Discovery and Segmentation of Activities in Video

Matthew Brand, *Member*, *IEEE*, and
Vera Kettnaker, *Member*, *IEEE*

Abstract—Hidden Markov models (HMMs) have become the workhorses of the monitoring and event recognition literature because they bring to time-series analysis the utility of density estimation and the convenience of dynamic time warping. Once trained, the internals of these models are considered opaque; there is no effort to interpret the hidden states. We show that by minimizing the entropy of the joint distribution, an HMM's internal state machine can be made to organize observed activity into meaningful states. This has uses in video monitoring and annotation, low bit-rate coding of scene activity, and detection of anomalous behavior. We demonstrate with models of office activity and outdoor traffic, showing how the framework learns principal modes of activity and patterns of activity change. We then show how this framework can be adapted to infer hidden state from extremely ambiguous images, in particular, inferring 3D body orientation and pose from sequences of low-resolution silhouettes.

Index Terms—Video activity monitoring, hidden Markov models, hidden state, parameter estimation, entropy minimization.

1 Hidden Markov Models

A discrete-time hidden Markov model is a mixture model augmented with dynamics by conditioning its hidden state in time t on that of time $t-1$. An HMM of N hidden states and Gaussian emission distributions is specified by the 4-tuple $\{P_{i|j}, P_i, \boldsymbol{\mu}_i, \boldsymbol{K}_i\}, 1 \leq i, j \leq N$, where $P_{i|j}$ are multinomial transition probabilities between hidden states, P_i is the initial probability of state i, and $\boldsymbol{\mu}_i, \boldsymbol{K}_i$ parameterize emission distributions for each state, in this case, means and covariances of the Gaussian densities Prob($\boldsymbol{x}|$ state i) = $\mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}_i, \boldsymbol{K}_i)$. The likelihood of a multivariate time-series $\boldsymbol{X}$ is the product of all transition and emission probabilities associated with a hidden state sequence $S = \{s_{(1)}, s_{(1)}, , s_{(T)}\}$, summed over all possible state sequences:

$$p(\boldsymbol{X}|\boldsymbol{\theta}) = \sum_{S \in \mathcal{S}^T} \left[P_{s_{(1)}} \mathcal{N}(\boldsymbol{x}_1; \boldsymbol{\mu}_{s_{(1)}}, \boldsymbol{K}_{s_{(1)}}) \prod_{t=2}^{T} P_{s_{(t)}|s_{(t-1)}} \mathcal{N}(\boldsymbol{x}_t; \boldsymbol{\mu}_{s_{(t)}}, \boldsymbol{K}_{s_{(t)}}) \right]. \tag{1}$$

Dynamic programming algorithms are available for the basic inference tasks: Given a time-series, the Viterbi algorithm computes the most probable hidden state sequence; the forward-backward algorithm computes the data likelihood and expected sufficient statistics of hidden events such as state transitions and occupancies. These statistics are used in Baum-Welch parameter reestimation to maximize the likelihood of the model given the data. The expectation-maximization (EM) algorithm for HMMs consists of forward-backward analysis and Baum-Welch reestimation iterated to convergence at a local likelihood maximum.

The principle of maximum likelihood (ML) is not valid for small data sets; in most vision tasks, the training data is rarely large enough to "wash out" sampling artifacts (e.g., noise) that obscure the data-generating mechanism's essential regularities. It is not widely appreciated that this is an acute problem in hidden-variable models, where most of the parameters are only supported by small subsets of the data. That, combined with the fact that the models have high-order symmetries that allow many different parameterizations of the same distribution, results in a learning problem that is riddled with local optima. Consequently, ML hidden-variable models are typically both under-fit, failing to capture the hidden structure of the signal, and over-fit, with a surfeit of weakly supported parameters that inadvertently model accidental properties of the signal such as noise and sample bias. This leads to poor predictive power and modest generalization that supports only limited inference tasks, such as classifying one of a small set of events of interest.

We advocate replacing the Baum-Welch formulæ with parameter estimators that minimize entropy. Entropy minimization exploits the duality between learning and compression to approximate an optimal separation between essential properties (regularities and hidden structure in the data that should be captured by the model) and accidental properties (noise and sampling artifacts that should be ignored). In doing so, it reveals hidden structures in the data that tend to be highly correlated with meaningful partitions of the data-generating mechanism's behavior.

In this article, we outline entropy minimization for HMMs and show how three video interpretation tasks can be treated as problems of inferring hidden state: annotating office activity, monitoring traffic intersections, and inferring 3D motion from monocular video. A common thread in these applications is the emphasis of inference over image processing or scene modeling; high-level inferences are made from relatively impoverished sensing via learned priors rather than engineered algorithms.

2 Relation to Vision and Learning Literatures

Small HMMs and HMM-based hybrids have enjoyed wide success in spoken word and visual gesture recognition, partly because it is feasible to hand-design an adequate transition topology, which is the dominating constraint in the learning problem. However, their usefulness for more complicated systems is seriously curtailed by the fact that for models of nontrivial size, one must probe for an appropriate topology using very expensive search techniques. Although the literature of HMM-based visual event classification is extensive, to our knowledge it does not touch on the focus of this article—discovering a set of event types that efficiently describes action in the video—so we will only review it categorically. One may consult the proceedings [11], [8], [14] to see the bulk of the visual monitoring and event recognition literature in the last two years: Over 30 such papers use a small battery of HMMs as a postvisual processing event classification engine. Nearly all use the HMMs as a standard Bayesian MAP classifier: Each HMM is trained on a few examples of the event of interest; after training, novel events are classified via likelihood ratios. The HMMs have a hand-designed topology, typically corresponding to a band-diagonal transition matrix; the number of bands and states is found by experimentation. Related models, such as dynamic Bayes' nets, also require careful hand-crafting. The problem of finding appropriate HMM topologies is the subject of intense research interest outside of the vision literature; [5] reviews 12 of the most current approaches to learning HMM topology, all involving heuristic generate-and-test search or heuristic clustering methods. In this article, we will explore an unsupervised approach in which entropy minimization automatically induces a partitioning of the signal into events of interest. This framework yields monotonic (hillclimbing) algorithms for simultaneous estimation of model topology and parameters. As our applications will show, the result is a single, sparsely connected HMM containing the entire classification engine. This allows us to

- *M. Brand is with Mitsubishi Electric Research Labs, 201 Broadway, Cambridge, MA 02139. E-mail: brand@merl.com.*
- *V. Kettnaker is with the Computer Science Department, Rensselaer Polytechnic Institute, Troy, NY 12180. E-mail: kettnv@cs.rpi.edu.*

Manuscript received 21 Apr. 1999; revised 28 Mar. 2000; accepted 28 Mar. 2000.
Recommended for acceptance by R. Collins.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number 109654.

fold event segmentation and event classification (often done separately and, thus, suboptimally) into a single inference problem for which an efficient solution is readily obtained via the Viterbi dynamic programming algorithm.

3 Entropy Minimization

We outline the entropy minimization framework here and refer readers to [3] for details and derivations. We begin with a dataset $\boldsymbol{X}$ and a hidden-variable probabilistic model whose parameters and structure are specified by the vector $\boldsymbol{\theta}$. In conventional training, one guesses the sparsity structure of $\boldsymbol{\theta}$ in advance and merely reestimates nonzero parameters to maximize the likelihood $P(\boldsymbol{X}|\boldsymbol{\theta})$. In HMMs, the sparsity structure of $\boldsymbol{\theta}$ encodes the transition topology of the finite-state machine (e.g., see Fig. 3); $\boldsymbol{\theta}$ is almost always designed by hand or, if time is not an issue, found via generate-and-test search over a large space of topologies. In entropic estimation, we learn the size of $\boldsymbol{\theta}$, its sparsity structure, and its parameter values simultaneously by minimizing the three entropies

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \big[H(\boldsymbol{\omega}) + D(\boldsymbol{\omega}\|\theta) + H(\boldsymbol{\theta})\big], \tag{2}$$

where $H(\omega)$ is the entropy of the data's expected sufficient statistics and can be interpreted as the expected cost of coding the data relative to the model, $D(\omega\|\theta)$ is the cross-entropy between the (expected) sufficient statistics of the data and the model distribution and measures the expected cost of coding aspects of the data not captured by the model, and $H(\boldsymbol{\theta})$ is an entropy measure on the model itself and, depending on its formulation, can be interpreted either as the entropy of the distribution or the expected coding costs of the model itself. This formulation identifies entropy minimization as learning by compression; recent results in algorithmic complexity theory state that any such method approximates an optimal strategy for model identification [15]. In this regard, it is kin to the minimum message length (MML) [16] and minimum description length (MDL) [13] frameworks, but differs in that its wholly continuous formulation yields well-behaved parameter estimators without appeal to code-based discretizations of the real line. Equation (2) and its estimators provide an embedding of the problem of finding model structure in a smooth differentiable function whose optimization is much more computationally attractive than MML/MDL-based model selection.

Minimizing (2) is equivalent to maximizing the posterior probability given by Bayes' rule,

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \Big[P(\boldsymbol{\theta}|\boldsymbol{X}) \propto P(\boldsymbol{X}|\boldsymbol{\theta}) e^{-H(\boldsymbol{\theta})}\Big]. \tag{3}$$

The prior $e^{-H(\boldsymbol{\theta})}$ expresses a desire for the smallest, least ambiguous, and most specific model that is compatible with the data. It is the prior that (asymptotically) maximizes the amount one can expect to learn from any data [6]. In information theory terms, the prior divides the posterior by the volume of typical set (inputs that the model accepts as likely), thereby favoring highly selective models. In Markov models, the prior also divides the posterior by the perplexity (branching factor) of the process, thereby favoring highly predictive models.

The maximum a posteriori (MAP) estimators for the component distributions of an HMM are as follows: For spread parameters such as the Gaussian covariance $\boldsymbol{K}$ over N samples $\{\boldsymbol{x}_1, ..., \boldsymbol{x}_n\}$, the entropic prior $|\boldsymbol{K}|^{-1}$ favors minimum volume covariances; the estimator is

$$\hat{\boldsymbol{K}} = \frac{\sum_i^N \boldsymbol{x}_i \boldsymbol{x}_i^\top}{N + Z}. \tag{4}$$

For multinomial densities over N alternatives, the entropic prior $\boldsymbol{\theta}^{\boldsymbol{\theta}} = \prod_i^N \theta_i^{\theta_i}$ favors near-deterministic odds; the estimator is given by the fix-point

$$\hat{\lambda} = \frac{1}{N}\sum_i^N \frac{\omega_i}{\theta_i} + Z \log \theta_i + Z, \tag{5}$$

$$\hat{\theta}_i = \frac{-\omega_i / Z}{W(-\omega_i e^{1-\lambda/Z}/Z)}, \tag{6}$$

where $\boldsymbol{\omega}$ is a vector of sufficient statistics (e.g., event counts), W is the Lambert inverse function satisfying $W(x)e^{W(x)} = x$, and Z is a negative temperature term. Z varies the strength of the prior under the control of a temperature variable $T = 1 - Z$ that is initialized high and decays to zero ($Z = 1$) over the course of training. This gives deterministic annealing within EM, which speeds learning and turns EM into a quasi-global optimizer.

These estimators have a bias that tends to concentrate evidential support on just a few parameters. Consequently, MAP reestimation extinguishes excess parameters and maximizes the information content of the surviving parameters. If one begins training with an overcomplete model (comprising the union of an exponential number of embedded submodels), entropic estimation whittles away any components of the model that are not in accord with the hidden structure of the signal. This allows us to learn the proper size and sparsity structure of a model. Of course, there is no "correct" number of states for a continuous signal, but if there is insufficient data to support many states, some will be automatically removed.

Because the likelihood function of an HMM is a sum over an exponential number of hidden state sequences, its entropy rate $H(\boldsymbol{\theta})$ is not directly calculable. It is, however, upper-bounded by the summed entropies of the HMM's component distributions, so independently minimizing the entropies of the components drives down the entropy of the whole. Moreover, in training this causes the distribution over hidden state sequences to collapse onto a single state sequence, at which point the likelihood function factors into its component distributions and the estimators above become exact.

Algorithmically, the application to HMMs is straightforward: One replaces the estimators in the maximization step of EM with those given above. T is initially set high, then made to decay exponentially. This obliterates initial conditions and forces EM to explore the large-scale structure of the energy surface, by holding entropy high, before committing to a region of parameter space. As entropy is driven low, the model simplifies as the distribution sharpens, parameters expire, and the emission distributions segregate. As in ordinary EM, training ends when the estimators converge to a fixed point.

One complication arises in the numerical instability of computing (5) for very small θ_i using finite-precision floating point numbers. Brand [5] shows how to use the expectation-step statistics to detect parameters that vary the entropy more than the fit and, therefore, contain almost no information. These can be profitably zeroed with a gain in the prior that exceeds the loss in the likelihood. This accelerates learning and can help significantly to sculpt an appropriate model structure out of the initial overcomplete model. As we will show below, entropic estimation of HMMs often recovers a finite-state machine that is illuminative of the structure of the mechanism that generated the data.

3.1 Examples

The left side of Fig. 1 shows an HMM entropically estimated from very noisy samples of a system that orbits in a figure-eight. Even though the data is noisy and has a continuation ambiguity where it crosses itself, the entropically estimated HMM recovers the deterministic structure of the system. The

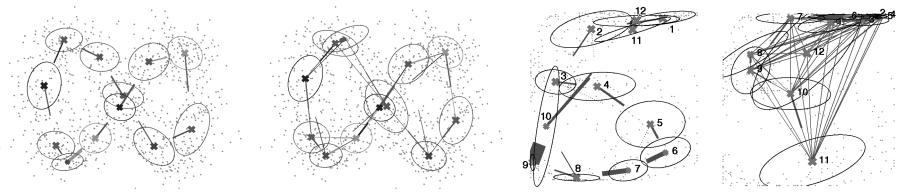

Fig. 1. Left: HMMs estimated entropically and conventionally from identical initial conditions and projected onto $\{x, y\}$ figure-eight training data (gray dots). $\times$s and ellipses indicate Gaussian means and covariances of the hidden states; half-arcs point to transitionable states. Right: Entropic and conventional HMM models of pen-strokes for the digit "5," estimated from identical initial conditions on pen-position data taken at 5 msec intervals from 10 different individuals. Any random walk on the entropic model produces a recognizable digit. (The state machine can begin in state 1 or state 2; state 10 is a pen-up.)

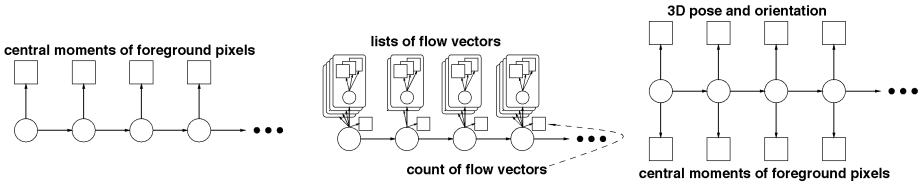

Fig. 2. Graphical models of the office activity HMM (Section 4), the traffic MOMC-HMM (Section 5), and the pose-estimation HMM (Section 6). Circles' represent hidden state variables; boxes represent observed variables; arrows indicate conditional dependencies. Time advances on the horizontal axis.

conventionally estimated HMM gets "lost" at the crossing, bunching states near the ambiguity and leaving many of them incorrectly overconnected. This allows multiple circuits on either loop as well as numerous small circuits on the crossing and on the lobes—it is even possible for a random walk to traverse the conventionally estimated HMM backward! The conventionally estimated HMM derives most of its selective power from the Gaussian emission distributions and is thus little more than a mixture model. Its transition structure allows backward motion, skipping states, orbits on just one lobe, jumps across lobes. In short, it does not strongly constraint the range of time-series the model will accept. In contrast, the entropically estimated HMM has essentially recovered the deterministic structure of data-generating mechanism.

Entropic estimation typically succeeds in extracting recognizable structure even where conventional estimation fails to produce a reasonable mixture model, as shown in the handwriting models in the right side of Fig. 1. Entropic estimation induces an interpretable automaton that captures essential structure and timing of the pen-strokes, as well as variations in their ordering between writers. In contrast, the conventional model is uninterpretable. It is the additional precision of entropic models that makes them useful for analysis of the complex time-series that arise in video monitoring.

We now turn to the uses of HMM and HMM variants in interpreting video streams. We present three quite varied systems whose useful output is a hidden state sequence. This sequence, and the associated data likelihood, supports annotation, anomaly detection, low bit-rate coding of scene activity, and, in the last example, reconstruction. The graphical models (graphs of dependence structure) of the three systems are depicted in Fig. 3.

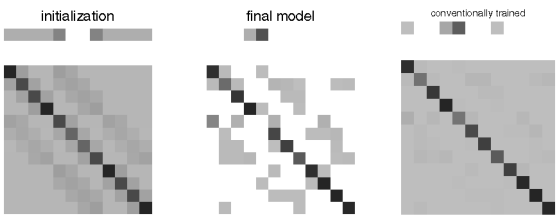

Fig. 3. The transition matrix at initialization (left), and sparsified by entropic estimation (middle). Conventional estimation fails to discover any such structure (right) and, in fact, does not move transition probabilities far from initialization. Darker entries signify higher probabilities; the top row shows initial-state probabilities.

4 Learning a Model of Office Activity

Office activity is an interesting test of entropic estimation's ability to discover hidden structure because of the challenging range of time spans: Fast events such as picking up the phone take just a half-second, while other activities such as writing take hours. The results below show that much of this structure can be discovered from lightweight, coarse visual tracking.

4.1 Image Representation

HMMs require a reasonably short observation vector which represents the content of each image. We used a very basic "blob" representation consisting of ellipse parameters fitting the single largest connected set of active pixels in the image. These pixels are identified by acquiring a static statistical model of the background texture and adaptive Gaussian color/location models of the foreground (pixels that have changed, ostensibly due to motion). Pixels are sorted into foreground or background according by likelihood ratio; morphological dilation connects the foreground pixels using a seed from the previous frame [17]. The observation vector consisted of the ellipse parameters $[\overline{x}, \overline{y}, \Delta\overline{x}, \Delta\overline{y},$ mass, Δmass, elongation, eccentricity], calculated from the mean and eigenvectors of a 2D Gaussian fitted to the foreground pixels. Approximately 30 minutes of data were taken at random at 4 Hz from an SGI IndyCam; after automatic deletion of blank frames (when the subject has left the field of view), roughly 21 minutes of training data remained.

4.2 Training

Three sequences ranging from 100 to 1,900 frames in length were used for entropic training of 12, 16, 20, 25, and 30-state HMMs. States were initialized so that their emission distributions tile the image. Transition probabilities were initialized to prefer motion to adjoining tiles; first-state probabilities were set to zero for nonedge states. It was found that variation in the initial emission distributions or state counts made little difference in the gross structure or performance of the final model. Training took six seconds on an SGI R10000 running Matlab.

4.3 Results

Entropic training yields a substantially simplified transition matrix (Fig. 3) which was automatically converted into a human-readable representation of characteristic office activity. Fig. 4 explains the resulting state machine. We found it fairly easy to label the states by watching the frames they claim in novel video. Once the states

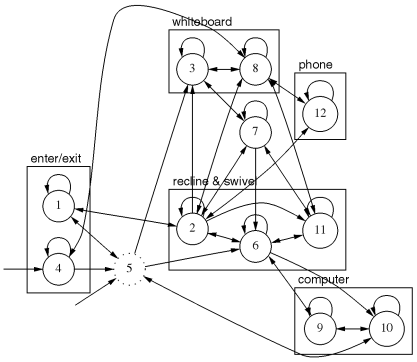

Fig. 4. An activity graph automatically generated from the transition matrix of an entropically estimated HMM. Each state label was determined by watching the video subsequence of all test frames for which the state had occupation probability > 0.99. Characteristic frames for several states are shown in Fig. 5. Some states deserve special explanation: State 5 is a gating state that does not model data but simplifies paths between other states; state 7 responds mainly to elongation and represents getting up and sitting down.

are labeled, the Viterbi state sequence of any novel video provides a frame-by-frame annotation of the activity viewed by the camera.

Note that most of the transitions were extinguished through training. Entropic estimation with larger initial state counts resulted in a similar qualitative structure. In contrast, standard maximum likelihood estimation from identical initial conditions consistently failed to produce interpretable models and generally did not move transition parameters far from their initial values.

4.4 Anomaly Detection

The ability of the model to detect unusual behavior was tested under several conditions to study the significance of the entropically estimated transitions. Four data sets were used: 1) training data, 2) held out test data, 3) reversed held out test data, and 4) data taken after the subject had consumed four cups of espresso. These data sets differ principally in the ordering, rhythm, and timing of actions and therefore emphasize the discriminative power of the transition parameters. There were three test conditions: 1) entropically estimated parameters, 2) conventionally estimated parameters, 3) transition parameters flattened to chance. Condition 3 tests whether the transitions or emission parameters are responsible for the model's selectivity (in conventional HMMs, the emission Gaussians strongly dominate [2]). Fig. 6 shows that the entropic HMM generalized better to test data and was most

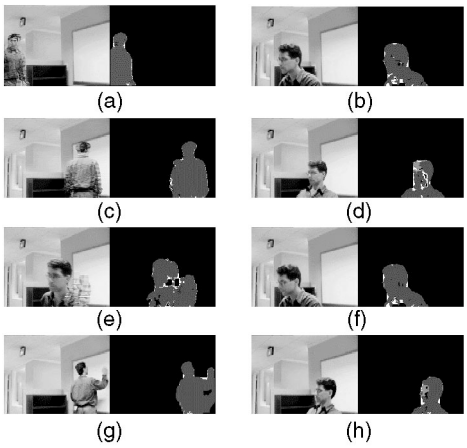

Fig. 5. Characteristic frames and foreground masks for several states in the HMM model of office activity. (a) Entering the room, (b) at the computer, (c) at the white board, (d) sitting, (e) getting the phone, (f) looking for a key, (g) writing, and (h) swiveling right.

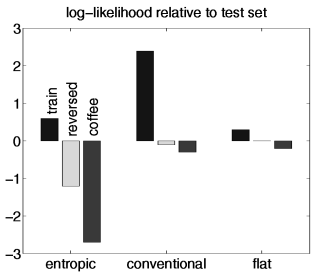

Fig. 6. Model log-likelihoods normalized to sequence length. The entropically estimated model is by far the most discriminative; the conventionally estimated model is the most over-fit. Likelihoods are plotted relative to the test set, which we take to be a de facto standard of normal activity.

successful in distinguishing abnormal behavior (backwards and jittery). The performance of the flattened model shows that little of that selectivity is due just to the emission parameters. In run-time mode, we detect anomalous behavior by looking for windows in which the HMM assigns a very low likelihood to the data. This particular model will signal an alarm if one is asleep in one's chair, but not if one is reading in the chair. It has also detected anomalous behavior due to the office occupant being distracted by events outside the office window and when the office is occupied by a worker with different habits.

This system uses an extremely simple set of visual features and will not observe multiple moving objects, except insofar as the person being tracked reacts to them. We address that limitation next.

5 Monitoring Traffic

Most monitoring applications will require inference about many simultaneously moving objects. Pedestrian plazas and vehicle roadways have varying numbers of participants that constantly enter and exit the scene. An HMM as traditionally formulated has limited applicability because it is fundamentally a model of a single hidden process, observing a single fixed-length observation vector in each time step. Here, we introduce a novel graphical model (joint distribution) that generalizes HMMs to take a varying number of observations per time step and solve for its MAP reestimation formulæ. This model will learn holistic modes of activity in the scene, e.g., the sorts of traffic modes that a traffic engineer would need to know when designing controls for an intersection.

Rather than attempt simultaneous tracking of tens of variable-sized objects, with all the attendant sources of error, we will learn a distribution over low-level image processes. Our image representation is a variable-length list $V^t = \{v_1^t, , v_{|V^t|}^t\}$ of flow vectors between two subsequent images. The list is variable-length because flow vectors smaller than some fixed "noise" magnitude are discarded. The set of moving image fragments in any one frame form several clusters roughly corresponding to objects; over time these clusters roughly follow the geometry of traffic lanes, preferred paths, etc. The patterns of moving traffic are choreographed by the (invisible) traffic lights into phases of horizontal and vertical traffic, as well as implicit subphases with different frequencies of right and left turns. Our model will have to "learn" the typical locations and directions of the moving pixels, as well as the dynamic changes of these patterns through entropy minimization.

The multimodal distribution of the moving pixels can be captured with multivariate Gaussian mixture models. HMMs can be extended to handle multiple observations per time step by treating each frame's flow-list as an observation sequence for the mixture model at that time step. Since mixture models provide a measure on the relative distribution of traffic behavior but not the traffic density, we further augment the emission probabilities with

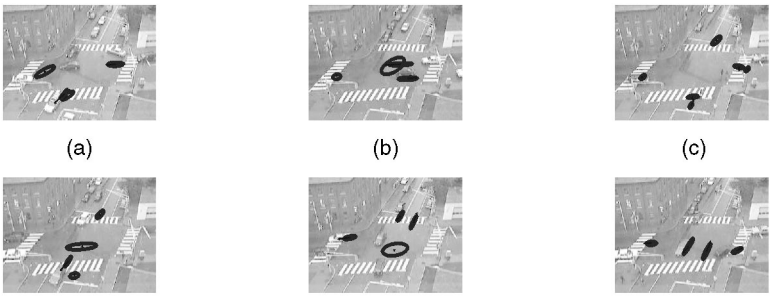

Fig. 7. HMM states learned from a 13,600 frame training sequence. The six MOMC-HMM states learned from a sequence of about four traffic cycles, superimposed on some of their typical frames. Ellipses indicate one standard deviation spatial iso-probability contours of each mixture component; arrows indicate the mean optical flow. Labels were determined as in Fig. 4. The number following the # sign is the mean number of flow vectors the state observes in each frame. (a) State 1: east → south west → south turns; #17, (b) State 2: east-west, all turns; #24, (c) State 3: pedestrians, stopping traffic; #3, (d) State 4: north-south traffic, waits, no turns; #19, (e) State 5: north → west turns; #13, and (f) State 6: north-south, traffic, frequent turns; #26.

a logarithmic flow vector counting variable. We call the resulting model a multi-observation-mixture+counter (MOMC) HMM. The probability that an MOMC-HMM state s_i with M mixtures will observe the list V_t is defined as:

$$P^{MOMC}(V^t|s_i) \quad \stackrel{def}{=} \quad f(|V^t|;\mu_c^i,\sigma_c^i)\prod_p^{|V^t|}\sum_{m=1}^{M} c_m^i\,\mathcal{N}(v_p^t;\boldsymbol{\mu}_m^i,\Sigma_m^i). \quad (7)$$

The first term f is a distribution on the observation count. Typically, f would be a Poisson density function. However, in our current application, the data is *overdispersed*, meaning that its mean and variance do not agree as required for a Poisson fit. A double-Poisson distribution is appropriate in such circumstances, but its normalization is nonanalytic, making it unusable in the setting of expectation-maximization. We currently use a left-truncated Gaussian of the logarithm of the observation counts; the reestimation formulæ below are given for this choice of f. The second term is simply a product of mixture densities, one for each observation. The Gaussian mixture components $\mathcal{N}(v^t;\boldsymbol{\mu}_m^i,\Sigma_m^i)$ are mixed by the coefficients c_m^i. In our application, the mixture Gaussians are 4D observing flow vectors in (x,y,dx,dy) space. Note that each state has its own set of mixture Gaussians. The mixture components model motion in particular directions and locations; the counter variable essentially models the combined surface area of the moving objects. Since the mixture likelihood of an empty set of observations is undefined, we remove frames without motion from the input sequence.

Maximum likelihood parameter reestimation formulæ can be derived by a suitable adaptation of the auxiliary function Q [1], [9]. Let us first define the single-path likelihood of the data for a model with parameters $\boldsymbol{\theta} = \{P_i, P_{i|j}, \mu_c^i, \sigma_c^i, c_m^i, \boldsymbol{\mu}_m^i, \Sigma_m^i\}$ for all states i,j and mixture components m. This is defined relative to a specific sequence of states $\{s_{(1)},\ldots,s_{(t)},\ldots,s_{(T)}\} \in \mathcal{S}^T$ and, for each state $s_{(t)}$ and observation p, a particular mixture component $k_{(t,p)} \in \{1\ldots M\}$:

$$L_{\boldsymbol{\theta}}(V,s,k) \quad \stackrel{def}{=} \quad P_{s_{(1)}}\left(\prod_{t=2}^{T} P_{s_{(t)}|s(t-1)}\right)\prod_{t=1}^{T}\cdot\mathcal{N}\left(\log|V^t|;\mu_c^{s_{(t)}},\sigma_c^{s_{(t)}}\right) \prod_{p=1}^{|V^t|} c_{k_{(t,p)}}^{s_{(t)}}\mathcal{N}\left(v_p^t;\boldsymbol{\mu}_{k_{(t,p)}}^{s_{(t)}},\Sigma_{k_{(t,p)}}^{s_{(t)}}\right).$$

With this definition, the likelihood of the data given the model can be written as the sum of $L_{\boldsymbol{\theta}}(V,s,k)$ over all state and mixture sequences. Finally, we define Q as

$$Q(\boldsymbol{\theta},\bar{\boldsymbol{\theta}}) \quad \stackrel{def}{=} \quad \sum_{s\in\mathcal{S}^T}\sum_{k\in\mathcal{K}} L_{\boldsymbol{\theta}}(V,s,k)\log L_{\bar{\boldsymbol{\theta}}}(V,s,k).$$

The structure of Q as defined above is analogous to that used by Juang et al. [9] and Liporace [10], having a unique maximum that implies a locally maximal improvement of the data likelihood at the same parameter value. Setting the derivation of Q for the new parameters to zero and introducing some Lagrange multipliers, we obtain the following reestimation formulæ for the counter means, the mixture coefficients, and the mixture means, respectively:

$$\hat{\mu}_c^s = \frac{\sum_{t=1}^{T}\gamma_{s,t}\log|V^t|}{\sum_{t=1}^{T}\gamma_{s,t}}, \quad (8)$$

$$\hat{c}_k^s = \frac{\sum_{t=1}^{T}\sum_{p=1}^{|V^t|}\gamma_{s,t}\frac{c_k^s\mathcal{N}(v_p^t;\boldsymbol{\mu}_k^s,\Sigma_k^s)}{\sum_{j=1}^{M}c_j^s\mathcal{N}(v_p^t;\boldsymbol{\mu}_j^s,\Sigma_j^s)}}{\sum_{t=1}^{T}\sum_{p=1}^{|V^t|}\gamma_{s,t}}, \quad (9)$$

$$\hat{\boldsymbol{\mu}}_k^s = \frac{\sum_{t=1}^{T}\sum_{p=1}^{|V^t|}\frac{c_k^s\mathcal{N}(v_p^t;\boldsymbol{\mu}_k^s,\Sigma_k^s)}{\sum_{j=1}^{M}c_j^s\mathcal{N}(v_p^t;\boldsymbol{\mu}_j^s,\Sigma_j^s)}\gamma_{s,t}v_p^t}{\sum_{t=1}^{T}\sum_{p=1}^{|V^t|}\frac{c_k^s\mathcal{N}(v_p^t;\boldsymbol{\mu}_k^s,\Sigma_k^s)}{\sum_{j=1}^{M}c_j^s\mathcal{N}(v_p^t;\boldsymbol{\mu}_j^s,\Sigma_j^s)}\gamma_{s,t}}, \quad (10)$$

where $\gamma_{s,t}$ = Prob(HMM hidden state s explains training frame t) are the hidden state occupation probabilities obtained from the standard forward-backward recursions on the HMM backbone (see [12]). The reestimation formulæ for the covariances are analogous to the reestimation of the means and the reestimation of the transition probabilities and initial state probabilities is identical to that of standard HMMs.

We convert the ML formulæ[1] into MAP entropy-minimizing formulæ by changing the normalizations (divisions in the expectation) according to (4) for covariances and (5) and (6) for coefficients.

5.1 Examples

We pointed the camera out the window at a busy traffic intersection in Cambridge. Sources of image noise include reflections in the windows of buildings in the scene, headlight glare on wet streets, and rapidly varying blur from dirt and rain on the window, which vibrates tympanically in the wind. The optical flow was spatially subsampled by a factor of 10 and thresholded at noise level.

1. We have given the batch-mode EM equations for completeness and because they are the most efficient, computationally. These updates can be converted to a fully online algorithm by windowing the forward-backward analysis over short subsequences and updating the model parameters according to their derivatives.

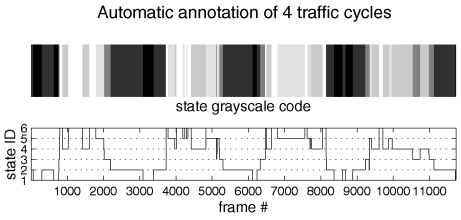

Fig. 8. The mode-filtered sequence of hidden states obtained from parsing an eight minute traffic sequence with a trained model. The plot on the top is a gray-scale rendering of the plot on the bottom. The state labels correspond to the ones in Fig. 7.

Fig. 9. Anomalous situation: A car driving in the wrong lane.

5.1.1 Analysis

Fig. 7 indicates the emission distributions of six MOMC-HMM states learned via entropic estimation. The learned states were fairly easy to interpret, even though the observed patterns were quite complex due to multiple lanes, turns, and Boston driving behavior. States 1 and 2 both represent horizontal traffic, but state 1 is active if there are either left or right turns into the left vertical lane on the lower half of the image. The HMM uses three states to represent vertical traffic; state 6 represents fluid traffic with frequent turns. State 4 is specialized for traffic scenes where a car slowly approaches the center of the crossing and waits there for an opportunity to make a left turn. Finally, state 5 is tuned to rather light North-South traffic with occasional right turns.

Fig. 8 shows the Viterbi hidden-state sequence[2] computed for a novel video sequence. The rhythm of traffic is quite clear: Dark bars correspond to horizontal traffic and light bars to vertical traffic.

5.1.2 Anomaly Detection

As in the office activity model, the MOMC-HMM supports detection of anomalous situations. To do so, we compute for each frame a forward-backward state estimation [12] over a frame neighborhood of size five. Likelihood minima indicate situations that are unusual even when their immediate temporal context is taken into account.

Fig. 9 shows one of the most prominent minima in the eight minute sequence—a case of creative driving in which the car marked by an arrow drives in the oncoming traffic lane in order to make an illegal left turn. The few other minima of that sequence correspond to frames with a lot of noise motion or unusually crowded traffic. Precision could probably be improved by better preprocessing, but the important advantage over other anomaly detection methods, e.g., [7], is that stochastic process models have the potential to detect anomalies that are only anomalies with respect to their dynamic context, e.g., cars running a red light.

2. For purposes of exposition, we renumbered the states so that semantically close states have neighboring numbers.

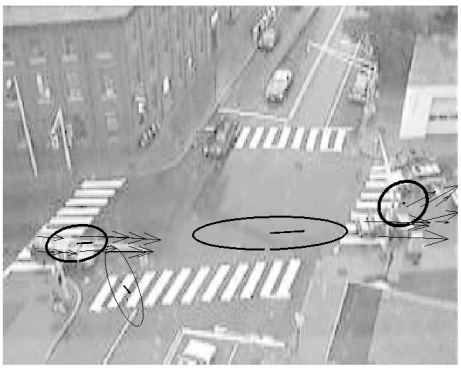

Fig. 10. Predictions: The car on the left is predicted to go either straight or to make a right turn. Arc thickness of the spatial iso-probability ellipses is drawn proportional to the probability of future activation. Gaussians with negligible support are omitted from the figure. The thin arrows originating at the two moving cars show the motion vectors in the current frame. Ellipses and motion vectors are scaled to improve legibility.

5.2 Prediction

It is also possible to make short-term predictions of traffic dynamics by projecting the current hidden state probabilities through the transition matrix. If we combine this with the current relative activations of the mixture components, we get an estimate of where the model is expecting to see motion in the next frame.

Fig. 10 shows how an MOMC-HMM with eight states of six mixtures each predicts for the car on the left to most likely go straight or to take a right turn. A few frames later, the car starts pulling to the right, at which point the support for the right-turn-Gaussian drastically increases.

6 Hidden State from Impoverished Sensing

What if the available sensing never provides enough information to learn the desired hidden state? Consider a rather literal application of Plato's allegory of the cave:[3] If one spent a lifetime observing nothing but shadows, one might never infer the hidden 3D structure that makes two silhouettes of an object two views of the same thing. On the other hand, one who has had experience of the world's 3D structure will have no trouble inferring the true nature of the phenomena causing the shadows. This is a nontrivial learning task because the mapping from shadows to 3D is many-to-many: A human pose casts a multitude of different shadows in different directions; a shadow can fall from a multitude of different poses. These ambiguities must be resolved with constraints from context—previous and subsequent shadows—and with constraints from knowledge of how the body moves. In principle, HMMs allow both kinds of constraints to be propagated forward and backward over arbitrarily long spans of time. The dynamic programming methods that do this are efficient and optimal provided that the HMM transition matrix is sufficiently sparse. Entropically estimated HMMs, in particular, are sparse enough to carry these constraints for hundreds of frames.

We can learn to infer 3D state from 2D evidence by first training on 3D data to learn the true dynamics of the system, then associating families of silhouettes to each hidden state by estimating a second set of emission distributions over the 3D data's silhouettes. We shall call the 3D behavior the *target* system and the observed 2D shadows the *cue* system. The first training phase yields a target HMM plus a matrix $\gamma_{s,t}$ = Prob(hidden state s

3. In the *Republic*, book VII, pp. 140-146, Plato compares our understanding of words and percepts to that of cave-dwellers who have never seen the world outside, only the shadows it casts on the walls of the cave.

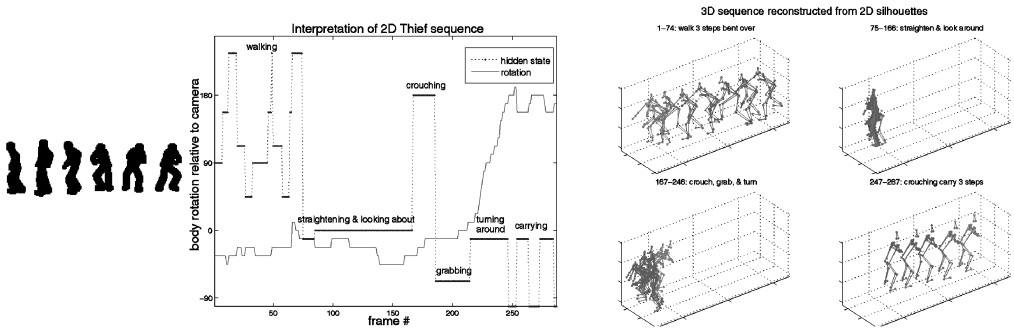

Fig. 11. Left: Some frames from the "thief" sequence. Middle: The hidden state sequence inferred from observing a sequence of silhouettes agrees nicely with a common-sense annotation of the action in the sequence. Right: Inferred 3D structure, rotated 120° from the viewpoint of the camera.

at time t). Using $\gamma_{s,t}$, we estimate a second set of emission means and covariances such that the HMM also observes the synchronized cue signal, specifically, shadows of each 3D pose:

$$\hat{\boldsymbol{\mu}}'_s = E_{\gamma_{s,t}}[\boldsymbol{x}_t] \tag{11}$$

$$\hat{\boldsymbol{\Sigma}}'_s = E_{\gamma_{s,t}}\left[(\boldsymbol{x}_t - \boldsymbol{\mu}'_s)(\boldsymbol{x}_t - \boldsymbol{\mu}'_s)^\top\right], \tag{12}$$

where $\boldsymbol{x}_t$ is the "shadow" observation vector and $E[\cdot]$ is the expectation operator. This associates 2D features to 3D hidden state, yielding a new HMM which has the dynamics of the 3D system, but is driven by 2D evidence.

To handle real vision data, we must also provide some invariances. Translation invariance is easily obtained by representing the shadow by its central moments and scale invariance is approximated by normalizing these moments with the vertical standard deviation. There is no representational trick that gives invariance to rotations out of the image plane, but these can be handled by the HMM itself. We remove all full-body rotation from the 3D training data and estimate an HMM. We then replicate this HMM once for each view, reestimating the emission distributions of each view-specific HMM to cover an appropriately rotated version of the 3D data and its 2D "projection." The 2D "projection" is actually 10 scale-invariant central moments calculated from the silhouette of a rendered "tin-man." We then link together all view-specific HMMs in the following manner: If $P_{i|j}$ is the probability of transitioning into state i from state j in the original HMM and state i' is the ith equivalent state in a duplicated HMM observing data rotated by ϕ, then $P_{i'|j} = P_{i|j}\mathcal{V}(\phi; 0, \kappa)$, where $\mathcal{V}$ is a circular von Mises density (for data normally distributed around a circle) whose concentration κ is set high. A state sequence on this Cartesian product HMM will thereby specify both pose and orientation at every time step. When processing evidence, the Viterbi state sequence will be steered by the shadows, yet constrained to follow the true dynamics of the 3D system.

6.1 Example

We trained an HMM on 800 frames of 3D motion-capture data, replicated to 32 view-angles, and reestimated emission distributions on the appropriated rotated and rendered training data. We then tested on a 2D silhouette sequence in which a "thief" approaches an object, stands up straight to look around, bends over to pick it up, then turns around and runs away, carrying the loot in both hands. Fig. 11 (middle) shows that the hidden state sequence, factored into pose and orientation information, decomposes the sequence into qualitative poses and motions that agree with our narrative. The system correctly infers that the two walking subsequences are in diametrically opposite directions even though the normalized silhouettes contain no information about which way a the figure is facing or moving. This orientation constraint is propagated 170 frames between the two subsequences.

If we augment the original 3D pose data Gaussians with velocity information (deltas), it is also possible to reconstruct a MAP 3D pose time-series $\boldsymbol{Y} = \{\boldsymbol{y}_1, \boldsymbol{y}_2, \boldsymbol{y}_3, ...\}$ from any hidden state sequence $\mathcal{S} = \{s_{(1)}, s_{(2)}, s_{(3)}, ...\}$:

$$\boldsymbol{Y}^* = \arg\max_{\boldsymbol{Y}} \log \prod_t \mathcal{N}([\boldsymbol{y}_t, \dot{\boldsymbol{y}}_t]; \boldsymbol{\mu}_{s_{(t)}}, \boldsymbol{K}_{s_{(t)}}) \mathcal{N}_P(\dot{\boldsymbol{y}}_t; \boldsymbol{0}, \boldsymbol{K}_P), \tag{13}$$

where $\dot{\boldsymbol{y}}_t = \boldsymbol{y}_t - \boldsymbol{y}_{t-1}$ and $\mathcal{N}_P(\dot{\boldsymbol{y}}_t; \boldsymbol{0}, \boldsymbol{K}_P)$ is an optional prior on velocities. In minimum-entropy models, equation (13) yields a large quadratic equation whose maximum can be found by solving a sparse system of linear equations [4]. This allows a qualitative reconstruction of the 3D structure (Fig. 12), biased by the training set (in this case, motion capture data for a hunched-back, video-game monster, which accounts for the differences in posture). Note that this gives an extremely compressed encoding of the 3D motion inferred from the video—just 10 bits per frame to encode the hidden state sequence.

7 Discussion

We have shown that, by minimizing the entropy of its component distributions, an HMM's internal state machine can be made to organize observed activity into highly interpretable hidden states that capture the dynamical regularities of the training set. Entropy-minimized models show markedly superior performance in traditional tasks such as classification and prediction. More importantly, the facts that the model is well-attuned to the data-generating mechanism's dynamics and that the HMM states are highly interpretable opens up several new useful tasks, in particular, video annotation, low bit-rate coding of scene activity, detection of anomalous behavior, and scene reconstruction from minimally informative sensing.

The discovered hidden states are not guaranteed to coincide with the events that we are interested in detecting, but in our experience they have always been interpretable and useful. In addition, the speech recognition literature details several methods for forcing the semantics of hidden states using hand-labeled

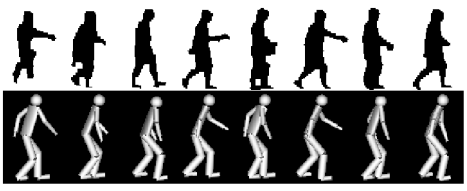

Fig. 12. Every 12th frame from a back-subtracted infrared night-time sequence and the corresponding inferred 3D structure, rendered for clarity.

training data (see [8] for a review). These labor-intensive techniques essentially "tell" the model what entropic estimation discovers in unsupervised learning—how to break up the signal into meaningful units.

ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers for helpful suggestions and comments. V. Kettnaker work on MOMC-HMMs while visiting Mitsubishi Electric Research Labs (MERL).

REFERENCES

[1] L. Baum, T. Petrie, G. Soules, and N. Weiss, "A Maximization Technique Occurring in the Statistical Analysis of Probabilistic Functions of Markov Chains," *Annals of Math. Statistics,* vol. 41, no. 1, pp. 164-171, 1970.

[2] Y. Bengio and P. Frasconi, "Diffusion of Credit in Markovian Models," *Advances in Neural Information Processing Systems,* G. Tesauro, D.S. Touretzky, and T. Leen, eds., vol. 7, pp. 553-560, MIT Press, 1995.

[3] M. Brand, "Pattern Discovery via Entropy Minimization," *Artificial Intelligence and Statistics,* D. Heckerman and C. Whittaker, eds., no. 7, Morgan Kaufmann, 1999.

[4] M. Brand, "Shadow Puppetry," *Proc. Int'l Conf. Computer Vision,* 1999.

[5] M. Brand, "Structure Discovery in Conditional Probability Models via an Entropic Prior and Parameter Extinction," *Neural Computation,* vol. 11, no. 5, pp. 1,155-1.182, 1999.

[6] M. Brand, "Exploring Variational Structure by Cross-Entropy Optimization," *Proc. Int'l Conf. Machine Learning,* P. Langley, ed., 2000.

[7] W. Grimson, C. Stauffer, R. Romano, and L. Lee, "Using Adaptive Tracking to Classify and Monitor Activities in a Site," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* pp. 22-29, 1998.

[8] F. Jelinek, *Statistical Methods for Speech Recognition.* MIT Press, 1998.

[9] B. Juang, S. Levinson, and M. Sondhi, "Maximum Likelihood Estimation for Multivariate Mixture Observations of Markov Sources," *IEEE Trans. Information Theory,* vol. 32, no. no. 2, pp. 307-309, 1986.

[10] L. Liporace, "Maximum Likelihood Estimation for Multivariate Observations of Markov Sources," *IEEE Trans. Information Theory,* vol. 28, no. 5, pp. 729-734, 1982.

[11] *Proc. Int'l Conf. Automatic Face and Gesture Recognition,* A. Pentland and I. Essa, eds., 1997.

[12] L.R. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," *Proc. IEEE,* vol. 77, no. 2, pp. 257-286, 1989.

[13] J. Rissanen, *Stochastic Complexity and Statistical Inquiry.* World Scientific, 1989.

[14] *Proc. DARPA Image Understanding Workshop,* T. Strat, ed., 1998.

[15] P. Vitanyi and M. Li, "Ideal MDL and Its Relation to Bayesianism," *ISIS: Information, Statistics and Induction in Science,* pp. 282-291, Singapore: World Scientific, 1996.

[16] C. Wallace and P. Freeman, "Estimation and Inference by Compact Coding," *J. Royal Statistical Soc., Series B,* vol. 49, pp. 240-251, 1987.

[17] C. Wren, A. Azarbayejani, T. Darrell, and A. Pentland, "Pfinder: Real-Time Tracking of the Human Body," *Proc. SPIE,* vol. 2,615, 1995.

[18] *Proc. Int'l Conf. Automatic Face and Gesture Recognition,* M. Yachida, ed., 1998.