UNIVERSITY OF CALIFORNIA, SAN DIEGO

Perceptual Inference in Generative Models

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

in

Cognitive Science

by

John R. Hershey

Committee in charge:

Martin Sereno, Chair
Javier Movellan, Co-chair
Terrence Sejnowski
Virginia de Sa
Jochen Triesch
Mohan Trivedi
Irina Gorodnitsky

2005

The dissertation of John R. Hershey is approved, and it is acceptable in quality and form for publication on microfilm:

_____

_____

_____

_____

_____
Co-Chair

_____
Chair

University of California, San Diego

2005

To Araceli

There is little hope that one who does not begin at the beginning of knowledge will ever arrive at its end.
—Hermann von Helmholtz (1821–1894)

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I have depended on the help and support of many people, to whom I owe far more gratitude than I can possibly express here. Nevertheless, I would like to thank my parents, Phoebe and John Hershey, for instilling me with an active curiosity; although it may have nearly killed me as a teenager, it has been my best tool in research. I owe especially heartfelt thanks to Araceli Orozco, for bearing with me and practically carrying me along through the culmination of this research. Her companionship and encouragement made it all worthwhile.

This thesis sprang from a dialogue over the years with Javier Movellan, whose creativity, insight, and enthusiasm for learning, have been an inspiration to me. Javier gave such close guidance in the writing of the thesis introduction and conclusion that he should be listed as an author. Special thanks to the co-authors on the published papers in this thesis, for many wonderful collaborations and for allowing me to include work that was really a collective effort: Javier Movellan, Mike Casey, Hagai Attias, Trausti Kristjansson, Nebojsa Jojic, Tim Marks, Josh Susskind, and J. Cooper Roddey. Mike Casey, Matthew Brand, and Hagai Attias were excellent mentors to whom I owe a great debt not only for hosting me during internships at Mitsubishi Electric Research Labs (MERL) and Microsoft Research (MSR), but also for the treasure trove of ideas that slipped into my head while working with them. Thanks to Mike Casey, who guided my first attempts at single-microphone and audio-visual source separation. Thanks to Matthew Brand, who got me started on non-rigid face tracking: the work on that subject herein is built on the foundation of Matt's innovative work, and inspired by the fruitful discussions we had while I was in Boston. Hagai Attias schooled me in variational methods for graphical models. I owe him a great debt for his insight and ideas, as well as his help and support. Trausti Kristjansson, as a collaborator and as a friend, I thank especially for being an imaginative thinker, always willing to entertain and share the earliest hints of a new idea. Thanks to Jochen Triesch for an especially close reading of the dissertation.

I would also like to extend thanks to my friends in the Machine Perception Lab at UCSD for creating an exciting environment where brainstorming was a way of life. Special thanks Chris Córdova, Suzanne duFour, Fred Dick, Mark Rackers, Paul Mineiro, Jude Mitchel, Jim Stewartson, Aaron Dukes, Carrie Seros, Ezra Moore, Sue Whitfield, Holly Berman, Pepita Gonzalez and many others for their encouragement, friendship, and the good times along the way. To the rest of my friends and family, who have woven bits of themselves into the fabric of my being, I certainly could not have done it without you.

To my committee and the Department of Cognitive Science, both of which embody such a broad range of perspectives and disciplines, thank you for having me.

The research presented in this thesis has been published elsewhere by myself and my collaborators. The contents of Chapter Two are adapted from [40] which was published in Advances in Neural Information Processing Systems in 2000. The contents of Chapter Three are adapted from [63] which was published in Computer Vision and Pattern Recognition (CVPR) in 2004. Chapter Four is adapted from [64], which was published in Computer Vision and Pattern Recognition Workshop on Generative Models for Vision in 2004. Chapter Five is arranged in three sections: Section One was adapted from [51], which was published in IEEE International Conference on Acoustics, Speech and Signal Processing in 2004, Section Two was adapted from [53], which was published in the Proceedings of IEEE Workshop on Automatic Speech Recognition and Understanding in 2003, and Section Three was adapted from [38], which was published in Advances in Neural Information Processing Systems in 2002. Chapter Six is adapted from [37], which was published in IEEE International Conference on Acoustics, Speech and Signal Processing in 2004. The co-authors listed in these publications supervised and collaborated with me on the research which forms the basis of these chapters.

VITA

February 21, 1968	Born, Charlottesville, Virginia,
1992	B. A., *magna cum laude*, University of California, Los Angeles
1992-1994	Software Engineer, IBM Corporation Santa Clara California [1994-1996] Software Engineer, EEG Systems Laboratory, San Francisco, CA.
1996-2000	Teaching assistant, Department of Cognitive Science, University of California San Diego
2001	Research Intern, Mitsubishi Electric Research Laboratory, Cambridge MA.
2002	Lecturer, Department of Cognitive Science, University of California San Diego
2002-2003	Research Intern, Microsoft Research, Redmond WA
2004-2005	Visiting Researcher, Microsoft Research, Redmond, WA
2005	Ph. D., University of California San Diego

PUBLICATIONS

Tim K. Marks, John Hershey, J. Cooper Roddey, Javier R. Movellan "Joint Tracking of Pose, Expression, and Texture" (in press) in Advances in Neural Information Processing Systems 17, 2005

John Hershey, Trausti Kristjansson, Zhengyou Zhang "Model-Based Fusion of Bone and Air Sensors for Speech Enhancement and Robust Speech Recognition" ISCA Workshop on Statistical and Perceptual Audio Processing 2004

Javier Movellan, John Hershey, Tim Marks, and J. Cooper Roddey, "3D Tracking of Morphable Objects Using Conditionally Gaussian Nonlinear Filters" CVPR Workshop on Generative Models for Vision 2004

Trausti Kristjansson, Hagai Attias, John Hershey, "Stereo Based 3D Tracking and Scene Learning, employing Particle Filtering within EM" European Conference on Computer Vision (ECCV) 2004

Trausti Kristjansson, John Hershey, Hagai Attias, "Single Microphone Source Separation using High Resolution Signal Reconstruction" IEEE International Conference on Acoustics, Speech and Signal Processing, 2004

Javier Movellan, Josh Susskind, John Hershey, "Large-Scale Convolutional HMMs for Real-Time Video Tracking" Computer Vision and Pattern Recognition (CVPR) 2004

John Hershey, Hagai Attias, Nebojsa Jojic, Trausti Kristjansson, "Audio-Visual Graphical Models for Speech Processing" IEEE International Conference on Acoustics, Speech and Signal Processing, 2004

Trausti Kristjansson, John Hershey, "High Resolution Signal Reconstruction" Proc. of IEEE Workshop on Automatic Speech Recognition and Understanding, 2003

John Hershey and Mike Casey, "Audio-Visual Sound Separation Via Hidden Markov Models" in Advances in Neural Information Processing Systems 14, 2002

John Hershey and Javier R. Movellan, "Audio Vision: Using Audio-Visual Synchrony to Locate Sounds" in Advances in Neural Information Processing Systems 12, 2000

ABSTRACT OF THE DISSERTATION

Perceptual Inference in Generative Models

by

John R. Hershey
Doctor of Philosophy in Cognitive Science

University of California San Diego, 2005

Martin Sereno, Chair

Javier Movellan, Co-chair

We see and hear so freely that to the casual observer it is not obvious that perception would be such a difficult problem for modern science to understand. David Marr suggested that an understanding of perception requires analyzing the problems it solves along with the assumptions necessary for a solution. In this thesis I maintain that generative probabilistic models are a powerful tool to implement Marr's approach. In generative models one has to explitly encode the assumptions and goals of perceptual problems, whereas specific knowledge of the world is gleaned from the sensory data by learning within the model.

This thesis explores the use of generative models for understanding perception in audio-visual systems as well as in the individual modalities. The cocktail-party problem of single-channel sound separation is addressed using competing sound models, including a novel factorial model that unites pitch tracking and formant-based models. A convolutional hidden Markov model for video tracking performs exact inference in maps of object location, using a novel technique to make this inference tractable for extremely large hypothesis spaces. A model of non-rigid 3D tracking is presented in which some simple assumptions unify template matching and optic flow under the same framework. Finally, an audio-visual model brings together aspects of each of these models to exploit

cross-modal information for speech enhancement. Along the way, key benefits of generative modeling, such as the flexibility of inference, the "explaining-away" phenomenon, and the "problem-level" formulation of the models, are introduced and discussed in light of the research presented.

Chapter 1

Introduction

Is the problem that we can't see? Or is it
that the problem is beautiful to me?
—David C. Berman

Everything we learn about the world arrives via the senses. We act in the world on the basis of perception, and our very survival depends on its accuracy. Our perceptual abilities are the origin of all acquired truth, and the medium of all beauty. Yet how this exquisite gift of nature actually functions is a complete mystery. If seeing is believing, how is it possible that the light focused on our retinas is transformed into beliefs? How can we confidently step from stone to stone across a river, when a slight visual miscalculation could prove fatal? To understand how perception works would be a momentous achievement. Nevertheless, perception is often taken for granted. We see and hear so freely that to the casual observer it is not obvious that perception would be such a difficult problem for modern science to understand.

To researchers of machine perception, who seek to understand perception well enough to implement it in a machine, our dazzling perceptual abilities are an inspiration as well as a humbling existence proof. With every doubling of the

computational power of available computers we get closer to an exciting time when computers will be able to perform as many raw computations per second as the human visual system. When that time comes, and it may be within our lifetimes, will we know enough about the problems of perception to implement a worthy artificial perceptual system? Can we harness this computational power to help us understand perception?

David Marr (1945-1980) [60] argued that understanding will come by treating the computation of perception as an empirical science: we must propose principles and computationally test hypotheses about the nature of perceptual problems. In other words, if we are to understand perception, we will have to analyze the assumptions and constraints necessary for perception, implement these assumptions in a computational system, and test this system with real sensory input. Given the complexity of perception, however, it seems likely that in such an analytical scientific approach, important aspects of the problem may escape our analysis.

Perhaps what is overlooked by an analytical approach may be found in the statistics of natural sensory data. Moreover, adaptation to these statistics is likely to be a vital component of perception, and thus any approach to perception must take learning seriously. Computational learning models have the benefit that they can adapt to the statistics of sensory data. A central tenet of this thesis is therefore that whereas we may aspire to a *problem-level* understanding of perception as envisioned by Marr, we are also obliged to extract information about the statistics of the sensory data via learning systems. To the extent that the two processes can be combined, we may be able to accumulate problem-level knowledge by proposing assumptions in a framework that allows for adaptation, studying what is learned by the system, and building new assumptions, based on what we have learned, into the next iteration.

Probabilistic models are an attractive choice for such a framework because they permit knowledge of the problem domain to be flexibly encoded in their assumptions. The assumptions, in turn, yield optimality principles for learning and

inference. This thesis explores the use of such models for understanding perception in audio-visual systems as well as in the individual modalities. In the chapters ahead, a history of psychological theories of perception and developments in machine perception is used as a backdrop for illustrating the need for a probabilistic problem-level approach. Each self-contained chapter in the thesis presents published work in which probabilistic models shed some light on perceptual problems. The main ideas of each chapter are summarized in an overview, at the end of this introduction. The conclusion highlights the main contributions of the research.

1.1 Theories of Perception

1.1.1 Helmholtz and Constructivism

The development of optics and the subsequent understanding of the formation of retinal images led to a fundamental problem in perception: how do we obtain knowledge of objects and their form from the two-dimensional retinal images? The invention of the stereoscope early in the 19th century led to a further question: how does stereoscopic vision allow us to see structure that is not apparent from a single view alone? Hermann von Helmholtz (1821-1894) studied the nature of image formation and construed this problem as *unconscious inference* about the state of the world on the basis of raw sensory signals and knowledge of those signals gained through interaction with the world. The three-dimensional structure of the scene, and the identity of the objects therein are not provided directly in the retinal image, but require perceptual inference to interpret. The following passage concerning the perception of illusions captures the essence of his views[95] p. 307):

> *We always believe that we see such objects as would, under conditions of normal vision, produce the retinal image of which we are actually conscious.* If these images are such as could not be produced by any normal kind of observation, we judge them according to their nearest resemblance; and in forming this judgment, we more easily neglect the

> parts of sensation which are imperfectly, than those which are perfectly, apprehended. When more than one interpretation is possible, we usually waver involuntarily between them; but it is possible to end this uncertainty by bringing the idea of any of the possible interpretations we choose as vividly as possible before the mind by a conscious effort of the will.

Thus, according to Helmholtz, perception selects the most likely state of the observable surroundings — the *scene*— given the sensory signal, and empirical knowledge of likely scenes.

A modern formulation suggests a Bayesian approach to vision in which we choose the parameters of the scene θ that maximize $p(\theta|x)$, where x is the sensory image [69]. If we encode the knowledge of the scene in a *prior probability* $p(\theta)$ and the resemblance of the sensory signal to that generated by a given scene in the *likelihood* of the sensory signal given the scene, $p(x|\theta)$, then we can compute the desired *posterior probability* $p(\theta|x)$ using Bayes' rule:

$$p(\theta|x) = \frac{p(x|\theta)p(\theta)}{p(x)}. \tag{1.1}$$

Such models are termed *generative* because they are defined in terms of observations generated from hidden variables. The likelihood term $p(x|\theta)$ can be seen as addressing a *synthesis* problem: Given the parameters θ of a scene it tells us what images are likely to be generated. Computing the posterior then solves an *analysis* problem: given an image the posterior tells us what scenes are likely to have generated the image. Having ideas about how scenes produce images helps us develop reasonable formulations of $p(x|\theta)$. Using Bayes' rule allows us to convert a synthesis problem into an analysis problem by computing $p(\theta|x)$. The computation of $p(\theta|x)$ is known as *inference* in probabilistic models. How one formulates such a model and finds the θ that maximizes this posterior probability, and what such methods can tell us about perception, is the subject of a *generative model* approach to perception. This formalism is described in more detail in Section 1.3.

Helmholtz' views are associated with the *constructivist* view of perception: the brain constructs representations of the world to explain sensory input.

Constructivism is one way to describe the rich three-dimensional structure we see, and the order we seem to impose upon simplistic two-dimensional sensory input such as the bistable three-dimensional structure we infer from a Necker cube line drawing.

The emphasis Helmholtz placed on the active role of the observer have a modern flavor ([96] p.183):

> We are not simply passive to the impressions that are urged on us, but we *observe*, that is, we adjust our organs in those conditions that enable them to distinguish the impressions most accurately. Thus, in considering an involved object, we accommodate both eyes as well as we can, and turn them so as to focus steadily the precise point on which our attention is fixed, that is, so as to get an image of it in the fovea of each eye; and then we let our eyes traverse all the noteworthy points of the object one after another.

He stressed the contingency between vision and other modalities such as touch and motor control in calibrating the relationship between our senses, via his famous experiments in which prismatic spectacles shifted the retinal image to one side and visual-tactile contingency caused adaptation of the visual sense of space.

Helmholtz also argued for interactive multi-modal perception in learning to see ([95] p.304):

> The meaning we assign to our sensations depends upon experiment, and not upon mere observation of what takes place around us. We learn by experiment that the correspondence between two processes takes place at any moment that we choose and under conditions that we can alter as we choose. Mere observation would not give us the same certainty...

He goes on to describe the infant developing to the stage where it turns a toy in front of its eyes to discover all of the perspective views it affords.

1.1.2 The Gestalt school

The *Gestalt* approach, led by Wertheimer, Köhler, and others, rebelled against the reductionism and empiricism of their predecessors. The structuralists such

as Wundt had tried to explain perceptual learning as a process of accreting associations between individual "sensory atoms" to form a representation of objects [69]. In contrast, the Gestalt school emphasized a *holistic* approach to vision, maintaining that the configuration of the parts of an image gives rise to a percept that cannot be reduced to the properties or associations of the individual parts. They formulated general principles underlying visual perception motivated by the idea of an innate *energy minimization* principle: the brain chooses a perceptual organization of sensory data that minimizes the representational energy required to account for them. The energy minimization principle was also called the *law of Prägnanz*: if a signal is consistent with a number of different groupings of an array of features, all other things being equal, the simpler grouping is preferred. Since there are arbitrarily many arrangements of features into groups given a particular image, these laws were required to choose the particular grouping that we tend to see.

The law of Prägnanz was seen as the unifying principle behind a number of individual constraints by which features would tend to be grouped together, such as *proximity*, *similarity*, *continuity*. Thus, for example, according to the *law of proximity* features that were close to each other tended to be grouped together, and likewise with similarity and continuity. Similarly, the *law of common fate* held that things that are moving in the same way tend to be grouped together. A higher-level principle of organization, the *law of familiarity*, maintained that features that form the shape of familiar objects tend to be grouped together. A major drawback of this approach was that the perceived grouping was subtle and subjective. It was also difficult to formulate the relative strength of the various laws. Ultimately, without a computational framework such a theory could not provide any real predictive value.

A modern formulation of Gestalt perception can be framed probabilistically. Minimizing an energy looks like maximizing a log probability and we can easily imagine a prior probability distribution that prefers simpler groupings of elements according to the energy function. A likelihood function would be required to

select the features that correspond to the image, and while the Gestaltists never defined how this was to be done, feature-based representations of local image areas in computer vision are now commonplace. In such a formulation, inferring the grouping would amount to computing a posterior distribution over different grouping hypotheses. A probabilistic formulation is nice because it provides a means for either specifying or learning the constraints implied by the grouping laws, and it provides a language for representing ambiguity or uncertainty among possible grouping hypotheses given an observation.

Once we have cast both Helmholtzian perception and Gestalt perception into a probabilistic framework, then we can compare them on an equal footing. The main difference seems to be the types of problems they tried to solve. In a probabilistic formulation these different problems would lead to different assumptions about the priors and likelihood function. The Gestaltists focused on the segmentation problem, or the problem of grouping together the sensations from different parts of the retinal image into separate objects. For the grouping problem, an approach with a Gestalt character would have a likelihood function that assigns greater likelihood to feature representations that better reproduce the observed image, and the prior over grouping hypotheses would be defined via interactions between the local feature representations, such that greater prior probability is assigned when there is greater similarity, proximity, and so on, between elements assigned to the same group. Helmholtz was more concerned with inferring the three-dimensional (3D) structure of the world. For a model reminiscent of Helmholtz, hypotheses would be 3D interpretations of the scene. We might define a likelihood function that assigns greater likelihood to 3D representations that more accurately reproduce the observation when projected back to the image. A prior probability distribution could be defined over likely 3D configurations, perhaps based on knowledge of the world. This prior combined with the likelihood function, would be used to compute a posterior distribution over 3D scenes. Seen from the perspective of this formulation, the two different approaches do not seem necessarily incompatible, but rather might be complementary. That

is, the problem of analysis of patterns and segmentations of surface features at one level complements the problem of inferring the three-dimensional structure of the world, in the sense that the two problems might be solved simultaneously using somewhat different prior knowledge in the solution of each one.

1.1.3 Behaviorism

Both the structuralist and Gestalt schools relied on a methodology consisting of introspection, in which trained subjects tried to report not just what they saw, but also the internal processes of seeing. This phenomenological approach resulted in subjects who would report introspective data that just happened to confirm the prevailing theories of the labs in which they were trained [80].

The *Behaviorism* movement of Watson, Thorndike, and (later) Skinner rebelled against these methodological problems and banished introspection along with any theory positing such internal states as beliefs, goals, or representations, deriding them as "folk psychology," "mentalism," or at their most charitable, "intervening variables." Instead the brain was treated as a *black box*: only objective stimuli and response characteristics were to be measured and related by theories. Behaviorism became the dominant psychological movement in the United States from about 1920 to 1960 and resulted in the development of mathematical theories of reinforcement learning which would later contribute to computational work. Behaviorists, however, completely neglected to account for how organisms interpret the contents of their stimuli. So for instance although they posited that an object could act as a stimulus they failed to address the question of how the object was to be segmented and recognized.

1.1.4 Gibson's Ecological Approach

The *ecological* approach, advanced by James J. Gibson, advanced a new emphasis on *ecological validity*. Gibson rejected the simplistic stimuli used in Gestalt and structuralist approaches, instead arguing that the characteristics

of the environment under which the organism evolved – its ecology – were of primary importance. Ecological validity also emphasized, as did Helmholtz, the importance of the organism as an active observer moving in the environment. Unlike Helmholtz, Gibson felt that motion in the environment constrained the problem of vision enough that inference would not be necessary. The theory was essentially nativist, however Gibson felt that only general perceptual strategies were inherited, rather than specific visual knowledge. Via evolution, however, the nativist and empiricist positions both agree on one thing: the structure of the environment determines what perception is. Whereas Gibson has been criticized for underestimating the difficulty of perception [60, 34], the ease of human perceptual performance implies that a satisfactory solution exists. What is unknown is what assumptions are necessary to find such solutions, and ho the inference process works.

Gibson argued for *direct perception* that was unmediated by mental processes, three-dimensional representations, or other "intervening variables." Taking an armchair approach, Gibson analyzed how "direct perception" might be achieved by looking at patterns in the environment. For example, in optic flow, a concept introduced by Gibson to describe local motion in image coordinates, the time to impact of an observer with a surface can be calculated directly without knowing the distance to the surface, or the relative velocity between the observer and the surface. Thus a motor system could be activated to avoid or engage the surface without any need to estimate distances. He called this type of percept an *affordance* to emphasize that it directly mediated action, rather than establishing a three dimensional representation. The idea originated in the Gestalt idea of the *physiognomic* character of perception which Kurt Koffka explained thus: "To primitive man, each object says what it is and what he ought to do with it: a fruit says 'eat me,' water says, 'drink me,' thunder says 'fear me,' and woman says, 'love me.'" ([69], p.410) Gibson elevated it to the primary mode of perception.

Affordances represent a key deviation from the constructivist tradition. Whereas the constructivists held that when we look at a chair, we first infer

that it is a chair that we are seeing, and then recall that chairs are for sitting, and then sit if we so desire. In the ecological approach we would directly sense that the surfaces of the object were oriented such that one could sit on them, and sitting would then be liable to occur. Gibson thus questioned the traditional assumptions about what representation was computed, and emphasized the role of perception in mediating visually-guided action, and the interactive role of the moving observer in perception. He supposed that direct perception of affordances involved "resonating" to the invariant information in the world, an idea that would form the basis of pattern recognition approaches to vision. Gibson failed, however, to specify how this process worked, how his primitives such as optic flow and texture gradients were to be computed, and converted into affordances. Nevertheless the concept of affordances continues to resurface in machine perception in approaches that deal with visually guided action.

1.1.5 Information Processing

Meanwhile other forces were bringing back a cognitive viewpoint, in which internal states and processes are important explanatory hidden variables. The development of digital computer introduced a new computational way of thinking about internal states and mechanisms. Simultaneously in psychology, Kenneth Craik's *The Nature of Explanation* "put back the missing step between stimulus and response, positing that an agent needs a model of its environment and possible actions in the form of beliefs and goals" ([80], p. 21). According to Craik, the stimulus must be translated into an internal representation, the representation may be manipulated by cognitive processes to derive new internal representations, and these representations are translated into action. This formulation, along with the specification that these processes were essentially computational, constitutes the *information processing* or *cognitive* view of perception which has become the dominant paradigm in perceptual theory.

This cognitive viewpoint carries a burden in the form of the *mind-brain*

problem: to define such internal states as beliefs and goals, without resorting to an unscientific dualism, one must reduce mental states to some properties of the brain. The prospect of interpreting the brain states in terms of the functional significance of the computation they perform might not seem at first like a contentious issue. The problem is that the same computation can in principle be done in a different physical implementation, such as a computer, so mental states perhaps cannot be uniquely reduced to brain states, and thus can only be uniquely described at the level of their computational function. Thus there must be some properties of the brain, at some microscopic level of analysis, that are sufficient but not necessary for the functional activity of the mind.

The dissociation between mental states and brain states created a kind of *software-hardware* dualism had the effect of polarizing research. Psychologists and computer scientists were insulated from the need to take the brain into account, whereas others felt that mental states could never have a scientific basis in the brain, and so folk psychology should simply be eliminated from science. At the same time despite this philosophical rift, the computational viewpoint eventually brought the fields of psychology, neuroscience, and computer science into cooperation with each other, and the field of *cognitive science* was able to begin synthesizing evidence from all three perspectives. Cognitive scientists now had an effective methodology: they could in principle use computers to implement their theories, test them on real data, and compare the results with neuroscience or psychology. On the other hand, the initial unification was achieved by thinking of the brain as the computer of the era. The brain is teaching us now that this was a very limited view of what a computer can be.

1.2 Machine Perception

Machine perception, the study of computational aspects of perception, broadly involves the integration of the senses and the understanding of one modality in terms of another. The work presented in this thesis strives for this

broader interpretation. Historically, however, most progress has been made in modality-specific communities such as computer vision and speech recognition. A historical review focusing on computer vision and speech recognition will therefore set the stage for the contributions of this thesis.

1.2.1 Early Artificial Intelligence

The development of modern computers by John von Neumann (1946), the work of Warren McColloch and Walter Pitts (1943) on a computational abstraction of neurons, the founding of the field of *cybernetics* by Norbert Wiener (1947), and the formulation of neural learning rules by Donald Hebb (1949), introduced the world to a new way of thinking about the brain.

A science of *artificial intelligence* (AI) was founded with the goal of instantiating the powers of the human brain in a programmable computer. The first artificial intelligence projects took advantage of the symbolic architecture of programmable computers, and demonstrated breakthroughs with problems such as theorem proving, problem solving, and playing simple games such as checkers [80]. These symbolic approaches involved a system of formal symbols representing knowledge, along with rules for manipulating them to form deductions, and methods of applying the rules to search the space of possible deductions. That a computer could prove mathematical theorems and play games this way was a dramatic result at the time, since these were some of the things that for humans seem to take deliberate mental effort. However these early successes spelled trouble for machine perception, because the style of sequential, discrete rule-based computation that framed all their problems was incompatible with the more continuous signal-processing approach that now dominates the field.

Early AI approaches to computer vision adopted the symbolic approach and applied it the problem of inferring three-dimensional (3D) structure from two-dimensional (2D) images in extremely simple environments consisting of geometrical objects such as blocks [80]. The first stage of such systems analyzed

features such the location and orientation of edges, and the types of vertices they form with each other. These were converted into a symbolic representation of a line drawing, and symbolic search techniques were employed to deduce the possible configurations of simple three-dimensional shapes that could give rise to the line drawings. This symbolic AI approach failed to become a viable machine perception paradigm. Its reliance on symbolic logic and discrete rules worked well for highly constrained toy domains, but did not scale well to a continuous multidimensional sensory domain married to a wild and wooly reality. Despite the development of many edge-detection schemes, line drawings could not be reliably inferred from real images, and real images do not consist of simple geometric shapes for which edges were an appropriate description. In general, the combinatorial explosion of possibilities in a realistic world made for a spectacular failure of the early symbolic approach to do anything useful outside of extremely simplified domains.

In their influential book *Pattern Classification and Scene analysis* [20], Richard O. Duda and Peter E. Hart advocated a two-pronged approach to machine perception emphasizing probabilistic methods at a time when probability was neglected by the AI community. They saw the need for both a *descriptive* approach which described the geometric structure in a scene, as well as a *classification* approach based in statistics. They outline a classification approach built on a well-understood Bayesian decision-theoretical framework. Statistical techniques had long been the tools with which scientists made inferences about underlying processes in the world from their measurements. Duda and Hart discussed how an array of statistical techniques, from supervised learning of classifiers to unsupervised clustering and dimensionality reduction, could be useful for perceptual inference on sensory data. An important contribution was showing how particular descriptive mechanisms such as edge detectors could be seen as arising from a set of probabilistic assumptions. They also described how incorporating knowledge of projective geometry into a system would help to determine the *projective invariants*, that is, properties of objects that remain unchanged from picture to picture. However they stopped short of linking this

idea with their probabilistic framework, and ultimately they faced deep problems of three-dimensional shape representation that persist into the present.

1.2.2 Marr's Levels of Understanding

David Marr, a theoretical neuroscientist and a computer vision researcher, is better known for his thoughts about how to conduct vision research than for his computer vision work. He was instrumental in elevating computational approaches to vision to a science and uniting that science with the study of the brain under his framework of *levels of understanding*. Marr argued that *ad hoc* approaches to computer vision and *descriptive* approaches to visual neuroscience did not explain *what* was being computed and *why* it was a good thing to compute, only *how* computation was performed. Therefore there were no principles for determining the purpose of a particular computation, or judging its optimality to that purpose. He posited what he called a "computational level of understanding", that specified the computational function of a system: what problem is solved, under what assumptions. The terms *problem level* and *function level* perhaps better capture the intent of this computational level. Beneath this problem level he left intact to some extent the old software/hardware dualism in the form of a "representation and algorithm" level (software), which specified abstractly what method was used to solve the problem, and a "implementation" level (hardware) which specified how the representation and algorithm were physically realized in a given system. For convenience, we refer to these two levels collectively as the *mechanism level*.

Marr felt that a perceptual system such as the brain's must be understood at all three levels of understanding. However he emphasized the primary importance on the problem level of understanding ([60], p. 27):

> Trying to understand perception by studying only neurons is like trying to understand bird flight by studying only feathers: It just cannot be done. In order to understand bird flight, we have to understand aerodynamics; only then do the structure of feathers and the different shapes of birds' wings make sense.

How tightly coupled the different levels of understanding are when applied to the brain, and how likely we are to achieve a problem-level understanding of the human visual system has been a matter of debate. Likewise, the boundary between the problem level and the mechanism level is difficult to define. Nevertheless, the emphasis on understanding the function of a particular computation in terms of the problem it solves, has been a crucial challenge to both computer vision and neuroscience and their cooperation in studying perception. Whereas the problem level of understanding has been influential over the years, the problems we formulate to understand perception have changed.

Marr also formulated some important general principles, which were conceptually orthogonal to his levels of understanding. He emphasized a modular approach to understanding vision, with the modules divided according to the types of problem that were simple enough to be understood in isolation. For example, the random-dot stereograms of Julesz isolated stereo disparity from other cues that might be obtained from forms and objects in an image, showing that we can robustly perceive depth from disparity cues alone. For Marr, this meant that we can study the problem of stereo vision by itself. He elevated this to a *principle of modular design*, arguing that such a design allows improvements to be made, whether by a human designer or by evolution, without risking damage to other systems. This type of approach does not rule out interaction between the modules, but nevertheless insists that our understanding is aided by a modular organization of a complex general problem into a collection of simpler specific problems. Marr points out that modularity assumptions had already had a huge impact on the progress of machine vision prior to his formulation of the principle ([60], p.102): "Information about the geometry and reflectance of visible surfaces is encoded in the image in various ways and can be decoded by processes that are almost independent. When this point was fully appreciated it led to an explosion of theories about possible decoding processes," for specific problems such as stereopsis, structure from apparent motion, shape from shading, and so forth.

Marr also promoted some practical principles. He once suggested an "inverse

square law" for theoretical research which stated that the value of a study was inversely proportional to the square of its generality [21]. Thus his emphasis was on understanding specific perceptual problems, rather than general theories of inference.

The *principle of graceful degradation* [60] stated that whenever possible a system should be robust with respect to corrupted input information. Although aimed at the algorithm and representation level, this principle specifies the conditions and assumptions under which a given system should operate, so it seems to apply as well to the problem level. At a practical level the insistence on robustness in natural conditions is an important defense against *ad-hoc* theories of perception.

The *principle of least commitment* [60] admonished against resorting to time-consuming "hypothesize-and-test" strategies unless the problem is difficult enough to warrant such an approach. An analogous principle seems fitting for the computational level: one should not break a simple problem into difficult intermediate problems. This principle seems to balance the modularity principle: too much modularity places a burden of commitment to intervening representations.

Marr, like many of his predecessors, thought that the best route to viewpoint- and lighting-independent object recognition was to first infer the invariant 3D structure of the world, and then use this structure to recognize objects. This *scene-analysis* or *descriptive* approach was the defining theme of classical computer vision. Along with Marr and his generation came a variety of classical mathematical vision approaches to specific aspects of scene analysis problems, or the 2D-to-3D problems generically referred to as *shape-from-X*, such as shape from shading, stereo depth perception, and shape from motion. In these approaches, unlike the early AI approaches, the objects could typically be composed of continuously varying surfaces, and typically representations were continuous-valued functions. The properties of image formation were carefully studied and incorporated into algorithms.

In this approach, a variety of simplifying assumptions were often made on the image formation process as well as the lighting and surface reflectivity properties. For instance, often the lighting was assumed to be diffuse, the surfaces were assumed to reflect light with equal intensity in all directions (that is, they are *Lambertian*), or the surface was assumed to reflect light with equal intensity (have a constant *albedo*) at all points, as a function of the angle of illumination. In addition the individual problems were often broken into sub-modules: for instance, shape-from-motion depended on motion, which required solving the *correspondence problem*, or computing what points correspond in the 2D image over time. The correspondence problem, in turn, depended on the selection of which points were "interesting" in the sense of being easy to track, and this selection became another module. Because these modules were typically designed to operate independently, in accordance with Marr's principle of modular design, their interfaces with each other had to be assumed. These simplifying assumptions, as well as the reduction of the problem to specific modules and sub-modules, allowed theoretical progress to be made.

However there were problems with this approach: perhaps researchers focused too much on modularity principles, and not enough on least commitment and graceful degradation principles. The methods studied were still brittle and required simplified conditions, the models could not learn from data or adapt to different conditions, the modules were not easy to combine to jointly estimate 3D from 2D using a combination of the cues, and it was not clear how to use the inferred 3D structures for recognition in realistic scenarios, where objects are irregularly illuminated, partially occluded, and surrounded by clutter.

Marr's analytical approach also seemed to foster rationalizations for certain mechanisms, such as his zero-crossing implementation of edge detection, and subsequent symbolic characterization of the zero-crossing representation. This analytical approach relied on assumptions about the goals and tasks of perception and its modules, which biased the resultant computational theory.

For example, so-called *early vision* is often posed as a problem of inverse optics,

with the goal of inferring the position and form of surfaces, and perhaps object boundaries, in a three dimensional scene from a set of two dimensional images. Similarly the problem of *early audition* is posed as a *cocktail-party problem*, or *auditory scene analysis* problem (inverse acoustics), in which the goal is to separate and perhaps locate representations of all of the acoustic sources in an area given signals from one or more ears or microphones.

However, it is an empirical question whether some tasks can be done directly without first locating and segmenting, and later pattern recognition techniques were able to avoid these inverse problems by operating directly on features to some extent. Marr never addressed the possibility that object recognition could help with 3D scene analysis, rather than the other way around. Marr also completely overlooked the probabilistic perspective: he never mentioned learning or adaptation to data, and he clearly failed to realize that probabilistic methods are a very useful analytical tool to pursue his problem level analysis.

1.2.3 Neural Networks

The *neural network* approach, which had been on the sidelines since the beginning of the theory of computation, took center stage in the 1980s. Whereas earlier work on pattern recognition with neuron-like elements such as [76] had shown that learning could be done in a simple single-layered architecture, these architectures had their limitations [62]. The reason conventional von Neumann style computer architectures took over in the 1950s, whereas the neurally inspired architectures stagnated, was the extreme difficulty in programming a neural network. It wasn't until methods such as the *generalized delta* or "backpropagation" learning rule, along with advances in computer power, had made it possible to construct more complex nonlinear neural network simulations to perform a particular task. As a result the *parallel distributed processing* (PDP) and *connectionist* movements [26, 79] saw an explosion of research in a variety of directions.

In some cases neural network algorithms were derived from a problem-level analysis: a notable case is Marr and Poggio's stereopsis network, which was derived from a problem-level analysis of the assumptions of stereo vision that lead to unique solutions [61]. However, much neural network research typically focused on learned mechanisms rather than understanding function. This focus on mechanisms was partly a product of the emphasis on *neurally-inspired* methods that tried to take into account the constraints of computation in the brain, namely a large number of highly interconnected, slow processing units. Neural network researchers believed that the style of computation in the brain was the important thing, and understanding how computation worked within this style was important for understanding the brain. [79]. At the time, theories in terms of mechanisms were a welcome relief from decades of psychological theories phrased in opaque, irreducible terminology. However, ultimately it was difficult to focus on the problem level because it was hard to design a useful neural network that satisfied a set of problem-level assumptions. Thus there was a focus on learning from data, not so much because of an empirical stance, but rather for practical reasons.

Regardless of its cause, the necessity of learning in neural networks led to an influx of statistical pattern recognition theory into the neural network community. Whereas the neural-network approach has also spawned a field of computational neuroscience that operates in terms of physiologically plausible mechanisms, gradually the theoretical side of the neural network approach has evolved into a statistical learning approach. As an index of this, a tradition at a theoretical neural network conference, *Neural Information Processing Systems* (NIPS), has been to use a current learning algorithm to analyze the title words associated with article acceptance and rejection for comical presentation at the closing banquet. In recent years "neural network" was most strongly associated with rejection, whereas "graphical model," "Bayesian," and "support vector machine" — all statistical approaches — were associated with acceptance.

1.2.4 The Black-Box Pattern Recognition Approach

Marr described the application of pattern recognition to perceptual problems thus: "The hope was that you looked at the image, detected features on it, and used the features you found to classify and hence recognize what you were looking at. The approach is based on an assumption which essentially says that useful classes of objects define convex or nearly convex regions in some multidimensional feature space where the dimensions correspond to the individual features measured. ... It's not true, unfortunately, because the visual world is so complex. Different lighting conditions produce radically different images, as do different vantage points." ([60] p.340-341).

In the extreme this approach becomes the brute-force application of computing power and training data. Nearest neighbor classifiers, for instance, assign the class of the nearest training data point in some feature space to incoming test cases. As the size of the training set increases, test performance tends to increase. However, typically more data is required for good performance than one would like to provide, and each additional training example produces diminishing reductions in error. More importantly, little light is shed on the nature of the problem being solved by such an approach. Furthermore, the resulting systems tend to be brittle in that they serve a narrow purpose and fail to adapt when confronted with input collected under conditions that differ in any way from the training data. To some extent this is a result of the combinatorial explosion of possible data conditions that confront perception. In hearing, there is a vast array of noise, reverberation and intrinsic sound variations, and likewise in vision there are viewpoint, lighting, clutter, and object variations. Accounting for every combination of these is difficult in a system without a modular or combinatorial design.

Speech Recognition

An intermediate version of such an approach, however, was beginning to work in speech recognition. Speech recognition research had begun in earnest without

the distraction of the auditory scene analysis problem, or *cocktail party* problem, in which the goal was to separate and localize the different sounds in the environment, because these problems were considered too hard.

Instead researchers focused on directly recognizing speech in clean conditions: no interfering noise, no reverberation, no unknown microphone transfer function. After a period of trying many different approaches, standardized databases were established, and the field settled into a dominant paradigm. Feature sets were devised to be as invariant as possible to different pronunciations of a phoneme, such as pitch or vocal tract size. The speech signal was modeled using *hidden Markov models* (HMMs), which are probabilistic models consisting of a discrete hidden state, state dynamics defined by transition probabilities, and state observation probabilities defined over the features. The *forward-backward* algorithm was an efficient unsupervised learning algorithm for training HMMs [6].

A few decades of improving the databases, architectures, features, and language models, brought the speech recognition problem tantalizingly close to a usable level of accuracy *in clean conditions*. However, the better the performance of such systems became, the more data and internal states were required to produce small improvements in accuracy. On top of that, the addition of environmental noise proved to be a critical stumbling block, perhaps because the field matured using databases of clean signals, and then later attempted to make their techniques robust with respect to noise. By the time databases for noisy speech recognition were available, the field was entrenched in a narrow paradigm that was only suitable for clean speech.

Visual Pattern Recognition

The pattern recognition approach to vision had shifted gears from the traditional AI strategy of scene analysis, followed by recognition. It was difficult to infer 3D, and somewhat easier to throw large quantities of data at a learning algorithm and let it figure out what was invariant to changes in lighting and

viewpoint. Thus instead of Marr's division of a single scene analysis problem into sub-problems that could be analyzed independently, the pattern recognition approach divided research according to categorization tasks that could be learned independently: each task used different feature sets, training databases, and learning algorithms. Specifically the categorizations tasks had to do with direct recognition of so-called *high-level* properties of interest. So for instance one system would be developed to recognize the identity of a face independently of emotional expression, and another to distinguish the emotional expression of the face independent of identity, while both were intended to be invariant to lighting and clutter. Not only were such clearly related tasks not encouraged to interact, they often were not even investigated by the same researchers!

Whereas this pattern recognition approach appeared to be working for speech, in vision it was harder to control within-class variance, however it was hoped that this could be overcome with the right feature spaces and classifiers. For a while progress on real problems was hampered by flawed databases that did not properly handle these sources of variation. For instance the DARPA FERET database for face recognition, introduced in 1993, although it contained lighting variation, did not contain variation in the background, so that any system trained on it would have little applicability to real-world scenarios. In addition, training was typically done on *registered* images in which the object was in a canonical position. Thus in order to find objects, such a system would have to be applied to every conceivable scale, translation, and rotation of the area of interest in a larger image, which at a practical level was thought to be very time consuming. Recognition of an object from different viewpoints was typically still a difficult problem, even with these simplifications. These problems were ultimately remedied to some extent. Data was collected in which backgrounds were as random and varied as possible, and clever schemes for allocating resources allowed rapid scanning of detectors for objects at different scales and translations [94], and viewed from different angles relative to frontal [57].

Despite significant progress, the pattern recognition approach as it was typically

applied had some important drawbacks. For one thing, it failed to explain the principles implemented by the learned mechanisms, or what invariances had been learned. The resulting mechanisms were clearly more accessible than the brain: one could keep a complete record of the activity in any situation, limited only by data storage space. However such resulting mechanisms were still essentially a black box in the sense of obscurity. One had to do a post-hoc probe of such a system with different stimuli to see what different parts of it responded to. [1]

The difficulty was that although the data itself formed one part of the problem-level specification, the other part — the assumptions being made about the data — were missing. Marr described earlier neuroscience ideas "according to which the brain was a kind of 'thinking porridge' whose only critical factor is how much is working at a time." ([60], p.340) Similarly the *black box* pattern recognition approach can be caricatured as a kind of "learning porridge" formulation of perception that simply exposes a learning algorithm to perceptual data without manipulating its assumptions. The learning algorithms were capable of taking us straight from data to mechanisms, without pausing at a level where understanding was possible. So whereas practical progress was being made, progress in understanding perception amounted to heuristics about which feature sets, training sets, and learning algorithms seemed to work in a particular task.

Ultimately it seems difficult to turn the black box approach into a full theory of perception. At least in its simplest formulation where an image is presented to a recognizer and an object category is returned, the black box approach fails to account for our important descriptive abilities, in particular our visual and auditory scene analysis abilities which allow us to perceive and interact with objects and sounds that we have never seen or heard before, even in the midst of clutter and noise. In addition, it is difficult within the black box pattern recognition approach to incorporate knowledge, such as that of projective geometry. Thus it is difficult

[1] As an index of the opacity of this process, neural network researchers have sometimes had to resort to exploratory statistical analysis on the activity patterns of their systems to interpret what has been learned by the network [22].

to test assumptions that the model makes about how sensory data are generated by the world.

1.2.5 Summary of Traditional Approaches

In summary there have been two general approaches to machine perception: a descriptive approach of scene analysis, and an invariant classification approach. The viewpoint and lighting cause variance in video, and the acoustics and interfering sounds cause variance in audio. The descriptive problem addresses precisely the extraction of the invariant properties of the world. The traditional artificial intelligence goal was to first solve the descriptive problem, yielding a 3D representation of the scene, and then to categorize the objects based on their geometry and reflectivity. An *analytical* approach sought to accomplish this scene analysis by incorporating knowledge of projective geometry and surface constraints into scene analysis systems. Later, with developments in pattern recognition, an *empirical* approach alternatively sought to directly accomplish the classification of objects by extracting image features and learning functions of the features that were invariant to lighting and viewpoint. In these invariant pattern recognition problems, the goal was to directly classify an object into either broad categories, such as what type of objects they are (is it a face, or a shovel?), or narrow categories, such as what properties the object has (is it Javier's face, or Virginia's? is it frowning or smiling?). Each approach has its pros and cons: from the analytical scene analysis approach we get an understanding of the problem, at the expense of a brittle system whereas the empirical black-box approach takes advantage of the statistics of the data to find aspects of the problem that might not occur to us, at the expense of a clear understanding of precisely how the system works. Probabilistic methods, addressed in the next section, have recently been used in a way that seems to be closing the gap between the two approaches, while at the same time leading to an understanding of the problem.

1.3 Probabilistic Modeling

Probabilistic models can incorporate knowledge of the problem, as well as learning from data, and thus are good candidates for a unifying framework for computer vision. From the 1990s on such models have been rapidly gaining importance as they are applied to interesting problems of vision and hearing. Tracking of flexible objects using probabilistic models of dynamics became one of the most important trends emerging from the *active vision* movement [12, 43]. The addition of flexible 3D models and optic flow computations to such paradigms have shown how some very simple projective geometry could provide the crucial constraints that allow optic flow to track objects [88, 11, 64]. Principled methods of combining modalities, have been applied to multi-modal processing, in which simple dependency assumptions led to cross-modal self-supervised learning [7, 37]. Multi-object systems in which problem level specifications of occlusion in video or masking in audio have led to mechanisms for segmenting partially hidden signals [91, 28, 38, 51, 53].

1.3.1 Bayesian inference

Bayesian inference provides a framework for posing and understanding perceptual problems. In this framework prior knowledge of the scene θ is encoded in a *prior probability* $p(\theta)$, knowledge of how the sensor signal x is generated from the scene, and how to measure deviations from the expected sensor signals, is encoded in the *likelihood* of the sensory signal given the scene, $p(x|\theta)$, then we can solve the *inverse problem* by computing the *posterior probability* $p(\theta|x)$ using Bayes' theorem:

$$p(\theta|x) = \frac{p(x|\theta)p(\theta)}{p(x)} \tag{1.2}$$

$$= \frac{1}{z}p(x|\theta)p(\theta), \tag{1.3}$$

where $z = \int p(x|\theta)p(\theta)d\theta$ is a normalization constant that does not depend on θ. The term, $p(x|\theta)$, is a function of two variables, and has two different interpretations depending on which variable is allowed to vary. Holding the state θ constant yields the *observation probability distribution*, which is the probability of observations x. Holding the observation constant yields the *likelihood function*, which is the probability of the observation as a function of different states in hypothesis space. The likelihood is not a probability distribution as it does not integrate to unity: this normalization is the role of the denominator in Bayes' rule.

Such a model is generative in the sense that it defines a theory of how, given the state of the world, the sensor data is generated. Given this *forward model* inference then solves an inverse problem, thus translating synthesis into analysis. Because parameters of the distribution of a random variable can also be treated as random variables, learning and inference are unified. Inference usually refers to computing the distribution of some hidden variables after the other variables are observed, whereas learning usually refers to updating the parameters of the model. In generative models, parameters refer to hidden variables that have the same value across a set of samples, whereas ordinary hidden variables have a different value for each sample. Thus if samples arrive over time, we can think of learning and inference as referring to the same underlying process applied at different time scales.

An advantage of the generative model is that although we may encode some of our understanding of the modality into the model we still use learning to extract knowledge from the sensory input. Because of the probabilistic formulation, there are principled methods of inference and learning that ensure the optimality of the inference given the assumptions of the model. Such models can easily be extended, for instance, by adding temporal dynamics, or dependencies between modalities, in a principled way while maintaining optimality properties. The generative model also allows us to use the same model for a variety of inference tasks, such as reconstructing the signals from one object in the absence of the others, or inferring properties of each object, such as its location or appearance.

1.3.2 Graphical models

This flexibility of generative models stems from their formulation in terms of a complete *joint probability distribution* over all variables, which allows us to infer any part of the model from any other, unlike *discriminative* approaches in which the direction of inference is typically inflexible. The simple model above can be written:

$$p(x, \theta) = p(x|\theta)p(\theta) \tag{1.4}$$

from the definition of conditional probability. The joint distribution $p(x, \theta)$ is a complete description of a probability model. However, the factorization, $p(x|\theta)p(\theta)$, is a useful way to specify the same distribution in terms of a forward model. Models in which we write the joint distribution in terms of conditional probability functions can be depicted in the *directed graphical model* formalism (undirected graphical models are also used in other situations) [48]. In this formalism the random variables are represented by nodes, and conditional probability functions are represented using directed edges, where the direction from parents to a child indicates that the model specifies a conditional distribution of the child given the parents (see Figure 1.1).

A particular factorization of the joint probabilities implies a particular graph and *vice-versa*. Moreover since such factorizations imply independence and conditional independence relations, the graph makes the same statements about independence as the factorization. For example, the statement

$$p(x_n, \theta) = \prod_{n=1}^{N} p(x_n|\theta), \tag{1.5}$$

implies that the x_i are independent given θ and corresponds to the graph in Figure 1.2. Often we deal with a set of variables x_n that are independent and identically distributed (i.i.d.) given a parameter θ, such as independent samples of a random variable. Such i.i.d sub-graphs can be represented using a *plate* notation in which a box is drawn around the repeated nodes, as in Figure 1.3.

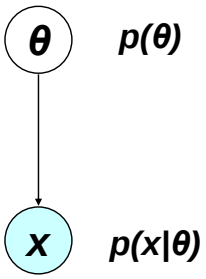


Figure 1.1: A simple directed graphical model. Observed variables are typically placed at the bottom of the graph and shaded.

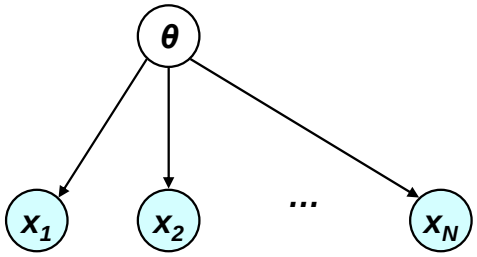


Figure 1.2: A graphical model depicting a series of random variables x_n that are conditionally independent given θ.

The key benefit of formalizing the probabilistic models as a graph is that it allows researchers to apply graph-theory to the solution of inference problems in more complicated models. The graph formalism allowed general solutions to be found for graphical models regardless of their specific structure. Thus, exact methods of inference, such as the *junction tree algorithm*, deterministic approximate methods such as *variational algorithms*, and stochastic methods such as *Markov chain Monte Carlo* (MCMC) and *Particle Filtering* methods can all be applied in general to many different models. Of course, one still has to find models in which the inference is effective and tractable using any of these methods [48].

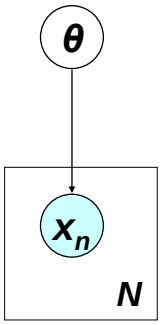


Figure 1.3: A graphical model as in Figure 1.3, but using a plate to represent that the variables x_n are conditionally independent and identically distributed given θ.

1.3.3 Examples

One of the interesting properties of generative models is the phenomenon of *explaining away*. Explaining away describes the interaction of two variables that are marginally independent, yet both influence a common child. If nothing is known about the child variable, then the parents remain independent, but if the child variable is observed, then the parents can become dependent given the child. This counterintuitive result is easiest to understand by example.

A simple case of explaining away is the fuel gauge example. Suppose the fuel gauge of your car indicates an empty tank: one hypothesis is that the tank is indeed empty, an alternate hypothesis is that the battery is dead, and thus the fuel gauge gives a faulty reading (for the sake of argument suppose that there are no other possible influences on the fuel gauge). This situation corresponds to the graphical model in Figure 1.4. After looking at the fuel gauge, you are inclined to think that either explanation is equally likely given your observation. You realize that if you turn on the lights, you will see if the battery is working, and this will help you guess if the gas tank is empty. If the lights do not come on, then the battery must be dead, and your confidence that you may have some fuel increases. This is what is meant by explaining away: the fact that the battery is

dead explains the fuel gauge reading – you may be out of gas as well, but your belief that you have gas is greater than if you thought the battery was working. Thus explaining away is a competition between explanations. In generative models, this competition is implemented in a principled way. The principle of explaining away allows generative models to infer the most likely explanations for an observation among several alternatives.

An illustration of Bayesian inference in perception is given by a one-dimensional pin-hole camera (see Figure 1.5). If x is the one-dimensional sensor image, and the world consists of a point light source at some hypothetical two-dimensional location relative to the observer specified in θ, then $p(x|\theta)$ might be defined via the expected pattern of light at the sensor image — the forward model — plus uncertainty due to sensor noise. Suppose that the observed image is dark except for a tiny spot at one location. The likelihood as a function of hypotheses, θ, given the observation would then have larger values concentrated around those points that could lead to observations similar to x, thus it will be concentrated around a ray extending out of the aperture of the camera through the actual point light source. Suppose prior knowledge tells us that the light source is roughly a certain distance away; that is, $p(\theta)$ is concentrated around an area in space at a certain distance from the camera. Then the posterior will be concentrated around the intersection of the posterior and the likelihood function, as depicted in Figure 1.5. The oblong shape of the posterior in this case indicates the uncertainty about the distance of the light from the camera given this one observation.

A more realistic vision model is discussed in Chapter 4 (see Figure 1.6). This model (*G-flow*) simultaneously tracks 3D pose, non-rigid motion, object texture and background texture. The beauty of this formulation is that we can use our knowledge of projective geometry to design the general form of the model, yet leave calibration parameters and noise distributions to be learned from data. Not everything has to be designed, and not everything has to be learned.

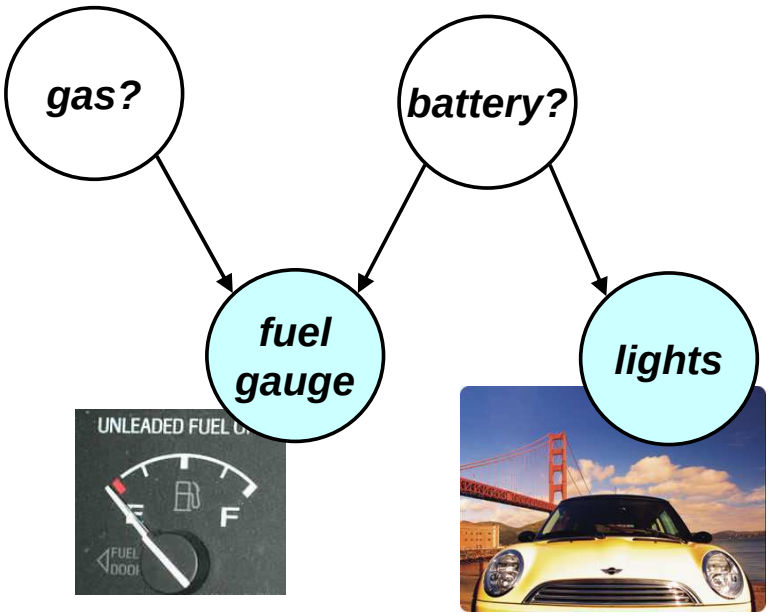


Figure 1.4: Explaining away: the fuel gauge example: Your fuel gauge is on empty, and you suspect that the battery may be dead, or you may be out of gas, or both. You turn on the lights to see if the battery is working. If the lights come on you tend to believe that you are out of gas, whereas if the lights are very dim you tend to believe that you may have gas.

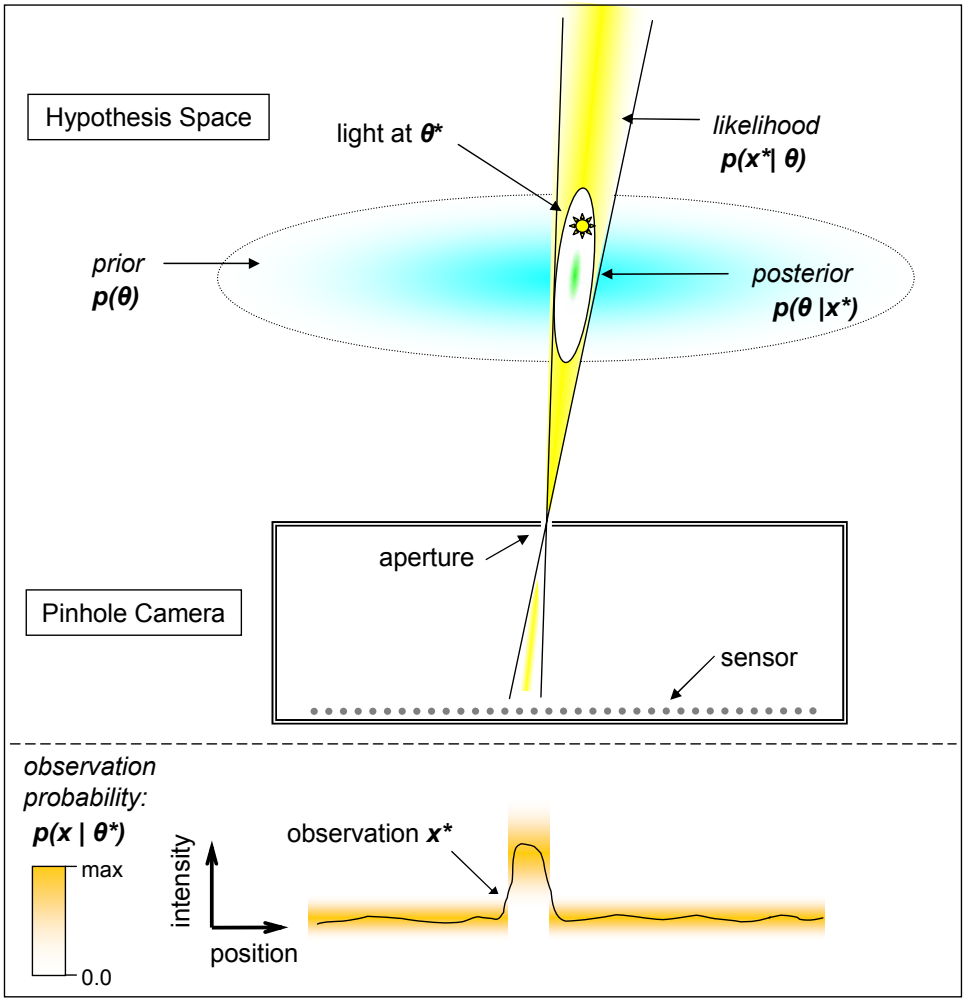


Figure 1.5: Toy example: Bayesian inference in a pinhole camera. Inside the camera the observation x^* is depicted as a plot (black) of intensity as a function of position received at the sensor (dotted line). The true position of the light point source in the world is $\theta*$ plotted as a star in hypothesis space. Given $\theta*$ the observation probability as a function of x is depicted as a concentration around the observation (orange). Given observation x^*, the likelihood function as a function of θ is plotted as a beam concentrated around points in hypothesis space consistent with x^* (yellow). Prior knowledge of likely hypotheses $p(\theta)$ is depicted as an elliptical distribution hovering some distance from the camera (blue). The posterior $p(\theta|x^*)$ is depicted as an elliptical distribution concentrated on the intersection between the likelihood and the prior (green).

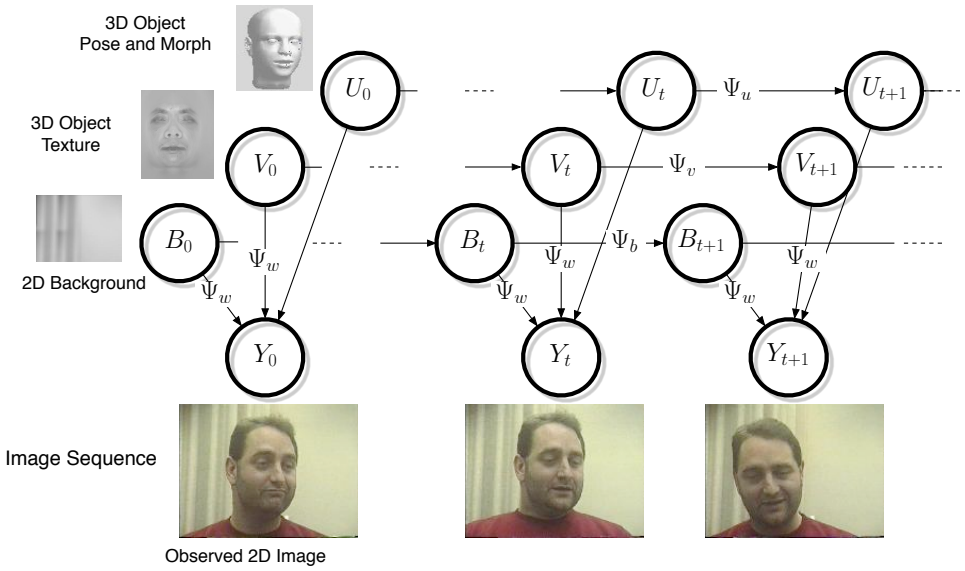


Figure 1.6: A more complicated graphical model described in Chapter 4, in which elements of the scene, such as the background, face pose, face morph (deformation), and surface appearance, all interact to explain the observed video.

1.3.4 Problem-Level Assumptions

The probabilistic modeling approach allows us to build in problem-level assumptions, such as, for instance, the assumptions that different objects do not occupy the same space, and individual objects cannot be in multiple places at once. Certainly some probabilistic models can be relatively unstructured. The Boltzmann machine [16], for instance, was a binary graphical model that, although extremely powerful in its learning ability, had little built-in structure, and was thus somewhat amorphous. However, probabilistic models can be structured in ways that reflect an understanding of the problem. For example, the Markov random field (MRF) approach of Geman and Geman [32], which addressed the problem of enhancing images, incorporated the topological structure of the hidden image, a model of spatial blurring, distortion, and noise that produce the observed image given the clean image. Inference in the model can be derived and implemented according to these constraints in a variety of ways. However the question of what

is inferred must be understood at a level above the implementation — at the problem-level.

For generative models, in particular, the assumptions can often be expressed in terms of properties of the world, and hence their validity is often easy to judge. If the assumptions are too tight in the sense that they are often violated by experience, or conversely if the assumptions are too loose in that they allow for sensory signals that never happen or fail to capture important regularities, then we can come up with ideas about how to adjust them to bring them closer to reality.

For example, models for vision often assume that there is diffuse lighting and all surfaces reflect light equally in all directions. Such an assumption limits the conditions under which inference will be optimal and allows us to predict when the method will be unreliable. However it also allows us to propose new assumptions to handle different situations.

1.3.5 Function versus Mechanism

Probabilistic methods explicitly represent uncertainty: a random variable is represented by a probability distribution over possible values, which can represent both the most likely values and how certain the system is about those values. Deterministic systems, such as neural networks, typically do not explicitly represent uncertainty. However there is a paradox: typically inference in probabilistic systems operates deterministically, and the representation of uncertainty may not be apparent when looking at its implementation. Thus, although probabilistic models can be distinguished from deterministic models in their specification of assumptions and representation of uncertainty at the functional level, the mechanisms that result from implementing inference in the probabilistic model may be indistinguishable from a particular deterministic model.

For instance, in a probabilistic classifier, the *maximum a posteriori* class (the most likely class given an observation) can be inferred from observations of the data. Any function that produces the same mapping from observations to states

— for instance, a simple decision threshold in the observation space – would be an implementation of optimal classification within that model. However, such a mapping may not have an explicit representation of the uncertainty in the posterior distribution of the classes, or the uncertainty in the probability of the observation given the classes. Thus, the whole question of the probabilistic versus deterministic nature of the system hinges upon the level of analysis at which the system is understood.

1.3.6 Explaining Mechanisms

The fact that probabilistic models define an optimality relationship between the assumptions of the model and the resulting inference algorithms means that they can help us understand existing mechanisms. They lend themselves to an axiomatic approach, in which axioms are stated in terms of assumptions of functional and probabilistic relationships between variables. These assumptions lead to optimal methods that can be compared to existing or intuitive methods of solving a particular problem. If they are similar we gain an understanding of the conditions under which the existing method is optimal, as well as a family of related methods that can be obtained by varying the assumptions.

An example of such an understanding of a computational system is given in the independent component analysis (ICA) literature. ICA, which finds linear combinations of its inputs that are statistically independent, was formulated by [8] in terms of a learning rule called *Infomax* that maximizes the mutual information between the inputs and outputs of a neural network. One application of ICA was that if its inputs were linear mixtures of various signals such as speech and music, the Infomax learning rule could often find the linear combinations that would separate the input signals.

The same problem can be posed as a generative model in which independence is assumed in the hidden variables, and the observed variables are linear combinations of the hidden variables [70]. Under this analysis the nonlinear function of the

neurons in Infomax plays the same role as the cumulative distribution function of the hidden variables in the generative model approach. Thus the generative model approach gave meaning to this nonlinearity by making explicit its assumption about the probability distribution of the sources. This new understanding made clear the conditions under which the algorithm would fail, namely when the assumed source distribution poorly represented the actual source distribution. It showed that under a special case, when the sources were assumed to be Gaussian, ICA not only would fail to separate the sources, but would reduce precisely to the well-known principle components analysis (PCA) problem.

The generative model framework permitted further development of the idea. The source distribution could now be learned [2] and the algorithm could be extended. For instance, [70] proposed an extension where the sources contained temporal correlations, and [49] proposed a Bayesian model in which prior knowledge of the signal propagation constraints (inverse-square law) could be used to simultaneously constrain the inference of the sources, their mixing strengths, and their spatial locations.

1.3.7 Caveats

One problem with the use of generative models is that their generality has a cost: they may not perform as well for a particular discrimination task as a system trained to do just that task. For instance, if the goal is to predict video from audio, it might be better to construct a discriminative model that specifically optimizes that prediction, rather than a model that tries to explain the whole joint distribution of audio and video. Generative models describe the entire joint distribution, so they are much more flexible than discriminative models: any variable can be treated as observed or hidden. It possible however to incorporate discriminative models, for instance to provide proposal distributions for Monte-Carlo inference techniques, as in [91, 24].

In the audio-visual experiments reported in Chapter 6 cross-modal learning

was difficult to control because they tended to explain the strong within-modality dependencies at the expense of the weak between-modality dependencies. A future direction is to investigate the conditions under which cross-modal learning is optimal in a generative model. Preliminary work suggests that cross-modal learning can be promoted by incorporating rich models of strong within-modality dependencies so that there is nothing left to explain but cross-modal dependencies.

Another problem is with the claim that the assumptions of a probabilistic model are hypotheses that can be tested by the model and that allow us to understand the mechanisms of its implementation. The flexibility in the implementation of a probabilistic model means that some of the behavior may have more to do with how the model is implemented than with the model itself. This raises an important advantage of the probabilistic framework: it requires one to specify a set of assumptions about the world, and a specification of what to be inferred, but leaves open a wide range of strategies for achieving inference, and for using the inference in decision-making. The caveat is that this flexibility can also be a liability for the approach, if the success or failure is a by-product of the choice of implementation rather than the model itself.

In addition, although it is sometimes tempting to suppose that a model explains a mechanism in the brain, there may be many different models that produce similar mechanisms. For instance, the *G-flow* model presented in Chapter 4 leads to a gabor-like representation similar to that of primary visual cortex simple cells. The images are first blurred because the assumptions that lead to optic flow demand spatial smoothness. Then spatial and temporal derivatives are taken, which follow from the Taylor series expansion in the Laplace approximation. The combination of a blurring function with a derivative leads to orientation-specific spatial filters. However, we cannot conclude that this explains what we observe in the brain: other hypotheses about their role, such as edge-detection, texture analysis, shape-from-shading, information maximization and redundancy reduction, lead to similar mechanisms [69, 45]. Perhaps this confluence of explanations reflects a deep principle, or perhaps it reflects serendipity. Ultimately when we do not understand

the context in which a mechanism operates, we must be careful not to jump to conclusions about its purpose.

Regarding the approach in general, one might argue that it is only a matter of time before we have a learning system that can learn to see without any assumptions just by observing data as it moves around in the world, and so forth. Furthermore if and when such a learning system works it may automatically organize into layers and modules that are easy to understand. This is an extreme caricature of the black-box learning approach. An answer is that this would be great if we can get it to work. However, we would still have to sort out which parts are important for what perceptual task. Another answer is that this is a big "if" and an even bigger "when." In the meantime we have no choice but to make assumptions, and choose frameworks that allow us to make informed assumptions, and yet still allow us to learn from data. There may be other architectures that do this besides graphical models, and there is always room for more approaches.

1.4 Thesis Overview

The remaining chapters of this thesis consist of articles that have been published or submitted for publication describing research that I have conducted over the past four years with various colleagues. I began this work in machine perception by looking at signal-level synchronies between audio and video as a means to quickly infer the location of a sound given signals from a single microphone and video camera (see Chapter 2). The resulting system worked well when the targets had local motion such as lip movements, but was susceptible to large motions in the video and complex changes in the audio. In addition since the model was discriminative, it was not obvious how to infer audio from video or vice verse. Thus it became clear that active tracking using generative models in both domains was necessary in order to simultaneously infer gross motion and dependencies between the fine-scale changes in the audio and video. Tracking in turn required more advanced models of the objects appearance in the video signal and its manifestation

in the audio signal, and how these signals change over time. After separately investigating video tracking models (see Chapters 3 and 4), and audio separation models (see Chapter 5), I returned to an audio-visual paradigm in which audio and video could be inferred from each other (see Chapter 6). Hence I have gone from using low-level audio-visual synchrony at one end, through more complex modeling of audio and video signals, and finally returned to the task of inferring one signal from the other. Along the way it turned out that taking uncertainty into account was critical to the success of the models I proposed. The generative model framework, in addition to providing principled learning methods, served to organize my thoughts on the perceptual problem to be solved.

1.4.1 Audio-Vision: Using Audio-Visual Synchrony to Locate Sounds

Chapter 2 presents a model of multi-modal perception. The task was to track the location of the speaker in image coordinates using only single-microphone and video camera signals of two people talking. The idea was to do this in a way that assumed no prior information about the appearance of likely sound sources. Such cross-modal information appears to be responsible for the adaptation of spatial hearing relative to vision in animals. Since this type of experiment had never been done before we wanted to find out if there was enough information in low-level signal features to locate the sound. Mutual information was used in a system to detect statistical dependencies between the amplitude of short segments of audio signal and the brightness of different image areas over time. We found that if the speakers did not move their heads while they spoke, the system could infer the source of the sound, however with gross head movements that were statistically unrelated to the sound the system could no longer make correct inferences.

1.4.2 Large-Scale Convolutional HMMs for Real-Time Video Tracking

Chapter 3 deals with a system for rapidly tracking a target face using very simple features: in this case the color of the pixels in the object. Because color is highly variable and background colors could be similar to the target color, a background model, and adaptation to the foreground model were necessary to achieve good performance. The model was framed as a generative model in which the entire image was generated as follows: random variables controlled the location and size of a face bounding box in the video. Pixels inside this bounding box were drawn independently from a face color distribution, and pixels outside the box were drawn from a background model distribution. In order to track the face in clutter it is important to have knowledge of the dynamics of faces. A novel convolutional hidden Markov model implements this knowledge in an extremely efficient way that allows the full filtering inference to be solved, and enables tracking using simple cues. The model also cooperates with a face finding module which supplies information about the location of the face every once in a while, allowing the color model to adapt to the changing lighting situation. This adaptation allows us to take advantage of color features such as intensity that would normally not be invariant to prevailing lighting conditions.

1.4.3 G-Flow: A Generative Model for Fast Tracking Using 3D Deformable Models

Chapter 4 concerns a three-dimensional generative model for flexible objects such as the face, in which simple projective geometry defines the forward model from face texture, pose, and shape, to the sensor image. The model is constrained by assumptions about face shapes incorporated into a face model. No other constraints are imposed. Remarkably the model unifies the two computer vision approaches of template matching and optic flow. A Laplace approximation yields

an optic flow computation from one frame to the next. As the model gains certainty of the texture appearance of the face, it can shift to a template matching approach that uses the inferred texture to find the face location and pose in each frame.

1.4.4 Single Microphone Source Separation

Chapter 5 presents three experiments in which a forward model of sound combination, combined with models of the individual sounds can separate single-channel mixtures of the sounds. In order to do so it was important to adopt a feature set that was very different from that typically used in speech recognition. The model exploits the harmonic structure of speech by employing a high frequency resolution speech model in the log-spectrum domain and reconstructs the signal from the estimated posteriors of the clean signal and the phases from the original noisy signal. We achieved substantial gains in signal to noise ratio (SNR) of enhanced speech as well as considerable gains in accuracy of automatic speech recognition in very noisy conditions. The model has interesting implications for perception since it relies on the "explaining-away" property of probabilistic models. The fact that one sound model explains the observed signal in a particular area of the spectrum, means that the other model need not explain it. The model's explanation of the observed mixture is the result of a competition between two models to explain different parts of the spectrum. Thus there are top-down aspects of pattern completion and filling in that are important to the good results we obtained.

1.4.5 Audio-Visual Graphical Models for Speech Processing

Chapter 6 introduces a novel generative model that combines audio and visual signals. The model allows cross-modal inference, enabling adaptation to audio-visual data. We formulated a flexible object representation in a way that

provides for unsupervised learning of an appearance-based manifold of prototypes in a low-dimensional linear subspace embedded in the high-dimensional space of pixels. The experiment in this chapter has a flavor of understanding the content of a video via its contingencies with the audio modality as measured by the contribution of video to speech enhancement. However because it is a generative model, inference can also be done in the other direction – that is, interpreting the audio via its contingency with video. This flexible combination of modalities demonstrates the elegance of generative models of perception.

Chapter 2

Audio-Vision: Using Audio-Visual Synchrony to Locate Sounds

Abstract

Psychophysical and physiological evidence shows that localization of acoustic signals is strongly influenced by their synchrony with visual signals. This effect, known as ventriloquism, is at work when sound coming from the side of a TV set feels as if it were coming from the mouth of the actors. The ventriloquism effect suggests that there is important information about sound location encoded in the synchrony between the audio and video signals. In spite of this evidence, audiovisual synchrony is rarely used as a source of information in machine perception tasks. In this chapter we explore the use of audio visual synchrony to locate sound sources. We developed a system that searches for regions of the visual scene that correlate highly with the acoustic signals and tags them as likely to contain an acoustic source. We discuss our experience implementing the system, present results on a speaker localization task and discuss potential applications of the approach.

2.1 Introduction

We present a method for locating sound sources by sampling regions of an image that correlate in time with the auditory signal. Our approach is inspired by psychophysical and physiological evidence suggesting that audio-visual contingencies play an important role in the localization of sound sources: sounds seem to emanate from visual stimuli that are synchronized with the sound. This effect becomes particularly noticeable when the perceived source of the sound is known to be false, as in the case of a ventriloquist's dummy, or a television screen. This phenomenon is known in the psychophysical community as the *ventriloquism effect*, defined as a mislocation of sounds toward their apparent visual source. The effect is robust in a wide variety of conditions, and has been found to be strongly dependent on the degree of "synchrony" between the auditory and visual signals [19, 10].

The ventriloquism effect is in fact less speech-specific than first thought. For example the effect is not disrupted by an upside-down lip signal [10] and is just as strong when the lip signals are replaced by light flashes that are synchronized with amplitude peaks in the audio signal [72]. The crucial aspect here is correlation between visual and auditory intensity over time. When the light flashes are not synchronized the effect disappears.

The ventriloquism effect produces an enduring localization bias, known as the *ventriloquism aftereffect*. Over time, experience with spatially offset auditory-visual stimuli causes a persistent shift in subsequent auditory localization. Exposure to audio-visual stimuli offset from each other by only 8 degrees of azimuth for 20-30 minutes is sufficient to shift auditory localization by the same amount. A corresponding shift in neural processing has been detected in macaque monkeys as early as primary auditory cortex[73]. In barn owls a misalignment of visual and auditory stimuli during development causes the realignment of the auditory and visual maps in the tectum[97, 82, 25].

The strength of the psychophysical and physiological evidence suggests that

audio-visual contingency may be used as an important source of information that is currently underutilized in machine perception tasks. Visual and auditory sensor systems carry information about the same events in the world, and this information must be combined correctly in order for a useful interaction of the two modalities. Audiovisual contingency can be exploited to help determine which signals in different modalities share a common origin. The benefits are two-fold: the two signals can help localize each other, and once paired can help interpret each other. To this effect we developed a system to localize speakers using input from a camera and a single microphone. The approach is based on searching for regions of the image which are "synchronized" with the acoustic signal.

2.2 Measuring Synchrony

The concept of audio-visual *synchrony* is not well formalized in the psychophysical literature, so for a working definition we interpret synchrony as the degree of mutual information between audio and spatially localized video signals. Ultimately it is a *causal* relationship that we are often interested in, but causes can only be inferred from effects such as synchrony. Let $a(t) \in \mathbb{R}^n$ be a vector describing the acoustic signal at time t. The components of $a(t)$ could be cepstral coefficients, pitch measurements, or the outputs of a filter bank. Let $v(x, y, t) \in \mathbb{R}^m$ be a vector describing the visual signal at time t, pixel (x, y). The components of $v(x, y, t)$ could represent Gabor energy coefficients, RGB color values, etc.

Consider now a set of s audio and visual vectors $\mathcal{S} = (a(t_l), v(x, y, t_l))_{l=k-s+1,\cdots,k}$ sampled at times $t_{k-s+1}, \cdots, t_k$ and at spatial coordinates (x, y). Given this set of vectors our goal is to provide a number that describes the temporal contingency between audio and video at time t_k. The approach we take is to consider each vector in $\mathcal{S}$ as an independent sample from a joint multivariate Gaussian process $(A(t_k), V(x, y, t_k))$ and define audio-visual synchrony at time t_k as the estimate of the mutual information between the audio and visual components of the process.

Let $A(t_k) \sim \mathcal{N}_n(\mu_A(t_k), \Sigma_A(t_k))$, and $V(x, y, t_k) \sim \mathcal{N}_m(\mu_V(x, y, t), \Sigma_V(x, y, t_k))$,

where μ represents means and Σ covariance matrices. Let $A(t_k)$ and $V(x, y, t_k)$ be jointly Gaussian, i.e., $(A(t_k), V(x, y, t_k)) \sim \mathcal{N}_{n+m}(\mu_{A,V}(x, y, t_k), \Sigma_{A,V}(x, y, t_k))$. The mutual information between $A(x, y, t_k)$ and $V(t_k)$ can be shown to be as follows

$$\begin{aligned} I(A(t_k); V(x, y, t_k)) &= H(A(t_k)) + H(V(x, y, t_k)) - H(A(t_k), V(x, y, t_k)) \\ &= \frac{1}{2}\log(2\pi e)^n |\Sigma_A(t_k)| \\ &\quad + \frac{1}{2}\log(2\pi e)^m |\Sigma_V(x, y, t_k)| \\ &\quad - \frac{1}{2}\log(2\pi e)^{n+m} |\Sigma_{A,V}(x, y, t_k)| \\ &= \frac{1}{2}\log\frac{|\Sigma_A(t_k)||\Sigma_V(x, y, t_k)|}{|\Sigma_{A,V}(x, y, t_k)|}. \end{aligned} \tag{2.1}$$

In the special case that $n = m = 1$, then

$$I(A(t_k); V(x, y, t_k)) = -\frac{1}{2}\log(1 - \rho^2(x, y, t_k)), \tag{2.2}$$

where $\rho(x, y, t_k)$ is the Pearson correlation coefficient between $A(t_k)$ and $V(x, y, t_k)$.

For each triple (x, y, t_k) we estimate the mutual information between $A(t_k)$ and $V(x, y, t_k)$ by considering each element of $\mathcal{S}$ as an independent sample from the random vector $(A(t_k), V(x, y, t_k))$. This amounts to computing estimates of the joint covariance matrix $\Sigma_{A,V}(x, y, t_k)$. For example the estimate of the covariance between the i^{th} audio component and the j^{th} video component would be as follows

$$S_{A_i,V_j}(x, y, t_k) = \frac{1}{s-1}\sum_{l=0}^{s-1}(a_i(t_{k-l}) - \bar{a}_i(t_k))(v_j(x, y, t_{k-l}) - \bar{v}_j(x, y, t_k)), \tag{2.3}$$

where

$$\bar{a}_i(t_k) = \frac{1}{s}\sum_{l=0}^{s-1} a_i(t_{k-l}), \tag{2.4}$$

$$\bar{v}_j(t_k) = \frac{1}{s}\sum_{l=0}^{s-1} v_j(x, y, t_{k-l}). \tag{2.5}$$

These simple covariance estimates can be computed recursively in constant time with respect to the number of timepoints. The independent treatment of pixels would lend itself well to a parallel implementation.

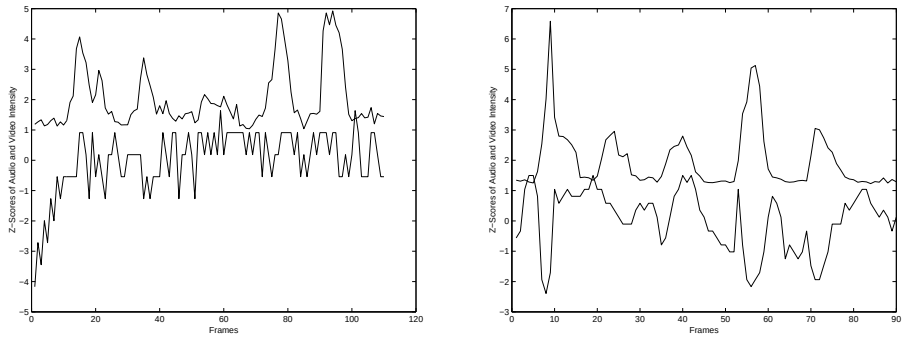
To measure performance, a secondary system produces a single estimate of the auditory location, for use with a database of labeled solitary audiovisual sources. Unfortunately there are many ways of producing such estimates so it becomes difficult to separate performance of the measure from the underlying system. The model used here is a centroid computation on the mutual information estimates, with some enhancements to aid tracking and reduce background noise.

2.3 Implementation Issues

A real time system was prototyped using a QuickCam on the Linux operating system and then ported to NT as a DirectShow filter. This platform provides input from real-time audio and video capture hardware as well as from static movie files. The video output could also be rendered live or compressed and saved in a movie file. The implementation was challenging in that it turns out to be rather difficult to process precisely time-synchronized audio and video on a serial machine in real time. Multiple threads are required to read from the peripheral audio and visual devices. By the time the audio and visual streams reach the AV filter module, they are quite separate and asynchronous. The separately threaded auditory and visual packet streams must be synchronized, buffered, and finally matched and aligned by time-stamps before they can finally be processed. It is interesting that successful biologial audiovisual systems employ a parallel architecture and thus avoid this problem.

2.4 Results

To obtain a performance baseline we first tried the simplest possible approach: A single audio and visual feature per location: $n = m = 1$, $v(x, y, t) \in \mathbb{R}$ is the intensity of pixel (x, y) at time t, and $a(t) \in \mathbb{R}$ is the average acoustic energy over the interval $[t - \Delta t, t]$, where $\Delta t = 1/30$ sec , the sampling period for the NTSC video signal. Figure 2.1 illustrates the time course of these signals for a

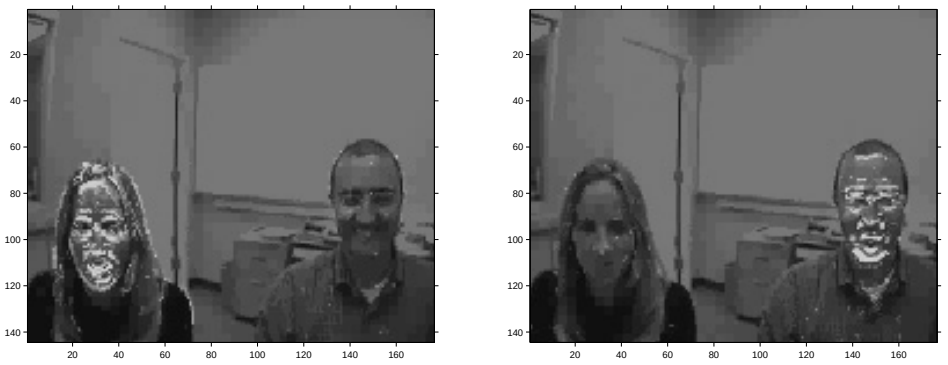


(a) M is talking. (b) J is talking.

Figure 2.1: Normalized audio and visual intensity across a sequence of frames in which a sequence of four numbers is spoken. The top trace is the contour of the acoustic energy from one of two speakers, M or J, and the bottom trace is the contour of intensity values for a single pixel, (147,100), near the mouth of J.

non-synchronous and a synchronous pair of acoustic energy and pixel intensity. Notice in particular that in the synchonous pair, 2.1(b), where the sound and pixel values come from the same speaker, the relationship between the signals changes over time. There are regions of positive and negative covariance strung together in succession. Clearly the relationship over the entire sequence is far from linear. However over shorter time periods a linear relationship looks like a better approximation. Our window size of 16 samples (i.e., $s = 16$ in equation 2.3 coincides approximately with this time-scale. Perhaps by averaging over many small windows we can capture on a larger scale what would be lost to the same method applied with a larger window. Of course there is a trade-off in the time-scale between sensitivity to spurious transients, and the response time of the system.

We applied this mutual information measure to all the pixels in a movie, in the spirit of the perceptual maps of the brain. The result is a changing topographic map of audiovisual mutual information. Figure 2.2 illustrates two snapshots in which different parts of the face are synchronous (possibly with different sign) with the sound they take part in producing. It is interesting that the synchrony



(a) Frame 206: M (at left) is talking. (b) Frame 104: J (at right) is talking.

Figure 2.2: Estimated mutual information between pixel intensity and audio intensity (bright areas indicate greater mutual information) overlaid on stills from the video where one person is in mid-utterance.

is shared by some parts, such as the eyes, that do not directly contribute to the sound, but contribute to the communication nonetheless.

To estimate the position of the speaker we computed a centroid were each point was weighted by the estimated mutual information between the correpsonding pixel and the audio signal. In order to reduce the intrusion of spurious correlations from competing targets, once a target has been found, we employ a Gaussian *influence function* [35]. The influence function reduces the weight given to mutual information from locations far from the current centroid when computing the next centroid. To allow for the speedy disengagement from a dwindling source of mutual information we set a threshold on the mutual information. Measurements under the threshold are treated as zero. This threshold also reduces the effects of unwanted background noise, such as camera and microphone jitter.

$$\hat{S}_x(t) = \frac{\sum_x \sum_y x\ \theta(\log(1-\hat{\rho}^2(x,y,t)))\psi(x,\hat{S}_x(t-1))}{\sum_x \sum_y \theta(\log(1-\hat{\rho}^2(x,y,t)))\psi(x,\hat{S}_x(t-1))} \tag{2.6}$$

where $\hat{S}_x(t)$ represents the estimate of the x coordinate for the position of the speaker at time t. $\theta(.)$ is the thresholding function, and $\psi(x,\hat{S}_x(t-1))$ is the influence function, which depends upon the position x of the pixel being sampled

and the prior estimate $\hat{S}_x(t-1)$. $\hat{\rho}^2(x,y,t)$ is the estimate of the squared correlation between the intensity in pixel (x,y) and the acoustic enery, when using the 16 past video frames. $-\frac{1}{2}\log(1-\hat{\rho}^2(x,y,t))$ is the corresponding estimate of mutual information (the factor, $-\frac{1}{2}$ cancels out in the quotient after adjusting the threshold function accordingly.)

We tried the approach on a movie of two people (M and J) taking turns while saying random digits. Figure 2.3 shows the estimates of the actual positions of the speaker as a function of time. The estimates clearly provide information that could be used to localize the speaker, especially in combination with other approaches (e.g., flesh detection).

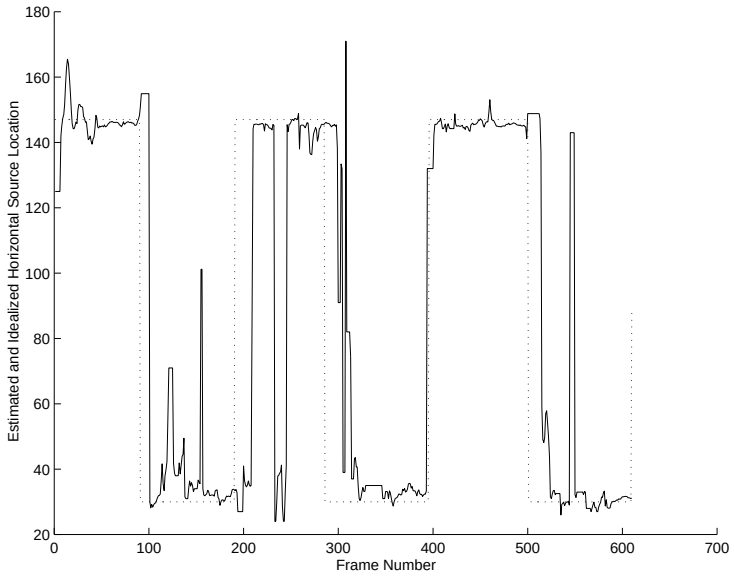


Figure 2.3: Estimated and actual position of speaker at each frame for six hundred frames. The sources, M and J, took turns uttering a series of four digits, for three turns each. The actual positions and alternation times were measured by hand from the video recording

2.5 Conclusions

We have presented exploratory work on a system for localizing sound sources on a video signal by tagging regions of the image that are correlated in time with the auditory signal. The approach was motivated by the wealth of evidence in the psychophysical and physiological literature showing that sound localization is strongly influenced by synchrony with the visual signal. We presented a measure of local synchrony based on modeling the audio-visual signal as a non-stationary Gaussian process. We developed a general software tool that accepts as inputs all major video and audio file formats as well as direct input from a video camera. We tested the tool on a speaker localization task with very encouraging results. The approach could have practical applications for localizing sound sources in situations where acoustic stereo cues are unavailable or unreliable. It also might be useful for learning the calibration of acoustic localization.

While the results reported here are very encouraging, more work needs to be done before practical applications are developed. For example we need to investigate more sophisticated methods for processing the audio and video signals. At this point we use average energy to represent the video and thus changes in the fundamental frequency that do not affect the average energy would not be captured by our model. Similarly local video decompositions, like spatio-temporal Gabor filtering, or approaches designed to enhance the lip regions may be helpful. The changing symmetry observed between audio and video signals might be addressed rectifying or squaring the normalized signals and derivatives. Finally, relaxing the Gaussian constraints in our measure of audio-visual contingency may help improve performance. While the work shown here is exploratory at this point, the approach is very promising: It emphasizes the idea of machine perception as a multimodal process it is backed by psychophysical evidence, and when combined with other approaches it may help improve robustness in tasks such as localization and separation of sound sources.

2.6 Acknowledgements

The contents of this chapter are adapted from [40] which was published in Advances in Neural Information Processing Systems in 2000. The coauthor, Javier Movellan, supervised and collaborated with me on the research which forms the basis of this chapter.

Chapter 3

Large-Scale Convolutional HMMs for Real-Time Video Tracking

Abstract

Bayesian filtering provides a principled approach for a variety of problems in machine perception and robotics. Current filtering methods work with analog hypothesis spaces and find approximate solutions to the resulting non-linear filtering problem using Monte-Carlo approximations (i.e., particle filters) or linear approximations (e.g., extended Kalman filter). Instead, in this chapter we propose digitizing the hypothesis space into a large number, $n \approx 100,000$, of discrete hypotheses. Thus the approach becomes equivalent to standard hidden Markov models (HMM) except for the fact that we use a very large number of states. One reason this approach has not been tried in the past is that the standard forward filtering equations for discrete HMMs require order n^2 operations per time step and thus rapidly become prohibitive. In our model, however, the states are arranged in two-dimensional topologies, with location-independent dynamics. With this arrangement predictive distributions can be computed via convolutions. In addition, the computation of log-likelihood ratios can also be performed via convolutions. We describe algorithms that solve the filtering equations, performing

this convolution for a special class of transition kernels in order n operations per time step. This allows exact solution of filtering problems in real time with tens of thousands of discrete hypotheses. We found this number of hypotheses sufficient for object tracking problems. We also propose principled methods to adapt the model parameters in non-stationary environments and to detect and recover from tracking errors.

3.1 Introduction

Bayesian filtering refers to the problem of making inferences about the values taken by random variables at time t based on a sequence of observations up to that time. In computer vision the observed variables are typically image sequences and the unobserved variables are analog in nature (e.g, pose and deformation parameters of an object). For this reason continuous state filtering approaches are the preferred choice. While exact analytical solutions exist for continuous filtering problems the needed assumptions (Gaussianity and Linearity) are too restrictive. Thus Monte-Carlo approximations (i.e., particle filters) have become the method of choice in computer vision and much work is being devoted to making these approximations as efficient as possible [15, 1, 92].

In speech recognition *hidden Markov models* (HMMs) are used to model the dynamics of speech. The observable data of interest are phoneme-like segments of speech and thus can be represented well with a small number of discrete states or *hypotheses* ($n_h \approx 50$). Furthermore the models are typically constrained for state transitions to have left-to-right constraints. The small number of states and state transitions allows the filtering problem to be solved exactly in real time. Unfortunately the filtering equations in the discrete case require order n_h^2 operations per time step and thus do not scale well for large, densely connected hypothesis spaces. This is arguably the main reason why HMMs are not as popular in computer vision as they are in speech recognition.

While in general the discrete filtering equations scale order n_h^2, in most

computer vision problems there is spatio-temporal structure that can be used to develop faster inference algorithms. In this chapter we present a new algorithm that works in order n_h operations. This makes it possible to solve inference problems in real time with tens of thousands of hypotheses. We exhaustively populate a continuous hypothesis space with a large number of discrete states and solve the resulting inference problem exactly. To do so efficiently we employ a double-integral technique to dramatically reduce computation time. Whereas previous models have discretized hypothesis spaces, as far as we know the use of the double integral technique to perform inference in such models is novel. We describe these techniques in the context of 2D tracking problems. Extensions are discussed in Section 3.8.

3.2 A Generative Model for 2D Tracking

We identify random variables with capital letters, and specific values taken by those variables with small letters. When possible we use shorthand notation and identify probability functions by their arguments. For example, $p(h_t)$ is shorthand for $p_{H_t}(h_t)$, the probability (or probability density) that the random variable H_t takes the specific value h_t. We use subscripted columns to designate sequences. For example $y_{1:t} = y_1 \cdots y_t$. Finally we reserve Greek letters for parameters.

We model the image generation process as follows (see Figure 3.1): First a parameter λ_t is chosen by a process described in Section 3.5. This parameter determines the location-dependent probability distribution of image features in the background $b_i(\cdot \mid \lambda_t)$, and the probability distribution of image features in the object of interest $o(\cdot|\lambda_t)$. The image features can be any function of a local patch of pixels, and can represent for instance texture, color, or motion or object categories (see [65]). Without loss of generality we formulate the model here using the color of individual pixels as the feature of interest [1].

[1] Here color refers to an rgb value and thus it may include intensity, not just hue and saturation.

The pixels rendered by the object are inside a rectangle of fixed aspect ratio $h_t = (x_t, s_t)$ centered at x_t, with scale parameter s_t. Generalizations to arbitrary rotations and non-rectangular hypotheses are easy (see Section 3.8) once this case is understood. The rectangle containing the object pixels is chosen with probability $p(x_t s_t \mid x_{t-1} s_{t-1}) = p(s_t \mid s_{t-1}) p(x_t \mid x_{t-1} s_{t-1})$.

Once h_t is known, we know which pixels are rendered by the background and which are rendered by the object of interest. For each pixel location u in the background, a color $y_t(u)$ is chosen with probability $b_i(y_t(u) \mid \lambda_t)$. For each pixel v in the object, a color $y_t(v)$ is chosen with probability $o(y_t(v) \mid \lambda_t)$

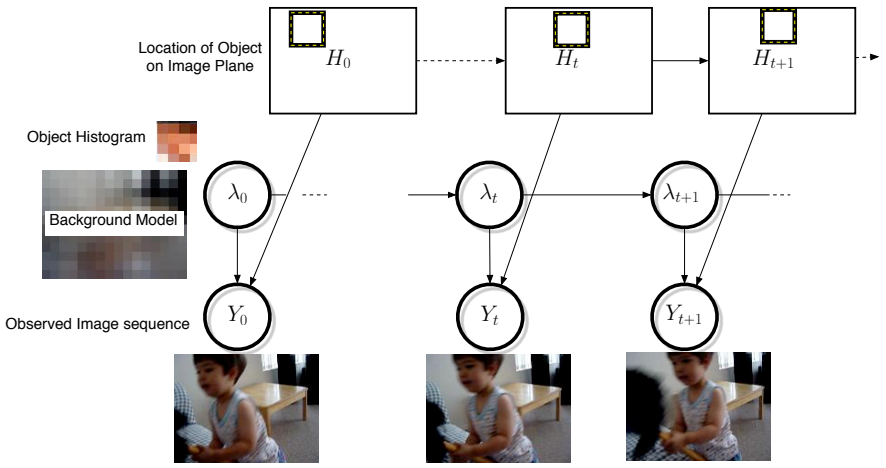


Figure 3.1: Graphical Appearance Model: The hidden variable H determines which pixels belong to the object and which belong to the background. The object pixels are rendered independently from an object histogram. The background pixels are rendered independently from a space variant background histogram model.

Image Likelihood Let y_t represent the image observed at time t and $y_t(u)$ the value taken by the pixel at location $u \in \mathbb{R}^2$ in that image. From the description of

the model above, it follows that:

$$\log p(y_t \mid x_t s_t \lambda_t) = \log \prod_{u \in h_t} o(y_t(u) \mid \lambda_t) + \log \prod_{u \notin h_t} b_i(y_t(u) \mid \lambda_t) \tag{3.1}$$

$$= \sum_{u \in h_t} \log \frac{o(y_t(u) \mid \lambda_t)}{b_i(y_t(u) \mid \lambda_t)} + Z(y_t, \lambda_t) \tag{3.2}$$

where

$$Z(y_t, \lambda_t) = \sum_u \log b_i(y_t(u) \mid \lambda_t) \tag{3.3}$$

The log-likelihood of a hypothesis h_t is a constant Z plus the sum of the log-likelihood ratios of all the pixels within that hypothesis.

Filtering Distribution Let $y_{1:t} = (y_1 \cdots y_t)$ represent an observed image sequence up to time t. Our goal is to compute the filtering distribution, i.e., the posterior distribution of h_t given $y_1 \cdots y_t$. Using the standard HMM update equations we have that the posterior probability of a hypothesis h_t is proportional to the product of the probability of the current image given the hypothesis times the predictive probability of each hypothesis given the past image sequence:

$$p(h_t \mid y_{1:t} \lambda_{1:t}) = \frac{p(y_{1:t-1} \mid \lambda_{1:t})}{p(y_{1:t} \mid \lambda_{1:t})} p(y_t \mid h_t \lambda_{1:t}) p(h_t \mid y_{1:t-1} \lambda_{1:t-1}) \tag{3.4}$$

$$p(h_t \mid y_{1:t-1} \lambda_{1:t-1}) = \sum_{s_{t-1}} p(s_t \mid s_{t-1}) \sum_{x_{t-1}} p(x_t \mid x_{t-1} s_{t-1}) p(h_{t-1} \mid y_{1:t-1} \lambda_{1:t-1}) \tag{3.5}$$

where $h_t = (x_t, s_t)$, $h_{t-1} = (x_{t-1}, s_{t-1})$. For each scale, we let the transition distribution $p(x_t \mid x_{t-1} s_t)$ be rectangular, uniform and shift invariant. This allows us to use cumulative probability maps to compute the predictive probability of each hypothesis with four operations per hypothesis, as described in Section 3.4.

3.3 Minimum Risk Estimation

In many applications we need to choose a single hypothesis per time step. In such cases it is reasonable to choose the hypothesis that minimizes the posterior risk, i.e., the expected average of an error function

$$\hat{h}_t = \underset{h}{\operatorname{argmin}}\, E(\rho(H_t; h) \mid y_{1:t}\ \lambda_{1:t}) \tag{3.6}$$

where ρ is an error function that measures the mismatch between two hypotheses. We experimented with two types of error functions: (1) The correct hypotheses get zero error and incorrect hypotheses get error 1; (2) An error function that measures average distance between corresponding object landmarks in two hypotheses. The first error function is minimized by choosing the hypothesis with maximum posterior probability (MAP).

The minimization of the second error function is described here. For generality we allow different object landmarks to have different weights, by using a normalized relevance map w. Let u represent a point on the image plane. Its standardized location with respect to the hypothesis x, s is $z = (u - x)/s$. The weight of this point is given by $w(z)$. Let μ_x, σ_x^2 represent the mean and variance of the hypothesis x, s with respect to the relevance map w, i.e.

$$\mu_x = \int u\ w(\frac{u-x}{s})du \tag{3.7}$$

$$\sigma_x^2 = \int (u-\mu_x)^2\ w(\frac{u-x}{s})du \tag{3.8}$$

Now consider a different hypothesis x', s'. According to this hypothesis the landmark u from hypothesis x, s is located at $\frac{s'}{s}(u - x) + x'$. The scaled average distance from equivalent landmarks follows:

$$\rho^2(x, s; x', s') = \frac{1}{\sigma_x^2} \int \|u - \frac{s'}{s}(u-x) + x'\|^2\ w(\frac{u-x}{s})du \tag{3.9}$$

This error function has an intuitive interpretation as the expected distance between corresponding landmarks in the two hypotheses: For example if $\rho^2 = 0.25$ the

average error is in the order of 0.5 times the scale of the standard object. After some simple derivations it can be shown that

$$\rho^2(x, s; x', s') = \left(\frac{s_i - s'_i}{s}\right)^2 + \left(\frac{\mu_x - x'}{\sigma_x}\right)^2 \tag{3.10}$$

For simplicity we can choose a relevance map such that $\mu_x = x$ and $\sigma_x = s$, in which case

$$\rho^2(x, s; x', s') = \left(\frac{s_i - s'_i}{s}\right)^2 + \left(\frac{x - x'}{s}\right)^2 \tag{3.11}$$

The error between two hypothesis (x, s) and (x', s') is simply the sum of the squared scaled difference of locations plus the squared scaled difference of the scales between the two hypotheses.

The minimum risk hypothesis for this error function can be found by differentiating the posterior risk with respect to x, s, setting it to zero and solving the resulting equation. The results are as follows:

$$\hat{s} = \frac{1}{E(1/S \mid y_{1:t}\ \lambda_{1:t})}; \quad \hat{x} = \frac{E(X/S \mid y_{1:t}\ \lambda_{1:t})}{E(1/S \mid y_{1:t}\ \lambda_{1:t})} \tag{3.12}$$

3.4 Computational Complexity

To compute the filtering distribution at time $t + 1$ first we need to to compute the predictive distribution at time $t + 1$. This is the distribution of hypothesis at time $t + 1$ based on the observed images up to time t. The predictive distribution contains all the information about the hypotheses prior to the observation of the image at time $t + 1$. It is obtained by propagating the filtering distribution at time $t + 1$ via the state transition function $p(h_{t+1} \mid h_t)$. Following Bayes' rule we then need to compute the likelihood of the image at time $t + 1$ for each possible hypothesis and multiply the predictive probability (prior) times the likelihood of each hypothesis. In this section we describe methods to acheive the desired results in order $n_h + n_p$ operations, making it possible to work with a very large number of hypothesis in real time.

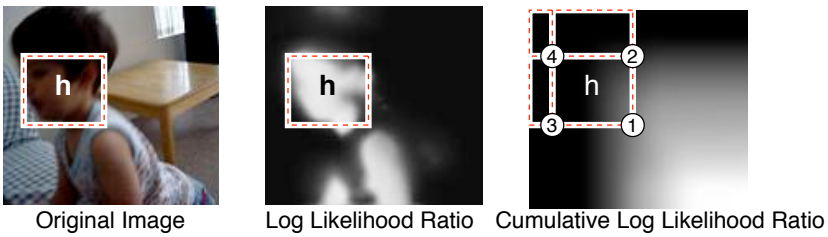


Figure 3.2: Double integral of log likelihood ratios: The log-likelihood of a hypothesis is the sum of the log-likelihood ratios of each pixel. This can be computed in 4 operations using cumulative log-likelihood ratio maps: $l(h) = L(1) - L(2) - L(3) + L(4)$, where $L(i)$ is the cumulative log likelihood ratio evaluated at pixel i.

Double Integral Likelihood Ratio Maps A brute force approach for computing the likelihood-ratios would require order $n_p \times n_h$ sums, where n_p is the number of pixels on the image and n_h the number of hypotheses. In practice we can compute the log-likelihood ratio of all the hypothesis using $n_p + 4n_h$ sums. First for each pixel location $x = (x_1, x_2)^T$ we compute the likelihood ratio of the value taken by that pixel

$$l(x) = \log \frac{o(y_t(x) \mid \lambda_t)}{b_x(y_t(x) \mid \lambda_t)} \tag{3.13}$$

This can be done using table-lookups for the likelihood-ratio function. Then we compute the double integral log-likelihood ratio map L:

$$L(x) = \sum_{u_1=0}^{x_1} \sum_{u_2=0}^{x_2} l(x) \tag{3.14}$$

Once L is known, the probability of each hypothesis can be computed in 4 operations (see Figure 3.2).

Double Derivative Predictive Maps For the general case the problem of computing the predictive probability from the filtering probability takes n_h^2

operations

$$p(x_{t+1}s_{t+1} \mid y_{1:t}\lambda_{1:t}) = \sum_{x_t,s_t} p(x_t s_t \mid y_{1:t}\lambda_{1:t})$$

$$p(x_{t+1}s_{t+1} \mid x_t s_t) \tag{3.15}$$

$$= \sum_{s_t} p(s_{t+1} \mid s_t) \sum_{x_t} p(x_t s_t \mid y_{1:t}\lambda_{1:t}) p(x_{t+1} \mid x_t s_t) \tag{3.16}$$

If the transition probabilities are shift invariant this amounts to a convolution operation for each of the scales, with cost of order $n_s \times n_p \log n_p$, where n_s is the number of scales under consideration. Here we propose a new approach that allows updating in $n_s \times (n_p + 4n_x)$ operations. The method relies on propagation of probability derivatives. Once the probability derivative map is computed, the actual probabilities are obtained by integration. Let the double derivative of a probability mass p be as follows follows

$$\nabla_x^2 p(x) = \sum_{i,j \in \{-1,1\}} p(x + u_{ij}) \tag{3.17}$$

where $u_{ij} = (i,j)^T$. Thus

$$\nabla_x^2 p(x_{t+1}, s_{t+1} \mid y_{1:t}\lambda_{1:t}) = \sum_{s_t} p(s_{t+1} \mid s_t)$$

$$\sum_{x_t} p(x_t s_t \mid y_{1:t}\lambda_{1:t}) \nabla_x^2 p(x_{t+1} \mid x_t s_t) \tag{3.18}$$

and since $p(x_{t+1} \mid x_t s_t)$ is a square centered at x_t and with height $2s_t$ it follows that

$$\nabla_x^2 p(x_{t+1} \mid x_t s_t) = \begin{cases} 1 & \text{if } x_{t+1} = x_t \pm (1,1)^T \\ -1 & \text{if } x_{t+1} = x_t \pm (-1,1)^T \\ 0 & \text{else} \end{cases} \tag{3.19}$$

Table 3.1 shows the cost of an iteration of the filtering algorithm. By use of cumulative log-likelihood ratio maps and rectangular transition probabilities, the cost is order $n_h + n_p$.

Task	Sum/Diffs	Prods/Ratios	Exps	LLRs	If
CLR	n_p			n_p	
LI	$4n_h$		n_h		
PD	$4n_h + n_p$				
UFD		n_h			
NFD	n_h	n_h			
MAP					n_h
MRS	n_h	1			
MRL	n_h	$n_h + 1$			

Table 3.1: Computational cost per iteration: n_h: Number of hypothesis; n_p: Number of pixels; LLR: Log-likelihood ratio of a single pixel; If: Logical "if" operation; CLR: Cumulative Likelihood Ratio Map; LI: Likelihoods; PD: Predictive Distribution; UFD: Unnormalized Filtering Distribution; NFD: Normalized Filtering Distribution; MAP: Maximum Posterior Hypothesis; MRS: Minimum Risk Scale; MRL: Minimum Risk Location. Log-likelihood ratios (LLR) and exponentials can be implemented via look-up tables.

Thus, to construct the gradient predictive probability map we just need to send four numbers per hypothesis (one for each of the corners of the square centered at the center of the hypothesis). Once the gradient map is built, the predictive probability can be obtained by integrating the map, which costs n_p operations per map and obtaining the value of the integral map at x_{t+1} for scale s_t (see Figure 3.3

$$p(x_{t+1}s_{t+1} \mid y_{1:t}) = \sum_{i=1}^{x_{t+1}(1)} \sum_{j=1}^{x_{t+1}(2)} \nabla_x^2 p(x_{t+1}s_{t+1} \mid y_{1:t}) \tag{3.20}$$

The standard forward filtering recurrence for HMMs requires order n_h^2 operations per time step, where n_h is the number of hypotheses. The use of cumulative probability maps reduces it to $8n_h$ algebraic operations and n_h exponentials. This allows filtering problems with about $100,000$ hypotheses to run in real time on a state of the art PC. In practice this provides more than enough resolution for difficult Tracking problems (see Section 3.6).

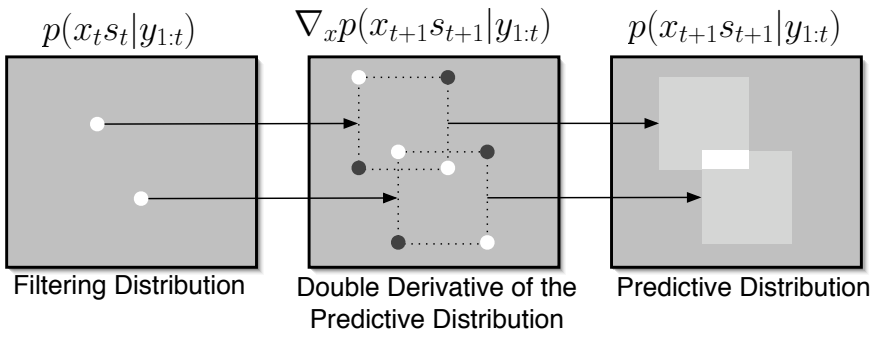


Figure 3.3: The double derivative method for computing the predictive distribution: For each hypothesis from the filtering distribution we add 4 numbers (2 positive and 2 negative) at the corners of the transition probability kernel for that hypothesis. The sum of all these numbers is the double derivative map of the predictive distribution. The double integral of this map gives us the desired predictive distribution.

3.5 Unknown and Non-stationary Model Parameters

The appearance of the background and the object of interest may change due to changes in illumination, camera movement, or the movement of objects in and out of the image plane. Thus we need a scheme to adaptively change the model parameters.

Let $\mathcal{M}$ represent the set of possible image generation models. When the model is unknown and non-stationary, optimal inference calls for marginalizing across all possible image sequence models

$$p(h_t \mid y_{1:t}) = \sum_{\lambda_{1:t}} p(\lambda_{1:t} \mid y_{1:t}) p(h_t \mid y_{1:t} \lambda_{1:t}), \tag{3.21}$$

where $p(\lambda_{1:t} \mid y_{1:t})$ is the parameter adaptation term. In practice we need to approximate this by: finding reasonable estimates $\hat{y}_{1:t}$ and letting $p(\lambda_{1:t} \mid y_{1:t}) \approx \delta(y_{1:t}, \hat{y}_{1:t})$, i.e.,

$$p(h_t \mid y_{1:t}) \approx p(h_t \mid y_{1:t} \hat{\lambda}_{1:t}) \tag{3.22}$$

This approximation while efficient is risky and thus methods are needed to detect when the approximation is not working well and to recover from error.

In our current implementation we rely on an auxiliary object detector whose main role is to help with parameter estimation and error recovery of the primary object tracker. The secondary detector is set to have very small number of false alarms, thus when it detects the object of interest we can safely assume that the object was there. The disadvantage of the auxiliary detector is higher computational cost than the primary detector and the fact that it can only detect the object of interest in a particular pose. In our experiments the auxiliary detector is a Viola & Jones [93] style detector of frontal/upright faces described in [65].

Non Stationary Environments We model changes in illumination, camera movement and background movement as a continuous time Poisson jump process: The background and object models are constant except for specific jump points that occur at unknown random times. The time between jump points is independent of previous jump points and is governed by an exponential density function with parameter θ. At jump points new model parameters are chosen from a distribution of known mean.

Let $T_1, T_2, \cdots$ represent the unknown times at which the object and background model changed (the jump times). Let $S_1, S_2, \cdots$ be the unknown object and background parameters chosen at jump times $T_1, T_2, \cdots$. Since these values are independent samples from a continuous random vector it follows that $P(S_i = S_j) = 0$ if $i \neq j$. Let λ_t represent the unknown color and background models at time t, i.e.,

$$\lambda_t = S_{T_i} \text{ for } T_i \leq t < T_{i+1}. \tag{3.23}$$

Suppose by time t the auxiliary detector has found the object of interest at times $\tau_1 < \tau_2 < \cdots < \tau_n \leq t$. By doing so it provided samples pixels from the background and from the object. Lee $x_i = (h_{\tau_i}, y_{\tau_i})$ represent the information provided by the auxiliary detector at time τ_i. Our goal is to use this information to obtain estimates of λ_t. One reasonable estimate is the posterior mean of λ_t given $x_1 \cdots x_n$. Let A_{n+1}

be the event that at least one jump occurred after τ_n. Thus

$$P(A_{n+1}) = p(\lambda_t \neq \lambda_{\tau_n}) = e^{-\theta(t-\tau_n)}. \tag{3.24}$$

For $j = 2, \cdots n$ let A_j represent the event that the last jump occurred between τ_{j-1} andτ_j. Thus

$$A_j = \cap_{i=j}^{n}\{\lambda_t = \lambda_{\tau_i}\} \cap_{i=1}^{j-1} \{\lambda_t \neq \lambda_{\tau_i}\}. \tag{3.25}$$

That is, the probability that at least a jump occurred between τ_{j-1} and no jump occurred afterwards:

$$P(A_j) = (1 - e^{-\theta(\tau_j - \tau_{j-1})})e^{-\theta(t-\tau_j)} \tag{3.26}$$

$$= e^{-\theta(t-\tau_j)} - e^{-\theta(t-\tau_{j-1})}. \tag{3.27}$$

Finally let A_1 be the event that the last jump occurred prior to τ_1. Thus

$$A_1 = \cap_{i=1}^{n}\{\lambda_t = \lambda_{\tau_i}\} \tag{3.28}$$

$$P(A_1) = e^{-\theta(t-\tau_1)}, \ \sum_{j=1}^{n+1} P(A_j) = 1. \tag{3.29}$$

Thus

$$\hat{\lambda}_t = E(\lambda_t \mid x_{1:n}) = P(A_{n+1})E(\lambda_t) + \sum_{j=1}^{n} P(A_j)E(\lambda_t \mid x_{j:n}\ A_j). \tag{3.30}$$

where $E(\lambda_t \mid x_{j:i}\ A_j) = E(\lambda_t \mid y_{\tau_1}h_{\tau_1} \cdots y_{\tau_i}h_{\tau_i})$ are the object and background histograms obtained by segmenting the images $y_{\tau_1} \cdots y_{\tau_i}$ into object and background as determined by $h_{\tau_1} \cdots h_{\tau_i}$ clumping all the object pixels together and all the background pixels from the same location together. After some algebra it can be shown that $\hat{\lambda}_t$ consists of weighted frequency counts of colors found in the object and the background locations, where the weight of a pixel decays exponentially with the length of time since the pixel was collected.

Error Detection and Recovery The auxiliary object detector may be slow or may run with low priority; thus it may provide information with some delay.

Suppose at time t the auxiliary detector tells us that at time $t-\Delta$ the correct hypothesis was $h_{t-\Delta}$. We also have information that at that time the tracker chose $\hat{h}_{t-\Delta}$. Ideally we should go back in time and propagate forward the new information. However due to the fact that we adapt the model parameters λ based on our knowledge about $h_{t-\Delta}$ this would require buffering the distribution $p(h_{t-\Delta} \,|\, y_1 : y_{t-\Delta} \hat{\lambda}_{t-\Delta})$ and the image sequence $y_{t-\Delta:t}$. If this information is lost by time t, we are presented with the problem of combining two experts whose opinions are derived from different information sources: (1) The auxiliary detector provides us with $p(h_t \mid h_{t-\Delta})$, which does not make any assumptions about λ. However if Δ is large, $p(h_t \mid h_{t-\Delta})$ will be almost flat, and thus uninformative. (2) The main tracker provides us with $\hat{p}(h_t \,|\, y_{1:t} \hat{\lambda}_{1:t})$, which relies on the assumption that $\lambda_{t:t}$ is a good estimate of the actual object and background models. A reasonable approach is to choose the minimum risk expert. The risk for expert 1 is

$$R_1 = \min_h E\left(\rho(H_t, h) \mid h_{t-\Delta}\right). \tag{3.31}$$

The risk for expert 2 is

$$R_2 = E\left(\rho(H_t, \hat{H}_t) \mid h_{t-\Delta}, \hat{h}_{t-\Delta}, \Delta\right). \tag{3.32}$$

If $R_1 < R_2$ we discard the current distribution and restart the system with the distribution $p(h_{t+\Delta} \mid h_t)$ proposed by the auxiliary detector. In practice we model R_2 using some reasonable heuristic (3.34) or by using a labeled dataset in which we estimate how the error of the system changes as a function of time Δ and the starting error $\rho(h_{t-\Delta}, \hat{h}_{t-\Delta})$.

3.6 Simulations

A video tracking simulation was performed on a dataset comprised of five minutes of video. Footage was collected from three subjects. Each subject performed two action sequences consisting of rapid camera movements, in plane translations, rotations, and hand/arm occlusions. The goal was to simulate

the very difficult tracking conditions typically found in pet robots. During the sequence the lighting was changed by adding two blue illumination sources. The resulting video footage was converted to 160x120 color images. The simulation was developed and tested on a 3.0 GHz Pentium 4 computer.

The model is specified by the initial distribution of H, the transition kernel $p(h_{t+1} \mid h_t)$, the average time $1/\theta$ between parameter jumps, and the prior distribution for object and background models, the error function ρ and the risk estimation method. In the experiments presented below we used the following architecture: The initial distribution for H was uniform across hypotheses, the transition kernel was uniform rectangular with width and height equal to 1/2 the scale of the parent hypothesis. The average time between model jumps was set at 5 seconds (i.e., $\theta = 0.2$ seconds. The face color model for the tracker was implemented as a twenty-bin histogram. The prior distribution for the background and object models was assumed to be flat. The prior histogram model for faces was based on the model published in Jones [47]. For risk estimation we used the following:

$$R_1 = \min_h E(\rho(H_t, h) \mid h_{t-\Delta}) = \max_{h_t} p(h_t \mid h_{t-\Delta}) \tag{3.33}$$

$$E\left(\rho(H_t, \hat{H}_t) \mid h_{t-\Delta}, \hat{h}_{t-\Delta}, \Delta\right) \approx p(h_{t-\Delta} \mid y_{1:t-\Delta}\lambda_{1:t-\Delta}). \tag{3.34}$$

The system could run in real time with a space of 100,000 hypotheses.

3.7 Previous Work

The use of double integral functions to measure rectangular sets is well known in measure theory, probability theory, and statistics. This approach is also used in computer animation for fast rendering of rectangular objects. Viola and Jones [93] popularized this method in computer vision. Stochastic filtering approaches to tracking have been popular for more than a decade in the computer vision community. Most approaches nowadays find approximate solutions to the filtering problem using Monte-Carlo methods (particle filters). Monte-Carlo approaches

Figure 3.4: Evolution of priors, likelihoods, and posteriors: The two rows of images represent hypotheses at two different scales. The left column represents the most likely hypotheses. The center image represents the prior distribution based on the previous image, the right-sice column shows the posterior distribution of hypotheses.

Figure 3.5: An example of uncertainty propagation: The left side shows the most probable hypotheses at time t, i.e., the filtering distribution. The image on the left shows the predictive distribution for time $t + 1$, i.e, the prior distribution for the nexst time step.

to filtering were first described in [36] and introduced to the computer vision community by [44]. Current work in this field has focused on the development of intelligent sampling methods to find good approximations with very few particles [1, 92, 87]. The discretization of hypothesis space has been explored previously, for instance in grid-based methods for map-building by robots, [86], or in models for tracking objects [46]. As far as we know this is the first work to point out that the double integral method can be used to efficiently compute likelihood-ratio maps, and that double derivative maps can be use to efficiently compute predictive probabilities in such discretized hypothesis spaces.

3.8 Extensions

While the specific architecture presented here is limited to upright rectangular hypotheses, additive log-likelihood functions, and rectangular transition probability kernels, extensions are possible that preserve the computational complexity of the method while providing great generality.

Complex geometries can be obtained by producing double cumulative functions at several orientations. The transition kernels or the features underlying the likelihood computation do not need to be uniform. For example, one can apply double cumulative sums recursively (cumulative sums of cumulative sums ...) to obtain Gaussian-like transition kernels and Gabor-like features. More complex likelihood-ratio functions are also possible, using such kernels as object windowing functions.

In this chapter we treat objects a single blobs. Multi-part objects can also be tracked using as many likelihood-ratio maps as object parts. Coarse histogram matching can also be done using one likelihood-ratio map per histogram bin.

3.9 A General Architecture for Machine Perception

We described methods for solving the stochastic filtering problem in order n_h operations. This allows us to work with tens of thousands of hidden states, and solve large-scale non-linear filtering problems exactly in real time. While we focused on visual tracking problems, the methods proposed here can be used in a wide variety of real-time machine perception and robotics problems.

The algorithms presented in this chapter exploit the fact that in many computer vision problems the computation of likelihoods and the propagation of probabilities are convolutional. The methods presented here solve these convolutions in order n_h operations. Frequency domain methods could also be used, which would work in order $n_p \log n_p$ operations.

HMMs are already the architecture of choice for speech recognition problems. Working with a similar architecture in vision facilitates approaching problems that require combination of acoustic and visual information (e.g., audio-visual tracking, audiovisual speech recognition). Particle filter approximations are being used in robotics for real-time inference and control problems[85, 87]. The architecture presented here may allow these problems to be solved more efficiently.

The proposed architecture shows a surprising resemblance to the functional architecture of visual cortex: A set of topographical organized hyper-columns, where each hyper-column has scaled and rotated replicas of the same detectors. The hyper-columns are interconnected by lateral connections (see Figure 3.6). The short distance lateral connections in the filtering algorithm take care of propagation of probability maps. These maps represent the posterior distribution of hypotheses given an observed video sequence. We can now efficiently simulate such systems on a grand scale.

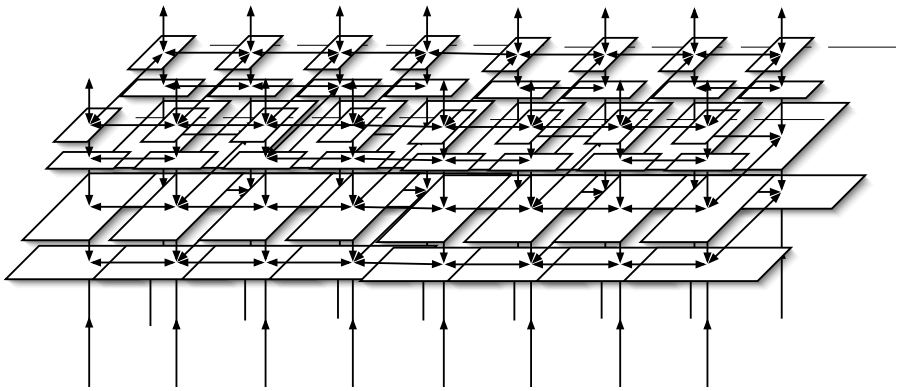


Figure 3.6: Topographical implementation of the algorithm: The optimal inference algorithm can be implemented using a topographically organized set of columns, where each column computes the likelihood ratio of rotated and scaled versions of an image patch.

3.10 Acknowledgements

The contents of this chapter are adapted from [63] which was published in Computer Vision and Pattern Recognition (CVPR) in 2004. My co-authors, Javier Movellan and Josh Susskind, supervised and collaborated with me on the research which forms the basis of this chapter.

Chapter 4

G-Flow: A Generative Model for Fast Tracking Using 3D Deformable Models

Abstract

We present a generative model (*G-flow*) and inference algorithm for simultaneous tracking of 3D pose, non-rigid motion, object texture and background texture. Under this model inference about pose and texture can be performed efficiently using a bank of Kalman filters for texture whose parameters are updated by an optic-flow-like algorithm. The inference algorithm unifies optic flow-based and texture-based tracking methods, dynamically adjusting the relative importance of each component in a principled manner. Classic optic flow and template-based algorithms emerge as special cases, and the conditions under which they are optimal are elucidated by the model. For instance, the Lucas-Kanade optic-flow algorithm is a special case that is optimal under certain conditions (complete certainty of the current location of the object in each frame, and knowledge of its texture only via its current location).

4.1 Introduction

Many approaches have been proposed in the computer vision literature to solve the object tracking problem. In general these can be divided into motion-based and template-based approaches. Motion-based approaches compute local estimates of optic flow, typically using a variation of the Lucas-Kanade optic-flow algorithm [58], then combine these estimates using global object constraints [14]. The advantage of a motion-based approach is that it makes few assumptions about the appearance of the object being tracked. When given two images y_t, y_{t+1} at two consecutive time steps, and the position of the object at time t, the approach gives us an estimate of the position of the object at time $t+1$. This method implicitly assumes good knowledge about the location of the object at each time step, and thus it has a tendency to drift as errors accumulate. Initialization and recovery from drift are open issues in motion-based approaches, and they are typically handled using heuristic methods.

At the other end of the spectrum template approaches assume good knowledge about the appearance of the object of interest. The advantage of these approaches is that they require little knowledge about the current location of the object, provided the template is correct. Local or global search methods are then used to find the pose that best fits the image plane. A known problem with template-based approaches is dealing with realistic sources of variation (pose, illumination, identity, expression, etc). Template-based methods typically rely on heuristics that allow for dynamic updating of the templates and periodic re-registration.

In practice, the issues of model initialization, dynamic update of templates, error detection, and re-initialization are still unsolved. Finding principled solutions to these problems is arguably the most important impediment to the widespread application of computer vision technology in daily life.

In this chapter, we present a generative model (G-flow) for video sequences. The model, while relatively simple, provides a rich framework for analyzing the problem

of how to dynamically combine motion-based and texture-based information in an optimal manner. A contribution of the model is that classic optic flow and template-based algorithms emerge as special cases of optimal inference under limited conditions. Optic flow is optimal when the location of the object is known and its appearance is unknown. Template-based algorithms are optimal in the opposite case. In practice optimal inference under G-flow comprises a combination of motion and template-based information that is dynamically re-weighted as new images are presented. Standard approximations can be used to solve the inference problem very quickly, allowing for on line, real time 3D pose and expression tracking, geometry estimation, and texture recovery.

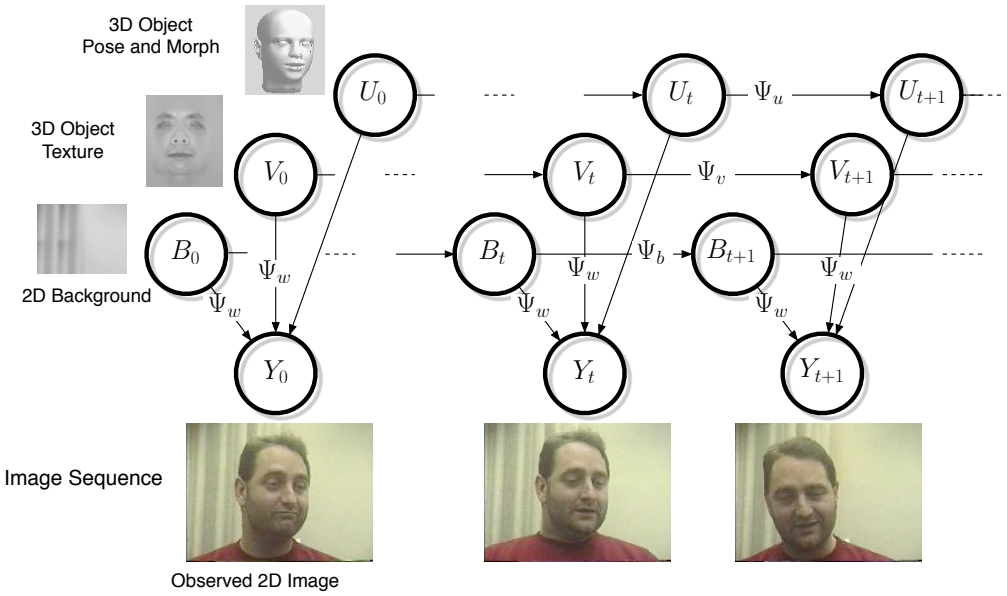


Figure 4.1: The G-flow video generative model: The pose and texture of the object in 3D are projected onto 2D and then combined with the background to generate the observed video sequence. The model parameters include the initial distributions π_u, π_v, π_b, the texture transition certainties $\Psi_v \Psi_b$, the rendering noise parameter Ψ_w and the pose transition probabilities $p(u_t \mid u_{t-1})$. Except for the pose transition probabilities, the distributions controlled by these parameters are assumed Gaussian. The goal is to make inferences about (U_t, V_t, B_t) based on the observed video sequence $Y_1 \cdots Y_t$.

4.2 Video generation model

Unless otherwise stated capital letters represent random variables, small letters represent specific values taken by random variables, and Greek letters represent fixed model parameters. When possible we use informal shorthand notation and identify probability functions by their arguments. For example, $p(x, y)$ is shorthand for the probability (or probability density) that the random variable X takes the specific value x and the random variable Y takes the value y. We use subscripted columns to indicate sequences. For example $X_{1:t} = X_1 \cdots X_t$. The term I_p stands for a $p \times p$ unit matrix. E stands for expected value, Var for covariance matrix and Var^{-1} for precision matrix, the inverse of the covariance matrix. The notation $A^n \otimes A^c$ refers to the set of $n \times c$ matrices whose elements are in the set A. The following terms will be used throughout the chapter:

- $y_t \in \mathbb{R}^p$, the vectorized version of an image with p pixels.
- $u_t \in \mathbb{R}^{2n}$ a vector containing the position of n points on the image plane. These n points are thought to belong to the same object, the rest of the points on the image plane belong to the background.
- $v_t \in \mathbb{R}^n$, $b_t \in \mathbb{R}^p$ vectors with the texture map of the object and background respectively. We refer to each element of v_t and b_t as a *texel*.
- $a_v : \mathbb{R}^{2n} \rightarrow \{0, 1\}^p \otimes \{0, 1\}^n$, a function whose input is the position of the object points on the image plane and whose output is a $p \times n$ matrix of zeroes and ones. If there is a one at row i, column j it means that the j^{th} object point projects on pixel i. There should be a total of n ones and at most a one per row.
- $a_b : \mathbb{R}^{2n} \rightarrow \{0, 1\}^p \otimes \{0, 1\}^p$ is a function whose input is the position of the object points on the image plane and whose output is a $p \times p$ diagonal matrix. If there is a one at row j, column j it means that the background texel j projects on pixel j on the image plane. We put the constraint that if $a_{vij} = 1$

then $a_{bjj} = 0$, i.e., if a pixel i is rendered by the object, it is not rendered by the background.

The functions a_v, a_b encapsulate the projection model and filtering effects of the imaging device.

Example: Suppose we have a 4-pixel image plane, $p = 4$, and a 2-point object $n = 2$. Suppose the object can only take 2 locations in 3D: $q_1 = (-1, 0, 1)$ $q_2 = (1, 0, 1)$. When at q_1 the object projects onto the two pixels on the left. When at q_2 it projects on the two pixels on the right.

$$a_v(q_1) = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} \; a_v(q_2) = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{pmatrix}$$

$$a_b(q_1) = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \; a_b(q_2) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

Model Specification: G-flow models the video sequence as a stochastic process governed by a partially observable difference equation (see Figure 4.1 and 4.2). There are three hidden processes: A background process B, an object motion process U, and an object texture process V. They generate images as follows: The object pose, U_t determines which pixels the object and background project on, which we formulate using the projection function $c(U_t) = \big(a_v(U_t), a_b(U_t)\big)$. The object and background textures V_t and B_t then project to the image Y_t via $c(U_t)$ with additive noise as formulated below:

$$Y_t = c(U_t) \begin{pmatrix} V_t \\ B_t \end{pmatrix} + W_t, \quad \text{for } t = 1, \cdots \tag{4.1}$$

The system dynamics are as follows:

$$\begin{array}{ll} U_t \sim p(u_t \mid u_{t-1}) & \text{for } t = 2, \cdots \\ V_t = V_{t-1} + Z^v_{t-1} & \text{for } t = 2, \cdots \\ B_t = B_{t-1} + Z^b_{t-1} & \text{for } t = 2, \cdots \end{array}$$

$p(u_t|u_{t-1})$ is the pose transition distribution, Z^v, Z^b, W are sequences of zero mean, Gaussian processes independent of each other and of the initial conditions. Their respective precision matrices are Ψ_v, Ψ_b, Ψ_w. The form of the pose distribution is left unspecified for the sake of generality. Because the image generation process is nonlinear as a function of pose, our methods must accommodate this nonlinearity anyway, and hence we need not restrict the motion dynamics to a Gaussian form.

The model is specified by the following terms: (1) Initial conditions, which consist of a distribution for the object position U_1, and Gaussian distribution of object and background texture, V_1 and B_1, all of which are independent of each other. In addition we assume the variance of V_1 is diagonal and the variance of B_1 is a scalar times a unit matrix. (2) The precision matrices for the state transitions, Ψ_v, Ψ_b. (3) The pose transition distribution $p(u_t \mid u_{t-1})$. (4) The precision matrix for the image rendering noise is of the form $\Psi_w = I_p \sigma_w^{-1}$, where σ_w is a scalar. The imaging model (e.g., perspective projection) determines the functions a_v and a_b.

Structure of the Inference Problem: Inference requires computing the distribution of pose and texture given an observed sequence of images. The main difficulty in solving this problem centers around the motion posterior U_t. Since the object and background textures are not a linear function of the position of the pixel, then the observed images Y_t will in general not be a linear function of U_t. However, if $U_{1:t}$ were known then the object and background texture processes V_t, B_t would be linear and Gaussian and thus could be solved using Kalman filter equations with time variant parameters, as determined by $U_{1:t}$. This suggests the following scheme: Use approximate methods to obtain highly probable samples, of $U_{1:t}$, then use Kalman filtering equations to determine the distribution of $V_{1:t}B_{1:t}$

for each sample. Another important aspect of the problem, that we want to use to our advantage, is that the observed images have a strong spatio-temporal structure.

4.3 Filtering Distribution

Our goal is to find an expression for the filtering distribution $p(u_t v_t b_t \mid y_{1:t})$, for $t = 0, \cdots$. Using the law of total probability we have that

$$p(u_t v_t b_t \mid y_{1:t}) = \int p(u_t v_t b_t u_{1:t-1} \mid y_{1:t}) du_{1:t-1} \tag{4.2}$$

$$= \int p(u_t v_t b_t \mid u_{1:t-1} y_{1:t}) p(u_{1:t-1} \mid y_{1:t}) du_{1:t-1} \tag{4.3}$$

We can think of the first term $p(u_t v_t b_t \mid u_{1:t-1} y_{1:t})$ as *the opinion* about u_t, v_t, b_t of an expert that believes in the past the object was at $u_{1:t-1}$. The second term of the equation $p(u_{1:t-1} \mid y_{1:t})$ is the *credibility* of that expert.

4.3.1 The Opinion Equations

The opinion of expert $u_{1:t-1}$, can be written as the product of the opinion about pose U_t times the opinion about texture V_t, B_t given pose.

$$p(u_t v_t b_t \mid u_{1:t-1} y_{1:t}) = p(v_t, b_t \mid u_{1:t} y_{1:t}) p(u_t \mid u_{1:t-1} y_{1:t}) \tag{4.4}$$

Texture opinions: Because V_1, B_1 are Gaussian, the distribution of $V_t B_t$ given $u_{1:t-1} y_{1:t-1}$ is also Gaussian with a mean and covariance that can be obtained using time dependent Kalman estimation equations (a.k.a. the correction equations)

$$Var^{-1}(V_t, B_t \mid u_{1:t}, y_{1:t}) = Var^{-1}(V_t, B_t \mid u_{1:t-1}, y_{1:t-1}) + c(u_t)' \Psi_w c(u_t) \tag{4.5}$$

$$E(V_t, B_t \mid u_{1:t}, y_{1:t}) = Var(V_t, B_t \mid u_{1:t}, y_{1:t}) \big[Var^{-1}(V_t, B_t \mid u_{1:t-1}, y_{1:t-1}) E(V_t, B_t \mid u_{1:t-1}, y_{1:t-1}) + c(u_{t-1})' \Psi_w y_{t-1} \big] \tag{4.6}$$

This requires the distribution of V_t, B_t given $u_{1:t-1}, y_{1:t-1}$, which can be obtained using the Kalman prediction equations

$$\begin{aligned} E(V_t, B_t \mid u_{1:t-1}, y_{1:t-1}) &= E(V_{t-1}, B_{t-1} \mid u_{1:t-1}, y_{1:t-1}) \\ Var(V_t, B_t \mid u_{1:t-1}, y_{1:t-1}) &= Var(V_{t-1}, B_{t-1} \mid u_{1:t-1}, y_{1:t-1}) \\ &\quad + \begin{pmatrix} \Psi_v^{-1} & 0 \\ 0 & \Psi_b^{-1} \end{pmatrix} \end{aligned} \tag{4.7}$$

Note the expected value $E(V_t, B_t \mid u_{1:t}, y_{1:t})$ contains texture maps (templates) for the object and background. $Var(V_t, B_t \mid u_{1:t}, y_{1:t})$ keeps the degree of uncertainty about the object and background templates. Due to the fact that pixels cannot be simultaneously rendered by the object and background, i.e., $a_{vij}(u_t) = 1 \rightarrow a_{bjj}(u_t) = 0$, and a_v is a permutation matrix, and a_b is diagonal, it can be shown that $Var(V_t B_t \mid u_{1:t} y_{1:t})$ has the same structure as $Var(V_0 B_0)$, i.e., it is diagonal, and the variances of all the B_t elements given $u_{1:t}, y_{1:t}$ are equal.

Pose Opinions: The projection function $c(u_t)$ determines how the object and background templates render the image plane, i.e., which pixels are rendered by the object and which are rendered by the background. Since the effect of u_t on the likelihood function is non-linear, we will not attempt to find an analytical solution for the pose opinion equations. Instead we will find the most probable value of u_t, given $u_{1:t-1}, y_{1:t}$ for each expert and approximate the distribution as a Gaussian bump about that point. Note

$$\begin{aligned} p(u_t \mid u_{1:t-1}, y_{1:t}) &= \frac{p(y_{1:t-1} \mid u_{1:t-1})}{p(y_{1:t} \mid u_{1:t-1})} \, p(u_t \mid u_{t-1}) \\ &\quad p(y_t \mid u_{1:t}, y_{1:t-1}) \end{aligned} \tag{4.8}$$

where

$$\begin{aligned} &p(y_t \mid u_{1:t}, y_{1:t-1}) = \\ &\quad \int p(v_t, b_t \mid u_{1:t-1}, y_{1:t-1}) p(y_t \mid u_t, v_t, b_t) dv_t db_t \end{aligned} \tag{4.9}$$

using the fact that V_t, B_t are independent of U_t given $u_{1:t-1}, y_{1:t-1}$, i.e.,

$$\begin{aligned}
p(u_t, v_t, b_t \mid u_{1:t-1}, y_{1:t-1}) &= \int p(v_{t-1}, b_{t-1} \mid u_{1:t-1}, y_{1:t-1}) \\
&\quad p(u_t, v_t, b_t \mid u_{1:t-1}, v_{t-1}, b_{t-1}) dv_{t-1} db_{t-1} \\
&= \int p(v_{t-1}, b_{t-1} \mid u_{1:t-1}, y_{1:t-1}) p(u_t \mid u_{t-1}) \\
&\quad p(v_t, b_t \mid v_{t-1}, b_{t-1}) dv_{t-1} db_{t-1} \\
&= p(u_t \mid u_{1:t-1}, y_{1:t-1}) \int p(v_{t-1}, b_{t-1} \mid u_{1:t-1}, y_{1:t-1}) \\
&\quad p(v_t, b_t \mid v_{t-1}, b_{t-1}, u_{1:t-1}, y_{1:t-1}) dv_{t-1} db_{t-1} \\
&= p(u_t \mid u_{1:t-1}, y_{1:t-1}) p(v_t, b_t \mid u_{1:t-1}, y_{1:t-1}) \qquad (4.10)
\end{aligned}$$

We saw in the previous section that $p(v_t, b_t \mid u_{1:t-1}, y_{1:t-1})$ is Gaussian. Since $p(y_t \,|\, u_t, v_t, b_t)$ is also Gaussian it follows that $p(y_t \,|\, u_{1:t}, y_{1:t-1})$ is Gaussian with the following mean and variance:

$$E(Y_t \mid u_{1:t}, y_{1:t-1}) = c(u_t) E(V_t, B_t \mid u_{1:t-1} y_{1:t-1}) \qquad (4.11)$$

$$\begin{aligned}
&Var(Y_t \mid u_{1:t}, y_{1:t-1}) = \Psi_w^{-1} \\
&+ c(u_t) Var(V_t, B_t \mid u_{1:t-1}, y_{1:t-1}) c(u_t)' \qquad (4.12)
\end{aligned}$$

Let $\mathcal{O}(u_t)$ be an ordered set of indices to the pixels rendered by the object according to u_t. For $i \in \mathcal{O}(u_t)$ let $\mu_v(u_{1:t}, i)$ be the texel from the object texture map $E(V_t \mid u_{1:t-1}, y_{1:t-1})$, that renders the image pixel i as determined by u_t. Let $\sigma_v(u_{1:t}, i)$ be the variance of that texel. For $j \notin \mathcal{O}(u_t)$ let $\mu_b(u_{1:t}, j)$ be the texel from the background texture map $E(B_t \,|\, u_{1:t-1}, y_{1:t-1})$, that renders the image pixel j as determined by u_t, and let $\sigma_b(u_{1:t}, j)$ the variance of that texel. It follows that

$$\begin{aligned}
&\log p(y_t \mid u_{1:t}, y_{1:t-1}) = -\frac{1}{2} \log |Var(Y_t \mid u_{1:t}, y_{1:t-1})| \\
&- \frac{1}{2} \sum_{i \in \mathcal{O}(u_t)} \frac{(y_t(i) - \mu_v(u_{1:t}, i))^2}{\sigma_v(u_{1:t}, i) + \sigma_w} \\
&- \frac{1}{2} \sum_{j \notin \mathcal{O}(u_t)} \frac{(y_t(i) - \mu_b(u_{1:t}, i))^2}{\sigma_b(u_{1:t}, i) + \sigma_w} \qquad (4.13)
\end{aligned}$$

Moreover u_t simply permutes $Var(Y_t \mid u_{1:t}, y_{1:t-1})$ and $E(Y_t \mid u_{1:t}, y_{1:t-1})$. Thus $|Var(Y_t \mid u_{1:t}, y_{1:t-1})|$ is constant with respect to u_t. Let

$$\hat{u}_t(u_{1:t-1}) = \underset{u_t}{\operatorname{argmax}}\ p(u_t \mid u_{1:t-1}, y_{1:t}) \tag{4.14}$$

Thus

$$\begin{aligned}\hat{u}_t(u_{1:t-1}) &= \underset{u_t}{\operatorname{argmax}}\, p(u_t \mid u_{t-1}) p(y_t \mid u_{1:t}, y_{1:t-1}) \\ &= \underset{u_t}{\operatorname{argmin}}\, \frac{1}{2} \sum_{i \in \mathcal{O}(u_t)} \frac{(y_t(i) - \mu_v(u_{1:t}, i))^2}{\sigma_v(u_{1:t}, i) + \sigma_w} \\ &+ \frac{1}{2} \sum_{j \notin \mathcal{O}(u_t)} \frac{(y_t(i) - \mu_b(u_{1:t}, i))^2}{\sigma_b(u_{1:t}, i) + \sigma_w} - \log p(u_t \mid u_{t-1})\end{aligned}$$

Moreover, since

$$\begin{aligned}\sum_{j \notin \mathcal{O}(u_t)} \frac{(y_t(i) - \mu_b(u_{1:t}, i))^2}{\sigma_b(u_{1:t}, i) + \sigma_w} &= \sum_j \frac{(y_t(i) - \mu_b(u_{1:t}, i))^2}{\sigma_b(u_{1:t}, i) + \sigma_w} \\ - \sum_{j \in \mathcal{O}(u_t)} \frac{(y_t(i) - \mu_b(u_{1:t}, i))^2}{\sigma_b(u_{1:t}, i) + \sigma_w}&\end{aligned} \tag{4.15}$$

and $\sum_j \frac{(y_t(i) - \mu_b(u_{1:t}, i))^2}{\sigma_b(u_{1:t}, i) + \sigma_w}$ is constant with respect to u_t, it follows that

$$\begin{aligned}\hat{u}_t(u_{1:t-1}) = \underset{u_t}{\operatorname{argmin}}\, \frac{1}{2} \sum_{i \in \mathcal{O}(u_t)} \Bigg(&\frac{(y_t(i) - \mu_v(u_{1:t}, i))^2}{\sigma_v(u_{1:t}, i) + \sigma_w} \\ &- \frac{(y_t(i) - \mu_b(u_{1:t}, i))^2}{\sigma_b(u_{1:t}, i) + \sigma_w} \Bigg) - \log p(u_t \mid u_{t-1})\end{aligned} \tag{4.16}$$

$\hat{u}_t(u_{1:t-1})$ can be found very quickly using a Gauss-Newton method. The inverse Hessian $\hat{\sigma}_t(u_{1:t-1})$ also falls out easily from the Gauss-Newton method. The posterior distribution can then be approximated as a Gaussian $g(\cdot \mid \hat{u}_t(u_{1:t-1}), \hat{\sigma}_t(u_{1:t-1}))$ centered at $\hat{u}_t(u_{1:t-1})$ and with variance $\hat{\sigma}_t(u_{1:t-1})$.

Optic Flow as a Special Case: Suppose $p(u_t \mid u_{t+1})$ is uninformative, the background is a white noise process, i.e. $\sigma_b(u_t, i) \to \infty$ for all t, i and by time $t-2$ we are completely uncertain about the object texture, i.e.

$$Var(V_{t-1} \mid u_{1:t-2}, y_{1:t-2}) \to \infty \tag{4.17}$$

It follows that

$$E(V_t \mid u_{1:t-1}, y_{1:t-1}) = a_v(u_{t-1})y_{t-1} \tag{4.18}$$

i.e, our object texture map at time t is determined by the pixels from y_{t-1} that according to u_{t-1} are rendered by the object. Thus

$$\begin{aligned} &\underset{u_t}{\text{argmax}}\ p(u_t \mid u_{t-1}, y_{1:t}) = \\ &= \underset{u_t}{\text{argmin}} \sum_{i \in \mathcal{O}(u_t)} \frac{\left(y_t(i) - a_v(u_{t-1})y_{t-1}(i)\right)^2}{\sigma_v(u_t, i) + \sigma_w} \end{aligned} \tag{4.19}$$

The most probable u_t is that which minimize the mismatch between the image pixels rendered by the object at time $t-1$ and the image at y_t shifted according to u_t. The Lucas-Kanade optic flow algorithm is simply the Newton-Gauss method as applied to minimize this error function.

Template matching as a Special Case: If $p(u_t \mid u_{t-1})$ is uninformative, the background is a white noise process and by time $t-2$ we are certain about the object texture map, i.e., $Var(V_{t-1} \mid u_{1:t-2}, y_{1:t-2}) = 0$, then

$$E(V_t \mid u_{1:t-1}, y_{1:t-1}) = E(V_t \mid u_{1:t-2}, y_{1:t-2}) \tag{4.20}$$

and

$$\begin{aligned} &\underset{u_t}{\text{argmax}}\ p(u_t \mid u_{t-1}, y_{1:t}) = \\ &= \underset{u_t}{\text{argmin}} \sum_{i \in \mathcal{O}(u_t)} \frac{\left(y_t(i) - \mu_v(u_t, i)\right)^2}{\sigma_w} \end{aligned} \tag{4.21}$$

where $\mu_v(u_t, i)$ is simply the fixed object template, shifted by u_t. This is the error function minimized by standard template match algorithms.

General Case: In general minimizing (4.16) results in a weighted sum of optic flow and template matching, with the weight of each approach depending on the certainty about the object template.

Importance Sampling: Suppose we are given a set of pose sequences $\{u^{(i)}_{1:t-1} : i = 1 \cdots n_{t-1}\}$. For each of these sequences we can obtain unbiased statistics from $p(u_t \mid u_{1:t-1}y_{1:t})$ using importance sampling [27]. We generate a set of independent samples $\{u^{(i,j)}_t : j = 1 \cdots s^{(i)}_t\}$ from a Gaussian distribution centered at $\hat{u}_t(u^{(i)}_{1:t-1})$ with variance proportional to $\hat{\sigma}_t(u^{(i)}_{1:t-1})$ and assign each sample a weight proportional to the ratio between the sampling distribution and the posterior distribution:

$$\hat{p}(u_t \mid u^{(i)}_{1:t-1}, y_{1:t}) = \sum_{j=1}^{s^{(i)}_t} \delta(u_t - u^{(i,j)}_t) \frac{w_t(i,j)}{\sum_{k=1}^{s^{(i)}_t} w_t(i,k)} \tag{4.22}$$

$$w_t(i,j) = \frac{p(u^{(i,j)}_t \mid u^{(i)}_{t-1}) p(y_t \mid u^{(i)}_{1:t-1} u^{(i,j)}_t y_{1:t-1})}{g(u^{(i,j)}_t \mid \hat{u}_t(u^{(i)}_{1:t-1}), \alpha\hat{\sigma}_t(u^{(i)}_{1:t-1}))} \tag{4.23}$$

where $\hat{p}$ stands for an unbiased estimate of the corresponding probability term and $\alpha > 0$ is a parameter that determines the sharpness of the sampling distribution. As $\alpha \to 0$ we simply choose $\hat{u}_t(u_{1:t-1})$, the state that maximizes the posterior probability $p(u_t \mid u_{1:t-1}, y_{1:t})$.

4.3.2 Credibility Equations

The credibility of the expert $u^{(i)}_{1:t-1}$ is proportional to the product of a prior term and a likelihood term

$$p(u^{(i)}_{1:t-1} \mid y_{1:t}) = \frac{p(u^{(i)}_{1:t-1} \mid y_{1:t-1}) p(y_t \mid u^{(i)}_{1:t-1}, y_{1:t-1})}{p(y_t \mid y_{1:t-1})} \tag{4.24}$$

In Section 4.3.3 we explain how to obtain running estimates for the prior $p(u^{(i)}_{1:t-1} \mid y_{1:t-1})$. Regarding the likelihood, note that

$$\begin{aligned} p(y_t \mid u_{1:t-1}, y_{1:t-1}) &= \int p(y_t u_t \mid u_{1:t-1}, y_{1:t-1}) du_t \\ &= \int p(y_t \mid u_{1:t}, y_{1:t-1}) p(u_t \mid u_{t-1}) du_t \end{aligned} \tag{4.25}$$

We already generated a set of samples $\{u_t^{(i,j)} : j = 1 \cdots s_t^{(i)}\}$ from $p(u_t \mid u_{1:t-1}^{(i)} y_{1:t})$. We can now use these samples to obtain an unbiased estimate of the likelihood

$$p(y_t \mid u_{1:t-1}^{(i)}, y_{1:t-1}) = \int p(y_t \mid u_{1:t-1}^{(i)}, u_t, y_{1:t-1}) p(u_t \mid u_{t-1}^{(i)}) du_t$$
$$= \int p(y_t \mid u_{1:t-1}^{(i)}, u_t, y_{1:t-1}) g(u_t \mid \hat{u}_t(u_{1:t-1}^{(i)}), \hat{\sigma}_t(u_{1:t-1}^{(i)}))$$
$$\frac{p(u_t \mid u_{t-1}^{(i)})}{g(u_t \mid \hat{u}_t(u_{1:t-1}^{(i)}), \hat{\sigma}_t(u_{1:t-1}^{(i)})} du_t \approx \frac{\sum_{j=1}^{s_t^{(i)}} w_t(i,j)}{s_t^{(i)}} \tag{4.26}$$

If we only sample the most probable state $\hat{u}_t(u_{1:t-1})$ then the likelihood is approximated by the maximum value of the integrand.

4.3.3 Combining Opinion and Credibility

Opinion and credibility can be combined to obtain running estimates of the filtering distribution.

Initialization:

- Obtain n_1 samples $\{u_1^{(i)} : i = 1 \cdots n_1\}$ from $p(u_1)$. We refer to these samples as experts. For each expert the initial Gaussian prior distributions $p(v_1, b_1 \mid u_1^{(i)}) = p(v_1, b_1)$ are given as part of the model specification. The relative weight of the i^{th} expert, $r_1^{(i)}$ is set proportional to the probability of the image given the expert

$$r_1^{(i)} \propto p(y_1 \mid u_1^{(i)}) \tag{4.27}$$

  and the weights are normalized to add up to one. This provides a Monte-Carlo estimate of the filtering distribution at the first time step:

$$\hat{p}(u_1 \mid y_1) = \sum_{i=1}^{n_1} r_{t-1}^{(i)} \delta(u_1 - u_1^{(i)}) \tag{4.28}$$

Update:

- By time $t-1$ we are given n_{t-1} pose experts $\{u^{(i)}_{i:t-1} : i = 1 \cdots n_{t-1}\}$. Each expert $u^{(i)}_{1:t-1}$ comes with a relative weight $r^{(i)}_t$ and with the mean and variance of the filtering distribution for texture given that expert, i.e., $E(V_{t-1}, B_{t-1} \mid u^{(i)}_{1:t-1}, y_{1:t-1})$, $Var(V_{t-1}, B_{t-1} \mid u^{(i)}_{1:t-1}, y_{1:t-1})$. The weights provide an estimate of the filtering distribution for pose at time $t-1$, which serves as the prior for time t

$$\hat{p}(u_{1:t-1} \mid y_{1:t-1}) = \sum_{i=1}^{n_{t-1}} r^{(i)}_{t-1} \delta(u_{1:t-1} - u^{(i)}_{1:t-1}). \tag{4.29}$$

For each expert, we compute the most probable pose $\hat{u}_t(u^{(i)}_{1:t1})$ and estimate the uncertainty about that pose $\hat{\sigma}_t(u^{(i)}_{1:t-1})$.

Based on the distribution of relative weights $\{r^{(i)}_{t-1} : i = 1 \cdots n_{t-1}\}$ we assign a number of descendants to each expert. This is usually known as a resampling step in the particle filtering literature [27], which discusses the pros and cons of different resampling rules. Suppose the resampling rule assigns $s^{(i)}_t$ descendants to expert i. We then generate as many independent samples $\{u^{(i,j)}_t : j = 1 \cdots s^{(i)}_t\}$ from the distribution $g(\cdot \mid \hat{u}_t(u^{(i)}_{1:t-1}), \hat{\sigma}_t(u^{(i)}_{1:t-1}))$, and compute the importance weight of each sample $w_t(i,j)$. This provides an estimate for the opinion

$$\hat{p}(u_t \mid u^{(i)}_{1:t-1}, y_{1:t}) = \sum_{j=1}^{s^{(i)}_t} \delta(u_t - u^{(i,j)}_t) \frac{w_t(i,j)}{\sum_{k=1}^{s^{(i)}_t} w_t(i,k)} \tag{4.30}$$

and for the likelihood of each expert

$$\hat{p}(y_t \mid u^{(i)}_{1:t-1}) = \sum_{j=1}^{s^{(i)}_t} \frac{w_t(i,j)}{s^{(i)}_t} \tag{4.31}$$

The likelihood times the prior gives us the credibility of each expert

$$\hat{p}(u^{(i)}_{1:t-1} \mid y_{1:t}) \propto \frac{r^{(i)}_{t-1}}{s^{(i)}_t} \sum_{j=1}^{s^{(i)}_t} w_t(i,j) \tag{4.32}$$

From this we obtain $\hat{p}(u_{1:t} \mid y_{1:t})$,

$$\hat{p}(u_{1:t} \mid y_{1:t}) = \int \hat{p}(u_{1:t-1} \mid y_{1:t}) \hat{p}(u_t \mid u_{1:t-1}, y_{1:t}) du_{1:t-1} \tag{4.33}$$

$$= \sum_{i=1}^{n_{t-1}} \frac{\frac{r_{t-1}^{(i)}}{s_t^{(i)}} \sum_{j=1}^{s_t^{(i)}} w_t(i,j)}{\sum_{k=1}^{n_{t-1}} \frac{r_{t-1}^{(k)}}{s_t^{(k)}} \sum_{l=1}^{s_t^{(k)}} w_t(k,l)} \delta(u_{1:t-1} - u_{1:t-1}^{(i)}) \sum_{m=1}^{s_t^{(i)}} \delta(u_t - u_t^{(i,m)}) \frac{w_t(i,m)}{\sum_{n=1}^{s_t^{(i)}} w_t(i,n)}$$

$$= \sum_{i=1}^{n_{t-1}} \sum_{j=1}^{s_t^{(i)}} \delta(u_{1:t} - u_{1:t}^{(i)} u_t^{(i,j)}) \frac{\frac{r_{t-1}^{(i)}}{s_t^{(i)}} w_t(i,i)}{\sum_{k=1}^{n_{t-1}} \sum_{l=1}^{s_t^{(k)}} \frac{r_{t-1}^{(k)}}{s_{t-1}^{(k)}} w_t(k,l)} \tag{4.34}$$

Note this behaves a set of experts $\{u_{1:t}^{(i)} : i = 1 \cdots n_t\}$ obtained by concatenating descendants to all the experts that generated them and dropping all the experts that did not generate any. The relative weight of the new expert $u_t^{(k)}$ formed by concatenating $u_t^{(i,j)}$ to $u_{1:t}^{(i)}$ is as follows

$$r_t^{(k)} \propto \frac{r_{t-1}^{(i)}}{s_t^{(i)}} w_t(i,j) \tag{4.35}$$

normalized so that the weights add up to one. In addition the texture opinions for each expert are found using the Kalman filter equations.

4.4 Tracking 3D deformable objects

The spatial location of the n points on the object varies with time due to rigid transformations (rotation, scale, translation) and non-rigid transformations (e.g., changes in expression). The rigid transformations are controlled by a rotation matrix R_t and a displacement vector D_t. The non-rigid transformations are modeled as linear combinations of a set of k 3-Dimensional morph keys

$\phi(1), \cdots, \phi(k)$. Here $\phi(i)$ is an $n \times 3$ dimensional matrix, containing the 3-D position of the n object points on key-morph i. Let

$$\phi = (\phi(1), \cdots, \phi(k)) \tag{4.36}$$

be an $n \times 3k$ matrix containing all the morph keys. The vector $C_t \in \mathbb{R}^k$ contains the morph coefficients of the object at time t.

Let $X_t \in R^{2n}$ contain the 2-D coordinates of the projection of the n object points onto an image plane, i.e. X_{2i-1}, X_{2i} are the horizontal and vertical coordinates of the projection of the i^{th} point. Under weak perspective projection we have that

$$X_t = \beta U_t \tag{4.37}$$

where

$$\beta = \begin{pmatrix} 1 & b_{11}(1) & b_{12}(1) & \cdots \\ 0 & 0 & \cdots & \cdots \\ \vdots & \vdots & \vdots & \vdots \\ 1 & b_{11}(n) & b_{12}(n) & \cdots \\ 0 & 0 & \cdots & \cdots \end{pmatrix} \begin{pmatrix} b_{3k}(1) & 0 & 0 & \cdots & \cdots & 0 \\ 0 & 1 & b_{11}(1) & b_{12}(1) & \cdots & b_{3k}(1) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ b_{3k}(n) & 0 & 0 & \cdots & \cdots & 0 \\ 0 & 1 & b_{11}(n) & b_{12}(n) & \cdots & b_{3k}(n) \end{pmatrix} \tag{4.38}$$

and

$$\begin{aligned} U_t = (&D_t(1), R_t(1,1)C_t(1) \cdots R_t(1,3)C_t(k), D_t(2), \\ &R_t(2,1)C_t(1) \cdots, R_t(2,3)C_t(k))' \end{aligned} \tag{4.39}$$

The matrix β contains the set of fixed animation morphs and the random variable U_t contains 3D pose and expression parameters. Standard techniques exist to recover the values of R_t and D_t once U_t is known [14].

To apply G-flow to this problem we need to find methods to find values for u_t that maximize $p(u_t \mid u_{1:t-1}, y_{1:t})$.

Let $\bar{y}_t$ represent a matrix version of y_t, and $\bar{v}_t$ a matrix version of the object texture map $\bar{E}(\bar{V}_t \mid u_{1:t-1}, y_{1:t-1})$. For the case in which the background is a white noise process, maximizing $p(u_t \mid u_{1:t-1}, y_{1:t})$ is equivalent to minimizing

$$L(u_t, u_{t-1}) = \sum_{i=1}^{n} (\bar{y}_t(x_i) - \bar{v}_t(\beta u_t))^2 \tag{4.40}$$

Brand [14] showed that functions of this type can be optimized in real time using the Newton-Gauss algorithm.

4.5 Comparison to Other Approaches

Inference in G-flow belongs to a class of non-linear filtering problems known as "conditionally Gaussian problems". They can be solved using non-linear filtering techniques for the non-linear part and then propagate the solution to the linear part using time dependent Kalman filters. In the particle filtering literature this approach is known as Rao-Blackwelization [1].

A major problem with applications of particle filters to video tracking is the so called "Needle in a Haystack" problem (see Figure 4.4). The simplest approach to particle filtering starts with a set of samples from the filtering distribution at time $t-1$. For each particle samples are taken from the state transition distribution. Then the image at time t is observed and each particle is weighted by the image likelihood function. Unfortunately in most tracking problems the likelihood function is highly peaked at the correct location (the needle) and relatively flat at the incorrect locations (the haystack). If the samples miss the peak of the likelihood function they will provide a very inefficient estimate of the filtering

distribution. The problem gets worse as the number of parameters increases. For example, in 2D the likelihood function may not only be highly peaked but it may also have a strong orientation . Samples will be wasted by random sampling at the wrong location or with the wrong orientation (see Figure 4.5). Here we reduce this problem by explicitly computing the peak and orientation (i.e., precision matrix) of the opinion distribution distribution once the image has been observed. This is possible due to the the fact that the observed sequence (video images) is smooth in space and time, something that may not be necessarily the case for filtering problems in general.

[18] used an extended Kalman filter approach for a problem in which the 2D pose of an object and the texture of the object were tracked simultaneously. This limited the approach to unimodal solutions, which are known to be risky for tracking problems. They did not take advantage of the conditionally Gaussian nature of the problem and did not incorporate background information.

Brand [14] showed that one can combine the outputs of optic-flow solutions computed independently at different image points, along with their uncertainty, to find the rigid motion and non-rigid deformation parameters that best fit those flow solutions. We found that the approach is formally equivalent to directly propagating the linear constraints without intermediate computation of optic flow, which is the approach we use in our simulations. [89] presented an approach to propagate general non-rigid motion constraints on top the standard optic flow algorithm. Both [14] and [89] rely on unimodal state distributions, and do not learn object or background texture maps.

4.6 Simulations

We collected video of a person while making a variety of facial expression on command. An additional motion capture session was used to create a 3D model of the face and a set of 3D animation morphs. We are currently working on a system

that will automatically find face geometry parameters based on a large dataset of 3D faces.

Twenty particles were initialized using the first frontal pose and propagated using the G-flow algorithm. A video of the entire sequence is available at our Web site. Figures 4.6 and 4.7 shows the distribution of particles for 3D pose and animation coefficients as a function of time. Note how the system can maintain multimodal distributions when necessary.

The system can run at about $1/n$ real time in Matlab, where n is the number of particles. Unfortunately more rigorous testing has not been possible at this time due to the lack of ground truth data for non-rigid object motion. This is a difficult problem because determining ground truth usually requires placing visible marks on the face for stereo motion capture. Such marks would make the tracking problem much easier. However to remedy this situation work is in progress to create data-sets from which ground-truth can be extracted by using a combination of infra-red cameras and marks, along with visible light cameras.

4.7 Conclusions

We presented a generative model (G-flow) for video sequences. The model provides a useful framework for studying the problem of how to dynamically combine motion and appearance information in a principled manner. Current optic flow and template based algorithms emerge in this model as optimal inference processes under specific conditions. In more realistic conditions optimal inference consists of a dynamically weighted combination of motion and appearance based information.

4.8 Acknowledgements

Chapter Four is adapted from [64], which was published in Computer Vision and Pattern Recognition Workshop on Generative Models for Vision in 2004. My

co-authors, Javier Movellan, Tim Marks, and J. Cooper Roddey, supervised and collaborated with me on the research which forms the basis of this chapter.

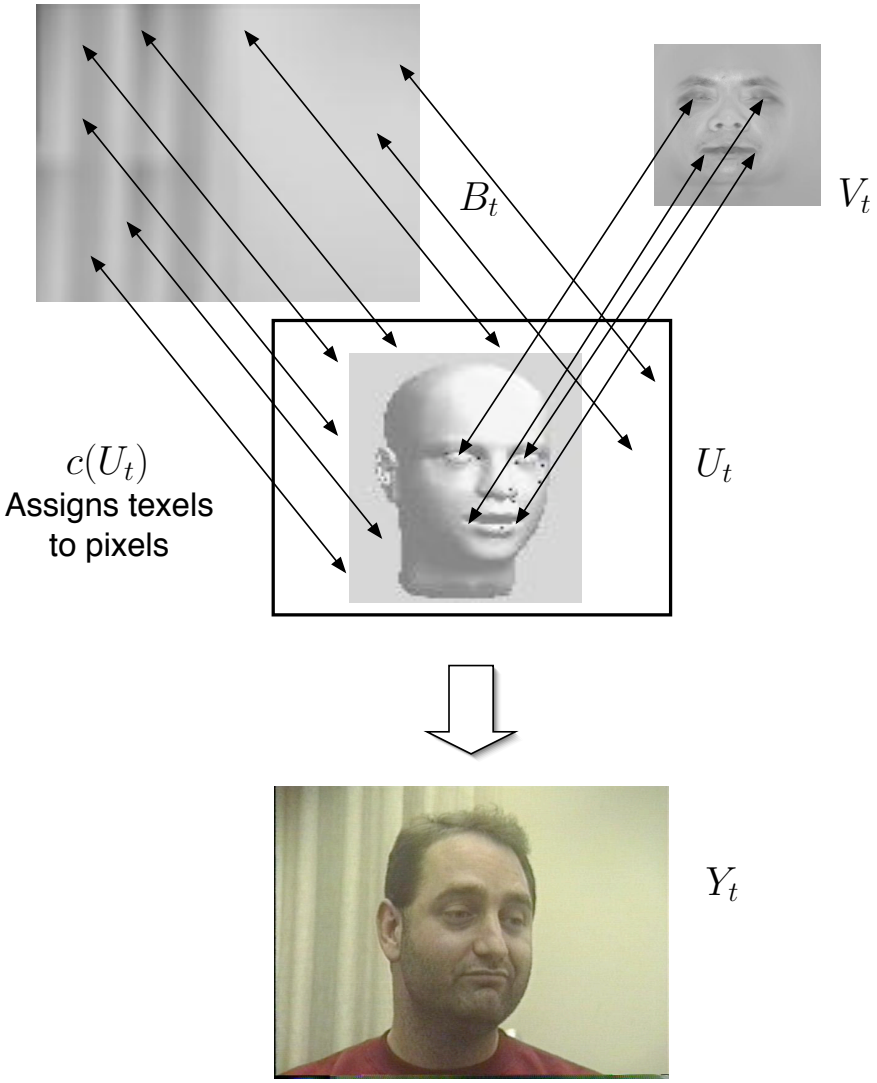


Figure 4.2: The G-Flow projection model: $c(U_t)$ determines which texel is responsible for rendering each pixel on the image plane. Some of these will be rendered by object texels, some by background texels.

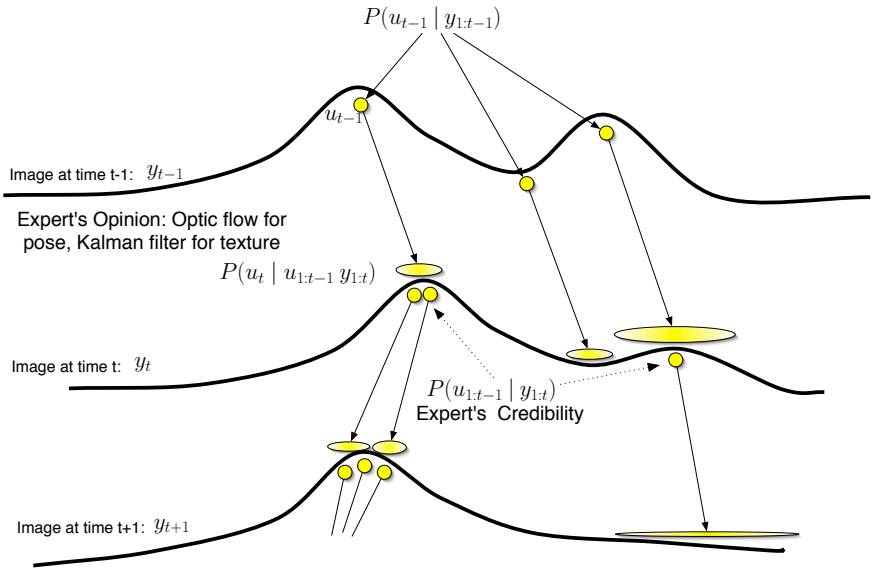


Figure 4.3: An algorithm for solving the G-flow inference problem.

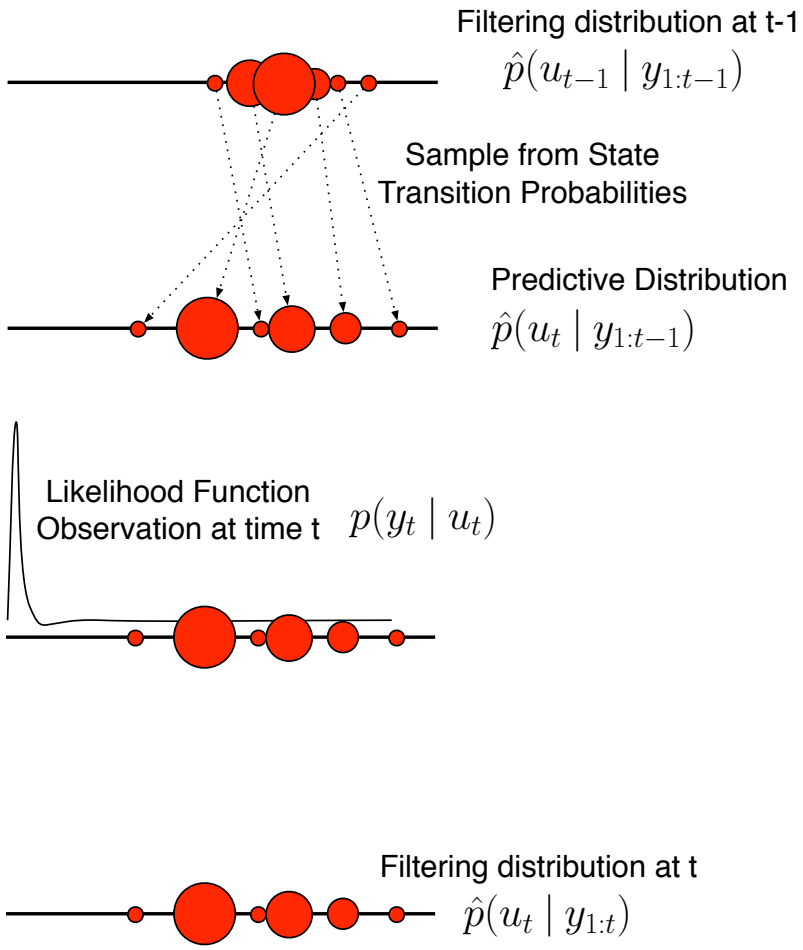


Figure 4.4: A 1D version of the Needle in a Haystack Problem: If the likelihood function for the image at time t is very peaked, blind sampling approaches are likely to miss it and provide inefficient estimates of the filtering distribution. In G-flow this problem is reduced by explicitly computing the peak of the distribution after the image has been observed and sampling about that peak.

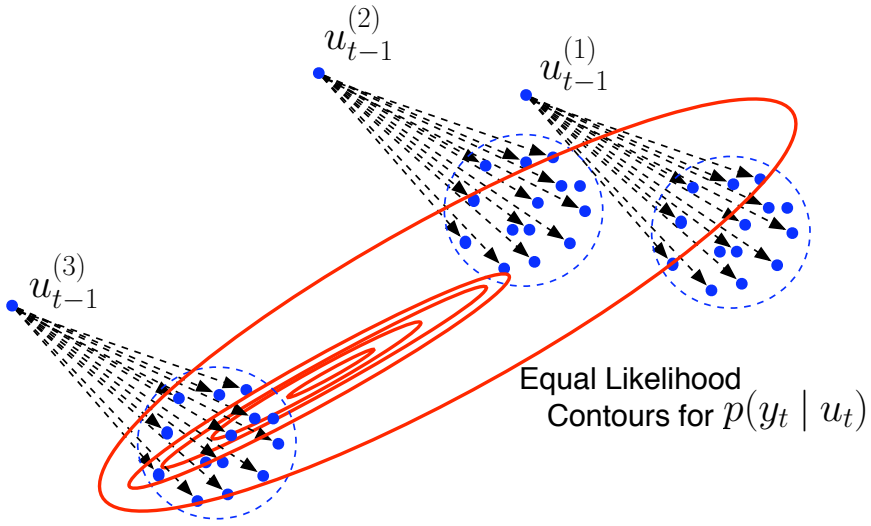


Figure 4.5: A 2D version of the Needle in a Haystack Problem: The likelihood function is peaked and oriented. The descendants of particles $u_{t-1}^{(1)}$ and $u_{t-1}^{(2)}$ distribute about low likelihood regions due to poor location and blind sampling. The descendants of particle $u_{t-1}^{(3)}$ are well located but the sampling distribution does not have the right orientation, thus wasting a large number of particles. In G-flow the problem is reduced by explicitly computing the peak and orientation (i.e., the precision matrix) of the opinion distribution.

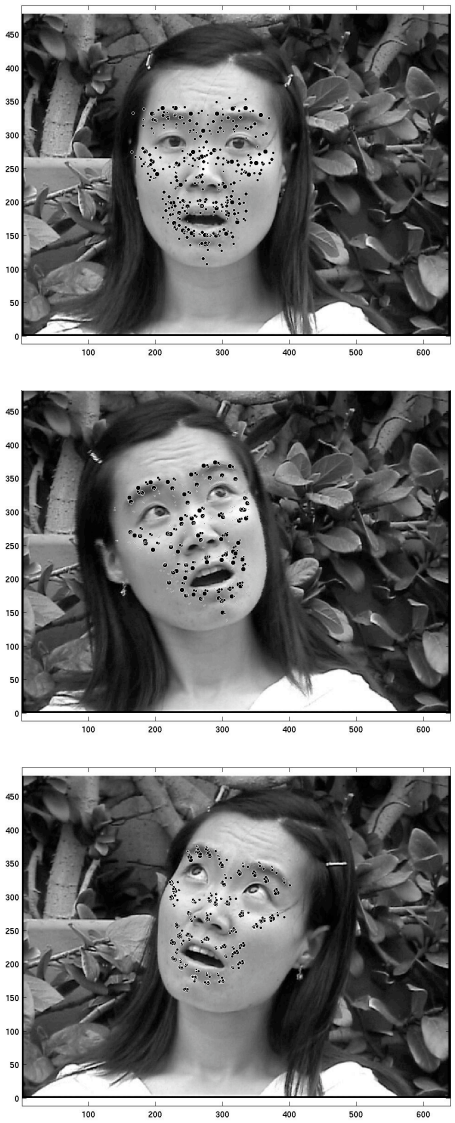


Figure 4.6: Particle filtering results: This figure displays the locations of object points for 10 particles from an early frame (top) to a later frame (bottom) of the video sequence. The radius of the circles is proportional to the weight of the particles.

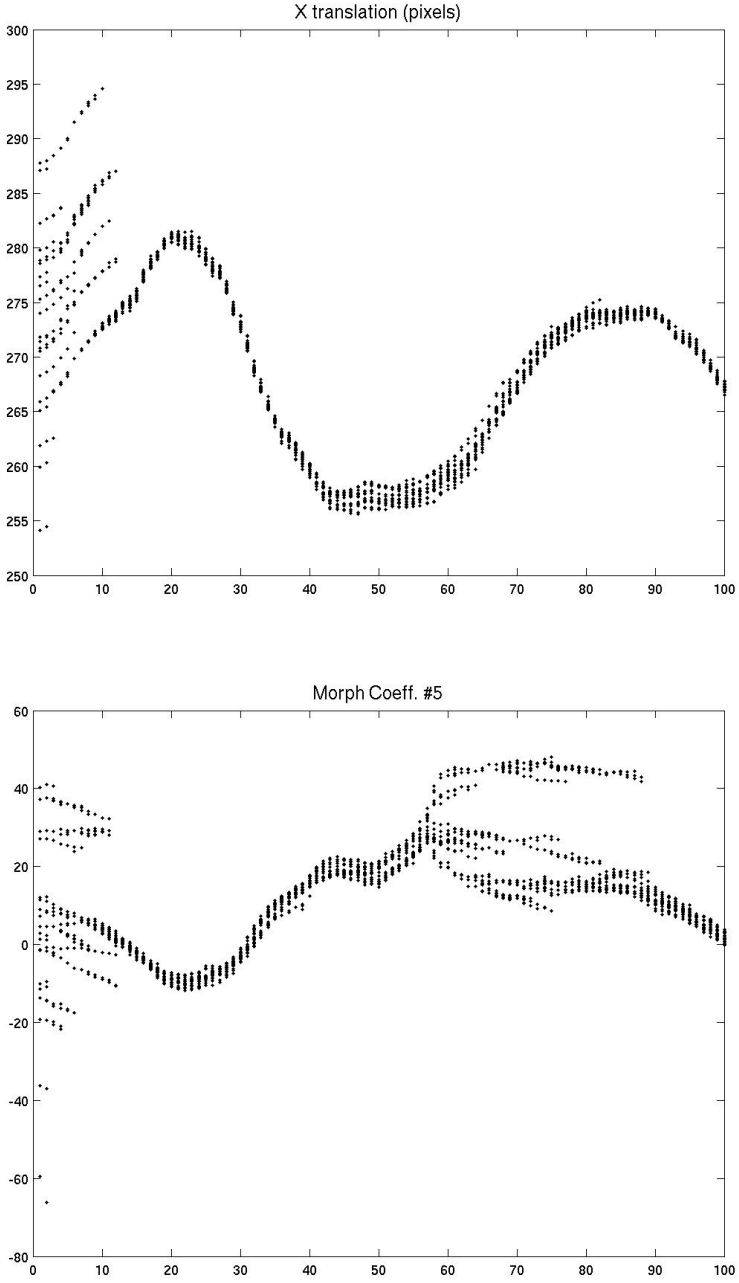


Figure 4.7: Tracking results for a video sequence: There were 6 pose parameters and 4 morph parameters. The graphs contain the filtering distribution for 2 pose parameters using 20 particles.

Chapter 5

Single Microphone Source Separation

5.1 High-Resolution Signal Reconstruction

Abstract

We present a framework for speech enhancement and robust speech recognition that exploits the harmonic structure of speech. We achieve substantial gains in signal to noise ratio (SNR) of enhanced speech as well as considerable gains in accuracy of automatic speech recognition in very noisy conditions.

The method exploits the harmonic structure of speech by employing a high frequency resolution speech model in the log-spectrum domain and reconstructs the signal from the estimated posteriors of the clean signal and the phases from the original noisy signal.

We achieve a gain in signal to noise ratio of 8.38 dB for enhancement of speech at 0 dB. We also present recognition results on the Aurora 2 data-set. At 0 dB SNR, we achieve a reduction of relative word error rate of 43.75% over the baseline, and 15.90% over the equivalent low-resolution algorithm.

5.1.1 Introduction

A long standing goal in speech enhancement and robust speech recognition has been to exploit the harmonic structure of speech to improve intelligibility and increase recognition accuracy.

The source-filter model of speech assumes that speech is produced by an excitation source (the vocal cords) which has strong regular harmonic structure during voiced phonemes. The overall shape of the spectrum is then formed by a filter (the vocal tract). In non-tonal languages the filter shape alone determines which phone component of a word is produced (see Figure 5.2). The source on the other hand introduces fine structure in the frequency spectrum that in many cases varies strongly among different utterances of the same phone.

This fact has traditionally inspired the use of smooth representations of the speech spectrum, such as the Mel-frequency cepstral coefficients, in an attempt to accurately estimate the filter component of speech in a way that is invariant to the non-phonetic effects of the excitation[71].

There are two observations that motivate the consideration of high frequency resolution modeling of speech for noise robust speech recognition and enhancement. First is the observation that most noise sources do not have harmonic structure similar to that of voiced speech. Hence, voiced speech sounds should be more easily distinguishable from environmental noise in a high dimensional signal space[1].

A second observation is that in voiced speech, the signal power is concentrated in areas near the harmonics of the fundamental frequency, which show up as parallel ridges in the spectrogram (see Figure 5.2). In a noisy environment, the local signal to noise ratio along the ridge is greater than the average SNR.

Figure 5.1 shows the estimate of a clean speech vector, the noisy input vector (car noise), and the true clean speech vector for comparison. The horizontal axis shows frequency in Hertz, and the vertical axis shows log-energy of the amplitude of

[1]Even if the interfering signal is another speaker, the harmonic structure of the two signals may differ at different times, and the long term pitch contour of the speakers may be exploited to separate the two sources [39].

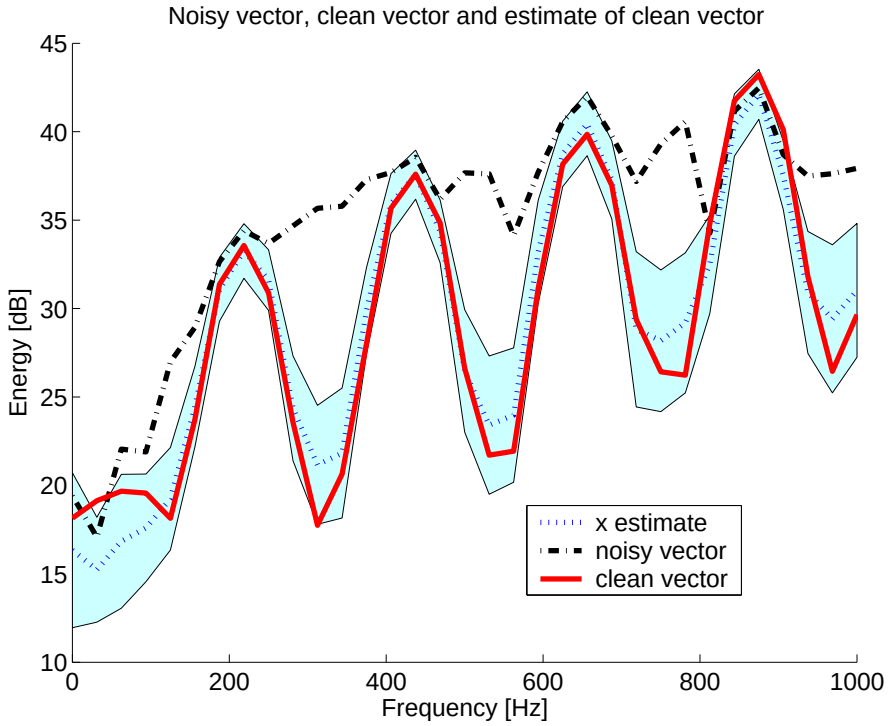


Figure 5.1: Estimation Results: The noisy input vector (dot-dash line), the corresponding clean vector (solid line) and the estimate of the clean speech (dotted line), with shaded area indicating the uncertainty of the estimate (one standard deviation). Notice that the uncertainty on the estimate is considerably larger in the valleys between the harmonic peaks. This reflects the lower SNR in these regions. The vector shown is frame 100 from Figure 5.2

each frequency. The regularly spaced peaks are the harmonics of the fundamental frequency. Notice that at the low end of the frequency range, the true signal is 'submerged' in the noise, whereas the harmonic peak at c.a. 670Hz and 900Hz emerge from the noise. Notice also that the first standard deviation (shown as a shaded area) of the estimate is large in the valleys, where the SNR is low and smaller around the harmonic peaks, where the SNR is higher. The method for producing the clean speech estimate is discussed in section 5.1.2.

Researchers have sought to exploit this localization of signal power, both in the time domain and in the frequency domain. Methods for achieving this goal include

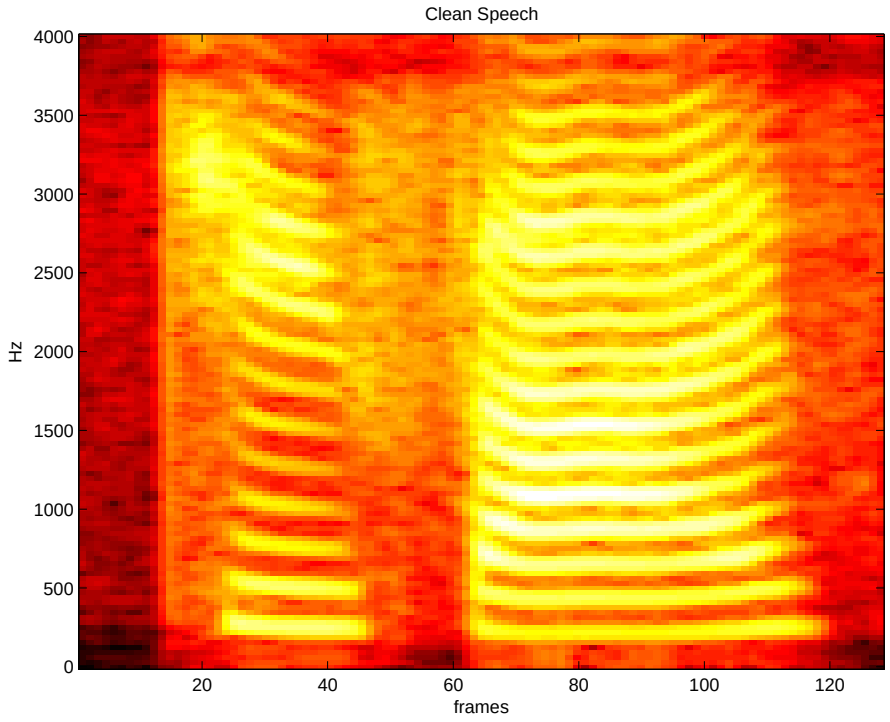
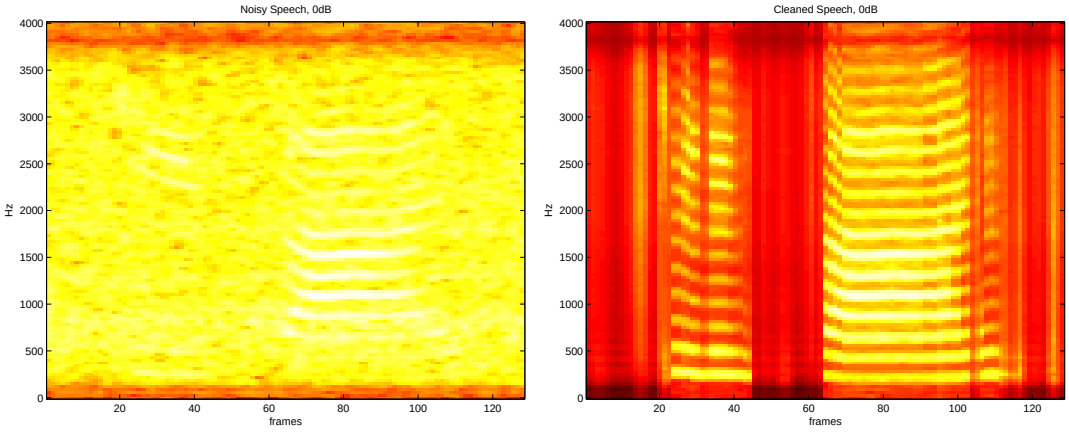


Figure 5.2: Spectrogram of clean speech. The words 'TWO FIVE' are being spoken.

alignment and gating of the glottal impulses in the time domain[59], and tracking the pitch as a pre-processing stage[81, 84]. Such approaches use highly constrained voicing models that are incongruous to the modeling of other aspects of the speech signal and employ modularized, multistage processing where aspects of the voicing are processed separately[68]. These approaches have been vulnerable to noise because of implicit independence assumptions or because the voicing estimation does not take noise into account. In addition, there may be excitation patterns and artifacts of the signal analysis that are poorly captured by such highly constrained models for harmonic structure. In contrast, our approach is to use a single high resolution log-spectrum model for both excitation and filter and train a model capable of capturing the relevant structures.



(a) Spectrogram of speech at 0 dB (b) Spectrogram of cleaned speech at 0 dB.

5.1.2 Model based signal enhancement

The core of the method involves calculating posteriors for the high frequency resolution log-spectrum $p(x|y)$, given the noisy speech. We employ the Algonquin framework [29, 52] to calculate these posteriors.

The model for noisy speech in the time domain is (omitting the channel for clarity)

$$y[t] = x[t] + n[t]. \tag{5.1}$$

where $x[t]$ denotes the clean signal, $n[t]$ denotes the noise, and $y[t]$ denotes the noisy signal. In the Fourier domain, the relationship becomes

$$Y(f) = X(f) + N(f) \tag{5.2}$$

where f designates the frequency component of the FFT. This can also be written in terms of the magnitude and the phase of each component:

$$|Y(f)|\angle Y(f) = |X(f)|\angle X(f) + |N(f)|\angle N(f) \tag{5.3}$$

where $|Y(f)|$ is the magnitude of $Y(f)$ and $\angle Y(f)$ is the phase.

We model only the magnitude components and do not explicitly model the phase components. The relationship between the magnitudes is

$$|Y(f)|^2 = |X(f)|^2 + |N(f)|^2 + 2|X(f)||N(f)|cos(\theta) \tag{5.4}$$

where θ is the angle between X and N. For the purposes of modeling, we assume the we can model the last term as a noise term, hence we approximate this relationship between magnitudes as

$$|Y(f)|^2 = |X(f)|^2 + |N(f)|^2 + e \tag{5.5}$$

where the e is a random error [52]. Next we take the logarithm and arrive at the relationship in the high resolution log-magnitude-spectrum domain

$$y = x + \ln(1 + \exp(n - x)) + \epsilon \tag{5.6}$$

where ϵ is assumed to be Gaussian. Hence, we can also write this relationship in terms of a distribution over the noisy speech features y as

$$p(y|x, n) = N(y; x + \ln(1 + \exp(n - x)), \psi) \tag{5.7}$$

where ψ is the variance of ϵ, and $N(y|\mu, \psi)$ denotes a normal density function in y with mean μ and variance ψ.

The transformations that we have applied to the model above are the same as the first steps in the calculation of the Mel frequency cepstrum features with the exception that we did not perform the Mel-scale warping before applying the log transform. For example, in the Aurora front end[41], the Mel-scale warping, smooths out the harmonics and reduces the dimensionality of the feature vector from 128 dimensions to 23 dimensions. The result of omitting the Mel-scale warping is that we do not smooth out the speech harmonics.

For the purpose of signal reconstruction, we are interested in likely values of clean speech, given the noisy speech. By recasting this relationship in terms of a likelihood $p(y|x, n)$, and using prior models for speech $p(x)$ and noise $p(n)$, we can arrive at a posterior distribution for the clean speech vector $p(x|y)$. This will be described in the next section.

By inverting the procedure described above we can reconstruct an estimate of the clean signal. To do this we find the MMSE estimate for clean speech $\hat{x}$ and

calculate the inverse Fourier transform

$$\hat{x}[t] = IFFT(\exp(\hat{x}) \cdot \angle Y) \tag{5.8}$$

where $\hat{x} = \int xp(x|y)dx$. In this reconstruction, we have used the original phases from the noisy signal.

5.1.3 Inference

We now turn our attention to the procedure for estimating the posterior for the clean speech log-magnitudes $p(x|y)$. For this we employ the Algonquin method. Extensive evaluations of this framework have been performed in the context of robust speech recognition. In previous work, speech and noise models have either been in the "low-resolution" log-Mel-spectrum domain, or in the truncated cepstrum domain. Here we briefly outline the Algonquin procedure. Detailed discussions can be found in [29, 52].

At the heart of the Algonquin method is the approximation of the posterior $p(x|y)$ by a Gaussian.

The true posterior

$$p(x|y) = c \int p(y|x,n)p(n)p(x)dn \tag{5.9}$$

is non-Gaussian, due to the non-linear relationship in Eqn. (5.6). In Eqn. (5.9) c is a normalizing constant, $p(n)$ is the noise model, and $p(x)$ is the speech model, and $p(y|x,n)$ is the likelihood function discussed above.

We use a mixture of Gaussians to model both speech and noise. Hence

$$p(x) = \sum_s p(s)p(x|s) = \sum_s \pi_s N(x|\mu_s^s, \Sigma_s^x) \tag{5.10}$$

and similarly for $p(n)$. The construction of the speech model will be discussed below.

Due to the non-linear relationship between x and n for a given y, the true posterior $p(x|y)$ is non-Gaussian. We wish to approximate this posterior with a Gaussian posterior. The first step is to linearize the relationship between x and n.

For notational convenience, we write the stacked vector $z = [x^T n^T]$ and we introduce the function $g(z) = x + \ln(1 + \exp(n - x))$.

If we linearize the relationship of Eqn. (5.6) using a first order Taylor series expansion at the point z_0, we can write the linearized version of the likelihood

$$p_l(y|x,n) = p_l(y|z) = N(y; g(z_0) + G(z_0)(z - z_0), \Psi) \tag{5.11}$$

where z_0 is the linearization point and $G(z_0)$ is the derivative of g, evaluated at z_0. We can now write a Gaussian approximation to the posterior for a particular speech and noise combination as

$$p_l(x, n, y|s^x, s^n) = p_l(y|x,n)p(x|s^x)p(n|s^n) \tag{5.12}$$

It can be shown[52] that the $p(x, n|y, s^x, s^n)$ is jointly Gaussian with mean

$$\eta_s = \Phi_s \left[\Sigma_s^{-1}\mu_s + G^T\Psi^{-1}(y - g - Gz_0)\right] \tag{5.13}$$

and covariance matrix

$$\Phi_s = \left[\Sigma_s^{-1} + G^T\Psi^{-1}G\right]^{-1} \tag{5.14}$$

and the posterior mixture probability $p(y|s^x, s^n)$ can be shown to be

$$\gamma_s = |\Sigma_s|^{-1/2}|\Psi|^{-1/2}|\Phi_s|^{1/2} \cdot \exp\Bigg[-\frac{1}{2}(\mu_s^T\Sigma_s^{-1}\mu_s + (y_{obs} - g + Gz_0)^T\Psi^{-1}(y_{obs} - g + Gz_0) - \eta_s^T\Phi_s^{-1}\eta_s)\Bigg].$$

The choice of the linearization point is critical to the accuracy of the approximation. Ideally, we would like to linearize at the mode of the true posterior. In the Algonquin algorithm, we attempt to iteratively move the linearization points towards the mode of the true posterior. In iteration i of the algorithm, the mode of the approximate posterior in iteration $i-1$, μ_{i-1} is used as a linearization point of the likelihood, i.e. $z_i = \mu_{i-1}$. The algorithm converges in 3-4 iterations.

5.1.4 Speech Model

Speech modeling for enhancement and speech recognition usually involves dimensionality reduction which removes the voice harmonics. This is either done explicitly, such as by using the Mel-warping, or implicitly, such as by using a small auto-regressive model. The filter and excitation components of the generative speech model are relatively independent, since voiced speech sounds can be spoken at any pitch. To model a particular speech sound in high resolution, one would expect to need an instance of the voiced acoustic model at each possible pitch.

A first approximation is to model the filter and excitation components independently. To construct such a model, one would lifter the 128 frequency component speech vectors to produce 128 component filter (vocal tract) features and 128 component excitation (vocal cords) features. This approach has the advantage that the models are compact, and independent temporal dynamics can be efficiently employed on each component, as in [39]. However, the model over-generates speech by allowing combinations of unvoiced excitation and voiced filters and vice versa, and the computations required for temporal dynamics may be too costly in many cases.

An alternate strategy is to simply train a single non-factored high-resolution speech model. In the experiments described below, we used non-factored Gaussian mixture models (GMM). We trained two models: a speaker independent gender independent model, and a speaker independent gender dependent model. The speaker independent, gender independent model had 512 mixtures, and 128 frequency components, while the gender dependent model had 512 mixtures for the male component and 512 mixtures for the female component. These models were trained in the standard way[74], by initializing using vector quantization, and then using Expectation Maximization to find the parameters of the GMMs.

Although this approach is not as efficient as the factored model, with respect to the number of parameters required to represent combinations of voiced filters at different pitches, it has the advantage that it does not over-generate speech.

5.1.5 High-Resolution Signal Reconstruction

To reconstruct the signal, we first calculate high-resolution log-spectral features of the noisy input signal as described in section 5.1.2. In the feature extraction stage, we used hamming windows of length 25 ms, and the frame rate of 10 ms. A corresponding synthesis window is designed such that the analysis window multiplied by the synthesis window, and overlapped with neighboring analysis-synthesis windows at the frame rate, sums to unity at each time point.

We smooth the high-resolution log-spectrum features across frames by filtering them temporally with a simple FIR filter with parameters [0.25 0.5 0.25]. Without this smoothing step, the inference algorithm tends to produce spurious errors.

The Algonquin algorithm is then used to infer the posterior distributions over the clean speech. In the results reported below, we used the MMSE estimate based on $p(x|y)$.This is then exponentiated and used as a point estimate for $|X(f)|$. Alternately, we could use the MMSE estimate of $\widehat{|X(f)|^2} = E[\exp(x)]$. However, the fact that the speech recognizer operates on the log spectrum domain motivates the former rather than the latter estimate.

We then reconstruct each frame of the signal, by use of the inverse Fourier transform, as in Eqn. (5.8), where the phase components are the phases of the noisy signal. The frames are then overlapped and added together using the tapered synthesis window described above.

5.1.6 Results

We tested the high-resolution signal enhancement for speech enhancement as well as for robust speech recognition.

5.1.7 Speech Enhancement Results

In informal listening tests, the subjective quality of the enhanced speech was reported to be exceptionally good. At very low SNR (-5 dB and 0 dB), the most

	-5 dB	0 dB	5 dB	10 dB	15 dB	20 dB
ΔSNR	10.76	8.38	6.27	3.95	1.28	-1.94
ΔSNR_{seg}	6.82	6.58	6.12	5.35	4.29	2.87

Table 5.1: Gains in SNR for car noise at a range of SNRs. The two measures of SNR are for standard SNR and Segmental SNR. For segmental SNR, we used a window of 25 ms, a SNR floor of -10 dB and an SNR ceiling of 35 dB.

notable distortion in the enhanced speech is flutter due to the inference algorithm assigning low energy fricatives to periods of silence, as well as silences in low energy voiced portions. At higher decibel levels (15 dB and 20 dB) the enhanced speech is almost indistinguishable from clean speech.

In Table 5.1 we give dB gains for the car noise condition of the Aurora data set. The first row shows SNR computed over the whole waveform, while the second row shows segmental SNR, computed using a floor of -10 dB and a ceiling of 35 dB.

5.1.8 Aurora Speech Recognition Results

To assess the performance of high-resolution signal reconstruction for speech recognition, we ran experiments on the Aurora 2 data-set. The Aurora 2 data-set contains spoken digits, artificially mixed with various noise types at Signal to noise ratios of -5 dB to 20 dB, in addition to unaltered clean speech. There are 1001 test files in each condition, where each test file contains from 1 to 7 spoken digits. In the experiments below, we report results for the Car noise condition. This condition has relatively stationary noise which allows us to use a single Gaussian noise model, estimated from the first 20 frames of each file. Other conditions such as "Subway" require larger noise models to handle the non-stationary aspect. In previous work, it has been shown [52] that using low-resolution Algonquin with larger noise models, as well as adapting the noise model will produce considerable gains in recognition accuracy, at the expense of higher computational complexity.

The standard low-resolution Algonquin method produces estimates of clean parameters in the 23 dimensional log-Mel-spectrum domain. For the recognition

experiments, these are converted to cepstrum parameters directly, by taking the discrete cosine transform. For the high-resolution signal reconstruction experiments, the time domain signal was reconstructed and the standard Aurora front end was then used to produce cepstrum parameters from the time domain signals.

The graph in Figure 5.3 shows the recognition accuracy for the Car noise condition of Set A of the Aurora 2 data-set, using multi-condition training of the acoustic models. We used the standard Aurora back-end, which is an HTK based recognizer with 16 state, left-to-right word models with 3 mixture acoustic models in each state. Figure 5.5 shows the change in absolute Word accuracy over the baseline, and Figure 5.4 shows the change in word error rate due to high-resolution processing.

The baseline of 86.52% is shown as the bottom line in Figure 5.3. The result for "low-resolution" log-Mel-spectrum is the middle line in Figure 5.3. The speech model used was a Gaussian mixture model with 256 components, of 23 dimensions each. The low-resolution Algonquin algorithm achieves an average recognition accuracy of 90.12% for the Car noise condition, which is a relative reduction error rate of 13.26%.

The results for high-resolution signal reconstruction with a speaker independent, gender dependent model is the top line in Figure 5.3. The average accuracy is 91.14%, which is a relative reduction in average word error rate of 15.62% over the baseline. Using gender independent high-resolution models achieves a slightly lower average accuracy of 91.04%.

It is more interesting to compare the recognition rates for low-resolution Algonquin and high-resolution Algonquin. Interestingly, the gains are mostly achieved at -5 dB and 0 dB. The increases in word accuracy are 5.28% and 13.48% absolute (16.95% and 19.02% reduction in WER respectively), while at higher SNRs the recognition rates are almost identical. This indicates that the advantages of using voicing information are mostly at very low signal-to-noise

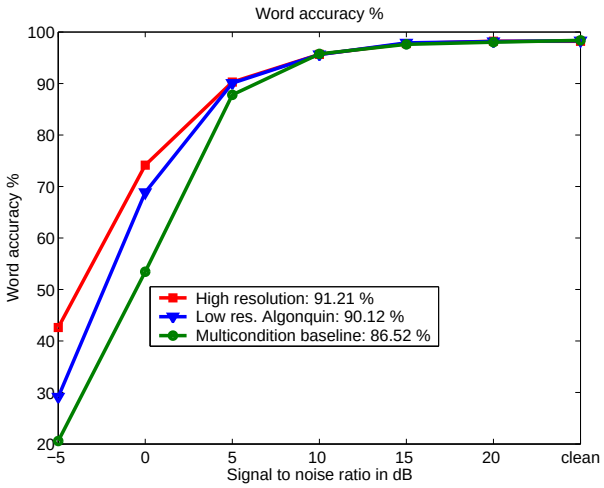


Figure 5.3: Word accuracy of High-Resolution Signal Reconstruction using gender-dependent models, Low-Resolution Algonquin and Aurora Multi-condition Baseline for the Car noise condition

ratios. It also supports the assumption that voicing information is not helpful for speaker-independent recognition of clean speech in non-tonal languages.

5.1.9 Discussion and Conclusions

Our findings support the hypothesis that high-resolution spectral information is quite useful for enhancing noisy speech and substantially helps recognition in very noisy conditions. At the same time, our findings are consistent with the widely held assumption that low-resolution spectral components are sufficient for speaker-independent recognition of clean speech.

The traditional approach for exploiting harmonic structure is to employ parametric models with a small number of parameters for the excitation component of the signal. This can lead to heterogeneous models and make it difficult to jointly estimate parameters related to excitation and filter in noisy conditions. The model presented in this section avoids such pitfalls by employing a combined excitation-filter speech model. The size of model required is surprisingly small. Our model presents an advantage over models that factorize the excitation and filter

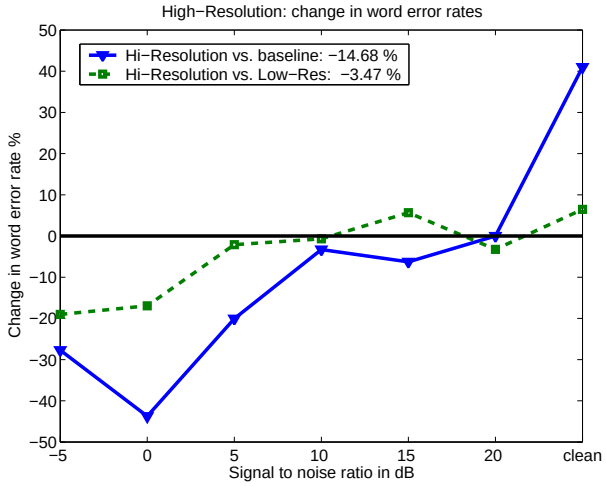


Figure 5.4: Word error rate of High-Resolution method as compared to Baseline, and Low-Resolution Algonquin.

components in that we can model statistical dependencies between the excitation and filter components of a signal.

We have incorporated this information into a probabilistic model in a principled way that is compatible with the current paradigm in speech processing.

5.2 Single Microphone Source Separation using High-Resolution Signal Reconstruction

Abstract

We present a method for separating two speakers from a single microphone channel. The method exploits the fine structure of male and female speech and relies on a strong high frequency resolution model for the source signals.

The algorithm is able to identify the correct combination of male and female speech that best explains an observation and is able to *reconstruct* the component signals, relying on prior knowledge to 'fill in' regions that are masked by the other speaker.

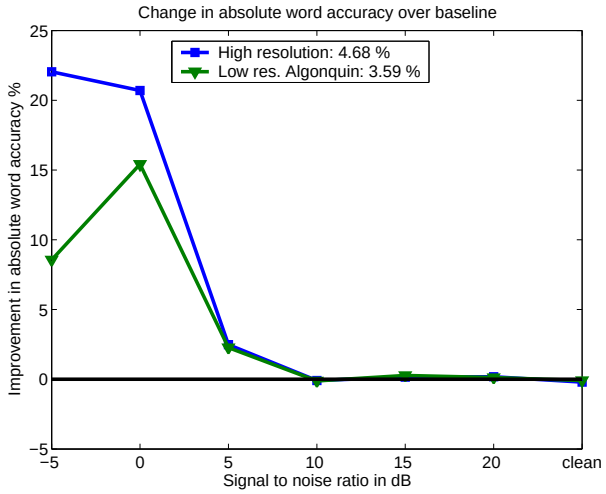


Figure 5.5: Change in absolute Word Accuracy of High-Resolution Signal Reconstruction using Gender Dependent models and Low-Resolution Algonquin compared to the Aurora Multi-condition Baseline for the Car noise condition.

The two speaker single microphone source separation problem is one of the most challenging source separation scenarios and few quantitative results have been reported in the literature. We provide a test set based on the Aurora 2 data set and report performance numbers on a portion of this set. We achieve results of 5.51 dB average increase in SNR for male speakers and 6.59 dB for female speakers.

5.2.1 Introduction

Source separation involves recovering two or more signals that have been mixed. When multiple microphones are available the phase between the different signals can be exploited to recover the composite signals. A large body of work revolves around exploiting phase information for source separation[75]. Source separation via Independent Component Analysis (ICA)[9, 3] relies on multiple signals as well.

The most challenging case for source separation is when only one signal is available. In this case, one has to rely exclusively on the prior knowledge of the signals to be separated.

Previous work in the area of single microphone source separation has used less

accurate approximations to the mixing process[39] or sub-band representation of speech[78] which remove important correlations in the speech signal.

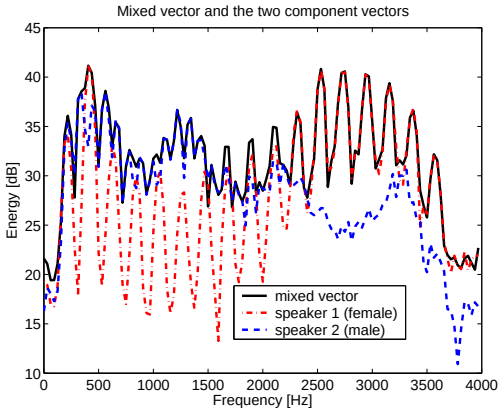
The core inference method used in this work has been extensively studied in the context of robust speech recognition[52], using low dimensional representations of speech. Recently we have shown [54] that the harmonic structure of speech is of substantial value for separating speech from noise, in very noisy conditions. In this section, we extend the method for the cross-speaker condition, where the competing signal is a second speaker.

Figures 5.2.1(a)-5.2.1(c) shows the result of running the algorithm on a single frame of the input (frame 100 from Figure 5.2.4(a)). Only frequencies 1200 Hz -2600 Hz are shown for clarity. Figure 5.2.1(a) shows the input to the algorithm (black heavy line), the female component feature vector(red dotted line) and male component feature vector(blue dashed line).

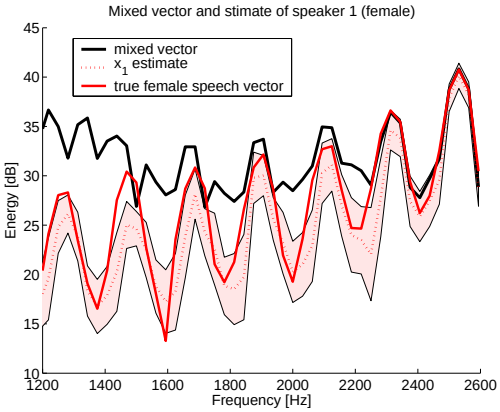
Intuitively, the algorithm has identified the best combination of male and female speech, that explains the observation. Notice that the amplitude of the male speaker is stronger in the lower half of the frequency range shown and the female speaker is stronger in the upper half of the frequency range. In the middle of the frequency range, the amplitudes are in a similar range. Notice that due to the log scale the mixed signal is effectively equal to the maximum of the two signals if one signal is considerably stronger than the other signal as happens on both ends of the frequency range. Notice also that when the values are in a similar range (e.g. at 1900 Hz) the mixed signal is not effectively equal to them maximum[2].

Figure 5.2.1(b) shows the posterior estimate for the female signal. As can be seen in Figure 5.2.1(a), the female signal is effectively masked by the male speaker in the lowest part of the frequency range and vice versa. The algorithm is able to *reconstruct* the signal in the area where the female signal masked based on the prior knowledge of female speech encoded in the speech model. Notice that the algorithm finds a remarkably good estimate.

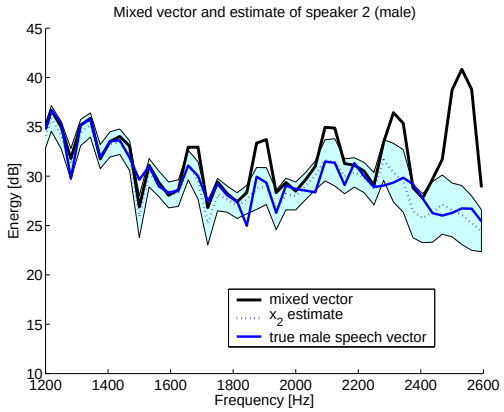
[2]in this case, the max approximation is sub-optimal[78].



(a) Mixed signal input feature vector



(b) Posterior estimate for speaker 1 (female).



(c) Posterior estimate for speaker 2 (male)

Figure 5.6: Speech separation spectra: (a) Mixed signal input feature vector (solid black line), the posterior mode for for speaker 1 (red dotted line) and the mode of the estimate for speaker 2 (blue dashed line). (b) The posterior estimate for speaker 1 (female). Notice that signal is effectively masked in the lower portion of the frequency range. The algorithm is able to *reconstruct* these values due to the strong prior model. The shaded area represents the uncertainty of the estimate and is the first standard deviation. Notice that the uncertainty is larger for 'submerged' estimates. (c) Posterior estimate for speaker 2 (male).

In the areas where the female signal is 'submerged' in the male signal, the uncertainty of the estimate is much larger than where the signal dominates. The uncertainty is quantified by the variance of the posterior and is represented by the shaded area in the figure.

Figure 5.2.1(c) shows the posterior estimate for the male signal. Similarly we see that the signal has been reconstructed where it is effectively masked, and the uncertainty is larger in these areas.

5.2.2 High-Resolution Source Separation

The core of the method involves calculating posteriors for the high frequency resolution log-spectrums $p(x_1|y)$ and $p(x_2|y)$ of the two speakers, given the mixed signals. We employ the Algonquin framework [29, 52] to calculate these posteriors. The derivation given here is exactly equivalent to the derivation when the interfering signal is noise.

The model for mixed speech in the time domain is

$$y[t] = x_1[t] + x_2[t]. \tag{5.15}$$

where $x_1[t]$ denotes the first speaker, $x_2[t]$ denotes the second speaker, and $y[t]$ denotes the mixed signal. In the Fourier domain, the relationship becomes

$$Y(f) = X_1(f) + X_2(f) \tag{5.16}$$

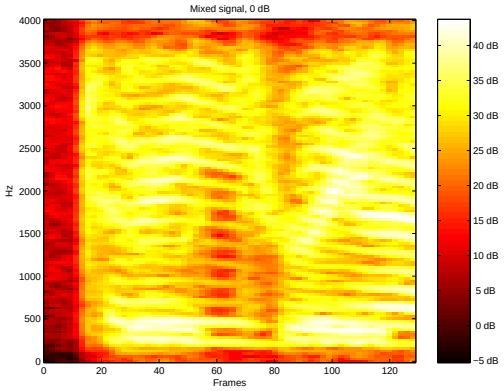
where f designates the frequency component of the FFT. This can also be written in terms of the magnitude and the phase of each component:

$$|Y(f)|\angle Y(f) = |X_1(f)|\angle X_1(f) + |X_2(f)|\angle X_2(f) \tag{5.17}$$

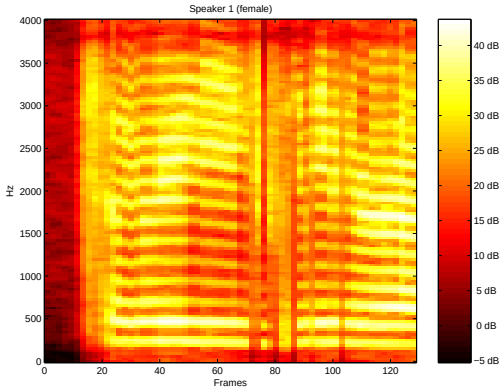
where $|Y(f)|$ is the magnitude of $Y(f)$ and $\angle Y(f)$ is the phase.

We model only the magnitude components and do not explicitly model the phase components. The relationship between the magnitudes is

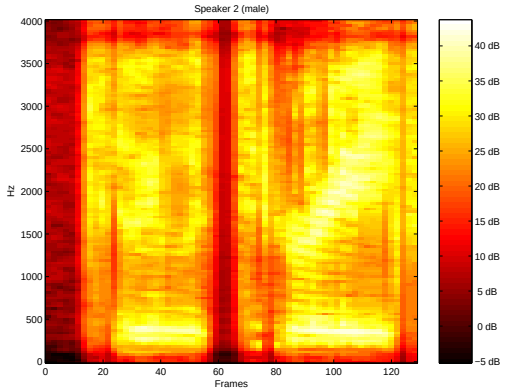
$$|Y(f)|^2 = |X_1(f)|^2 + |X_2(f)|^2 + 2|X_1(f)||X_2(f)|cos(\theta) \tag{5.18}$$



(a) Spectrogram of mixed signal



(b) Reconstructed spectrogram for speaker 1 (female).



(c) Reconstructed spectrogram for speaker 2 (male).

Figure 5.7: Speech separation spectrograms: (a) The spectrogram of the mixed signal. (b) The reconstructed spectrogram for signal 1 (female). (c) The reconstructed spectrogram for signal 2 (male).

where θ is the angle between X_1 and X_2. For the purposes of modeling, we assume the we can model the last term as a noise term, hence we approximate this relationship between magnitudes as

$$|Y(f)|^2 = |X_1(f)|^2 + |X_2(f)|^2 + e \tag{5.19}$$

where the e is a random error [52]. Next we take the logarithm and arrive at the relationship in the high resolution log-magnitude-spectrum domain

$$y = x_1 + \ln(1 + \exp(x_2 - x_1)) + \epsilon \tag{5.20}$$

where $y = \log(|Y(f)|^2)$, x_1 and x_2 are similarly defined and ϵ is assumed to be Gaussian. Hence, we can also write this relationship in terms of a distribution over the mixed speech features y as

$$p(y|x_1, x_2) = N(y; x_1 + \ln(1 + \exp(x_2 - x_1)), \psi) \tag{5.21}$$

where ψ is the variance of ϵ, and $N(y|\mu, \psi)$ denotes a normal density function in y with mean μ and variance ψ.

The transformations that we have applied to the model above are the same as the first steps in the calculation of the Mel frequency cepstrum features with the exception that we did not perform the Mel-scale warping before applying the log transform.

For the purpose of signal reconstruction, we are interested in likely values of the two composite signals, given the noisy speech. By recasting this relationship in terms of a likelihood $p(y|x_1, x_2)$, and using prior models for the two signals $p(x_1)$ and $p(x_2)$, we can arrive at a posterior distribution for the joint distribution $p(x_1, x_2|y)$ from which we can easily get the posterior distributions for the component signals $p(x_1|y)$ and $p(x_1|y)$. This will be described in the next section.

By inverting the procedure described above we can reconstruct an estimate of each signal. To do this we find the MMSE estimate for the signal $\hat{x_1}$ and calculate the inverse Fourier transform

$$\hat{x_1}[t] = IFFT(\exp(\hat{x_1}) \cdot \angle Y) \tag{5.22}$$

where $\hat{x_1} = \int x_1 p(x_1|y)dx_1$. The same is done for $\hat{x_2}$. In this reconstruction, we have used the original phases from the mixed signal.

5.2.3 Inference

We now turn our attention to the procedure for estimating the posterior for the clean speech log-magnitudes $p(x_1|y)$. For this we employ the Algonquin method. Extensive evaluations of this framework have been performed in the context of robust speech recognition. In previous work, speech and noise models have either been in the "low-resolution" log-Mel-spectrum domain, or in the truncated cepstrum domain. Here we briefly outline the Algonquin procedure. Detailed discussions can be found in [29, 52].

At the heart of the Algonquin method is the approximation of the posterior $p(x_1, x_2|y)$ by a Gaussian.

The true posterior

$$p(x_1, x_2|y) \propto p(y|x_1, x_2)p(x_2)p(x_1) \tag{5.23}$$

is non-Gaussian, due to the non-linear relationship in Eqn. (5.20). In Eqn. (5.23) $p(x_1)$ is the model for the first speaker, $p(x_2)$ is the model for the second speaker, and $p(y|x_1, x_2)$ is the likelihood function discussed above.

We use a mixture of Gaussians to model both speech signals. Hence

$$p(x_1) = \sum_{s_1} p(s_1)p(x_1|s_1) = \sum_{s_1} \pi_{s_1} N(x_1|\mu_{s_1}^{x_1}, \Sigma_{s_1}^{x_1}) \tag{5.24}$$

and similarly for $p(x_2)$. The construction of the speech models will be discussed below.

Due to the non-linear relationship between x_1 and x_2 for a given y, the true posteriors $p(x_1, x_2|y)$ is non-Gaussian. We wish to approximate this posterior with a Gaussian posterior. The first step is to linearize the relationship between x_1 and x_2.

For notational convenience, we write the stacked vector $z = [x_1^T x_2^T]^T$ and we introduce the function $g(z) = x_1 + \ln(1 + \exp(x_2 - x_1))$.

If we linearize the relationship of Eqn. (5.20) using a first order Taylor series expansion at the point z_0, we can write the linearized version of the likelihood

$$p_l(y|x_1, x_2) = p_l(y|z) = N(y; g(z_0) + G(z_0)(z - z_0), \Psi) \tag{5.25}$$

where z_0 is the linearization point and $G(z_0)$ is the derivative of g, evaluated at z_0. We can now write a Gaussian approximation to the posterior for a particular speech and noise combination as

$$p_l(x_1, x_2, y|s^{x_1}, s^{x_2}) = p_l(y|x_1, x_2)p(x_1|s^{x_1})p(x_2|s^{x_2}) \tag{5.26}$$

It can be shown[52] that the $p(x_1, x_2|y, s^{x_1}, s^{x_2})$ is jointly Gaussian with mean

$$\eta_s = \Phi_s \left[\Sigma_s^{-1}\mu_s + G^T\Psi^{-1}(y - g - Gz_0)\right] \tag{5.27}$$

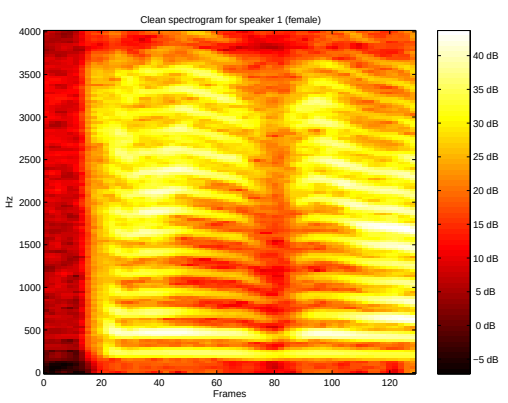
and covariance matrix

$$\Phi_s = \left[\Sigma_s^{-1} + G^T\Psi^{-1}G\right]^{-1} \tag{5.28}$$

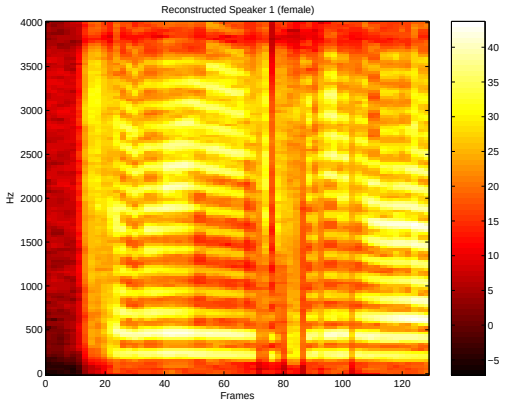
and the posterior mixture likelihood $p(y|s^{x_1}, s^{x_2})$ can be shown to be

$$\gamma_s = |\Sigma_s|^{-1/2}|\Psi|^{-1/2}|\Phi_s|^{1/2} \cdot \exp\Bigg[-\frac{1}{2}(\mu_s^T\Sigma_s^{-1}\mu_s + (y_{obs} - g + Gz_0)^T\Psi^{-1}(y_{obs} - g + Gz_0) - \eta_s^T\Phi_s^{-1}\eta_s)\Bigg].$$

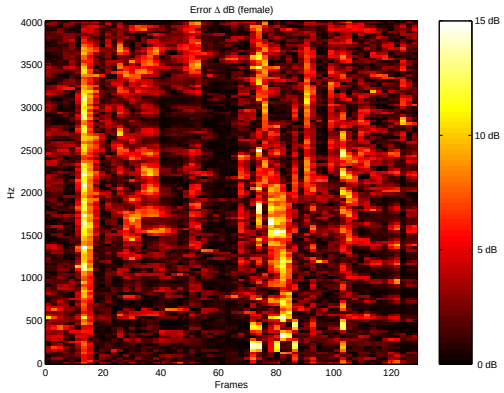
The choice of the linearization point is critical to the accuracy of the approximation. Ideally, we would like to linearize at the mode of the true posterior. In the Algonquin algorithm, we attempt to iteratively move the linearization points towards the mode of the true posterior. In iteration i of the algorithm, the mode of the approximate posterior in iteration $i-1$, μ_{i-1} is used as a linearization point of the likelihood, i.e. $z_i = \mu_{i-1}$. The algorithm converges in 3-4 iterations.



(a) Spectrogram of original signal 1 (female)



(b) Spectrogram for reconstructed speaker 1 (female).



(c) Absolute dB error.

Figure 5.8: Reconstruction errors: (a) The spectrogram for the original clean female signal. (b) The restored spectrogram for speaker 1 (female). (c) The error in absolute dB. The scale has been changed to make the errors clearer. Note that the errors do not suggest that the male speaker is substantially present in the spectrogram.

5.2.4 Experiments

As mentioned above, we use Gaussian mixture models (GMM) to model the speakers. We trained two speaker independent gender dependent models. Each model had 512 mixtures of 128 dimensions. The training set was the clean training set from the Aurora 2 robust speech recognition data set.

Exact inference of a single frame of speech requires the evaluation of every combination of the female and male speaker models. As each models contains 512 mixtures the number of combinations that must be evaluated is 262144. Each combination requires 3-5 iterations in 128 dimensions. Hence, exact inference has complexity $O(m \cdot n \cdot d \cdot i)$ where m is the number of mixtures in speaker model 1, n is the number of mixtures in speaker model 2, d is the number of dimensions (frequency bins) and i is the number of iterations of the algorithm. The computational complexity is therefore considerable.

The test set was constructed from the Aurora 2 test-set. This set contains files with spoken digits, sampled at 8k Hz. Files from test Set A were mixed together at equal signal powers (i.e. 0dB SNR). Log spectrum feature vectors were computed using an analysis window of 25 ms and a shift of 10 ms.

We ran the algorithm on 17 files from this test set. No adaptation due to signal gain was necessary for this task, as the training and test sets have similar signal levels.

Figure 5.2.2(a) shows a spectrogram for a portion of a file from the test set. Figure 5.2.2(b) shows the spectrogram for the separated female signal and Figure 5.2.2(c) shows the spectrogram for the separated male signal. Notice that the characteristics of the male and female spectrograms are different, where the fundamental frequency of the female speaker is higher, and the harmonics are spaced further apart. Notice that the harmonics of the male signal are not clearly visible. This aliasing may be reduced by lengthening the analysis window.

Figure 5.2.4(a) shows the spectrogram for the original female component signal. Compare this to the spectrogram for the recovered signal in Figure 5.2.4(b). The

absolute dB errors are shown in Figure 5.2.4(c). The error plot shows that very little of the male speaker remains in the female signal.

The average average gain in SNR was 6.59 dB for the separated female signal, and 5.51dB for the separated male signal. The separation of the female signal is better on average than the male signal. Interestingly, the model works best when there is complete overlap. In low energy frames of the female signal the male signal tends to leak into the separated female signal, but not vice versa.

The acoustic quality of the separated signals is impressive given the difficulty of the task. The suppression of the unwanted speaker in the restored signal is substantial, and leakage from the unwanted speaker is often barely audible. The suppression of the unwanted speaker is better than the above numbers suggest, as the algorithm also introduces some distortion [3].

5.2.5 Discussion and Future Work

The male-male and female-female cross-talking scenarios require that the two speaker models be the same. As this is a symmetrical problem, the components that generate an observation may be correctly identified, but without temporal dependencies, we cannot associate the components through time. The complexity of the inference problem is not substantially increased introducing time dynamics, however the estimation of speaker models is more involved. We are currently exploring ways to do this.

We are also pursuing approximate inference techniques that promise orders of magnitude reduction in computational complexity without resorting to sub-optimal factorizations or mixing approximations.

In this section we have proposed a new method for the cross-talker source separation task, that relies on strong high frequency resolution models of speech. We provide a test set based on the Aurora 2 test set and give quantitative results

[3] Please contact the author for audio samples.

for a portion of this set. The acoustic quality of the results is impressive for this new method.

5.3 Speech Separation with Factorial Hidden Markov Models

Abstract

We propose a method to exploit audio and potentially visual cues to enable speech separation under non-stationary noise and with a single microphone. We revise and extend HMM-based speech enhancement techniques, in which signal and noise models are factorially combined, to employ novel signal HMMs in which the dynamics of narrow-band and wide band components are factorial. We avoid the combinatorial explosion in the factorial model by using a simple approximate inference technique to quickly estimate the clean signals in a mixture. We also explore incorporating visual lip information. We present a preliminary evaluation of this approach using a small-vocabulary audio-visual database, showing promising improvements in machine intelligibility for speech enhanced using audio information for mixtures of male and female speech. Video information was found to be useful in cases where the speech is of the same gender. the speaker

5.3.1 Introduction

We often take for granted the ease with which we can carry on a conversation in the proverbial cocktail party scenario: guests chatter, glasses clink, music plays in the background: the room is filled with ambient sound. The vibrations from different sources and their reverberations coalesce translucently yielding a single time series at each ear, in which sounds largely overlap even in the frequency domain. Remarkably the human auditory system delivers high-quality impressions of sounds in conditions that perplex our best computational systems. A variety

of strategies appear to be at work in this, including binaural spatial analysis, and inference using prior knowledge of likely signals and their contexts. In speech perception, vision often plays a crucial role, because we can follow in the lips and face the very mechanisms that modulate the sound, even when the sound is obscured by acoustic noise. We introduce a method of speech enhancement using factorial hidden Markov models (fHMMs). We focus on speech enhancement rather than speech recognition for two reasons: first, speech conveys useful paralinguistic information, such as prosody, emotion, and speaker identity, and second, speech contains useful cues for separation from noise, such as pitch. In automatic speech recognition (ASR) systems, these cues are typically discarded in an effort to reduce irrelevant variance among speakers and utterances within a phonetic class.

In speech recognition, HMMs are commonly used because of the advantages of modeling signal dynamics. This suggests the following strategy: train an audio-visual HMM on clean speech, infer the likelihoods of its state sequences, and use the inferred state probabilities of the signal and noise to estimate a sequence of filters to clean the data. In cases where background noise also has regularity, such as the combination of two voices, another HMM can be used to model the background noise. Ephraim [23] first proposed an approach to factorially combining two HMMs in such an enhancement system. In [33] an efficient variational learning rule for the factorial HMM is formulated, and in [77, 5] fHMM speech enhancement was recently revived using some clever tricks to allow more complex models.

The fHMM approach is amenable to audio-visual speech enhancement in many different forms. In the simplest formulation, which we pursue here, the signal observation model includes visual features. These visual inputs constrain the signal HMM and produce more accurate filters. Below we present a prototype architecture for such a system along with preliminary results.

5.3.2 Factorial Speech Models

One of the challenges of using speech HMMs for enhancement is to model speech in sufficient detail. Typically, speech models, following the practice in ASR, ignore *narrow-band*, spectral details (corresponding to upper cepstral components) which carry pitch information, because they tend to vary across speakers and utterances for the same word or phoneme. Instead such systems focus on the smooth, or *wide-band*, spectral characteristics (corresponding to lower cepstral components) such as are produced by the articulation of the mouth. Such wide-band spectral patterns loosely represent *formant* patterns, a well-known cue for vowel discrimination. In cases where the pitch or other narrow-band properties, of the background signals differ from the foreground speech, and have predictable dynamics, such as with two simultaneous speech signals, these components may be helpful in separating the two signals. Figure 5.9 illustrates the analysis of two words into wide-band and narrow-band components.

Wide-band and narrow-band representations of speech are derived by filtering the log power spectrum, or *liftering* the signals into two components. Low-pass filtering the log spectrum yields a wide-band component and high-pass filtering the log spectrum yields a narrow-band component. Using the fourier transform of the log spectrum, the *cepstrum*, we define the wide-band component as the log spectrum derived from lower cepstral coefficients, and the narrow-band component as the log spectrum derived from upper cepstral coefficients. For a given state, these are assumed to be gaussian random variables with diagonal covariances. These components add linearly in the log spectral domain, to form a complete representation of the signal. Thus these components represent signals that are multiplied in the spectral domain, and convolved in the time domain. Hence, with the right liftering cutoff point, they tend to relegate the excitation of the vocal cords to the narrow-band component, and the resonances of the mouth to the wide-band.

However, the wide-band and narrow-band variations in speech are only loosely

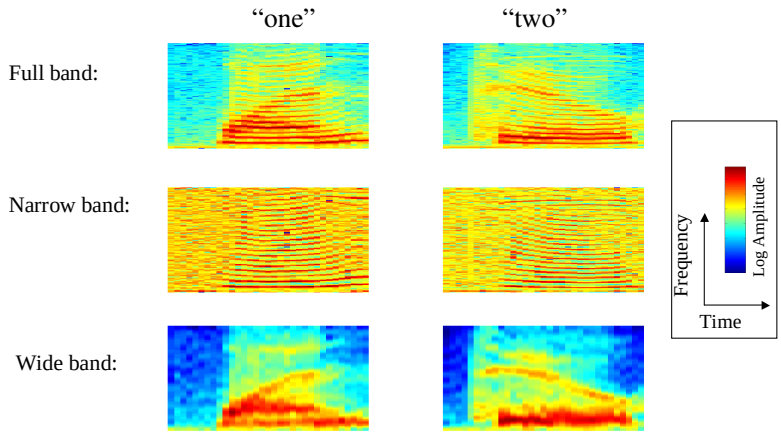
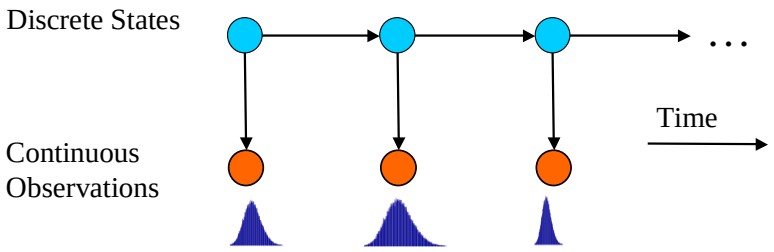


Figure 5.9: Liftering decomposition of speech: full-band, narrow-band, and wide-band log spectrograms of two words. The wide-band log spectrograms (bottom) are derived by low-pass filtering the log spectra (across the frequency domain), and the narrow-band log spectrograms (middle) derived by high pass filtering the log spectra The full log spectrogram (top) is the sum of the two.

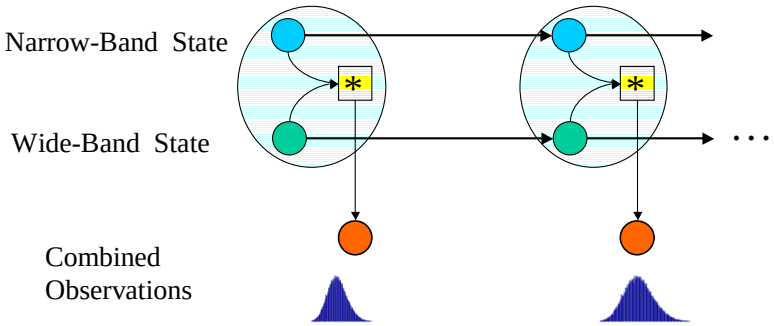
coupled. For instance, a given formant is likely to be uttered with many different pitches and a given pitch may be used to utter any formant. Thus a model of the full spectrum of speech would have to have enough states to represent every combination of pitches and formants. Such a model requires a large amount of training data and imposes serious computational burdens. For instance in [77] a model with 8000 states is employed. When combined with a similarly complex noise model, the composite model has 64 million states. This is expensive in terms of computation as well as the number of data points required for inference. Even if a more modest model with a few hundred states per voice is used, the modeling and implementation of the state transition matrix can be daunting.

To parsimoniously model the complexity of speech, we employ a factorial HMM for a single speech signal, in which wide and narrow-band components are represented in sub-models with independent dynamics. We therefore train the two submodels independently using Gaussian observation probability density functions (p.d.f.) on the wide-band or narrow-band log spectra, with diagonal covariances

for the sake of simplicity. Figure 5.10(a) depicts the graphical model for a single wide or narrow-band component.



(a) simple HMM



(b) factorial speech HMM

Figure 5.10: Factorial HMM model of speech: single HMMs are trained separately on wide-band and narrow-band speech signals (a) and then combined factorially in (b) by adding the means and variances of their observation distributions

To combine the sub-models, we have to specify the observation p.d.f. for a combination of a wide and a narrow-band state, over the log-spectrum of speech prior to liftering. Because the observation densities of each component are Gaussian, and the log-spectra of the wide and narrow-band components add in the log spectrum, the composite state has a Gaussian observation p.d.f., whose mean and variance is the sum of the component observation means and variances. Although the states of the two sub-models are marginally independent they are typically conditionally dependent given the observation sequence. In other words we assume that the state dependencies between the sub-models for a given speech signal can be explained entirely via the observations. Figure 5.10(b) depicts the

combination of the wide and narrow-band models, where the observation p.d.f.'s are a function of two state variables.

When combining the signal and noise models (or two different speech models) in contrast, the signals add in the frequency domain, and hence in the log spectral domain they longer simply add. In the spectral domain the amplitudes of the two signals have log-normal distributions, and the relative phases are unknown. There is no closed form distribution for the sum of two random variables with log-normal amplitudes and a uniformly distributed phase difference. Disregarding phase differences we apply a well-known approximation to the sum of two lognormal random variables, in which we match the mean and variance of a lognormal random variable to the sum of the means and variances of the two component lognormal random variables [31]. Phase uncertainty can also be incorporated into an approximation; however in practice the costs appear to outweigh the benefits.[4] Figure 5.11(a) depicts the combination of two factorial speech models, where the observation p.d.f.s are a function of two state variables.

Using the log-normal observation distribution of the composite model we can estimate the likelihood of the speech and noise states for each frame using the well known forward-backward recursion. For each frame of the test data we can compute the expected value of the amplitude of each model in each frequency bin. Taking the expected value of the signal in the numerator and the expected value of the signal plus noise in the denominator yields a Wiener filter which is applied to the original noisy signal enhancing the desired component. When we have two speech signals one person's noise is another's signal and we can separate both by the same method.

[4]The uncertainty of the phase differences can be incorporated by modeling the sum as a mixture of lognormals that uniformly samples phase differences. Each mixture element is approximated by taking as its mean the length of the sum of the mean amplitudes when added in the complex plane according a particular phase difference, and as its variance the sum of the two variances. This estimation is facilitated by the assumption of diagonal covariances in the log spectral domain.

5.3.3 Incorporating vision

We incorporate vision after training the audio models in order to test the improvement yielded by visual input while holding the audio model constant. A video observation distribution is added to each state in the model by obtaining the probability of each state in each frame of the audio training data using the forward-backward procedure, then estimating the parameters of the video observation distributions for each state, in the manner of the Baum-Welch observation re-estimation formula. This procedure is iterated until it converges. In this way we construct a system in which the visual observations are modular. Figure 5.11(b) depicts the structure of the resulting speech model.

Such a method in which audio and visual features are integrated early in processing is only one of several approaches. We envision other late integration approaches in which audio and visual dynamics are more loosely coupled. What method of audio-visual integration may be best for this task is an open question.

5.3.4 Efficient inference

In the models described above, in which we factorially combine two speech models, each of which is itself factorial, the complexity of inference in the composite model, using the forward-backward recursion, can easily become unmanageable. If K is the number of states in each subcomponent, then K^4 is the number of states in the composite HMM. In our experiments K is on the order of 40 states, so there are 2,560,000 states in the composite model. Naively each composite state must be searched when computing the probabilities of state sequences necessary for inference. Interesting approximation schemes for similar models are developed in [77, 5]. We develop an approximation as follows.

Rather than computing the forward-backward procedure on the composite HMM, we compute it sequentially on each sub-HMM to derive the probability of each state in each frame. Of course, in order to evaluate the observation probabilities of the current sub-HMMs for a given frame, we need to consider the

state probabilities of the other three sub-HMMs, because their means and variances are combined in the observation model. These state probabilities and their associated observation probabilities comprise a mixture model for a given frame. The composite mixture model still has K^4 states, so to defray this complexity during forward-backward analysis of the current sub-HMM, for each frame we approximate the observation mixtures of each of the other three sub-HMMs with a single Gaussian, whose mean and variance matches that of the mixture. Thus we only have to consider the K states of the current model, and use the summarized means and variances of the other three HMMs as auxiliary inputs to the observation model. We initialize the state probabilities in each frame with the equilibrium distribution for each sub-HMM. In our experiments, after a handful of iterations, the composite state probabilities tend to converge. This method and can be also be seen as an approximate belief propagation or sum-product algorithm [55].

5.3.5 Data

We used a small-vocabulary audio-visual speech database developed by Fu Jie Huang at Carnegie Mellon University[5] [42]. These data consist of audio and video recordings of 10 subjects (7 males and 3 females) saying 78 isolated words commonly used for numbers and time, such as,"one" "Monday", "February", "night", etc. The sequence of 78 words is repeated in 10 different takes. Half of these takes were used for training, and one of the remaining takes was used as the test set.

The data set included outer lip parameters extracted from video using an automatic lip tracker, including height of the upper and lower lips relative to the corners the width from corner to corner. We interpolated these lip parameters to match the audio frame rate, and calculate time derivatives.

Audio consisted of 16-bit, 44.1 kHz recordings which we resample to 8000 kHz. The audio was framed at 60 frames per second, with an overlap of 50%, yielding

[5]see http://amp.ece.cmu.edu/projects/AudioVisualSpeechProcessing/

264 samples per frame.[6] The frames were analyzed into cepstra: the wide-band log spectrum is derived from the lower 20 cepstral components and the wide-band log spectrum from the upper cepstra.

5.3.6 Results

Speaker dependent wide and narrow-band HMMs having 40 states each were trained on data from two subjects ("Anne" and "Chris") selected from the training set. A PCA basis was used to reduce the log spectrograms to a more manageable size of 30 dimensions during training. This resulted in some non-zero covariances near the diagonal in the learned observation covariance matrices, which we discarded. An entropic prior and parameter extinction were used to sparsify the transition matrices during training [13].

The narrow-band model learned states that represented different pitches and had transition probabilities that were non-zero mainly between neighboring pitches. The narrow-band model's video observation probability distributions were largely overlapping, reflecting the fact that video tells us little about pitch. The wide-band model learned states that represented different formant structures. The video observation distributions for several states in the wide-band model were clearly separated, reflecting the information that video provides about the formant structure.

Subjectively the enhanced signals sound well separated from each other for the most part. Figure 5.12(a) (bottom) shows the estimated spectrograms for a mixture of two different words spoken by the same speaker – an extremely difficult task. To quantify these results we evaluate the system using speech recognizer, on the slightly easier task of separating the speech of the two different speakers, whose voices were in different but overlapping pitch ranges.

A test set was generated by mixing together 39 randomly chosen pairs of words,

[6]Sine windows were used in analysis and synthesis such that their product forms windows that sum to unity when overlapped 50%. The windowed frames were analyzed using a 264-point fast Fourier transform (FFT). The phases of the resulting spectra were discarded.

one from each subject, such that no word was used twice. Each word pair was mixed at five different signal to noise ratios (SNRs), with the SNR provided to the system at test time.[7] The total number of test mixtures for each subject was thus 195. The separated test sounds were estimated by the system under two conditions: with and without the use of video information.

We evaluated the estimates on the test set using a speech recognition system developed by Bhiksha Raj, using the CMU Sphinx ASR engine.[8] Existing speech HMMs trained on 60 hours of broadcast news data were used for recognition.[9] The models were adapted in an unsupervised manner to clean speech from each speaker, by learning a single affine transformation of all the state means, using a maximum likelihood linear regression procedure [56]. The recognizer adapted to each speaker was tested with the enhanced speech produced by the speech model for that speaker, as well as with no enhancement.

Results are shown in figure 5.12(b). Recognition was greatly facilitated by the audio enhancement. Results with the use of vision were not significantly better than with audio alone in the case of a male and female speaker. However in informal experiments, when both voices were of the same gender, the video was helpful in disambiguating which signal came from which person.

5.3.7 Discussion

We have presented promising techniques for audio-visual speech enhancement. We introduced a factorial HMM to track both formant and pitch information, as well as video, in a unified probabilistic model, and demonstrated its effectiveness

[7] Estimation of the SNR is necessary in practice; however this subject has been treated elsewhere [23] and is beyond the scope of this thesis.

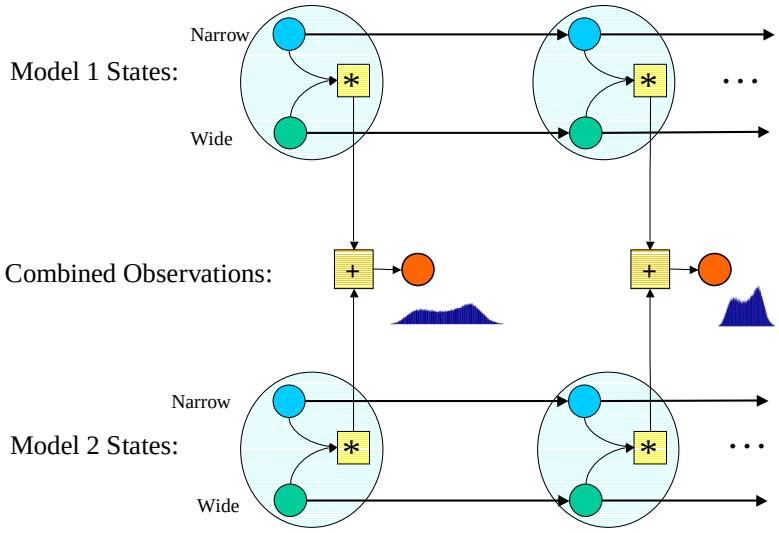
[8] see http://www.speech.cs.cmu.edu/sphinx/.

[9] These models represented every combination of three phones (*triphones*) using 6000 states tied across triphone models, with a 16-element Gaussian mixture observation model for each state. The data were processed at 8 kHz in 25ms windows overlapped by 15ms, with a frame rate of 100 frames per second, and analyzed into 31 Mel frequency components from which 13 cepstral coefficients were derived. These coefficients with the mean vector removed, and supplemented with their time differences, comprised the observed features

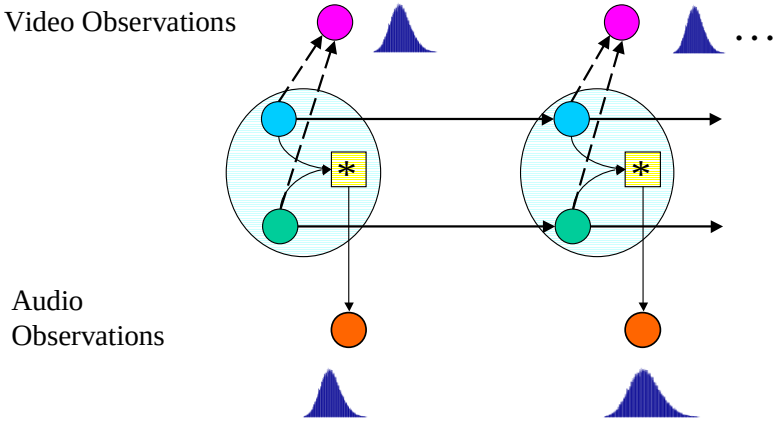
in speech enhancement. The results are tentative given the small sample of voices used; however they suggest that further study with a larger sample of voices is warranted. It would be useful to compare the performance of a factorial speech model to that of each factor in isolation, as well as to a full-spectrum model. Measures of quality and intelligibility by human listeners in terms of speech and emotion recognition, as well as speaker identity, will also be helpful in further demonstrating the utility of these techniques.

5.4 Acknowledgements

Section One of this chapter was adapted from [51], which was published in IEEE International Conference on Acoustics, Speech, and Signal Processing in 2004. My co-authors, Trausti Kristjansson and Hagai Attias, supervised and collaborated with me on the research which forms the basis of this section. Section Two was adapted from [53], which was published in the Proceedings of IEEE Workshop on Automatic Speech Recognition and Understanding in 2003. My co-author, Trausti Kristjansson, supervised and collaborated with me on the research which forms the basis of this section. Section Three was adapted from [38], which was published in Advances in Neural Information Processing Systems in 2002. My co-author, Michael Casey, supervised and collaborated with me on the research which forms the basis of this section.

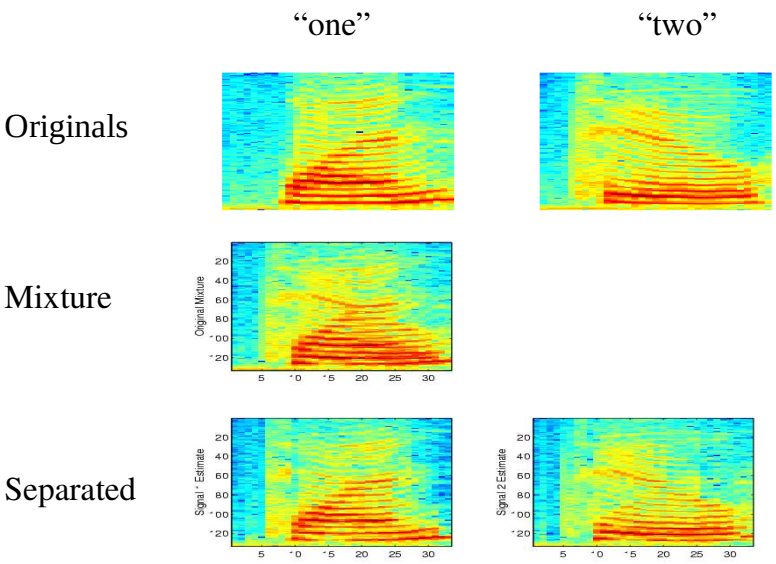


(a) dual factorial HMM

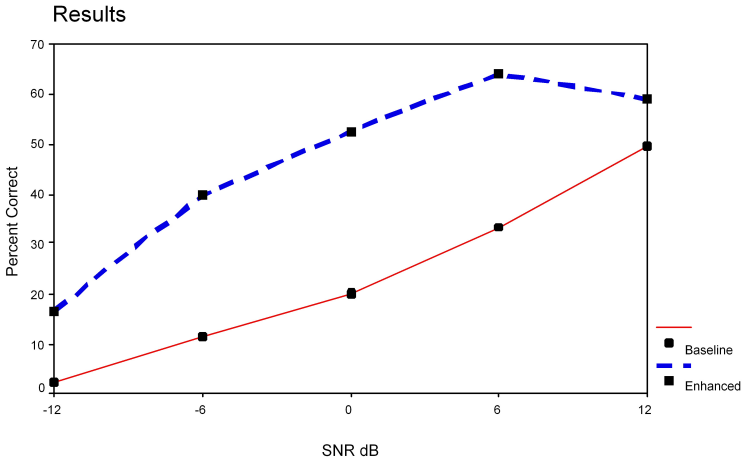


(b) speech fHMM with video

Figure 5.11: Combining speech fHMMs: (a) two speech fHMMs are combined and (b) video observations are added to one of the speech fHMMs.



(a) signal separation spectrograms



(b) automatic speech recognition

Figure 5.12: Resulting spectrograms and recognition rates: (a) spectrograms of separated speech signals for a mixture of two words spoken by the same speaker, and (b) speech recognition performance for 39 mixtures of two words spoken by different speakers.

Chapter 6

Audio Visual Graphical Models for Speech Processing

Abstract

Perceiving sounds in a noisy environment is a challenging problem. Visual lip-reading can provide relevant information but is also challenging because lips are moving and a tracker must deal with a variety of conditions. Typically audio-visual systems have been assembled from individually engineered modules. We propose to fuse audio and video in a probabilistic generative model that implements cross-model self-supervised learning, enabling adaptation to audio-visual data. The video model features a Gaussian mixture model embedded in a linear subspace of a sprite which translates in the video. The system can learn to detect and enhance speech in noise given only a short (30 second) sequence of audio-visual data. In addition it can learns to track the lips as they move around in the video. We show some results for speech detection and enhancement, and discuss extensions to the model that are under investigation.

6.1 Introduction

We often take for granted the ease with which we can carry on a conversation in the midst of noise. Sounds from different sources coalesce and obscure each other making it difficult to resolve what we hear into its constituent parts, and identify its source and content. This auditory scene analysis problem confounds current automatic speech recognition systems, which can fail to recognize speech in the presence of very small amounts of interfering noise. It is well known that in humans, vision often plays a crucial role, because we often have an unobstructed view of the lips that modulate the sound. In fact lipreading can enhance speech recognition in humans as much as removing 15 dB of noise [83]. This fact has motivated efforts to use video information for tasks of audio-visual scene analysis, such as speech recognition and speaker detection [67].

Such systems have typically been built using separate modules for tasks such as tracking the lips, extracting features, and detecting speech components, where each module is independently designed to be invariant to different speaker characteristics, lighting conditions, and noise conditions. One problem with systems designed for a variety of conditions is that there is typically a tradeoff between average performance across conditions and performance on any one condition. Thus a system that can adapt to one's face under the current lighting condition may perform better than one trained for a variety of conditions without adaptation. Another pitfall of modular audio-visual systems is that the modules may be integrated in an *ad hoc* way that neglects information about the uncertainty within models, as well as neglecting statistical dependencies between the modalities. The two problems are related in that unsupervised adaptation is greatly facilitated by seeking agreement between modules in different modalities [17].

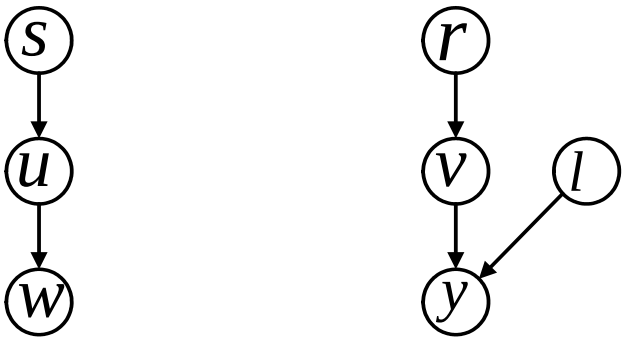
We address the integration and the adaptation problems of audio-visual scene analysis by using a probabilistic generative model to combine video tracking, feature extraction, and tracking of the phonetic content of audio-visual speech.

A generative model offers several advantages. It allows us to capture and exploit dependencies between modalities. It gives us principled methods of inference and learning across modalities that ensure the Bayes optimality of the system. It allows us to extend the model, for instance by adding temporal dynamics, in a principled way while maintaining optimality properties. It also allows us to use the same model for a variety of inference tasks, such as enhancing speech by reading lips, detecting whether a person is speaking, or predicting the lips using audio.

In previous work it has been shown that a generative model could capture dependencies between time delays of the speech signal in two microphone signals and motion in a camera of the image of the speaker [7]. In that system the cross-modal calibration parameters were automatically discovered during unsupervised learning, and the audio time delay signal was able to bootstrap learning of the visual tracking, yielding much better tracking when the multi-modal system was adapted jointly than when the models were adapted independently in each modality. Audio-visual speech recognition has been explored in a variety of papers [67]. Speaker localization has been handled in other systems such as [40]. Unsupervised learning of video tracking has been developed for example in [28]. Adaptation to noise conditions has been demonstrated in for example [4].

Here we develop a generative model that accomplishes aspects of all of these works. It fuses audio and video by learning the dependencies between the noisy speech signal from a single microphone and the fine-scale appearance and location of the lips during speech. One possible scenario for this model is that of a human computer interaction: a person's audio and visual speech is captured by a camera and microphone mounted on the computer, along with other interfering signals in the room: machine noise, another speaker, and so on.

We construct the dependencies between elements in the model based on knowledge of the relationships between variables that generate the data. For instance knowing what the lips look like helps us infer the speech signal in the presence of noise because they are causally related. The converse is also true: we can use what is being said to help infer the appearance of the lips, along with



(a) Audio Model (b) Video Model

Figure 6.1: Audio and Video Models

the camera image, and our belief about where the lips are in the image. In turn, we need to know what the lips look like in order to find them in the image. We parameterize these relationships in a tractable way to define our model.

In the rest of the section we present the inference and learning rules of the model, and describe some experiments using it to detect and enhance speech in the presence of noise, and while tracking the lips in video. Finally we suggest possible extensions to the model.

6.2 Audio Model

The generative model for audio shown in 6.1(a) is as follows. A windowed short segment or *frame* of the observed microphone signal is represented in the frequency domain as $w_k \in \mathbb{C}$ where k indexes the frequency band. This observed quantity is described as the clean speech signal u_k amplified by scalar h and corrupted by Gaussian noise having *precision* (inverse variance) ϕ_k. The speech signal is in turn modelled as a zero mean Gaussian mixture model with state variable s and state-dependent precision σ_{sk}, which corresponds to the inverse power of the

frequency band k for state s. Thus the audio model is

$$\begin{aligned} p(u \mid s) &= \prod_k \mathcal{N}(u_k \mid 0, \sigma_{sk}) \\ p(s) &= \pi_s \\ p(w \mid u) &= \prod_k \mathcal{N}(w_k \mid hu_k, \phi_k) \ . \end{aligned} \tag{6.1}$$

where for the complex sub-band components u_k a Gaussian distribution is defined as $\mathcal{N}(u \mid \rho, \sigma) = \frac{\sigma_k}{\pi} e^{-\frac{\sigma_k}{|} u_k - \rho_k|^2}$ with mean ρ_k and precision σ_k. This is a joint distribution over the real and imaginary parts of u_k, hence the power of two disparity from the usual Gaussian.

We model the audio using a zero mean Gaussian, rather than the traditional cepstral coefficients used in speech recognition. One advantage of this approach is that we can easily extend the model to use phase from inferred microphone delay as in [7]. To use cepstral coefficients derived from the log power spectrum and accommodate inferences about phase is a challenging problem. In addition, the inference of the clean speech in noise is greatly simplified, both mathematically and computationally. The use of nonlinear features such as cepstral components requires either iterative optimization procedures ([29]) or approximations ([30]) to perform noise compensation. Furthermore, whereas cepstral components may work well for speech recognition, high-resolution spectral components may work well for speech enhancement in noisy conditions, because it can take advantage of fine structure in either the signal or the interference [50].

6.3 Video Model

The video model describes an observed frame of pixels from the camera, y as a noisy version of a hidden template v shifted in two dimensions by discrete location parameter l. v in turn is described as a weighted sum of linear basis functions, $A_j \in \mathbb{R}^{N \times 1}$ which make up the columns of A with weights given by hidden variables r. Such a model constitutes a factor analysis model that helps

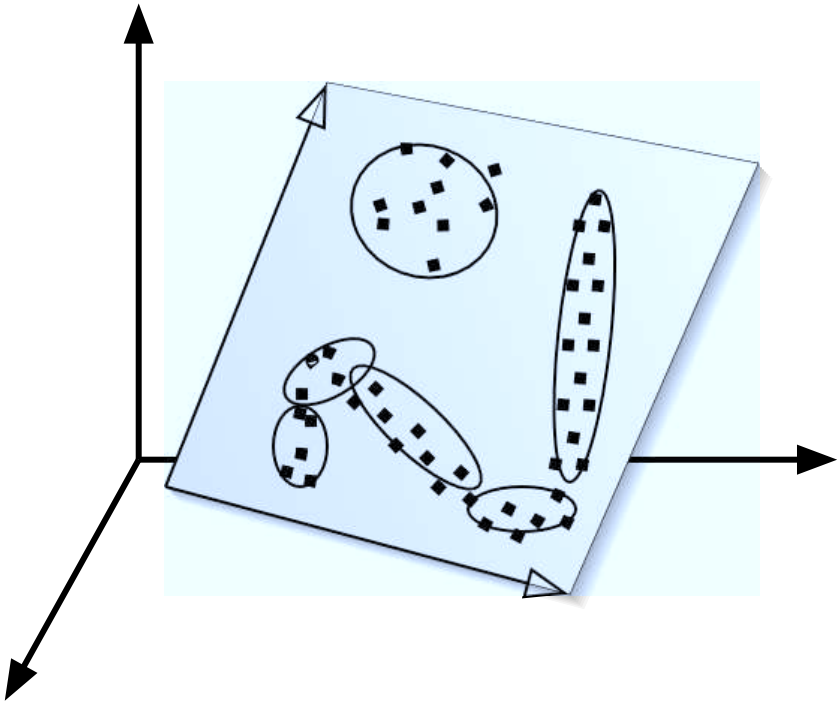


Figure 6.2: Video Model as Embedded Subspace Model

explain the covariance among the pixels in the template v within a linear subspace spanned by the columns of A. This arrangement uses far fewer parameters than the full covariance matrix of v while capturing the most important variances and provides a low-dimensional space of causes, r. In figure 6.2 r is projected into the subspace of v spanned by the columns of A. It is the further structure within this subspace that we hope to describe using audio.

The video model is parameterized as

$$\begin{aligned} p(l) &= constant \\ p(v \mid r) &= \prod_i \mathcal{N}(v_i \mid \sum_j A_{ij} r_j + \mu_i, \nu_i) \\ p(y \mid v, l) &= \prod_i \mathcal{N}(y_i \mid v_\xi(x_i - x_l), \lambda) \ . \end{aligned} \tag{6.2}$$

where ν_i is the conditional precision of each pixel, and μ_i captures part of the

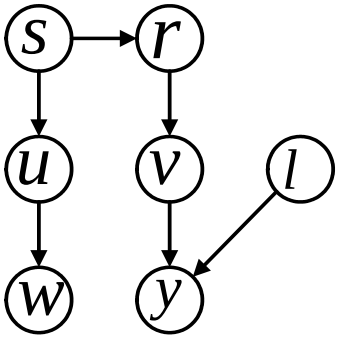


Figure 6.3: Audio-Visual Model

mean that doesn't depend on the factors. The mapping between two-dimensional coordinates and vector indices is handled by the expression $v_\xi(x_i - x_l)$ in which $x_i \in R^{2\times 1}$ is the position of the ith pixel, $x_l \in R^{2\times 1}$ is the position represented by discrete variable l, and $\xi(x)$ is the index of v corresponding to two-dimensional position x.

6.4 Audio-Visual Model

Each model by itself is fairly simple, but by exploiting cross-modal fusion we can obtain a system that is more than just the sum of its parts. We fuse the two models together by allowing the mean and precisions of the hidden video factors r to depend on the states s as illustrated in Figure 6.3:

$$p(r \mid s) = \prod_j \mathcal{N}(r_j \mid \eta_{sj}, \psi_{sj}) \ . \tag{6.3}$$

The discrete variable s now controls the location and directions of covariance of a video representation that is embedded in a linear subspace of the pixels. Thus we can now represent a nonlinear manifold embedded in a linear subspace, as illustrated in Figure 6.2.

6.5 Inference

A variational expectation maximization (EM) algorithm that decouples l from v can be derived to simplify the computation. The posterior $p(u, s, r, v \mid y, w)$ has the factorized form

$$p(u, s, r, v \mid y, w) = q(u \mid s)q(s)q(r \mid s)q(v \mid r, l)q(l) \; . \tag{6.4}$$

For u we get

$$\begin{aligned} q(u \mid s) &= \prod_k \mathcal{N}(u_k \mid \bar{\rho}_{sk}, \bar{\sigma}_{sk}) \\ \bar{\rho}_{sk} &= \frac{1}{\bar{\sigma}_{sk}} h\phi_k w_k \\ \bar{\sigma}_{sk} &= h^2\phi_k + \sigma_{sk} \; . \end{aligned} \tag{6.5}$$

For v we get

$$\begin{aligned} q(v \mid r) &= \prod_i \mathcal{N}(v_i \mid \sum_j \bar{A}_{ij} r_j + \bar{\mu}_i, \bar{\nu}_i) \\ \bar{\nu}_i &= \lambda E_l \alpha_{i+l} + \nu_i \\ \bar{\mu}_i &= \frac{1}{\bar{\nu}_i}(\nu_i \mu_i + \lambda E_l \alpha_{i+l} y_{i+l}) \\ \bar{A}_{ij} &= \frac{\nu_i}{\bar{\nu}_i} A_{ij} \; . \end{aligned} \tag{6.6}$$

For r we get

$$\begin{aligned} q(r \mid s) &= \mathcal{N}(r \mid \bar{\eta}_s, \bar{\psi}_s) \\ \bar{\eta}_s &= \bar{\psi}_s^{-1} \left[\psi_s \eta_s + A^T \mathrm{diag}(\nu)(\bar{\mu} - \mu)\right] \\ \bar{\psi}_s &= A^T \mathrm{diag}(\nu - \frac{\nu^2}{\bar{\nu}})A + \psi_s \end{aligned} \tag{6.7}$$

where $\mathrm{diag}(\nu)$ is a diagonal matrix with the elements of ν along the diagonal

For s we get

$$\begin{aligned} q(s) &= \bar{\pi}_s \\ \log \bar{\pi}_s &= \log \pi_s + \sum_k \left(\log \frac{\sigma_{sk}}{\bar{\sigma_{sk}}} - \phi_k \mid w_k - h\bar{\rho}_k \mid^2 - \sigma_{sk} \mid \bar{\rho}_{sk} \mid^2 \right) \\ &\quad + \log \mid \psi_s \bar{\psi}_s^{-1} \mid - \frac{1}{2} \sum_j \psi_{sj} (\bar{\eta}_{sj} - \eta_{sj})^2 \\ &\quad - \frac{1}{2} \sum_i \nu_i \left[\sum_j (\bar{A}_{ij} - A_{ij}) \bar{\eta}_{sj} + \bar{\mu}_i - \mu_i \right]^2 \\ &\quad - \frac{\lambda}{2} \sum_i \left[E_l \alpha_{i+l} (y_{i+l} - \sum_j \bar{A}_{ij} \bar{\eta}_{sj} - \bar{\mu}_i)^2 + (\bar{A} \bar{\psi}_s^{-1} \bar{A}^T)_{ii} \right] \end{aligned} \tag{6.8}$$

For l we get

$$\begin{aligned} q(l) &\propto e^{f(l)} p(l) \\ f(l) &= -\frac{\lambda}{2} \sum_i \alpha_{i+l} \left(y_{i+l} - \sum_{sj} \bar{A}_{ij} \bar{\pi}_s \bar{\eta}_{sj} - \bar{\mu}_i \right)^2 . \end{aligned} \tag{6.9}$$

All of the expectations with respect to the hidden location random variable l can be shown to be equivalent to a convolution, and can be efficiently carried out using a fast Fourier transform. To enhance the audio we infer expected value of the audio using the posteriors of u and s calculated above: $\mathrm{E}(u|w,v) = \sum_s \bar{\pi}_s \bar{\rho}_s$. We then invert the Fourier transform and overlap and add using a lapping synthesis window matched to the analysis window.

6.6 Learning

In the M-step we compute the model parameters. The update rules use sufficient statistics which involve two types of averages. We denote by E average w.r.t. the posterior q at a given frame n, and we denote by $\langle \cdot \rangle$ average over frames n. The subscript n will be omitted.

For σ we get

$$\frac{1}{\sigma_{sk}} = \langle |\bar{\rho_{sk}}|^2 + \frac{1}{\bar{\sigma_{sk}}} \rangle \tag{6.10}$$

For h, ϕ we get

$$\begin{aligned} h &= \frac{\mathrm{Re}\sum_k \phi_k \langle w_k E u_k^\star \rangle}{\sum_k \phi_k \langle E \mid u_k \mid^2 \rangle} \\ \frac{1}{\phi_k} &= \langle \mid w_k \mid^2 \rangle - 2h\mathrm{Re}\langle w_k E u_k^\star \rangle + \langle E \mid u_k \mid^2 \rangle \end{aligned} \tag{6.11}$$

where

$$\begin{aligned} Eu_k &= \sum_s \bar{\pi}_s \bar{\rho}_{sk} \\ E \mid u_k \mid^2 &= \sum_s \bar{\pi}_s \left(\mid \bar{\rho}_{sk} \mid^2 + \frac{1}{\bar{\sigma}_{sk}} \right) \end{aligned} \tag{6.12}$$

For A, μ, ν we get

$$\begin{aligned} A &= [\langle E v r^T \rangle - \langle E v \rangle \langle E r^T \rangle][\langle E r r^T \rangle - \langle E r \rangle \langle E r^T \rangle]^{-1} \\ \mu &= \langle E v - A E r \rangle \\ \nu^{-1} &= \mathrm{diag}^{-1} \langle E v v^T - A E r v^T - \mu E v^T \rangle \end{aligned} \tag{6.13}$$

where diag^{-1} in the last equation extracts the diagonal of the matrix as a vector. For the averages we have

$$\begin{aligned} Er &= \sum_s \bar{\pi}_s \bar{\eta}_s \\ Err^T &= \sum_s \bar{\pi}_s \left(\bar{\eta}_s \bar{\eta}_s^T + \bar{\psi}_s^{-1} \right) \\ Ev &= \sum_s \bar{\pi}_s \left(\bar{A} \bar{\eta}_s + \bar{\mu} \right) \\ Evr^T &= \sum_s \bar{\pi}_s \left[\left(\bar{A} \bar{\eta}_s + \bar{\mu} \right) \bar{\eta}_s^T + \bar{A} \bar{\psi}_s^{-1} \right] \\ Evv^T &= \sum_s \bar{\pi}_s \left[\left(\bar{A} \bar{\eta}_s + \bar{\mu} \right) \left(\bar{A} \bar{\eta}_s + \bar{\mu} \right)^T + \bar{A} \bar{\psi}_s^{-1} \bar{A}^T + \bar{\nu}^{-1} \right] \end{aligned} \tag{6.14}$$

Finally, for η, ψ we get

$$\begin{aligned} \eta_{sj} &= \langle \bar{\eta}_{sj} \rangle \\ \frac{1}{\psi_{sj}} &= \langle (\bar{\eta}_{sj} - \eta_{sj})^2 + \left(\bar{\psi}_s^{-1} \right)_{jj} \rangle \end{aligned} \tag{6.15}$$

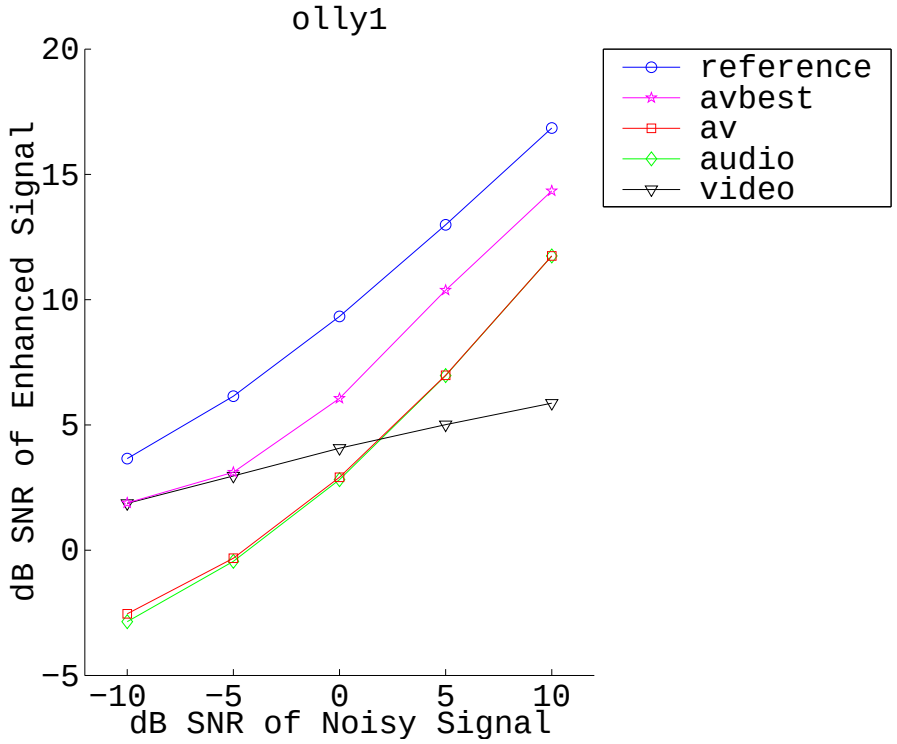


Figure 6.4: Audio-Visual Enhancement Results: For inference, the "video" condition used video only, the "audio" condition used the noisy audio only, the "av" condition weighed the audio and video equally, the "avbest" condition selected the best weights in terms of SNR, and the "reference" condition used clean audio to infer the state.

6.7 Experiments

We conducted experiments to demonstrate the viability of the technique for the tasks of speech enhancement and speech detection. The data consisted of video from the Carnegie Mellon University Audio-Visual Speech Processing Database[1]. We trained a speaker-dependent model having 32 states and 16 subspace basis functions on a 30-second sequence of the face of subject "Jon" cropped around the lip area, with accompanying clean audio speech, then trained the noise model on 10 seconds of an interfering audio signal which in this case happened to be another speaker. The model was then tested on a set of data not used during training, consisting of three different 30-second sequences of the same speaker, mixed with different segments of interfering audio signal.

In order to maximize performance it was necessary to vary the contribution of the audio and video components to the state posterior. At test time we vary the log likelihood of audio and video according to the scheme depicted in 6.5, where a single parameter α controls the relative weights. This scheme ensures that when at one extreme we have a valid audio only model, at the other we have a valid video only model, and in between we have the unaltered audio-visual model. We tested inference under five different settings of *alpha*: the *video* condition used video only($\alpha = 0$) , the *audio* condition used the noisy audio only ($\alpha = 1$), the *av* condition weighed the audio and video log likelihoods equally ($\alpha = 0.5$), the *avbest* condition selected the weights on the log likelihoods to maximize signal-to-noise ratio of the enhanced signal. In order to have an idea of how well we would do if the video provided as much information about the state as the clean audio itself, we used the *reference* condition, in which clean audio was used to infer the state, prior to enhancing the noisy audio.

Signal-to-noise ratio (SNR) was calculated for the enhanced audio signal relative to the clean signal in the time domain (i.e., $SNR = -10 \log_{10} \frac{1}{n} \sum_n (x[n] - y[n])^2$ where x is the clean time domain signal, y is the estimated signal) . Results

[1]by Fu Jie Huang http://amp.ece.cmu.edu/projects/AudioVisualSpeechProcessing

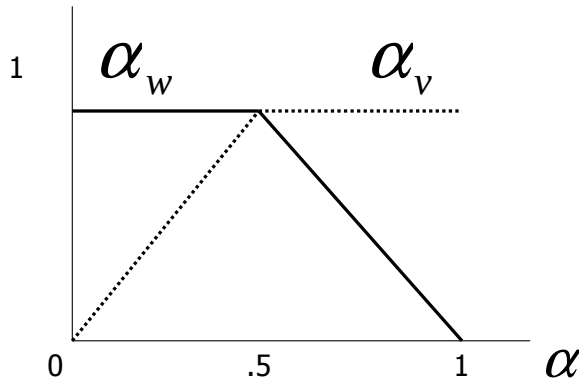


Figure 6.5: Weighting of Audio and Video: $L = \alpha_w \log p(w) + \alpha_v \log p(v)$

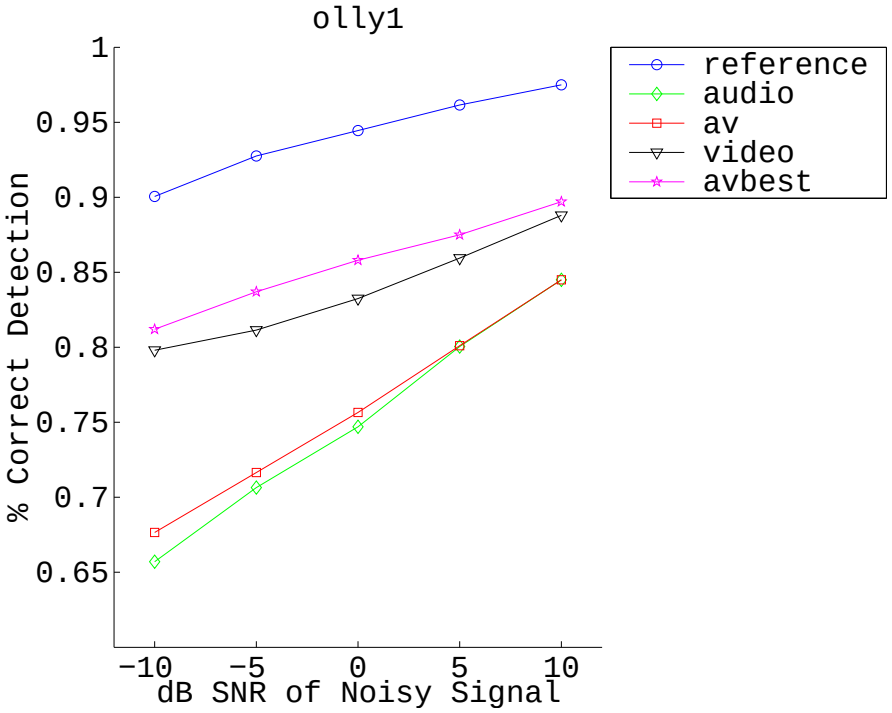


Figure 6.6: Audio-Visual Detection Results

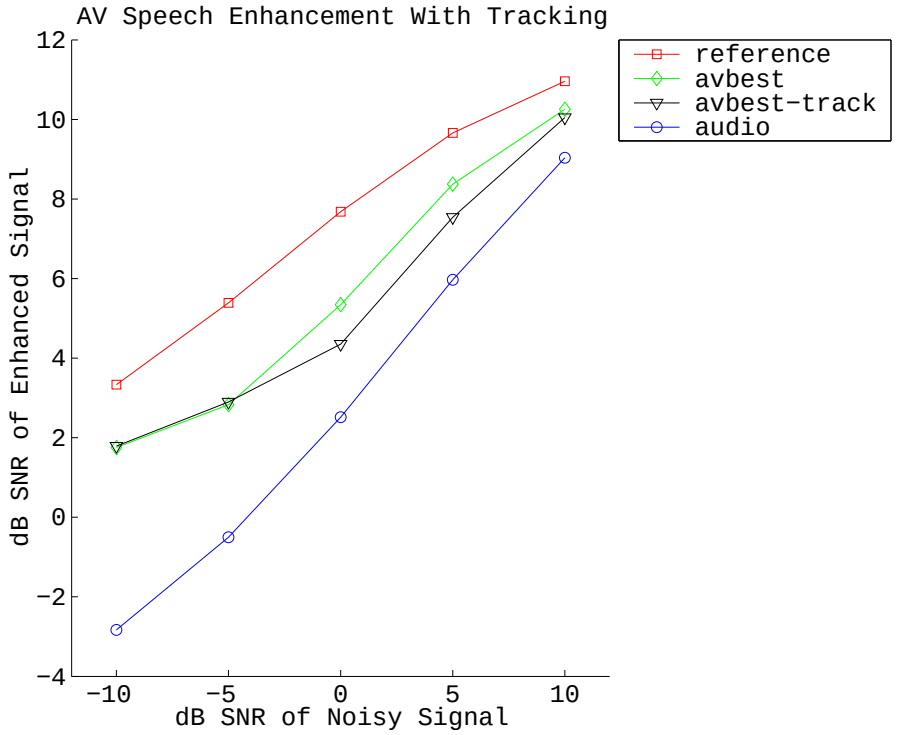
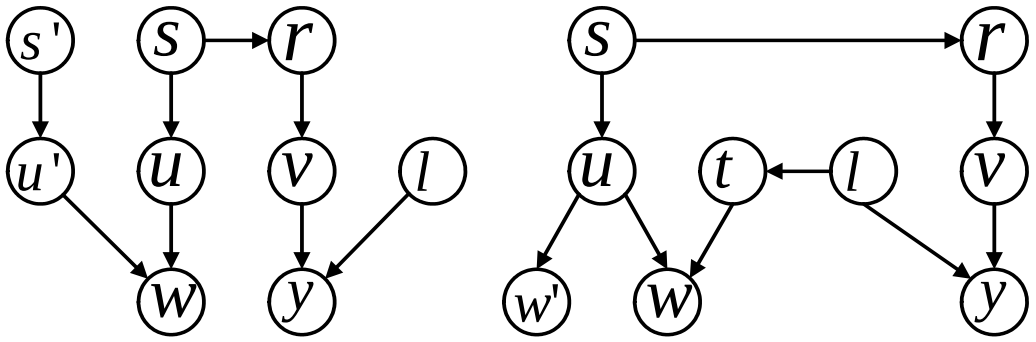


Figure 6.7: Tracking Unaligned Video: "avbest-track" depicts enhancement performance while tracking moving lips, "avbest" is as before for stationary video

for each condition are shown in Figure 6.6. With noisier signals the video-only condition provides better enhancement than audio input. When both audio and video are used without re-weighing the relative importance of each modality, performance is about the same as for audio-only. However, when the balance between audio and video (α is adjusted to maximize performance, the combination does at least as good as either modality alone, and at higher SNRs performance is better than either modality alone. The α values that improved enhancement tended to favor video, especially at lower SNRs.

One plausible explanation for strong video contribution at low SNRs is that with an interfering speaker it is difficult for the audio side of the model to detect when the target speaker is speaking, which is something that is may be easier to determine from video. To test the speech detection performance we turned



(a) Mixture Noise Model (b) Two Microphone Extension

Figure 6.8: Extensions to the Model: (a) Adding a mixture model with state s' and noise observation u'. (b) Adding another microphone w' and relating the hidden time delay t between the microphones to the position l in the video.

the enhancement system into a speech detector by thresholding the power of the enhanced signal in each frame and comparing the resulting classification to that obtained by thresholding the clean signal in the same way. The results shown in Figure 6.7 indicate that speech detection performance with the best setting of α (the setting used for enhancement in the previous experiment) is about the same as that of the video-only model. In contrast, performance under these two conditions diverges in the enhancement experiment at greater SNRs, suggesting that the contribution of video to speech enhancement via detecting the presence of the target speech is further augmented by the contribution of audio to speech enhancement.

In another experiment we used video from the same set, in which the lips are artificially translated in random directions. Figure 6.7 shows that tracking is able to almost completely compensate for lip motion.

6.8 Extensions

The systematic nature of the graphical model framework allows us to integrate our generative audio-visual model with other submodules that we have

investigated. In particular, the simplistic noise model we have used can be replaced with a mixture model, as depicted in Figure 6.8(a), where state s' and noise observation u' can better describe non-stationary interference. The addition of another microphone would further improve both noise robustness and tracking. The model with this extension is depicted in Figure6.8(b), where the second microphone w' is the hidden time delay t between the microphones to the position l in the video, as in [7].

The model could also be extended with dynamics, making it a form of hidden Markov model. This would also open up the possibility of exploring time-asynchrony between audio and video streams which may help in interpreting anticipatory motion of the lips. We also intend to explore other applications of the current model, such as unsupervised speaker localization.

6.9 Conclusions

We have derived and implemented the inference and learning rules for a novel audio-visual model. The model is capable of tracking a simple object in video as it translates and changes shape within a low-dimensional linear subspace of pixels. We have shown that the model can be applied to audio-visual speech enhancement, and that useful relationship between audio and video can be learned from small amounts of data. Thus it may be possible to adapt such a system to the prevailing noise and lighting as well as individual differences among speakers in a given situation. Although results are preliminary, we feel this is a promising step toward a completely unsupervised system that can usefully combine the two modalities in a principled way.

6.10 Acknowledgements

The contents of this chapter are adapted from [37], which was published in IEEE International Conference on Acoustics, Speech and Signal Processing in 2004. My

co-authors, Hagai Attias, Nebojsa Jojic, and Trausti Kristjansson, supervised and collaborated with me on the research which forms the basis of this chapter.

Chapter 7

Conclusion

The research presented in this thesis illustrates how a generative model approach offers some unique advantages toward an understanding of perception. There are several interesting properties of generative probabilistic models which serve to highlight different facets of the research presented in this dissertation. The *axiomatic approach* helps us understand existing algorithms in terms of assumptions made about the world. The explicit formulation of *independence assumptions* in generative probabilistic models places strict limits on what relationships can be encoded in the model and help our understanding of what relationships are important in the data. Often we can perform useful perceptual inference when we assume things are independent even when we know they are not, and such independence assumptions often lead to computational savings. The encoding of *uncertainty* in probabilistic models adds an important dimension to our models, and helps us answer questions about *where the information is* in the data. The phenomenon of *explaining away* results when different models have to compete to explain the observed data, and underscores the importance of *knowing when you don't know* about what is observed. Generative models are particularly suited for unsupervised *adaptation* which is a perceptual computation between inference and learning in time-scale that can greatly simplify more complex inference problems.

7.1 Contributions

Specific contributions, along with highlights of the corresponding models that reflect some of these advantages are summarized here:

- **Template matching and optic flow were integrated under the same framework and conditions under which each approach is optimal were thus explained.**

  The *G-flow* or *generative flow* model of Chapter 4 unifies two seemingly different basic vision methods: *optic flow* and *template matching*, and provides a principled method for weighing the relative contributions of each in a tracking situation. This work demonstrates how graphical models can help us understand existing algorithms.

  To perform inference in this model, we developed principled learning of a template in the context of structure-from-motion. A new form of particle filtering was developed that took advantage of continuity in the observations, allowing the particles to be sampled from a tight proposal distribution.

  Thinking about this problem as a graphical model also forced us to worry about the background. The pixels that were outside the object being tracked have to be generated by something in this framework. Supplying a background model gave us the opportunity to exploit explaining-away to help locate the object.

- **Convolutional hidden Markov models, were devised in which maps of states have location-independent state transitions, leading to efficient exact inference.**

  This model, presented in Chapter 3, illustrates the importance of uncertainty and use of assumptions of dynamics. The tracking of uncertainty over time allows a system to maintain multiple hypotheses about the world, rather

than a single one. Modeling dynamics improves robustness in clutter, by constraining hypotheses about the motion to realistic trajectories.

Figure 7.1 shows the color tracker following the face, when another large face-colored object comes on screen. Under the face color model this new object is more likely than the face itself, however the assumption of smooth motion prevents the tracker from switching to the new object.

Figure 7.1: Evolution of priors, likelihoods, and posteriors: The two rows of images represent hypotheses at two different scales. The left column represents the most likely hypotheses. The center column represents the prior distribution based on the previous image, the right column shows the posterior distribution of hypotheses.

The use of a background model illustrates *explaining away*. A particular color might be prevalent in the face, but also present in the background. Without a background model the system would be likely to think that the background was the object of interest. However, with a background model to explain away those areas, the object model was forced to concentrate on other more distinguishing colors.

Adaptation is also demonstrated in this model. The color of a face is highly dependent on the lighting in the environment. The lighting on the face can change dramatically even for someone sitting at a desk: for instance, when they turn one direction or another the intensity of illumination can vary dramatically. Thus color trackers typically use a model of hue, which

ignores variations in intensity and saturation. We adapted the color model to the face and background, by periodically identifying its location using a color-independent face finding system. With this adaptation, invariance to illumination changes was not as important, and there was an advantage to using a full RGB color model instead of just hue.

- **Single-microphone sound separation was addressed using high-resolution speech models, including a novel factorial model that unifies pitch tracking and formant tracking.**

  Independent component analysis (ICA) takes advantage of independence constraints to infer the signals emitted from each source given observations of multiple different mixtures of the sounds, such as recordings form microphones at different locations. For ICA, relatively little information need be known about the source signals because several mixtures are available. What is important is the independence assumption. However humans seem to excel at hearing sounds in a single mixture of source signals, a condition in which independent component analysis completely fails. To perform such a feat some knowledge of the sounds appears to be necessary. The generative model framework allows us to propose richer models of sounds along with the same independence assumption, and thereby solve this more difficult problem.

  In the sound separation experiments of Chapter 5, two independent models of sound generation competed to explain the observed spectrum. This competition automatically solved the problem of determining which parts of the sound were masked by the other signal, and effectively treated them as missing variables. Thus each model could "explain away" part of the spectrum, leaving the rest to be explained by the other model. The inference of the hidden sound spectrum, automatically reconstructed these hidden components. Figure 7.1 shows the original mixture and clean signal log spectra, and Figure 7.1 shows the reconstructed female speech sounds with

error bars. Notice that the uncertainty of the reconstructed signals is much smaller where it is observed than where it is masked.

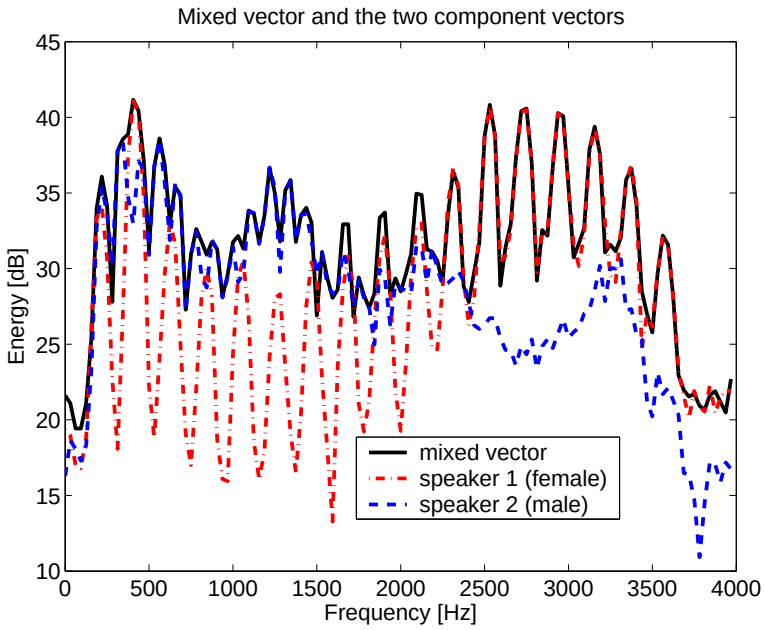


Figure 7.2: Speech spectra: Mixed signal input feature vector (solid black line), speaker 1 feature vector (red dotted line) and speaker 2 feature vector (blue dashed line)

Independence assumptions also can influence the model complexity, and therefore control for over-fitting with limited data. In Chapter 5 two different sound separation models are proposed, a Gaussian mixture model (GMM) without dynamics and the hidden Markov model (HMM), with dynamics. The GMM assumes independence between two speech signals and independence over time. Because the full spectrum is modeled a large number of states (512) are required to well represent the many different possible speech spectra. Thus we cannot trivially add dynamics to obtain an HMM model. Naively formulated, such a model would be intractable because it generates very large 512×512-state transition matrices. However when the problem is to separate very similar signals, such mixtures of speech produced

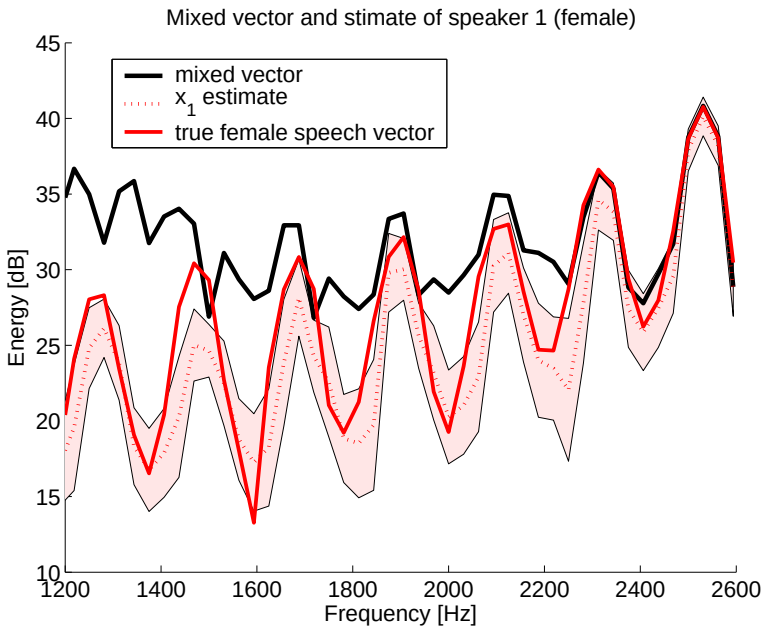


Figure 7.3: Speech posterior spectra: The posterior estimate for speaker 1 (female). Notice that signal is effectively masked in the lower portion of the frequency range. The algorithm is able to *reconstruct* these values due to the strong prior model. The shaded area represents the uncertainty of the estimate and is the first standard deviation. Notice that the uncertainty is larger for 'submerged' estimates.

by speakers of the same gender, constraints on dynamics are necessary. By observing that the signal can be divided into quasi-independent subspaces (representing the excitation and filter components of speech – see Chapter 5.3), it is possible to reduce the complexity of the dynamics by having fewer states in each subspace. Exact inference is still difficult in this model but approximate inference by iteratively updating sub-models makes the problem easily tractable and yields good results.

- **A novel cross-modal graphical model was developed that can perform both speech enhancement and other inference tasks such as video enhancement while tracking motion in the video.**

  Traditionally the study of perception has been divided like Aristotelian

faculties into separate modalities. However it is well known that there are important cross-modal effects such as the McGurk effect of speech, in which lipreading influences the interpretation of the phonemes that are heard. Cross-modal learning has an important thread of research ([90, 17]) that makes the case that the errors and noise in a task is different in different modalities, hence their combination can be greater than the sum of the parts. The work presented in Chapter 2 demonstrated that low-level audio-visual synchrony could be a useful cue for audio-visual integration.

The beauty of using a generative model framework for multi-modal perception is that inference can be performed on any of the hidden variables, given any of the observed variables. In the audio-visual generative model proposed in Chapter 6, it was possible to use the same model to infer either the clean audio signal, or the uncorrupted video signal.

The generative video model itself was a novel appearance-based manifold of prototypes in a low-dimensional linear subspace embedded in the high-dimensional space of pixels. This provided a flexible object representation and allowed unsupervised learning and inference to be performed. In the model, the video was understood via its contingencies with the audio, as measured by the contribution of video to speech enhancement.

An additional benefit of the probabilistic formulation used in this model, is that it allows the model to know when one modality becomes less reliable [66]. We call this principle "knowing when you don't know." The trick is to allow a noise model to adapt to current noise conditions in each modality. When an unusual noise situation occurs in one modality, the noise adapts to explain the additional variance, this causes the posterior for that modality to become highly uncertain, and the model naturally relies on the clean modality.

7.2 Summary

This thesis advocates an approach in which generative graphical models encode high-level knowledge of perceptual problems. Advantages of generative models were discussed, and original work was presented that exemplifies this approach. These contributions represent a diverse body of work in which important principles of perception were employed to constrain the models.

Bibliography

[1] C. Andrieu and A. D. N. de Freitas. Rao-blackwellised particle filtering via data augmentation. In *Advances in Neural Information Processing Systems*, number 13. MIT Press, Cambridge, Massachusetts, 2001.

[2] H. Attias. Independent factor analysis. *Neural Computation*, 11(4):803–51, 1999.

[3] H. Attias. Independent factor analysis with temporally structured sources. In *Advances in Neural Information Processing (NIPS)*, 1999.

[4] H. Attias, A. Acero, J. Platt, and L. Deng. Speech denoising and dereverberation using probabilistic models. 2002.

[5] H. Attias, J. C. Platt, A. Acero, and L. Deng. Speech denoising and dereverberation using probabilistic models. In *Advances in Neural Information Processing Systems 13*. 2001.

[6] L. E. Baum, T. Petrie, G. Soules, and N. Weiss. A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. 41:164–171, 1970.

[7] M. Beal, H. Attias, and N. Jojic. Audio-video sensor fusion with probabilistic graphical models. In *Proc. ECCV*, 2002.

[8] A. J. Bell and T. J. Sejnowski. An information-maximization approach to blind separation and blind deconvolution. *Neural Computation*, 7(6):1004–1034, 1995.

[9] A. J. Bell and T. J. Sejnowski. An information-maximization approach to blind separation and blind deconvolution. *Neural Computation*, 7(6):1129–1159, 1995.

[10] P. Bertelson, J. Vroomen, G. Wiegeraad, and B. de Gelder. Exploring the relation between McGurk interference and ventriloquism. In *Proceedings of the 1994 International Conference on Spoken Language Processing*, volume 2, pages 559–562, 1994.

[11] M. B. R. Bhotika. Flexible flow for 3D nonrigid tracking and shape recovery. In *CVPR*, 2001.

[12] A. Blake and A. E. Yuille. *Active vision*. MIT Press, Cambridge, MA, 1992.

[13] M. Brand. Structure learning in conditional probability models via an entropic prior and parameter extinction. *Neural Computation*, 11(5):1155–1182, 1999.

[14] M. Brand. Flexible flow for 3D nonrigid tracking and shape recovery. In *CVPR*, 2001.

[15] H. Burkhardt and B. Neumann, editors. *ICONDENSATION: Unifying Low-Level and High-Level Tracking in a Stochastic Framework.*, volume 1406 of *Lecture Notes in Computer Science*. Springer, 1998.

[16] G. E. H. David H. Ackley and T. J. Sejnowski. A learning algorithm for boltzmann machines. *Cognitive Science*, 9:147–169, 1985.

[17] V. de Sa and D. Ballard. Category learning through multi-modality sensing. *In Neural Computation*, 10(5), 1998.

[18] F. Dellaert, S. Thrun, and C. Thorpe. Jacobian images of super-resolved texture maps for model-based motion estimation and tracking. In *Proc. IEEE Workshop Applications of Computer Vision*, 1998.

[19] J. Driver. Enhancement of selective listening by illusory mislocation of speech sounds due to lip-reading. *Nature*, 381:66–68, 1996.

[20] R. O. Duda and P. E. Hart. *Pattern classification and scene analysis*. John Wiley & Sons, New York, NY, 1973.

[21] S. Edleman and L. M. Vaina. David marr. *International Encyclopedia of the Social and Behavioral Sciences*, 2001.

[22] J. L. Elman. Finding structure in time. *Cognitive Science*, 14(2):179–211, 1990.

[23] Y. Ephraim. Statistical-model based speech enhancement systems. *Proceedings of the IEEE*, 80(10):1526–1554, 1992.

[24] I. R. Fasel, B. Fortenberry, and J. R. Movellan. GBoost: A generative framework for boosting with applications to real-time eye coding. *Computer Vision and Image Understanding*, under review.

[25] D. E. Feldman and E. I. Knudsen. An anatomical basis for visual calibration of the auditiory space map in the barn owl's midbrain. *The Journal of Neuroscience*, 17(17):6820–6837, 1997.

[26] J. A. Feldman and D. H. Ballard. Connectionist models and their properties. *Cognitive Science*, 6:205–254, 1982.

[27] G. S. Fishman. *Monte Carlo Sampling: Concepts Algorithms and Applications*. Sprienger-Verlag, New York, 1996.

[28] B. Frey and N. Jojic. Learning mixture models of images and inferring spatial transformations using the em algorithm. In *Computer Vision and Pattern Recognition (CVPR)*, 1999.

[29] B. Frey, T. Kristjansson, L. Deng, and A. Acero. Learning dynamic noise models from noisy speech for robust speech recognition. *Advances in Neural Information Processing (NIPS)*, 2001.

[30] M. Gales and S. Young. An improved approach to the hidden markov model decomposition of speech and noise. *In Proc. of ICASSP*, pages 233– 236, 1992.

[31] M. J. F. Gales. *Model-Based Techniques for Noise Robust Speech Recognition*. PhD thesis, Cambridge University, 1996.

[32] S. Geman and D. Geman. Stochastic relaxation, gibbs distribution, and the bayesian restoration of images. *IEEE Trans. Patt. Anal. Mach. Intell.*, 6:721–741, 1984.

[33] Z. Ghahramani and M. Jordan. Factorial hidden markov models. In D. S. Touretzky, M. C. Mozer, and M. Hasselmo, editors, *Advances in Neural Information Processing Systems 8*, 1996.

[34] G. Gigerenzer and D. J. Murray. *Cognition as Intuitive Statistics*. Lawrence Erlbaum Associates, 1987.

[35] C. Goodall. M-estimators of location: an outline of the theory. In D. C. Hoglin, F. Mosteller, and J. W. Tukey, editors, *Understanding robust and exploratory data analysis*, Wiley series in probability and mathematical statistics. Applied probability and statistics, chapter 11, pages 339–403. John Wiley, New York, 1983.

[36] J. E. Handschin and M. D. Q. Monte-carlo techniques to estimate the conditional expectation in multi-stage non-linear filtering. *International Journal of Control*, 9(5):547–559, 1969.

[37] J. Hershey, N. Attias, H.and Jojic, and T. Kristjansson. Audio visual graphical models for speech detection and enhancement. In *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Montreal Canada, May 17-21 2004. IEEE.

[38] J. Hershey and M. Casey. Audio-visual sound separation via hidden markov models. In T. G. Dietterich, S. Becker, and Z. Ghahramani, editors, *Advances in Neural Information Processing Systems 14*, pages 1173–1180, Cambridge, MA, 2002. MIT Press.

[39] J. Hershey and M. Casey. Audio-visual sound separation via hidden markov models. In T. G. Dietterich, S. Becker, and Z. Ghahramani, editors, *Advances in Neural Information Processing Systems 14*, pages 1173–1180, Cambridge, MA, 2002. MIT Press.

[40] J. Hershey and J. R. Movellan. Audio vision: Using audio-visual synchrony to locate sounds. In *In Advances in Neural Information Processing Systems 12. S. A. Solla, T. K. Leen and K. R. Muller (eds.) 813-819. MIT Press.*, 2000.

[41] H.-G. Hirsch and D. Pearce. The aurora experimental framework for the performance evaluation of speech recognition systems under noisy conditions. *Proc. of ISCA ITRW Workshop on Automatic Speech Recognition*, 2000.

[42] F. J. Huang and T. Chen. Real-time lip-synch face animation driven by human voice. In *IEEE Workshop on Multimedia Signal Processing*, Los Angeles, California, Dec 1998.

[43] M. Isard and A. Blake. Contour tracking by stochastic propagation of conditional density. *Proc. European Conf. Computer Vision*, pages 343–356, 1996.

[44] M. Isard and A. Blake. Contour tracking by stochastic propagation of conditional density. In *Proc. European Conf. Computer Vision*, volume 58, pages 343–356, Cambridge, UK, 1996.

[45] B. A. J. and T. J. Sejnowski. Edges are the independent components of natural scenes. In *Advances in Neural Information Processng Systems*, volume 9. MIT, 1996.

[46] N. Jojic and B. Frey. Learning flexible sprites in video layers. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR-01)*, pages 199–206. IEEE, 2001.

[47] J. M. R. M. J. Jones. Statistical color models with application to skin detection. *IEEE Computer Vision and Pattern Recognition*, 1:1274–1280, 1999.

[48] M. I. Jordan. Graphical models. *Statistical Science (Special Issue on Bayesian Statistics)*, 2004.

[49] K. H. Knuth. A bayesian approach to source separation. In *Proceedings of the First International Workshop on Independent Component Analysis and Signal Separation: ICA'99, Aussios, France, Jan. 1999*, pages 283–288, 1999.

[50] T. Kristiansson and J. Hershey. High resolution signal reconstruction. In *Proceedings of IEEE Automatic Speech Recognition and Understanding Workshop*, 2003. in press.

[51] T. Kristiansson, J. Hershey, and H. Attias. Single microphone source separation using high resolution signal reconstruction. In *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Montreal Canada, May 17-21 2004. IEEE.

[52] T. Kristjansson. *Speech Recognition in Adverse Environments: A Probabilistic Approach*. PhD thesis, University of Waterloo, Waterloo, Ontario, Canada, April 2002.

[53] T. Kristjansson and J. Hershey. High resolution signal reconstruction. In *In Proc. of IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU 2003)*, 2003.

[54] T. Kristjansson and J. Hershey. High resolution signal reconstruction. In *In Proc. of IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU 2003)*, 2003.

[55] F. R. Kschischang, B. Frey, and H.-A. Loeliger. Factor graphs and the sum-product algorithm. *IEEE Trans. Inform. Theory*, 47(2):498–519, 2001.

[56] C. J. Leggetter and P. C. Woodland. Maximum likelihood linear regression for speaker adaptation of the parameters of continuous density hidden markov models. *Computer Speech and Language*, 9:171–185, 1995.

[57] S. Z. Li, L. Zhu, Z. Q. Zhang, A. Blake, H. Zhang, and H. Shum. Statistical learning of multi-view face detection. In *European Conference on Computer Vision (ECCV)*, Copenhagen, Denmark, May 28-June 2 2002.

[58] B. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 1981.

[59] D. Macho and Y. M. Chen. Snr-dependent waveform processing for improving the robustnes of asr front-end. *In Proc. of ICASSP*, 2001.

[60] D. Marr. *Vision*. Freeman, New York, 1982.

[61] D. Marr and T. Poggio. Co-operative computation of stereo disparity. *Science*, 194:283–287, 1976.

[62] M. L. Minsky and S. Papert. *Perceptrons: An Introduction to Computational Geometry*. MIT Press, Cambridge, Mass., 1969.

[63] J. Movellan, S. J., and J. Hershey. Large-scale convolutional hmms for real-time video tracking. In *Computer Vision and Pattern Recognition (CVPR)*, Washington, DC, June 27 - July 2nd 2004. IEEE. in review.

[64] J. Movellan, T. Marks, J. Hershey, and J. Roddey. 3d tracking of morphable objects using conditionally gaussian nonlinear filters. In *Computer Vision and Pattern Recognition (CVPR) Workshop on Generative Models for Vision*, Washington, DC, June 27 - July 2nd 2004. IEEE.

[65] J. R. Movellan, B. Fortenberry, and I. Fasel. A generative framework for real-time object detection. *UCSD MPLab Technical Report 2003.02*, 2003.

[66] J. R. Movellan and P. Mineiro. Bayesian robustification for audio visual fusion. In M. Kearns, editor, *Advances in Neural Information Processing Systems*, pages 742–748. MIT Press, Cambridge, Massachusetts, 1998.

[67] C. Neti, G. Potamianos, J. Luettin, I. Matthews, H. Glotin, D. Vergyri, J. Sison, A.Mashari, and J. Zhou. Audio-visual speech recognition, final workshop 2000 report. Technical report, The Johns Hopkins University, Baltimore, MD, October 2000.

[68] S. Oberle and A. Kaelin. Hmm-based speech enhancement using pitch period information in voiced speech segments. *International Symposium on Circuits and Systems ISCAS*, 27:114–120, 1997.

[69] S. E. Palmer. *Vision science: Photons to phenomenology.* Bradford Books/MIT Press, Cambridge, MA, 1999.

[70] B. A. Pearlmutter and L. C. Parra. A context-sensitive generalization of ica. In *International Conference on Neural Information Processing*, Hong Kong, 1996.

[71] L. R. Rabiner and B. H. Juang. *Fundamentals of Speech Recognition.* Prentice-Hall, 1993.

[72] M. Radeau and P. Bertelson. Adaptation to auditory-visual discordance and ventriloquism in semi-realistic situations. *Perception and Psychophysics*, 22:137–146, 1977.

[73] G. H. Recanzone. Rapidly induced auditory plasticity: The ventriloquism aftereffect. *Proceedings of the National Academy of Sciences, USA*, 95:869–875, 1998.

[74] R. A. Redner and H. F. Walker. Mixture densities, maximum likelihood and the em algorithm. *SIAM Review*, 26(2):195–239, 1984.

[75] S. Rennie, P. Aarabi, T. Kristjansson, B. Frey, and K. Achan. Robust variational speech separation using fewer microphones than speakers. In *In Proc. of the 2003 IEEE Conference on Acoustics, Speech, and Signal Processing (ICASSP 2003)*, 2003.

[76] F. Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65:386–408, 1958.

[77] S. T. Roweis. One microphone source separation. In *Advances in Neural Information Processing Systems 13*. 2001.

[78] S. T. Roweis. Factorial models and refiltering for speech separation and denoising. *Eurospeech*, 2003.

[79] D. E. Rumelhart and J. L. McClelland. *Parallel Distributed Processing, Volume 1:Foundations, (ed. w/ PDP Research Group).* MIT Press Cambridge, Massachusett, 1986, 1986.

[80] S. Russel and P. Norvig. *Artificial Intelligence: a Modern Approach.* Prentice Hall, 1995.

[81] M. Seltzer, J. Droppo, and A. Acero. A harmonic-model-based front end for robust speech recognition. *Eurospeech*, 2003. To appear.

[82] M. P. Stryker. Sensory maps on the move. *Science*, pages 925–926, 1999.

[83] W. H. Sumby and I. Pollack. Visual contribution to speech intelligibility in noise. *Journal of the Acoustical Society of America*, 26:212–215, 1954.

[84] J. Tabrikian, S. Dubnov, and Y. Dickalov. Speech enhancement by harmonic modelling via map pitch tracking. *In Proc. of ICASSP*, pages 549–552, 2002.

[85] S. Thrun. Particle filters in robotics.

[86] S. Thrun, W. Burgard, and D. Fox. A probabilistic approach to concurrent mapping and localization for mobile robots. *Machine Learning*, 31(1-3):29–53, 1998.

[87] S. Thrun, J. Langford, and V. Verma. Risk sensitive particle filters. In *Neural Information Processing Systems (NIPS)*, 2001.

[88] L. Torresani, D. Yang, G. Alexander, and C. Bregler. Tracking and modeling non-rigid objects with rank constraints. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2001.

[89] L. Torresani, D. Yang, G. Alexander, and C. Bregler. Tracking and modeling non-rigid objects with rank constraints. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2001.

[90] J. Triesch and C. von der Malsburg. Democratic integration: Self-organized integration of adaptive cues. *Neural Computation*, 13(9):p. 2049–2074, 2001.

[91] Z. Tu, X. Chen, A. Yuille, , and S.-C. Zhu. Image parsing: Segmentation, detection, and object recognition. In *9th IEEE International Conference on Computer Vision (ICCV),*, October 2003.

[92] R. van der Merwe, A. Doucet, N. de Freitas, and E. Wan. The unscented particle filter. In T. Leen, T. Dietterich, and V. Tresp, editors, *Advances in Neural Information Processing Systems*, number 12. MIT Press, Cambridge, Massachusetts, 2000.

[93] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2001.

[94] P. Viola and M. Jones. Robust real-time object detection. Technical Report CRL 20001/01, Cambridge ResearchLaboratory, 2001.

[95] H. von Helmholtz. *On the sensations of tone.* Dover, 1885/1954.

[96] R. M. Warren and R. P. Warren. *Helmholtz on Perception: Its Physiology and Development*. John Wiley and Sons, 1968.

[97] W. Zheng and E. I. Knudsen. Functional selection of adaptive auditory space map by GABAA-mediated inhibition. pages 962–965, 1999.