Coupled Hidden markov models for complex action recognition

Matthew Brand, Nuria Oliver, and Alex Pentland
brand@media.mit.edu
Vision and Modeling Group, MIT Media Lab
Cambridge, MA 02139-1130

Abstract

*We present algorithms for coupling and training hidden Markov models (*HMM*s) to model interacting processes, and demonstrate their superiority to conventional* HMM*s in a vision task classifying two-handed actions.* HMM*s are perhaps the most successful framework in perceptual computing for modeling and classifying dynamic behaviors, because they offer dynamic time warping, a learning algorithm, and a clear Bayesian semantics. However, the Markovian framework makes strong restrictive assumptions about the system generating the signal—that it is a single process having a small number of states and an extremely limited state memory. The single-process model is often inappropriate for vision (and speech) applications, resulting in low ceilings on model performance. Coupled* HMM*s provide an efficient way to resolve many of these problems, and offer superior training speeds, model likelihoods, and robustness to initial conditions.*

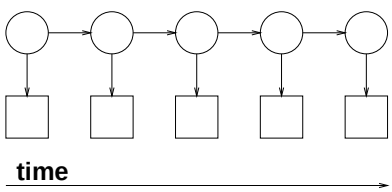
1 Introduction

Computer vision is turning to problems of perceiving and interpreting action, sparking interest in models of dynamical behavior used elsewhere in perceptual computing, particularly hidden Markov models (HMMs). HMMs are presently the most favored model in speech and vision, mainly because they can be learned from data and they implicitly handle time-varying signals. Their clear Bayesian semantics also makes them well-suited for computing with uncertainties.

An HMM is a quantization of a system's configuration space into a small number of discrete states. A single finite discrete variable s indexes the current state of the system. State changes, approximating the dynamics of the system, are described by a table of transition probabilities $P_{s(t)=i|s(t-1)=j}$. This representation succeeds to the degree that the system fits the Markov condition: Any information about the history of the process needed for future inferences must be reflected in the current state. Consequently, HMMs are ill-suited to systems that have compositional state, e.g., multiple interacting processes that have structure in both time and space. For example, in video signals one might want to model the behavior of players in a sport, or, more generally, of participants in multi-place action verbs such as "A gave B the C." We present algorithms for coupling and training HMMs to model interactions between processes that may have different state structures and degrees of influence on each other. These problems often occur in vision, speech, or both—coupled HMMs are well suited to applications requiring sensor fusion across modalities.

2 HMMs and the Markov condition

An HMM is a described by a tuplex $\{S, P_{i|j}, P_i, P_i(o)\}$, consisting of a set of discrete states $S = \{s_1, s_2, s_3, \ldots, s_N\}$, state-to-state transition probabilities $P_{s(t)=i|s(t-1)=j}, 1 < i, j < N$, prior probabilities for the first state $Ps(0) = i$, and output probabilities for each state $P_{s(t)=i}(o(t))$. Graphically, Markov models are often depicted "rolled out in time" as probabilistic independence networks:



Square nodes represent the observations $o(t)$; circular nodes represent the hidden state variable $s(t) \in S$; horizontal arcs represent the transition matrix $P_{s(t)|s(t-1)}$; and parameters associated with the vertical arcs determine the probability of an observation given the current state $P_{s(t)}(o(t))$, e.g., the parameters may be means and covariances of multivariate Gaussians. The state variable and the output vary over time, and at any any time t, memory is limited to the value of state variable $s(t-1)$.

Conventional extensions to the basic Markov model are generally limited to increasing the memory of the system (durational modeling), which give the system compositional state in time. We are interested in systems that have compositional state in *space*, e.g., more than one simultaneous state variable. Recently, Jordan, Saul, and Ghahramani have developed a variety of higher-order HMMs, including factorial HMMs [4] for independent processes; linked HMMs [7] that model noncausal (contemporaneous) symmetrical influences; and hidden Markov decision trees [6] that feature a cascade of noncausal influences from master to slave HMMs. The training algorithms are based on an equivalence between HMMs and a class of Boltzmann

machine architectures with tied weights [8, 9]. The linked HMM excepted, these algorithms use approximation methods from mean field theory in physics.

We present an exact algorithm for coupling two HMMs with causal (temporal), possibly asymmetric influences. Theoretical and empirical arguments for this architecture's advantages can be found in [2]. To illustrate the difference between causal and noncausal couplings, imagine modeling opponents in a tennis match: The noncausal HMM couplings can represent the fact that it is unlikely to see both players playing net simultaneousy; the causal HMM coupling can represent the fact that one player rushing to the net will drive the other back and restrict the kinds of returns he attempts.

The coupling algorithm is based on projections between component HMMs and a joint HMM; in principal it is also possible to derive an approximation algorithm in the mean field framework or an exact algorithm using junction-tree representations [5]. We sketch the algorithm here; a detailed exposition including convergence properties and performance analysis can be found in [2].

3 Coupling and Factoring HMMs

We obtain a joint HMM C from two component HMMs A, B by taking the Cartesian product of their states and transition parameters.

$$\{C\} = \{A\} \times \{B\} \quad (1)$$

$$c_{ij} = a_i \wedge b_j \quad (2)$$

$$P_{c_{ik}|c_{jl}} = P_{a_i|a_j} P_{b_k|b_l} \quad (3)$$

Exploiting the sum-to-one property of probabilities, linear projections will factor the joint HMM back into its components.

$$P_{a_i|a_j} = \sum_l P_{b_l} \sum_k P_{c_{ik}|c_{jl}} \quad (4)$$

$$P_{b_k|b_l} = \sum_j P_{a_j} \sum_i P_{c_{ik}|c_{jl}} \quad (5)$$

where $P_{b_l} = 1/|\{B\}|$ and $P_{a_j} = 1/|\{A\}|$ in the absence of any posterior probabilities.

This projections factors the $(|\{A\}| \cdot |\{B\}|)^2$-dimensional transition table of the joint HMM into $|\{A\}|^2-$ and $|\{B\}|^2$-dimensional transition tables which parameterize two component HMMs. Note that we may just as easily define a projection which factors out the interaction between the component HMMs:

$$P_{a_i|b_l} = \sum_j P_{a_j} \sum_k P_{c_{ik}|c_{jl}} \quad (6)$$

$$P_{b_k|a_j} = \sum_l P_{b_l} \sum_i P_{c_{ik}|c_{jl}} \quad (7)$$

whose inverse is

$$P_{c_{ik}|c_{jl}} = P_{a_i|b_l} P_{b_k|a_j} \quad (8)$$

This is the basis of an algorithm in which a joint HMM is trained via standard HMM methods but constrained to factor consistently along both projections. As each reestimation propels it up through likelihood space, we factor and reconstitute it, thus simultaneously training the component HMMs. Here we formulate the algorithm with factoring after reestimation of the joint HMM; factoring can also be done after forward-backward analysis, so that reestimation can occur in the component HMMs, e.g.:

$$P_{a(t)=i,a(t-1)=j|O} = \frac{\sum_k \sum_l C_{jl,t-1} \cdot P_{ik|jl} \cdot P_{c(t)=ik}(o(t)) \cdot C'_{ik,t}}{P(O)} \quad (9)$$

$$\sim \frac{P_{a(t)=i}(o(t))}{P(O)} \sum_k C'_{ik,t} \sum_l \left(C_{jl,t-1} \cdot P_{ik|jl} \right) \quad (10)$$

$$\sim \frac{P_{a(t)=i}(o(t)) \cdot P_{i|j}}{P(O)} \left(\sum_k C'_{ik,t} \right) \left(\sum_l C_{jl,t-1} \right) \quad (11)$$

where C and C' are the forward and backward variables for the joint HMM. Eqns. 10,11 are approximations that allow substantial speed-ups but sacrifice some information.

In principle, factoring and reconstitution can violate the conditions under which convergence is guaranteed, also, eqns. 3 and 8 may not be consistent; in [2] we develop conditioning steps which restore the convergence property. In practice, the algorithm appears to perform nearly as well without the conditioning steps, and in either case it works more robustly than a single HMM.

Note that we do not take the Cartesian product of the output parameters. They are reestimated directly in the component HMMs using posterior component state probabilities. This has three advantages: (1) $O(2N)$ output parameters are reestimated instead of $O(N^2)$; (2) the statistics are more robust; (3) forward-backward analysis and run-time Viterbi analysis are considerably faster, since the bulk of computation is in computing multivariate Gaussians and this is reduced by $O(N)$. E.g., recognition with a CHMM can be considerably faster than with an HMM with the same number of states.

4 Experiments

T'ai Chi Ch'uan is a Chinese martial art and meditative exercise, consisting of stylized full-body and upper-body gestures. most signals generated by human activity, gestures included, are the result of multiple interacting processes. In gesture, the arms are neither independent nor wholly mutually determined; some form of interactional modeling is appropriate.

Visually, a simple way to decompose upper-body gestures is to treat each arm as a process. Using a self-calibrating stereo blob tracker [1], we obtained 3D

hand tracking data for three T'ai Chi gestures involving arm-motions: the left[1] single whip, the left cobra, and the left brush knee. Figure 4 illustrates the gestures, the blob-tracking, and the feature vectors.

4.1 Details of data collection

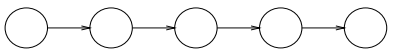
We collected 52 sequences, roughly 17 of each gesture. The extracted feature vector consisted of the 3D (x, y, z) centroid (mean position) of each of the blobs that characterize the hands. All the gestures were performed by the same person, seated in a swivel chair and moving her upper body and hands. Each gesture began with both hands in a rest or neutral position and ended with the hands in a gesture-specific final position or returning to neutral position. The experiments were oriented to a single word recognition task; the extension to continuous gesture trains is the same as with conventional HMMs. The main sources of noise were blob instabilities, variations in the performance of each gesture, and variations in initial body rotation and position from sequence to sequence. The extracted feature vector, being simple (x, y, z) positions, reflects this noise directly.

4.2 Data preprocessing

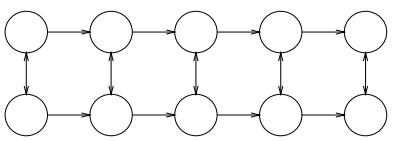
The frame rate of the vision system varied from 15-30 Hz. We resampled the data using time-stamped frames and cubic spline interpolation to produce a 30Hz signal, then low-pass filtered with a 3Hz cutoff. Similar preprocessing is used by Campbell *et al.* [3], who go on to convert the feature vector to head-centered cylindrical coordinates velocities $(dr, d\theta, dz)$ for rotation and shift invariance; we remain with raw 3D (x, y, z) coordinates.

4.3 Results of training different architectures of HMMs

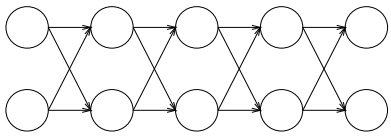
Three HMM architectures, reflecting different independence structures between hidden states, were trained and tested:



1. Conventional HMMs: HMMs ranging in complexity from 2 to 7 states were trained and tested on the data. The best performing models were kept for comparison: 2-state HMMs for cobra and single whip; a 5-state HMM for brush knee.



2. Linked HMMs (a simplification of CHMMs with symmetric noncausal joint probabilities between chains): 2 and 3 per-HMM state LHMMs were similarly evaluated, yielding 2+2-state LHMMs for the cobra and single whip, and a 3+3-state LHMM for the brush knee.



3. Coupled HMMs: Similarly, testing with a small range of CHMMs yielded good models with 3+3-state CHMMs for cobra and brush knee, and a 3+2-state CHMM (a 3-state chain coupled with a 2-state chain) for the single whip gesture. This latter configuration can intuitively be explained because in the single whip gesture one hand moves back and forth while the other hand is mostly stationary, i.e. the complexity of the temporal structure of each hand is different.

Once the apropriate state counts were established, each model was trained 50 times on 5 randomly selected instances of gesture, and the best (highest-likelihood) models were kept for comparison. We did this because HMMs are known to produce models of varying quality, even when trained repeatedly with the same data.

We expected the CHMMs to outperform the LHMM because the coordinate constraints between the arms are asymmetric and temporally mediated. Similarly, we expected both higher-order HMMs to outperform the conventional HMMs because the arms are not perfectly coordinated; any such variation must simply be represented as noise in the single HMM.

4.4 Results of classification test

To compare the performance of the three previously described architectures in a classification task, we computed the maximum likelihood model for each of the models and for each of the 52 sequences, i.e., for each sequence and for each architecture, we selected the gesture whose likelihood was the highest. Figure 2 shows the per-sequence likelihoods for each of the models. The classification accuracies are:

	Single HMMs	Linked HMMs	Coupled HMMs
accuracy	69.2308%	36.5385%*	94.2308%
# params	25+30+180	27+18+36	36+18+36

The bottom row shows the number of degrees of freedom in the largest best-scoring model: state-to-state probabilities + output means + output covariances.

We were surprised by the low accuracy (*) of the LHMM in classifying all the sequences. This is because the LHMM model of the cobra did not not correctly recover the temporal structure; having a very low discrimination power, it claimed all sequences with a high likelihood. In fact, the LHMM performed significantly better than the HMM on the other two gestures.

We note that Campbell et al. [3] were able train conventional HMMs with $(x_l, y_l, z_l, x_r, y_r, z_r)$ feature vectors to classify 18 different T'ai Chi gestures using with accuracies as high as 94%. The HMMs had carefully tuned transition topologies and were each trained on 18 examples of gestures constrained not to have rotational or transitional variation (with variation, rates

[1] Many T'ai Chi forms have mirror-image counterparts.

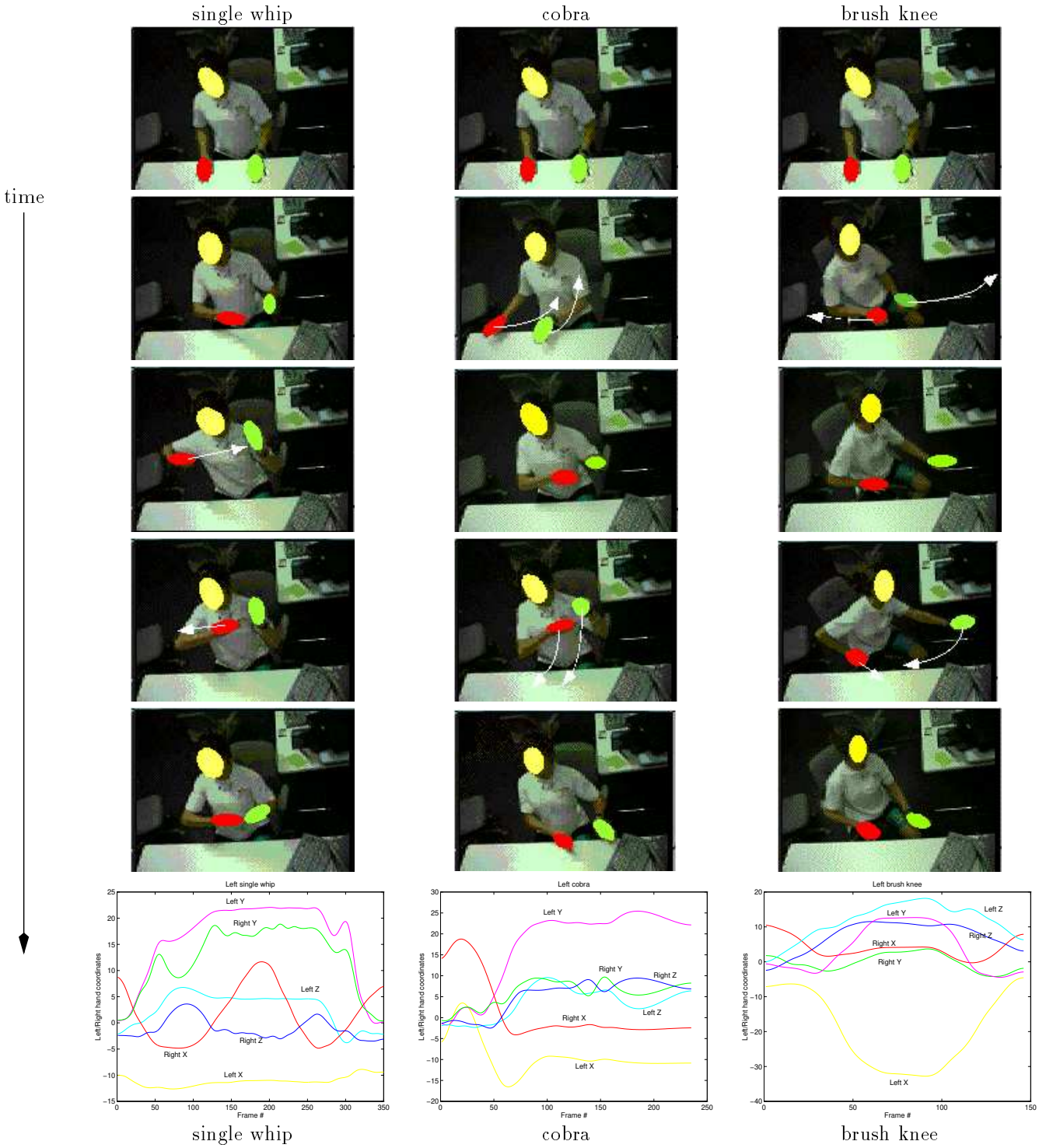


Figure 1: Hand tracking of three gestures: Selected frames overlaid with hand blobs from vision. Graphs in the bottom row show the evolution of the feature vector over time. Sequences may be viewed at `http://www.media.mit.edu/~brand/taichi.html`

fell to 34%). Similar circumstances would certainly raise the rates shown in the table above.

4.5 Sensitivity analysis

HMMs are notoriously sensitive to the random values assigned to parameters at initialization of training. To test the sensitivity of final model likelihoods to initial conditions, we randomly initialized each architecture, trained it on 5 examples of a gesture taken randomly, and tested it on all sequences of that gesture. This was repeated 50 times per gesture and architecture. The likelihoods of the testing sets conditioned on recovered models was computed and mean and variance statistics were computed for each gesture and model. The resulting Gaussian distributions are depicted in figure 3, which shows the probability distribution of the per gesture likelihood for coupled, linked and single HMMs.

As may be expected, conventional HMMs were quite sensitive to the initial values of the parameters. Linked HMMs were generally less sensitive, with a sensitivity (variance) that appears to depend on the structure of the gesture. Finally, on average coupled HMMs were least sensitive to initial conditions and produced the highest likelihood models—even in the case of the single whip, in which one hand is mostly stationary. In sum, CHMMs reliably produce better models.

These results also show why the HMMs performed as well as they did in the classification test. In choosing the best-of-50, we took models from the right (optimal) end of the distribution. Had we picked typical models (the mean), the HMMs would have done quite a bit worse than their already mediocre performance.

5 Conclusion

Hidden Markov models (HMMs) are used widely in perceptual computing as trainable, time-flexible classifiers of signals that originate from processes like speech and gesture. We believe that a conventional HMM is indeed the *wrong* model in that most interesting signals fail to satisfy the very restrictive Markov condition. Speech recognition researchers have grown increasingly frustrated with the performance of HMMs for this very reason, and vision researchers will run into it even faster. We have presented a mathematical framework for coupled hidden Markov models (CHMMs) which offers a way to model multiple interacting processes without running afoul of the Markov condition. CHMMs couple HMMs with temporal, asymmetric conditional probabilities between the chains. To demonstrate their superiority to conventional HMMs, we used a variety of HMM-based architectures to do visual classification of two-handed gestures from T'ai Chi, a martial art. CHMMs produce higher likelihood models with better discriminatory power in fewer epochs *and* these models run faster than comparable HMM in a modified Viterbi algorithm. Finally, these higher-order HMMs are far less sensitive to inital conditions than conventional HMMs, e.g., they are more reliable. We also compare CHMMs with linked HMMs (LHMMs), which have atemporal, symmetric joint probabilities between chains. LHMM architectures have been proposed as a desirable higher-order HMM architecture, but experiments show that CHMMs also significantly outperform LHMMs.

6 Acknowledgements

Special thanks to Andy Wilson for basic Matlab HMM code; Ali Azarbayejani for the self-calibrating stereo hand tracker; Dave Becker for T'ai Chi guidance; and Mike Jordan for illuminating conversations about HMMs.

References

[1] Ali Azarbayejani and Alex Pentland. Real-time self-calibrating stereo person-tracker using 3-D shape estimation from blob features. In *Proceedings, International Conference on Pattern Recognition*, Vienna, August 1996. IEEE.

[2] Matthew Brand. Coupled hidden markov models for modeling interacting processes. November 1996. Submitted to *Neural Computation*.

[3] Lee W. Campbell, David A. Becker, Ali Azarbayjani, Aaron F. Bobick, and Alex Pentland. Invariant features for 3-D gesture recognition. In *Proceedings, International Conference on Automatic Face and Gesture Recognition*, pages 157–162, Killington, VT, 1996. IEEE.

[4] Zoubin Ghahramani and Michael I. Jordan. Factorial hidden Markov models. In David S. Touretzky, Michael C. Mozer, and M.E. Hasselmo, editors, *Advances in Neural Information Processing Systems*, volume 8, Cambridge, MA, 1996. MIT Press.

[5] F. V. Jensen, S. L. Lauritzen, and K. G. Olesen. Bayesian updating in recursive graphical models by local computations. *Computational Statistical Quarterly*, 4:269–282, 190.

[6] Michael I. Jordan, Zoubin Ghahramani, and Lawrence K. Saul. Hidden Markov decision trees. In David S. Touretzky, Michael C. Mozer, and M.E. Hasselmo, editors, *Advances in Neural Information Processing Systems*, volume 8, Cambridge, MA, 1996. MIT Press.

[7] Lawrence K. Saul and Michael I. Jordan. Boltzmann chains and hidden Markov models. In Gary Tesauro, David S. Touretzky, and T.K. Leen, editors, *Advances in Neural Information Processing Systems*, volume 7, Cambridge, MA, 1995. MIT Press.

[8] Padhraic Smyth, David Heckerman, and Michael Jordan. Probabilistic independence networks for hidden Markov probability models. AI memo 1565, MIT, Cambridge, MA, February 1996.

[9] C. Williams and G. E. Hinton. Mean field networks that learn to discriminate temporally distorted strings. In *Proceedings, connectionist models summer school*, pages 18–22, San Mateo, CA, 1990. Morgan Kaufmann.

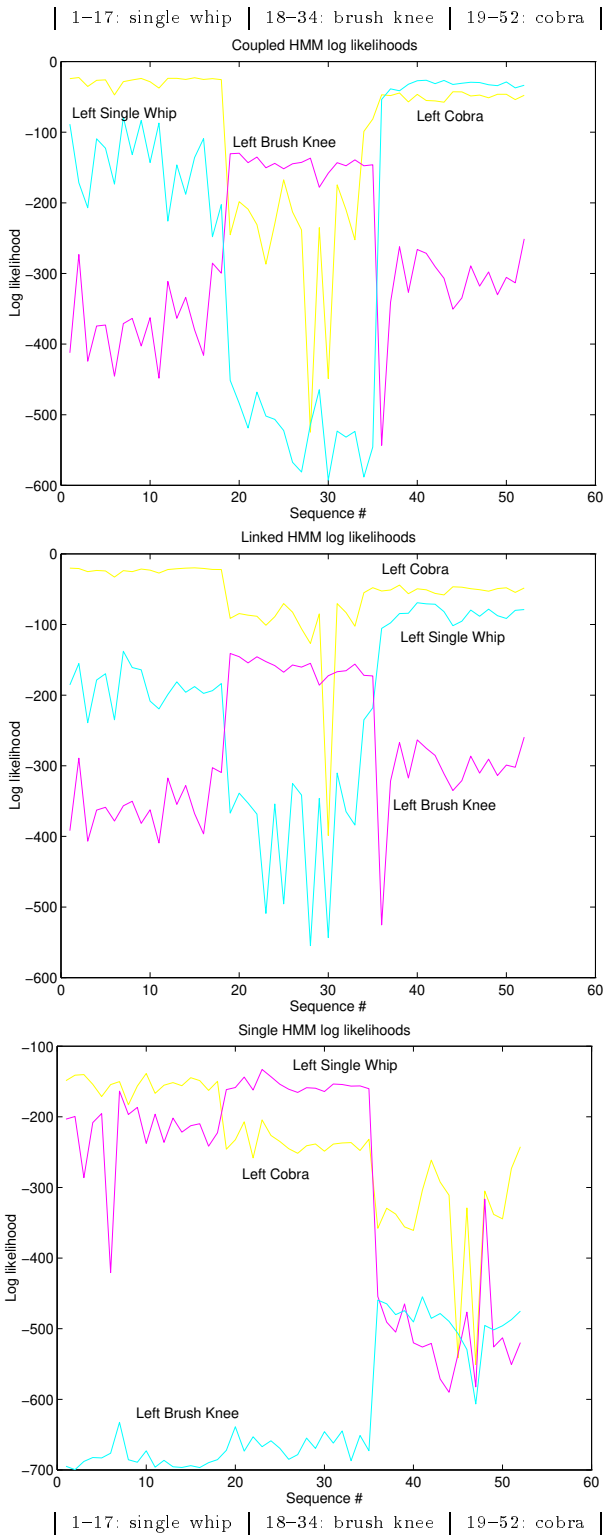


Figure 2: Classification by the CHMM, LHMM, and HMM, showing per-sequence normalized log likelihood. Only the CHMM attains the right discrimination structure

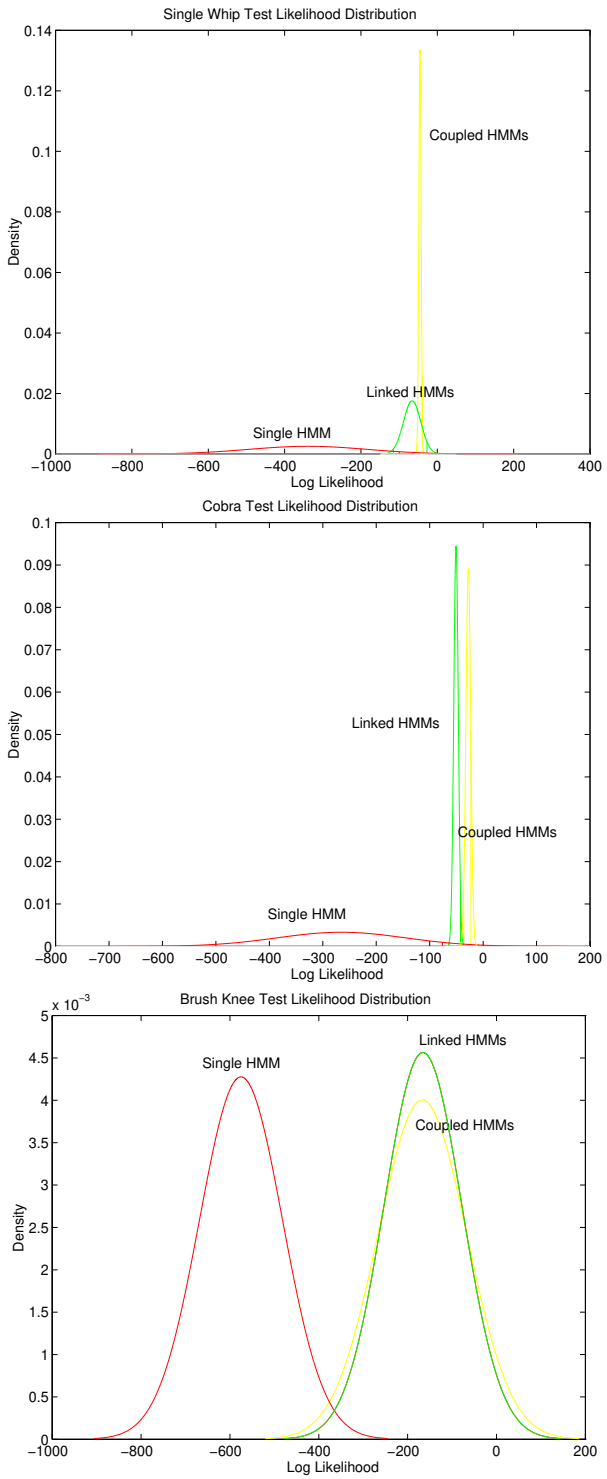


Figure 3: Likelihood probability distribution for each HMM type, learning single whip, cobra, and brush knee gestures, respectively. The CHMM produces the most likely models with a high consistency, indicated by the rightmost distributions.