Facing Uncertainty:
3D Face Tracking and Learning with Generative Models

A dissertation submitted in partial satisfaction of the
requirements for the degree
Doctor of Philosophy

in

Cognitive Science

by

Tim Kalman Marks

Committee in charge:

James Hollan, Chair
Javier Movellan, Co-Chair
Garrison Cottrell
Virginia de Sa
Geoffrey Hinton
Terrence Sejnowski
Martin Sereno

2006
ABSTRACT OF THE DISSERTATION

Facing Uncertainty:
3D Face Tracking and Learning with Generative Models

by

Tim Kalman Marks
Doctor of Philosophy in Cognitive Science

University of California San Diego, 2006

James Hollan, Chair
Javier Movellan, Co-Chair

We present a generative graphical model and stochastic filtering algorithm for simultaneous tracking of 3D rigid and nonrigid motion, object texture, and background texture from single-camera video. The inference procedure takes advantage of the conditionally Gaussian nature of the model using Rao-Blackwellized particle filtering, which involves Monte Carlo sampling of the nonlinear component of the process and exact filtering of the linear Gaussian component. The smoothness of image sequences in time and space is exploited using Gauss-Newton optimization and Laplace’s method to generate proposal distributions for importance sampling.

Our system encompasses an entire continuum from optic flow to template-based tracking, elucidating the conditions under which each method is optimal, and introducing a related family of new tracking algorithms. We demonstrate an application of the system to 3D nonrigid face tracking. We also introduce a new method for collecting ground truth information about the position of facial features while filming an unmarked subject, and introduce a data set created using this technique.

We develop a neurally plausible method for learning the models used for 3D face tracking, a method related to learning factorial codes. Factorial representations play a fundamental role in cognitive psychology, computational neuroscience, and machine learning. Independent component analysis pursues a form of factorization proposed by
Barlow [1994] as a model for coding in sensory cortex. Morton proposed a different form of factorization that fits a wide variety of perceptual data [Massaro, 1987b]. Recently, Hinton [2002] proposed a new class of models that exhibit yet another form of factorization. Hinton also proposed an objective function, contrastive divergence, that is particularly effective for training models of this class.

We analyze factorial codes within the context of diffusion networks, a stochastic version of continuous time, continuous state recurrent neural networks. We demonstrate that a particular class of linear diffusion networks models precisely the same class of observable distributions as factor analysis. This suggests novel nonlinear generalizations of factor analysis and independent component analysis that could be implemented using interactive noisy circuitry. We train diffusion networks on a database of 3D faces by minimizing contrastive divergence, and explain how diffusion networks can learn 3D deformable models from 2D data.
Chapter I

Introduction

The problem of recovering the three dimensional structure of a scene or object from two-dimensional visual information has long been a focus of the computer vision and artificial intelligence communities. Marr [1982] and his contemporaries, for example, proposed a number of computational theories for “decoding” 3D structure from low-level properties of 2D images, endeavoring to recover shape from shading, structure from apparent motion, depth from optical flow, surface orientation from surface contours, depth using stereopsis, and so on. Like many of his predecessors, Marr saw inferring the 3D structure of the world as a critical step towards viewpoint- and lighting-independent recognition of objects.

Much of the challenge of machine perception lies in creating systems to accomplish perceptual tasks that humans perform effortlessly, such as object identification and tracking. The human visual system can take in a noisy collection of jumbled pieces of low-level visual information and quickly determine, for example, that a few feet away is a woman’s face, turned slightly to the left and smiling. Humans’ ability to determine high-level structural and semantic information from low-level 2D observations is an example of the general perceptual problem of determining the “hidden” root causes of our observations.

**Discriminative vs. Generative Models** Computer models of perception that attempt to determine high-level information from low-level signals generally fall into two categories: discriminative, and generative. The goal of the discriminative approach is
to find functions that map directly from observed data (e.g., observed images) to the underlying causes of those data (e.g., a head’s location, orientation, and facial expression). Typical examples of discriminative models include multi-layer perceptrons (neural networks) and support vector machines that are trained in a supervised manner. Discriminative models such as these can be described as “black box” approaches to perception: the system can perform the task successfully without it being clear just how the task is being accomplished. An important part of the analysis of such a system is often to discover the principles behind the way that the system has learned to accomplish its task.

From a probabilistic point of view, we can think of discriminative models as direct methods for learning the mapping from the observed values of a random variable, $X$, to a probability distribution over the values of a hidden variable, $H$. In probability notation, a discriminative model provides a direct formulation of $p(H \mid X)$, the distribution of possible hypotheses given the observed data. Discriminative models have certain advantages. For example, once a neural network or support vector machine has been trained to perform a task, the performance of the task can be quite efficient computationally. However, the discriminative approach has not yet proven successful for difficult machine perception tasks, such as recovering 3D structure from 2D video of a deformable object.

In contrast, generative approaches begin with a forward model of how the hidden variable (the value to be inferred) would generate observed data. This is useful in situations for which the problem of how observations are generated from causes is better understood than the problem of how causes are inferred from observations. For instance, 3D computer graphics, the processes that are used in computer animation and video games to produce a realistic 2D image from a known 3D model, are much better understood than the inverse problem of inferring the 3D scene that produced an observed 2D image. The generative approach leverages our knowledge of the forward process by asking: according to my model of how observable patterns are generated, what hidden cause could have produced the information observed? The generative approach enables us to leverage the theoretical knowledge and specialized computing hardware that we already have for doing 3D animation, to help us solve the inverse problem of recognizing 3D structure from 2D scenes.

A probabilistic generative model provides an explicit probability distribution
\( p(H) \), called the prior distribution, over the possible values of the hidden variable. The prior represents internal knowledge that the system has before any data are observed. Lack of knowledge is modeled as an uninformative (uniform) prior distribution, expressing the fact that we may not have a priori preferences for any hypothesis. In addition, the generative model provides the likelihood function \( p(X \mid H) \), the distribution over the values of the visible variable given the value of the hidden variable. Together, these provide a model of the joint distribution of the hidden and observed variables:

\[
p(H, X) = p(X \mid H)p(H).
\]

If we have a probabilistic causal model (i.e., a generative model), we can use Bayes Rule to infer the inverse model, the posterior distribution \( p(H \mid X) \), which represents the probabilities that each of the various causes \( h \) could have produced the observed data \( x \):

\[
p(H \mid X) = \frac{p(X \mid H)p(H)}{p(X)}.
\]  

Bayes Rule (I.1), also known as Bayes Theorem, is named for Reverend Thomas Bayes, a Presbyterian minister in 18th century England. Figure I.1 shows a portrait of Bayes as well as a photograph of his final resting place.

Early approaches to computer vision utilized models that did not explicitly represent uncertainty, which often resulted in a lack of robustness to natural variations in data. More recently, the computer vision community has embraced probabilistic models, which contain explicit representations for uncertainty. Rather than simply representing a single value for a quantity of interest, random variables represent a probability distribution over all possible values for that quantity. Uncertainty in the posterior distribution indicates how certain the system is of its conclusions. The posterior distribution expresses not only what the system knows, but also what it does not know.

Probabilistic models are crucial when multiple opinions, each with different levels of uncertainty, need to be combined. Suppose two different systems each estimate a different value for the same variable. If the systems do not provide any measure of their relative certainties, it can be difficult or even impossible to combine the two estimates effectively. Intuitively, we should perform some sort of weighted average, but we have no sound basis for determining the weights to give each system’s estimate. In contrast, probability theory tells us how to combine distributions in an optimal way. Probability theory is the glue that allows information from multiple systems to be combined in a
I.1 Overview of the thesis research

I.1.1 G-flow: A Generative Probabilistic Model for Video Sequences

Our overall approach in Chapter II is as follows. We use the physics of image generation to propose a probabilistic generative model of video sequences. Noting that the model has a distinct mathematical structure, known as a conditionally Gaussian stochastic process, we develop an inference algorithm using techniques that capitalize on that special structure. We find that the resulting inference algorithm encompasses two standard computer vision approaches to tracking, optic flow and template matching, as special cases. This provides a new interpretation of these existing approaches that suggests the conditions under which each is optimal, and opens up a related family of principled manner. In order to integrate systems, then, it is invaluable to have an explicit representation of uncertainty, which generative approaches provide naturally.
Visual Tracking of 3D Deformable Objects  In Chapter II, we present a probabilistic generative model of how a moving, deformable 3D object such as a human face generates a video sequence. In this model, the observations are a time-indexed sequence of images of the object (the face) as it undergoes both rigid head motion, such as turning or nodding the head, and nonrigid motion, such as facial expressions. We then derive an optimal inference algorithm for finding the posterior distribution over the hidden variables (the rigid and nonrigid pose parameters, the appearance of the face, and the appearance of the background) given an observed video sequence.

The generative model approach is well suited to this problem domain. We have much prior knowledge about the system that can be incorporated with generative models more easily than with discriminative models. For example, we can incorporate our knowledge about the physics of the world: how heads and faces can move, as well as how three-dimensional objects form two-dimensional images on the camera’s image plane. We can even take advantage of 3D graphics-accelerated hardware, which was originally designed to facilitate implementation of the forward model, to help us solve the inverse problem. Because the generative model makes explicit its prior knowledge about the world, we can also learn this prior information, the parameters of the model, from examples. If we wish to change our assumptions later (e.g., from weak perspective projection to a perspective camera model), it is crystal clear how the forward model needs to change. In addition, explicitly specifying the forward model helps us to understand the nature of the problem that needs to be solved. Deriving an optimal inference algorithm for the generative model can shed new light on existing approaches, and can provide insight into the types of problems the brain might need to solve in order to accomplish the same task.

One of the great advantages of generative models over black-box models, is that the assumptions that our generative model makes about the world are stated explicitly (rather than incorporated implicitly). This enables formal consideration of how relaxing the assumptions would affect the optimal inference procedure. In addition, not only is it often easier to alter a generative model to accommodate changing circumstances or changing assumptions, but it is also often easier to combine two generative models into
a single model than it is to combine two discriminative models.

Current systems for 3D visual tracking of deformable objects can be divided into two groups: template-based trackers whose appearance (texture) models are constant over all time, and flow-based trackers whose appearance (texture) models at each video frame are based entirely on the image of the previous frame. Flow-based tracking methods make few assumptions about the texture of the object being tracked, but they require precise knowledge of the initial pose of the object and tend to drift out of alignment over time. The appearance information in a flow-based model is only as good as its alignment in the previous frame. As a result, the alignment error builds over time, which can lead to catastrophic results.

In contrast, template-based approaches are more robust to position uncertainty. However, template-based trackers require good knowledge of the texture appearance of the object, and are unable to adapt when the object appearance changes over time (e.g., due to changes in lighting or facial expression). In short, flow-based trackers are good at adapting to changes in appearance, but their memory of the object’s appearance is fleeting, which leads to growing alignment error. Template-based models are good at initial alignment and at re-aligning when they get off track, but they are unable to adapt to changing circumstances. The best of both worlds would be an appearance model in the middle of the conceptual continuum from template-based to flow-based, which could reap the benefits of both types of appearance model without suffering from their limitations.

As we describe in Chapter II, by defining a generative model and deriving an optimal inference algorithm for this model, we discovered that two existing approaches to object tracking, template matching and optic flow, emerge as special limiting cases of optimal inference. This in turn sheds new light on these existing approaches, clarifying the precise conditions under which each approach is optimal, and the conditions under which we would expect each approach to be suboptimal. In addition to explaining existing approaches, optimal inference in G-flow also provides new methods for tracking nonrigid 3D objects, including an entire continuum spanning from template-matching to optic flow. Tests on video of moving human faces show that G-flow greatly outperforms existing algorithms.
The IR Marks Data Set: Ground Truth Information from an Unmarked Face

Prior to this thesis research, there has been no video data set of a real human face that is up to the task of measuring the effectiveness of 3D nonrigid tracking systems. The reason is the difficulty of obtaining ground truth information about the true 3D locations of the points being tracked, some of which are located on smooth regions of the skin. The purpose of nonrigid tracking systems is to track the locations of face points when there are no observable markers on the face. Some researchers tracking rigid head motion [La Cascia et al., 2000; Morency et al., 2003] obtain ground truth rigid head pose during the collection of video test data, by attaching to the head a device that measures 3D position and orientation. Because the device only needs to measure the rigid pose parameters, and not the flexible motion of individual points on the face, it can be mounted atop the head without obscuring the facial features that the system observes.

Nonrigid tracking systems present greater challenges, however, because the points on the face that the system is to track must remain unobscured even as their positions are being measured. The traditional method for measuring 3D flexible face motion is to attach visible markers to the face and then label the positions of these markers in video taken by multiple cameras. Needless to say, the presence of visible markers during data collection would make it impossible to test the system’s performance on an unmarked face. Typically, developers of nonrigid face-tracking systems demonstrate a system’s effectiveness simply by presenting a video of their system in action, or testing on a more easily controlled simulated data set.

We developed a new collection method utilizing an infrared marking pen that is visible under infrared light but not under visible light. This involved setting up a rig of visible-light cameras (to which the infrared marks were not visible) for collecting the test video, plus three infrared (IR) cameras (to which the infrared marks were clearly visible), and calibrating all of the cameras both spatially and temporally. We collected three video sequences simultaneously in all cameras, and reconstructed the 3D ground truth information by hand-labeling several key frames from the IR cameras in each sequence. We use this data set to rigorously test the performance of our system, and to compare it to other systems. We are making this data set, called IR Marks, freely available to other researchers in the field, to begin filling the need for facial video with nonrigid ground truth information.
I.1.2 Diffusion Networks for automatic discovery of factorial codes

In Chapter II, we describe the G-flow inference algorithm assuming that the system already has a model of the 3D geometry of the deformable object. In Chapter III, we explain how such a model can be learned using a neurally plausible architecture and a local learning rule that is similar to Hebbian learning. Surprisingly, the problem reduces to that of developing factorial codes. In Chapter III, we derive rules for learning factor analysis in a neurally plausible architecture, then show how these rules can be used to learn a deformable 3D face model.

Diffusion Neural Networks  Recently, Movellan et al. [Movellan et al., 2002; Movellan and McClelland, 1993] have proposed a new class of neural net, the diffusion network, which has real-valued units that are updated in continuous time. Diffusion networks can be viewed as a generalization of many common probabilistic time series models (see Figure I.1.2). Whereas standard continuous-time, continuous-state recurrent networks are deterministic, diffusion networks are probabilistic. In diffusion networks, the internal state (pre-synaptic activation) of each unit is not simply the weighted sum of its inputs, but has an additional Gaussian noise term (a diffusion process) added. This adds an extra level of neural realism to the networks, because in real-world systems such as the brain, some noise is inevitable. Rather than trying to minimize or avoid the noise that is present in real systems, diffusion networks exploit this noise by making it an integral
part of the system. Knowing natural systems’ propensity for taking advantage of features of the environment, it is quite possible that the brain similarly exploits the noise in its internal circuitry.

A diffusion network is similar to a Boltzmann machine [Ackley et al., 1985; Hinton and Sejnowski, 1986] in that the output (post-synaptic activation) of each unit is not a deterministic function of the inputs to the unit. Like the Boltzmann machine, a diffusion network continues to evolve over time, never settling into a single state, but rather settling into an equilibrium probability distribution over states, known as a Boltzmann distribution. In fact, a diffusion network with symmetric connections can be viewed as a continuous Boltzmann machine: a Boltzmann machine with continuous (real-valued) states that are updated in continuous time.

One type of Boltzmann machine that has received special interest is the restricted Boltzmann machine (RBM). The units in a restricted Boltzmann machine are divided into two subsets, or layers: the visible layer and the hidden layer, which consist, respectively, of all of the units that represent observed variables, and all of the units that represent hidden variables. Inter-layer connections (connecting a hidden unit with a visible unit) are permitted in the RBM, but intra-layer connections (hidden-hidden or visible-visible connections) are prohibited. Recently, Hinton [2002] introduced a new learning algorithm, contrastive divergence learning, that can be used to train restricted Boltzmann machines much more efficiently than the traditional Boltzmann machine learning algorithm [Ackley et al., 1985; Hinton and Sejnowski, 1986].

In Chapter III, we consider linear diffusion networks, diffusion networks for which the unit activation function (the mapping from pre-synaptic to post-synaptic activation) is linear. We focus on linear diffusion networks that have the same architecture as the restricted Boltzmann machine: the units are partitioned into hidden and visible layers, with intra-layer (hidden-visible) connections but no inter-layer connections (no hidden-hidden nor visible-visible connections). We call these linear factorial diffusion networks (linear FDNs).

We prove in Chapter III that linear FDNs model the exact same class of data distributions that can be modeled by factor analysis, a linear Gaussian probabilistic model that has been used to model a wide range of phenomena in numerous fields. This is somewhat surprising, because the linear FDN generative model is a feedback network,
whereas the generative model for factor analysis is feedforward. The existence of a
neurally plausible method for learning and implementing factor analysis models means
that the brain could be capable of using not only factor analysis, but a host of nonlinear
and non-Gaussian extensions of factor analysis.

Not only do factorial diffusion networks share the same architecture as the re-
stricted Boltzmann machine, but like the RBM, they can be trained efficiently using
contrastive divergence. In Chapter III, we derive the contrastive divergence learning
rules for linear FDNs, and use them to learn the structure of 3D face space from a set
of biometric laser scans of human heads.

**Learning 3D Deformable Models with Contrastive Hebbian Learning** In
Chapter II, we derive and test a highly effective system for tracking both the rigid
and nonrigid 3D motion of faces from video data. The system uses 3D deformable mod-
els of a type that, as we describe in Chapter III, can be learned by a neurally plausible
network model with a local, Hebbian-like learning rule.

This does not prove beyond a reasonable doubt that the human brain uses 3D
deformable models to track faces and other flexible objects. Nonetheless, this dissertation
does demonstrate that the brain has both the motive (efficient, accurate on-line face
tracking) and the means (a neurally plausible architecture with an efficient learning
rule) to use flexible 3D models.

### I.2 List of Findings

The following list itemizes the main contributions of this dissertation to the lit-
erature.

**Chapter II**

- Describe a generative graphical model for how 3D deformable objects generate 2D
  video sequences.
- Derive an optimal inference algorithm, a type of Rao-Blackwellized particle filter,
  for inferring the time sequence of an object’s rigid pose (position and orientation of
the object in 3D world coordinates) and nonrigid pose (e.g., facial expressions), as well as the object and background texture (appearance), from an observed image sequence.

- Demonstrate that this filtering algorithm contains two standard computer vision algorithms, optic flow and template matching, as special limiting cases.

- Show that this filtering algorithm encompasses a wide range of new approaches to filtering, including spanning a continuum from template matching to optic flow.

- Develop an infrared-based method for obtaining ground truth 3D surface data while collecting video of a deformable object (a human face) without visible markings.

- Use this method to collect a new video dataset of a deforming face with infrared-based ground truth measurements, the first of its kind, which we are making publicly available to other researchers in the field.

- Evaluate the performance of the aforementioned 3D tracking system using this new data set, and demonstrate that it outperforms existing algorithms.

- Derive an expression for the second derivative of a rotation matrix with respect to the exponential rotation parameters. This new expression can be used in a wide variety of probabilistic applications in computer vision and robotics to obtain estimates of uncertainty.

Chapter III

- Explore a new class of neurally plausible stochastic neural networks, diffusion networks, focusing on the subclass of diffusion networks that has linear unit activation functions and restricted connections: linear Factorial Diffusion Networks (linear FDNs).

- Prove that this subclass of feedback diffusion networks models the exact same class of distributions as factor analysis, a well-known approach to modeling distributions based on a feedforward generative model.
• As a corollary, show that the factor analysis model factorizes in two important senses: it is Morton separable, and it is a Product of Experts.

• Demonstrate that principal component analysis (PCA) can be modeled by diffusion networks as a limiting special case of the linear Factorial Diffusion Network model for factor analysis.

• Derive learning rules to show that linear FDNs can be trained using an efficient, local (Hebbian-like) learning technique known as contrastive divergence [Hinton, 2002].

• Demonstrate the effectiveness of these learning rules by training a linear FDN to model a database of 3D biometric scans of human faces.

• Show that a neurally plausible model, the linear FDN, can learn a 3D deformable model of a human face, which could then be used by the system of Chapter II to track natural head and face motion from monocular video.