Predicting Daily Behavior via Wearable Sensors

Brian Clarkson and Alex Pentland

{clarkson, sandy}@media.mit.edu

We report on ongoing research into how to statistically represent the experiences of a wearable computer user for the purposes of day-to-day behavior prediction. We combine natural sensor modalities (camera, microphone, gyros) with techniques for automatic labeling. We have also taken the next required step to build robust statistical models with an extensive data collection experiment, the "I Sensed" series, a 100 day data set consisting of full surround video, audio, and orientation.

Keywords: contextual computing, peripheral sensing, computer vision, computer audition, wearable computing

I. Introduction

Is a person's day-to-day behavior predictable? We are concerned with this question because it is exactly the question that needs to be answered if we are going to build agents (wearable or not) that anticipate. Agents that don't anticipate can react and reconfigure based on the present [1] and the past, but generally don't extrapolate into the future. This is a severe limitation because agents without predictive power cannot engage in preventive measures, "meet you half way", nor engage in behavior modification. This is not to say that a clever engineer couldn't herself notice a particular situation that is clearly indicative of some future state, and thus, manually program an agent to anticipate that future state when the situation occurs. However, definitely for a wearable agent and possibly others, typical situations span the entire complex domain of real life where it is unreasonable to manually design such anticipatory behavior into an agent. [2]

There are many ways to pose the question of predictability. In rough terms, prediction is being able to say with some level of certainty that if **A** happens then some time in the future **B** will happen. What we haven't specified yet is what domain is **A** and **B** coming from. There is a whole spectrum of possibilities for **A** and **B** that has to do with how detailed the agent's sensory input is. Can the agent understand what is being spoken and understand facial expression? Or, can it only know that there are speech-like sounds and something moving? The problem with these two ends of the spectrum of sensor detail is that sensor detail seems to be positively correlated with usefulness. It is our belief and purpose of this work that even at the lower end of sensor detail there are useful artificial intelligence systems that can be built, especially in such complex and rich domains as a wearable affords.

Theoretically, the question of how predictable a person's day-to-day behavior is moot if we have access to a complete description of the state of the world, right down to the electron spins in the user's fingernails. Then supposedly we can just apply the laws of physics and simulate into the future. This inane statement just implies that we will always have to deal with an incomplete description of the world. A realistic approach is to start with the coarsest description of the state of the world, see what can be deduced from it and then move to a slightly finer description. You stop when you have reached the limit of your sensing capabilities or the level of privacy invasion outweighs the benefits.

In this work we will take a straightforward approach to answering this question of predictability. First we will report on an extensive data collection experiment that allows us to build predictive models of a person's day-to-day behavior. Then we will address the problem of building coarse descriptions of the world from these wearable sensors, such as cameras, microphones, and gyros. Last, we describe the results of prediction on these coarse descriptions.

II. Data Collection

The first phase in answering the question of human predictability is to accumulate a series of events and situations experienced by one person over an extended period of time.

A. The "I Sensed" Series: 100 Days of Experiences

The main requirement of learning predictive models from data is to have enough repeated trials of the experiment from which to estimate robust statistics. Ideally experiential data recorded from an individual over a number of years would be ideal. However, other forces such as the computational and storage requirements needed for huge data sets force us to settle for something smaller. We chose 100 days (14.3 weeks) because, while it is a novel period for a data set of this sort, its size is still computationally tractable (approx. 500 gigabytes).

Figure 1: The Data Collection wearable when worn.

The wearable was worn from mid-April to mid-July of 2001 by the author. Refer to the last page (Figure 4) of this paper for actual excerpts from this data set on 4 different scenes: eating lunch, walking up stairs, in a conversation, and rollerblading.

The protocol of the experiment were as follows:

- Data collection commences each day from approx. 10am and continues until approx. 10pm. This varies based on the sleeping habits of the experimental subject.
- The times that the data collection system is not active or worn by the subject is logged and recorded. Such times are typically when: batteries fail, sleeping, showering, and working out.
- In addition to the visual, aural, and orientation sensor data collected by the wearable, the subject is also required to keep a rough journal of his high-level activities to within the closest half hour. Examples of high-level activity are: "Working in the office", "Eating lunch", "Going to meet Michael", etc. while being specific about who, where, and why.
- Every 2 days the wearable is "emptied" of its data, by uploading to a secure server.
- Persons who normally interact with the subject on a day-to-day basis and have a possibility of having a potentially private conversation recorded are asked to sign a consent form and an agreement is made by the experimenters to not disclose the data in way without further consent.

B. The Data Collection Wearable

The sensors chosen for this data set are meant to mimic the human senses. They include visual (2 camera, front and back), auditory (1 microphone), and gyros (for 3 degrees of orientation: yaw, pitch and roll). These match up with the eyes, ears, and inner ear, while taste and smell are not covered because the technology is not available yet. Other possibilities for sensors that have no good reason for being excluded are temperature, humidity, accelerometers, and bio-sensors (e.g. heart-rate, galvanic skin response, glucose levels). The properties of the 3 sensor modalities are as follows: (see Figure 3)

Audio: 16kHz, 16bits/sample (normal speech is generally only understandable for persons in direct conversation with the subject.)

Front Facing Video: 320x240 pixels, 10Hz frame rate (faces are generally only recognizable under bright lighting conditions and from less than 10ft away.)

Back Facing Video: 320x240 pixels, 10Hz frame rate (faces are generally only recognizable under bright lighting conditions and from less than 10ft away.)

Orientation: Yaw, roll, and pitch are sampled at 60Hz. A zeroing switch is installed beneath the left strap which is meant to trigger whenever the subject puts on the wearable. Drift is only reasonable for periods of less than a few hours.

The wearable is based on a backpack design for comfort and wardrobe flexibility. The visual component of the wearable consists of 2 USB cameras (front- and rear-facing) modified to be optically compatible with 200° field-of-view lenses (adapted from door viewers). This means that we are recording light from every direction in a full sphere around the user (but not with even sampling of course). The front-facing camera is sewn to the front strap of the wearable and the rear-facing camera is contained inside the main shell-like compartment. The microphone is attached directly below the front-facing camera on the strap. The orientation sensor (InterTrax2 from Intersensed Inc. with its magnetic field zeroing feature disabled) is housed inside the main compartment. Also in the main compartment are a PIII 500MHz cell computer (CellComputing Inc.) with a 10GB HDD (enough storage for 2 days) and 4 Sony Infolithiums NP-F960 (operating time: ~10 hrs.). The polystyrene shell (see Figure 1) was designed and vacuum-formed to fit the components as snuggly as possible while being aesthetically pleasing, presenting no sharp corners for snagging, and allowing the person reasonable comfort while sitting down.

Since this wearable is only meant for data collection, its input and display requirements are minimal. For basic on/off, pause, record functionality there are click buttons attached to the right-hand strap (easily accessible by the left-hand by reaching across the chest). These buttons are chorded for protection against accidental triggering. All triggering of the buttons (intentional or otherwise) is recorded along with the sensor data. Other than the administrative functions, the buttons also provide a way for the subject to mark salient points in the sensor data. The only display provided by the wearable is 2 LEDs, one for power and the other for recording.

C. *The Data Journal*

Organizing, accessing, and browsing such a large amount of video, audio, and gyro data is a non-trivial engineering task. So far we have a system that allows us to fully transcribe the "I Sensed" series and to access it arbitrarily in a multi-resolution and efficient manner. This ability is essential for learning and feature extraction techniques talked about later in this paper. All data (images, frames of audio, button presses, orientation vectors, etc.) are combined and time synchronized in our data journaling system to millisecond accuracy (see Figure 2).

III. Scene Analysis

The purpose of the scene analysis step is to extract the events that we care about predicting. This might include but is not limited to:

Motion events {walking, turning, running, etc.}
Location events {entering, leaving}
People events {meeting, speech, etc.}

Since we have access to visual, aural, and orientation data, many of these events can conceivably be detected and classified. We address a few in this section.

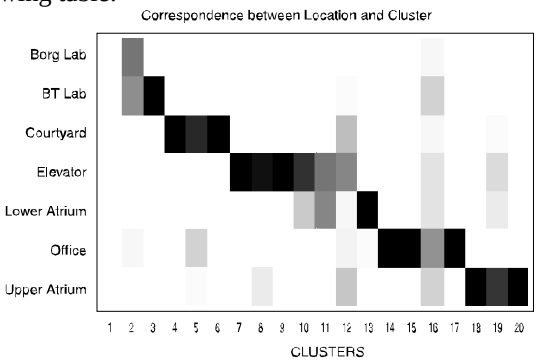
A. *Location*

"Where are you?" is one of the most basic facts about your state. Many basic decisions and events are conditioned on your location and state of your location (e.g. turning down a hallway, meeting someone, turning on the light). For these reasons it is likely to be a powerful clue for guessing what you will do next.

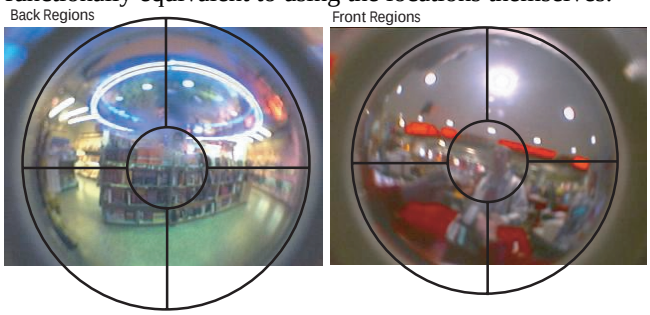
In previous experiments [3] with similar but smaller datasets, we showed that location can be successfully classified (conditioned on time) by likelihood ratio tests with Hidden Markov Models on image histogram-like features (visual ambience). Furthermore, we showed that adding similarly coarse audio features (to capture aural ambience) and combining with the visual features sometimes gave higher or lower classification performance.

	Correct Acceptance (%)			Correct Rejection (%)		
Locations	**A+V**	**A**	**V**	**A+V**	**A**	**V**
BorgLab	95.9	19.1	97.1	92.1	56.2	84.9
BTLab	93.3	63.8	88.3	97.3	48.0	98.8
Courtyard	83.1	38.2	93.0	92.2	64.9	76.6
Elevator	63.6	52.1	62.8	99.8	58.0	98.4
Lower Atrium	95.7	88.7	87.3	60.9	26.8	56.3
Upper Atrium	95.0	56.3	96.0	60.7	52.3	61.4
Office	89.9	42.6	71.1	96.0	87.3	93.5

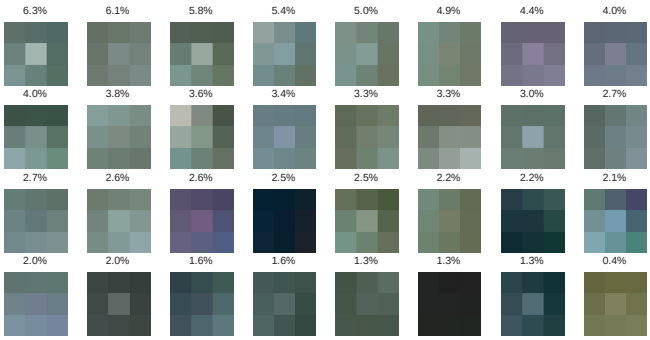
However, building classifiers for all the possible locations in the 100-day dataset, while possible, is not necessary until you need a human-readable label. For our prediction experiments we need an event corresponds to a location. A small experiment shows how we can obtain this simply by clustering the visual features. We clustered the visual features using K-Means and 20 Gaussian centroids. Then we tabulated the correspondence between these 20 clusters and the 7 locations that the data spanned. This results in the following table:



The result is quite pleasing. There is a strong many-to-one mapping of clusters to locations. Of course this almost follows from the fact that we are able to separate these locations with HMMs. Each location is marked by its own particular dynamic of how it switches between a few simple modes of visual appearance. For example notice both the Borg Lab and the BT Lab locations share the cluster #2. However, the BT Lab switches between clusters #2 and #3 while the Borg Lab only exhibits a single mode, cluster #2. This is what allows us to distinguish them apart, even though sometimes the two locations are visually similar. The point of this mini-experiment is to show that if we build a predictor in terms of these visual clusters, it is functionally equivalent to using the locations themselves.



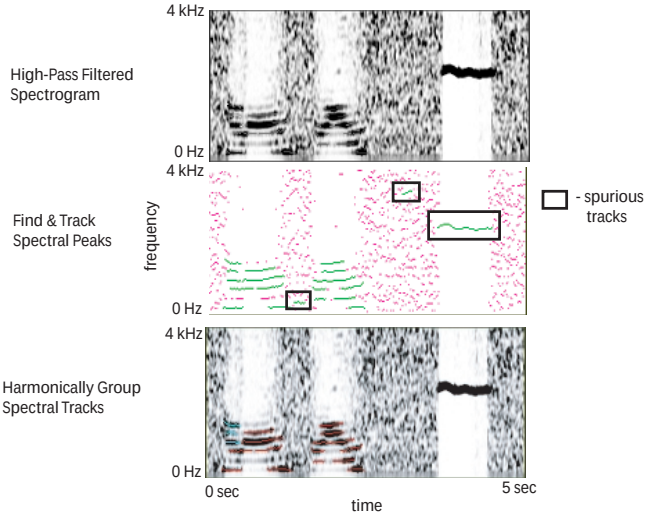
So, we take the front and back views from the "I Sensed" data set and divide them into 5 regions each (see figure ?). Full covariance Gaussians are estimated from the luminance and chrominance (HSV) color values in each region. This yields 9 parameters (3 means, 6 covariances) per region. This set of features is then clustered into 32 clusters using K-Means. The front view clusters are shown here, sorted according to the percentage of the time that they are active.



Visually there is a clear correspondence between indoor and outdoor, night and day scenes. Other correspondences, such as with exact locations, are more difficult to see without calculating the same features for each location and comparing.

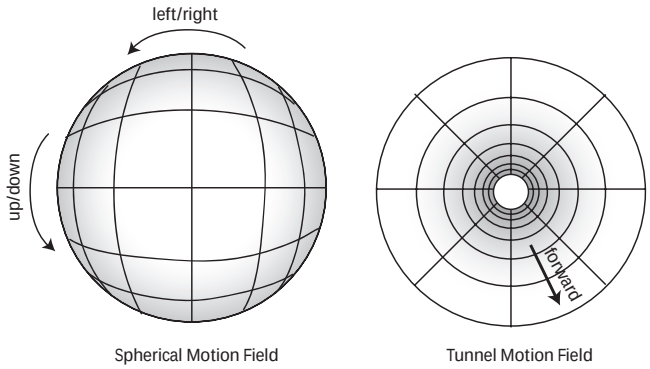
B. Speech Events

To extract speech events from the recorded audio we used a rule-based search system based on the spectrogram. After estimating a power spectrogram from the audio signal, we high-pass filter each frequency bank to remove any biases caused by slowly changing (> 1 minute) sound sources in the environment (e.g. fan noise, wind, hums, etc.). Spectral peaks are then found and tracked in the spectrogram. Finally, these spectral tracks are grouped according their harmonic relationship with one another (related to the harmonic sieve pitch estimator [4]). Thus if two tracks share the same fundamental frequency, they are grouped together as being caused by the same pitched source. Tracks that are not grouped are thrown away as spurious noise. The remaining groups are further filtered, removing the harmonic sounds that don't fall in the typical fundamental frequency range for speech (80-300 Hz).



C. Motion Events

Motion was estimated with a combination of information from the electronic gyros and regularized optical flow from the visual field. The optical flow estimation was necessary because the gyros were found to precess excessively at times and because we wished to estimate the forward motion.



The optical flow in the forward view was calculated using the Lucas-Kanade point tracker [5]. This yields an N-point motion field, which we regularize with two different geometries: spherical and cylindrical (or tunnel). A sphere centered on the optical center of the ultra wide-angle lens was used to estimate the left/right and up/down motion of the camera. A tunnel with its vanishing point at the optical center of the lens was used to estimate the forward motion.

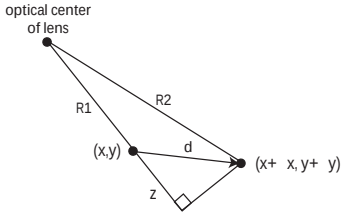
Let $\{(x_i, y_i, \Delta x_i, \Delta y_i)\}_{i=1\ldots N}$ be the points and their displacements received from the Lucas-Kanade tracker with a particular pair of consecutive video frames. We project these displacements on to the constrained motion geometries:

Spherical Motion Field

$$\begin{bmatrix} \Delta\theta_x \\ \Delta\theta_y \end{bmatrix} = \begin{bmatrix} \cos^{-1}\left(\frac{x}{\sqrt{R^2 - y^2}}\right) - \cos^{-1}\left(\frac{x+\Delta x}{\sqrt{R^2 - y^2}}\right) \\ \cos^{-1}\left(\frac{y}{\sqrt{R^2 - x^2}}\right) - \cos^{-1}\left(\frac{y+\Delta y}{\sqrt{R^2 - x^2}}\right) \end{bmatrix}$$

Tunnel Motion Field

$$\Delta z = \frac{R_2^2 - d^2 - R_1^2}{2R_1}$$



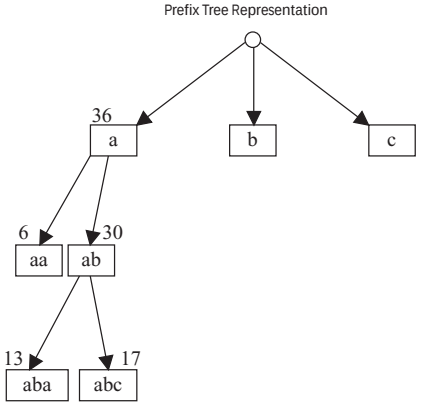
Quantities are defined according to the figure on the right.

The final motion deltas are estimated from these projected displacements by iteratively minimizing a robust error norm (truncated least-squares).

The motion deltas are too noisy for dead reckoning, but they are suitable for input into a classifier that outputs discrete motion labels: {left, right, forward, still}. Thus we trained a simple HMM classifier, one HMM for each of the 4 classes, and used Viterbi to provide a decoding of each frame into one of {left, right, forward, still}.

IV. PREDICTION

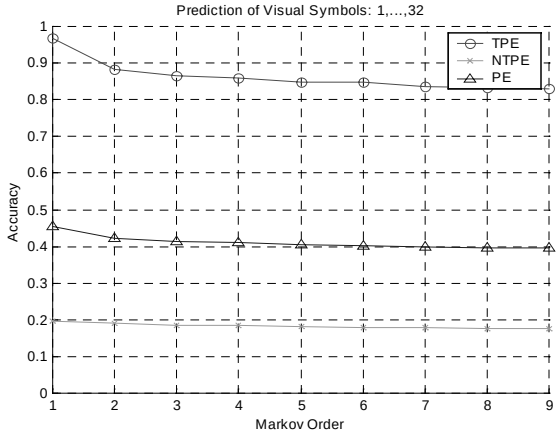
Now our prediction task can be reduced to prediction on a sequence of discrete symbols. The extensive research from text prediction is therefore applicable here. The most popular models in text prediction have simply been the N-gram [6] models typically used in speech-to-text systems and language identification systems. These N-gram models are Nth order Markov processes where the state space corresponds to the symbol alphabet. These N-gram models can be efficiently represented by trees called prediction suffix (or prefix) trees [7]. These trees are memory- and access-efficient data structures for the conditional probability table for $P(s_t \mid s_{t-1},...,s_{t-N})$ where s_t is the symbol at time *t*. They are particularly efficient when the conditional probability table is sparse (i.e. only a small subset of all possible subsequences, $(s_{t-1},...,s_{t-N})$, have non-zero counts) which is usually the case for high-order Markov models. This efficiency directly translates into us being able to estimate and use higher and higher order Markov models.

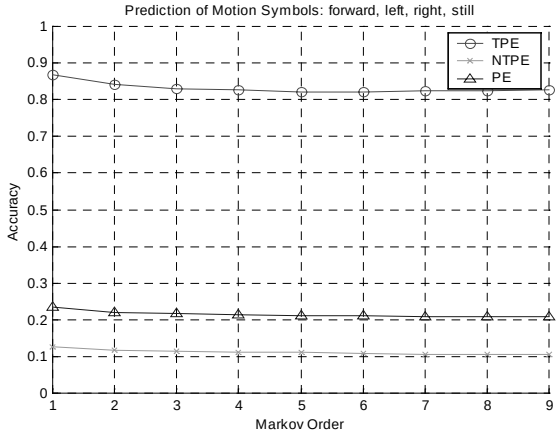


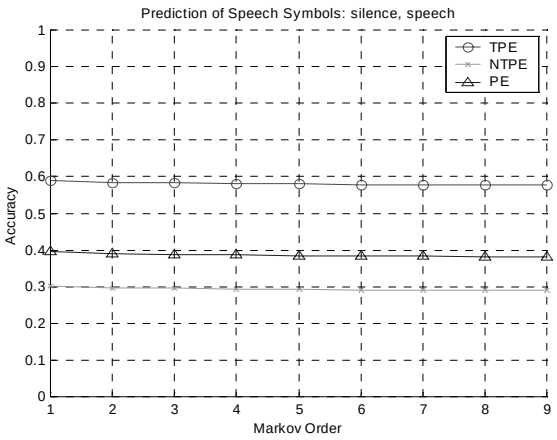
The first set of experiments that we completed use these prediction prefix trees to evaluate our ability to predict the next symbol from a history of symbols, separately in each event modality. We chose 10 consecutive days from the "I Sensed" data set, 9 days for training, 1 day for testing. In order to fairly evaluate the prediction results we further divided the prediction error rate (PE) into two other error rates: transition prediction error rate (TPE) and non-transition prediction error rate (NTPE). The TPE rate only counts errors when the symbol changes from one time step to the next (i.e. on transitions to other symbols). The NTPE rate only counts errors when the symbol is not changing from one time step to the next (i.e. on self-transitions). It was necessary to further specify the prediction error in terms of these rates because it is very common to get deceptively good PE rate but do very poorly on either, but not both, TPE or NTPE. For example, a sequence with lots of repeated symbols in a row could have a low NTPE but high TPE, together yielding a low PE. The relationship between the 3 rates is:

$$PE = \frac{\left(\begin{array}{c}\#\text{ of}\\ \text{transitions}\end{array}\right)TPE + \left(\begin{array}{c}\#\text{ of non-}\\ \text{transitions}\end{array}\right)NTPE}{\text{Length of Sequence}}$$

The following three charts give the error rates of prediction using prediction prefix trees for the visual, motion, and speech symbols/events. While all PE rates are below chance, the rates are not very good. The error rates monotonically decrease with increasing Markov order, as expected, however the error rate bottoms out very quickly. Overall it seems the error rate is quite independent of Markov order, which is clearly indicative of a problem. A possible cause for this problem suggests itself when we look at the prediction results for the 10 days when using the entire 10 days for both training and testing. The problem lies in the fact that our estimation method (frequency counts) is too sensitive to noise and thus doesn't generalize very well. This problem becomes more serious at the higher orders, which could explain the lack of error reduction with the higher order Markov models.







So the next natural step is to smooth or regularize our prefix trees. In order to do this, we need notion of distance between the subsequences, $(s_{t-1},...,s_{t-N})$. This is because counts for one subsequence need to be distributed according to some kernel (e.g. Gaussian) amongst nearby sequences. This represents the uncertainty in the subsequence that was observed. These results are forthcoming.

V. CONCLUSIONS

Improving the Predictions

Compressing these trees allows even higher orders. Work on compressing these trees includes [7] who prunes branches that do not yield additional predictive power (variable order Markov model), and any of the clustering techniques for compressing the space of the subsequences over $(s_{t-1},...,s_{t-N})$.

A promising approach inspired by the Information Bottleneck method [8] is to use the prediction that a particular subsequence yields as a means of defining a distance amongst $(s_{t-1},...,s_{t-N})$. This means that if two subsequences yield similar predictions (or more exactly similar probability distributions over the next symbol) then the distance between them is small. We can make this explicit by defining the following distance measure:

$$D(S_1,S_2) = KL\{P(s_t \mid S_1) \| P(s_t \mid S_2)\}$$

where KL is the KL-divergence measure between two probability distributions, and, S_1 and S_2 are two subsequences from $(s_{t-1},...,s_{t-N})$. Clustering with this distance measure means that clusters will be chosen that try to preserve the predictions of the original uncompressed tree. We can then acccurately represent these clusters with time-inhomogeneous Markov chains (i.e. time-dependent transition tables) that then allow us to do prediction. These results are forthcoming.

This same type of clustering can be used to compress the entire sequence into a shorter sequence or coarser representation. Extracting another prediction prefix tree on this coarser layer and using it to predict the finer layer yields a multi-resolution or hierarchical predictor. Continuing this to coarser and coarser layers can theoretically increase the Markov order of our predictor without the explosion in state-space size. These results are forthcoming.

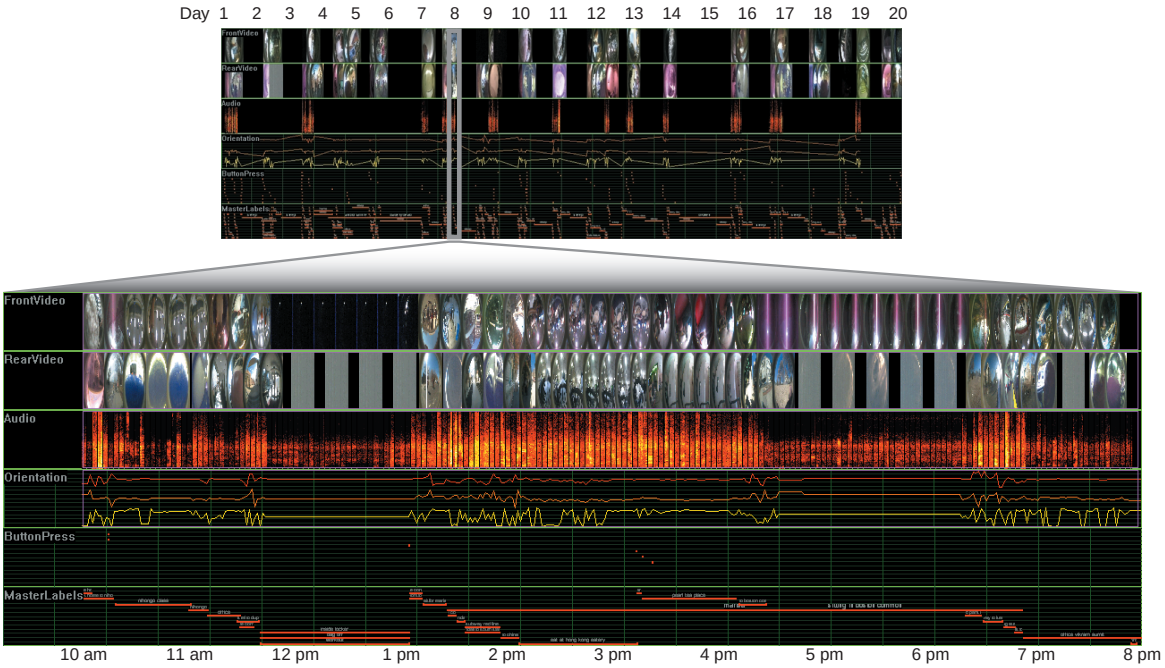


Figure 2: The Data Journal System: provides a multiresolution representation of the time-synchronized sensor data.

VI. BIBLIOGRAPHY

1. Lieberman, H. and Maulsby, D., *Instructible Agents: Software that just keeps getting better*. IBM Systems Journal, 1996. 35(3&4): p. 539-556.

2. Nakamura, Y., Ohde, J.y., and Ohta, Y., *Structuring Personal Activity Records based on Attention -- Analyzing Videos from Head-mounted Camera.* International Conference on Pattern Recognition, 2000. 4: p. 222-225.

3. Clarkson, B., Mase, K., and Pentland, A., *Recognizing User's Context from Wearable Sensor's: Baseline System.* Vismod Technical Report #519, 2000.

4. O'Shaugnessy, D., *Speech Communication: Human and Machine*. 1987: Addison-Wesley.

5. Lucas, B.D. and Kanade, T., *An iterative image registration technique with an application to stereo vision.* Proceedings of the 7th International Joint Conference on Artificial Intelligence, 1981: p. pp. 674--679.

6. Pereira, F.C., Singer, Y., and Tishby, N., *Beyond Word N-Grams.* Computational Linguistics, 1996.

7. Ron, D., Singer, Y., and Tishby, N., *The power of amnesia: learning probabilistic automata with variable memory length.* Kluwer Academic Publishers. Machine Learning, 1996. 25: p. 2-3.

8. Tishby, N., Pereira, F., and Bialek, W., *The Information Bottleneck Method.* The 37th annual Allerton Conference on Communication, Control, and Computing, 1999: p. 10.

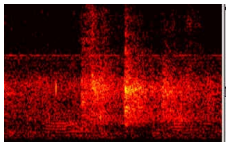
The Data Collection Wearable

Figure 3: The Data Collection Wearable Schematic

Rear View

Orientation

Front View

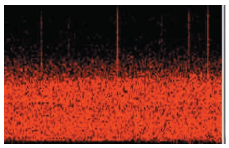


Audio Spectrogram

Scene 1: Eating Lunch

Scene 2: Walking Up Stairs

Front View



Audio Spectrogram

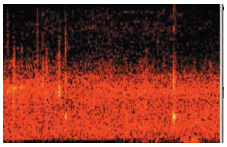
Rear View

Orientation

Rear View

Orientation

Front View

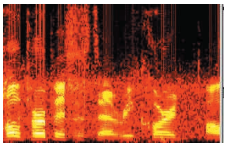


Audio Spectrogram

Scene 3: Rollerblading

Scene 4: In A Conversation

Front View



Audio Spectrogram

Rear View

Orientation

Figure 4: Some excerpts from the "I Sensed" series