Latent-Dynamic Discriminative Models for Continuous Gesture Recognition

Louis-Philippe Morency Ariadna Quattoni Trevor Darrell
MIT Computer Science and Artificial Intelligence Laboratory
{lmorency,ariadna,trevor}@csail.mit.edu

Abstract

Many problems in vision involve the prediction of a class label for each frame in an unsegmented sequence. In this paper, we develop a discriminative framework for simultaneous sequence segmentation and labeling which can capture both intrinsic and extrinsic class dynamics. Our approach incorporates hidden state variables which model the sub-structure of a class sequence and learn dynamics between class labels. Each class label has a disjoint set of associated hidden states, which enables efficient training and inference in our model. We evaluated our method on the task of recognizing human gestures from unsegmented video streams and performed experiments on three different datasets of head and eye gestures. Our results demonstrate that our model compares favorably to Support Vector Machines, Hidden Markov Models, and Conditional Random Fields on visual gesture recognition tasks.

1. Introduction

Visual gesture sequences tend to have distinct internal sub-structure and exhibit predictable dynamics between individual gestures. In this paper, we introduce a new visual gesture recognition algorithm which can capture both sub-gesture patterns and dynamics between gestures. Our Latent-Dynamic Conditional Random Field (LDCRF) model is a discriminative approach for gesture recognition. In contrast to generative approaches (e.g., Hidden Markov Models [26, 4]), our model discovers latent structure that best differentiates visual gestures and can distinguish subtle motion patterns such as natural head nods and eye gaze aversion [14].

An LDCRF offers several advantages over previous discriminative models. In contrast to Conditional Random Fields (CRFs) [11], our method incorporates hidden state variables which model the sub-structure of gesture sequences. The CRF approach models the transitions between gestures, thus capturing extrinsic dynamics, but lacks the ability to represent internal sub-structure. In contrast to Hidden-state Conditional Random Fields (HCRFs) [18], our method can learn dynamics between gesture labels and can be directly applied to label unsegmented sequences.

The LDCRF model thus combines the strengths of CRFs and HCRFs by capturing both extrinsic dynamics and intrinsic sub-structure. It learns the extrinsic dynamics by modeling a continuous stream of class labels, and it learns internal sub-structure by utilizing intermediate hidden states. Since LDCRF models include a class label per observation (see Figure 1), they can be naturally used for recognition on un-segmented sequences, overcoming one of the main weaknesses of the HCRF model. By associating a disjoint set of hidden states to each class label, inference on LDCRF models can be efficiently computed using belief propagation during training and testing. Our results on visual gesture recognition demonstrate that LDCRF models compare favorably to models based on Support Vector Machines (SVMs), HMMs, CRFs and HCRFs[1].

2. Related Work

There is a wide range of related work for visual gesture recognition (see surveys [28] and [6]). Recognition of head gestures has been demonstrated by several authors, using generative models of eye and/or head position over time. Kapoor and Picard presented a technique to recognize head nods and head shakes based on two Hidden Markov Models (HMMs) trained and tested on 2D coordinate results from an eye gaze tracker [8]. Kawato and Ohya developed a technique for head gesture recognition using "between eyes" templates [9]. Fugie *et al.* also used HMMs to perform head nod recognition [4]. They combined head gesture detection with prosodic recognition of Japanese spoken utterances to determine strongly positive, weak positive, and negative responses to yes/no type utterances. HMMs [19] and related models have been used to recognize arm gestures [2] and sign language gestures [1, 22].

Recently many researchers have worked on modeling eye gaze behavior for the purpose of synthesizing a realistic Embodied Conversational Agent (ECA). Colburn *et*

[1] C++ and Matlab implementations of our LDCRF model as well as CRF and HCRF models can be downloaded at http://sourceforge.net/projects/hcrf.

al. use hierarchical state machines to model eye gaze patterns in the context of real-time verbal communication [3]. Fukayama *et al.* use a two-state Markov model based on amount of gaze, mean duration of gaze, and gaze points while averted [5]. Lee *et al.* use an eye movement model based on empirical studies of saccade and statistical models of eye-tracking data [12]. Pelachaud and Bilvi proposed a model that embeds information on communicative functions as well as on statistical information of gaze patterns [17].

A significant amount of recent work has shown the power of discriminative models for specific sequence labeling tasks. In the speech and natural language processing community, Conditional Random Field (CRF) models have been used for tasks such as word recognition, part-of-speech tagging, text segmentation and information extraction [11]. In the vision community, Sminchisescu *et al.* applied CRFs to classify human motion activities (i.e. walking, jumping, etc) and showed improvements over an HMM approach [21]. Kumar *et al.* used a CRF model for the task of image region labeling [10]. Torralba *et al.* introduced Boosted Random Fields, a model that combines local and global image information for contextual object recognition [24]. An advantage of CRFs is that they can model arbitrary features of observation sequences and can therefore accommodate overlapping features.

When visual phenomena have distinct sub-structure, models that exploit hidden state are advantageous. Hidden-state conditional random fields (HCRFs), which can estimate a class given a segmented sequence, have been proposed in both the vision and speech community. In the vision community, HCRFs have been used to model spatial dependencies for object recognition in cluttered images [18] and for arm and head gesture recognition from segmented sequences [27]. In the speech community, a similar model was applied to phone classification [7]. Since they are trained on sets of pre-segmented sequences, these HCRF models do not capture the dynamics between gesture labels, only the internal structure. In both [7] and [27], HCRFs were applied to segmented sequences, leaving segmentation as a pre-processing step.

Sutton *et al.* [23] presented a dynamic conditional random field (DCRF) model whose structure and parameters are repeated over a sequence. They showed results for sequence segmentation and labeling where the model was trained using loopy belief propagation on a fully-observed training set. As stated by the authors, training a DCRF model with unobserved nodes (hidden variables) makes their approach difficult to optimize. Our LDCRF incorporates hidden state variables with an explicit partition; inference can be efficiently computed using belief propagation during both training and testing.

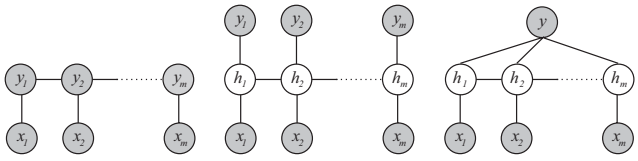

Figure 1. Comparison of our LDCRF model with two previously published models: CRF [11] and HCRF [7, 27]. In these graphical models, x_j represents the j^{th} observation (corresponding to the j^{th} frame of the video sequence), h_j is a hidden state assigned to x_j , and y_j the class label of x_j. Gray circles are observed variables. The LDCRF model combines the strengths of CRFs and HCRFs in that it captures both extrinsic dynamics and intrinsic structure and can be naturally applied to predict labels over unsegmented sequences. Note that only the link with the current observation x_j is shown, but for all three models, long range dependencies are possible.

3. Latent-Dynamic Conditional Random Fields

Several problems in vision can be thought of as discrete sequence labeling tasks over visual streams. We focus on problems where the goal is to predict a class label at each point in time for a given sequence of observations. In the case of human gesture recognition, a sequence of video frames is given and the goal is to predict a gesture label per frame. We are interested in visual sequences that exhibit both dynamics within each class label (i.e. intrinsic structure) and varied transitions between class labels (i.e. extrinsic structure).

Our task is to learn a mapping between a sequence of observations $\mathbf{x} = \{x_1, x_2, ..., x_m\}$ and a sequence of labels $\mathbf{y} = \{y_1, y_2, ..., y_m\}$. Each y_j is a class label for the j^{th} frame of a video sequence and is a member of a set $\mathcal{Y}$ of possible class labels. Each frame observation x_j is represented by a feature vector $\phi(x_j) \in \mathbf{R}^d$, for example, the head velocities at each frame. For each sequence, we also assume a vector of "sub-structure" variables $\mathbf{h} = \{h_1, h_2, ..., h_m\}$. These variables are not observed in the training examples and will therefore form a set of hidden variables in the model.

Given the above definitions, we define a latent conditional model:

$$P(\mathbf{y} \mid \mathbf{x}, \theta) = \sum_{\mathbf{h}} P(\mathbf{y} \mid \mathbf{h}, \mathbf{x}, \theta) P(\mathbf{h} \mid \mathbf{x}, \theta). \qquad (1)$$

where θ are the parameters of the model.

To keep training and inference tractable, we restrict our model to have disjoint sets of hidden states associated with each class label. Each h_j is a member of a set $\mathcal{H}_{y_j}$ of possible hidden states for the class label y_j. We define $\mathcal{H}$, the set of all possible hidden states to be the union all $\mathcal{H}_y$ sets.

Since sequences which have any $h_j \notin \mathcal{H}_{y_j}$ will by definition have $P(\mathbf{y} \mid \mathbf{h}, \mathbf{x}, \theta) = 0$, we can express our model as:

$$P(\mathbf{y} \mid \mathbf{x}, \theta) = \sum_{\mathbf{h}:\forall h_j \in \mathcal{H}_{y_j}} P(\mathbf{h} \mid \mathbf{x}, \theta). \quad (2)$$

We define $P(\mathbf{h}|\mathbf{x}, \theta)$ using the usual conditional random field formulation:

$$P(\mathbf{h}|\ \mathbf{x}, \theta) = \frac{1}{\mathcal{Z}(\mathbf{x}, \theta)} \exp\left(\sum_k \theta_k \cdot \mathbf{F}_k(\mathbf{h}, \mathbf{x})\right), \quad (3)$$

where the partition function $\mathcal{Z}$ is defined as

$$\mathcal{Z}(\mathbf{x}, \theta) = \sum_{\mathbf{h}} \exp\left(\sum_k \theta_k \cdot \mathbf{F}_k(\mathbf{h}, \mathbf{x})\right),$$

F_k is defined as

$$\mathbf{F}_k(\mathbf{h}, \mathbf{x}) = \sum_{j=1}^{m} f_k(h_{j-1}, h_j, \mathbf{x}, j),$$

and each feature function $f_k(h_{j-1}, h_j, \mathbf{x}, j)$ is either a state function $s_k(h_j, \mathbf{x}, j)$ or a transition function $t_k(h_{j-1}, h_j, \mathbf{x}, j)$. State functions s_k depend on a single hidden variable in the model while transition functions t_k can depend on pairs of hidden variables.

3.1. Learning Model Parameters

Our training set consists of n labeled sequences $(\mathbf{x_i}, \mathbf{y_i})$ for $i = 1...n$. Following [10, 11], we use the following objective function to learn the parameter θ^*:

$$L(\theta) = \sum_{i=1}^{n} \log P(\mathbf{y}_i \mid \mathbf{x}_i, \theta) - \frac{1}{2\sigma^2}||\theta||^2 \quad (4)$$

The first term in Eq. 4 is the conditional log-likelihood of the training data. The second term is the log of a Gaussian prior with variance σ^2, i.e., $P(\theta) \sim \exp\left(\frac{1}{2\sigma^2}||\theta||^2\right)$.

We use gradient ascent to search for the optimal parameter values, $\theta^* = \arg\max_\theta L(\theta)$, under this criterion. Given Equations 2 and 3 , the gradient of $\log P(\mathbf{y}_i \mid \mathbf{x}_i, \theta)$ for one particular training sequence $(\mathbf{x_i}, \mathbf{y_i})$ with respect to the parameters θ_k associated with a state function s_k can be written as (details omitted for space):

$$\begin{array}{l} \sum_{j,a} P(h_j = a \mid \mathbf{y}, \mathbf{x}, \theta) s_k(j, a, \mathbf{x}) \\ -\sum_{\mathbf{y}', j, a} P(h_j = a, \mathbf{y}' \mid \mathbf{x}, \theta) s_k(j, a, \mathbf{x}) \end{array} \quad (5)$$

where

$$P(h_j = a \mid \mathbf{y}, \mathbf{x}, \theta) = \frac{\sum_{\mathbf{h}:h_j=a \wedge \forall h_j \in \mathcal{H}_{y_j}} P(\mathbf{h} \mid \mathbf{x}, \theta)}{\sum_{\mathbf{h}:\forall h_j \in \mathcal{H}_{y_j}} P(\mathbf{h}\ |\mathbf{x}, \theta)} \quad (6)$$

Notice that given our definition of $P(\mathbf{h}|\mathbf{x}, \theta)$ in Equation 3, the summations in Equation 6 are simply constrained versions of the partition function $\mathcal{Z}$ over the conditional random field for $\mathbf{h}$. This can be easily shown to be computable in $O(m)$ using belief propagation [16], where m is the length of the sequence.

The gradient of our objective function with respect to the parameters θ_k associated to a transition function t_k can be derived the same way. The marginal probabilities on edges necessary for this gradient, $P(h_j = a, h_k = b \mid \mathbf{y}, \mathbf{x}, \theta)$, can also be computed efficiently using belief propagation. In our experiments, we performed gradient ascent with the BFGS optimization technique. All the models required fewer than 300 iterations to converge.

We train our LDCRF model on data labeled with class labels (but not hidden states), yielding a classifier which can be run directly on unsegmented visual sequences. We have found that assuming each class label has a disjoint set of hidden states significantly simplifies model training, but is still powerful enough to improve recognition performance over conventional discriminative sequence methods. Our LDCRF model has a similar computational complexity to a fully observable CRF.

3.2. Inference

For testing, given a new test sequence $\mathbf{x}$, we want to estimate the most probable label sequence $\mathbf{y}^*$ that maximizes our conditional model:

$$\mathbf{y}^* = \arg\max_{\mathbf{y}} P(\mathbf{y} \mid \mathbf{x}, \theta^*) \quad (7)$$

where the parameter values θ^* are learned from training examples. Assuming each class label is associated with a disjoint set of hidden states, the previous equation can be rewritten as:

$$\mathbf{y}^* = \arg\max_{\mathbf{y}} \sum_{\mathbf{h}:\forall h_i \in \mathcal{H}_{y_i}} P(\mathbf{h} \mid \mathbf{x}, \theta^*) \quad (8)$$

To estimate the label y_j^* of frame j, the marginal probabilities $P(h_j = a \mid \mathbf{x}, \theta^*)$ are computed for all possible hidden states $a \in \mathcal{H}$. Then the marginal probabilities are summed according to the disjoint sets of hidden states $\mathcal{H}_{y_j}$ and the label associated with the optimal set is chosen. As discussed in the previous subsection, the marginal probabilities can efficiently be computed using belief propagation. While another option would be to compute the Viterbi path, in our experiments we use the above maximal marginal probabilities approach to estimate the sequence of labels since it minimizes the error per frame.

3.3. Feature Functions

In our model, $|\mathcal{H}| \times |\mathcal{H}|$ transitions functions t_k are defined one for each hidden state pair (h', h''). Each transition

function is expressed as,

$$t_k(h_{j-1}, h_j, \mathbf{x}, j) = \begin{cases} 1 \text{ if } h_{j-1} = h' \text{ and } h_j = h'' \\ 0 \text{ otherwise} \end{cases}$$

It is worth noticing that the weights θ_k associated with the transition functions model both the intrinsic and extrinsic dynamics. Weights associated with a transition function for hidden states that are in the same subset $\mathcal{H}_{y_i}$ will model the substructure patterns, while weights associated with the transition functions for hidden states from different subsets will model the external dynamic between gestures.

The number of state functions, s_k, will be equal to the length of the feature vector, $\phi(x_j)$, times the number of possible hidden states, $|\mathcal{H}|$. In the case of head gesture recognition where the rotational velocity (yaw, roll and pitch) is used as input, the length of our feature vector, $\phi(x_j)$, will be 3 dimensions per frame. If our model has 6 hidden states (3 per label) then the total number of state functions, s_k, (and total number of associated weights θ_k) will be $3 \times 6 = 18$.

3.4. Synthetic Example

We illustrate how our LDCRF model can capture both extrinsic dynamics and intrinsic structure using a simple example. A synthetic dataset was constructed containing sequences from a gesture class and a background class. Subsequences belonging to the gesture class consisted of three sub-gesture samples simulating the beginning, middle and end of a gesture and were created by sampling from three Gaussian distributions in a deterministic order. The background subsequences were generated by randomly selecting k samples from a mixture of seven Gaussians, where k is the length of the subsequence, picked at random between 1 and 10. Both the training and testing datasets consisted of 200 1-dimensional sequences of variable length (30-50 samples per sequence). Synthetic sequences were constructed by randomly concatenating subsequences sampled from gesture and background classes[2]

Given this dataset we trained both a LDCRF model with three hidden states per labels and a CRF. The CRF model was only able to recognize 72% (equal error rate) of the test examples with this simple synthetic dataset, while our LDCRF model has perfect performance. Figure 2 shows the sequence of hidden labels assigned by our LDCRF model for a sequence in the testing set. As this figure suggests the model has learned the intrinsic structure of the class using one of its hidden states to recognize each of the substructures. In the following section we present results comparing LDCRF performance with five baseline models on natural gesture datasets.

[2]Source code of this synthetic example is part of the LDCRF distribution.

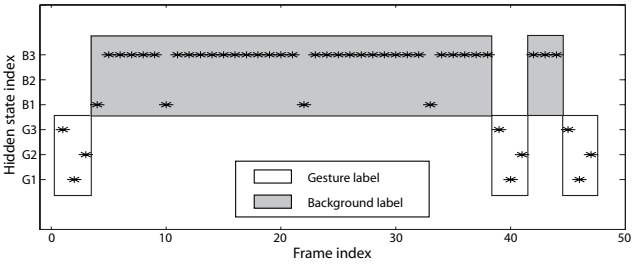

Figure 2. This figure shows assignment of hidden states given by the LDCRF model for a sample sequence from the synthetic dataset. As we can see the model has used the hidden states to learn internal sub-structure of the gesture class.

4. Experiments

We evaluate the performance of our LDCRF model for visual gesture recognition using head and eye motion datasets. Our datasets came from three user studies: (1) head gestures when interacting with a physical avatar [20], (2) head gesture-based widgets for a non-embodied interface [13], and (3) eye gestures for interaction with a virtual embodied agent [14].

In this section we describe the three datasets used in our experiments, present the baseline models used to evaluate the performance of our LDCRF model and describe our experimental methodology. In the following section we report and discuss the results.

4.1. Datasets

MELHEAD This dataset consisted of head velocities from 16 human participants interacting with Mel, an interactive robot [20]. Each interaction lasted between 2 to 5 minutes. Head pose tracking was performed online using the adaptive view-based appearance model [15]. At each frame ($\sim$25Hz), the tracker logged with timestamps the 3D position and orientation of the head.

Human participants were video recorded while interacting with the robot to obtain ground truth labels including the start and end points of each head nod. From these ground truth labels, each frame was labeled as either being a head-nod or background gesture/motion. A total of 274 head nods were naturally performed by the participants while interacting with the robot. All other types of head gestures (e.g. head shakes, look away, no motion) were labeled as being in the background class.

WIDGETSHEAD This dataset consisted of head velocities from 12 participants who interacted with gesture-based widgets [13]. Similar to the first dataset, head pose was estimated using the adaptive view-based appearance model [15]. The video sequences were manually annotated

for ground truth. Each frame was labeled as either being a head-nod or background gesture/motion; from 79 minutes of interaction, 269 head nods were labeled. All other types of head gestures (e.g. head shakes, look away, no motion) were labeled as being in the background class.

AVATAREYE This dataset consisted of eye gaze estimates from 6 human participants interacting with a virtual embodied agent [14]. The goal is to recognize gaze aversion gestures — eye movements to empty or uninformative regions of space, reflecting "look-away" or "thinking" — from unsegmented video sequences. Each video sequence lasted approximately 10-12 minutes, and was recorded at 30 frames/sec. During these interactions, human participants would rotate their head up to +/-70 degrees around the Y axis and +/-20 degrees around the X axis, and would also occasionally translate their head, mostly along the Z axis.

For each video sequence, eye gaze was estimated using the view-based appearance model described in [14] and for each frame a 2-dimensional eye gaze estimate was obtained. The dataset was labeled with the start and end points of each gaze aversion gesture as described in [14]. Each frame was manually labeled as either a gaze-aversion gesture or a background gesture/motion; the latter included sections of video where people were looking at the avatar or performing eye deictic gestures.

4.2. Models

In our experiments, the LDCRF model is compared with five models: Conditional Random Field (CRF), Hidden-state Conditional Random Field (HCRF), Hidden Markov Model (HMM) and Support Vector Machine (SVM) methods. Note that we tested two HMM configurations: an HMM with a sliding window (referred to as HMM-S in the results section) and an HMM that incorporates external dynamic (referred to as HMM).

CONDITIONAL RANDOM FIELD As a first baseline, we trained a single CRF chain model where every gesture class has a corresponding state label. During evaluation, marginal probabilities were computed for each state label and each frame of the sequence using belief propagation. In our baseline implementation the optimal label for a specific frame is selected to be the label with the highest marginal probability. In our case, to be able to plot ROC curves of our results, the marginal probability of the gesture was thresholded at each frame, and the frame was given a positive label if the marginal probability was larger than the threshold. The objective function of the CRF model contains a regularization term similar to the regularization term shown in Equation 4 for the LDCRF model. During training and validation, this regularization term was validated with values $10^k, k = -3..3$.

SUPPORT VECTOR MACHINE As a second baseline, a multi-class SVM was trained with one label per gesture using a Radial Basis Function (RBF) kernel. Since the SVM does not encode the dynamics between frames, the training set was decomposed into frame-based samples, where the input to the SVM is the head velocity or eye gaze at a specific frame. The output of the SVM is a margin for each class. This SVM margin measures how close a sample is to the SVM decision boundary [25]. The margin was used to plot the ROC curves. During training and validation, two parameters were validated: C, the penalty parameter of the error term in the SVM objective function, and γ, the RBF kernel parameter. Both parameters were validated with values $10^k, k = -3..3$.

HIDDEN MARKOV MODEL As a third baseline, an HMM was trained for each gesture class. We trained each HMM with segmented subsequences where the frames of each subsequence all belonged to the same gesture class. This training set contained the same number of frames as the one used for training the other models except frames were grouped into subsequences according to their label. As we stated earlier, we tested two configurations of Hidden Markov Models: an HMM evaluated over a sliding window (referred to as HMM-S in our experiments) and a concatenated HMM that incorporates external dynamics (referred to as HMM). For the first configuration, each trained HMM is tested separately on the new sequence using a sliding window of fixed size (1.3 seconds, which is equal to the average gesture length). The class label associated with the HMM with the highest likelihood is assigned to the frame at the center of the sliding window.

For the second configuration, the HMMs trained on subsequences are concatenated into a single HMM with the number of hidden states equal to the sum of hidden states from each individual HMM. For example, in a binary recognition problem where each individual HMM is trained using 3 hidden states, the concatenated HMM will have 6 hidden states. To estimate the transition matrix of the concatenated HMM, we compute the Viterbi path of each training subsequence, concatenate the subsequences into their original order, and then count the number of transitions between hidden states. The resulting transition matrix is then normalized so that its rows sum to one. At testing, we apply the forward-backward algorithm on the new sequence, and then sum at each frame the hidden states associated with each class label. The resulting HMM can be seen as a generative version of our LDCRF model. During training and validation, we varied the number of states from 1 to 6 and the number of Gaussians per mixture from 1 to 3.

HIDDEN-STATE CONDITIONAL RANDOM FIELD As a fourth baseline, we trained a HCRF model on all gesture classes as described in [27]. Since HCRFs cannot model dynamics between gestures, we trained the HCRF on segmented sub-sequence (the same training set as the HMM-S model). At testing, the HCRF model is applied on the new sequence using a sliding window of fixed size (1.3 seconds). The class label with the highest likelihood is assigned to the frame at the center of the sliding window. During training and validation, we varied the number of hidden states (from 2 to 6 states) and the regularization term (with values $10^k, k = -3..3$).

LATENT-DYNAMIC CONDITIONAL RANDOM FIELD Our LDCRF model was trained using the objective function described in Section 3.1. During evaluation, we compute ROC curves using the maximal marginal probabilities of Equation 8. During training and validation, we varied the number of hidden states per label (from 2 to 6 states per label) and the regularization term (with values $10^k, k = -3..3$).

4.3. Methodology

For all three datasets, the experiments were performed using a K-fold testing approach where K sequences were held out for testing while all other sequences were used for training and validation. This process was repeated N/K times, where N is the total number of sequences. For the `MelHead`, `WidgetsHead` and `AvatarEye` datasets, K was 4, 3 and 1 respectively. For validation, we performed holdout cross-validation where a subset of the training set is kept for validation. The size of this validation set was equal to 4, 3 and 1 for the `MelHead`, `WidgetsHead` and `AvatarEye` datasets respectively. The optimal validation parameters were chosen based on the equal error rates on the validation set.

All three datasets contained an unbalanced number of gesture frames compared to background frames. To have a balanced training set and to reduce the training time, the training dataset was preprocessed to create a smaller training dataset containing an equal number of gesture and background examples. The training set was a set of sequences where each sequence either was uniformly background class, or contained an example of the gesture class, with a buffer of background frames before and after the gesture. The size of the buffer before and after the gesture randomly varied between 2 and 50 frames. Background subsequences were randomly extracted from the original sequences with length varying between 30-60 frames.

Each experiment was also repeated with different input feature window sizes. A window size equal to one means that only the feature vector at the current frame was used to create the input feature. A window size of five means that the input feature vector at each frame is a concatenation of the feature vectors from five frames: the current frame, the two preceding frames, and the two future frames.

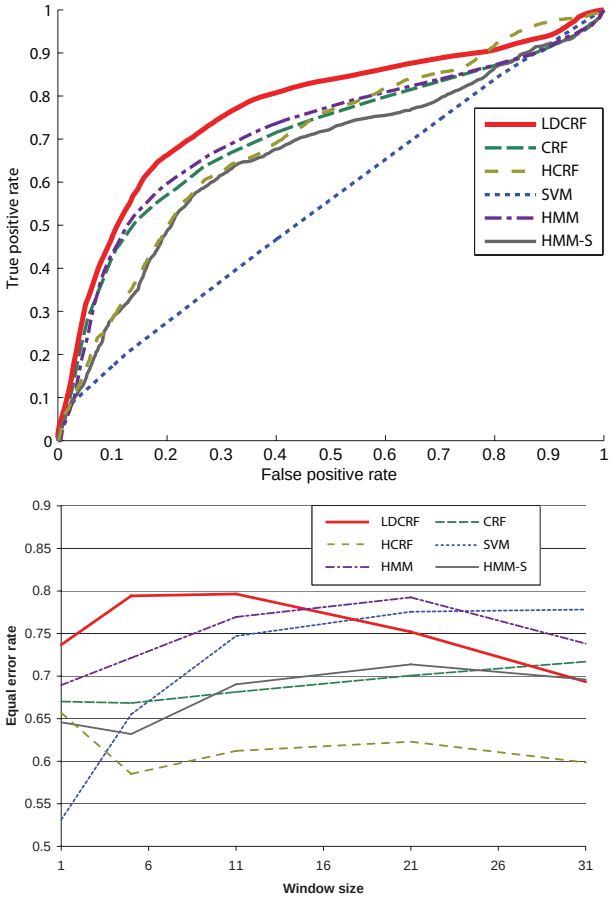

Figure 3. Results with `WidgetsHead` dataset: (top) ROC curves for a window size of one (no long range dependencies); (bottom) Accuracy at equal error rates as a function of the window size.

5. Results and Discussion

In this section, the results of our experiments for head and eye gesture recognition are presented. We compared all six models (SVM, CRF, HMM, HMM-S, HCRF and LDCRF) on three datasets. For the ROC curves shown in this section, the true positive rate is computed by dividing the number of recognized frames by the total number of ground truth frames. Similarly, the false positive rate is computed by dividing the number of falsely recognized frames by the total number of background frames.

Figure 3 compares the ROC curves from the six models for a window size of one for the `WidgetsHead` dataset. A plot of the Equal Error Rate (EER) — recognition rate at which both the true positive rate and the true negative rate

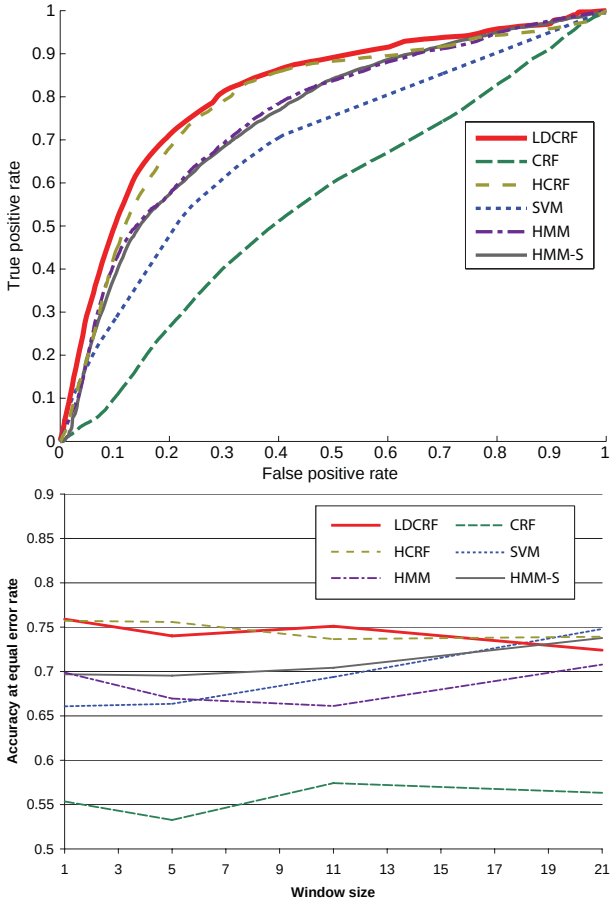

Figure 4. Results with `MelHead` dataset: (top) ROC curves for a window size of one; (bottom) Accuracy at equal error rates as a function of the window size.

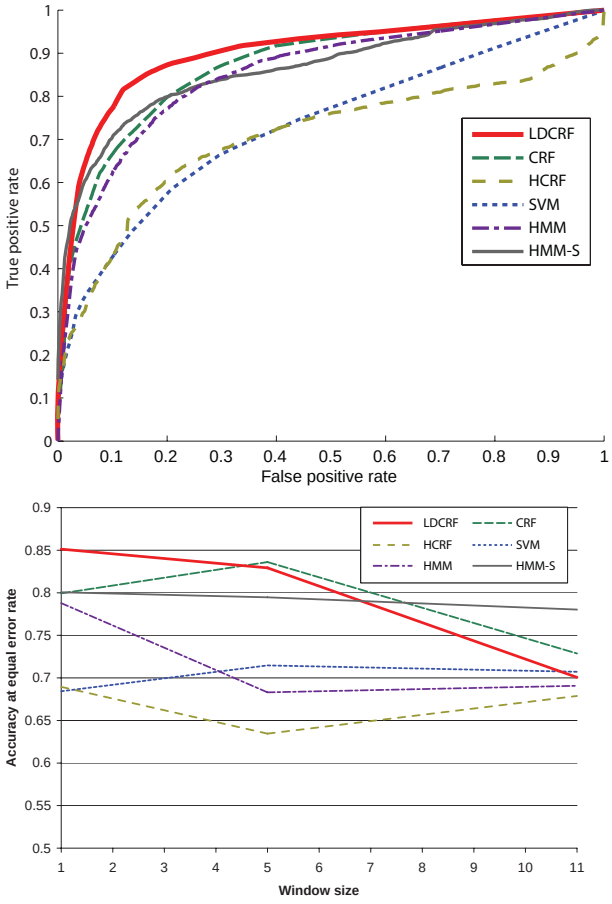

Figure 5. Results with `AvatarEye` dataset: (top) ROC curves for a window size of one; (bottom) Accuracy at equal error rates as a function of the window size.

are equal — as a function of the window size is also shown for each model. As can be seen in the figure, the LDCRF model outperforms baseline methods.

For online gesture recognition, an important model property is the ability to perform accurate recognition without requiring future observations (corresponding to smaller window size). In both the ROC curves and the EER plots of Figure 3, the LDCRF model outperforms all the other models when using a small window size. This difference is statistically significant according to a paired t-test on the EER accuracies per participant.

Figure 4 shows the recognition results for the `MelHead` dataset and Figure 5 shows results for the `AvatarEye` dataset. Similar to the `WidgetsHead` dataset, the LDCRF outperforms the other models when evalutated on these two datasets. It is particularly interesting to compare the CRF and HCRF performances in the EER plots of these two datasets. For the `MelHead` dataset, the HCRF performs better than the CRF model while for the `AvatarEye` dataset, the CRF model does better. This can be explained by the observation that transitions between gestures are well defined in the `AvatarEye` dataset while the internal sub-structure is the prominent factor in the `MelHead` dataset. The LDCRF model combines the strengths of CRFs and HCRFs by capturing both extrinsic dynamics and intrinsic sub-structure and as such exhibits the best performance on both datasets.

In the `MelHead` dataset, the human participants were standing in front of the robot, and were able to move freely about their environment, making the class of `other-gesture` quite diversified and challenging for generative approaches such as HMMs. The `AvatarEye` dataset had only 6 participants and 77 eye gestures. We can see in Figure 5 how this small dataset affects the LDCRF model when the window size increases. This effect was not as prominent for larger datasets, as observed in Figures 3 and 4. Even with this small dataset, LDCRF outperforms the five other models with a maximum accuracy of 85.1% for a window size of one.

6. Conclusion

In this paper we presented a discriminative framework for simultaneous sequence segmentation and labeling which can capture both intrinsic and extrinsic class dynamics. Our LDCRF model incorporates hidden state variables which model the sub-structure of a class sequence and learn dynamics between class labels. We performed experiments on the task of recognizing human gestures from unsegmented video streams and showed how our model outperforms state-of-the-art generative and discriminative modeling techniques on three different datasets. As future work we plan to evaluate our model on other visual sequence labeling problems that exhibit intrinsic and extrinsic sequence dynamics, such as activity detection and audio-visual speech recognition.

References

[1] M. Assan and K. Groebel. Video-based sign language recognition using hidden markov models. In *Int'l Gest Wksp: Gest. and Sign Lang.*, 1997.

[2] M. Brand, N. Oliver, and A. Pentland. Coupled hidden markov models for complex action recognition. In *CVPR*, 1996.

[3] R. A. Colburn, M. F. Cohen, and S. M. Ducker. The role or eye gaze in avatar mediated conversational interfaces. Technical Report MSR-TR-2000-81, Microsoft Research, July 2000.

[4] S. Fujie, Y. Ejiri, K. Nakajima, Y. Matsusaka, and T. Kobayashi. A conversation robot using head gesture recognition as para-linguistic information. In *ROMAN*, pages 159–164, September 2004.

[5] A. Fukayama, T. Ohno, N. Mukawa, M. Sawaki, and N. Hagita. Messages embedded in gaze of interface agents — impression management with agent's gaze. In *CHI '02*, pages 41–48, 2002.

[6] D. M. Gavrila. The visual analysis of human movement: A survey. *CVIU'*, 73(1):82–98, 1999.

[7] A. Gunawardana, M. Mahajan, A. Acero, and J. C. Platt. Hidden conditional random fields for phone classification. In *INTERSPEECH*, 2005.

[8] A. Kapoor and R. Picard. A real-time head nod and shake detector. In *PUI*, November 2001.

[9] S. Kawato and J. Ohya. Real-time detection of nodding and head-shaking by directly detecting and tracking the between-eyes. In *FG*, pages 40–45, 2000.

[10] S. Kumar and M. Herbert. Discriminative random fields: A framework for contextual interaction in classification. In *ICCV*, 2003.

[11] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: probabilistic models for segmenting and labelling sequence data. In *ICML*, 2001.

[12] S. P. Lee, J. B. Badler, and N. I. Badler. Eyes alive. In *SIGGRAPH '02*, pages 637–644, 2002.

[13] L.-P. Morency and T. Darrell. Head gesture recognition in intelligent interfaces: The role of context in improving recognition. In *IUI*, Australia, 2006.

[14] L.-P. Morency and T. Darrell. Recognizing gaze aversion gestures in embodied conversational discourse. In *ICMI*, Banff, Canada, November 2006.

[15] L.-P. Morency, A. Rahimi, and T. Darrell. Adaptive view-based appearance model. In *CVPR*, volume 1, pages 803–810, 2003.

[16] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, 1988.

[17] C. Pelachaud and M. Bilvi. Modelling gaze behavior for conversational agents. In *IVA*, pages 15–17, September 2003.

[18] A. Quattoni, M. Collins, and T. Darrell. Conditional random fields for object recognition. In *NIPS*, 2004.

[19] L. R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.

[20] C. Sidner, C. Lee, L.-P. Morency, and C. Forlines. The effect of head-nod recognition in human-robot conversation. In *Conference on Human-Robot Interaction*, March 2006.

[21] C. Sminchisescu, A. Kanaujia, Z. Li, and D. Metaxas. Conditional models for contextual human motion recognition. In *ICCV*, 2005.

[22] T. Starner and A. Pentland. Real-time asl recognition from video using hidden markov models. In *ISCV*, 1995.

[23] C. Sutton, K. Rohanimanesh, and A. McCallum. Dynamic conditional random fields: Factorized probabilistic models for labeling and segmenting sequence data. In *ICML*, 2004.

[24] A. Torralba, K. Murphy, and W. Freeman. Contextual models for object detection using boosted random fields. In *NIPS*, 2004.

[25] V. Vapnik. *The nature of statistical learning theory*. Springer, 1995.

[26] S. Wang and D. Demirdjian. Inferring body pose from speech content. In *ICMI*, 2005.

[27] S. Wang, A. Quattoni, L. Morency, D. Demirdjian, and T. Darrell. Hidden conditional random fields for gesture recognition. In *In CVPR*, 2006.

[28] Y. Wu and T. S. Huang. Vision-based gesture recognition: A review. *LNCS*, 1739:103+, 1999.