Urban Heartbeat: From Modelling to Application

Syed Ali Hasani*, Roozbeh Jafari
Depts. of Biomedical Engineering, *Computer Science and Electrical Engineering
Texas A&M University
College Station, TX, USA
rjafari@tamu.edu

***Abstract*—Sensors and actuators are finding their way into our lives and our surroundings at a very fast pace. These heterogeneous sensors deployed in the environment can prove to be useful in providing insights into the behavior and trends of the environment. In this work, we capture a part of that knowledge and propose a novel concept called Urban Heartbeat using data captured by various sensors that essentially identify periodic activities in the environment. The Urban Heartbeat can be leveraged to identify when an unexpected event has occurred or is about to occur for more effectively preparing the residents and the officials. We first develop techniques to find couplings between sensors using multiple operators, in cases when direct measurement of a parameter is not possible. Next, we define an algorithm that can be used to find quasi-periodic patterns from time series data that has spatiotemporal deviations. We then introduce the concept of Urban Heartbeat, which uses data from heterogeneous sensors to find useful contextual information and trends in the environment. We define the heartbeat as what is normal for the environment. The Urban Heartbeat can be used not only to differentiate between normal and abnormal trends thereby giving us the ability to detect anomalies but also in making predictions about the user or environment behavior with meaningful earliness. We also show how we build heartbeat for a lab environment, and learn useful information about subjects and make predictions about their behavior in the lab**

Keywords—Urban Heartbeat; Pattern Detection; Couplings; Kolmogorov Complexity; IoT; Anomaly Detection

I. INTRODUCTION

With the increasing number of devices in the world of the Internet of Things (IoT), researchers in the IoT community have been looking into how to use the data from these devices to measure, infer and understand the environment around us. These sensors and actuators are quickly becoming a part of our environment. Different models and techniques have been proposed and envisioned about how these devices can be connected together and how they interact [1, 2]. However, understanding how to use the data collected from these devices in more meaningful and efficient ways is crucial. Intelligent use of raw sensor data can exploit the capabilities of a device and turn small sensors into even more powerful devices. For example, extracting location, timing and activity related information from raw sensor data stream, such as RSSI or IMU, can generate a context for a subject.

This work was supported in part by the National Science Foundation, under grants ECCS-1509063, and the TerraSwarm, one of six centers of STARnet, a Semiconductor Research Corporation program sponsored by MARCO and DARPA. Any opinions, findings, conclusions, or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the funding organizations

The vision of IoT outlines billions of devices connected together. They provide opportunities for many applications in healthcare, military, industry, transportation and in creating new interesting systems in the fields of monitoring environment, air quality and urban environments [3]. One key feature of IoT is that it incorporates many different modalities. By fusing all these modalities together, from heterogeneous set of devices, we can generate context that is rich in information. Opportunistic learning and usage of sensor data requires that systems need to be adaptable to the number and type of sensors in the environment [4,5].

Wearable sensors and environmental sensors (*i.e.,* those deployed in the environment) in the framework of IoT have the unique ability to monitor an environment or subjects over a long time period and capture patterns that occur in the environment. Such long-term data collection can be used to learn the behavior of a subject. Using these sensors, normal and abnormal trends can be determined. Therefore, they can be employed in detecting anomalies. The patterns that occur periodically or frequently can be used to make long-term predictions. This, in turn, could be very useful to use resources more efficiently based on prediction.

In this paper, we propose the concept of 'Urban Heartbeat'. We define Urban Heartbeat as knowledge of contextual information that can determine normal and abnormal trends, while also giving us knowledge of periodic behaviors related to an environment or subject. Contextual information is the key to providing task specific information or services to the user [6]. As an example, an office has a heartbeat that can be built by looking at behavioral patterns of employees (*e.g.* people come in at 8am, take lunch break around noon and then leave around 5pm). If this pattern is repeated over a long period, then the behavior can be reflected by the heartbeat for the office. Urban Heartbeat can therefore be employed in detecting anomalies, making predictions with meaningful earliness and in gaining insights and knowledge about behavioral patterns in our environment. This will be useful in learning of new opportunities to advance capabilities of context aware applications. Context aware applications that are intelligent enough to make decisions and improve quality of life is not a new idea, and there has been an interest in the

978-1-5090-6517-2/17/$31.00 ©2017 IEEE

research community to explore this domain [7]. However achieving this goal on an urban scale is a big challenge. Urban Heartbeat can be the approach in this direction.

We envision that Urban Heartbeat can be used to monitor 'Urban Health', which can be key in predicting faults, detecting anomalies, building context, resource utilization, application optimization and in studying the behavior of interesting physical interactions in the physical and cyber world. Urban Heartbeat will also serve as enabler to new means of improving health care for society and cities. For example, while it is easy to monitor air quality to warn patients of asthma, a concept like heartbeat can actually learn what other conditions affects patients while they experience an attack. Is there some specific physical activity always involved? Are there eating habits involved in some way? Such questions can be answered by knowing heartbeat of subject. Urban heartbeat can also give us insights to what is normal for an environment. If something is not normal, what is the best way to react to it? For instance, what does it mean if several subjects have 'elevated' heartbeat at the same time? What actions would help alleviate the situation?

However, finding Urban Heartbeat has its challenges. First, we need to have data over a long time in a real environment to demonstrate this concept. This requires a data collection setup that can last for a prolonged duration, have participants that are committed and a large amount of data that must be handled properly. We then need techniques to look for periodicities in the data to extract periodic patterns and contextual information. The major challenge is that both the periods and periodic patterns in the data are unknown and have spatiotemporal deviations. The patterns that we are interested in may be quasi periodic in nature but also slightly different, hence our technique needs to extract similar patterns. Another important challenge is utilizing data form a heterogeneous set of sensors. The algorithms need to be robust enough to handle data fusion and dynamic environment, in which sensors can come and go from the environment. In this paper, we define the concept of Urban Heartbeat, and present the result from experiment carried out by collecting real data. To solve data fusion, we propose the idea of finding couplings between them. The couplings serve as an abstraction layer above which we do not need to know what type of sensors were used. Urban Heartbeat is a notion that can be built without knowledge of the types of sensors that are present in the environment as long as we can find couplings between them. Above all, we introduce an algorithm to detect periodic patterns that can be used to build the notion of Urban Heartbeat.

Another key challenge is the concern for user privacy [8]. Urban Heartbeat requires collection of large amount of data. The data gathered contains information about the environment as well as use. This can be invasive to user privacy. The research community has introduced concepts like participatory sensing [9]. However, this means that algorithms designed to sense Urban Heartbeat, must be robust enough to handle missing data that the user decides not to share.

II. Related Works

Broadly speaking, our work covers three areas: Determining couplings, finding periodic patterns and their temporal information, and predictions. In this section, we give a brief overview of the research previously done in these respective fields.

A. Couplings

Kuni et al. introduced the concept of using weak classifiers and operators to find interactions between sensors [10]. They created weak operators like covariance operator, peak count, zero crossing etc. and later fused them together to generate a strong classifier. They employ the Adaboost algorithm to generate a strong classifier [11]. We borrow from this concept in finding couplings between sensors. Similarly, Philipose et al. used RFID tags to detect interactions between users and objects [12]. We can find couplings or interactions between sensors using certain criteria or schemes, while employing different sensors.

B. Pattern Discovery/Sequential Pattern Mining

Li et al. have proposed an algorithm 'Periodica', that uses the Fourier transform and auto correlation to find periods and mine periodic behaviors from data [13]. However, they use the concept of reference spots. Elfeky et al. have defined the concept of symbol periodicity and segment periodicity and proposed algorithms based on convolution to find periods [14]. Their work can detect obscure patterns (*i.e.*, unknown patterns). While period detection can be done in $O(nlogn)$ time for a time series of length n, their work uses positions of the symbols in time series to find obscure patterns. WRAP algorithm was proposed based on DTW [15]. The algorithm is robust and can work even when there is noise in time series data while finding periodicities. Researchers typically focus on developing an algorithm to find potential periods in general [16, 17] or an algorithm for finding periods and patterns for a given problem. Ma et al. have worked on detecting periods that are quasi-periodic [18]. However, their method of using chi-squared test requires an expected value of occurrence for a particular behavior. This puts a constraint on where this technique can be used. For instance, while it is predictable that a person would take lunch break, how many breaks in total he takes during a day is a hard question to answer, and thus a threshold for chi-squared test is hard to determine while looking for such patterns.

Another area related to finding the patterns is employing data mining techniques. Agarwal et al. introduced the concept of sequential data mining (SPM) [19] and now there are several algorithms for mining sequential patterns [20, 21]. The main challenge while doing this is multiple scans of the database, which increases the time in finding these patterns while also requiring multiple accesses to memory where data is stored. In this paper, the proposed algorithm requires only one scan of the database.

C. Predictions

Quasi-Periodic patterns can be used to make predictions about some activity related to those patterns or infer a user's end goal (*i.e.,* intention). Jeung et al. used periodic patterns to predict future movements for moving objects [3]. Similarly, by studying the migration patterns of birds one can predict where they will be in a particular season and learn the migration trends [13]. There has been an increasing interest in predicting human activity. To the best of our knowledge, most of the work in this field has been done through computer vision with use of cameras. Li et al have used a probabilistic suffix tree along with context to predict complex activities using a camera [22]. Similarly, by using video streams, human activity prediction was done by Ryoo [23]. The work of Pei et al. about goal inference and intent prediction is another such example that is based on video parsing. While using cameras and computer vision techniques, to do predictions, is effective, it definitely raises privacy concerns [24]. Use of a camera to monitor activities in an environment for day-to-day activities may be invasive to a user's privacy. Our work is based around sensors of different types and modalities. The notion of Heartbeat can include motion sensors, proximity sensors, pressure sensors and IR sensors, which are less invasive to the user's privacy. We would like to make a point that Urban Heartbeat is developed to work irrespective of type of sensors. Therefore, video and audio can be used for detecting heartbeat as well.

III. Complexity of Time Series

Our approach to heartbeat uses the notion of complexity of time series. Therefore, we find it important to introduce complexity first. The complexity of a time series can characterize its nature and characteristics. It captures transitions, peaks and valleys in time series data; and it can help us distinguish between one series and another, however as Kozarzewski notes, there is no formal definition of complexity of time series [25]. There is also no agreed way of quantifying complexity of time series. There are several complexity measures including Kolmogorov complexity [26], variants of entropy [27, 28], and complexity invariant distance [29]. Each estimates how complex the time series is. The basic idea is that similarly complex series' will have the same complexity.

In our work, we use Kolmogorov complexity. Kolmogorov complexity as described by Kaspar et al. [26] works by building a vocabulary of what makes up a time series. Any new addition to the vocabulary results in increase of complexity. Another definition of Kolmogorov complexity can be given as: If s is a time series then if a description of s, $d(s)$, is of minimal length (*i.e.* it uses the fewest bits), it is called a minimal description of s. Thus, the length of $d(s)$ is the Kolmogorov complexity of s, written $K(s)$. Symbolically

$$\boldsymbol{K}(\boldsymbol{s}) = |\boldsymbol{d}(\boldsymbol{s})| \quad (1)$$

For example, the string '*abababababababab'* can be described as "ab 8 times" which is a string of 10 characters. It follows from this that if we are looking for similar patterns in a time series, then if a pattern repeats itself, the increase in complexity due to those similar patterns in the time series will be zero. In our work, we exploit this fact as described in what follows in the paper.

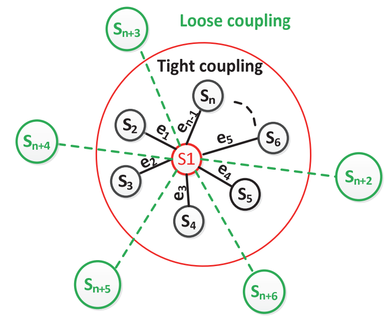

Figure 1 Couplings between S1 and other Sensors

IV. Problem Formualtion

Urban Heartbeat is defined as the set of periodic patterns and their contextual information in a given environment. It can help us predict failures, monitor changes in the urban environment, notice unusual activity and provide information on opportunities for optimizing the environment by knowing context. We believe that such information can be extracted from looking at data from sensors over time. Hence, our problem can be defined as

Analyze the time series obtained from multiple sensors to extract quasi-periodic patterns of couplings between sensors. Analyze the extracted patterns and group similar patterns together to gain useful information and build notion of heartbeat.

A. Couplings

At any given instant of time, t, a sensor may be interacting with any other sensor in the environment. The interactions can be of any type, and different modalities, either individually or collectively, can help us find them. As an example, if two users with wearable sensors on their wrists shake hands, an interaction can be detected by virtue of similar motion and proximity. Similarly, if a user sits on a chair, and the user and chair both have sensors, we can find this interaction or activity based on proximity of the person with the chair, the motion of the person and the chair, or pressure on the chair. Fig. 1 depicts the notion of couplings. $S1$ is the sensor of interest.. There are n sensors around $S1$. Edges between $S1$ and other sensors have weights that indicate coupling level. The weights can be binary *i.e.* either the sensors are coupled or they are not coupled. Tight couplings may be differentiated from loose couplings by noticing which sensors couple more frequently.

To model these interactions in time, we consider a graph having all the sensors as vertices.

Definition 1: if a graph G represents interactions between sensors in an environment at time instant t*, then set V representing vertices of G contains all of the sensors, and we have an edge between* $v \in G.V$ *and* $u \in G.V$ *if there is an interaction between sensor represented by u and sensor represented by v.*

Definition 2: If in graph G, there is an edge between v ∈ G.V and u ∈ G.V, sensor represented by u and sensor represented by v are coupled at time instant t.

To find couplings we can create simple operators. They can operate on data from different modalities and find couplings. As an example, we explain an operator in detail here, and use the Received Signal Strength Indicator (RSSI) data to demonstrate our concept. RSSI can indicate if two sensors, often mobile, are within close proximity of each other. The operator that we use in this case is a proximity operator, based on a threshold classifier.

The Proximity operator looks at the RSSI data and outputs if a sensor is coupled with any other sensor. More formally, let *X* and *Y* represent the time series of observations from two sensors.

$$X = x_1, x_2, \dots x_n \qquad n \in \mathbb{N} \tag{2}$$
$$Y = y_1, y_2 \dots y_m \qquad m \in \mathbb{N} \tag{3}$$

We define a proximity operator as:

$$c(x_k, y_k) = \begin{cases} 1, & < \eta \ and \ 1 < k < n \\ 0, & otherwise \end{cases} \tag{4}$$

Where x_k and y_k are the observations of the two sensors at time instant *k*. η is the threshold that can be set based on application. In case of proximity operator, if the RSSI is below threshold, there is no coupling between two sensors as they are far apart. The threshold can vary, based on the pair of sensors we are looking at.

Another operator that can operate on data from accelerometer and gyroscope can be a covariance operator. For the covariance operator, we find the covariance matrix of two data streams from two Inertial Measurement Unit (IMU) sensors and then based on a threshold classifier, classify them as coupled or not coupled. The covariance operate indicates if two sensors are moving *together* or not, hence offering insights into the physical coupling of sensors. For example, when two users are shaking hands, their wrist worn sensors are moving together observable by the covariance operator.

B. Interval Graphs

The couplings are modelled as interval graphs over time. An interval graph, as shown in Fig. 2, gives us a time series of coupling levels, in our case, indicating when there was a coupling. In Fig 2, a solid line indicates when there was a coupling. An absence of solid line in the graph indicates no coupling intervals. Interval graphs additionally give us the temporal characteristics of the couplings, from which it is easier to extract periodic patterns of interests and periods for these patterns. We can define *I* as the interval graph given by:

$$\boldsymbol{I(X,Y) = c(x_1, y_1) \dots c(x_n, y_n) \qquad 1 < n < \infty} \tag{5}$$

Where *X* and *Y* are the two sensors.

C. Extracting Quasi-Periodic Patterns

Without loss of generality, we refer to $I(X,Y)$ as *I*, and $c(x_k, y_k)$ as C_k. Let $T = \{T_1, T_2, \dots T_p\}$ be a set of subsets of *I*, each of which repeat itself in *I* and *p* is the number of elements in T . These are the candidate periodic patterns. To extract these patterns complexity of a time series is measured. Similar

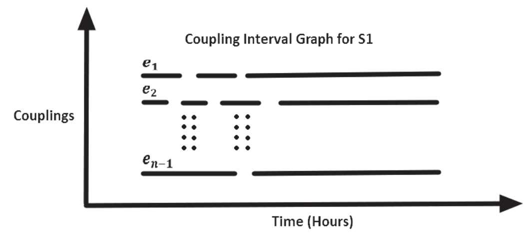

Figure 2 Sample interval graph between S1 and other sensors

patterns have the same complexity or in other words, adding a pattern to a similar pattern will have little increase in complexity. Hence, if we start from the beginning and keep adding one element of the time series at a time and measure complexity at every instant, we can get the complexity over time. This trend can show us where similar patterns appear. Adding a similar pattern to the time series will result in no increase in the complexity, therefore the complexity curve will appear flat at those points. This feature is exploited to extract the patterns. Let us denote the complexity of any series *S* by *Complexity(S)*.

Let S_k be a subset of *I*

$$S_K = C_1, C_2, \dots C_q \qquad 1 < q < n \tag{6}$$

Then to find the complexity trend of this series

$$\boldsymbol{Trend = \{Complexity_m(S_m)\} \qquad m = 1, 2, 3, \dots} \tag{7}$$

To extract candidate patterns we can use the slope of the curve. Each flat section of curve would give one instance of quasi-periodic pattern, and hence we can find them by:

$$TT_q = \min\big(diff(Trend)\big) \qquad 1 < q < p \tag{8}$$

Where there are *p* minimums in the trend curve.

D. Heartbeat

The notion of Urban Heartbeat is being defined as knowledge of quasi-periodic patterns and their temporal characteristics that can be used to gain useful contextual information. It is very important that we not only extract each instance of the periodic patterns but also group them based on their characteristics and temporal properties do learn context. This will help us detect a normal heartbeat over time and detect anomalies. We employ hierarchal clustering to do this grouping. The distance measure used to do the clustering should take into account the similarity of two patterns. Similarity can be accounted for based on complexity of the series. The distance measure should also take into account the temporal characteristics of the pattern. We take into account two temporal characteristics, the difference in duration of two patterns, and the difference in occurrence time. Let us define two patterns as:

$$T_a = \{C_a, \dots C_i\} \qquad 1 < a < i, a < i < n \tag{9}$$
$$T_b = \{C_b, \dots C_j\} \qquad 1 < b < j, b < j < n \tag{10}$$

As the similarity-metric of two patterns T_a and T_b, let *K* be the difference in complexities of T_a and (T_a *U* T_b). Here $T_a \cup T_b$ is the concatenation of T_a and T_b.Then

$$K = |complexity(T_a) - complexity(T_a \cup T_b)| \quad (11)$$

Let Δ represent the difference in duration of two patterns. The start and end times are given by subscript $(a, b), (i, j)$ respectively. Hence

$$\Delta = |(i - a) - (j - b)| \quad (12)$$

and τ represents the difference in occurrence time or start of a particular pattern in the day.

$$\tau = |b' - a| \quad (13)$$

Where b' is b normalized to 24 hour period. This means that if we have several days of data, we segment the data by the day and consider time of the day in hours. We can now consider K, Δ, and τ as a three dimensional space. In the ideal case, when there is a perfect match between two patterns, the Euclidian distance will be 0. We use this Euclidian distance as the distance metric.

$$\boldsymbol{d} = \sqrt{\boldsymbol{K^2} + \boldsymbol{\tau^2} + \boldsymbol{\Delta^2}} \quad (14)$$

d is used as measure of distance in hierarchical clustering which will be explained later.

The objective of the hierarchical clustering is to group instances of quasi-periodic patterns together and identify information about their characteristics. We define the objective function as following.

Let T_i and T_j be any two candidate patterns from T, and t_{x_i}, t_{y_i}be the start and end time of T_i, and t_{x_j} and ,t_{y_j} be the start and end time of T_j. Hierarchal clustering can then be implemented based on the following objective function:

$$\mathbf{arg}\,\boldsymbol{min} \sum_{i,j} \boldsymbol{d}(\boldsymbol{T_i}, \boldsymbol{T_j}) \quad (15)$$

$$\boldsymbol{t_{es}} \leq \boldsymbol{t_{xi}} \leq \boldsymbol{t_{ls}} \quad (16)$$

$$\boldsymbol{t_{ee}} \leq \boldsymbol{t_{xj}} \leq \boldsymbol{t_{le}} \quad (17)$$

$$\boldsymbol{t_{es}} \leq \boldsymbol{t_{yi}} \leq \boldsymbol{t_{ls}} \quad (18)$$

$$\boldsymbol{t_{ee}} \leq \boldsymbol{t_{yi}} \leq \boldsymbol{t_{le}} \quad (19)$$

$t_{es}, t_{ls}, t_{ee}, t_{le}$ are the earliest start time, latest start time, earliest end time and latest end times for the given group. $d(T_i, T_j)$ is the similarity distance between two patterns under consideration to determine if they are instances with similar behavioral pattern.

V. Experimental Setup

A. Sensor

The experiments use custom sensor nodes called MotionNet that have been designed and developed by an academic lab. The processing element of the sensors is an ARM Cortex M3 based CC2538 chip from Texas Instruments and the sensor is capable of communicating over Bluetooth and Zigbee [30]. Data can also be stored locally on a micro SD card. The sensor nodes also have an InvenSense MPU9150, which is a 9-axis IMU that provides accelerometer, gyroscope and magnetometer data [22]. Fig. 3 shows the sensor. We use the Contiki OS on the sensors to collect data [31]. Contiki OS enables ZigBee, which we use for communication between sensors.

Figure 3 MotionNet

B. Setup

We created an experimental setup in our lab, and our lab members volunteered to be the users. The overall setup of the experiment, involves several sensors, including i) Server, ii) Beacons, and iii) Wearables.

The server node allocates time to each sensor to communicate with other in the environment. We have five beacons fixed in the environment, sending out one packet/second. There are four users in the lab each carrying a set of two sensors. One of the two sensors is worn on the wrist and is called ‘wrist sensor’, while the second one is kept in pocket at all times, and is called the ‘pocket sensor’. All sensors collect IMU data as well as RSSI values for every other sensor in the environment and store the data on a micro SD card. IMU data is captured at 40Hz, and RSSI values from all other sensors are gathered once every second. Our experiment is approved by the IRB.

C. Data Collection

Data is collected and stored locally on the sensors on a micro SD card. The data stored consists of observations. We make three kinds of observations, which are: i) RSSI reading, ii) IMU Sensor reading, iii) UTC Time

The observations in our case can be a sensor reading or timing information. Each observation is stored as soon as it is received. All of this data is stored in a binary file and is parsed in MATLAB for processing. Our system collects data that is a discrete time series of sensor readings enabling us to look at the time series to determine patterns and learn temporal characteristics of the patterns. To keep the data well synchronized, the sensors receive UTC time from the server every 20 seconds.

The sensors also have two local clocks. One is based on a 32 kHz crystal oscillator, whereas the other one is based on 32 MHz clock source. With every observation, we also have these two time stamps.

D. Gold Standard

A camera with a fisheye lens is placed in the lab to record video of the environment. We can use the video to verify where a subject was in the lab, as well as what they were doing to verify our observations from the data. We employ image-tracking algorithms to determine the subject's position at any time, and use this as a gold standard. We also use the video frames and sequences to verify results for activity recognition or couplings based on motion. After using a frame, the frame is blurred to deidentify the users.

VI. RESULTS

A. Heartbeat

We collected extensive data to build a heartbeat for subjects in the lab. We consider data between every pair of sensors in the environment and employ our technique to build the notion of heartbeat. We use 14 days of data as training data to extract a pattern and cluster it with similar instances, to get an interesting insight about the environment. Each day's data is for 24 hours. We focus on an application that targets prediction of human activity and detecting anomalies in the behavior. The application and results are merely to demonstrate how simple applications can leverage this novel concept of heartbeat. We believe that sophisticated and powerful applications can be built based on this concept, but we will leave the application space exploration for future work.

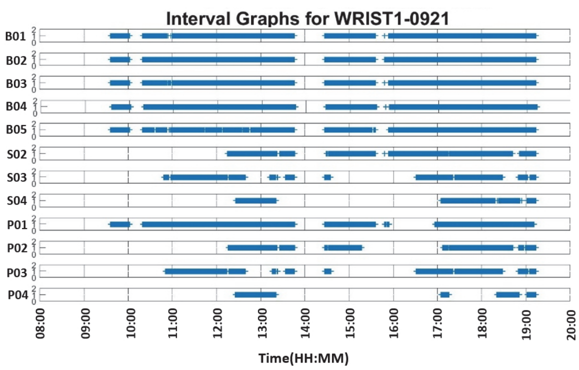

Figure 4 Interval Graphs for a user with all other sensors

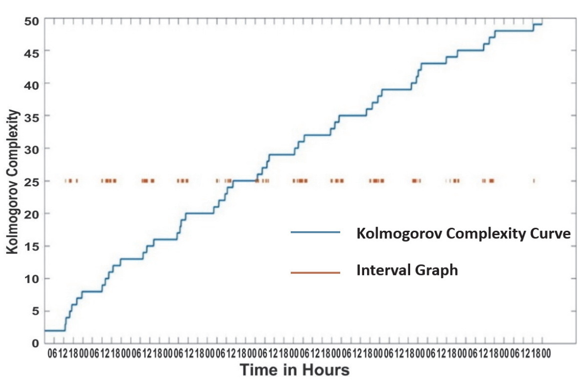

Figure 5 Kolmogorov Complexity Curve for a given Interval Graph

The data collected from several sensors is processed to generate interval graphs. Fig 4 shows the interval graph for a user with respect to other sensors in the environment. The x-axis is the time in hours from 8 am to 10 pm. Fig 4 actually has 12 plots one for each of the 12 sensors around it. The sensors include 4 beacon nodes on the walls of the lab area, the subject's own pocket sensor, and the pocket sensors and wrist worn sensors of the three other users in the environment. The solid lines represent the time when the user was in the lab and interacting with the sensors. A region where there is no solid line is when there was no interaction (*i.e.,* the two sensors are not within close proximity).

To extract the candidate patterns from a series of these graphs containing several days of data, we use our proposed technique leveraging the complexity curve. All candidate patterns are assumed to be at least 10 minutes long. While we created the heartbeat for three users in the lab, we selected one of them to demonstrate the effectiveness systematically. Fig. 5 shows the Kolmogorov complexity curve for 14 days of data (in blue) and its associated interval graph (in red). The interval graph is for one of the wrist sensors and beacon 1 inside the lab. Note that in our work, we consider an interval of 10 minutes to be the minimum acceptable duration and hence down sample the data accordingly using an averaging filter. There is a sample for each second and we combine data of 10 minutes by averaging data over that period. Although a 10-minute duration was selected heuristically, it appears to offer a reasonable time granularity for presence of users in the lab. This also helps in reducing the processing time while generating the complexity trend curve and extracting candidate patterns. The complexity curve becomes flat at various intervals. These are the points where any pattern in the data set repeats itself. Using the temporal knowledge and the index of time series, we can extract the patters in the data set. We extracted the patterns and used hierarchal clustering to group similar patterns together.

Fig. 6 shows the constructed clusters. The clusters group instances of similar behavior together. We note that several clusters are formed for the users and by comparing it to interval graphs and looking back at our video gold standard, we observed that each cluster represents a behavior for a part of the day. These can be used for prediction as well as anomaly detection. The results for those follow, but first we explain the clusters formed in Fig. 6. In our case from out training data, 48 patterns that were candidate instances of similar behaviors were extracted. All of these are clustered based on similarity in terms of complexity and temporal characteristics as shown in Equations (14) and (15). The patterns are grouped into 5 clusters. We notice that these patterns represent a person's daily routine (*i.e.* the time they arrive in lab, the night time when they are absent from the lab, the time they take a lunch break around 12:30pm, dinner break around 6:30pm and leaving the lab around 7:30pm

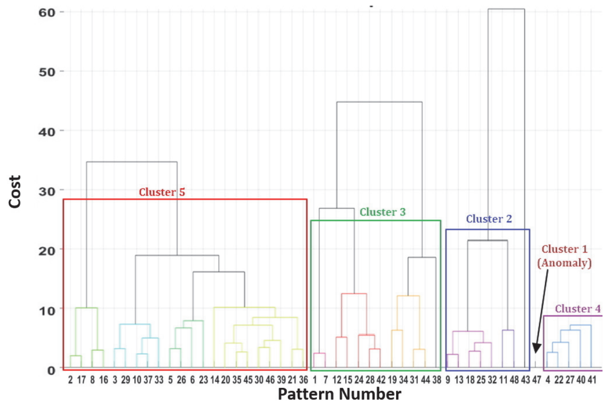

Figure 6 Hierarchical Clustering

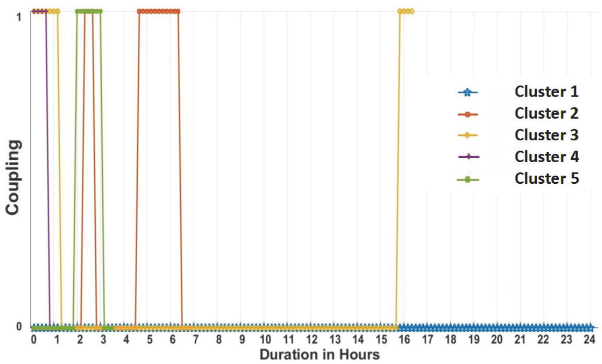

Figure 7 Representative Templates for each Cluster Group

every day). These can be best represented by obtaining the representative patterns for each cluster. Fig. 7 shows the representative patterns. The representative patterns are obtained by measuring the summation of distance between one pattern and every other pattern in the cluster. The pattern with the least distance is picked as the representative pattern for that cluster. Hence, Fig. 7 shows the patterns that best represent each cluster. Table 1 lists the cluster group, time of occurrence and behavior represented. The intervals between 1200hrs and 1300hrs may be represented in two different clusters because of slightly different patterns in lunch breaks of the user – 'Morning (1)' and 'Morning (2)'. In some cases, they came late to the lab, while in other cases the user came early, worked for a couple of hours and then went for lunch. We also note that Cluster 4 represents when the person took

Table 1 Summary of Clusters and Behavioral Representation

Cluster	Time	Behavior Label
1	-	Anomaly
2	12-6pm	Morning(1)
3	7pm-12am	Night
	12am-(+1)11am	Past Midnight
4	6-7pm	Dinner Break
5	10-6pm	Morning (2)

breaks between 1700hrs and 1800hrs, this did not happen regularly and hence a separate cluster was constructed for these.

B. *Prediction and Anomaly Detection*

We notice that cluster 1 has only one pattern. This is an anomaly, and something that happened only once in two weeks, hence it was grouped in a separate cluster. By looking at our video data, we were able to verify that during this data collection, the person was in the lab, however they did not wear the sensor properly, hence causing the anomaly. Because this was something different from their normal heartbeat, we were able to identify the anomalous behavior.

While cluster 1 represents a clear anomaly, heartbeat also captures behaviors that are likely to occur with slight variation. Hence, it is possible to use this insight to learn a behavior for long duration, which can be termed as heartbeat. It is possible to predict the behavior within an environment, and capture deviations and variations to find anomalies. From the experiment, different behavioral patterns were learned. Using the representative templates and temporal knowledge of each of the, we built a simple prediction mechanism. The prediction predicts the behavior at any time, based on representative templates. We used data collected in month of December and January for training. For testing the prediction, we use 10 days of November and 10 days of February. Moreover, we did not use Cluster 1 in prediction as that was an anomaly. Furthermore, patterns that roll over to the next day are broken into two parts, at midnight.

In our data, we have patterns that represent three particular parts of the day. The work routine including lunch break. The evening routine, which sometimes included dinner break, and the past midnight pattern. The results are shown in Fig. 8. For parts of the day with two possible patterns, we observed that error for one of the patterns would be minimum, unless there was an anomaly. Using this we can detect further anomalies or 'elevated heartbeat'. We observe two spikes in past midnight part. These were caused because of overall deviation from the daily routine. In one case, the user stayed out of the lab throughout the day, causing a similar pattern as that of past midnight, to emerge earlier. In the other case, the user left the lab early. Similarly, in the training data, we learned several instances where the user would take a dinner

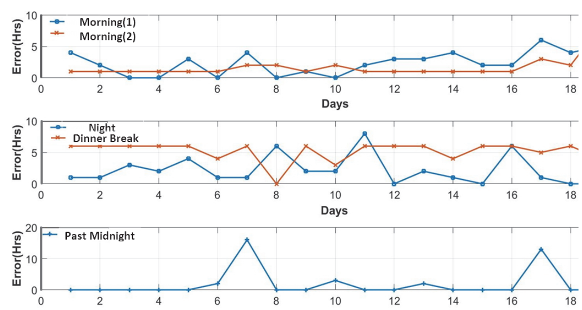

Figure 8 Prediction Error and Anomaly Detection

break, come back to lab and then leave. However, in case of test data, this did not happen and dinner break was itself an anomaly for the user. We also observed at least three spikes in night routine that were caused by user staying longer in the lab then routine. For the morning part, at least one of the two patterns usually happened and we observed one big spike, caused by the user by staying out of the lab for the day.

VII. Future Work And Conclusion

In this paper, we introduced a novel concept of Urban Heartbeat that can be constructed leveraging data from devices in the environment. Urban Heartbeat captures the contextual information about patterns and behaviors that occur regularly in the environment.

Plans for the future study include development of a network of heterogeneous sensors that can be used to build a more robust heartbeat for a variety of environmental paradigms. The predictions can also be used to optimize the environment and provide a variety of services. We plan to work on understanding different heartbeats and not only learn what it means to have a normal heartbeat versus abnormal heartbeat but also how to react to it.

References

[1] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of things (iot): A vision, architectural elements, and future directions," *Future Generation Computer Systems*, vol. 29, no. 7, pp. 1645–1660, 2013.

[2] E. A. Lee, B. Hartmann, J. Kubiatowicz, T. S. Rosing, J. Wawrzynek, D. Wessel, J. M. Rabaey, K. Pister, A. L. Sangiovanni-Vincentelli, S. A. Seshia *et al.*, "The swarm at the edge of the cloud." *IEEE Design & Test*, vol. 31, no. 3, pp. 8–20, 2014.

[3] H. Jeung, Q. Liu, H. T. Shen, and X. Zhou, "A hybrid prediction model for moving objects," in *Data Engineering, 2008. ICDE 2008. IEEE 24th International Conference on*. Ieee, 2008, pp. 70–79.

[4] D. Roggen, G. Troester, P. Lukowicz, A. Ferscha, J. d. R. Millán, and R. Chavarriaga, "Opportunistic human activity and context recognition," *Computer*, vol. 46, no. 2, pp. 36–45, 2013.

[5] B. Guo, D. Zhang, Z. Wang, Z. Yu, and X. Zhou, "Opportunistic iot: exploring the harmonious interaction between human and the internet of things," *Journal of Network and Computer Applications*, vol. 36, no. 6, pp. 1531–1539, 2013.

[6] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos, "Context aware computing for the internet of things: A survey," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 1, pp. 414–454, 2014.

[7] Ö. Yürür, C. H. Liu, Z. Sheng, V. C. Leung, W. Moreno, and K. K. Leung, "Context-awareness for mobile sensing: A survey and future directions," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 1, pp. 68–93, 2016.

[8] D. Christin, "Privacy in mobile participatory sensing: Current trends and future challenges," *Journal of Systems and Software*, vol. 116, pp. 57–68, 2016.

[9] F. Restuccia, S. K. Das, and J. Payton, "Incentive mechanisms for participatory sensing: Survey and research challenges," *ACM Transactions on Sensor Networks (TOSN)*, vol. 12, no. 2, p. 13, 2016.

[10] R. Kuni, Y. Prathivadi, J. Wu, T. R. Bennett, and R. Jafari, "Exploration of interactions detectable by wearable imu sensors," in *Wearable and Implantable Body Sensor Networks (BSN), 2015 IEEE 12th International Conference on*. IEEE, 2015, pp. 1–6.

[11] R. E. Schapire, "Explaining adaboost," in *Empirical inference*. Springer, 2013, pp. 37–52.

[12] M. Philipose, K. P. Fishkin, M. Perkowitz, D. J. Patterson, D. Fox, H. Kautz, and D. Hähnel, "Inferring activities from interactions with objects," *Pervasive Computing, IEEE*, vol. 3, no. 4, pp. 50–57, 2004.

[13] Z. Li, B. Ding, J. Han, R. Kays, and P. Nye, "Mining periodic behaviors for moving objects," in *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2010, pp. 1099–1108.

[14] M. G. Elfeky, W. G. Aref, and A. K. Elmagarmid, "Periodicity detection in time series databases," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 17, no. 7, pp. 875–887, 2005.

[15] ——, "Warp: time warping for periodicity detection," in *Data Mining, Fifth IEEE International Conference on*. IEEE, 2005, pp. 8–pp.

[16] P. Indyk, N. Koudas, and S. Muthukrishnan, "Identifying representative trends in massive time series data sets using sketches." in *VLDB*, 2000, pp. 363–372.

[17] C. Berberidis, W. G. Aref, M. Atallah, I. Vlahavas, A. K. Elmagarmid *et al.*, "Multiple and partial periodicity mining in time series databases," in *ECAI*, vol. 2, 2002, pp. 370–374.

[18] S. Ma and J. L. Hellerstein, "Mining partially periodic event patterns with unknown periods," in *Data Engineering, 2001. Proceedings. 17th International Conference on*. IEEE, 2001, pp. 205–214.

[19] R. Agrawal and R. Srikant, "Mining sequential patterns," in *Data Engineering, 1995. Proceedings of the Eleventh International Conference on*. IEEE, 1995, pp. 3–14.

[20] J. Pei, J. Han, and W. Wang, "Constraint-based sequential pattern mining: the pattern-growth methods," *Journal of Intelligent Information Systems*, vol. 28, no. 2, pp. 133–160, 2007.

[21] X. Yan, J. Han, and R. Afshar, "Clospan: Mining closed sequential patterns in large datasets," in *In SDM*. SIAM, 2003, pp. 166–177.

[22] K. Li and Y. Fu, "Prediction of human activity by discovering temporal sequence patterns," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 36, no. 8, pp. 1644–1657, 2014.

[23] M. Ryoo, "Human activity prediction: Early recognition of ongoing activities from streaming videos," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 1036–1043.

[24] M. Pei, Y. Jia, and S.-C. Zhu, "Parsing video events with goal inference and intent prediction," in *Computer vision (iccv), 2011 ieee international conference on*. IEEE, 2011, pp. 487–494.

[25] B. Kozarzewski and H. Sucharskiego, "Multilevel time series complexity," *Journal of Applied Computer Science*, vol. 19, no. 2, pp. 61–71, 2011.

[26] F. Kaspar and H. Schuster, "Easily calculable measure for the complexity of spatiotemporal patterns," *Physical Review A*, vol. 36, no. 2, p. 842, 1987.

[27] S. Gonzalez Andino, R. Grave de Peralta Menendez, G. Thut, L. Spinelli, O. Blanke, C. Michel, and T. Landis, "Measuring the complexity of time series: an application to neurophysiological signals," *Human brain mapping*, vol. 11, no. 1, pp. 46–57, 2000.

[28] W. Aziz and M. Arif, "Complexity analysis of stride interval time series by threshold dependent symbolic entropy," *European journal of applied physiology*, vol. 98, no. 1, pp. 30–40, 2006.

[29] G. E. Batista, X. Wang, and E. J. Keogh, "A complexity-invariant distance measure for time series." in *SDM*, vol. 11. SIAM, 2011, pp. 699–710.

[30] Z. Alliance *et al.*, "Zigbee specification," 2006.

[31] A. Dunkels, O. Schmidt, N. Finne, J. Eriksson, F. Österlind, and N. T. M. Durvy, "The contiki os: The operating system for the internet of things," *Online], at http://www. contikios. org*, 2011.