M.I.T Media Laboratory Perceptual Computing Section Technical Report No. 379
Appears in Second International Workshop on Face and Gesture Recognition, Killington VT Oct. 1996

Invariant features for 3-D gesture recognition

Lee W. Campbell   David A. Becker   Ali Azarbayejani   Aaron F. Bobick   Alex Pentland
Room E15-383, The Media Laboratory
Massachusetts Institute of Technology
20 Ames St., Cambridge, MA 02139
{elwin, dbecker, ali, bobick, sandy}@media.mit.edu

Abstract

Ten different feature vectors are tested in a gesture recognition task which utilizes 3D data gathered in real-time from stereo video cameras, and HMMs for learning and recognition of gestures. Results indicate velocity features are superior to positional features, and partial rotational invariance is sufficient for good performance.

1 Introduction

This paper reports the results of our investigation into invariant features for gesture recognition. Video is an excellent input modality for gesture recognition because it nonintrusive; however until now most video input gesture recognizers used view-based features. Real-time stereo tracking enables computation of 3D world coordinates, from which we can extract view-invariant features.

In principle, many possible sets of features contain enough information to reconstruct the original gesture, and hence should be adequate for recognition, but the optimal feature choice is not obvious. Some features interact poorly with noise in the system, while others do not take full advantage of the powerful recognition technology available today, e.g. the ability to compensate for timing variation.

To investigate this issue we compared recognition performance on ten different feature vectors derived from a single set of 18 T'ai Chi gestures listed in Table 1. We construct a training set of 54 six-gesture "sentences;" and test on another 54 sentence set. A "sentence" is a set of six gestures performed in sequence which captures the co-articulation exhibited by T'ai Chi. Figure 1 shows a sample sentence.

The choice of T'ai Chi as a gesture set was motivated by its appropriateness for use in an application currently being developed by Becker and Pentland called *Staying Alive* [2]. *Staying Alive* is a virtual reality imagery and relaxation tool being built for cancer patients. Because numerous studies have shown that inducing the "relaxation response" in a person enhances the effectiveness of the immune system [7], one of the goals of the *Staying Alive* system is to help the user relax. For that reason, the system was developed such that the user interacts with it through the practice of T'ai Chi.

18 T'ai Chi gestures	
opening form	grab birds tail
left wave clouds	right wave clouds
left cobra	right cobra
left brush knee	right brush knee
left roll arms	right roll arms
left single whip	right single whip
left high horse	right high horse
left defl parry	right defl parry
strike ears fists	apparent closeup

Table 1. Names of the 18 T'ai Chi gestures used in our experiments.

T'ai Chi [12] is among the most popular forms of martial arts in China and the world. Though like all martial arts, this exercise includes movements for attack and defense, the Yang style, from which our set of gestures was chosen, is meant to help its practitioner achieve relaxation and wholeness. Yang style T'ai Chi movements are slow, smooth, and gentle.

To capture the T'ai Chi movements in real-time we utilized a wide baseline stereo vision system developed by Azarbayejani and Pentland [1] which produces head and hand positions to an accuracy of approx. 2cm at up to 30 frames/sec. The several feature vectors derived from this data were input to Hidden Markov Model (HMM) recognizers, for which we used the HTK2.0 HMM toolkit from Entropic Research [13].

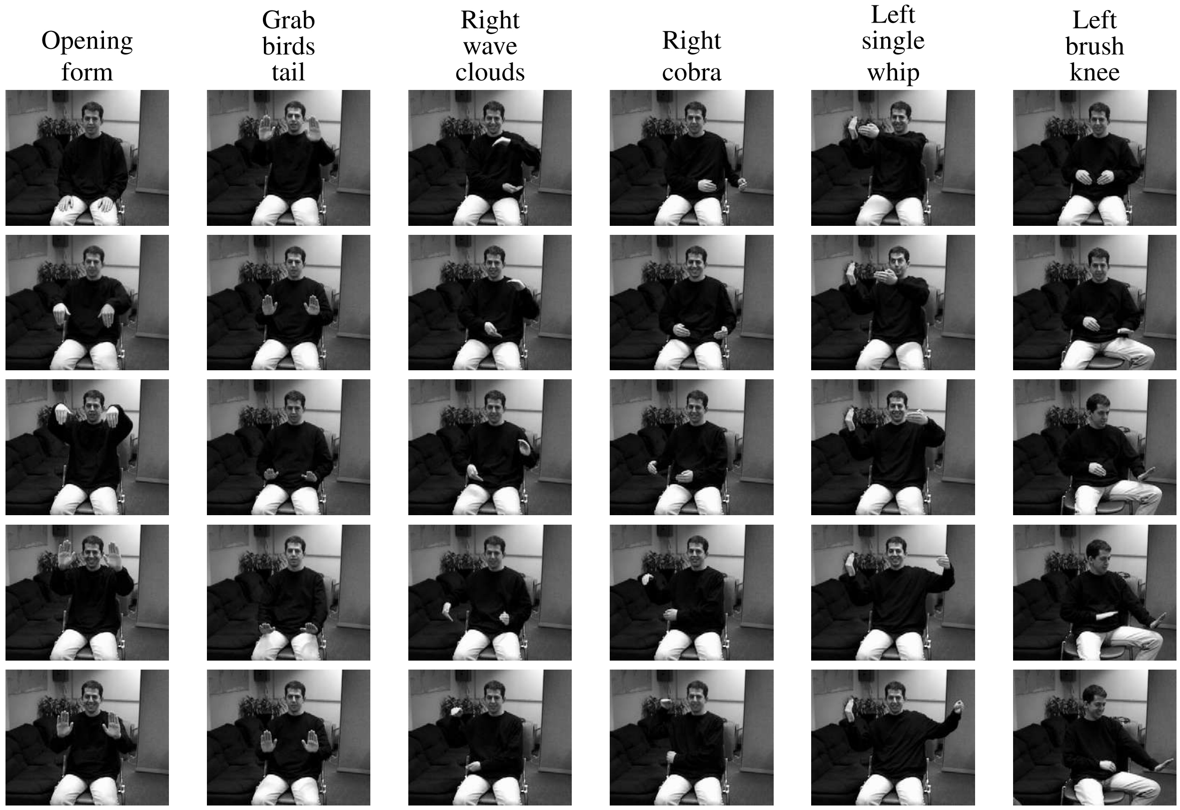

Figure 1. A sample sentence of six T'ai Chi moves.

2 Related Work

HMMs have been applied to speech recognition with great success [4], and have begun to be applied in the vision community where time variation is significant; e.g. gesture and action recognition. Many early approaches used view-based signals as feature vectors. The earliest such application was by Yamato et al. [11] to recognize tennis swings, using as features a 25×25 pixel binarized camera image. Schlenzig et al. [6] used HMMs to recognize gestures in sequences, using a rotation invariant representation of a binary image, processed through a neural net. Wilson and Bobick [10] incorporated multiple representations in an HMM framework, using eigen image weights as features.

In a non-HMM approach Cui and Weng [3] used maximally discriminating (MDF) image weights as features, and a vector quantization of a low dimensional MDF space as a distance metric.

Stereo tracking of hands has been reported by Rehg and Kanade [5] who recover a 26 DOF description of joint angles in the hand; however they have not yet applied this to recognition.

Our approach of using functions of coordinates as feature vectors was inspired by the work of Starner and Pentland [8] [9] who used HMMs to recognize 40 American sign language gestures in real-time 2D video imagery, using the 2D image coordinates and orientation of the hands as feature vectors. We felt their basic approach would adapt to T'ai Chi, and that a 3D measurement system would allow a more robust approach to the problem of user translation and rotation.

3 HMMs for gesture recognition

First order Markov processes describe systems which jump from state to state over time; change from the present state to a new state is stochastic and governed by probabilities which are only a function of the present state (the Markovian property). Thus an n-state first order Markov process is described by the states $Q = q_1, \ldots, q_n$, an n-vector π where π is the probability of starting in state i, and an $n \times n$ matrix $\mathbf{A}$ of state transition probabilities where a_{ij} is the probability of jumping to state q_j given the present state is q_i.

Hidden Markov Models [4] describe a doubly stochastic process, in which the present state of the system is not directly observable; instead the system outputs a symbol at each time-step and the symbol is stochastically chosen from

Feature vector	Advantages	Disadvantages
(x, y, z) in world coords with or without head	contain the most "context;" allow the automatic dynamic time warping of the HMM to compensate for speed variations	Sensitive to both shift and rotation
(dx, dy, dz)	shift invariant	Sensitive to rotations; no DTW
(r, θ, z) hands rel. to head	shift invariant; allows DTW	inhomogeneous; mixes head with hand
$(dr, d\theta, dz)$	shift and horizontal rotation invariant	inhomogeneous; mixes head with hand; no DTW
$(dr, rd\theta, dz)$	shift and horizontal rotation invariant	mixes head with hand; no DTW
$(ds, \log(\rho), dz)$ speed, curvature, dz	shift and horizontal rotation invariant, homogeneous	least context; requires second deriv; no DTW

Table 2. Comparison of the different feature vectors.

among a discrete set. Continuous HMMs are a generalization in which the output symbol is a continuous random variable or vector. If there are m possible symbols the HMM is described by $(Q, \pi, \mathbf{A}, \mathbf{B}, V)$ where V is the set of output symbols $v_1, \ldots, v_n$, and $\mathbf{B}$ is an $n \times m$ matrix of output symbol probabilities such that b_{jk} is the probability of outputting symbol v_k given the present state is q_j. For a continuous HMM, $\mathbf{B}$ contains parameters of a probability density function (typically a mixture of Gaussians) governing possible output values.

Forward chaining HMMs are a special case of HMMs which have zero probability of returning to an earlier state; they have no cycles in them.

We model each of our gestures as a different forward chaining continuous HMM: the observable symbols are some function of the coordinate estimates of the vision system; the states and transition probabilities are determined automatically by the Baum–Welch algorithm during the HMM training process.

Recognition works like this: given a sequence of 3D coordinates estimated by the vision system, which HMM most likely emitted it? This is determined (using the Viterbi algorithm) by calculating, for each HMM, the log-probability that the HMM gave rise to that sequence of symbols, and picking the maximum (or rejecting all of them if all are below a threshold). In this work, each sequence of data is a "sentence" which contains a string of six T'ai Chi gestures, and the Viterbi algorithm finds the corresponding most probable six HMMs. (The ordering of gestures within a string is random.)

Forward chaining continuous HMMs are appropriate to 3D gesture recognition because gesture performance and measurement is a stochastic process: there is an underlying deterministic intent to make the gesture in a particular form; added to that is actuator noise (muscles are imprecise) and sensor noise (the stereo vision system has estimation error). Thus, some function $\mathbf{f}(x, y, z)$ of the sequence of coordinates of the hands, as sampled over time and measured by the vision system can be considered to be the observable symbols of a Hidden Markov process.

4 Feature Vector Selection

What is a good choice for the function $\mathbf{f}(x, y, z)$; i.e. what is a good feature for the HMM?

In our case, we receive raw (x, y, z) data from the vision system, from which we can deduce positions and velocities with respect to various coordinate reference systems. For our investigations we chose to consider the raw position (x, y, z), the Cartesian velocity (dx, dy, dz), the polar velocity with angular velocity term $(dr, d\theta, dz)$, the polar velocity with tangential velocity term $(dr, rd\theta, dz)$, and two sets with instantaneous speed ds and local curvature ρ: $(ds, \log(\rho), dz)$, $(ds, \log(\rho ds), dz)$. The raw position and Cartesian velocities are relative to a world-centered coordinate system fixed to the stereo vision system. Speed and curvature are local properties of the paths traced out by the hands and are coordinate system independent. The polar velocities are computed relative to a body-centered coordinate system by treating the head as the origin, measuring rotation around a vertical axis.

Since the head position is used in this way, it can not be a separate feature in the polar representations, but it is a separate feature in the Cartesian feature sets. We constructed ten feature vectors for our experiments, tabulated in Table 2. They consist of either positions, Cartesian velocities, polar velocities, or curvature; and either include or omit the head components.

To motivate these choices of feature vectors, consider a person sitting at a fixed location and orientation with respect to the vision system, and performing a set of gestures, and assume the HMM trains on and recognizes (x, y, z) coordinates of hands. Then the states of the continuous HMM will correspond to density functions in space that the hands pass through. If the hands move faster or more slowly during a test gesture, the Viterbi algorithm will compensate for this; effectively performing dynamic time warping. However, if the gesture is made "smaller," it will only pass through the fringes of the density functions and will receive a lower log-probability score. Also, this feature is not invariant to translation or rotation.

Now consider the velocity of the hands (dx, dy, dz) as a training and testing feature: states of the HMM will correspond to velocity vectors at different times during the gesture. This feature is shift invariant but not rotation invariant. It also gives up some of the dynamic time warping ability of the Viterbi algorithm: if, during a test gesture, the hands move at lower or higher speed, the symbols will fall more to the fringes of a state. However, if the hand moves at the same speed for a shorter period, time warping can occur.

Ideally, a set of features should be shift and rotation invariant; isotropically noisy in the measurement space; able to make use of Viterbi time warping; and contain as much "context" as possible (i.e. values unique to each stage of a gesture). Unfortunately, these ideals cannot all be fulfilled at once; they trade off against each other. For example: imagine moving your hand in a perfect circle. Curvature and speed (ρ, ds) will be completely invariant to rotations and translations but will be constant and thus exhibit no "context." At the other extreme, (x, y, z) coordinates can easily distinguish the top, bottom, and sides of the circle, but they require it to be performed each time at the same location. In between these extremes, (dx, dy, dz) and $(dr, d\theta, dz)$ trade off some invariance for some context. An additional problem with derivatives is that they amplify high frequency noise.

This problem is present in the curvature features because curvature is a function of the second derivative and thus inherently more noisy than velocity features. A few tests using acceleration as a feature showed that second derivative noise was a hindrance to recognition. Another problem with curvature is it does not fall in a Gaussian distribution which the HMM expects; particularly when the hand is almost stationary it tends to generate very high curvatures and very low velocities. Taking $\log(\rho)$ or $\log(\rho ds)$ shapes the distribution to be more Gaussian and improves HMM recognition rates.

Including dz as a feature means that at best we can achieve invariance to rotations in the horizontal plane, but not to out of plane rotations. However, people tend to orient themselves with respect to gravity, so out of plane rotations will be unusual.

5 Description of Experiments

The experiments were designed to test different feature vectors computed from the (x, y, z) coordinates output by the vision system. We began with 108 sentences, (each formed of six randomly permuted gestures; no two sentences are alike). An experimental run consisted of training the 18 HMMs (one for each of 18 gestures) with a training set of 54 prerecorded sentences; and then attempting recognition of the other 54 sentences. The feature vector was then scored on its recognition rate.

5.1 Data Preprocessing

The vision system returns time-stamped (x, y, z) for the head and left and right hands, gathered at a rate of 15–30Hz. The recovered trajectories have up to 2cm RMS error[1] and are not sampled uniformly in time. Hence we resampled it in time to a 30Hz sampling rate (using cubic splines) and low-pass filtered with a 3Hz cutoff before computing feature vectors. The parameters r and θ were calculated by transforming coordinates so the head was at the origin and the x-axis was in front of the performer (when at the original position); derivatives were computed with short FIR filters.

5.2 Data Collection

All sentences were performed by the same individual. Due to space constraints the performer was seated in a swivel chair and performed the movements with his upper body and hands. Each sentence begins with hands in a rest position; the six gestures are performed in a flowing manner characteristic of T'ai Chi, and the hands are returned to the rest position (there is actually a 19th HMM to recognize the rest position; it is almost 100% accurate and was not counted in the recognition rate.) The sentences were permuted in such a way that each group of three forms a complete run through the 18 gestures; thus the training and testing sets each contain 18 examples of each gesture. 72 of the sentences (the "original") were performed with the performer sitting in the same position and orientation (as best we could manage; they were performed on two different nights). An additional 18 (the "shifted set") were performed with a translational shift of 14 inches from from the original position, and the final 18 (the "rotated set") were performed with both an 8in shift and 45° rotation (Due to space limitations, the performer rotated about 20° relative to the cameras and the remaining 25° were added synthetically).

All of the 54 sentences in the training set were selected randomly from the original set. The testing set has three groups: an original position group; a shifted group, and a rotated group. This allows us to measure performance separately on shifted and rotated data.

5.3 HMM Topology

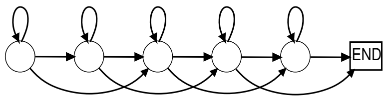

Figure 2. The five state HMM used for recognition.

The HMMs for the 18 gestures for each of the feature vectors reported in Table 3 all had the same topology: five state, forward chaining with jumps possible to the same state or each of the next two states. The symbol was a continuous valued vector of 6 to 9 elements with diagonal covariance. No duration modeling was used. A rough grammar constrained recognition to match the model of "rest, six gestures, rest."[1] As can be seen from Figure 2, the HMM has "skip" transitions; during training the HMMs for some gestures may learn to skip one or more states entirely; thus this topology can model either a 3, 4, or 5 state gesture.

6 Results

As can be seen from Table 3, the feature vector $(dr, d\theta, dz)$ had the best overall recognition rates. This shows the strength of rotational invariance: yet, this feature set tended to confuse the 'single whip' with the 'brush knee' gesture, while (dx, dy, dz) tended to confuse 'cobra' and 'cloud' gestures. These confusions are understandable because portions the gestures are very similar.

The most surprising result is the performance of $[(dx, dy, dz)$ no head] on rotated data. Despite 45° of rotation it achieved 77% accuracy; even though it is not designed to be rotation invariant. It is interesting that combining (dx, dy, dz) and $(dr, d\theta, dz)$ feature sets did slightly worse than $(dr, d\theta, dz)$ alone. This is probably due to lack of enough training data to estimate covariances for the larger feature set. With too little data, the system will over-train on the training data.

A number of additional experiments were performed to determine why curvature did so poorly as a feature; we found that three factors combined to lower recognition rates. One factor is derivative noise, which we confirmed by trying acceleration as a feature. The second factor is that speed and curvature are both always positive – this is analogous to taking the absolute value of a feature – and it results in less context and more opportunities for confusion. Although torsion might disambiguate curvature, it is a function of third derivatives and suffers from worse noise. The third factor is that curvature did not have a Gaussian distribution; we were able to compensate for this somewhat by taking $\log(\rho ds)$ which shaped it to be more Gaussian.

All feature sets perform worse on shifted and/or rotated data. A theory of why this is so is that the measurements of the stereo vision system may not be homogeneous in space; hence measurement noise would depend on the position of the performer. In practice this can be compensated for by collecting training data with the performer at several different positions and rotations to sample the noise at various locations.

In order to gauge the statistical significance of the reported error rates, we used the $[(x, y, z)$ with head] feature set and trained on several random groups of 54 sentences from the 'original position' set; each time testing on the remaining 18 sentences. The recognition rate from these tests fluctuated from 87% to 94% with an rms error of 2.6%; this scales to 1.5% rms in the 'Total' column, so a reported 3% difference in that column is significant. In practice we found that smaller differences are significant: we found that a training set tends to increase or decrease recognition rates for all feature sets; the above analysis assumed uncorrelated results for feature sets.

In earlier tests we experimented with three state and four state HMMs and found that five states gave the best recognition results. Though some gestures have five states, others used skip transitions to represent the gesture as having four states. Sometimes one of the states models the duration of the gesture rather than its complexity: long dwell times in a state can be represented with two similar states in sequence. Using the duration modeling features of the HMM might reduce the number of states required for good recognition.

Our data shows that "polar" velocity features are best for recognition of these T'ai Chi gestures, indicating there is still a great deal of "context" present in the velocity features. With more training data, we would no longer need to constrain the covariance matrix to be diagonal, yielding better recognition rates.

7 Conclusion

We have performed gesture recognition on a set of 18 T'ai Chi gestures, and compared the performance of ten different feature vectors based on 3D hand and head tracking data. Our basic results confirm our expectations that features designed to be invariant to shift and/or rotation perform better in the presence of shifted and rotated input. Cartesian velocity features perform better in the presence of translational shifts; polar velocity features perform better in the presence of rotation.

[1] Automatic segmentation is done by the HTK system when given a grammar and a crude starting guess; here short duration rests and equal duration gestures.

Feature	Notes	Orig.	Shift	Rot.	Overall
(x, y, z) no head	Sensitive to shift and rotation	94%	5%	3%	34%
(x, y, z) with head		92%	16%	7%	38%
(dx, dy, dz) no head		98%	92%	77%	89%
(dx, dy, dz) with head	head features reduce recognition	94%	94%	74%	87%
(r, θ, z) no head		94%	92%	19%	68%
$(dr, d\theta, dz)$ no head	errors with single whip	96%	96%	93%	95%
$(dr, rd\theta, dz)$ no head		95%	96%	91%	93%
$(dr, d\theta, dx, dy, dz)$ no head	redundant features	94%	93%	89%	92%
$(ds, \log(\rho ds), dz)$ no head		74%	69%	72%	72%
$(ds, \log(\rho), dz)$ no head		75%	66%	69%	70%

Table 3. Recognition rates of the different feature vectors. The percentages reported are computed as: Accuracy $=$ ((# gestures correctly recognized $-$ # deleted) $/$ ground truth total # gestures) $\times$ 100%

Furthermore, our results highlight several important issues associated with the general problem of choosing features for gesture recognition systems. First is that choosing the right set of features can be crucial to the performance. There is a design choice that goes into every implementation of gesture recognition systems and since this choice can greatly impact performance, it is critical that we understand the effect of feature choice on performance.

Second, redundant features can help reject noise, resulting in lower error rates, yet we have found that adding features with large noise levels can actually degrade performance given a finite training set. Thus, the formula for choosing features is not a simple one and both the nature and quality of features must be taken into account as well as the available training set.

Design of recognition systems, then, remains an art for which careful consideration of feature design can lead to significantly improved results. This work has contributed to the understanding of this phenomenon and we hope that future work can be geared towards automatic methods of deducing optimal feature sets for particular problems.

References

[1] A. Azarbayejani and A. Pentland. Real-time self-calibrating stereo person tracking using 3-D shape estimation from blob features. In *Proceedings of 13th ICPR*, Vienna, Austria, August 1996. IEEE Computer Society Press.

[2] D. A. Becker and A. Pentland. Staying alive: A virtual reality visualization tool for cancer patients. In *Proc. of the AAAI Workshop on AI, Alife, and Entertainment*, Portland, 1996. To appear in August, 1996.

[3] Y. Cui and J. Weng. Learning-based hand sign recognition. In *Proc. of the Intl. Workshop on Automatic Face- and Gesture-Recognition*, Zurich, 1995.

[4] L. R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proc. of the IEEE*, 77(2):257–285, February 1989.

[5] J. M. Rehg and T. Kanade. Visual tracking of high DOF articulated structures: an application to human hand tracking. In *Proc. ECCV*, volume 2, pages 35–46, Stockholm, 1994.

[6] J. Schlenzig, E. Hunter, and R. Jain. Recursive identification of gesture inputs using hidden markov models. *Proc. 2nd Ann. Conf. on Applications of Computer Vision*, pages 187–194, December 1994.

[7] S. E. Sims. Relaxation training as a technique for helping patients cope with the experience of cancer: a selective review of the literature. *Journal of Advanced Nursing*, Sep 1987.

[8] T. E. Starner. Visual recognition of american sign language using hidden markov models. Master's thesis, Massachusetts Institute of Technology, February 1995.

[9] T. E. Starner and A. Pentland. Real-time american sign language recognition from video using hidden markov models. Technical Report 375, MIT Media Lab Vision and Modeling Group, 20 Ames St, Cambridge MA, 1995. (http://www-white.media.mit.edu/vismod/publications/).

[10] A. D. Wilson and A. F. Bobick. Learning visual behavior for gesture analysis. In *Proc. IEEE Int'l. Symp. on Comp. Vis.*, Coral Gables, Florida, November 1995.

[11] J. Yamato, J. Ohya, and K. Ishii. Recognizing human action in time-sequetial images using hidden markov model. In *Proc. 1992 IEEE Conf. on Computer Vision and Pattern Rec.*, pages 379–385. IEEE Press, 1992.

[12] S. Yen-Ling. *T'ai Chi Ch'uan: The Basic Exercises*. Sugawara Martial Arts, 1990.

[13] S. Young. *HTK: Hidden Markov Model Toolkit V1.5*. Cambridge Univ. Eng. Dept. Speech Group and Entropic Research Lab. Inc., Washington DC, February 1989.