Value Directed Learning of Gestures and Facial Displays

Jesse Hoey and James J. Little
Department of Computer Science
University of British Columbia
Vancouver, BC, CANADA

Abstract

This paper presents a method for learning decision theoretic models of facial expressions and gestures from video data. We consider that the meaning of a facial display or gesture to an observer is contained in its relationship to context, actions and outcomes. An agent wishing to capitalize on these relationships must distinguish facial displays and gestures according to how they help the agent to maximize utility. This paper demonstrates how an agent can learn relationships between unlabeled observations of a person's face and gestures, the context, and its own actions and utility function. The agent needs no prior knowledge about the number or the structure of the gestures and facial displays that are valuable to distinguish. The agent discovers classes of human non-verbal behaviors, as well as which are important for choosing actions that optimize over the utility of possible outcomes. This value-directed model learning allows an agent to focus resources on recognizing only those behaviors which are useful to distinguish. We show results in a simple gestural robotic control problem and in a simple card game played by two human players.

1. Introduction

Human non-verbal behaviors, including facial displays and hand gestures, occur due to many factors, including communication, emotion, speech and physiology [17, 13]. These behaviors are seldom performed or interpreted by humans in isolation, but are usually embedded in a rich context of objects, events, human actions, and human utilities. Further, human non-verbal behaviors are often used purposefully [7]. For example, facial displays and hand gestures are used in conversation for dialogue control [3], such as turn-taking. This paper describes a method for the automatic learning and analysis of purposeful, context-dependent, human non-verbal behavior by a human-interactive agent. The agent can use the method to learn classes of human displays[1] and the relationship between the displays and the context, the agent's actions, and the agent's utility function. No prior knowledge about the structure of displays or the number of displays is necessary as inputs. The agent learns which displays (and how many) are conducive to achieving value in the context. The model we propose can be used to learn the meaning of any non-verbal displays: it can be used equally well for the modeling of faces and gestures, or for both at once.

Most systems for human motion analysis attempt to *recognize* either purported characteristic behaviors, or the predefined atomic units which make up such behaviors. The result is a machine whose inputs are labeled video sequences or static images, and whose outputs are characteristic behavior labels. For example, much research has been devoted to the recognition of emotional expressions in the human face [2]. Similarly, gesture recognition has focussed on learning from labeled examples of significant gestures [18]. Systems have also been built for the automatic detection of the basic units of muscular activity in the human face (action units or AUs) [21].

However, these systems all rely on some method for expert labeling of a training data set. Not only is this process time consuming, but it also unnecessarily constrains the resulting models to the types of gestures believed to be important by the experts. Further, such research simply attempts to recognize characteristic expressions, as if this by itself was the goal. The systems are not easily adaptable, and do not generalize well.

The model we propose is a partially observable Markov decision process, or POMDP [11], which combines the recognition of displays with their interpretation and use in a utility-maximization framework. Video observations are integrated into the POMDP using a dynamic Bayesian network, which creates spatial and temporal abstractions amenable to decision making at the high level. The parameters of the model are learned from training data using an *a posteriori* constrained optimization technique, such that an agent can learn to act based on the displays of a human through observation. We do not train classifiers for individual displays, and then combine them in the model. Rather, the learning process *discovers* clusters of non-verbal behaviors and their relationship to the context automatically. This paper presents work that builds upon our previous explorations into the modeling of facial displays with

[1] We use the term *display* to refer to both facial and gestural displays

POMDPs [8]. The contributions of this paper are a demonstration of the same model applied to a simple gesture recognition task, and the inclusion of value-directed structure learning for determining the number of important clusters in a training corpus. The idea is that a perceptual agent need only make those distinctions which are necessary for predicting future reward. While this idea has been explored in the machine learning literature [12], this paper shows how it can be used in a realistic domain, involving large continuous output spaces over video sequences.

Other researchers have looked at *unsupervised* learning of non-verbal gesture categories [10, 22], but have yet to complete the picture with the addition of utilities and actions. POMDPs have been used for control of robots [20], and spoken dialogue management [16], among other applications. Darrell and Pentland used POMDPs for control of an active camera [4]. Their POMDP model was trained to foveate regions which contained information of interest, such as the hands during gesturing. However, their work is focussed on computing policies in a reinforcement learning setting. They do not learn the number of behaviors, and they separate visual recognition from decision making.

Section 2 describes our POMDP model for display understanding, including the observation function (Section 2), the methods for learning the parameters of the POMDP and for solving the POMDP (Section 2.2 and 2.3), and the value-directed structure learning technique (Section 2.4). Section 3 presents our results on data of two interactions.

2 Gesture and Facial Display Understanding using POMDPs

A POMDP is a probabilistic temporal model of an agent interacting with the environment [11], shown as a Bayesian network in Figure 1(a). A POMDP is similar to a hidden Markov model in that it describes observations as arising from hidden states, which are linked through a Markovian chain. However, the POMDP adds actions and rewards, allowing for decision theoretic planning.

A POMDP is a tuple $\langle S, A, T, R, O, B \rangle$, where S is a finite set of (possible unobservable) states of the environment, A is a finite set of agent actions, $T : S \times A \rightarrow S$ is a transition function which describes the effects of agent actions upon the world states, $R : S \times A \rightarrow \mathcal{R}$ is a reward function which gives the expected reward for taking action A in state S, O is a set of observations, and $B : S \times A \rightarrow O$ is an observation function which gives the probability of observations in each state-action pair. A POMDP model allows an agent to predict the effects of its actions upon his environment, and to choose actions based on its predictions.

To use POMDPs for display understanding, we must admit that the environment may include other intelligent

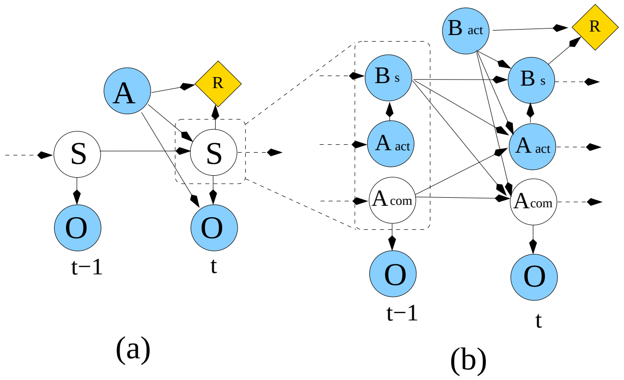

Figure 1: (a) Two time slices of general POMDP. (b) Two time slices of factored POMDP for display understanding. The state, S, has been factored, and conditional independencies have been introduced.

agents, which puts us in the realm of multi-agent games. However, we can take a decision analytic approach to games, in which each agent decides upon a strategy based on his subjective probability distribution over the strategies employed by other players. Essentially, a decision analytic agent simply includes the strategies and internal states of all other agents as part of his internal state. In the following, we will refer to the two agents we are modeling as “Bob” and “Ann”, and we will discuss the model from Bob’s perspective. Figure 1(b) shows a factored POMDP model for display understanding in simple interactions [2]. The state of Bob’s POMDP is factored into Bob’s private internal state, Bs, Ann’s action, $Aact$, and Ann’s display, $Acom$, such that $S_t = \{Bs_t, Aact_t, Acom_t\}$. While Bs and $Aact$ are observable, $Acom$ is not, and must be inferred from video sequence observations, $\mathbf{O}$. In general, both $Aact$ and Bs may also be unobservable. However, we wish to focus on learning models of displays, $Acom$, and so we will use games in which $Aact$ and Bs are fully observable.

The transition function is factored into four terms. The first involves only fully observable variables, and is the conditional probability of the state at time t under the effect of both player’s actions: $\Theta_S = P(Bs_t | Aact_t, Bact, Bs_{t-1})$. The second is over Ann’s actions given Bob’s action, the previous state, and her previous display: $\Theta_A = P(Aact_t | Bact, Acom_{t-1}, Bs_{t-1})$. The third describes Bob’s expectation about Ann’s displays given his action, the previous state and her previous display: $\Theta_D = P(Acom_t | Bact, Bs_{t-1}, Acom_{t-1})$. The fourth describes what Bob expects to see in the video of Ann’s face, $\mathbf{O}$, given his high-level descriptor, $Acom$: $\Theta_O = P(\mathbf{O}_t | Acom_t)$. For example, for some state of $Acom$, this function may assign high likelihood to sequences in which Ann smiles.

[2] Factored representations write the state space as the cross product of a set of multinomial, discrete variables, and allow conditional independencies in the transition function, T, to be exploited by solution techniques.

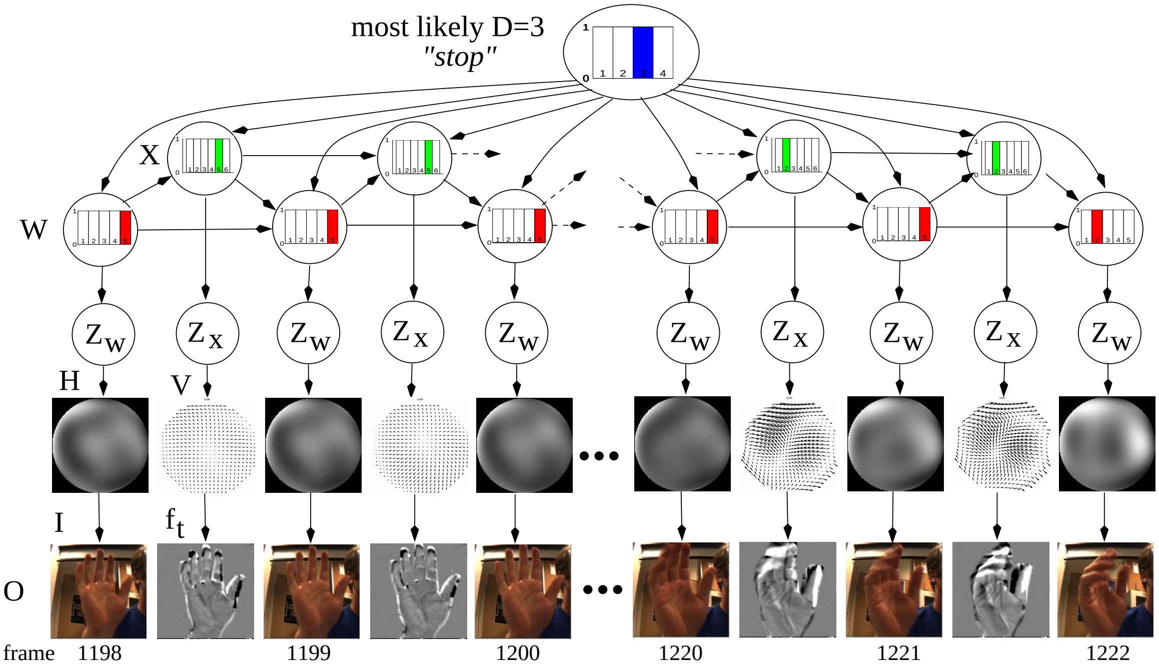

Figure 2: POMDP observation model explaining sequence of “stop” gesture. See text for details.

This value of $Acom$ is only assigned meaning through its relationship with the context and Bob’s action and utility function. We can, however, look at this observation function, and interpret it as an $Acom$ = ’smile’ state. For clarity in the following, we rename the variables as $C_t = \{Bact_t, Bs_{t-1}\}$, $A_t = Aact_t$, and $D_t = Acom_t$. The likelihood of a sequence of data, $\underset{1,T}{\{\mathbf{OCA}\}} = \{O_1 \ldots O_T, C_1 \ldots C_T, A_1 \ldots A_T\}$, is

$$P(\underset{1,T}{\{\mathbf{OCA}\}}|\Theta) = \sum_k \Theta_{O,k} \sum_l \Theta_A \Theta_D P(D_{T-1,l}, \underset{1,T-1}{\{\mathbf{OCA}\}}|\Theta)$$

where $\Theta_{O,k}$ is the observation probability given $D_{T,k}$, the k^{th} value of the mixture state, D, at time T. The observations, $\mathbf{O}$, are temporal sequences of finite extent. We assume that the boundaries of these temporal sequences will be given by the changes in the fully observable context state, C and A. There are many approaches to this problem, ranging from the complete Bayesian solution in which the temporal segmentation is parametrised and integrated out [6], to specification of a fixed segmentation time [14].

2.1 Observation model

We now must compute $P(\mathbf{O}|D)$, where $\mathbf{O}$ is a sequence of video frames, and D is the display descriptor, $Acom$. We have developed a generative model for constructing temporally and spatially abstract descriptions of sequences of displays from video [9, 8]. We give a brief outline of the method here. Figure 2 shows the model as a Bayesian network being used to assess a sequence of a person’s hand performing a “stop” gesture. This model is a mixture of coupled hidden Markov models.

Our observations consist of the video image regions, I, and the temporal derivatives, f_t, between pairs of images over these regions. We assume here that the image regions are given at each frame. The temporal derivatives (along with spatial derivatives) induce a dense optical flow field, by assuming that the image intensity structure is locally constant across short periods of time (the *brightness constancy assumption*). The optical flow field is a projection of the 3D scene velocity to the image plane, and gives the motion in the image at each pixel. Thus, the measurements we start from contain simultaneous descriptions of the instantaneous configuration and dynamics of the body. The task is first to spatially summarise both of these quantities, then to temporally compress the entire sequence to a distribution over high level descriptors, D.

The spatial abstraction of images and temporal derivatives occurs in the two vertical chains in Figure 2, culminating in distributions over the multivariate random variables, W and X, for images and temporal derivatives, respectively. W and X correspond to classes of instantaneous configuration and dynamics of the region of interest in the training data. For example, the configuration classes may correspond to characteristic facial poses, such as the apex of a smile. The dynamics classes are motion classes, and may correspond to, for example, motion during expansion of the face to a smile.

The same method is used for spatial abstraction of both the configuration and dynamics of the face. Image regions and optical flow fields are each projected to a predetermined set of basis functions, yielding finite dimensional feature vectors, Z_w and Z_x, respectively. The basis set is complete and orthogonal, such that Z_w and Z_x can be used to reconstruct images and flow fields to an arbitrary degree of accuracy, given sufficient basis projections. The basis functions are ordered by their spatial frequencies, such that low orders represent gross structure in images and flow fields, and higher orders represent more complex structures. Using a pre-determined basis set defers any commitment to particular types of motion to higher levels of processing, without affecting computational efficiency. We use the basis of Zernike polynomials, which have useful properties for modeling flow fields [9] and images [19]. Zernike polynomials are defined over a unit disk, and are complete and orthogonal, such that the feature vectors can be used for reconstruction of images or flow fields. The distributions of each of the feature vectors (for configuration, Z_w, and dynamics, Z_x) are modeled by a mixture of Gaussians distribution, where the mixture components are labeled as states of W and X. The mixture models at this stage also include feature weights as priors on the cluster means [9]. These feature weights obviate the need to choose which basis functions are useful for classification. Figure 3 shows the output distributions of the Gaussian mixture model in the dynamics chain, X, plotted along the two most significant features, for the same model as was shown in Figure 2 ($D = 4$). Reconstructed flow fields are shown for the means of two of the states of X, as well as the trajectory for the sequence in Figure 2.

The dynamics and configuration variables, X and W, each form Markovian chains, called the *dynamics* and *configuration* processes, which are coupled. Temporal abstraction is achieved using a mixture model at the high level, where the mixture components are coupled hidden Markov models. Thus, each state of the high level display descriptor, D, *generates* a coupled hidden Markov model. The CHMM, in turn, generates images (through the configuration chain) and temporal derivatives (through the dynamics chain) for each time step in the sequence.

This mixture model can compute the likelihood of a video sequence given the display descriptor, $P(\mathbf{O}|D)$:

$$P(\{\mathbf{O}\}_{1,T}|D_T) = \sum_{ij}\Theta_f\Theta_I\Theta_{Xijk}\sum_{kl}\Theta_{Wjkl}P(X_{T-1,k},W_{T-1,l}\{\mathbf{O}\}_{1,T-1}|D_T)$$

where Θ_{Xijk} and Θ_{Wjkl} are the transition functions in the coupled chains, and $\Theta_f = P(f_t|X_{T,i})$ and $\Theta_I = P(I_t|W_{T,j})$ are the likelihoods of temporal derivatives and image regions given dynamics and configuration states, respectively. Details can be found in [9].

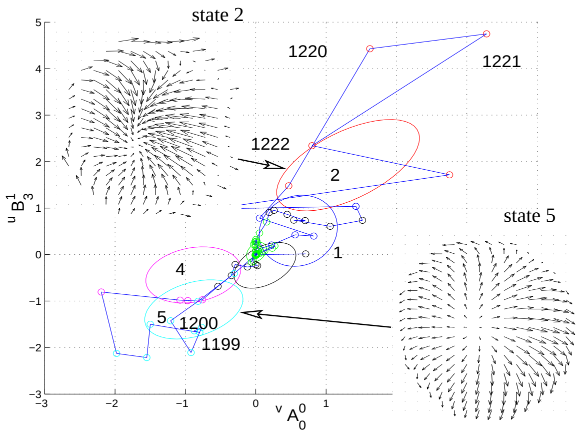

Figure 3: Gaussian output distributions of dynamics mixture model $D = 4$. Level curves of the covariance for each state of $X = 1 \ldots 6$ are shown. Reconstructed flow fields for two of the Gaussian means correspond to movement towards (state 5) and away from (state 2) the camera. The trajectory for a sequence classified as this model is also shown.

2.2 Learning POMDPs

We use the expectation-maximization (EM) algorithm [5] to learn the parameters of the POMDP. It is important to stress that the learning takes place over the *entire* model simultaneously: both the output distributions, including the mixtures of coupled HMMs, and the high-level POMDP transition functions are all learned from data during the process. The learning classifies the input video sequences into a spatially and temporally abstract finite set, *Acom*, and learns the relationship between these high-level descriptors, the observable context, and the action. Learning the POMDP parameters is to find the set of parameters, $\mathbf{\Theta}^*$, which maximize the posterior density of all observations and the model, $P(\mathbf{OCA\Theta})$, subject to constraints on the parameters. The EM algorithm eases this maximization by writing it as

$$\mathbf{\Theta}^* = \arg\max_{\mathbf{\Theta}}\left[\sum_{\mathbf{D}} P(\mathbf{D}|\mathbf{OCA\Theta}')\log P(\mathbf{DOCA}|\mathbf{\Theta}) + \log P(\mathbf{\Theta})\right]$$

The "E" step of the EM algorithm is to compute the expectation over the hidden state, $P(\mathbf{D}|\mathbf{OCA\Theta}')$, given $\mathbf{\Theta}'$, a current guess of the parameter values. The "M" step is then to perform the maximization which, in this case, can be computed analytically by taking derivatives with respect to each parameter, setting to zero and solving for the parameter. The resulting update equations for the parameters of the POMDP transition functions are the same as for an *input-output* hidden Markov model [1]. The updates to the output CHMM distributions are very similar to those for a normal

HMM, except that evidence is propagated backwards and forwards through both X and W chains. Equations for the updates to the output distributions of the CHMMs, including to the feature weights, can be found in [9].

2.3 Solving POMDPs

If observations are drawn from a finite set, then an optimal policy of action can be computed for a POMDP [11] using dynamic programming over the space of the agent's belief about the state, $b(s)$. However, if the observation space is continuous, as in our case, the problem becomes much more difficult. In fact, there are no known algorithms for computing optimal policies for such problems. Nevertheless, approximation techniques have been developed, the simplest of which simply considers the POMDP as a fully observable MDP (the *MDP approximation*): the state, S, is assigned its most likely value in the belief state, $S = \arg\max_s b(s)$. This approximation will be sufficient for the examples we present. Dynamic programming updates consist of computing value functions, V^n, where $V^n(s)$ gives the expected value of being in state s with a future of n stages to go, assuming the optimal actions are taken at each step. The actions that maximize V^n are the policy with n stages to go. These value functions are computed by setting $V^0 = R$ (the reward function), and then iterating [11]

$$V^{n+1}(s) = R(s) + \max_{a \in \mathcal{A}} \left\{ \sum_{t \in \mathcal{S}} Pr(t|a,s) \cdot V^n(t) \right\} \quad (1)$$

The actions that maximize Equation 1 form the approximately optimal n stage-to-go policy, $\pi^n(s)$,

2.4 Value directed structure learning

The value function, $V(s)$, gives the expected value for the decision maker in each state. However, there may be parts of the state space which are indistinguishable (or nearly so) with respect to certain characteristics, such as value or optimal action choice. These indistinguishable states can be grouped or merged together to form an *aggregate* or *abstract* state. The set of abstract states *partitions* the state space according to some characteristic. States of the original MDP which are part of the same abstract state are not distinguishable insofar as decisions go. Eliminating the distinctions between them by merging states can lead to efficiency gains without compromising decision quality.

In fact, such state aggregation is a form of structure learning based upon the value of states. This *value-directed* structure learning is in contrast to more data dependent structure learning, in which the structure is determined solely based upon the statistical distribution of the data, and the complexity of the model. For example, many structure learning algorithms use some simplicity prior (such as the minimum description length [22]), and find a trade-off between the model's precision and complexity.

We now discuss a particular technique for value-directed state aggregation applied to learning the number of facial displays or gestures that need to be distinguished in our learned POMDP. As we have mentioned, the state space is represented in a factored POMDP as a product over a set of variables. In our model, the values of one of these variables, $Acom$, are the (unlabeled) gestures or facial displays. This variable splits the value function into N_a pieces, V_i, one for each value, i, of the variable $Acom$. Each such V_i gives the values of being in any state in which $Acom = i$. A similar split occurs for the policy, yielding sub-policies, π_i, giving the actions to take for each $Acom = i$. The V_i can be compared by computing the difference between them, $d_{ij} = ||V_i - V_j||$, where $||X|| \equiv max\{x : x \in X\}$ is the supremum norm. Two sub-policies, π_i and π_j, are considered equivalent if the optimal actions agree for every state, denoted $\pi_i \wedge \pi_j$. These comparisons are used in the following algorithm for learning the number of display states, N_a. The algorithm starts by assigning N_a to be as large as the training data will support, and prunes redundant states.

```
repeat
  1.learn the POMDP model
  2.compute $V_i$ and $\pi_i$ $\forall$ $i$
  3.compute $d_{ij} = ||V_i - V_j||$ $\forall$ $(i,j), i \neq j$
  4.if $\exists (i,j)(\pi_i \wedge \pi_j)$
  5.    $\{i,j\} = \arg\min_{\{kl\}}(d_{kl} \forall \{k,l\} \mid \pi_k \wedge \pi_l)$
  6.    merge states $i$ and $j$
  7.    $N_a \leftarrow N_a - 1$
    end
until $N_a$ stops changing
```

There are many potential ways to merge states at step 6, but we simply delete one of the the redundant states. Note that the algorithm could also start with $N_a = 2$ and add states until redundancies appear, but we have not experimented with this version [12]. The new states could be initialized randomly, or as a current state with added noise.

3 Experiments

We investigated the use of our POMDP model for modeling hand gestures and facial displays. The hand gestures were designed for simple robotic direction control, and were recorded in a training session with a single stationary camera. The rewards were explicitly assigned during the learning process by the operator. We recorded facial displays and player actions during a card game, played by two humans. The reward function was the points the players won in the game.

3.1 Hand Gestures for Robot Control

We recorded a set of examples of four hand gestures, designed for simple robotic direction control: *forwards*, *stop*, *go left* and *go right*. A dozen examples of each gesture were performed by a single subject in front of a stationary camera during a training session. Video was grabbed from a firewire camera at 150 × 150 with a narrow field of view. The region of interest was taken to be the entire image, and so no tracking was required. Clearly, this would only be possible with a static camera. Sequences were taken of a fixed length of 90 frames. A robotic agent (not embodied at this stage) chose actions in response to each gesture according to a random policy, and was rewarded by the operator's *good* or *bad* action, $Aact$, for choosing the correct action.

It is important to re-state that these experiments are not meant to demonstrate a general gesture recognition system. It is clear that, with this simple tracking and registration method (taking the whole image), this system would not deal with the high variability in gesture orientation or speed. These experiments are meant as a simple demonstration of the value-directed structure learning techniques: they show how our system can correctly discover the number of *meaningful* gestures in a simple interaction.

We trained the POMDP with $N_a = 6$ states. The value function and policy are shown in Figure 4 as decision diagrams. The policies for states d_2 and d_5 are equivalent and their values are identical, and so the algorithm merges them first by simply deleting state d_5. The POMDP is re-trained, resulting in a five-state value function (not shown), in which two more states are found to agree and are merged. Again the POMDP is re-trained, this time giving a value function and policy in which no displays are found to be redundant, shown in Figure 5. Figure 2 showed part of a sequence of a *stop* gesture classified as model d_3. Figures 6 and 7 show parts of sequences of *forwards* and *left* gestures, classified as model d_4 and d_1, respectively, in the new 4-state POMDP. The figures show the images and temporal derivatives along the bottom row, the expected values of the image projections and the flow fields given the merged model in the middle row, and the expected values of the distributions over the dynamics and configuration states, X and W, along the top row.

To evaluate how well the model chooses actions, we performed a cross-validation experiment in which the POMDP was trained on all but one sequence of each gesture. The model was then used to choose actions based upon the four sequences left out. If the action is correct, one reward is given. This process is repeated for 12 different sets of four test sequences, and the total rewards gathered give an indication of how well the model performs on unseen data. Out of a total of $12 \times 4 = 48$ rewards available, the model collected 47, for a total success rate of $47/48$ or 98%. The one failure was due to a mis-classification of a "left" gesture as a "right" gesture due to a large rightwards motion of the hand at the beginning of the stroke. The final POMDP models learned that there were $N_a = 4$ states in all 12 cases.

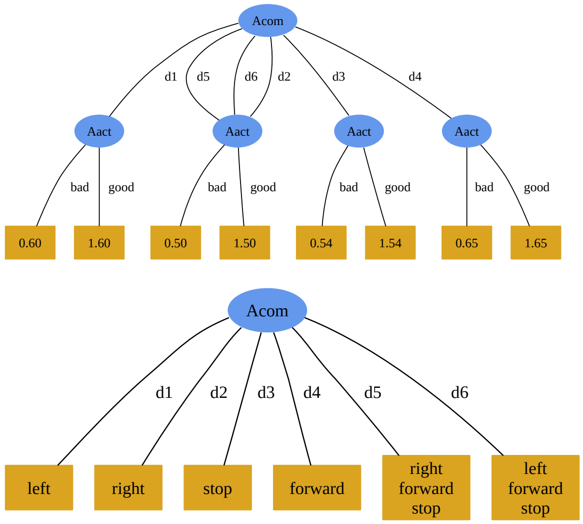

Figure 4: Original six-state value function (top) and policy (bottom), shown as decision diagrams. States are the labels on each path from the root to a leaf, which contains the value or optimal action for that state.

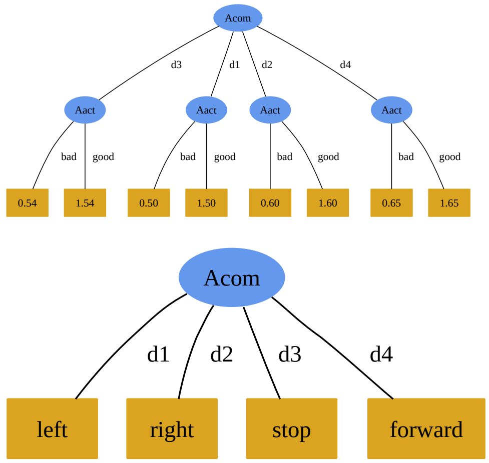

Figure 5: Final four-state merged value function (top) and policy (bottom). The value function

3.2 Facial Displays in Games

We trained the POMDP model on videos of two humans playing a cooperative card game. In each round of the game, players attempt to play matching cards after an initial phase in which they can communicate with each other through a real-time video link (with no audio). There are

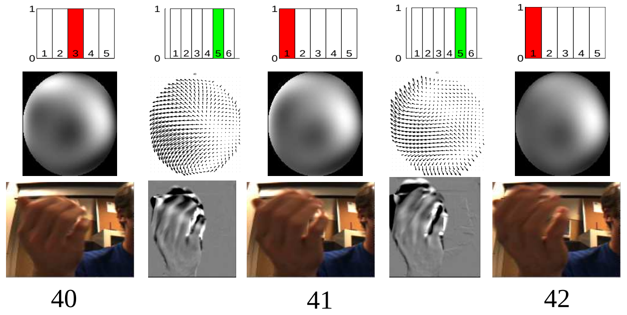

Figure 6: Part of a sequence of a “forwards” gesture, classified as model d_4.

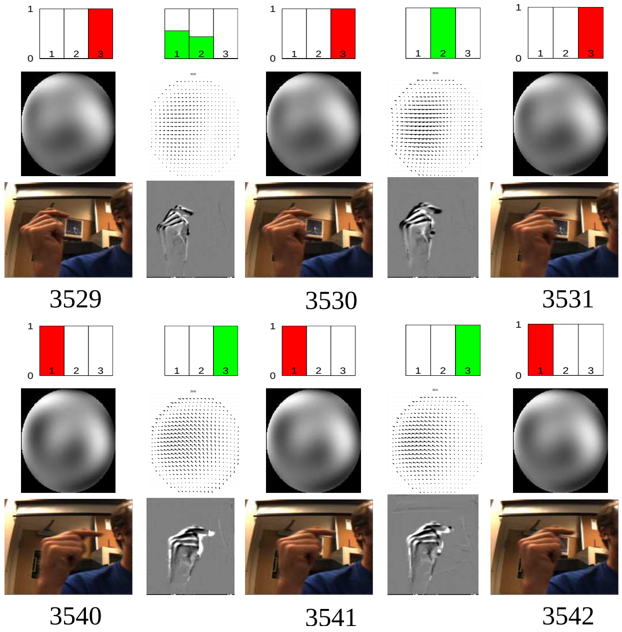

Figure 7: Part of a sequence of a “left” gesture, classified as model d_1.

no game rules concerning the video link, so there are no restrictions placed on communication strategies the players can use. The players naturally came up with simple head gestures to help them win the game: nodding and shaking. The facial regions of the players were tracked in the video using an optical flow based tracker, with corrections from an exemplar database [9].

The data was split into training and test sets, and our POMDP model with $N_a = 4$ display states was learned with the training set. The learning discovered appropriate motion sequence models for each of the head gestures the players were using. Two of the learned display states described neutral displays with little motion, while one described head nods, and the other head shakes [8].

An approximate two-stage policy of action was computed using the MDP approximation, and the structure learning algorithm described in Section 2.4 was applied. Two states were merged, resulting in a three-state model. The two merged models both described “null” sequences, with little facial motion. After merging, the three states corresponded to head shakes (d_2), head nods (d_3), and a null display (d_1).

Although the training data set was large enough to learn models of the head gestures, it was small for learning a POMDP, resulting in sub-optimal policies for many of the states not visited in the training data. In order to attenuate the effects of the lack of training data, we may assume that player’s do not have any particular preference over card suits, such that the conditional probability tables should be symmetric under permutation of suits. Therefore, we can “symmetrise” the probability distributions by simply averaging over the six card suit permutations. The merged and symmetrised three-state model was applied to the test data, the POMDP inferred the facial displays that the players were using, and was able to predict the human player’s actions in $6/7$ test cases and $19/20$ training cases.

Figure 8 shows example frames from a sequence in which the subject shook her head. The entire sequence was classified as facial display state d_3 by the final merged model with three states. Figure 9 shows example frames from a sequence in which the subject nodded her head, classified as facial display state d_2 by the final merged model.

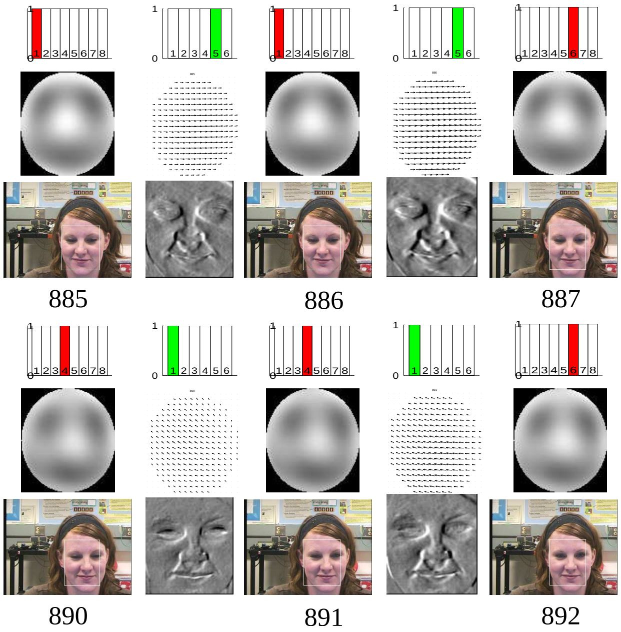

Figure 8: Part of a sequence of subject shaking her head, classified as model d_3.

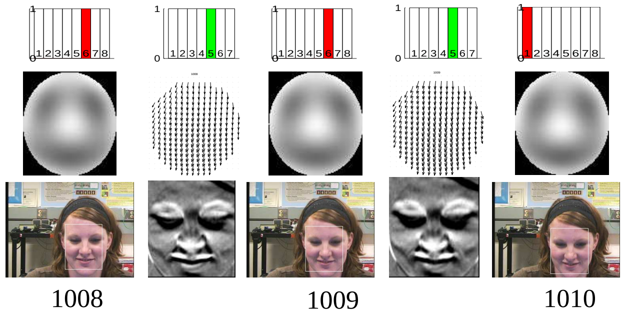

Figure 9: Part of a sequence of subject nodding, classified as model d_2.

4 Conclusions

We have presented a method for learning decision theoretic models of purposeful human non-verbal displays using partially observable Markov decision processes. It discovers spatially and temporally abstract categories of motion sequences and their relationship with actions, utilities and context automatically from video. No prior knowledge about the types of displays expected in an interaction is needed to train the model. The learned values of states are used to discover the number of display classes which are important for achieving value in the context of the interaction. This type of value-directed structure learning allows an agent to only focus resources on necessary distinctions. Our work is primarily focused on learning the parameters and structure of POMDPs. To demonstrate this learning, we use solution techniques that approximate the POMDP as a fully observable MDP. In future work, these approximations will be relaxed, but the concepts of value-directed learning will remain [15].

Acknowledgements: Supported by the Institute for Robotics and Intelligent Systems (IRIS), a Canadian Network of Centres of Excellence, and by a Precarn scholarship. The authors thank Nicole Arksey, Pascal Poupart and our anonymous reviewers.

References

[1] Y. Bengio and P. Frasconi. Input-output HMMs for sequence processing. *IEEE Transactions on Neural Networks*, 7(5):1231–1249, September 1996.

[2] M. Black and Y. Yacoob. Tracking and recognizing rigid and non-rigid facial motions using local parametric models of image motions. *IJCV*, 25(1):23–48, 1997.

[3] J. Cassell, J. Sullivan, S. Prevost, and E. Churchill, editors. *Embodied Conversational Agents*. MIT Press, 2000.

[4] T. Darrell and A. Pentland. Active gesture recognition using partially observable Markov decision processes. In *13th IEEE ICPR*, Vienna, Austria, 1996.

[5] A. Dempster, N.M.Laird, and D. Rubin. Maximum likelihood from incomplete data using the EM algorithm. *Journal of the Royal Statistical Society*, 39(B):1–38, 1977.

[6] S. Fine, Y. Singer, and N. Tishby. The hierarchical Hidden Markov Model *Machine Learning*, 32(1):41–62, 1998.

[7] A. J. Fridlund. *Human facial expression: an evolutionary view*. Academic Press, San Diego, CA, 1994.

[8] J. Hoey and J. J. Little. Decision theoretic modeling of human facial displays. In *Proc. ECCV*, Prague, CZ, 2004.

[9] J. Hoey and J. J. Little. Decision theoretic modeling of human facial displays. Technical Report TR-04-02, University of British Columbia, Department of Computer Science, 2004.

[10] A. Jebara and A. Pentland. Action reaction learning: Analysis and synthesis of human behaviour. In *IEEE Workshop on The Interpretation of Visual Motion*, 1998.

[11] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra. Planning and acting in partially observable stochastic domains. *Artificial Intelligence*, 101:99–134, 1998.

[12] R. A. McCallum. Overcoming incomplete perception with utile distinction memory. In *Proc.* 10^{th} *ICML*, 1993.

[13] D. McNeill. *Hand and Mind: What Gestures Reveal about Thought*. University of Chicago Press, Chicago, IL, 1992.

[14] N. Oliver, E. Horvitz, and A. Garg. Layered representations for human activity recognition. In *Proceedings of International Conference on Multimodal Interfaces*, Pittsburgh, PA, October 2002.

[15] P. Poupart and C. Boutilier. Value-directed compression of POMDPs. In *NIPS*, 15, pages 1547–1554, MIT Press, 2003.

[16] N. Roy, J. Pineau, and S. Thrun. Spoken dialogue management using probabilistic reasoning. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics (ACL2000)*, Hong Kong, 2000.

[17] J. A. Russell and J. M. Fernández-Dols, editors. *The Psychology of Facial Expression*. Cambridge University Press, Cambridge, UK, 1997.

[18] T. Starner and A. P. Pentland. Visual recognition of american sign language using hidden Markov models. In *International Workshop on Automatic Face and Gesture Recognition*, pages 189–194, Zurich, Switzerland, 1995.

[19] C.-H. Teh and R. T. Chin. On image analysis by the methods of moments. *IEEE Trans. PAMI*, 10(4):496–513, July 1988.

[20] S. Thrun. Monte Carlo POMDPs. In *NIPS* 12, pages 1064–1070. MIT Press, 2000.

[21] Y. Tian, T. Kanade, and J. F. Cohn. Recognizing action units for facial expression analysis. *IEEE Trans. PAMI*, 23(2), February 2001.

[22] M. Walter, A. Psarrou, and S. Gong. Data driven gesture model acquisition using minimum description length. In *Proc. BMVC*, Manchester, UK, September 2001.