mathbf{t}_n, \mathbf{x}_n|\theta)$ over the latent space, this latter equation (1.5) can be expressed in the following way:

$$\mathcal{L} = \sum_{n=1}^{N} \log p(\mathbf{t}_n|\theta) \quad = \quad \sum_{n=1}^{N} \log\left[\int p(\mathbf{t}_n, \mathbf{x}_n|\theta) d\mathbf{x}_n\right]$$

where a set of arbitrary distributions $Q_n(\mathbf{x}_n)$ can be introduced without losing generality:

$$\mathcal{L} = \sum_{n=1}^{N} \log p(\mathbf{t}_n|\theta) \quad = \quad \sum_{n=1}^{N} \log\left[\int Q_n(\mathbf{x}_n) \frac{p(\mathbf{t}_n, \mathbf{x}_n|\theta)}{Q_n(\mathbf{x}_n)} d\mathbf{x}_n\right]$$

By means of Jensen's inequality, a lower bound to this log-likelihood can be computed:

$$\begin{aligned}\mathcal{L} &= \sum_{n=1}^{N} \log\left[\int Q_n(\mathbf{x}_n) \frac{p(\mathbf{t}_n, \mathbf{x}_n|\theta)}{Q_n(\mathbf{x}_n)} d\mathbf{x}_n\right] \geq \\ &\geq \sum_{n=1}^{N} \left[\int Q_n(\mathbf{x}_n) \log \frac{p(\mathbf{t}_n, \mathbf{x}_n|\theta)}{Q_n(\mathbf{x}_n)} d\mathbf{x}_n\right] = \mathcal{F}(Q, \theta)\end{aligned}$$

This lower bound is known as a *free energy* term (Hinton 1998 [69]; Jordan 1999 [52]; Lawrence 2000 [58]), and, it corresponds to the sum of the Kullback-Leibler divergence (of the approximating Q-functions and the true posterior) and the marginal log-likelihood:

$$\mathcal{F}(Q, \theta) \quad = \quad \sum_{n=1}^{N} \left[\int Q_n(\mathbf{x}_n) \log \frac{p(\mathbf{x}_n|\mathbf{t}_n, \theta) p(\mathbf{t}_n|\theta)}{Q_n(\mathbf{x}_n)} d\mathbf{x}_n\right] =$$

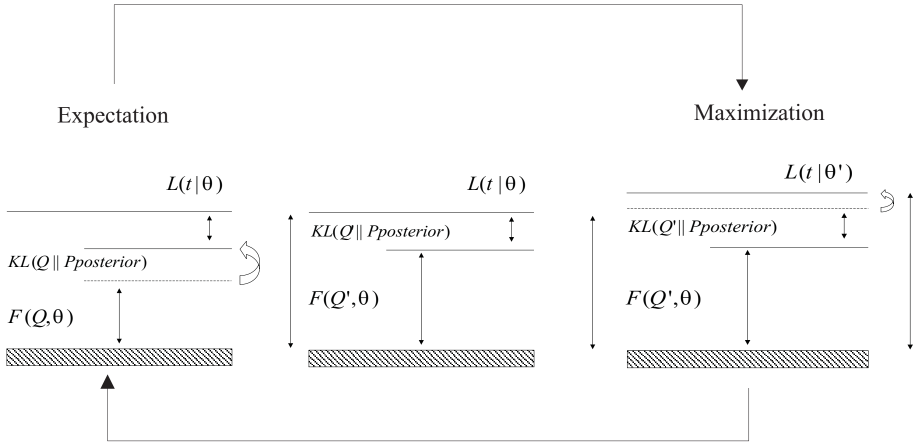


Figure 1.4: At a first step, this variational approach maximizes $\mathcal{F}(Q,\theta)$ as a functional of Q and the parameters θ, which is equivalent to minimizing the Kullback-Leibler divergence of the approximating Q-functions with respect to the a posteriori distribution for the latent variables. After this step, there is a second one that implies maximizing the free energy $\mathcal{F}(Q,\theta)$ with respect to the model's parameters θ. This second step takes the values for the approximating Q-functions as fixed.

$$
\begin{aligned}
&= \sum_{n=1}^{N}\left[\int Q_n(\mathbf{x}_n)\log\frac{p(\mathbf{x}_n|\mathbf{t}_n,\theta)}{Q_n(\mathbf{x}_n)}d\mathbf{x}_n\right]+\sum_{n=1}^{N}\log p(\mathbf{t}_n|\theta) = \\
&= -\sum_{n=1}^{N}KL(Q_n(\mathbf{x}_n)||p(\mathbf{x}_n|\mathbf{t}_n,\theta))+\sum_{n=1}^{N}\log p(\mathbf{t}_n|\theta)
\end{aligned}
$$

Two main question arise when it comes to deal with lower/upper bounds of a function to be optimized is:

- *What are the conditions to be satisfied by the marginal distribution?* The log-likelihood $p(\mathbf{t}_1,\ldots,\mathbf{t}_N|\theta)$ must have at least one finite maximum value and this function must be log-concave.

- *When a lower bound is useful?* Taking into account the first issue, a lower bound is of great utility when it concerns an iterative process of non-decreasing updates.

The non-decreasing updating of a lower bound such as $F(Q,\theta)$ of a log-likelihood function (satisfying the condition above) implies getting closer to the maximum value at each step. Since such a maximum value is finite, there will be a time in the procedure when the free energy $F(Q,\theta)$ reaches that value. Maximizing $F(Q,\theta)$ has to be performed in two steps: i) first, with respect to the approximating Q-functions, and, ii) second, with respect to the model's parameters.

In this case, the Expectation-Maximization consists of two optimization steps: one implies finding the "nearest"[6] Q-distribution to the *a posteriori* probabilities for the latent variables, and another that involves maximizing with respect to the model's parameters (figure 1.4):

Expectation Assume an intermediate stage with θ_k (k-th iteration). Now, the statement "*Compute the sufficient statistics for the latent variables posterior distributions* $p(\mathbf{x}_n|\mathbf{t}_n,\theta)$" can be translated into:

$$Q_n(\mathbf{x}_n) = \mathrm{argmax}_{Q(\mathbf{x})} \left[\mathcal{F}(Q(x),\theta_k)\right] \tag{1.6}$$

that can be found by taking functional derivatives on $\mathcal{F}(Q,\theta)$,

$$\frac{\delta \mathcal{F}}{\delta Q_n(\mathbf{x}_n)} \quad = \quad \frac{\delta}{\delta Q_n(\mathbf{x}_n)} \left\{ \sum_{n'=1}^{N} \left[\int Q_{n'}(\mathbf{x}_{n'}) \log \frac{p(\mathbf{x}_{n'}|\mathbf{t}_{n'},\theta_k)}{Q_{n'}(\mathbf{x}_{n'})} d\mathbf{x}_{n'} \right] \right\} = 0$$

whose solution is:

$$Q_n(\mathbf{x}_n)^{k+1} = p(\mathbf{x}_n|\mathbf{t}_n,\theta_k) \tag{1.7}$$

which is determined by its *sufficient statistics*, i.e., the expected moments.

Maximization Given the new value Q^{k+1} the function to be maximized w.r.t. the model's parameters θ is $\mathcal{F}(Q^{k+1},\theta)$:

$$\theta^{k+1} = \mathrm{argmax}_{\theta} \left[F(Q^{k+1},\theta)\right] \tag{1.8}$$

Thus, after substituting the computed values for Q^{k+1} into $\mathcal{F}(Q^{k+1},\theta)$, the quantity to be maximized is:

$$\theta^{k+1} = \mathrm{argmax}_{\theta} \left[\sum_{n=1}^{N} \int d\mathbf{x}_n p(\mathbf{x}_n|\mathbf{t}_n,\theta^k) p(\mathbf{x}_n,\mathbf{t}_n|\theta) \right] \tag{1.9}$$

Assuming that the log-likelihood $p(\mathbf{t}_1,\ldots,\mathbf{t}_N|\theta)$ has at least one finite maximum value, there is only one remaining question:

$$\mathcal{F}(Q^{k+1},\theta^{k+1}) \geq \mathcal{F}(Q^{k+1},\theta^k) \geq \mathcal{F}(Q^k,\theta^k)?$$

in other words, is $\mathcal{F}(Q^{k+1},\theta)$ monotonically increasing at each step? To answer this question, log-concavity requirement on the log-likelihood $\log p(\mathbf{t}_1,\ldots,\mathbf{t}_N|\theta)$ must be employed (see Appendix A).

1.4 Specific Latent Variable Models

In this section, some of the most well-known latent variable models are described in order to show the relevant features of generative modelling taxonomy (fig. 1.5). Three categories are employed for classification:

[6] In terms of the Kullback-Leibler divergence.

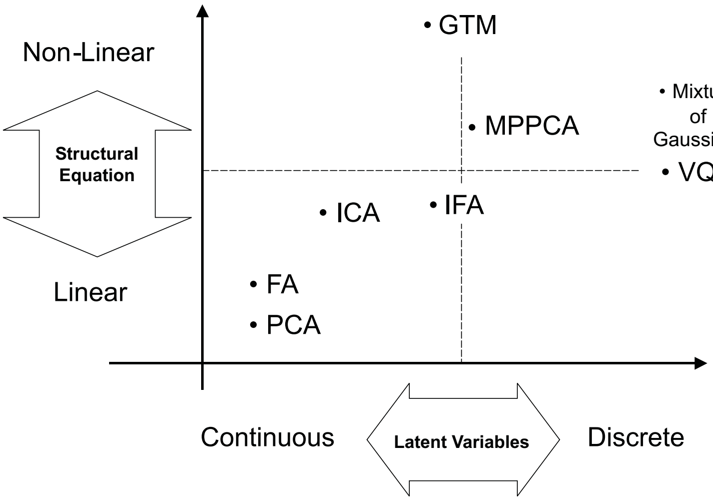


Figure 1.5: Visual classification of the most well-known latent variable models.

- The **noise** model. For instance, a latent variable model is called **normal** when both the prior in latent space and the noise model are normal.
- The structural equation, i.e., mapping. Latent variable models can be classified as **linear** and **nonlinear** according to the corresponding character of the mapping f.
- The latent variables' nature. The model that describes a data set can consist of **continuous**, **discrete** latent variables or a certain combination of both types.

These three points categorize the most utilized models in Pattern Recognition. We specially consider *Principal Component Analysis* (PCA) and a *mixture* version of PCA in order to emphasize the main relevant issues relying on this type of modelling data. Further in this thesis, the rest of models are considered and described taking into account the context of specific applications.

The massive diversification of latent variable models, in the state if the art, arises when considering *continuous latent variables.* These are used for obtaining a **compact** description (and representation) of data. Inside this group, *linear models* are the most exploited[7] ones:

Factor Analysis (FA) Factor analysis (Bartholomew, 1987 [7]; Everitt, 1984) uses a Gaussian distributed prior and noise model, and a linear mapping from data space to latent space. The data space noise model is normal centered at $f(x)$ with diagonal covariance matrix. The log-likelihood has infinite equivalent maxima resulting from orthogonal rotation of the factors.

[7] Computational cost saving and interpretability are two relevant issues that lead to considering this type of modelling.

Independent Factor Analysis (IFA) Independent component analysis (ICA) or blind source separation consists of recovering independent sources, i.e. subspaces that should have minimum statistical dependence among features, given only sensor observations that are unknown linear mixtures of the sources (Comon, 1989 [23]; Cardoso, 1989 [21]; Hyvärinen [42, 44, 43]).

The main difference in this case relies on the distribution considered for the latent variable space $p(\mathbf{x})$, where its *sufficient statistics* goes beyond second order moments. An interesting work that helps understanding this modelling is Independent Factor Analysis (Attias, [4]), which uses a factorial Gaussian mixture prior distribution. In this case, discrete components on the latent variable structure must be introduce in order to model multi-modal behaviors on the latent space. This approach embraces the different concepts previously introduced in the ICA framework, such as super-gaussianity, sub-gaussianity, kurtosis, etc...

Non-Negative Matrix Factorization (NNMF) Based on the psychological and physiological evidence for parts-based representations in the brain, Nonnegative Matrix Factorization (NMF) aims at learning the parts of objects instead of a holistic representation such as those learnt by the previously mentioned methods. This is done by including a nonnegativity constraint that allows only additive, not subtractive, combinations of the features. In this case, the difference with respect to the other linear techniques mainly relies on the selection of a specific noise model in the data space $p(\mathbf{t}|\mathbf{x})$ (for instance a Poisson distribution [60, 59]). In this case, the *lower bound* selected in the optimization algorithm leads to *multiplicative update rules* (Kivinen 1995 [56, 57],) instead of the additive usual ones. This fact enforces variables and parameters to remain non-negative during the optimization process. Further in this thesis, more details are given related to this specific type of non-negative constraints.

Regarding non-linear models, the Generative Topographic Mapping (Bishop 1998 [13]) arose as a potential application of the GTM is visualization of high-dimensional data:

Generative Topographic Mapping (GTM) is a non-linear latent variable model, intended for modelling continuous, intrinsically low-dimensional probability distributions, embedded in high-dimensional spaces. It can be seen as a non-linear form of principal component analysis or factor analysis. It also provides a principled alternative to the self-organizing map a widely established neural network model for unsupervised learning resolving many of its associated theoretical problems. Since the GTM is non-linear, the relationship between data and its visual representation may be far from trivial. There are two principal limitations of the basic GTM model. The computational effort required will grow exponentially with the intrinsic dimensionality of the density model. The other limitation is the inherent structure of the GTM, which makes it most suitable for modelling moderately curved probability distributions of approximately rectangular shape. When the target distribution is very different to that, the aim of maintaining an "interpretable" structure, suitable for visualizing data, may come in conflict with the aim of providing a good density model. Interpretability of high dimensional data is one of the main subjects studied in this thesis, where an algorithm for video clustering and visualization of scene linking is presented.

Two relevant advantages are introduced: i) the algorithm performs online clustering and low dimensional embedding, and ii) it is significantly reliable for high dimensional case.

Principal Component Analysis is studied in depth in the following section, since it has been the most popular dimensionality reduction technique with an extensive range of applications in Computer Vision. Many reasons support its popularity. For instance, from a computational point of view, this technique offers a closed form solution through Singular Value Decomposition (SVD). Moreover, its solutions are straightforward to interpret, since it is a linear technique that can be understood in terms of pure data rotations, and which involves a fewer number of parameters than Factor Analysis or the other presented linear techniques.

1.4.1 Principal Component Analysis (PCA)

Historically, Principal components analysis (PCA) has been seen as "the classical technique" for dimension reduction. PCA has been widely used as a preprocessor for pattern recognition applications to reduce the input dimension before building classifiers.

PCA was first proposed by Hotelling (1933) [41] for dimension reduction. Anderson (1958) [1] used PCA to reduce the number of variables by eliminating linear combinations with small variance. Oja (1983) [71] discussed PCA and related techniques. A closely related orthogonal expansion to PCA is the Karhunen-Loeve (K-L) [54, 65] expansion (Watanabe 1965 [108]) which was originally conceived in the framework of continuous second-order stochastic processes. When restricted to a finite dimensional case and truncated after a few terms, the K-L expansion is equivalent to a PCA expansion.

In the latent variable framework, Principal Component Analysis can be seen as a maximum likelihood *factor analysis* in which the noise model is gaussian and isotropic. This simple fact, already reported in the early factor analysis literature, seems to have gone unnoticed until Tipping and Bishop [103] and Roweis (1998) [84] recently rediscovered it. The approach of considering an isotropic noise model in factor analysis had already been adopted in the Young-Whittle factor analysis model (Young, 1940 [114]; Whittle, 1952 [111]).

According to section 1.3, a latent variable model is specified by the functional forms of:

- the prior model in latent space $p(\mathbf{x})$

- the smooth mapping $f : X \to T$ from latent space to data space, i.e. the causal relation between latent variables and observations.
- the noise model in data space $p(\mathbf{t}|\mathbf{x})$

all of which are equipped with parameters. In this particular case, PCA defines a model for the latent space via a normal distribution:

$$p(\mathbf{x}) = \frac{1}{(2\pi)^{q/2}} e^{-\frac{1}{2}\mathbf{x}^T\mathbf{x}} \tag{1.10}$$

The structural equation (mapping) is defined as a **linear** combination of the latent variables $\mathbf{x}$:

$$\mathbf{t}_n = W\mathbf{x}_n + \mu + \mathbf{e}_n \tag{1.11}$$

where $\mathbf{e}_n$ is a noise vector for the observation $\mathbf{t}_n$ and $\{W, \mu\}$ are the parameters corresponding to this particular mapping. Finally, the noise model for $\mathbf{e}_n$ completely determines this latent variable model. This noise model is gaussian and isotropic[8], i.e.,

$$p(\mathbf{t}_n|\mathbf{x}_n) = \frac{1}{(2\pi\sigma^2)^{d/2}} \exp\left\{-\frac{1}{2\sigma^2}|\mathbf{t}_n - W\mathbf{x}_n + \mu|^2\right\} \tag{1.12}$$

The use of the fundamental equation (1.2), under these two specifications for the latent space and the noise model, yields the following likelihood function:

$$p(\mathbf{t}_n) = \frac{1}{(2\pi)^{d/2}|C|^{1/2}} \exp\left\{-\frac{1}{2}(\mathbf{t}_n - \mu)^T C^{-1}(\mathbf{t}_n - \mu)\right\} \tag{1.13}$$

where $C = \sigma^2 I + WW^T$. At this point, two main relevant and practical issues deserve to be mentioned when considering PCA under a probabilistic approach:

- Given a specific instance for the model parameter's $\{W, \mu, \sigma^2\}$, this likelihood function gives a measure of the novelty of a new data point.
- PCA can easily be extended to a mixture of such models.

Using *Bayes'* theorem, the *posterior distribution* of the latent variables $\mathbf{x}_n$ given the observation $\mathbf{t}_n$ is given by:

$$p(\mathbf{x}_n|\mathbf{t}_n) = \frac{1}{(2\pi)^{q/2}|M|^{1/2}} \exp\left\{-\frac{1}{2}(\mathbf{x}_n - <\mathbf{x}_n>)^T M^{-1}(\mathbf{x}_n - <\mathbf{x}_n>)\right\} \tag{1.14}$$

which is defined by its *sufficient statistics*:

$$\begin{aligned} <\mathbf{x}_n> &= M^{-1}W^T(\mathbf{t}_n - \mu) && (1.15)\\ <\mathbf{x}_n\mathbf{x}_n^T> &= M^{-1} + <\mathbf{x}_n><\mathbf{x}_n^T> && (1.16) \end{aligned}$$

[8] If we consider this model consisting of a diagonal covariance matrix for the noise model $\sigma^2 I \to \Lambda$ instead, then Factor Analysis is recovered.

where $M = (\sigma^2 I + W^T W)$.

The computation of the *sufficient statistics* for PCA yields the first step of the EM algorithm in order to estimate the model parameters from data. Moreover, in this particular model, all the required integrals are easy to obtain, since this model only involves gaussian distributions. Taking into account these facts, the model estimation follows this *two-steps* procedure:

E step According to equation (1.7), this step is computed through the use of *sufficient statistics* in equations (1.15) and (1.16).

M step According to the forms of the prior distribution in equation (1.10) and the noise model in equation (1.12), the parameter estimation (eq. (1.8)) can be given in a closed form solution for all:

$$\begin{aligned}
W &= \left[\sum_{n=1}^{N} (\mathbf{t}_n - \mu) < \mathbf{x}_n >^T\right] \left[\sum_{n=1}^{N} < \mathbf{x}_n \mathbf{x}_n^T >\right]^{-1} \\
\mu &= \frac{1}{N} \sum_{n=1}^{N} \mathbf{t}_n \\
\sigma^2 &= \frac{1}{N} \sum_{n=1}^{N} \left\{ |\mathbf{t}_n - \mu|^2 + tr \left[W^T W < \mathbf{x}_n \mathbf{x}_n^T > \right] - 2(\mathbf{t}_n - \mu)^T W < \mathbf{x}_n > \right\}
\end{aligned}$$

These steps must be iterated until a certain degree of convergence of the log-likelihood (eq. (1.5)). There exists a unique (although possibly degenerate, if some eigenvalues are equal) maximum likelihood estimate closely related to the q principal components of the data. If the sample covariance matrix $S = \frac{1}{N} \sum_{n=1}^{N} (\mathbf{t}_n - \mu)(\mathbf{t}_n - \mu)^T$ is decomposed by Singular Value Decomposition (SVD) as $S = UVU^T$, with $V = \text{diag}(v_1, \ldots, v_d)$ containing the eigenvalues (ordered decreasingly) and $U = (\mathbf{u}_1, \ldots, \mathbf{u}_d)$ the associated eigenvectors, then

$$\begin{aligned}
W &= U_q (V_q - \sigma^2 I)^{1/2} \\
\sigma^2 &= \frac{1}{d-q} \sum_{k=q+1}^{d} v_k
\end{aligned}$$

where $U_q = (\mathbf{u}_1, \ldots, \mathbf{u}_q)$ and $V = \text{diag}(v_1, \ldots, v_q)$. Nevertheless, from a computational point of view, there are situations with very high dimensionality such as rasterized images ($\mathcal{O}(10^5)$). In that case, it is not feasible to store the covariance matrix S, and, its SVD can be quite difficult to compute since $\mathcal{O}(d^2)$ matrices are involved in the computation. On the other hand, not all the principal directions U_d are necessary. In fact, only the q-first are needed. Therefore, when dealing with very high dimensional spaces, the use of the EM algorithm may be more practical. Moreover, it only requires inversions of $\mathcal{O}(q^2)$ matrices.

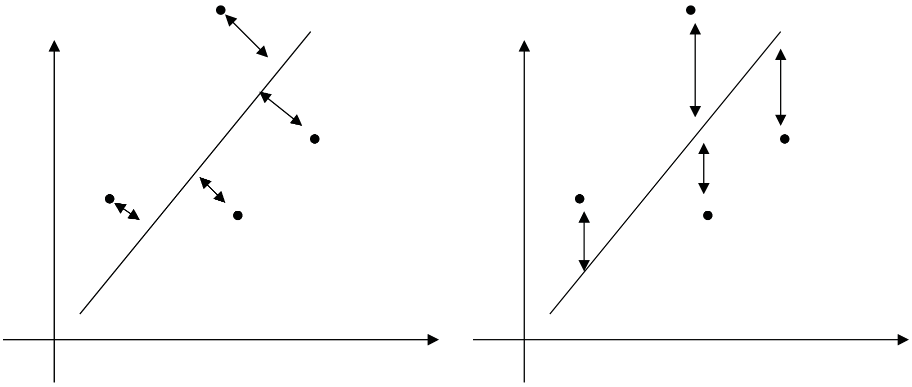


Figure 1.6: Left: Perpendicular noise model. Right: Biased noise model.

Geometrical Interpretation of PCA

One of the main issues that comes along with Generative Models is *interpretability.* Given that most of problems in Computer Vision deal with very high dimensions, solutions are occasionally hard to interpret. In this section, a couple of examples are introduced in order to visualize the resulting eigenvectors of a PCA solution. To this end, both examples are based on low dimensional data:

Fitting 1D Lines Line fitting is extremely useful. In many applications, objects are characterized by the presence of straight lines (e.g. many industrial parts have straight edges of one form or another). In this case, we consider data points of dimension $d = 2$. Since we assume that no coordinate is privileged, noise must have the effect on both x and y coordinates. Otherwise, the model would be biased as in figure 1.6 (right side). We model data points as being generated by an abstract point along the line to which is added a vector perpendicular to the line, with a length given by a zero mean, Gaussian random variable. This means that the distance from data points to the line has a normal distribution. By setting this up as a maximum likelihood problem, we obtain a fitting criterion that chooses a line that minimizes the sum of distances between data points and the line. Such a line is defined by a unit vector $\mathbf{u}$ indicating the line direction, and a point μ that constrains the line to a specific position in the data space. The latent space has dimension $q = 1$ and it is defined by the line. Dimensionality reduction is performed, since all the points can be expressed as a result of multiplying the direction vector $\mathbf{u}$ by a factor x (coordinate in the latent space) plus a point μ:

$$\mathbf{t}_n = \mathbf{u}x_n + \mu + \mathbf{e}_n \tag{1.17}$$

with $t_n \in \mathbb{R}^2$ and $x_n \in \mathbb{R}^1$. The latent space coordinates can be obtained through the orthogonal projection of each data point $\mathbf{t}_n$ onto the line:

$$x_n = \mathbf{u}^T(\mathbf{t}_n - \mu) \tag{1.18}$$

Given these projections and the generative structural equation (1.17), a reconstructed vector version can be obtained for each data point $\mathbf{t}_n$:

$$\hat{\mathbf{t}}_n = \mathbf{u}\mathbf{u}^T(\mathbf{t}_n - \mu) + \mu \tag{1.19}$$

Literature refers to this process as a filtering process. In fact the reconstructed vector $\hat{\mathbf{t}}_n$ lies on the line that generated the observations, since noise has been removed. Thus, a concluding remark is that **PCA can be used for filtering data in order to remove identical independent distributed Gaussian noise**. In fact, this approach corresponds to the original perspective given to PCA. The estimation of the line parameters $\{\mathbf{u}, \mu\}$ can be either performed through the previously introduced EM algorithm or by means of minimizing the reconstruction error (i.e. minimizing the sum of perpendicular distances between data points and the line)[9]:

$$\begin{aligned}
\mathcal{E} &= \sum_{n=1}^{N} |\hat{\mathbf{t}}_n - \mathbf{t}_n|^2 = \\
&= \sum_{n=1}^{N} |\mathbf{u}\mathbf{u}^T(\mathbf{t}_n - \mu) + \mu - \mathbf{t}_n|^2 = \\
&= \sum_{n=1}^{N} \left\{ |\mathbf{t}_n - \mu|^2 - (\mathbf{t}_n - \mu)^T \mathbf{u}\mathbf{u}^T (\mathbf{t}_n - \mu) \right\} = \\
&= \sum_{n=1}^{N} |\mathbf{t}_n - \mu|^2 - \mathbf{u}^T \sum_{n=1}^{N} \left\{ (\mathbf{t}_n - \mu)(\mathbf{t}_n - \mu)^T \right\} \mathbf{u}
\end{aligned}$$

In order to enforce orthonormality constraints, a Lagrange multiplier over $\mathbf{u}$ is employed:

$$\mathcal{E} + \lambda(1 - \mathbf{u}^T\mathbf{u}) \tag{1.20}$$

From a Least Square Estimation (LSE) approach, the parameters of this model can be obtained by taking derivatives w.r.t. $\{\mathbf{u}, \mu\}$ and equating them to zero:

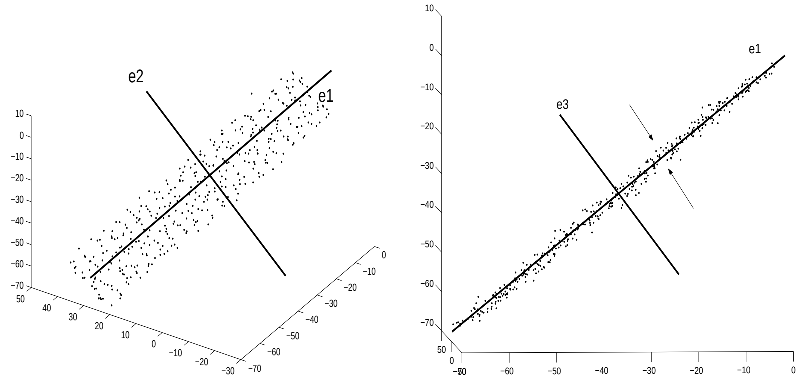
$$\mu = \frac{1}{N} \sum_{n=1}^{N} \mathbf{t}_n$$

and

$$\sum_{n=1}^{N} \left\{ (\mathbf{t}_n - \mu)(\mathbf{t}_n - \mu)^T \right\} \mathbf{u} = \lambda \mathbf{u}$$

which turns out into an eigenvalue problem; the eigenvector with the largest associated eigenvalue minimizes the reconstruction error. Therefore, in this particular example, the different ingredients introduced for building a latent variable model can be interpreted by means of a geometrical perspective:

[9] Both techniques lead to the same results.



(a) (b)

Figure 1.7: Two different views of a planar surface and its principal axes: View (a) shows the two principal axes of bigger variance $e1$ and $e2$, and, view (b) shows the variance with respect to the lowest variance axis $e3$.

- the **latent space** is defined by the direction vector $\mathbf{u}$,
- the **latent coordinates** $\mathbf{x}_n$ through the projections onto the line, and,
- the **reconstruction error** as the perpendicular distance to the line.

Note that these are the three points mentioned in section 1.3 and flavored by the LSE perspective.

Fitting 2D Planes The approximation of a surface in terms of planar patches is use further in this thesis for extracting useful geometrical information from data (breaking curves, 3D contours for reconstructing of an object through assembling broken parts, etc). The first and easiest sort of surface to start with is a planar surface described by a few number of points. This type of surfaces can be represented by a specific set of coordinate orthogonal axes adapted to the points spatial distribution. These are obtained by means of a linear regression that fits a $2D$ plane minimizing the orthogonal distance to the mentioned plane. A plane, actually, can be described by just two degrees of freedom that locate any point belonging to it. These two degrees of freedom are scalar values that measure the distance of a point along each of the axes.

The estimation of the principal axes is performing through Principal Component Analysis as well. The result of applying PCA on set of $3D$ points is a set of 3 unitary vectors and 3 scalar values. The unitary vectors are known as *eigenvectors*, and the scalar values as *eigenvalues*. The eigenvalues give a notion of the importance of a specific axis with respect to the others. This importance measure is actually the variance -in statistical terms- of the data along each axis (see fig.3.2). This means that an axis with large variance associated has the data distributed in a larger portion of space, than another axis with lower variance associated. In other words, when it comes to fit a plane to a planar distribution of points, there is an axis with negligible variance (fig.3.2(b)), which determines the noise of the planar distribution. The larger is the amount of variance in that direction, the lower is the likelihood of the point distribution to be a plane. PCA algorithm is coordinate free, which means that results are adapted to the nature of the data distribution but to the frame of reference where data is represented. This makes the technique independent of the absolute position, which is quite useful when dealing with pottery (pieces in general) scans.

1.4.2 Mixtures of Linear Models

Linear models are quite useful since their simplicity, low computational cost and interpretability (from a geometrical point of view). However, there are situations where the distribution of data can be more complex than a linear model can cope with. For instance, a $2D$ surface with regions of high curvature requires a more complex model in order to capture such a behavior. The same way a $1D$ sinusoidal signal can not be well explained through one single line. In these situations, fitting data through a linear model will produce misleading results. Regarding this latter $2D$ *plane fitting example*, there are two important issues to be considered: on one hand, the *noise* in data originated from observations (scanning, etc...) and, on the other, the *error* due to model assumptions. In the framework we are dealing with, the error due to the model comes from fitting a plane to a region, and regions with high curvature yield a high error measure in the estimates, i.e, there is a trade off between error in the estimates and model complexity. In this particular framework, the distinction must be performed between **noise** and **geometrical features** such as curvature. Nevertheless, in a more general framework, complexity can also arise when data is distributed in many clusters, which is due to considering different *classes* of observations. One single *projection* can again lead to confusing explanations of such behaviors, i.e. mixing many different clusters in one class. This is a typical problem when facing **classification**.

The more ambitious is the task we aim to perform, the more complex the hypotheses to be formulated on the model. Highly complex models are much more difficult to estimate, which means that inferring the latent variables, that describe the variabilities in the observations, may not be a feasible task.

In these cases, it is advisable applying the *finite mixtures* approach, or what is the same, focussing on small subproblems which can be studied with simpler techniques. In most of cases, simplicity goes tied to linearity. Even though, the introduction of non-linearity can offer the possibility of modelling a much richer family of structures, non-linearity also brings potential problems. Since real data is typically corrupted by noise, there is a risk that a non-linear model captures not only systematic non-linearities in a data set, but also random artifacts due to noise. There is also a problem related to interpreting the results obtained through non-linear models. Much simpler and easier to understand for posterior evaluations are linear models.

In order to cope with **global nonlinear** behaviors we study the advantages of dealing with local combinations of linear sub-models. Moreover, we take into account that many problems offer the possibility of representing data through vectors. Typically, these vectors lie on a manifold, which can be either linear or non-linear. The underlying idea is based on geometry; a non-linear manifold can be approximated through linear patches. Essentially, the work presented in this thesis is based on analyzing Computer Vision problems where a combination of linear sub-models is applied.

The advantage of finite mixtures of latent variable models is that they can place different latent variables models in different regions of data space, where each latent variable model models locally the data. This allows the use of simple **local models** (e.g. linear-normal, like factor analysis or principal component analysis) that build a complex **global model** (piecewise linear-normal). In other words, finite mixtures of latent variable models combine **clustering** with **dimensionality reduction**.

Finite mixtures of latent variable models can be constructed in the usual way as:

$$p(\mathbf{t}) = \sum_{m=1}^{M} p(m)p(\mathbf{t}|m) \tag{1.21}$$

where:

- A set of local latent subspaces $\mathcal{X}_m$ of dimension q_m (not necessarily equal) are defined in terms of a set of: mappings $f_m : \mathcal{X}_m \to \mathcal{T}$, noise models $p(\mathbf{t}|\mathbf{x}, m)$, prior distributions $p(\mathbf{x}|m)$ and a likelihood distribution $p(\mathbf{t}|m)$ given by each fundamental equation:

$$p(t|m) = \int_{\mathcal{X}_m} p(\mathbf{x}|m)p(\mathbf{t}|\mathbf{x}, m)d\mathbf{x} \tag{1.22}$$

- There is a set of mixing coefficient $p(m)$ that explain the relevance of each sub-model among the rest inside the model.

This finite mixture model falls into the class of latent variable models taking both **discrete** and **continuous** natures of latent variables (see fig. 1.5). In fact, the likelihood function $p(\mathbf{t})$ for the observations is obtained by integrating (i.e. summating) over the latent variables (discrete and continuous) as before:

$$p(\mathbf{t}) = \sum_{m=1}^{M} \int_{\mathcal{X}_m} p(\mathbf{t}, \mathbf{x}, m) = \sum_{m=1}^{M} \int_{\mathcal{X}_m} p(m)p(\mathbf{x}|m)p(\mathbf{t}|\mathbf{x}, m) \tag{1.23}$$

The parameter estimation can be performed through the variational approach introduced for the EM algorithm. No additional problems arise in this particular case; we just have to take into account that we are dealing with discrete and continuous variables. Taking into account the *missing information* framework presented in section 1.3, there are now two issues to be considered: not only the latent coordinates $\mathbf{x}_n$ have to be *inferred* for a particular data point $\mathbf{t}_n$, but also the index m of the mixture component that generated it:

E step This requires the computation of the posterior probabilities $p(m|\mathbf{t}_n)$ and $p(\mathbf{x}_n|m, \mathbf{t}_n)$:

$$p(m|\mathbf{t}_n) \quad = \quad \frac{p(m)p(\mathbf{t}_n|m)}{\sum_{m'=1}^{M} p(m')p(\mathbf{t}_n|m')}$$

Regarding the inference of $p(\mathbf{x}_n|m, \mathbf{t}_n)$, we proceed as in section 1.3 computing the *sufficient statistics* for each sub-model.

M step This results in several update equations for the parameters. The update equations for the mixing proportions are independent of the type of latent variable model used:

$$p(m)^{t+1} \quad = \quad \frac{1}{N}\sum_{n=1}^{N} p(m|\mathbf{t}_n)^t p(m)^t$$

The equations for the rest of the parameters (from the individual latent variable models) depend on the specific functional form of $p(\mathbf{t}_n|\mathbf{x}_n, m)$ and $p(\mathbf{x}_n|m)$, but often they are averages of the usual statistics weighted by $p(m|\mathbf{t}_n)$ and computed in a specific order.

Some examples of mixtures of linear models can be found in the literature: Ghahramani [85] and Hinton (1997) [39] construct a mixture of Factor Analyzers, later Tipping and Bishop (1999) [103] define mixtures of Principal Component Analyzers (a particular form of mixtures of factor analyzers). In general, mixtures of latent variable models whose distribution in data space $p(\mathbf{t})$ results in a Gaussian mixture (such as mixtures of factor analyzers or PCAs) have two advantages over usual mixtures of Gaussian distributions:

- Each component latent variable model locally models both the (linear) mapping and the noise, rather than just the covariance.
- They use fewer parameters per component, e.g. $d(q+1)$ for a factor analyzer versus $d(d+1)/2$ for a Gaussian (of course, q should not be too small for the model to remain good).

A mixture of diagonal Gaussians and a mixture of spherical Gaussians can be seen, as limit cases, as a mixture of factor analyzers with zero factors per component model and a mixture of principal component analyzers with zero principal components per component model, respectively. Thus, Gaussian mixtures explain the data by assuming that it its exclusively due to noise-without any underlying (linear) structure.

1.5 Discussion

The methodology presented in this thesis has the purpose of finding information by learning a flexible model from observations. This sort of algorithms may not explain the physical underlying phenomena that produce the observed data, but they have the advantage of simulating the observed behavior without *knowledge of the domain*. Since many variables and complex relations among them can affect observations, the construction and estimation of a physically-derived model can be impractical tasks. Enormous difficulties materialize when the purpose consists in distinguishing categories of events that are irrelevant to the observations of interest. The distance, in terms of suitability, between **generative models** and **physical models** notably increases when dealing with high dimensional data. In fact, high-dimensionality arises for several reasons, such as stochastic variations and the measurement process.

Many fields in science encounter such problems. Particularly in Computer Vision, the collection of features that can be obtained from images usually suffers from noise and, in most cases, from high-dimensionality. Nevertheless, interesting results on recognition and classification can be obtained without the necessity of explaining observations through the Image Formation process. Of course, this decision is always governed by the specific purpose of each problem. It is not surprising therefore that Latent Variable models have been extensively applied in diverse fields such as psychometrics[10] and "natural sciences" (e.g. in botany, biology, geology or engineering). Those different science areas have in common that explaining their observed phenomena through the roots of physics (quantum phenomena, particle physics, etc) is neither practical nor suitable.

This type of problems can be faced through Generative Models, since they offer: i) dimensionality reduction, and ii) stochastic noise modelling. In addition to this, it is worth noting that the probabilistic approach given to the formulation yields not only a manner of modelling noise, but also a way of including *knowledge of the domain* in terms of a priori information (prior distributions). In this sense, practical methods to explain observations can be enriched through additional information that we may possess from external sources of knowledge (for instance, shape and time continuity, boundaries, etc).

Two main points of discussion have an essential presence in this thesis: i) the combination of **global** and **local** information, and ii) the complexity of a model in terms of **linearity** and **non-linearity**.

The way information can be combined mainly relies on the feature selection process, i.e., the collection of variables that form the observed space. Infinite possibilities arise from this particular point. This fact mostly determines the optimality and reliability of the set of potential solutions for a given problem, and, it is often a matter of fine cuisine, where creativity plays a fundamental role. For instance, information can be extracted from a set of measurements on the pixels of an image (pixel based operations), from global transformations to the raw image data, from considering many

[10] Historically, the idea of latent variables arose primarily from psychometrics, beginning with the g factor of Spearman (1904) [96] and continuing with other psychologists such as Thomson , Thurstone and Burt , who were investigating the mental ability of children as suggested by the correlation and covariance matrices from cognitive tests variables. This eventually led to the development of Factor Analysis. Bartholomew (1987) [7] gives more historical details and references. In fact, Factor Analysis has been applied to a number of "natural science" problems as well as other kinds of latent variable models recently developed, such as the application of GTM or of ICA.

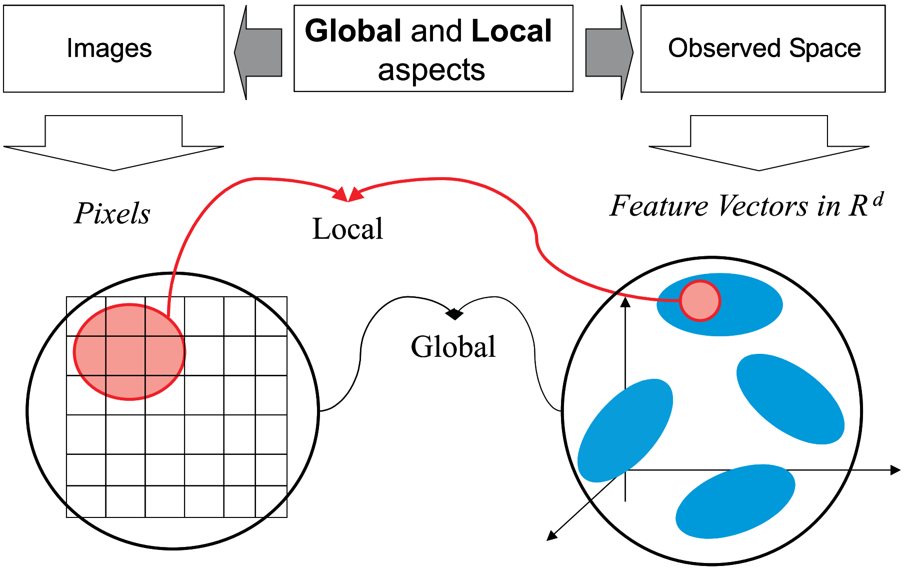


Figure 1.8: Global and local connotations.

images at the same time, filter responses, geometric measurements, etc.

Global and local concepts correspond to space and time in our field. Regarding to the spatial distribution of information, two essential aspects must be considered: global and local connotations can be related either to the vicinity of pixels in an image or to the proximity of feature vectors in the observed space (see fig. 1.8). This thesis shows problems that possess one or both aspects of locality when it comes to extract information from raw data. An interesting situation, where both conceptions blend into a single one, is 3D range data, where pixel locations are 3D coordinates[11] and feature vectors in the observed space can be the same 3D coordinates as well. In the following chapter, a deeper analysis of this phenomenon is introduced as an example of *soft*-dimensionality reduction. In fact, 3D geometry is particularly useful in order to understand and visualize some features of the introduced density distribution-based techniques.

The complexity of a model comes from the particular specification of each of the four ingredients needed for building a certain Generative Model: the latent variables, the model parameters, the structural equation, and the noise model. Regarding to these components, a taxonomy has been presented in previous section 1.4. From an algorithmic point of view, complexity in this case translates into the linear and non-linear concepts, which often are related to the estimation process. It has been emphasized that simplicity, interpretability, and a low computational cost are always desirable. In this sense, linear models are more likely to hit these targets. Nonetheless, there are situations where a linear model cannot cope with. The origin of such

[11] Pixel values are binary: 1 if there is a point in a cell of the 3D space, 0 otherwise.

situations can be: a specific model for the latent variables that is far from being Gaussian (IFA, ICA), a non-linear structural equation (GTM, VQ, Mixture Models), or a non-normal noise model (ICA, IFA, robust statistics). Inside, this group of non-linear behaviors, we mainly focus on Mixture Models, since they have a very attractive property: they are complex non-linear models built from simple linear models. They in fact apply the old rule of divide-and-conquer, more explicit or less. Further, in this thesis, we present some different ways of implementing this rule.

Another issue that deserves to be pointed out is that a probabilistic approach is given to the formulation of the models presented in this thesis. This approach permits: i) the combination of several probabilistic methods in a **mixture** in a natural way, ii) comparing a method with other probabilistic methods as well as constructing **statistical tests**, iii) the **prediction** of any variable(s) as a function of any other variable(s) by using conditional probabilities, and iv) the natural extension to Bayesian analysis for **model comparison** through the use of prior distributions as well as the inclusion of **external sources** of knowledge.

There is a strong relation between Least Square Estimation techniques and probabilistic methods (Maximum Likelihood-based). Historically, the first kind used to precede the second one. In some way, most problems are tackled first through the idea of minimizing some reconstruction error, however subsequently such techniques must be recast in a probabilistic formulation if some of the properties mentioned before are desired. For instance, Principal Component Analysis was traditionally not considered a latent variable model. It was first thought of by Pearson (1901)[75] and developed as a multivariate technique by Hotelling (1933)[41]. The most popular technique for dimensionality reduction, PCA, has recently been recast in the form of a particular kind of factor analysis thus as a latent variable model [14, 12, 85].

Selecting relevant information and building a model to explain the observations are issues that coincide with the purpose of finding a powerful representation for each specific problem. This is, in fact, the role of the model parameters. The choice of an appropriate representation for data takes significant relevance when it comes to dealing with symmetries and the modes of variation of a pattern. In this framework, representation consist in a set of basic primitives (or perceptual units, in a semantic approach) that transform a complex problem based on observations into a manageable one. In addition to this, this pursuit usually implies that the number of degrees of freedom in the data distribution is lower than the coordinates used to represent it (**dimensionality reduction**). These simplifications of the estimation problem rely on a proper mechanism of combining such primitives in order to give an optimal description for the observations (**the structural equation**).

1.6 Outline of this Thesis

This thesis begins with an introduction to Latent Variables, the Expectation-Maximization algorithm and some well-known specific latent variable models.

After this introduction, the thesis is divided in two parts. A first part is dedicated to Dimensionality reduction, where two examples are introduced in order to show the connections among latent space, internal symmetries and intrinsic degrees of freedom. Both examples illustrate the combination of global and local information with real data. A second chapter in this part shows the advantages of using combinations of linear models for describing non-linear behaviors of the observed 3D range data.

The second part of the thesis focuses on problems related to temporal sequences of images. The meaning of the global and local information combination is extended to time. Two chapters exploit this idea in order to build mosaic images from video sequences. A third chapter, in this second part, attempts to model the internal temporal symmetry of a video sequence with the purpose of extracting a certain number of summarizing iconic representative image-like structures. In a similar methodology, the first part of this thesis already introduced the connection between the structural equation defining the relationship between latent variables and observations and the continuous one-parameter group. The attempt of extracting semantic units that summarize a video sequence is also treated from an online classification approach. This fourth chapter of the second part classifies and automatically determines the number of classes and their corresponding representatives according to a Bayesian cross-validation criterion.

Finally, the last chapter of the second part, analyzes periodic motions in video sequences. The algorithm classifies the different types of periodic motions according to their different frequencies. An analytic study is performed in order to discern when classification is possible.

This thesis finishes with a summary and concluding remarks in order to point out how the same protocols, when representing observations and dealing with different levels of information, can be applied to apparently different phenomena within the Computer Vision framework.

Part I

Dimensionality Reduction

Chapter 2

Dimensionality Reduction

2.1 Introduction

Latent Variable models introduce a generative model for dimensionality reduction. The aim of this chapter is to connect the presented statistical techniques with the problems of pattern detection and recognition in Computer Vision.

It is a common fact in the Pattern Recognition community that the number of variables used to describe observations is rather larger than desired, (for instance, when it comes to classification, since the curse of dimensionality yields nonintuitive effects on results). This phenomenon can be attributed to the use of general-purpose sensor devices, which consist in transforming a certain form of energy (e.g. light, heat, sound, etc) into stimuli. However, the representation of stimuli, in terms of some perceptually meaningful features, is purpose-driven. Frequently, processing the raw sensory input in order to obtain a purpose-driven perceptual categorization of stimuli implies dealing with high dimensionality (see figure 2.1).

Consider a system with the specific purpose of recognizing a particular object category. Our input signal can be a particular spatial distribution of pixel values (gray/color). Either performing measurements (geometrical features) on the obtained image, or, taking the image as a vector (by row concatenation of the pixel grid), the result is a feature vector that represents an observation. If we proceed through taking an image (of n width and m height) as a vector in lexicographic order (figure 2.2), the feature space will be of nm dimensions, which can be significantly high, e.g. $\mathcal{O}(10^4)$. This means that a nearly infinite number of different images can be represented through this system of coordinates, and, on the other hand, just a small portion of the space is related to a certain object category.

In fact, images of natural scenes are characterized by a high degree of statistical regularity owing to the morphological consistency of the objects. In this sense, such a pixel-based representation is highly redundant, which means that there is a correlation among pixel values. Given a set of different observations from objects concerning the

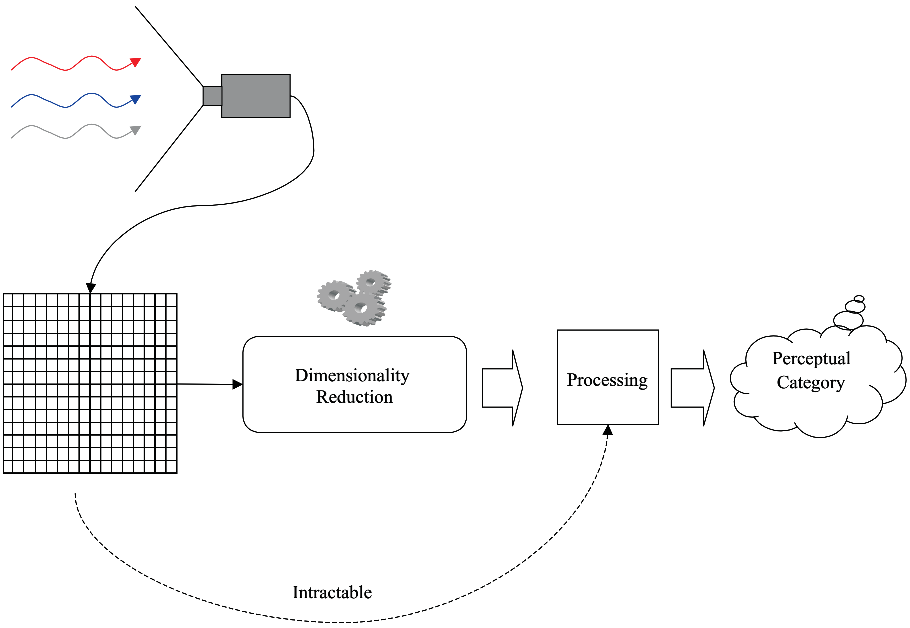


Figure 2.1: The dimensionality reduction problem. A camera captures an image, which is transformed into a collection of pixel values distributed on a grid. The dimension of this problem must be reduced before processing in order to represent a perceptual category. Otherwise, this sort of problems are often intractable.

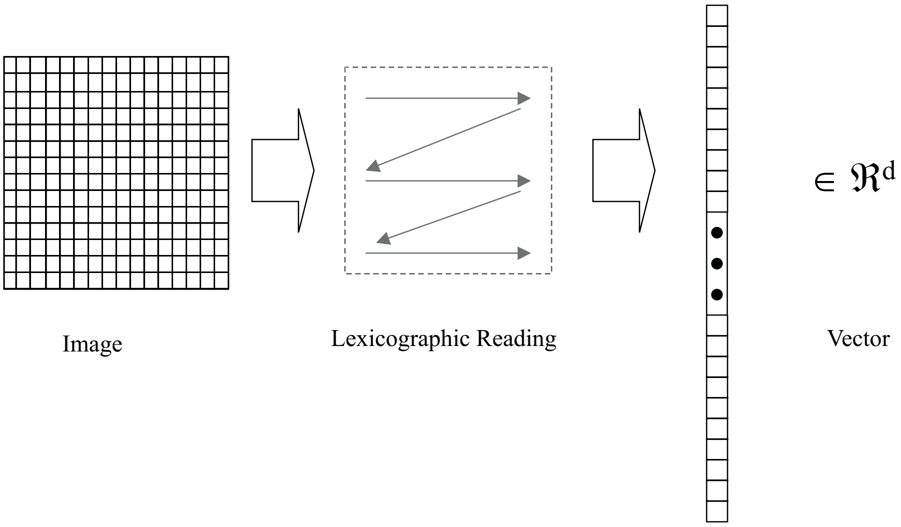


Figure 2.2: Vector concatenation through lexicographic ordering of an image.

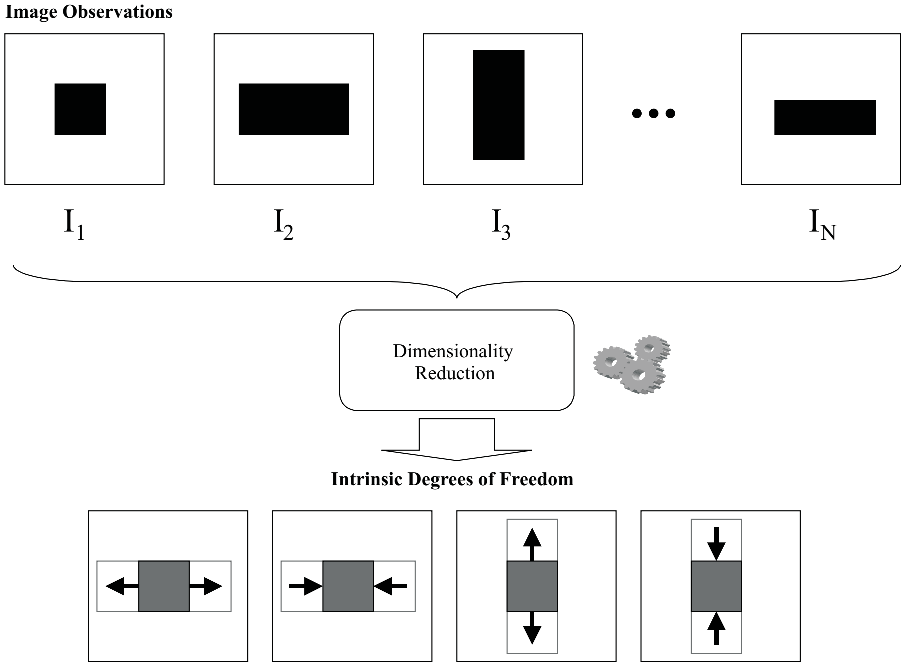


Figure 2.3: Example of *hard dimensionality* reduction problems. Intrinsic degrees of freedom of a set of image observations. Variations are produced in vertical and horizontal directions only. Therefore this set of high-dimensional observations can be represented through few components that describe the modes of variations present in data.

same category, the variability in each pixel value is constrained to a certain range, which is influenced by the rest of pixel value variations.

Therefore, in a number of occasions it can be useful or even necessary to first reduce the dimensionality of the data to a manageable size, keeping as much of the original information as possible, and then feed the reduced-dimension (intrinsic dimensionality) data into the system (see figure 2.1).

More generally, whenever the intrinsic dimensionality of a data set is smaller than the actual one, dimensionality reduction can bring an improved understanding of the data apart from a computational advantage. Dimensionality reduction can also be seen as a feature extraction or coding procedure, or in general as a representation in a different coordinate system.

Dimension reduction can be categorized in three purpose-guided classes:

- *Hard dimensionality reduction problems*, in which the data have dimensionality ranging from hundreds to perhaps hundreds of thousands of components, and usually a drastic reduction (possibly of orders of magnitude) is sought. The

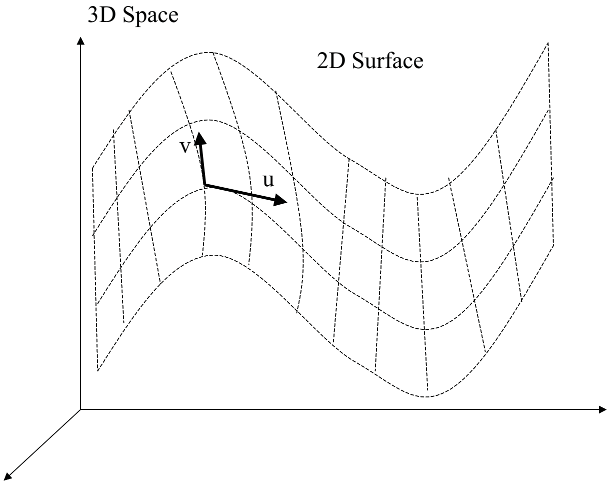


Figure 2.4: Example of soft dimensionality reduction problem. A 2D surface embedded into a 3D space that has 2 degrees of freedom: u and v vectors.

components are often repeated measures of a certain magnitude in different points of space or in different instants of time (see figure 2.3).

- *Soft dimensionality reduction problems*, in which the data is not too high-dimensional (less than a few tens of components), and the reduction not very drastic. Typically, the components are observed or measured values of different variables, which have a straightforward interpretation (geometrical features). In addition, there are problems involving 3D range data that imply a lower intrinsic dimension (2D surfaces or 1D curves, see figure 2.4 and 2.6). In this chapter, we show how a transformation of a 3D range data problem into a lower dimensional representation brings a way of denoising and reconstructing surfaces from missing data.

- *Visualization problems*, in this thesis, we consider a third kind of problems, which involve either soft or hard dimension reduction. The specific purpose of visualizing data has a particular effect (topological and topographical constraints) on the construction of models for reducing the number dimensions. In this thesis, we describe a method for visualizing/summarizing relations among shots in video sequences (hyper-linking) (see figure 2.5).

2.2 Intrinsic Degrees of Freedom

Determining the intrinsic dimensionality of a process given a sample of it is central to the problem of dimensionality reduction. Two main categories of problems implying

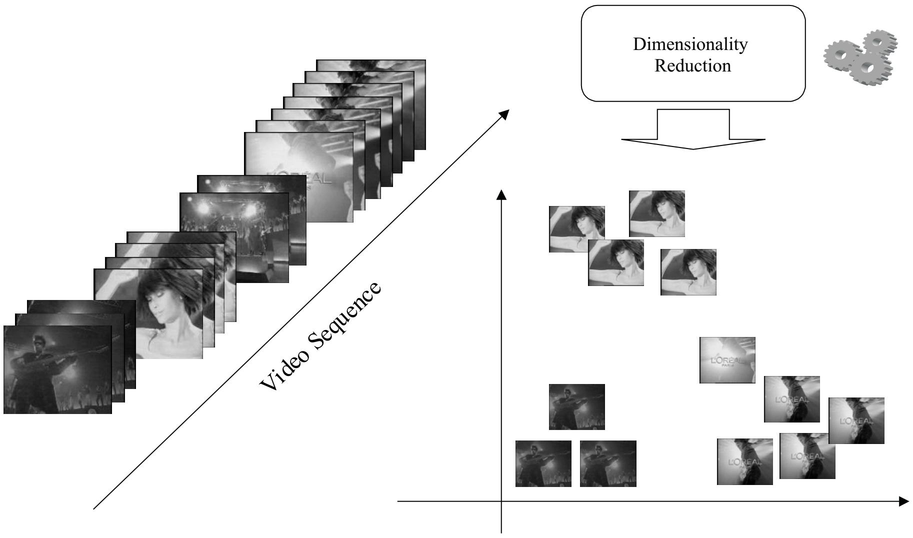


Figure 2.5: Visualization purposes of dimensionality reduction. A video sequence is visualized in a 2D space in order to provide a system for video navigation.

dimension reduction can be described in terms of the prior knowledge of the internal degrees of freedom:

- A priori information on symmetries regarding the distribution of data brings a powerful assistance to build a model for explaining and predicting behaviors. Consider a 3D data distribution generated from scanning an axially symmetric surface. In this case, an axis of symmetry and a profile curve are sufficient for representing such a surface. Moreover, the dimension reduction has gone from 3D to 1D (profile curve) thanks to the introduction of symmetry assumptions (see figure 2.6). Including this kind of external knowledge to a model helps reconstructing the surface even when considering only part of the observations. Another example is a spherical distribution of points, where the surface can be described just by one scalar value (radius). The variations with respect to the generating model presented in this type of problems are taken as noise.
- In other situations, where no prior knowledge on the intrinsic dimension is given, the possibility of under/overfitting must be taken into account. This fact has relative importance depending on the problem goals. For instance, unsupervised learning of clustering models for classification requires a careful analysis in order to provide a meaningful interpretation of the different estimated classes. In these situations, Bayesian techniques are employed in order to automatically determine the number of necessary mixture components.

In this chapter, we show a couple of examples: one is related with object detection in images, and the other corresponds to 3D data reconstruction. Both problems

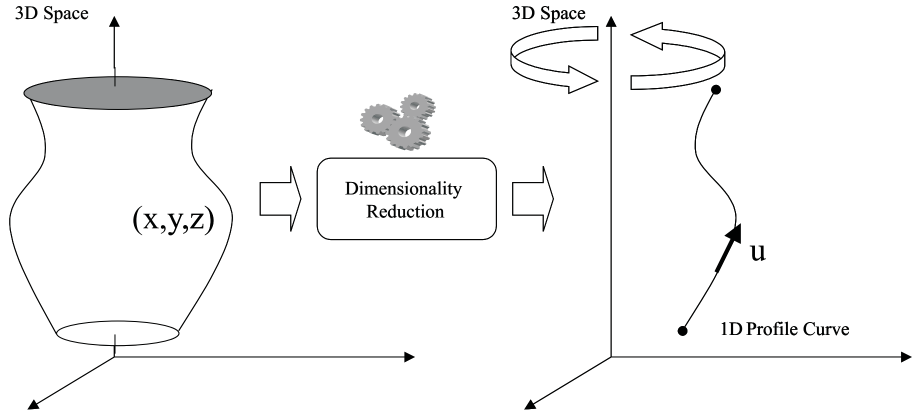


Figure 2.6: An example of **prior external knowledge** on symmetries regarding the distribution of data. Intrinsic degrees of freedom of an axially symmetric surface embedded in a 3D space. An one-dimensional profile curve describes the surface through the polar evolution on one axis of revolution.

pretend to show the advantages of dealing with generative models. Before presenting these two problems, we first analyze two interesting situations that are useful for introducing these examples. On one hand, the *curse of dimensionality* concerning recognition is studied in order to point out the advantages of modelling data though Probabilistic PCA instead of using the Euclidean distance. On the other, some relevant reasons that justify local dimensionality reduction are detailed.

2.2.1 PCA versus Euclidean Distance

In this section, a problem related with the *curse of dimensionality* is illustrated in order to compare the consequences of modelling data through a spherical Gaussian distribution and Probabilistic Principal Component Analysis.

The *curse of dimensionality* refers to the problems associated with multivariate data analysis as the dimensionality increases. This sort problem mainly arises in pattern classification[1], where more than one distance measure must be compared in order to identify to which class belongs a test pattern. In this area, interesting analyzes on the geometry of high dimensional spaces have been performed by Bellman (1961)[8], Silverman (1986)[94], Wegman 1990 [109] and Scott (1992)[90]. These studies are related to the *properties* that are involved in the estimation of density functions, such as the exponential growth of hyper-volumes as a function of dimensionality.

In this thesis, the problems we deal with are related to recognition, which can be seen as a one-class classification problem. A substantial difference with respect to

[1] In practice, the curse of dimensionality means that, for a given sample size, there is a maximum number of features above which the performance of our classifier will degrade rather than improve.

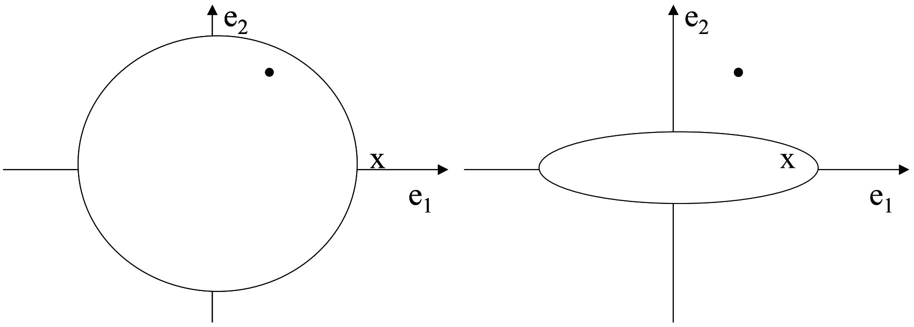


Figure 2.7: Euclidean distance versus log-likelihood distance analysis.

many-classes classification problems is that no relative distances are to be compared. However, certain relevant effects appear in high dimensional spaces depending on the way data is modelled.

A common property of recognition systems is that they require of a decision mechanism to discern if a test pattern belongs to the learned model for a set of training observations. Metric approaches base this decision on a distance measure. Intuitively, the idea of distance quantifies how far is a test pattern from a learned model. In fact, for several reasons[2], the most well-known and exploited measure is the Euclidean distance. One of the easiest models for recognizing one-class patterns is based on the Euclidean distance from a data point to the sample mean, which comes from minimizing the sum of squared distances of each sample to it (Least Squares Estimation approach).

Inside the latent variable framework, this type of models can be considered as a Principal Component Analyzers with zero principal components. In other words, data is modelled through assuming one spherical Gaussian distribution centered in the sample mean. In this sense, we say that no dimension reduction has been performed.

The aim of this section is to show that such a simple model may incur in false-positives/negatives during a test process, since the distribution of data is hardly taken into account. Consider the following 2D problem, where data has 1D intrinsic dimension. By using PCA, we can estimate the principal direction where data lies on, as well as, the direction of noise. In this problem, spherical Gaussian modelling considers all directions to be equally important. In this sense when it comes to performing a test process, some samples will be considered to belong to the learned model. On the other hand, if we consider a model that discerns noise directions and intrinsic dimensions, the number of false-positives/negatives will be reduced. This example is illustrated in figure 2.7, where a test point (black dot) is considered to belong to the learned model when assuming a spherical Gaussian distribution with

[2] There is a natural predisposition to explain phenomena through the intuition acquired in our 3D Euclidean world.

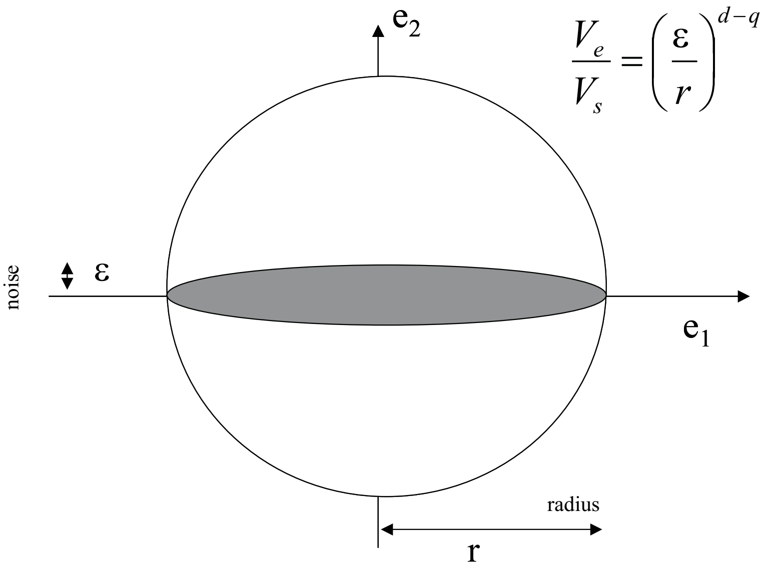

e$_2$
$\frac{V_e}{V_s} = \left(\frac{\varepsilon}{r}\right)^{d-q}$
noise
ε
e$_1$
radius
r

Figure 2.8: Analysis of the curse of dimensionality concerning recognition. The use of an appropriate distance measure based on dimensionality reduction defines a volume smaller than the one generated through an Euclidean distance assumption. When increasing dimensions this facts becomes more relevant since the volume ratio tends to zero.

no principal components, while, it does not belong to the model that assumes just one intrinsic degree of freedom. On the other hand, a test sample (x point in figure 2.7) can be rejected by a model based on a spherical Gaussian distribution while it is accepted by the PCA model.

These two models differ in the estimation of the space volume that confines the data distribution. For instance, inscribing an ellipsoid in a sphere determines the ratio of volume that is outside the ellipsoid. When increasing dimensionality this ratio tends to zero as an exponential decay (see figure 2.8). This means that the likelihood of failing in recognizing a pattern increases with dimensionality.

The previous example has been introduced with the aim of intuitively showing the contribution of the number of dimensions to the difference between the presented models. However, a more accurate analysis can be performed in order to expose the effects of dimensionality. Consider a data set with only one principal component describing the observations. PCA will give us two types of variance: one $\lambda_{\max}$ corresponding to the distribution of data along the principal direction, and another ϵ related to noise in the rest of dimensions. In this case different methods (Bayesian approaches, threshold, etc) can be used to determine why there is only one intrinsic dimension that represents data. Usually, a proportion $\nu = 90\%$ or more of the spectrum energy, $L = \sum_{i=1}^{d} \lambda_i$, should be contained in the principal component eigenvalues. We take this issue into consideration for a posterior reasoning.

When considering an spherical Gaussian model, we can see that the radius of the hyper-sphere containing the observations is the mean of the PCA eigenvalues:

$$r = \frac{1}{d}\sum_{i=1}^{d}\lambda_i = \frac{1}{d}\left[\lambda_{\max} + (1-\nu)L\right] \tag{2.1}$$

where d is the number of dimensions. Therefore, the volume of a hyper-sphere is proportional to:

$$V_s \sim r^d \tag{2.2}$$

According to the energy of the power spectrum $L = \sum_{i=1}^{d}\lambda_i$ and the proportion ν taken by the largest eigenvalue $\lambda_{\max}$, the noise amplitude ϵ can be expressed as follows:

$$\epsilon = \frac{L(1-\nu)}{d-1} \tag{2.3}$$

This equation assumes that noise is equally distributed on the rest of dimensions. The volume for the ellipsoid in this case is proportional to:

$$V_e \sim \lambda_{\max}\epsilon^{d-1} \tag{2.4}$$

The proportion factor for both hyper-volumes is the same and corresponds to the volume of a hyper-sphere of unit radius:

$$V_s(r=1) = \frac{\pi^d}{\Gamma(d/2+1)} \tag{2.5}$$

Since our purpose in comparing a ratio of volumes, this dimension dependent constant is not taken into account. The corresponding ratio V_s/V_e of the sphere volume with respect to the ellipsoid volume is:

$$R = \frac{V_s}{V_e} = \frac{r^d}{\lambda_{\max}\epsilon^{d-1}} \tag{2.6}$$

Considering $1-\nu$ sufficiently small and the number of dimensions d large ($d-1 \approx d$), the ratio R can be approximated to:

$$R \approx \frac{\left[\frac{1}{d}\lambda_{\max}\right]^d}{\lambda_{\max}\epsilon^{d-1}} \approx \left(\frac{\lambda_{\max}}{d\epsilon}\right)^d \tag{2.7}$$

Let us rewrite this ratio in terms of the portion of energy ν of the power spectrum L confined in $\lambda_{\max}$:

$$R = \left(\frac{\nu L}{d\frac{(1-\nu)L}{d-1}}\right)^d = \left(\frac{\nu}{1-\nu}\right)^d \tag{2.8}$$

where ϵ has been substituted by its value in equation (3.10). Therefore, we can conclude that *for $\nu > \frac{1}{2}$ the ratio between the sphere volume and the ellipsoid volume* *in high-dimensional spaces tends to infinity as an exponential function of the dimension d.* Moreover, this result concurs with the previous simple reasoning of inscribing an ellipsoid in a sphere. This fact is interpreted as follows: **the number of false-positives in the spherical Gaussian model with respect to the PCA model increases as an exponential function of the number of dimensions d.**

The same reasoning can be straightforwardly extended to more than one intrinsic dimension:

$$R = \frac{V_s}{V_e} = \frac{r^d}{\prod_{i=1}^{q} \lambda_i \epsilon^{d-q}} = \frac{\left[\frac{1}{d}\nu L\right]^d}{\prod_{i=1}^{q} \lambda_i \left(\frac{L(1-\nu)}{d-q}\right)^{d-q}} \tag{2.9}$$

Again, assuming $1-\nu$ small and $d >> q$:

$$\begin{aligned} R &= \frac{V_s}{V_e} = \frac{\left(\frac{1}{d}\nu L\right)^q}{\prod_{i=1}^{q} \lambda_i} \left(\frac{\nu}{1-\nu}\right)^{d-q} \approx \\ &\approx \left(\frac{1}{d}\right)^q \frac{(\sum_{i=1}^{q} \lambda_i)^q}{\prod_{i=1}^{q} \lambda_i} \left(\frac{\nu}{1-\nu}\right)^d \end{aligned}$$

where the equation $\nu L = \sum_{i=1}^{q} \lambda_i$ has been employed. At this point, we can apply the Cauchy-Schwarz inequality

$$\left(\frac{1}{q}\sum_{i=1}^{q} \lambda_i\right)^q \geq \prod_{i=1}^{q} \lambda_i \tag{2.10}$$

since all the eigenvalues are $\lambda_i > 0 \; \forall \; i$, such that $1 \leq i \leq q$, in order to show that:

$$\frac{(\sum_{i=1}^{q} \lambda_i)^q}{\prod_{i=1}^{q} \lambda_i} \geq q^q \tag{2.11}$$

Thus,

$$\frac{V_s}{V_e} \geq \left(\frac{1}{d}\right)^q q^q \left(\frac{\nu}{1-\nu}\right)^d \tag{2.12}$$

From this equation, it is straightforward to show that:

$$\lim_{d\to+\infty} \frac{V_s}{V_e} \geq \lim_{d\to+\infty} \left(\frac{1}{d}\right)^q q^q \left(\frac{\nu}{1-\nu}\right)^d = +\infty \tag{2.13}$$

for $d >> q$ and $\nu >> \frac{1}{2}$.

Summary

This section has described a particular type of effects that occur when dealing with high-dimensional spaces. To this end three steps have been employed progressively augmenting the level of complexity. It is worthy to emphasize that all three descriptions agree concerning the difference between the spherical Gaussian and the PCA models. This reasoning requires two conditions: i) the number of dimensions is larger than the number of internal degrees of freedom $d >> q$, and ii) the internal degrees of freedom hold most of the power spectrum energy $\nu >> \frac{1}{2}$.

Another relevant issue to point out is that: contrarily to our intuition, in high-dimensional distributions the tails are much more important than in one-dimensional ones [94]. This issue reinforces the statement introduced previously, (*the number of false-positives in the spherical Gaussian model with respect to the PCA model increases as an exponential function of the number of dimensions d*), since the tails of the spherical Gaussian model will contain a significant part of the mass.

2.2.2 Local Dimension Reduction

The combination of local linear models for explaining the global complex behavior of data is based on the following reasons:

- Taylor's theorem: any differentiable function becomes approximately linear in a sufficiently small region around a point (fig. 2.9(a)).
- The data manifold may actually consist of separate manifolds, which may or may not be connected together in one piece; i.e., it may be clustered (fig. 2.9(b)).
- The intrinsic dimensionality of the data may vary along the manifold (fig. 2.9(c)).
- The intrinsic dimensionality may not vary, but the orientation may vary as one moves along the manifold (fig. 2.9(d)).

The idea is that individual parts of a global data manifold can be estimated through simple linear models in order to cope with the global complex structure. In fact, using a complex global model able to represent a large number of manifolds (via a large number of parameters has several disadvantages:

- The power of the model is wasted in those areas of the space where the manifold is approximately linear.
- A large data set is required to fit a large number of parameters.
- Training becomes difficult because, due to the high flexibility of the model, the error function is likely to have a lot of local minima.

Focusing on small sub-problems, a global nonlinear manifold can be learned easily, fast and with few local minima. Moreover, the total number of parameters will be smaller, since some constraints on the local structure are applied in terms of prior knowledge. In this sense, local dimensionality reduction techniques require:

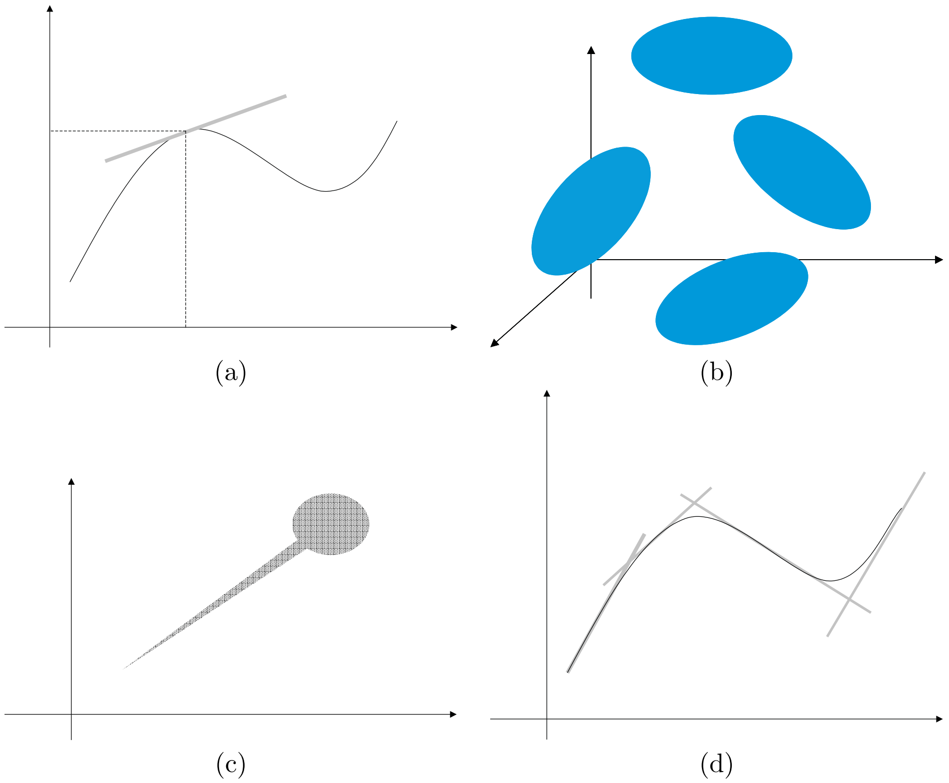


Figure 2.9: (a) Linear approximation of a function around a point. (b) Cluster distribution of data. (c) Data distribution with different local intrinsic dimensions. (d) Mixture of tangent linear models of a curve.

- Simple dimensionality reduction models as building blocks (typically PCA), usually distributed around the space and each one having a limited reach (hence the locality).
- A way to determine the dimensionality of each component.
- A responsibility assignation rule that, given a point in data space, assigns a weight or responsibility for it to each component. This can be seen as clustering.
- A way to learn both the local models (manifold fitting) and the responsibility assignment (clustering).

All these requirements can be handled by taking into account the probabilistic formulation presented in section 1.4.2 concerning mixtures of local linear models. From a probabilistic point of view, the concept of local models and responsibility assignation is naturally expressed as a mixture (of latent variable models) and was covered in section 1.4.2. The training criterion is then log-likelihood rather than reconstruction error, since the probability model attempts to model the noise as well as (and separately from) the underlying manifold. Formulating the local dimensionality reduction problem as a mixture of distributions results in a unified view of the whole model and its probabilistic nature brings a number of well-known advantages, in particular the fact that typically we can derive an EM algorithm that will train all parameters (those of the local models and those of the responsibility assignment) at the same time, with guaranteed convergence and often in a simple way: the E step assigns the responsibilities while the M step fits each local model. Under this approach, the responsibility assignment is carried out by the posterior probabilities for each mixture component. Nonetheless, this sort of computation, which has been widely employed in the literature, can be classified into two main groups according to the nature of the responsibilities:

- **Hard**: a single component receives all the responsibility and the rest receive no responsibility at all. It is a winner-take-all approach, usually a form of vector quantization.
- **Soft**: the responsibility is distributed among all components as a partition of unity, so that when training, a given data point will result in an update of all components; and when reducing dimensionality, the reduced-dimension representative will be the average of the local reduced-dimension representatives weighted by the respective responsibilities.

Usually, the suitability when using one of these two types of responsibility assignment relies on the specific nature and purposes of each problem. In fact, when it comes to classification, and thus clustering, data points for which more than one local model are significatively responsible are problematic. On the other hand, a soft assignment provides a continuous dimensionality reduction mapping, which is quite useful for connecting local and global approaches in terms of expected coordinates. In this sense, recently some new techniques on relating global and local coordinates

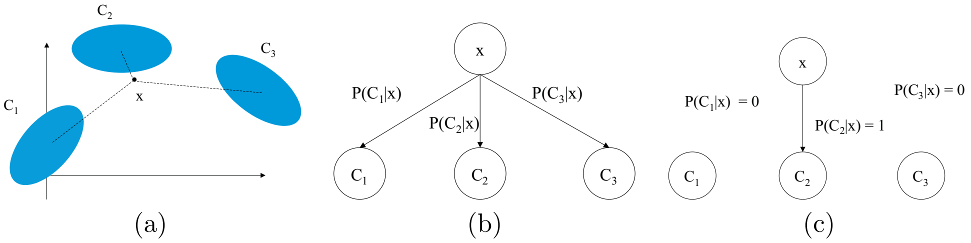


Figure 2.10: (a) A distance-like measure (posterior probability) is compared to each cluster for a given point. (b) Graphical model for a **soft** responsibility assignment. (c) Graphical model for a **hard** responsibility assignment.

have been presented with the purpose of exploratory data analysis and visualization. For instance the Local Linear Embedding (LLE) [86] is introduced as an unsupervised learning algorithm that computes low-dimensional, neighborhood-preserving embedding of high-dimensional inputs. Unlike clustering methods for local dimensionality reduction, LLE maps its inputs into a single global coordinate system of lower dimensionality, and its optimizations do not involve local minima. Nonlinear dimensionality reduction is also formulated using the Isomap technique [100, 101] that finds a Euclidean feature-space embedding of a set of observations that preserves as closely as possible their intrinsic metric structure.

2.3 Examples

To concrete some of the abstract concepts discussed in this chapter, we conclude with two examples of real-data applications:

- The first problem refers to applying a probabilistic version of PCA in order to recognize elongated structures in medical images. In fact, this is an example of dealing with a feature vector obtained from measurements on an image. According to the previous concepts on dimensionality reduction types and intrinsic degrees of freedom features, this problem can be categorized as follows:

  - Class of dimensionality reduction problem: **Hard**, regarding section 3.1.
  - **A priori information** on the intrinsic degrees of freedom: No. (Section 2.2).
  - **Global** dimension reduction problem with respect to the feature vector coordinates. (Section 2.2.2).

  In terms of locality referred to the pixel-grid, (see fig. 1.8), information is extracted from neighbor pixels in order to built a feature vector. After the detection process, a global use of the information is performed based on a perceptual organization of the detected structures in the snake framework.

- The second problem concerns 3D surfaces reconstruction from missing range data. This is embedded in the framework of Digital Archaeology. According to the previous concepts on dimensionality reduction types and intrinsic degrees of freedom features, this problem can be categorized as follows:
  - Class of dimensionality reduction problem: **Soft**, regarding section 3.1.
  - **A priori information** on the intrinsic degrees of freedom: Yes, symmetries on the generation of the type of the analyzed surfaces are exploited in order to represent them as a 1D problem. (Section 2.2).
  - **Local** dimension reduction problem with respect to the feature vector coordinates (3D points). This problem uses a **hard** assignation of responsibilities because of computational purposes. (Section 2.2.2).

  Information is exploited at two levels: local and global. Global information is perceived as a symmetry constraint on the possible values for the parameter estimates. Locally, data is divided in order to fit a sub-model to each portion.

2.3.1 An Example of PCA applied to Image Analysis

Part of the work presented in this thesis, concerning object detection/recognition in images, is focused on non-rigid vessel structures. This sort of problems have a relevant interest since both local and global information from the image must be suitably combined in order to obtain successful results. In fact, both aspects of information have the following implications:

Non-rigidity is the key factor that forces considering some local information of the structure, since the infinite number of possible global configurations of a vessel structure do not permit employing the standard global template matching techniques (see fig. 2.11). If we consider a cross section of the vessel structure along the tangent direction of the curve, a valley profile of the intensity values is obtained. In fact, all pixel locations belonging to the profile curve have this profile property. In fact, the profiles are the minimal information common to any linear structure, with the added value that are independent of morphological variabilities, i.e. curvature, orientation, etc.

Continuity and smoothness are conditions to be satisfied by this type of structures. Both assumptions concern the way global information is handled. In this particular case, these can be seen as a perceptual organization of the detected local structures.

The summarizing idea of this technique, which combines both aspects of information, is that: a similarity measure provided by the generative PCA model is used for building a likelihood map that indicates which pixels are potential parts of a vessel structure. This map behaves as a potential field for a deformable snake model in order to track the ridge of a vessel.

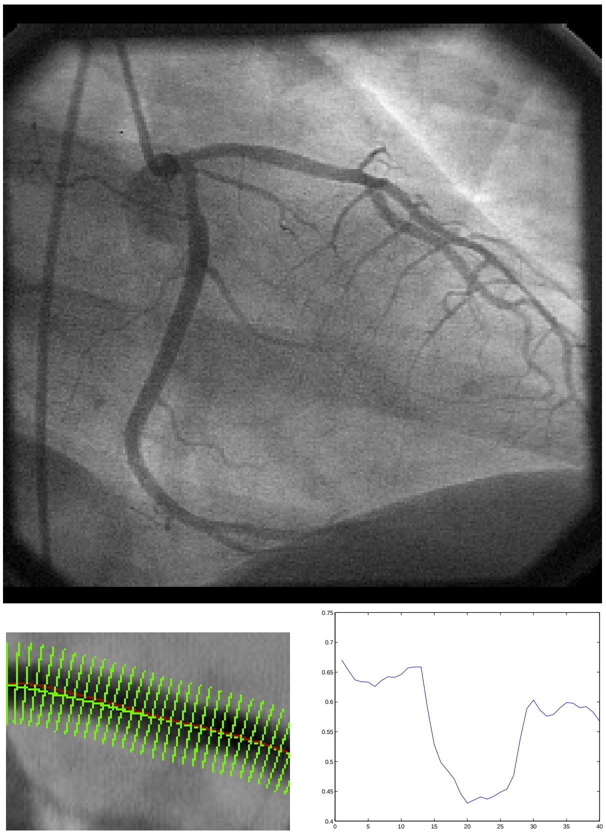


Figure 2.11: (Top) Example of a medical image with non-rigid vessel structures. (Bottom-left corner) Cross sections along the tangent direction of the vessel structure. (Bottom-right) Gray level profile for a pixel location.

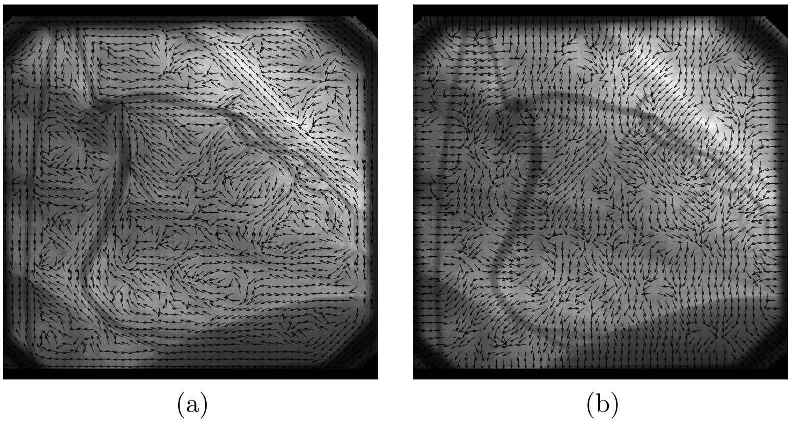


Figure 2.12: Directions associated to (a) the lower eigenvalue indicating a flow-like structure and larger one(b).

In this section, rather than introducing all the specifics involving the algorithm, we focus on the construction of a representation for the observed feature vectors. Thus, the part we are interested at this point concerns the localization of pixels belonging to a vessel structure.

Feature Vectors

The feature vectors correspond to the transversal gray level profiles, where 40 pixels are considered. This direction is obtained through the computation of the local orientation of the vessel at each pixel position. This process makes our technique view-point rotation invariant, since the local orientation is inherent to the vessel's shape. More specifically, the analysis of flow like structures (e.g. elongated objects) induces to considering the structure tensor field, which applied to an integration region ρ of the regularized image gradient ∇I_σ, measures the coherence between the regions and the searched structure [110]:

$$J_\rho(i,j) = K_\rho * (\nabla I_\sigma \nabla I_\sigma^T)(i,j) \quad (\rho \geq \sigma \geq 0) \tag{2.14}$$

where (i,j) are the image coordinates and K_ρ is a gaussian convolution kernel. The eigenvalues $\mu_{1,2}$ of the tensor (2.14) ($\mu_1 \geq \mu_2$) describe the average contrast variation in the eigendirections $\vec{w}_{1,2}$ ($\vec{w}_1 \perp \vec{w}_2$). The eigenvector associated to the lower eigenvalue, $\vec{w}_2$ is the orientation of lowest fluctuation, detecting the elongated flow, figure 2.12(a). The first eigenvector describes the directions of maximal grey-level variance fig. 2.12(b). This fact helps constraining the degrees of freedom of in the representation space due to orientation. In this sense, only shape variability is taken into account.

Intrinsic Degrees of Freedom

A reliable representation of these feature vectors must capture variabilities of shape due to the width of the profile as well as contrast and noise. To this end, we apply Principal Component Analysis technique in its probabilistic version, where the goals are:

- To show how dealing with a **similarity measure** helps to discern if a test sample belongs to the learned model.
- To show the **modes of variation** in the observations through the PC that explain the observation.
- To point out an example of the effects of dealing with **high dimensionality**, in terms of false/negative positives.

The first step is to collect a data set of examples corresponding to cross sections of vessel structures. This process determines which are the scales to be taken into account for a posterior detection process. One of the main goals for building a suitable representation of data is to reduce extra variabilities that can be present in the feature vectors. For instance, in this case, a normalization process can remove variations due to illumination contrast. Thus, from an original set presented in figure 2.13 (a), we obtain a normalized set in figure 2.13(b) where variations are only related to the profiles' shape. Moreover, we attempted to align all samples in order to avoid capturing translations in the model representation. This fact is determinant, otherwise more than the necessary intrinsic dimensions will be considered when building the model.

Parameters of the Model

After collecting a suitably processed data set, which takes only into account variations due to shape, the following step is to apply the PCA algorithm. This has been performed using the Singular Value Decomposition of the covariance matrix of the data set. The aim is to show the idea of intrinsic dimension by means of the eigenvalues and eigenvectors that codify the observations. The first issue that deserves to be commented is the form of the sample mean in figure 2.14(b). This corresponds to an ideal profile where the stochastic noise contribution has been removed. On the other hand, the power spectrum is used to ranking the eigenvectors in terms of their contribution to the reconstruction of a sample, in a Least Squares sense as in section 1.4.1.

In order to obtain an idea of the intrinsic dimensionality, we can analyze the accumulated version of the normalized power spectrum in figure 2.14(c). Close to 95% of the reconstruction power is contained in 9 Principal Components (see fig. 2.14(a)). This yields a latent space of 9 dimensions. Looking at the shape of the first 5 PC we can see that they mainly describe low frequency variations due to shape, such as width. The higher order terms are related to high frequency variations. Deciding the number of PC is a difficult task, since distinguishing noise perturbations from shape

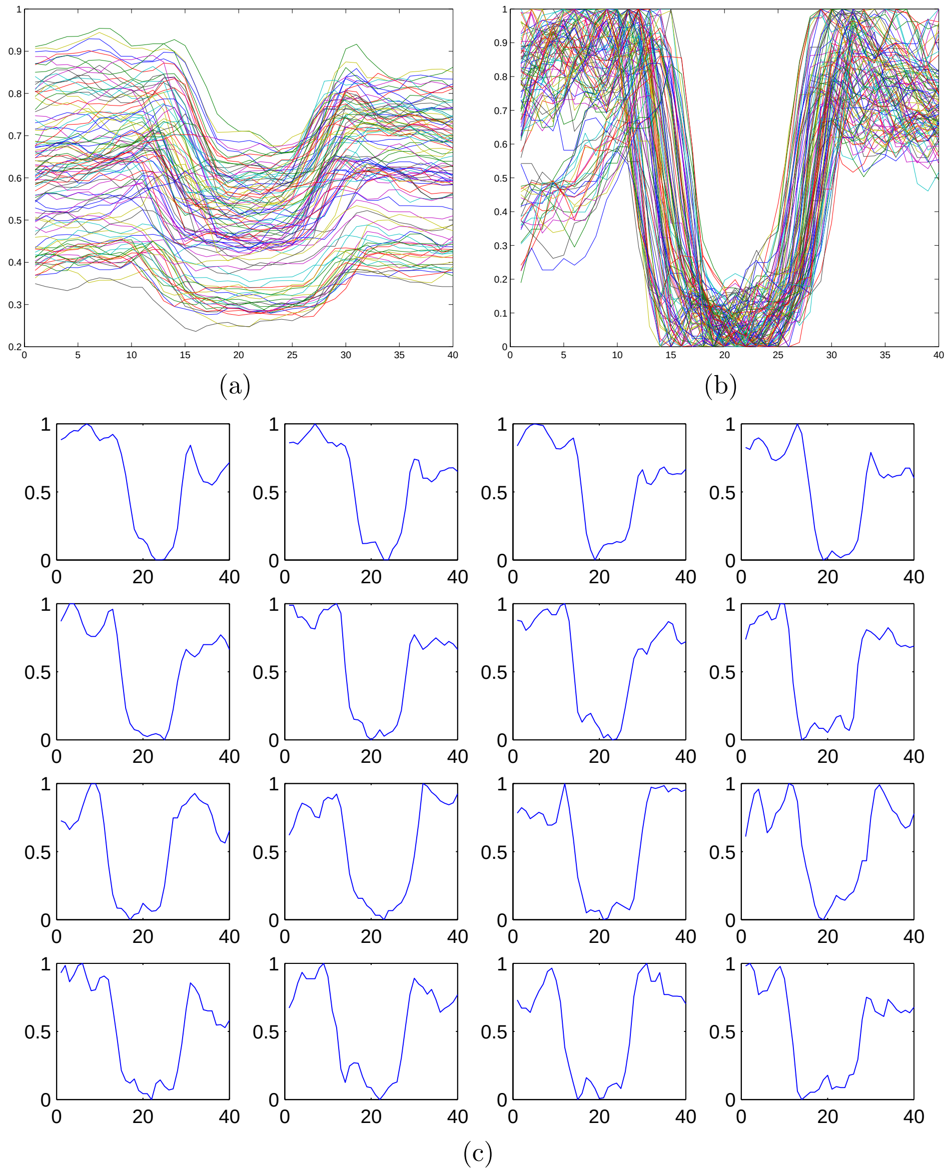
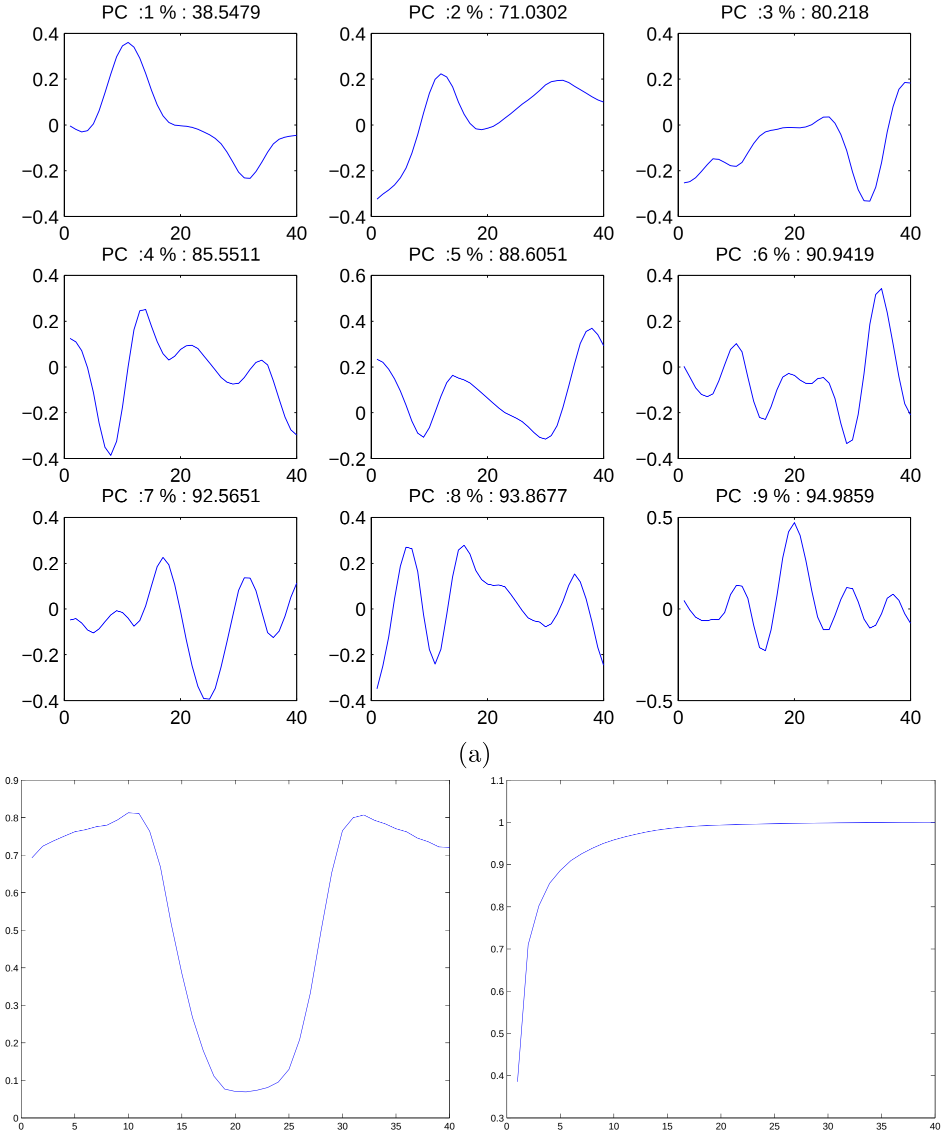


Figure 2.13: Some samples of the training data set. (a) Samples extracted from an image corresponding to the cross section of a vessel. (b) Samples after normalizing illumination. (c) Individual views of some samples.

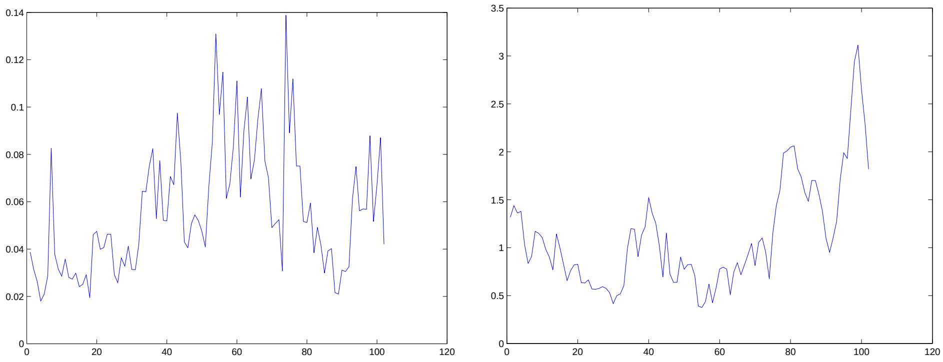


(a)

(b)

(c)

Figure 2.14: (a) 9 First Principal Components of the training data set. (b) Sample mean. (c) Accumulated sum of the normalized eigenvalue spectrum. Considering 5 PC implies 90% of reconstruction reliability.



(a) (b)

Figure 2.15: (a) Negative-loglikelihood of the training data set. (b) Euclidean distance of each sample to the mean.

variations requires a more sophisticate approach, such as a Bayesian framework that can be found in [12].

Recognition: PCA versus Euclidean Distance

The purpose of this section is to show an example version of the discussion introduced in section 2.2.1. We consider the negative-loglikelihood of each sample according to the parameters of the model previously presented. This sort of distance measure comes from the likelihood measure in equation (1.13). On the other hand, we take into account what happens if we measure through the Euclidean distance. Both plots are presented in figure 2.15. The aim of this is to focus on the higher error value for each measure. These values (0.14 and 3.2) are the upper bounds of our problem. From them, we can relatively compare both measures for a test sample. Note that for a test sample such as the one in figure 2.16 the Euclidean distance 2.7 is below the upper bound of the training set 3.2. In this sense, taking into account that all the training samples have been carefully selected so that, all of them are assumed to be vessel profiles, this profile ought to be accepted by the spherical Gaussian model, i.e., the model with zero PC. On the other hand, if we consider the likelihood measure provided by the PCA model with 9 PC, this error increases up to 2.7, which is 19 times far from the upper error bound of the training set, thus, this sample would be rejected. This fact makes relevant, the necessity of modelling data through reducing the dimensionality of the observations.

Finally, after learning a model for the vessel profile, we can apply the distance measure to each pixel location. In this process, the orientation vector field provided by the structure tensor in eq. (2.14) plays a crucial role. At each pixel location, a gray level profile of 40 pixels length is extracted. We take each pixel location in the image as the center of the profile line. The direction of extraction is given by the direction of maximum variance of the tensor field (e.g. the one provided by figure

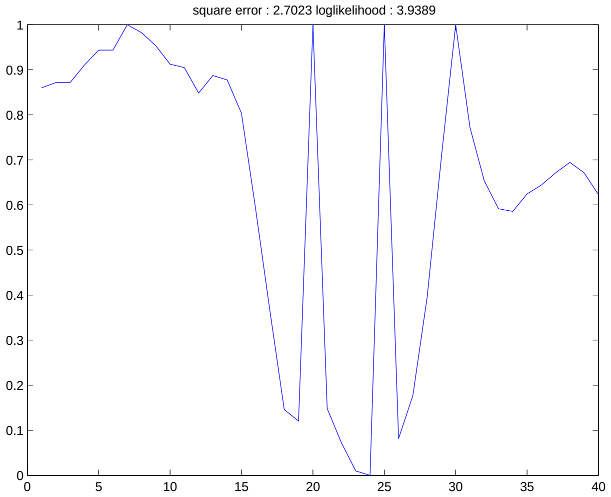


Figure 2.16: Test profile with an Euclidean distance of 2.7 and a negative-log-likelihood of 3.9.

2.12). Afterwards, each profile is compared with the learned model, and therefore, their corresponding distance measure is assigned to each pixel location, obtaining a negative-loglikelihood map. In order to present a first coarse level of segmentation, a threshold can be employed. To this end, the distance values 2 times bigger than the upper bound (see fig. 2.15) of the training data set have been rejected.

Still, some false responses are present in the resulting threshold images. Many reasons contribute to this fact: impulsive noise, low contrast regions, abrupt changes of illumination, etc... Different posterior techniques can be used in order to remove false responses. On one hand, we can use the idea introduced in our work [104]. Since, vessel present a parallel flow when applying the structure tensor (fig. 2.12), local parallelism (local average scalar product of flow vectors) enhances areas where a vessel is more likely to be. Moreover, a multi-scale approach for both structure tensor and vessel profiles will give better results.

2.3.2 A 3D Range Data Example of Local Models

A completely different problem is presented in this section. The purpose is to show the advantages of dealing with a mixture model in certain situations, where linear models are not sufficient.

This example deals with 3D range data, where the pixel locations correspond to 3D coordinates and the feature vectors in the observed space are the same 3D coordinates as well. This is actually, an interesting situation where both conceptions of locality mentioned in section 1.5 blend into a single one. In addition, 3D geometry problems are particularly useful, since they can be understood and visualized by means

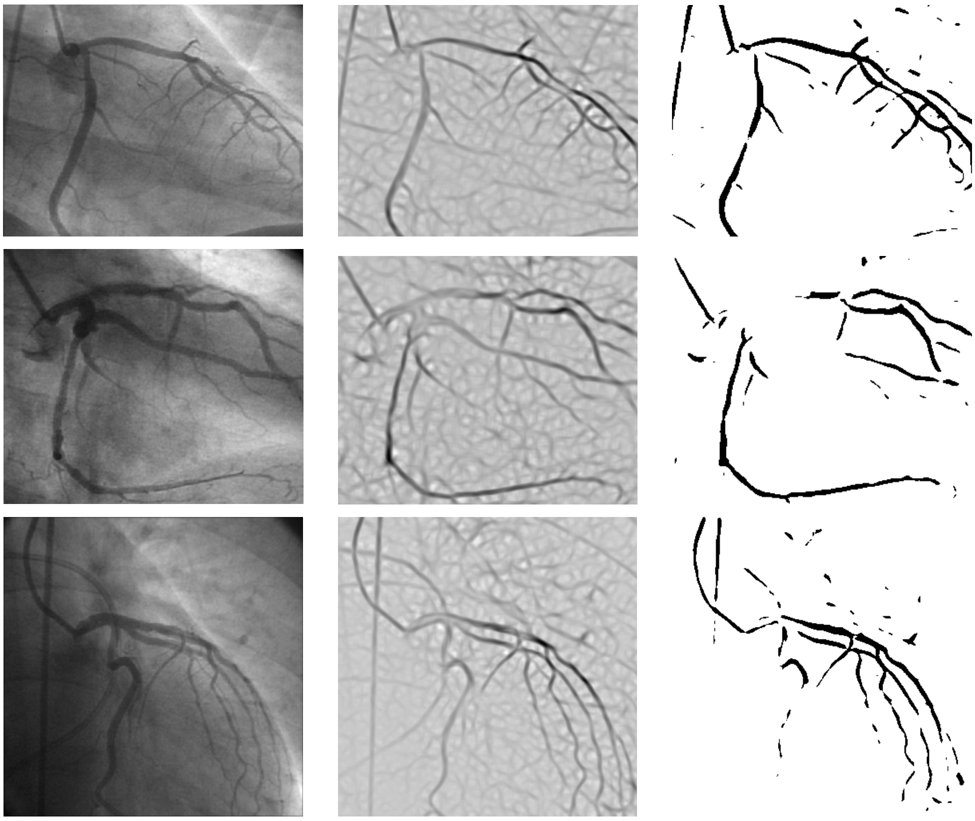


Figure 2.17: Left column shows the original images where the recognition algorithm has been applied. Central column shows the corresponding negative-loglikelihood map for each pixel location. Right column corresponds to a threshold on the model's distance measure for each pixel location.

of our spatial intuition. The idea of intrinsic degrees of freedom and dimensionality reduction is clearly based on symmetries and local approximations of surfaces through lower dimensional linear manifolds, such as planes and lines.

The interest is focused on fragments of archaeological pots. This is embedded in the framework of Digital Archaeology and it corresponds to part of our work presented in [113, 112, 24, 25, 72]. Many archaeological excavation sites are rich in fragments of pots, called sherds hereafter, which are either axially symmetric[3], or look as though they might have such rotational structure but really do not, e.g., the handles of a jar or flat sections of the surface of a plate. Two main reasons encourage the study of this sort of pieces:

- There is an access to **a priori information on the intrinsic degrees of freedom**. Axial symmetry is taken into account in order to reduce the problem from the observed 3 dimensions to 1D space.
- The archaeological pieces correspond to broken parts of a pot, and therefore, not all the information on the original pot is available. This fact leads to consider reconstruction algorithms that handle **missing data** and that provide a certain level of **prediction** for new points.

There is great scientific and cultural interest in the archaeological community in reconstructing these axially symmetric pots from the sherds found (see fig. 2.18). In fact, by taking advantage of axial symmetry, the number of possible combinations among patches, in the assembly search, is going to be reduced to a manageable one, i.e. attaching sherds together pairwise so that the axes, the profile curves and the break curves match.

This example presents a new technique for estimating the location and direction of the axis in cylindrical symmetrical data distributions. These distributions are 3D points obtained by means of CT and Laser scanners. The motivation behind this is to provide a setting for object reconstruction working with partial surface patches.

Previous Work

Parameter estimation in surfaces of revolution has recently been analyzed by Pottmann et al. [80, 81] from an algebraic geometry approach. In [81] authors describe a method that gives a result as closed linear form solution provided by a Plücker coordinates representation. In this coordinate system straight lines are represented in terms of six-tuples satisfying two constrains that involve their elements. The basic idea is to find the axis that generates the surface of revolution according to the following reasoning. If, for each point on the surface a straight line is drawn, which follows the normal direction to the surface, all the lines have to intersect the axis of revolution (see fig. 2.19). Consequently, taking a plane perpendicular to this axis, the lines belonging to

[3] the intersection of the pot outer surface with a plane perpendicular to the pot axis is a circle or nearly so.

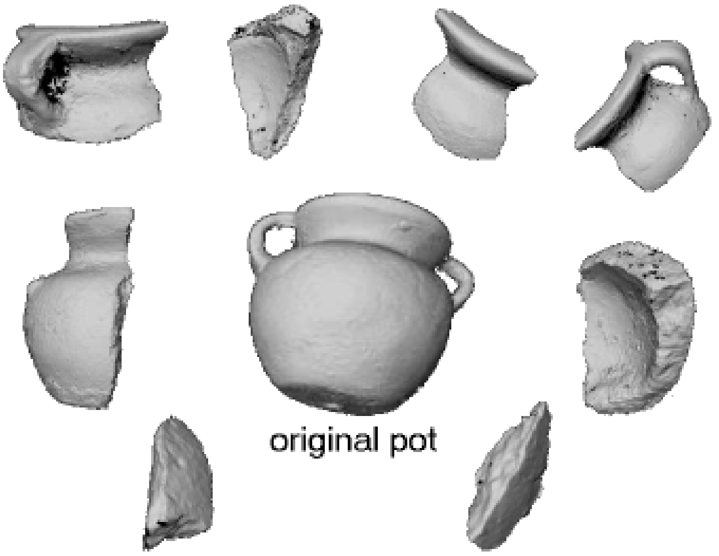


Figure 2.18: Constituent sherds of a cylindrical symmetric pot.

this plane will cross at a certain point. All the points on the surface that belong to this plane have the same radius. Therefore, given a data set and their normals, the goal is to find a straight line (axis of revolution) that minimizes the square distance to each line. This analysis strongly depends on the computation of the normals and noise in the observed data. However, global information, such as knowledge on profile curve properties, is not used. For instance, one of the characteristics of a noiseless surface is that the profile curve should have a very small thickness (tending to zero). Local information is related to extracting local geometrical properties among neighboring points in the surface. This information is in the normal directions associated with each point on the surface.

However, what distinguishes our work is that we use all the information available from the data, *local* and *global*, leading to a more accurate noise treatment. Given that we are dealing with surfaces of revolution, this type of cylindrical symmetric objects are described sufficiently by a profile curve and a location of the axis that generates its symmetry.

The Algorithm

The main idea of this algorithm is that the intersection of a plane perpendicular to the axis of revolution of an axially symmetric surface is a circle, the radius of which depends of the position of the plane with respect to the axis. This functional relation describes a profile curve in a 2D space that considers one axis to represent the radius and another one to represent the relative projected position across the axis of revolution.

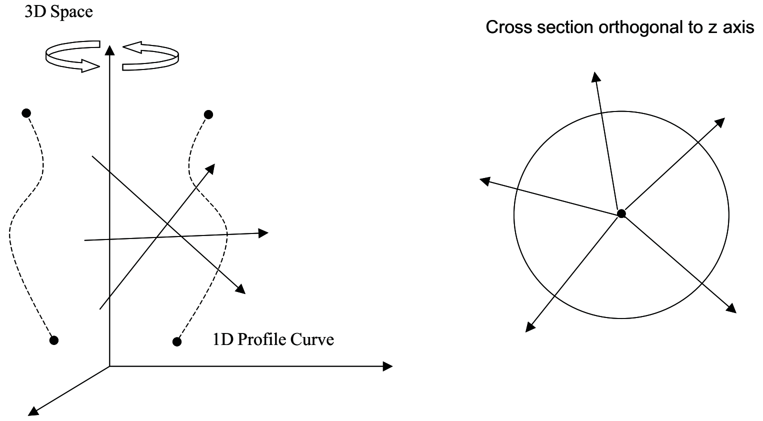


Figure 2.19: The extension of the normal directions to the surfaces intersect the axis of revolution.

Given the parameters for the axis: the orientation and the position, the surface can be represented in a 2-dimensional space. One dimension is referred to the distance perpendicular to the axis, and the other one corresponds to the distance from a point taken as origin along the axis. In this space, each point is mapped onto a circle in the 3D world, each line segment is back projected into a conical surface, and for any general one dimensional manifold we have a generic surface of revolution (see fig.2.20).

Considering this representation, if we are able to estimate the axis parameters for an incomplete data set, then the reconstruction of the complete surface is possible using the previous mentioned back projection from $2D$ to $3D$ spaces. We term generation to this process, since from partial data, the original surface is recovered (see fig. 2.20). Moreover, the inverse process, i.e., taking the 3D data configuration to its latent space representation (2D), can be seen as a procedure of redundance reduction, since the data is reduced to a new representation with fewer parameters without losing any information of the surface.

Let's describe mathematically the projection from the $3D$ space onto the profile curve space. First, assume that the axis parameters, i.e., the direction $\mathbf{v}$ and a point $\mathbf{q_0}$ belonging to the axis, are known. So that, for any $3D$ point $\mathbf{p} = (x', y', z',) \in \Re^3$ the projection to a point $(r, z) \in \Re^2$ of the latent space is given by:

$$\mathbf{p} = (x', y', z') \quad \longrightarrow \quad \omega = (r, z), \begin{cases} r = \sqrt{(\mathbf{p} - \mathbf{q_0})^T (1 - \mathbf{v}\mathbf{v}^T)(\mathbf{p} - \mathbf{q_0})} \\ z = (\mathbf{p} - \mathbf{q_0})^T \mathbf{v} \end{cases} \tag{2.15}$$

The complexity of the surface relies on the functional relation between the radius r and the parallel projection onto the axis z, i.e., $r = r(z)$. The model that attempts to estimate the axis parameters $(\mathbf{v}, \mathbf{q_0})$ has to make some assumptions about the profile curve behavior.

To this end, we propose a mixture model that considers different cross sections along the direction of the axis of revolution. The points belonging to one slice have in

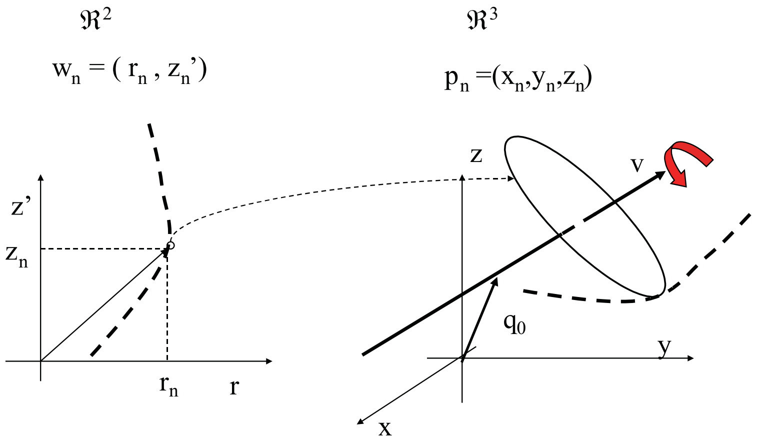


Figure 2.20: Generative mapping from the profile curve space (latent space) to the 3D observations space (back projection from $2D$ to $3D$).

common the same radius. Locally, data can be modelled as a circle orthogonal to the axis of revolution (see fig. 2.21). Data partitioning can be performed by dividing the height of the profile curve into several segments (e.g. equally distributed). This data partitioning considers a **hard** assignation of the responsibilities for each data point with respect to the different slices. For the m-th slice, the points $(\mathbf{p}_1^m, \ldots, \mathbf{p}_{N_m}^M)$ that are assigned to it satisfy the following implicit equation:

$$(x_n^m - x_0^m)^2 + (y_n^m - y_0^m)^2 = R_m^2 \tag{2.16}$$

where (x_n^m, y_n^m) are the XY coordinates of the point $\mathbf{p}_n^m$. The center coordinates (x_0^m, y_0^m) for all m partitions are constrained, since they belong to a line, which determined by the axis parameters (orientation and position):

$$\left.\begin{array}{ll} x_0^m & = m_x z_0^m + b_x \\ y_0^m & = m_y z_0^m + b_y \end{array}\right\} \tag{2.17}$$

These equations contain the for unknown parameters that specify the axis of revolution. Two of them m_x and m_y describe the slope of the line when it is projected onto the xz-plane and yz-plane, respectively. The remaining two parameters define the intersection of the line with the xy-plane at $z = 0$.

Using the equation of the axis (2.17) in the implicit equation (2.16) for the m-slice will determine the system of equations that describe the generation of this mixture model. First, let equation (2.16) be expanded:

$$\begin{array}{rcl} x^2 + x_0^2 - 2xx_0 + y^2 + y_0^2 - 2yy_0 - R^2 & = & 0 \\ x^2 + y^2 - 2xx_0 - 2yy_0 & = & R^2 - x_0^2 - y_0^2 \\ x^2 + y^2 - 2xx_0 - 2yy_0 & = & B \end{array}$$

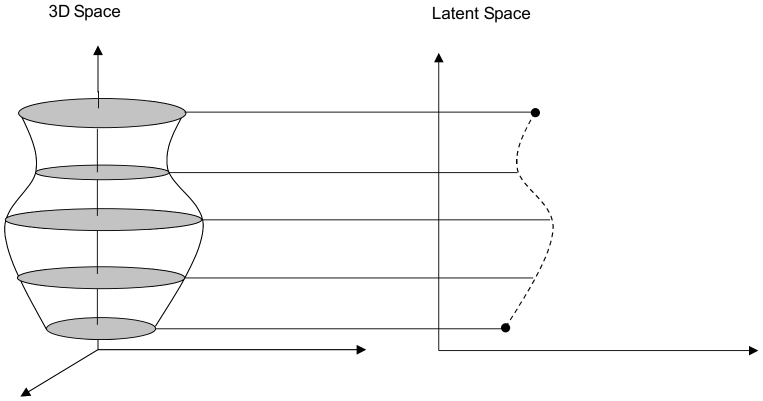


Figure 2.21: Orthogonal slices to the axis of revolution. The points belonging to a slice have the same radius in common.

where we have skipped the n and m indexes for the sake of notation. In addition, $B = R^2 - x_0^2 - y_0^2$ has been introduced as a new parameter for linearity purposes in the estimation algorithm. Note that, this can be performed without lost of generality since, R^2 still has one free degree of freedom of the equation $B = R^2 - x_0^2 - y_0^2$. Subsequently, we merge both equations (2.16) and (2.17):

$$x^2 + y^2 - 2x(m_x z + b_x) - 2y(m_y z + b_y) \quad = \quad B$$

which can be re-written as follows:

$$\frac{x^2 + y^2}{2} = [xz, x, yz, y, \frac{1}{2}] \begin{bmatrix} m_x \\ b_x \\ m_y \\ b_y \\ B \end{bmatrix}$$

This latter equation can be extended to a many-slices problem, where the number of parameters increases according to the number of radii taken into account with $\{B_1(R_1^2), \ldots, B_M(R_M^2)\}$:

$$\frac{x^2 + y^2}{2} = [xz, x, yz, y, 0, 0, \ldots, 0, \frac{1}{2}, 0, \ldots, 0] \begin{bmatrix} m_x \\ b_x \\ m_y \\ b_y \\ B_1 \\ B_1 \\ \vdots \\ B_M \end{bmatrix}$$

where the non-zero value $1/2$ indicates the mixture component that the point (x, y, z) is assigned to. For a distribution of N points and a mixture model of M components, we define the following vectors and matrix:

$$P_{N\times 1} = \begin{bmatrix} \frac{x_1^2+y_1^2}{2} \\ \vdots \\ \frac{x_n^2+y_n^2}{2} \\ \vdots \\ \frac{x_N^2+y_N^2}{2} \end{bmatrix}$$

and,

$$Q_{N\times 4+M} = \begin{bmatrix} x_1z_1, x_1, y_1z_1, y_1, & 0, & \ldots, 0, & \frac{1}{2}, 0, & \ldots, 0 \\ \vdots & \vdots & & & \\ x_nz_n, x_n, y_nz_n, y_n, & 0, & \ldots, 0, & 0, 0, & \ldots, 0, \frac{1}{2}, 0 \\ \vdots & \vdots & & & \\ x_Nz_N, x_N, y_Nz_N, y_N, & 0, & \ldots, 0, \frac{1}{2}, & 0, 0, & \ldots, 0 \end{bmatrix}$$

and,

$$\Omega_{4+M\times 1} = \begin{bmatrix} m_x \\ b_x \\ m_y \\ b_y \\ B_1 \\ B_1 \\ \vdots \\ B_M \end{bmatrix}$$

Thus, the matrix form for the structural equation is:

$$P_{N\times 1} = M_{N\times(4+M)}\Omega_{(4+M)\times 1} + E_{N\times 1} \tag{2.18}$$

where, $E_{N\times 1}$ is a Gaussian noise vector of N independent identically distributed variables. For a given observed point $\mathbf{p}_n = (x_n, y_n, z_n)$ the noise distribution corresponds to:

$$P(\mathbf{p}_n = (x_n, y_n, z_n) \mid \Omega) = \frac{1}{(\sqrt{2\pi\sigma^2})^3} \exp\left\{-\frac{1}{2\sigma^2}|P_n - M_n\Omega|^2\right\} \tag{2.19}$$

The third ingredient are the latent variables, which in this case correspond to the binary indices 0 or 1/2 that indicate which mixture component explains each point. At this point, two issues deserve to be pointed out before introducing the estimation process:

- This mixture model with hard assignations for each point with respect to the slices constrains all the mixture components through the axis of revolution. Therefore, rather than transforming the problem into many independent sub-problems, we have decomposed the problem into manageable sub-problems that are constrained to the prior knowledge imposed by axial symmetry.

- Ideally, if we align the surface's axis of revolution to the z axis, the slope parameters m_x and m_y will be equal to zero. Of course, if the axis of revolution is far from the z axis, this formulation is clearly biased, since only XY projections are used for the computation of the radii in equation (2.16). For this reason, it is advisable to take into account an algorithm such as Pottmann‘s in [81] for initializing the search of the axis of revolution.

Even though the Plucker-coordinates based algorithm seems to work well, we show in the experiments the contribution of considering global and local information in the same formulation. Thus, the use of the Plücker coordinates solution as starting point of our estimation algorithm can compensate the mentioned lacks in the use of information as well as it will reduce the cost of search of the maximum of the likelihood function.

Initialization

In order to give a graphical idea of the algebraic geometry approach[4], we show in figure (2.19) a surface with its corresponding normals and a profile curve projection (fig. 2.19 b). Given that all the normals of a surface of revolution have to cross the axis of symmetry, the purpose is to find the straight line that minimizes the distance with all the linear extensions that follow the normal directions. This straight line is considered in [81] as the axis of revolution, whose minimization scheme is a generalized linear least squares solution. A deeper insight shows us that the recipe to perform the solution is as follows:

Let $D = \{\mathbf{p_1}, \ldots, \mathbf{p_N}\}$ the data set of N points, and $D_{norm} = \{\mathbf{n}_1, \ldots, \mathbf{n}_N\}$ their corresponding normals on the surface.

1. For each couple of $\mathbf{p_k}$ and $\mathbf{n_k}$ $(k = 1, \ldots, N)$ compute their cross product, such that,
$$\mathbf{r}_k = \mathbf{p_k} \times \mathbf{n_k}$$
2. Form a six-tuple using the concatenation of the normals and the cross product set:
$$l_k = [\mathbf{r_k}, \mathbf{n_k}] \in \Re^6$$
3. Build the covariance matrix of the six-tuples set:
$$M = \sum_{k=1}^{N} l_k {l_k}^T$$
4. Due to the unit norm constraint on the normals, authors in [81] include the matrix $B = \text{diag}\{1, 1, 1, 0, 0, 0\}$ and reduce the problem to solve the equation
$$\det(M - \lambda B) = 0$$
whose solution corresponds to the smallest general eigenvalue $\lambda \geq 0$.

[4] For a further lecture, authors in [81] give a description based on projective geometry.

5. Let ϕ the (General SVD) solution, then the parameters of the axis $\mathcal{M} = \{\mathbf{v}, \mathbf{q_0}\}$ can be computed as follows:

   Let the six-tuple ϕ be in the form of two $3D$ concatenated vectors,

   $$\phi = [\mathbf{r}_\phi, \mathbf{n}_\phi]$$

   then the $3D$ parameters solution is:

   $$\mathbf{v} = \frac{\mathbf{r}_\phi}{\mid \mathbf{r}_\phi \mid} \tag{2.20}$$

   $$\mathbf{q_0} = \frac{\mathbf{n}_\phi \times \mathbf{r}_\phi}{\mid \mathbf{r}_\phi \mid^2} \tag{2.21}$$

The employed technique is based on a general singular value decomposition (GSVD) of a rank 3 matrix $[BM]_{6\times6}$, as well as, the optimal solution taken is the one that corresponds to the smallest general eigenvalue. The relation between the six-tuple and its $3D$ geometrically meaning is given by (2.20) and (2.21). These will be the starting point for our estimation algorithm, i.e., the first projection onto the profile curve space will be based on these results. With this beginning, we take an initial solution, that has not taken into account the global information and the prior knowledge of the profile curve, in order to apply an optimal adjustment that uses all the available information in the data.

In the appendix, we show another approach to derive this formulation. It is based on Lie Groups theory and perhaps more intuitive, since the formalism is performed keeping the 3D geometrical intuition.

Estimation Process

After initializing the model with Pottmann‘s solution, the estimation process for the parameters of this model is consists in two steps which are repeated until convergence:

- Align the data set to the z-axis according to the solution for the axis parameters.
- Divide data in subsets with respect to the corresponding z-values (deterministic latent variables inference). Estimate the new axis parameters through maximum likelihood of equation (2.19):

  $$\Omega = [M'M]^{-1}M'P \tag{2.22}$$

  and repeat the aligning step again.

A convergence criterium can be the noise variance σ^2, or the difference between consecutive estimates for the axis parameters. After convergence, the original axis can be found through estimating the Euclidean transformation (rotation + translation) that suffered the data set from its original configuration.

Experiments

There are still two open questions: i) *How many mixture components are necessary to describe data*? and, ii) *How Pottmann's solutions differ from the slices approach*? To answer them, we show some experiments with real data.

The election of the number of mixtures has a significant influence in the shape estimation as well as in the estimation of axis of symmetry. Regarding the idea of the certain degree of tolerance for splitting the profile space into segments, there will be nodes that do not need to be split up. The purpose of this is to avoid over-fitting and the confusion noise and geometrical properties of a surface such as curvature. There are two important issues to be considered: one hand, the *noise* in data originated from observations (scanning, etc...) and, on the other, the *error* due to model assumptions. Thus, this fact can lead to a misleading comprehension of the estimation results of any piece, since *the noise in the data* and the *uncertainty in the estimation* are not distinguished without the prior information. In particular, if we know a priori, by some other means, the number of the degrees of freedom (mixture components) that describe the complexity of the shape, the distinction between noise and uncertainty in the estimation of the parameters is clarified.

In the framework we are dealing with, the error due to the model comes from fitting a cylinder to a region, and regions with high curvature yield a high error measure in the estimates. In order to deal with the entanglement noise/error, we apply to our technique the *Minimum Description Length* [83], where the aim is to evaluate the plausibility of different alternative models explaining the same observations (distribution of points). This evaluation is performed on the Occam's Razor Principle which states that one should not make more assumptions than the minimum needed.

In this Bayesian framework, "sufficiently" means a trade off between the error measure in the estimates and the complexity of the model. It is considered more complex a model that describes data with many planes, than just one plane, since many more degrees of freedom (number of parameters) are involved. This model selection criterion translates into the introduction of a penalty term in equation (2.19). This penalty term comes from approximating the posterior distribution for a d-dimensional parameter set $\hat{\theta}$ by a Gaussian [33], so that the evidence for a data set D under a set of hypotheses $\{h_i\}$ is written as follows:

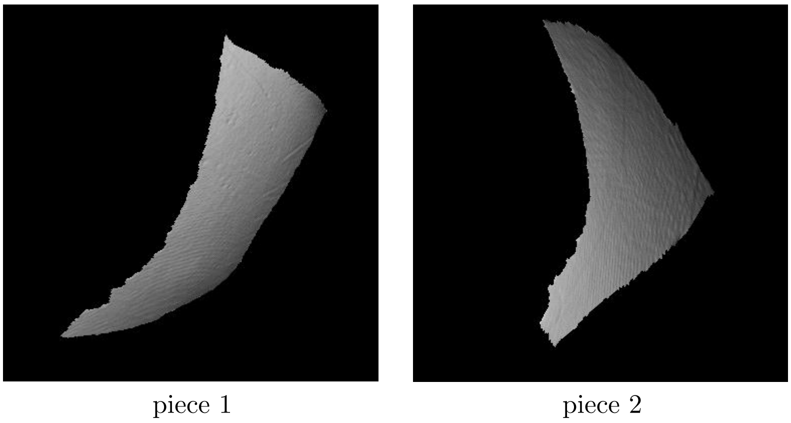
$$P(D|h_i) \approx P(D|\hat{\theta}, h_i) P(\hat{\theta}|h_i)(2\pi)^{d/2}|H|^{-1/2}$$

where $H = \nabla\nabla \log P(D|\hat{\theta}, h_i)$ is a Hessian matrix which measures how peaked the posterior is around the Maximum a Posteriori value. In other words, this measure tells us about the uncertainty of the estimates $\hat{\theta}$.

Therefore, the negative logarithm of the likelihood measure for the data set has now a penalty term concerning the variance in the estimates:

$$\mathcal{L} = \frac{1}{\sigma^2} \text{trace}\left((P - M\Omega)'(P - M\Omega)\right) + \log|M'M| \tag{2.23}$$

The first point that we notice is that when the number of mixture components is increased the complexity of the shape of the object is expected to be higher. This



piece 1 piece 2

Figure 2.22: Pieces of a cylindrical symmetric pot.

fact means that the number of degrees of freedom for the curvature is increased as well and the model of for shape leads to a deficiency in the use of the all available information in the data, i.e., the contribution of the global information is missed.

To perform the model selection we need to try with different mixture models and then compute the most reliable model that explains the complexity of the shape of the object. In the case of symmetries of revolution, the computation of a several number of models is not so computationally expensive. Thus, a good procedure is starting with a simple model, which contains one or two components, and then, increasing the number of mixtures. Given the estimates of the axis for each model, the following step is based on the comparison of the different uncertainty measures eq.(2.23). The quantity that expresses the least uncertainty corresponds to the most reliable model for the estimation of the axis, as well as, for the estimation of the profile curve of the object. Notice that, when the number of mixtures is increased the use of global information takes less relevance and the penalty term tends to infinity. This fact has an important significance, since within the limits of the real pieces it means that we have to analyze the uncertainty of a few number of models.

Figure 2.22 shows two small pieces that are use for comparing the presented methods. First, we note that both pieces a small in relation with their curvature. This is a relevant fact, since they approximate to a sphere. This sort of surfaces point out the necessity of taking into account global information. In fact, if we consider only local information such as in Plucker coordinate method, the algorithm may give unreliable solutions. Pottmann's algorithm presents a singularity when dealing with nearly spherical surfaces. We have applied our slices approach two both pieces, and compared to the solutions provided by Plucker coordinates. Figure 2.23 shows a plot of the 3D distribution of points for piece 1 and the estimate axis by means of our algorithm. At the bottom of figure 2.23, a comparison between the two profile curves is shown. The profile in blue is thinner and thus less scattered than the one provided by the Plucker approach. The same procedure is applied for piece 2 in figure 2.24. In this case, the solution obtained through Plucker approach is not a profile corresponding to the mentioned piece; no multiple values are given in the 3D spatial distribution for this specific piece.

Summary

In this example, we presented a probabilistic framework that provides a setting for the utilization of all the available information that lies in the data, in order to determine the generating axis of a surface of revolution from partial data, as well as, an estimation of its shape. The fact that we are dealing with a distribution of points that is governed by a specific symmetry led to consider a new representation (latent space) where the shape of the object is only expressed in terms of its intrinsic degrees of freedom. Considering the case of axial symmetries this representation encodes to the profile curve of the surface.

The probabilistic formulation plays a significant role connecting the $3D$ world and the latent space by means of a likelihood measure, which quantifies the adjustment of the data to a shape model for a given instance of the axis of symmetry. In addition to this, this formulation allows the incorporation of prior information to the model in a natural way. Both the choice of a sufficient representation and the use of prior information are the basis of our approach. On one hand, the definition of a suitable representation for the observations permits a concrete delimitation of the analysis of the complexity of the data set distribution. Furthermore, when a selected representation is sufficient, therefore the reconstruction of the complete object from partial data is permissible (see fig. 2.25). On the other, the use of prior knowledge on the complexity of the shape affords a connectivity between the global and the local information and an appropriate treatment of the noise in the data.

2.4 Symmetry, Lie Groups and Dimensionality Reduction

This section studies a novel approach of Linear Complexes algorithm [81] based on Lie Groups theory. The aim of this is to show a Generative Model formulation that intrinsically takes into account the necessity for reducing the number of dimensions when dealing with symmetries. In particular, we consider axial symmetries on 3D range data. Lie's group theory applied to Computer Vision is not new. In order to get an insight into this framework, we recommend [50], where a comprehensive view of its applications is developed.

The technique we present exploits the properties of Lie Algebras to provide a method for integrating rotational transformations in a consistent manner. The use of Lie Algebras gives a substantial advantage over traditional constrained methods for determining transformations, since the intrinsic degrees of freedom are clearly defined from the beginning of the formulation. In this framework, these degrees of freedom are related to the parameters that govern the symmetry transformation. We focus on the

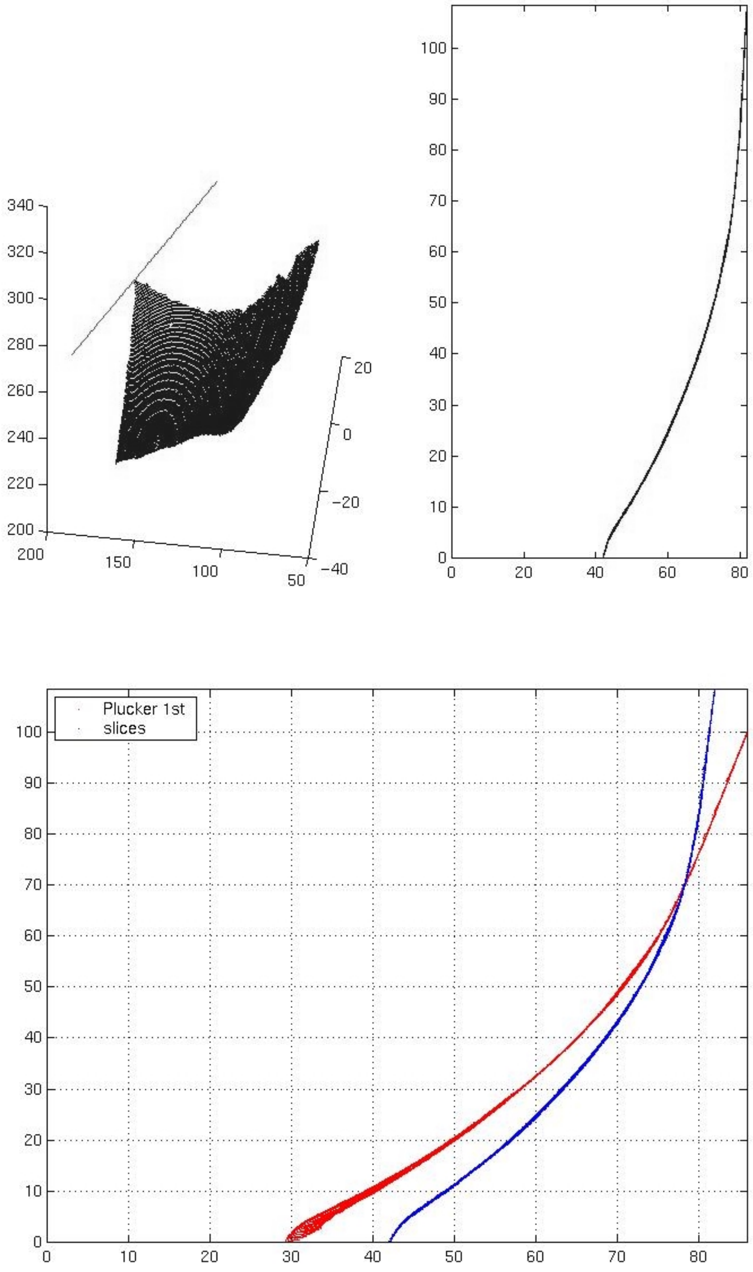


Figure 2.23: Top: 3D point distribution corresponding to piece 1 in figure 2.22. Bottom: Data mapped onto the profile curve space according to Pottmann‘s method (in red) and the mixture model technique (in blue).

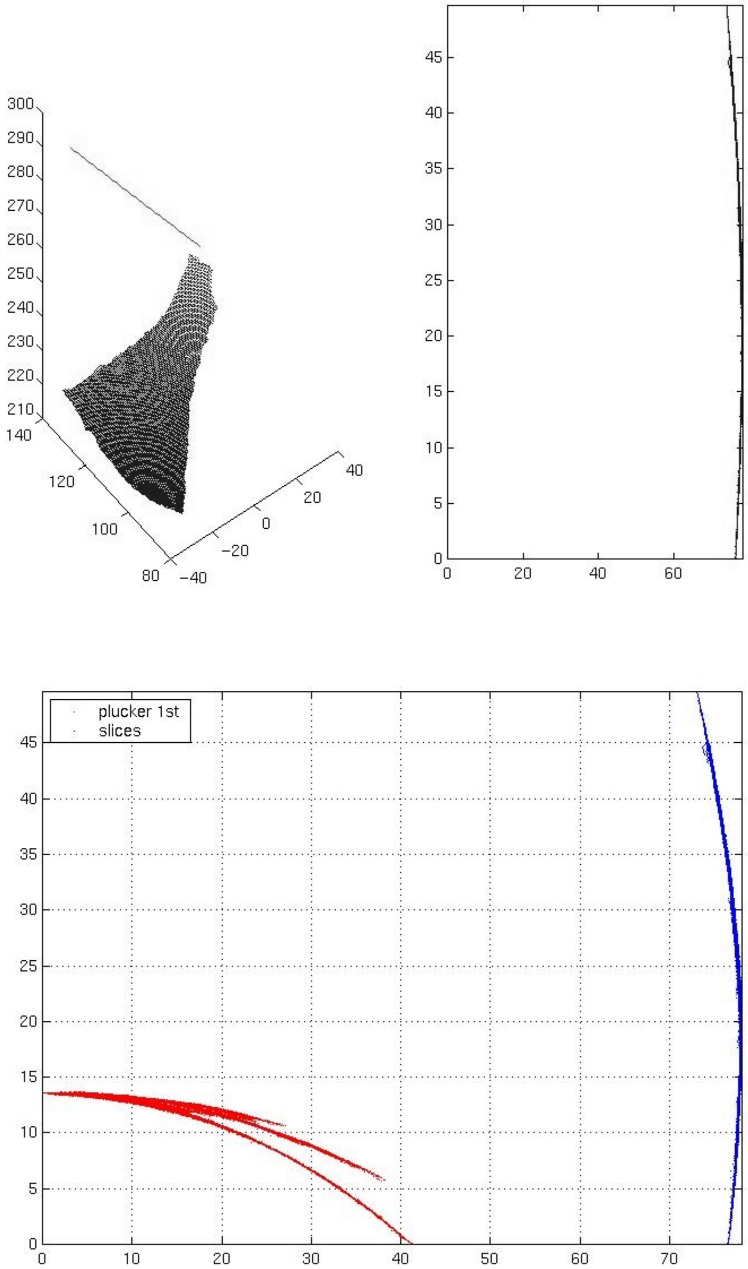


Figure 2.24: Top: 3D point distribution corresponding to piece 2 in figure 2.22. Bottom: Data mapped onto the profile curve space according to Pottmann‘s method (in red) and the mixture model technique (in blue).

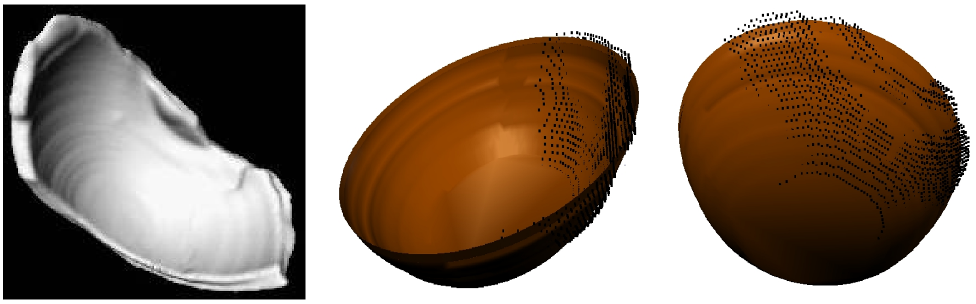


Figure 2.25: Piece of a cylindrical symmetric pot and its reconstruction aligned to the original data.

group of transformations in 3D space that leave the quantity $x^2+y^2+z^2$ invariant, i.e., the group of rotations in 3 dimensions, $SO(3)$. In fact, **invariance** and **symmetry** are two intrinsically related concepts, since a problem admitting a symmetry group G leads to a G-invariant representation. In physics and the Differential Equations community this is a well-known fact[5].

In our case, axial symmetry leads to a representation that is invariant with respect to rotation transformations around the axis of symmetry. Therefore, taking into account such a symmetry-invariance relationship, i.e. a specific way of reducing dimensionality, a posterior reconstruction of the whole surface from missing data is feasible. In other words, **the process that reduces dimensionality taking into account the internal symmetries of a problem provides a manner of dealing with missing data and makes possible predicting new points of the surface**.

There are three common ways to parameterize these 3D rotations:

- Successive rotations about three mutually orthogonal fixed axes.
- Successive about the z-axis, about the new y-axis, and then about the new z-axis. These are called *Euler* angles.
- The axis-angle representation, defined in terms of an axis whose direction is specified by a unit vector (two parameters) and a rotation about that axis (one parameter).

For instance, the first type of parametrization is based on the following rotation matrices:

[5] Noether's first theorem, (Noether, 1918), associates a conservation law for the Euler-Lagrange equations with every one-parameter symmetry group of the variational problem. For instance, translation invariance leads to conservation of linear momentum, rotation invariance leads to conservation of angular momentum, and time translation invariance leads to conservation of energy. Noether's second theorem, of application in relativity and gauge theories, produces dependencies among the Euler-Lagrange equations arising from infinite-dimensional variational symmetry groups.

$$
\begin{aligned}
R_1(\phi_1) &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\phi_1 & -\sin\phi_1 \\ 0 & \sin\phi_1 & \cos\phi_1 \end{pmatrix} \\
R_2(\phi_2) &= \begin{pmatrix} \cos\phi_2 & 0 & \sin\phi_2 \\ 0 & 1 & 0 \\ -\sin\phi_2 & 0 & \cos\phi_2 \end{pmatrix} \\
R_3(\phi_3) &= \begin{pmatrix} \cos\phi_3 & -\sin\phi_3 & 0 \\ \sin\phi_3 & \cos\phi_3 & 0 \\ 0 & 0 & 1 \end{pmatrix}
\end{aligned}
$$

This sort of transformations are called continuous since they are parameterized by a set of continuous parameters (angles). Thus, for each matrix and for each parameter ϕ a point (x, y, z) is mapped onto some point (x', y', z'). In addition, when the parameters of the transformation tend to zero, both points become the same.

2.4.1 Infinitesimal Transformations

The purpose of this section is to obtain an approach for continuous transformations that allows isolating the transformation parameter ϕ.

Consider a one parameter transformation group defined by the transformation matrix $R(\phi)$, such that a point $\mathbf{p} = (x, y, z)$ is mapped onto some point $\mathbf{p}' = (x', y', z')$ by the following relation:

$$\mathbf{p}' = R(\phi)\mathbf{p}$$

or which is the same,

$$\mathbf{p}(\phi) = R(\phi)\mathbf{p}$$

The key point to be considered is that when ϕ tends to zero, the associated transformation is the identity, recovering the initial point (x, y, z).

In a first approximation order, the relation between an image $\mathbf{p} = (x, y, z)$ and a near one transformed $\mathbf{p}(\delta\phi) = (x', y', z')$ can be expressed as:

$$\mathbf{p}(\delta\phi) \simeq \mathbf{p}(0) + \delta\phi \left.\frac{d\mathbf{p}(\phi)}{d\phi}\right|_{\phi=0} = \mathbf{p}(0) + \delta\phi \left.\frac{dR(\phi)}{d\phi}\right|_{\phi=0} \mathbf{p}(0)$$

where a matrix independent of the transformation parameter is applied on $\mathbf{p}(0)$:

$$G = \left.\frac{dR(\phi)}{d\phi}\right|_{\phi=0}$$

which is referred as the infinitesimal generator (or action) of the group of transformations for $R(\phi)$. Thus, a macroscopic transformation $R(\phi)$ can be built in terms of concatenating infinitesimal transformations, dividing the parameter ϕ in M parts and making $M \to \infty$:

$$p(\phi) = \lim_{M\to\infty} \left(1 + \frac{\phi}{M} G\right)^M \mathbf{p}(0) = e^{\phi G}\mathbf{p}(0)$$

Note, that to derive this macroscopic transformation, we have used the property of groups:

$$R(\phi_1 + \phi_2) = R(\phi_1)R(\phi_2)$$

In addition, the same result can be obtained considering a Taylor's expansion of the transformed point $\mathbf{p}(\phi)$:

$$\begin{aligned} \mathbf{p}(\phi) &= \sum_{n=0}^{\infty} \frac{\phi^n}{n!} \left.\frac{d^n R(\phi)}{d\phi^n}\right|_{\phi=0} \mathbf{p}(0) = \\ &= \sum_{n=0}^{\infty} \frac{\phi^n}{n!} G^n \mathbf{p}(0) = \\ &= e^{\phi G}\mathbf{p}(0) \end{aligned}$$

where we have taken into consideration the mentioned property of groups, which is used in the computation of the n-th order derivatives of $R(\phi)$:

$$\begin{aligned} \frac{dR(\phi)}{d\phi} &= \lim_{h\to 0} \frac{R(\phi + h) - R(\phi)}{h} = \\ &= \lim_{h\to 0} \frac{R(h)R(\phi) - R(\phi)}{h} = \\ &= \left(\lim_{h\to 0} \frac{R(h) - 1}{h}\right) R(\phi) = \\ &= GR(\phi) \end{aligned}$$

and, thus, the n-th derivative is written as follows:

$$\frac{d^n R(\phi)}{d\phi^n} = G^n R(\phi)$$

From this formulation, we can summarize that one-parameter Lie groups correspond to the continuous transformations $R(\phi)$ that satisfy the two following conditions:

- When the parameter value ϕ tends to zero the transformation $R(\phi)$ becomes the identity:

  $$R(\phi)|_{\phi=0} = I$$

- They satisfy the following differential equation:

  $$\frac{dR(\phi)}{d\phi} = GR(\phi)$$

  which is a consequence of the previous condition and the addition law applied to this group of transformations, i.e., $R(\phi_1 + \phi_2) = R(\phi_1)R(\phi_2)$.

2.4.2 Infinitesimal Generators for SO(3)

Three parameter independent infinitesimal generators can be obtained applying the previous conditions for Lie groups of transformations:

$$G_1 = \left.\frac{dR_1(\phi_1)}{d\phi_1}\right|_{\phi_1=0} = \frac{d}{d\phi_1}\left.\begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\phi_1 & -\sin\phi_1 \\ 0 & \sin\phi_1 & \cos\phi_1 \end{pmatrix}\right|_{\phi_1=0} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{pmatrix}$$

$$G_2 = \left.\frac{dR_2(\phi_2)}{d\phi_2}\right|_{\phi_2=0} = \frac{d}{d\phi_2}\left.\begin{pmatrix} \cos\phi_2 & 0 & \sin\phi_2 \\ 0 & 1 & 0 \\ -\sin\phi_2 & 0 & \cos\phi_2 \end{pmatrix}\right|_{\phi_2=0} = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{pmatrix}$$

$$G_3 = \left.\frac{dR_3(\phi_3)}{d\phi_3}\right|_{\phi_3=0} = \frac{d}{d\phi_3}\left.\begin{pmatrix} \cos\phi_3 & -\sin\phi_3 & 0 \\ \sin\phi_3 & \cos\phi_3 & 0 \\ 0 & 0 & 1 \end{pmatrix}\right|_{\phi_3=0} = \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

These generators have an interesting property defined by the Lie bracket:

$$[G_1, G_2] = G_1G_2 - G_2G_1 = G_3$$

and more generally:

$$[G_i, G_j] = \epsilon_{ijk}G_k$$

where ϵ_{ijk} is the Levi-Civita antisymmetric pseudo-tensor; for two indices with the same value (e.g., $i = j$), then $\epsilon_{iik} = 0$, and non-cyclic permutations change the sign, ±1.

These commutation relations define a "product" of two generators which yields the third generator. Thus, the set of generators is closed under this operation. Triple products, which determine whether or not this composition law is associative, can be written in a concise form using only the definition of the commutator, i.e., in the form of an identity, without any explicit reference to the quantities involved. This yields to the **Jacobi** identity:

$$[A, [B, C]] + [B, [C, A]] + [C, [A, B]] = 0$$

For the infinitesimal generators of the rotation group each of the terms in the Jacobi identity vanishes. Thus,

$$[A, [B, C]] = [[A, B], C]$$

so the product of these generators is associative. In the more general case, however, products of quantities defined in terms of a commutator are not associative.

The **Lie Algebra** associated with the Lie group from which the generators are obtained consists of quantities A,B,C,... defined by

$$A = \sum_{i=1}^{3} a_i X_i \quad B = \sum_{i=1}^{3} b_i X_i \quad C = \sum_{i=1}^{3} c_i X_i$$

where the $a_i, b_i; c_i$,... are real coefficients and from which linear combinations $\alpha A + \beta B$ with real α and β can be formed. The product is given by

$$[A, B] = -[B, A]$$

and the Jacobi identity is, of course, satisfied.

2.4.3 Axial Symmetry

A rotation generated around a specific axis of revolution can be described as a transformation of the following form:

$$\mathbf{p}(\phi) = e^{\phi A}(\mathbf{p}(0) - \mathbf{q}_0) + \mathbf{q}_0 \tag{2.24}$$

where A is the infinitesimal generator of transformation and μ is a point that belongs to the axis of revolution. Two operations have been applied to generate the new transformed point $\mathbf{p}(\phi)$: i) a translation provided by $\mathbf{q}_0$, and, a rotation ii) given by the exponential form of the rotation matrix.

In fact, equation (2.24) is the **structural equation** of our **generative model**, since a one dimensional latent variable ϕ is used to describe each point belonging to a surface of revolution.

In addition, given that the previous mentioned generators, G_1, G_2 and G_3, for the $SO(3)$ group form a basis in the associated Lie algebra, the infinitesimal generator A can be written as a linear combination of them:

$$A = \sum_{i=1}^{3} v_i G_i$$

where there is a straightforward relation of the components of the basis (v_1, v_2, v_3) and the coordinates of the direction vector of the axis of revolution where the rotation is performed. Since the axis of revolution is defined by a vector that corresponds to a linear combination of the basis x-plane, y-plane and z-plane, the combination of the basis of the rotations generators corresponds to the same components. Thus, the direction vector can be expressed as follows:

$$\mathbf{v} = \begin{pmatrix} v_1 \\ v_2 \\ v_3 \end{pmatrix}$$

Therefore, the axis of revolution is determined by the components of the direction vector $\mathbf{v}$ and a point $\mathbf{q}_0$.

On the other hand, the tangent vectors to the points of a surface of revolution can be computed through taking the derivative with respect to the rotation parameter ϕ:

$$\mathbf{w}(\phi) = \frac{d\mathbf{p}(\phi)}{d\phi} = Ae^{\phi A}(\mathbf{p}(0) - \mathbf{q}_0) = A(\mathbf{p}(\phi) - \mathbf{q}_0)$$

And these vector, since they are locally tangent to the surface, they must be orthogonal to the normals of the surface os revolution:

$$\mathbf{n}'\mathbf{w}(\phi) = 0 \tag{2.25}$$

$$\tag{2.26}$$

thus,

$$\mathbf{n}'A(\mathbf{p}(\phi) - \mu) = 0$$

Studying the product of the transposed normal vectors $\mathbf{n}'$ and the infinitesimal generator A, we can see that:

$$\begin{aligned}
\mathbf{n}'A &= (n_1, n_2, n_3)[v_1G_1 + v_2G_2 + v_3G_3] = \\
&= (n_1, n_2, n_3)\begin{pmatrix} 0 & -v_3 & v_2 \\ v_3 & 0 & -v_1 \\ -v_2 & v_1 & 0 \end{pmatrix} = \\
&= (\mathbf{n} \times \mathbf{v})'
\end{aligned}$$

Therefore, the equation that must be satisfied for all the points belonging to a surface of revolution with respect to an axis $(\mathbf{v}, \mathbf{q}_0)$ is:

$$(\mathbf{n} \times \mathbf{v})'(\mathbf{p}(\phi) - \mathbf{q}_0) = 0$$

and thus,

$$\begin{aligned}
\mathbf{v}'(\mathbf{p}(\phi) \times \mathbf{n}) + (\mathbf{v} \times \mu)'\mathbf{n} &= 0 \\
(\mathbf{v}|\mathbf{v} \times \mathbf{q}_0)\begin{pmatrix} \mathbf{p}(\phi) \times \mathbf{n} \\ \mathbf{n} \end{pmatrix} &= 0
\end{aligned}$$

The estimation of the model parameters can be performed using the same process introduced in section 2.3.2. In this latter equation corresponds to the same equation expressed in [81] which is used for estimating the axis direction $\mathbf{v}$ and location $\mathbf{q}_0$. Let $\mathbf{s} = [\mathbf{r_s}, \mathbf{n_s}]$ the (General SVD) solution of the matrix:

$$M = \sum_{k=1}^{N}[\mathbf{p}_k \times \mathbf{n}_k, \mathbf{n}_k]\begin{bmatrix} \mathbf{p}_k \times \mathbf{n}_k \\ \mathbf{n}_k \end{bmatrix}$$

which under a normalization constraint for the direction vector $\mathbf{v}$, is obtained from the following system of equations:

$$\det(M - \lambda B) = 0$$

where $B = \mathrm{diag}(1,1,1,0,0,0)$. Thus, the parameters of the axis $\mathcal{M} = \{\mathbf{v}, \mathbf{q}_0\}$ can be computed as follows:

$$\begin{aligned} \mathbf{v} &= \frac{\mathbf{r_s}}{\mid \mathbf{r_s} \mid} && (2.27) \\ \mathbf{q}_0 &= \frac{\mathbf{n_s} \times \mathbf{r_s}}{\mid \mathbf{r_s} \mid^2} && (2.28) \end{aligned}$$

which are the solutions proposed by Pottmann [81].

2.4.4 Implicit Surface Parametrization

This section briefly comments an alternative method that uses the introduced ideas in this chapter and takes advantage of dealing with global and local information at the same time.

Given that the computation of the normal vectors of an implicit surface is straightforward, the equation (2.25) can be rewritten in terms of a parametrization for the observed data set. In fact, this is a way of considering global information.

Let $F(\mathbf{p}) = 0$ be an implicit form of a surface. Thus, the normal vectors, can be achieved by:

$$\mathbf{n} = \nabla F(\mathbf{p})$$

and the equation satisfied by a surface of revolution is:

$$\begin{aligned} (\mathbf{a}|\mathbf{a} \times \mu) \begin{pmatrix} \mathbf{p}(\phi) \times \mathbf{n} \\ \mathbf{n} \end{pmatrix} &= 0 \\ (\mathbf{a}|\mathbf{a} \times \mu) \begin{pmatrix} \mathbf{p}(\phi) \times \nabla F(\mathbf{p}) \\ \nabla F(\mathbf{p}) \end{pmatrix} &= 0 \end{aligned}$$

where $\mathbf{p}(\phi) \times \nabla$ is the infinitesimal rotator applied to the surface implicit function $F(\mathbf{p})$. If we express the function $F(\mathbf{p})$ in terms of a series of monomials:

$$\begin{aligned} F(x,y,z) &= \alpha_0 + \alpha_1 x + \alpha_2 y + \alpha_3 z + \alpha_4 x^2 + \alpha_5 xy + \ldots \\ F(x,y,z) &= \Phi(x,y,z) \begin{pmatrix} \alpha_0 \\ \alpha_1 \\ \vdots \\ \alpha_d \end{pmatrix} = \\ &= \Phi(x,y,z)\mathbf{B} \end{aligned}$$

where Φ is a d-dimensional vector, then the normals correspond to:

$$\mathbf{n} = \nabla F(\mathbf{p}) = \nabla \Phi(x, y, z)\mathbf{B}$$

In this case, this estimation process requires a two step procedure:

- Compute the axis parameters as described before.
- Compute the surface parameters in terms of the points and the axis parameters:

$$\begin{aligned}[\mathbf{a}'(\mathbf{p} \times \nabla\Phi(x, y, z)) + (\mathbf{a} \times \mu)'\nabla\Phi(x, y, z)]\,\mathbf{B} &= 0 \\ \mathbf{L}(p, a, \mu)'\mathbf{B} = 0\end{aligned}$$

  In addition, the surface parameters $\mathbf{B}$ can be constrained to be normalized, i.e., $\mathbf{B}'\mathbf{B} = 1$. Thus, this problem is transformed again into a SVD problem, i.e. find $\mathbf{B}$ such that:

$$\sum_{n=1}^{N} \mathbf{L}_n \mathbf{L}_n' \mathbf{B} = \lambda \mathbf{B}$$

  where a data set of N points has been taken into account. The optimal solution is the eigenvector $\mathbf{B}$ with lowest eigenvalue λ.

These two steps are iterated until a selected degree of convergence of the error. Note, that the estimation of the axis parameters is given by the estimation of the surface coefficients, and the estimation of the coefficients is given by the value of the axis parameters. Nonetheless, we consider more stable from a numerical point of view the method that we present in section 2.3.2, since it is more constrained. As mentioned in the introduction, dividing a problem into linear subproblems is more controllable than a complex nonlinear problem from a numerical approach.

2.5 Discussion

This chapter has introduced the problem of dimensionality reduction as the search for a tractable and reduced coordinate representation of a submanifold of a high dimensional Euclidean space. This is a problem so far not yet solved in a satisfactory and general way. Nonetheless, certain specific features can be analyzed in order to categorize the different types of problems that involve the necessity for reducing the number of dimensions. This classification has been performed in terms of: i) the specific purpose for perceptual categorization, ii) the use of global and local information, iii) the availability of prior information on the intrinsic degrees of freedom, and, iv) the role of symmetries when reducing dimensionality.

Purpose-driven perceptual categorization. Usually, there is not a direct relation between the stimuli provided by a general-purpose sensor device and its corresponding perceptual category. A complex learning task must be involved in order to provide such a connection. In fact, the basic forms of energy, and their possible combinations, are a reduced number compared to the infinite possible perceptual categories corresponding to objects, actions, relations among objects, etc. In addition, the observations provided by a sensor correspond to a set of variables, whose number is rather larger than desired. More specifically, in Computer Vision, the most used sensors are digital cameras and laser scanners. Both represent objects in terms of a set of variables that are highly **correlated**. For instance, given a set of different observations from objects concerning the same category, the variability in each pixel value is constrained to a certain range, which is influenced by the rest of pixel value variations. In this sense, dimensionality reduction can be seen as a **feature extraction** or coding procedure, or in general as a representation in a different coordinate system.

Three classes of dimensionality reduction problems have been presented in terms of the purpose that guides a problem:

- Hard dimensionality reduction problems.
- Soft dimensionality reduction problems.
- Visualization problems.

The first two points have been exemplified with one problem related to recognition in images and another corresponding to 3D range data. The purpose of the first example was to show how a feature vector can be extracted from images in order to define a specific category of objects, which in this case corresponded to non-rigid vessel structures. The second one has been presented in order to show an example of *soft dimensionality reduction* by means of 3D range data. Both cases present a strong correlation among variables and observations due to redundancy in the modes of variations in the first problem and symmetries in the second one.

In addition, this chapter has analyzed the advantages of modelling sub-manifolds when it comes to **recognition**. Many problems in Computer Vision require of a decision mechanism to discern if a test pattern belongs to the same category as the learned model for a set of training observations. In this area, metric approaches base this decision on a distance measure, which is determined by the employed model. The success of a model, in terms of recognition error, mainly depends on how this distance measure is defined and the number of dimensions that are involved. In this sense, one section has been focussed on *the curse of dimensionality* in recognition problems. Particularly, a comparison between PCA and the spherical Gaussian model has been performed in order to point out the differences in terms of false-positives when increasing dimensionality.

Global and local treatment of information. Local dimensionality reduction is an issue that takes relevance when dealing with complex behaviors of data. The idea is that individual parts of a global data manifold can be estimated through simple linear models in order to cope with the global complex structure. In fact, using a complex global model able to represent a large number of manifolds (via a large number of parameters) has several disadvantages:

- The power of the model is wasted in those areas of the space where the manifold is approximately linear.
- A large data set is required to fit a large number of parameters.
- Training becomes difficult because, due to the high flexibility of the model, the error function is likely to have a lot of local minima.

The formulation of a model in terms on **linear local sub-models** leads to consider data partitioning and class assignation. Probabilistic mixture models deal with classification through the use of posterior probabilities. This sort of assignation is called **soft partitioning**. There are other situations where **hard clustering** can be suitable due to computational reasons. In this case, a single component receives all the responsibility and the rest receive no responsibility at all.

Usually, the suitability when using one of these two types of responsibility assignment relies on the specific nature and purposes of each problem. In fact, when it comes to classification, and thus clustering, data points for which more than one local model are significatively responsible are problematic. On the other hand, a soft assignment provides a continuous dimensionality reduction mapping, which is quite useful for connecting local and global approaches in terms of expected coordinates.

Prior information and symmetries. The presented examples have been selected according to the access to the prior information on the intrinsic degrees of freedom. The vessel-structure detection problem corresponds to an example where no prior information is given on the latent space dimension.

On the other hand, the problem related to finding the axis of revolution of a distribution of points counted on the use of prior information on the axial symmetry. This chapter has shown that a priori information on symmetries regarding the distribution of data brings a powerful assistance to build a model for explaining and predicting behaviors. A straightforward association between Generative Models and Lie's continuous groups theory has been studied in order to reduce dimensionality according to the symmetries in the problem. **Invariance** and **symmetry** concepts have been introduced in order to show that taking into account the internal symmetries of a problem provides a manner of dealing with **missing data** and it makes possible **prediction**.

The selection of the two different examples presented in this chapter had the purpose of illustrating how the mentioned points are taken into account when reducing dimensionality. The aim was to cover the main theoretical background that has been used in the applications presented in this thesis.