Action representation and recognition

- Core difficulties
  - The configuration of the body remains difficult to understand. This may not be essential to understand what is going on (appearance-based approach, etc.).
  - There is no natural taxonomy of activity.
  - Composition creates fearsome complexity.

Temporal scale

- Very short timescales
  - no much happens
- Medium timescales
  - running, walking, jogging, jumping, punching, kicking, reaching
- Long timescales: motions are complex composites
  - visiting an ATM, reading a book, cooking a meal

Action Recognition Using HMMs

- Feature extraction
- Training
- Testing
Action Recognition Using HMMs

- Feature extraction
- Training
- Testing

Coupled Hidden Markov Model

- HMMs are usually formulated with multivariate pdfs on the output variables.
- If there are multiple processes generating those channels, one must hope that these processes evolve in lockstep since any variation between them is modeled as noise.
- Two processes may interact without wholly determining each other. E.g. Players in a tennis game, two hand movements.
Coupled HMM
(Nefrian, Liang, Pi, Liu, Mao and Murphy, 2002)

The parameters in CHMM

\[
\begin{align*}
\pi^0(i) &= P(q^0 = i) \\
\beta^0(i) &= P(O_t | q^0 = i) \\
\alpha^0_{i,j} &= P(q^0 = i, q^1_{t-1} = j, q^1_{t-1} = k) \\
b^0(i) &= \sum_{m=1}^{M} \omega^0_{i,m} N(O_t, \mu^0_{i,m}, \Sigma^0_{i,m})
\end{align*}
\]

Training

- Initialization
  \[
  \delta_0(i,j) = \pi^0(i) \beta^0(i) b^0_j
  \]
  \[
  \psi_0(i,j) = 0
  \]

- Recursion
  \[
  \delta_t(i,j) = \max_{k,l} \{ \delta_{t-1}(k,l) a_{i,k} a_{j,l} b^0_l \}
  \]
  \[
  \psi_t(i,j) = \arg \max_{k,l} \{ \delta_{t-1}(k,l) a_{i,k} a_{j,l} \}
  \]

Optimal state sequence using the Viterbi algorithm

- Termination
  \[
  P = \max_{i,j} \{ \delta_T(i,j) \}
  \]
  \[
  \{q^T, q^T\} = \arg \max_{i,j} \{ \delta_T(i,j) \}
  \]

- Backtracking
  \[
  \{q^t, q^t\} = \psi_{t+1}(q^t+1, q^t+1)
  \]
The results so far (r – a training sequence)

The sequence of states

\[ Q = q_r^{a,v} \ldots q_r^{a,v} \ldots q_r^{a,v} \]

The sequence of mixture components

\[ S = s_r^{a,v} \ldots s_r^{a,v} \ldots s_r^{a,v} \]

\[ s_r^{a,v} = \max_{m=1,\ldots,M_{i,v}} P(O_t^{a,v} | q_r^{a,v} = i, m) \]

Two Latent Variables

\[ \gamma_{r,l}^{a,v}(i, m) = \begin{cases} 1, & \text{if } q_r^{a,v} = i, s_r^{a,v} = m, \\ 0, & \text{otherwise} \end{cases} \]

\[ \epsilon_{r,l}^{a,v}(i, k, l) = \begin{cases} 1, & \text{if } q_r^{a,v} = i, q_{r,l}^{a,v} = k, q_{r,l}^{a,v} = l, \\ 0, & \text{otherwise} \end{cases} \]

Asynchronous HMM (Bengio, NIPS, 2003)

- Model the joint probability of asynchronous sequences describing the same event.
- Early integration
- Later integration
- The key idea is to temporarily stretch one stream in order to obtain a better match between the corresponding frames.
The model

For the sake of simplicity, let us present here the case where one is interested in modeling the joint probability of two asynchronous sequences, denoted \( x^T \) and \( y^T \) with \( S \leq T \) without loss of generality:

\[
e(i, t) = P(\tau_t = s|\tau_{t-1} = s - 1, q_t = i, x^T_1, y^T_1)
\]

the probability that the system emits the next observation of sequence \( y \) at time \( t \) while in state \( i \). The additional hidden variable \( \tau = s \) can be seen as the alignment between \( y \) and \( x \) (and \( x \) which is aligned with \( q \)). Hence, we model \( P(x^T, y^T, q^T) \).

\[
P(q_t = i, \tau_t = s|\tau_{t-1} = s - 1, x^T_1, y^T_1) = \frac{\alpha(i, s, t) \beta(i, s, t)}{P(x^T_1, y^T_1)}
\]

\[
P(q_t = i, \tau_t = s|\tau_{t-1} = s, x^T_1, y^T_1) = \frac{\alpha(i, s, t) \beta(i, s, t)}{P(x^T_1, y^T_1)}
\]

\[
P(q_t = i, \tau_t = s|\tau_{t-1} = s; x^T_1, y^T_1) = \frac{P(q_t = i|q_{t-1} = s)}{P(x^T_1, y^T_1)} \cdot \frac{\sum_{x,s-1} \alpha(j, s - 1, t - 1) \beta(j, s, t) + \sum_{x,s} \alpha(j, s, t - 1) \beta(j, s, t) \cdot (1 - \alpha(i, t)) \beta(i, s, t)}{\sum_{x,s} \alpha(j, s, t - 1) \beta(j, s, t) \cdot (1 - \alpha(i, t)) \beta(i, s, t)}
\]

Modeling individual and group actions in meetings with layered HMMs (Dong, et al. 2005)

- **audio**
  - 12 channels, 48 kHz
  - 1 microphone array

- **video**
  - 3 cameras
  - all synchronized
Experiment setup

- 59 meetings (30/29 train/test)
- four-people, five-minute
- scripts
  - schedule of actions
  - natural behavior
- features: 5 f/s

Modeling individual and group actions in meetings with layered HMMs (Dong, et al. 2005)

- decompose the recognition problem
- both layers use HMMs
  - individual action layer: \textbf{I-HMM:} various models
  - group action layer: \textbf{G-HMM}

Multi-modal turn-taking

- describes the group discussion state
  \[ A = \{ \text{‘discussion’}, \text{‘monologue’ \times 4}, \text{‘white-board’}, \text{‘presentation’}, \text{‘note-taking’}, \text{‘monologue + note-taking’ \times 4}, \text{‘white-board + note-taking’}, \text{‘presentation + note-taking’} \} \]

- individual actions
  \[ I = \{ \text{‘speaking’}, \text{‘writing’}, \text{‘idle’} \} \]

- actions are multi-modal in nature

Example

- Person 1: S, W, S, W, W
- Person 2: W, S, S, W, W
- Person 3: W, S, S, W, W
- Person 4: S, W, S

- Presentation: Used
- Whiteboard: Used
- Group Action: Used
multimodal feature extraction: video

- head + hands blobs
  - skin colour models (GMM)
  - head position
  - hands position + features (eccentricity, size, orientation)
  - head + hands blob motion
- moving blobs from background subtraction

recognition with two-layer HMM

- each layer trained independently

- compared with single-layer HMM
  - smaller observation spaces
  - I-HMM trained with much more data
  - G-HMM less sensitive to feature variations
  - combinations can be explored

models for I-HMM

- early integration
  - all observations concatenated
  - correlation between streams
  - frame-synchronous streams

- multi-stream (Dupont, TMM 2000)
  - HMM per stream (a or v), trained independently
  - decoding: weighted likelihoods combined at each frame
  - little inter-stream asynchrony
  - multi-band and a-v ASR

- asynchronous (Bengio, NIPS 2002)
  - a and v streams with single state sequence
  - states emit on one or both streams, given a sync variable
  - inter-stream asynchrony

linking the two layers

- hard decision
  i-action model with highest probability outputs 1; all other models output 0.

- soft decision
  outputs probability for each individual action model

\[
P'_k = \sum_{i \in M_a} P(q_i = i | x_k^i)
\]
experiments: data + setup

- 59 meetings (30/29 train/test)
- four-people, five-minute
- scripts
  - schedule of actions
  - natural behavior
- features: 5 f/s

results: individual actions

<table>
<thead>
<tr>
<th>Method</th>
<th>Features</th>
<th>FER (%)</th>
<th>STD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Early Int.</td>
<td>Visual</td>
<td>34.17</td>
<td>3.64</td>
</tr>
<tr>
<td></td>
<td>Audio</td>
<td>23.48</td>
<td>2.70</td>
</tr>
<tr>
<td>MS-HMM</td>
<td>Audio-Visual</td>
<td>9.98</td>
<td>2.65</td>
</tr>
<tr>
<td>A-HMM</td>
<td>Audio-Visual</td>
<td>7.42</td>
<td>1.33</td>
</tr>
</tbody>
</table>

- visual-only ➤ audio-only ➤ audio-visual
- asynchronous effects between modalities
- accuracy: speaking: 96.6 %, writing: 90.8 %, idle: 81.5 %

results: group actions

- multi-modality outperforms single modalities
- two-layer HMM outperforms single-layer HMM for a-only, v-only and a-v
- best model: A-HMM
- soft decision slightly better than hard decision

<table>
<thead>
<tr>
<th>Method</th>
<th>Features</th>
<th>AER (%)</th>
<th>STD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single-layer</td>
<td>Visual</td>
<td>48.80</td>
<td>3.74</td>
</tr>
<tr>
<td></td>
<td>Audio</td>
<td>36.12</td>
<td>4.12</td>
</tr>
<tr>
<td></td>
<td>Audio-visual</td>
<td>33.77</td>
<td>2.90</td>
</tr>
<tr>
<td>Two-layer</td>
<td>Visual</td>
<td>47.18</td>
<td>2.89</td>
</tr>
<tr>
<td></td>
<td>Audio</td>
<td>37.12</td>
<td>2.14</td>
</tr>
<tr>
<td></td>
<td>Audio-visual</td>
<td>33.77</td>
<td>2.90</td>
</tr>
<tr>
<td>Early Int.</td>
<td>hard</td>
<td>37.98</td>
<td>2.97</td>
</tr>
<tr>
<td></td>
<td>soft</td>
<td>16.55</td>
<td>1.48</td>
</tr>
<tr>
<td>MS-HMM</td>
<td>hard</td>
<td>17.27</td>
<td>2.90</td>
</tr>
<tr>
<td></td>
<td>soft</td>
<td>15.80</td>
<td>1.60</td>
</tr>
<tr>
<td>A-HMM</td>
<td>hard</td>
<td>26.05</td>
<td>2.80</td>
</tr>
<tr>
<td></td>
<td>soft</td>
<td>17.33</td>
<td>1.40</td>
</tr>
</tbody>
</table>

8% improvement

How to compare two sequences
Kovar & Gleicher (2004, Siggraph)

- A direct numerical comparison can be used to determine similarity between two sequences.
- With existing metrics, a large distance may reflect either that motions are unrelated or that they are different variations of the same actions.
- There is no easy to distinguish these two cases.

Variation

Figure 1: Logically similar motions may be numerically dissimilar. Left: A standing front kick vs. a looping side kick. Note the differences in the arms, torso posture, and kick trajectory. Right: While these two reaching motions have somewhat similar skeletal postures, the changes in posture are in completely opposite directions.

Criteria for numerical similarity

- First, corresponding frames should have similar skeleton poses.
- Second, frame correspondences should be easy to identify. That is, related events in the motions should be clearly recognizable.
- How to find frame correspondences? Time alignment

Dynamic time warping

- We start with computing for every pair of frames, forming a grid of distances.
- The time alignment is a path on this grid from one corner to the corner that minimizes the total cost of its cells.
**Time alignment**

- Continuous, monotonic and non-degenerate

**Optimal Path**

**Local optimality**

Second, frame correspondences should be easy to identify. That is, related events in the motions should be clearly recognizable.

- If a cell on the time alignment is a horizontal or vertical 1D local minimum, then the frame correspondence is strong in the sense that holding one frame fixed and varying the other only yields more dissimilar skeletal poses.

Ideally, every cell on the time alignment would be a local minimum since then each correspondence would be obvious.
Valid region

Figure 5: Local minima (magenta) are extended to form the valid region (yellow).

- We compute all 1D local minima and extend each one along the directions in which it is a minimum (horizontal, vertical, or both) until a cell is encountered whose value is larger than a threshold.
- We call the resulting region on the grid the valid region. The time alignment is restricted to the valid region.

Similarity Measurement

- With existing metrics, a large distance may reflect either that motions are unrelated or that they are different variations of the same actions.
- A small distance is a reasonable indicator of similarity.
- We can find close motions and then use them as new queries to find more distant motion.
- Manifold learning algorithms that use local neighborhoods of points to calculate the overall distance (Roweis and Saul, Science, 2000).

Fit Locally, Think Globally

We expect each data point and its neighbors to lie on or close to a locally linear patch of the manifold.

Each point can be written as a linear combination of its neighbors. The weights chosen to minimize the reconstruction error.

$$\min_{W} \sum_{i} \left| X_i - \sum_{j=1}^{K} W_{ij} X_j \right|^2$$  \hspace{1cm} (1)
Properties

- The weights that minimize the reconstruction errors are invariant to rotation, rescaling and translation of the data points.
- The weights characterize the intrinsic geometric properties of each neighborhood.
- The same weights that reconstruct the data points in D dimensions should reconstruct it in the manifold in d dimensions.

Dimension Reduction

Summary

- LLE attempts to preserve local geometry of the data by mapping nearby points on the manifold to nearby points in the low dimensional space
  - Find k nearest neighbors in X space
  - Solve for reconstruction weights W
  - Compute embedding coordinates Y using weights W.

Examples
With existing metrics, a large distance may reflect either that motions are unrelated or that they are different variations of the same actions. A small distance is a reasonable indicator of similarity. We can find close motions and then use them as new queries to find more distant motion. Manifold learning algorithms that use local neighborhoods of points to calculate the overall distance (Roweis and Saul, Science, 2000).

**Motion-energy image (MEI)**

We refer to these binary cumulative motion images as motion-energy images (MEI). Let $I(x, y, t)$ be an image sequence, and let $D(x, y, t)$ be a binary image sequence indicating regions of motion, for many applications image differencing is adequate to generate $D$. Then the binary MEI $E_t(x, y, t)$ is defined

$$E_t(x, y, t) = \bigcup_{i=0}^{t-1} D(x, y, t-i)$$

**MEI examples**

Figure 3: MEIs of peeling action over 0° viewing angle. The smooth change implies only a coarse sampling of viewing direction is necessary to recognize the action from all angles.
Motion-History Image (MHI) with a decay factor

\[ H_t(x, y, t) = \begin{cases} \tau & \text{if } D(x, y, t) = 1 \\ \max \{0, H_{t-1}(x, y, t) - 1\} & \text{otherwise} \end{cases} \]