Spatial-Temporal Anomaly Detection using Security Visual Analytics via Entropy Graph and Eigen Matrix

Matthew Sinda and Qi Liao
Department of Computer Science
Central Michigan University
Mount Pleasant, Michigan 48859
Email: {sinda1m, liao1q}@cmich.edu

***Abstract*—Much of the big data which is produced is due to IoT devices and various sensor networks. This data often comes with spatial as well as temporal properties that can tell investigators many things about the environment in which they are located. For security practitioners, how to find abnormal activities or anomalies in the vast amount of spatial-temporal dynamic data is a daunting task. We present a system, STAnD, to assist investigators in determining patterns within these spatial-temporal data sets. The analysis conducted by using this program can support correlating events in both the spatial and temporal domains which will lead the investigators to determine probable causes for potential malicious events.**

***Index Terms*—spatial-temporal anomaly detection, security visualization, big data analytics, entropy graph, eigenvectors**

I. Introduction

The data explosion we are currently in is being caused by the inclusion of the Internet of Things (IoT) domain into our daily lives as well as the proliferation of social media, which has been shown can be used as mobile sensors. It is estimated that of all the data generated, 90% has been produced in the last two years [1]. These IoT devices and social media sensors provide a level of convenience to us but can also be used to collect and deliver characteristics on the environment they are placed. This generated data can take the form of spatial data, having a geographical component to it, temporal data, entries that vary across time, or both. These devices act as sources which generate data entries as often as several thousand per second or more, to generating data entries only when triggered. As this data grows and becomes more complex, it will become more challenging for investigators to gain some level of intuition on the data using traditional means.

While there has been research on spatial-temporal anomaly detection [2]–[9], there is a lack of a user interface to correlate events from both the temporal and spatial dimensions in order to find the underlying causes of malicious activities. This work focuses on improving the ability to derive connections between spatial and temporal events as well as allowing the user to categorize incidents as not only abnormal, but to draw conclusions about suspected anomalies and determine if they are malicious. We developed a tool with methods aimed at the identification of anomalies in a dataset such as the one described and to assist in the rapid classification of events. These events are classified through the use of entropy calculations in spatial data and community detection algorithms as a means to assist in the visual identification. Entropy is particularly useful to determining spatial data outliers as it allows a simplistic manner to compare one item's movements with another within a given region. The specifics behind how this is accomplished will be illustrated in the sections that follow. In addition, physical movement graphs were designed to model movements recorded by either stationary or mobile sensors and to further visualize the underlying entropy data.

The temporal data is classified through eigenvector calculations to help narrow the scope of an investigation in the spatial dataset. Eigenvectors allow for many data attributes to be compared against each other in an efficient manner and is one of the primary reasons for inclusion in this work. Additionally, due to how eigenvectors are calculated, they allow for efficient identification of anomalous data. Eigenvector matrices and charts, sensor reading overlays and correlation matrix were also designed to visualize the temporal evolution of sensor data. As will be shown, using multiple methods allows for each view to mitigate the shortcomings of other views, and shows the data to the investigator from a different perspective.

The remainder of this paper is organized as follows: section II discusses related works as they pertain to spatial and temporal anomaly detection, section III discusses our methods for detecting anomalies, section IV applies our methods and visualization tools to a case study, and section V concludes our work.

II. Related Work

Spatial-temporal anomaly detection is an important research topic and has many applications. There has been either algorithmic or visual approaches to identifying anomalies in the corresponding data. For example, researchers have focused on a cluster centric approach [2] by utilizing the fuzzy c-means clustering algorithm to place events into similar groupings. To overcome difficulties in spatial-temporal clustering a sliding time window has been proposed to assign an anomaly score based on past behavior of an item. In addition, Bayesian networks have been utilized to identify anomalies in sensor data [3], [4] in order to assist in filtering false positives in data returned from sensors. Bayesian networks have also been used to identify air quality

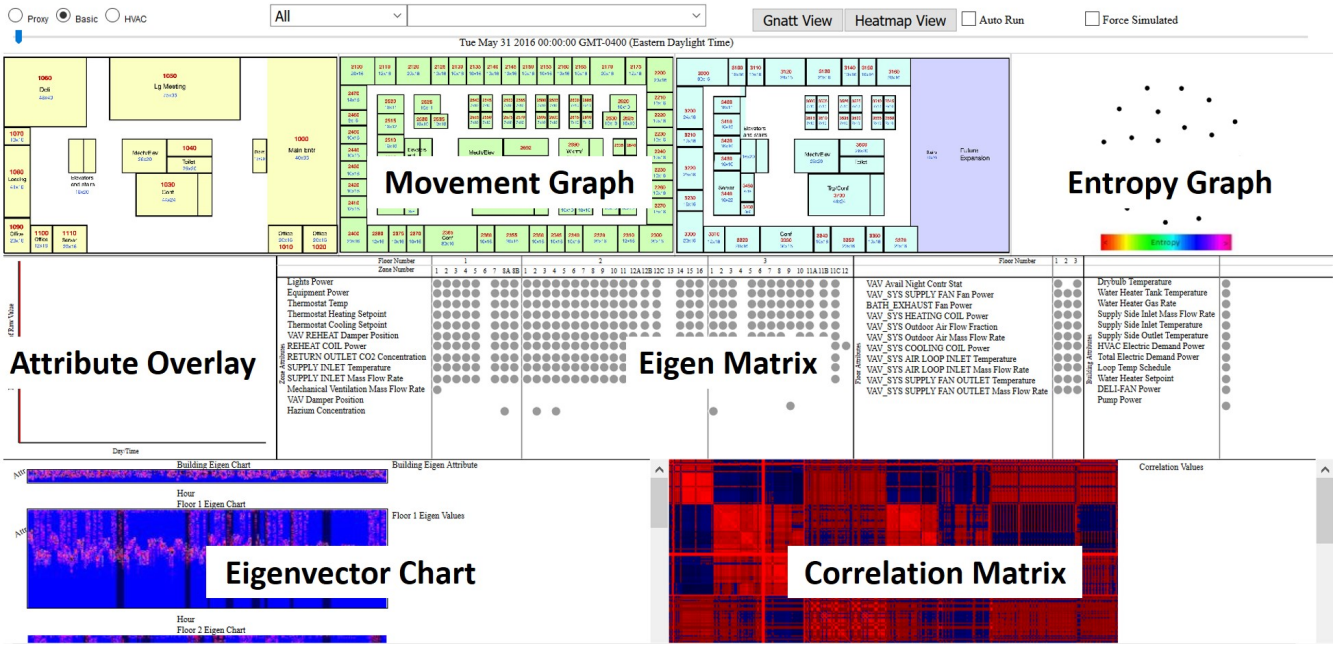


Fig. 1: Overview of Spatial-Temporal ANomaly Detection (STAnD) tool.

sensors that are in a failing state [3] and trends in gas concentration values over time [4].

It has been shown that entropy can be used to find patterns over a user's web log [10]. By conducting the entropy calculations over this data, a degree of symmetry in the connections on the computer being analyzed can be acquired. The conclusion is the more balanced the distribution of connections, the higher the entropy values. They pair this entropy calculation with clustering to visualize the pattern in types of connections the user creates. Entropy is also utilized to assist in the security of e-commerce by combining information entropy with a neural network solution to identify abnormal traffic in a mobile payment system [11].

Eigenvectors may be used to determine events in a mobile communication graph [12]. Abnormalities are identified when a node's eigenvector value deviates from a determined normal value. Through their tool, they were able to correctly identify interesting patterns that matched actual events. A dependency matrix and the eigenvectors were used as a natural way to perform feature extraction and suggest attributes which the user should consider interesting [13].

There have been methods used for visualization spatial and temporal data [14]. Many of the tools they discuss, show the data in varying levels of detail. By providing an interface, the work conducted in [15] created a tool that users can utilize to drill down on details of various data, they provide a tool which allows individuals to gather information on possible anomalies by starting investigations in the most likely area of an anomaly. An interactive visualization tool [9] may be used to identify spatial-temporal anomalies by combining the spatio-temporal clustering algorithm (GridScan) and by plotting 2D or 3D visual objects overlaid in Google Maps. Their case studies demonstrate its effectiveness over large-scale enterprise network traffic log as well as Air Quality Index (AQI) and PM2.5 data in China.

The work conducted in [16] also makes use of visualizing objects in both a 2D and 3D space to showcase datasets. Their case study shows the effectiveness of being able to see movement data in multiple dimensions and how each trajectory is related to another. Other visualization approaches such as [17] have created a tool which allows for both domain and data mining experts to analyze a chosen dataset. Their approach allows the data to be viewed from different perspectives giving the expert the relevant data they need in manner that best suits their particular analysis.

Tools such as ScatterBlogs [18] utilize social media feeds as input into the system. In this case, the information gathered from each social feed is utilized as the spatial and temporal data types. The tool utilizes an enhanced version of the Lloyd cluster algorithm to identify anomalies and provide authorities with relevant real time situational awareness. One of the advantages of using ScatterBlogs is in its relatively simplistic design and ability to handle and detect clusters in 1-2 million input feeds on a daily basis.

III. Methods for Detecting Anomalies

STAnD was designed to allow for analysis of a dataset from a higher level of abstraction while simultaneously giving the user the ability to examine events with more fine-grained analysis as the need arises. To support this, there are six views in STAnD (discussed in the following sections) that allow the investigator to connect spatial with temporal events at different levels of detail. An example of the main interface of STAnD can be observed in Figure 1. Each of these views works in concert with one another through

attribute or time selection to allow the user to connect suspected events between views, determine approximate causes to a threat, and in creating a connection between spatial and temporal events. Events in STAnD are defined as something happening that triggers being logged in the dataset that is under examination. These events could be either movement based, as shown in the spatial data, or could be recorded at a set interval as observed in the temporal data.

A. Movement Graph

The Movement Graph allows for individual movements to be overlaid over a physical structure. This gives the user the ability to relate how locations are interconnected and how the popularity, or lack thereof, changes over the course of time. In this view, the x and y coordinates represent a location in the physical structure. Nodes in this visualization represent a collection of spatial events that occur within a particular zone or region. These nodes are placed in the center of the designated area. Nodes sizes change based on the number of events happening at that location for a given time period. The links in the graph are used to show how movement events between nodes are connected.

B. Entropy Graph

In information theory, entropy is defined as the degree of uncertainty or impurity of a given attribute in a dataset. It is typically used to split a dataset into components in the most efficient manner to allow a decision to be made quickly. However, it can also be used to determine outliers in a dataset. Due to how entropy is calculated, it can provide a more fine-grained measure on the distribution of numeric values when compared with other statistical methods [19].

$$EventEntropy = \frac{movement_{ind}}{movement_{total}} * \log \frac{movement_{ind}}{movement_{total}} \tag{1}$$

Entropy in STAnD is calculated through Equation 1. The raw movement event entropy value itself is not used in determining anomalies for the entropy graph. Rather, abnormalities in spatial data are identified through determining outliers in entropy values for a given time set. Specifically, all the movement event entropy values for a given time period are compared to the average for that period, and the value which is the most extreme is identified as the most anomalous spatial event. In this equation, $movement_{ind}$ represents the total number of movements an item created over the specified period and $movement_{total}$ reflects the total number of movements that were created over the period for all items. Once calculated, the value is used to represent the raw entropy of an individual over the designated time.

The entropy value for each movement event is shown in the corresponding link color. These values are manually mapped to one of forty bins where red represents a lower entropy value and purple shows a higher value. This color range was taken from a standard RGB color wheel. Colors were manually selected by starting at pure red and moving around the circle in a clockwise manner until forty unique colors were selected. These values were then laid out in a linear fashion and mapped to values between zero and one. A range of forty bins were used rather than two to give better granularity between each color bin.

The Entropy Graph also allows the investigator to see how zones or regions are connected per event. However, in this view, there is no physical overlay. This permits the user to focus more on time based events rather than how they might be physically related. As in the Movement Graph, the nodes represent the center of a zone or region and the links show a movement event between two zones. Additionally, node sizes represent the number of items at a particular location at a given time.

Community detection, which attempts to partition a graph into a set of disjoint communities, may be used to assist in identifying outliers in spatial events. STAnD uses the Louvain modularity community detection algorithm [20] to place the nodes in the spatial events into groups. While the algorithm does suffer from minor accuracy when compared against other community detection algorithms, the Louvain method does allow for better scaling to large datasets and hence the primary reason for inclusion in this work. As stated, this algorithm operates on an optimized modularity which is calculated through Equation 2. As this method is a greedy approach, the goal is to maximize modularity shown as ΔQ, and thereby the group membership.

$$\Delta Q = [\frac{\sum_x + k_{i,x}}{2m} - \frac{\sum_t + k_i}{2m}^2] - [\frac{\sum_x}{2m} - (\frac{\sum_t}{2m})^2 - (\frac{k_i}{2m})^2] \tag{2}$$

In this equation, $\sum_x$ is the total weights of the links inside the graph C, while $\sum_t$ is the total weights of the links connected to nodes in the graph. k_i is the sum of weights corresponding to the links incident to the node i. $k_{i,x}$ is the total of the weights of the links from i to nodes in the graph and m is the total of the weights of all the links in the network [20]. In STAnD, the raw entropy values are used as the weights in the community detection graph rather than the processed average entropy for each entry. This was done to allow the Louvain algorithm to process the highs and lows of the entropy values without unnecessarily altering them.

Similar to entropy values, raw community membership identifiers do not indicate an abnormal entry, but rather abnormalities are identified by the largest standard deviation of entropy values associated with each community. As the time line progresses and the graph is built, the identification of outliers is done though manual identification. The standard deviation values are mapped to a color using a similar color mapping method where blue shows a lower value and red represents a higher value. The community membership is then shown in the corresponding node color. The identified event can be used as an alternate means to identify and classify an event and provide additional evidence that the item in question is of an approximate cause to a malicious event.

The method described by using the community detection algorithm is slightly different from what one might consider

a normal use for grouping. A common use for anomaly detection through grouping is by using a density function to either flag a cluster as an anomaly based on how sparse or full a grouping is with data. The community method described in the aforementioned paragraphs is still used to group similar events together, but in STAnD, the underlying data within each community is still used to identify an anomaly. An illustration of how this can be used to assist in the identification of anomalies will be shown in section IV-C.

C. Attribute Overlay

This view allows for the selection of up to eight attributes in order to allow comparisons between items at a lower level of granularity. The view is limited in the number of attributes that can be shown to allow for distinct colors to be selected and also reduce the amount of clutter in the view.

The Attribute Overlay plots a selected attribute's raw values as a percentage. The decision was made to use the percentage change rather than the raw value to allow for attributes which maintain some sort of minimum value to be compared against attributes that reside at a higher, or maximum value. By doing this, attribute behaviors can be compared without having one overshadow the other.

D. Eigen Matrix

Eigenvectors are often used in principal component analysis, feature extraction, to highlight the general behavior of attributes over time, and can be used to determine data anomalies. Eigenvectors can show over an $n*n$ dependency matrix which attribute or column is of most interest to an investigator. As eigenvectors are identifying attributes of interest, their applications can be widely used. The formula for calculating an eigenvector in STAnD is located in Equation 3. In this equation, $\mathbf{D}(t)$ represents the $n * n$ matrix, λ is the Lagrange multiplier, or in other references is identified as the eigenvalue, latent value, or the characteristic of $\mathbf{D}(t)$. The eigenvector itself is represented by $\tilde{u}$.

$$\mathbf{D}(t)\tilde{u} = \lambda\tilde{u} \tag{3}$$

The Eigen Matrix view is a plot of known temporal attributes along the x-axis and, if applicable, their location along the y-axis relative to the physical structure being examined. The circles represent a presence of an attribute at a location while the color represents a value of the underlying eigenvector for a given time period. The time period will vary from dataset to dataset, but should be short enough to gather sufficient entries to determine if trends exist in the data.

Similarly to the entropy values, the raw eigenvector value's do not indicate the presence of an abnormality, but rather are identified by the largest difference between one eigenvector's values and the subsequent vector's values for a given time interval when compared with other attributes of the same period. These values were manually mapped to one of forty color bins with a color manually assigned to each bin. The color blue was used to represent a lower change in value and the color red showcases a higher change. Shading of the color shows the degree of transformation between time periods. This creates a linear representation of color to match the values.

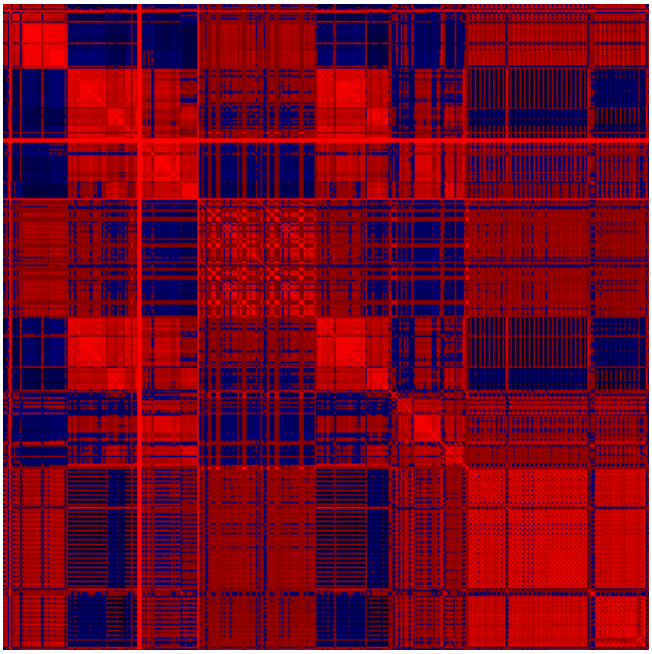


Fig. 2: Example of STAnD's ordered correlation matrix.

E. Eigenvector Chart

To compliment the Eigen Matrix, the Eigenvector Charts are created from the difference in eigenvector values from the current hour's eigenvector and the subsequent vector's values. The individual eigenvectors are calculated from a dependency matrix, which in this application consists of a correlation matrix. The underlying correlation matrix is calculated over a time period's worth of data. As with other views in STAnD, this period various between datasets, however, it should be large enough to determine trends in attributes.

This creates a view of the temporal data from a higher level of abstraction and gives the user a summary of the data, where the Eigen Matrix view allows the user to see details of the temporal data at a lower level of detail. Rather than taking the entire dataset and create an eigenvector over that data, it is separated into subcomponents which allow for a better view of the data and preserves potential relationships between attributes. A similar color scheme was used as in the Eigen Matrix view where blue shows a lower change and red represents a higher transformation in eigenvector values.

F. Correlation Matrix

The Correlation Matrix is created from the entire set of raw values in the temporal data by using the Pearson Product-Moment Correlation. This method is used as analysis conducted in STAnD is more focused around the linear relationship between values rather than the ranked values that are used in the Spearman correlation calculation.

Event	Description
Normal	An event who's presence can be reasonably explained
Abnormal	An event which deviates from normal but who's presence can be reasonably explained
Malicious	An event which deviates from normal and their presence indicates something wrong is happening

TABLE I: Listing of event types and description.

While the ordering of attributes does not effect the final values in the calculations, it does affect any visualization that is created from the calculated data. Subsequently, the data is ordered such that attributes which handle similar measurements are co-located. Once the visualization is created, it allows for easier identification and justification for patterns. As an example, Figure 2 shows large changes in patterns across the diagonal. These differences are reflective of changes in attribute categories. A similar color scheme was used in this visualization, where blue represents a lower correlation or values closer to negative one and red shows a higher correlation or values closer to positive one. The color black is used to show where there is no correlation between attributes.

Data: Anomaly Detection
Result: determine approximate cause for an event
Input: $temporalData, spatialData$
$eigenChart := eigen(temporalData)$
$entropyGraph := entropy(spatialData)$
for *each time event* $(t_i) \in eigenChart$ **do**
  for a_i *at* t_i **do**
    if e_{value} *for* $a_i > max$ **then**
      $max := (e_{value}, e_{location}, e_{time})$
  end
  for *each* $e_{location} \wedge e_{time}$ *in* max **do**
    if $e_{location}$ $\exists$ $entropyGraph$ *at* e_{time} **then**
      return $entropyGraph(e_{location}, e_{time})$
    else
      $\tau := time\_Thresold$
      $\delta := distance\_Thresold$
      while $t_x \neq \tau$ *or* $l_y \neq \delta$ **do**
        if l_y $\exists$ $entropyGraph$ *at* t_x **then**
          return $entropyGraph(l_y, t_x)$
        end
  end
end

Algorithm 1: Basic algorithm for determining if an event is an anomaly or is malicious

G. Defined Neighborhood

STAnD is designed to assist in the identification of abnormal events in both spatial and temporal data. By identifying events in both data types, it allows the investigator to create a connection between events and thereby classify the identified event as normal, abnormal, or malicious. The definitions which are used in STAnD for event identification can be found in Table I. It is worth noting that a single abnormal event does not necessarily indicate that it is malicious. However, having multiple abnormal events is a strong indicator of a potentially malicious event.

H. Event Identification Algorithm

The basic structure for identifying anomalies and determining if they are malicious can be observed in Algorithm 1. The desired output of this algorithm is to determine if an identified event in the temporal data is an approximate cause of a spatial event. In this algorithm, t_i represents each time event in the temporal dataset; a_i is an attribute that is identified to have a maximum difference in eigenvector values and includes the values: e_v, the location: e_l, and the identified time: e_t.

The location and time are used to start investigations into the spatial dataset. If there is a spatial event at that time and location, it is returned as an approximate cause to the event. Otherwise, both the time and date parameter must be adjusted, within a set threshold, to see if something happened before or nearby that might have caused the temporal event. These thresholds are identified in algorithm 1 as τ for time and δ to represent distance. These thresholds are largely determined by the dataset being examined. As an example, in section IV, there are two thresholds used: a period of up to 48 hours for the weekend and 8 hours for weekdays. These time periods were used to account for when individuals were not scheduled to work over the weekend and to account for differences in shifts during the week. The distance parameter, as in section IV, was compared against the physical structure to determine most likely connections between areas.

The new time and distance to compare against are represented as t_x and l_y respectively. If an event is found within those thresholds, the spatial event is returned as approximate cause, otherwise, no spatial event can be found that can accurately explain a reason for the temporal event.

IV. Office Building - A Case Study

To demonstrate STAnD's capabilities, we used the Visual Analytics Science and Technology (VAST) challenge 2016 dataset, which contains a fourteen day simulation of a three story office building with employees working various shifts. There are several sensors placed throughout the building to record entries for employee movement and for keeping track of building properties such as thermostat settings, airflow rate, water pressure and carbon dioxide levels among others. The employee movements are captured through a proximity card sensor network that tracks an employee's movement as they enter a zone in the building. The building sensors are tracked through a separate network defined as the heating, ventilation and cooling (HVAC) network. Entries in this section are captured every five minutes. In addition, there are four sensors in the building to detect the presence of a factitious hazardous chemical called hazium. According to the dataset description, the presence of this chemical in any amount is suspicious.

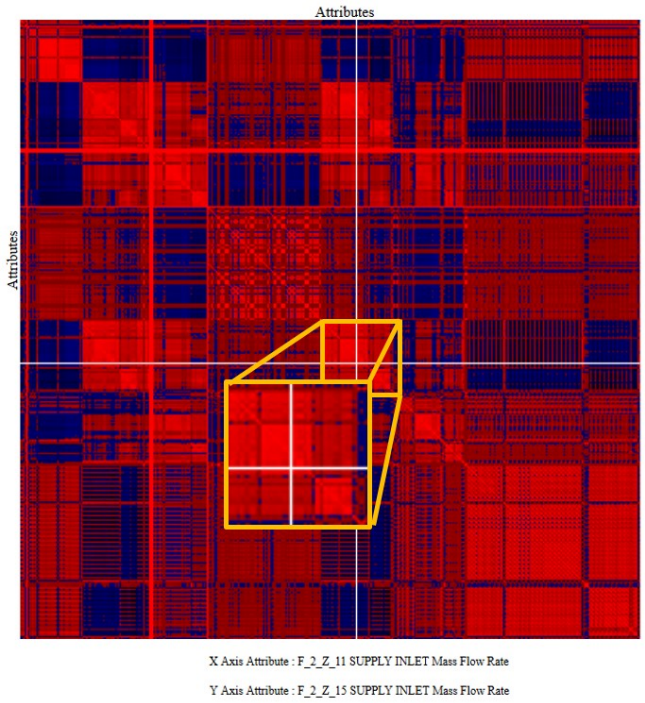


Fig. 3: Normal event identified in the Correlation Matrix.

A. Identification of Neighborhood

These four hazium sensors are used to help refine the scope of investigation using STAnD. As such, the question that will be answered is: "Can it be determined which employee is of approximate cause to the hazium spike and what might have caused said spike?" While any dataset can be examined with various traditional statistical methods, it is difficult for these procedures to show a connections from one entry to another. The strength in STAnD is each view allows the data to be viewed through a different process. While the visualizations can be used individually, they are more powerful when used together to help identify anomalies and filter between event types. Tying events together will be done by showing how events in the temporal data are connected to entries observed in the spatial views.

B. Identification of Normal Event

An example of what is considered a normal event, can be observed in Figure 3. As seen in the call out box, one can observe two of the attributes that were selected. Those attributes that were selected handle the same measurement in different areas of the building - "Supply Inlet Mass Flow Rate". The color red, or a high correlation value, at this location should make sense as what is known of the building, there isn't a shut off specific for each zone.

As additional confirmation of the attributes behavior, the values can be observed in the attribute overlay as seen in Figure 4. In this figure, the two attributes are shown in there own attribute overlay to allow the reader to observe that the attributes behave similarly in both locations.

C. Identification of Abnormal Event

As stated, for this dataset, any presence of hazium is considered hazardous, but to what extent is unknown. To assist in the identification of the presence of hazium, a darker line was added in the Eigenvector Charts. This line represents any increase in the hazium value for a given hour. It can be observed in Figure 5, that the building has it's first hazium increase. As any amount of a potentially hazardous chemical has no logical explanation, this event should be considered an anomaly. However, that analysis is minimal and does nothing to suggest a potential cause.

In order to further identify a possible cause or related event to the hazium increase, the Eigen Matrix can be used to identify the specific attribute or attributes that were also acting anomalous at the same time as the hazardous chemical increase as identified in Figure 5. As can be seen in Figure 6, there is one abnormal sensor reading at the same hour and location of the hazium increase. This attribute can be identified as the "VAV Reheat Damper Position" located on floor 3 HVAC zone 6.

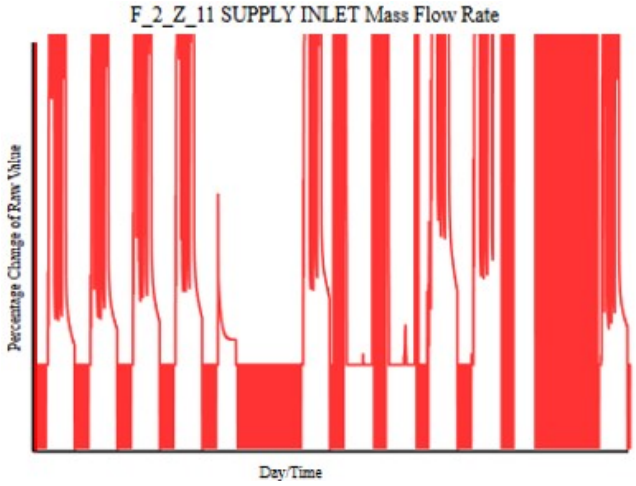
When viewed in the attribute overlay and compared against the hazium levels, it can be observed that at the time of the hazium increase, the "VAV Reheat Damper Position" moves rapidly to a closed setting. Because the eigenvectors are calculated from a correlation matrix of the temporal data, these events must then be related.

Additionally, this event can be corroborated with the spatial movements of the employees. There are two individuals whose movements at the time of the hazium increase are most abnormal from all others of the same time period. These employees are P. Young and K. Herrero. The movement events for all employees in the same hour leading up to the hazium event can be observed in the Entropy Graph in Figure 7. To repeat, the raw entropy value does not indicate the presence of an anomaly, but rather the most extreme outlier in a grouping that indicates an anomaly. However, in cases where the extreme value is close to another value as in Figure 7, community detection can assist in filtering events.

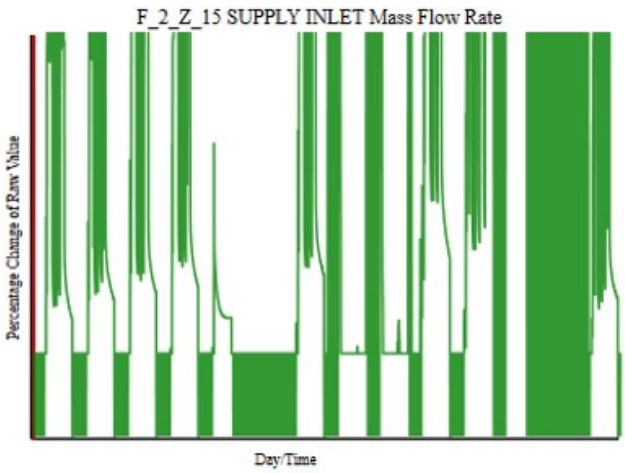
This gives the events of interest those that are surrounding the purple nodes in Figure 7. Upon further investigation it can be discovered that only one employee, P. Young, is located on floor 3 in proximity to the zone with the hazium increase. K. Herrero's movements should only be considered abnormal but not malicious as they are located some distance from floor 3 at the time of the increase. It is worth noting that prior to the hazium increase, no other employees are in the effected zone.

By looking at the Movement Graph for P. Young at the time of the hazium increase, we can further confirm that this employee's movements are abnormal. This employee's movements leading up to the hazium increase can be observed in Figure 8

As a possible explanation for this event, it is offered that because P. Young works in the facilities department, they would have knowledge of the building's inner structure and be aware of when certain building functions happen or



(a) Floor 2 zone 11.



(b) Floor 2 zone 15.

Fig. 4: Attribute Overlay for Supply Inlet Mass Flow Rate.

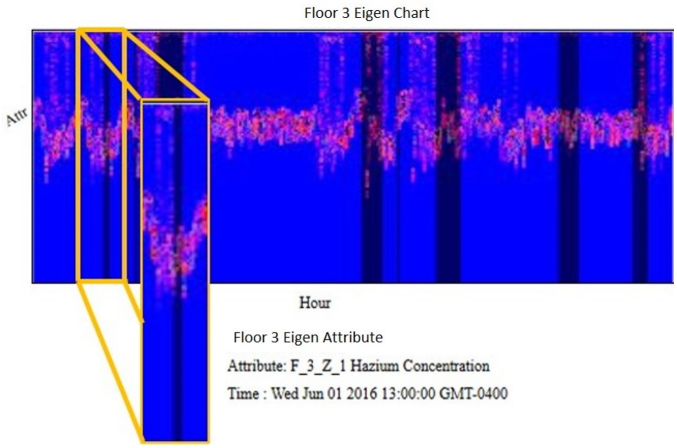


Fig. 5: Hazium increase identified in floor 3 Eigenvector Chart.

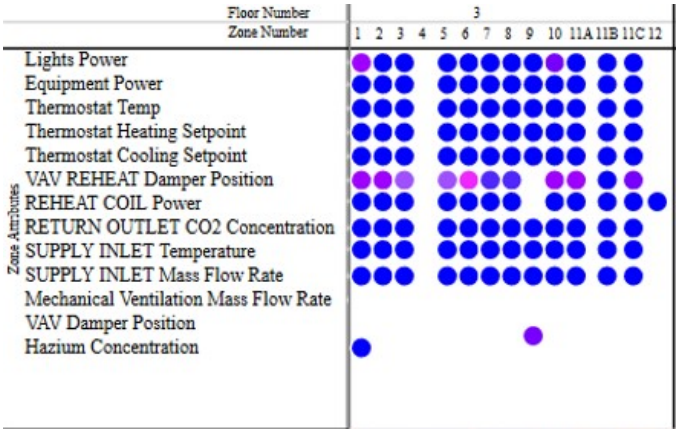


Fig. 6: Anomalous attribute identified on floor 3 in the Eigen Matrix.

knows how to trigger them. As P. Young's movements were always anomalous in the minutes prior to the hazium event, it is possible that there was an opportunity to stop him before the actual event was triggered.

D. Other Notable Events

There is a large spike in CO2 readings on one of the weekends in this dataset. There are no employees as being listed as present in the building at the same time of the rise in CO2 readings. These CO2 levels should correlate with the number of persons in a location. There maybe a logical explanation for this as the CO2 levels are high in the main entrance, a large conference room, the southern half of the first floor corridor and the first floor conference room. It is possible that there is an event taking place here that does not include employees. What's most interesting about this event is that STAnD indicates that the reheat damper position is abnormal, suggesting that the damper's position

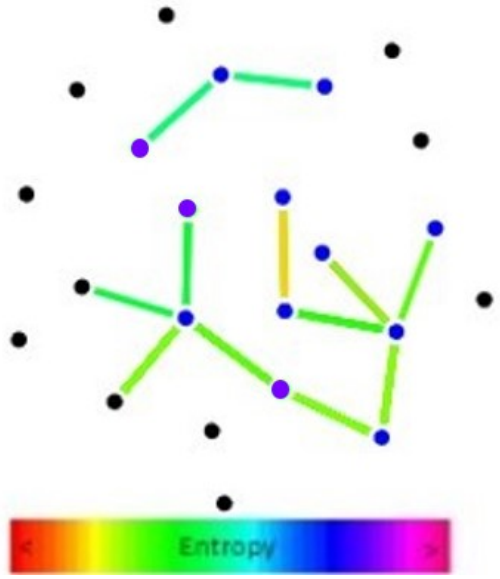


Fig. 7: Anomalous spatial movements at 1:10pm 1 June.

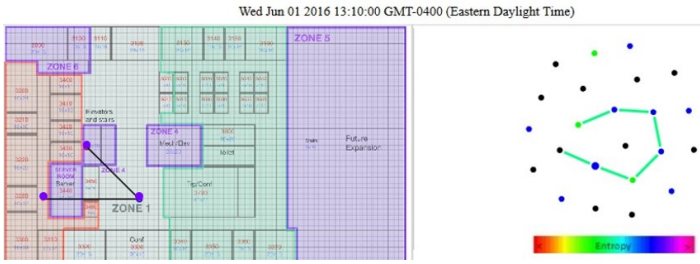


Fig. 8: Movement graph for P. Young for floor 3 leading up to the hazium event.

Events Identified	Temporal Events	Spatial Events	Both found together
24	20	20	18

TABLE II: Total anomalies found for the office building dataset.

is working to remove the higher than normal CO2 levels when it otherwise would not have.

In the dataset on floor 3, there is a zone labeled as future expansion. Over the course of the fourteen days, there is no indication of an employee entering that zone. However, the CO2 readings show a continuous rise and fall. These readings are not flagged as abnormal due to the similarities in patterns exhibited on other zones of this floor for the sensor type. However, other sensors - reheat damper position and various temperature sensors - indicate an abnormality. There does not appear to be a correlation between this zone and others in terms of potentially malicious events. It is possible that these sensors are triggering events due to possible construction work taking place in this area.

The total number of anomalies found can be observed in Table II. In each abnormality found, an individual exhibits suspicious behavior in the lead up to a possible temporal anomaly. The total number of events was then determined by the closest event that preceded a hazium increase.

V. Conclusion

As we continue to move into the big data era, the amount of data from people and devices containing both spatial and temporal dimensions will continue to grow fast. How to connect the dynamics of events and quickly detect abnormalities is important yet challenging. Through this work it has been shown that the use of entropy and eigenvectors are a viable choice for determine normal and abnormal in spatial-temporal datasets. The identification of these events with this tool, can help investigators determine approximate causes to suspected security events. By using several visualization methods, one can get a better view of the data, as well as tie suspicious abnormal events together to determine causes. Through the case study, STAnD demonstrates its ability to showcase datasets of multiple dimensions and reinforce knowledge gained.

References

[1] X. Wu, X. Zhu, G. Q. Wu, and W. Ding, "Data mining with big data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 1, pp. 97–107, Jan 2014.

[2] H. Izakian and W. Pedrycz, "Anomaly detection and characterization in spatial time series data: A cluster-centric approach," *IEEE Transactions on Fuzzy Systems*, vol. 22, no. 6, pp. 1612–1624, Dec 2014.

[3] E. W. Dereszynski and T. G. Dietterich, "Spatiotemporal models for data-anomaly detection in dynamic environmental monitoring campaigns," *ACM Trans. Sen. Netw.*, vol. 8, no. 1, pp. 3:1–3:36, Aug. 2011.

[4] X. R. Wang, J. T. Lizier, O. Obst, M. Prokopenko, and P. Wang, *Spatiotemporal Anomaly Detection in Gas Monitoring Sensor Networks*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 90–105.

[5] G. Tandon and P. K. Chan, "Spatio-temporal anomaly detection for mobile devices," Tech. Rep., 2007.

[6] W. C. Young, J. E. Blumenstock, E. B. Fox, and T. H. McCormick, "Detecting and classifying anomalous behavior in spatiotemporal network data," in *Proceedings of the 2014 KDD workshop on learning about emergencies from social information (KDD-LESI 2014)*, 2014, pp. 29–33.

[7] S. Zhou, W. Shen, D. Zeng, M. Fang, Y. Wei, and Z. Zhang, "Spatial temporal convolutional neural networks for anomaly detection and localization in crowded scenes," *Signal Processing: Image Communication*, vol. 47, pp. 358 – 368, 2016.

[8] I. C. Paschalidis and G. Smaragdakis, "Spatio-temporal network anomaly detection by assessing deviations of empirical measures," *IEEE/ACM Transactions on Networking (TON)*, vol. 17, no. 3, pp. 685–697, 2009.

[9] T. Zhang, Q. Liao, L. Shi, and W. Dong, "Analyzing spatiotemporal anomalies through interactive visualization," *Informatics*, vol. 1, no. 1, pp. 100–125, 2014.

[10] C. Xu, S. Chen, and J. Cheng, "Network user interest pattern mining based on entropy clustering algorithm," in *Proceedings of the 2015 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery*, Nanjing, China, Sept 2015, pp. 200–204.

[11] Z. Zhang, X. Wang, and L. Sun, "Mobile payment anomaly detection mechanism based on information entropy," *IET Networks*, vol. 5, no. 1, pp. 1–7, 2016.

[12] L. Akoglu and C. Faloutsos, "Event detection in time series of mobile communication graphs," in *Proceedings of the 2010 Army Science Conference*, 2010, pp. 77–79.

[13] T. Idé and H. Kashima, "Eigenspace-based anomaly detection in computer systems," in *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '04. New York, NY, USA: ACM, 2004, pp. 440–449.

[14] N. Andrienko, G. Andrienko, and P. Gatalsky, "Exploratory spatio-temporal visualization: an analytical review," *Journal of Visual Languages & Computing*, vol. 14, no. 6, pp. 503 – 541, 2003, visual Data Mining.

[15] Y. Zhang, G. Li, C. Lai, Q. Liu, S. Chen, L. Feng, T. Ye, S. Chen, R. Zue, Z. Zhang, Z. Wang, X. Huang, F. Xu, L. Yu, S. Zhang, Q. Li, and X. Yuan, "STAD-HD: Spatial temporal anomaly detection for heterogeneous data through visual analytics," in *Proceedings of the 2016 VIS*, Baltimore, MA, USA, 2016.

[16] U. Demar and K. Virrantaus, "Spacetime density of trajectories: exploring spatio-temporal patterns in movement data," *International Journal of Geographical Information Science*, vol. 24, no. 10, pp. 1527–1542, 2010.

[17] P. Compieta, S. D. Martino, M. Bertolotto, F. Ferrucci, and T. Kechadi, "Exploratory spatio-temporal data mining and visualization," *Journal of Visual Languages & Computing*, vol. 18, no. 3, pp. 255 – 279, 2007, visual Languages and Techniques for Human-GIS Interaction.

[18] D. Thom, H. Bosch, S. Koch, M. Wrner, and T. Ertl, "Spatiotemporal anomaly detection through visual analysis of geolocated twitter messages," in *Proceedings of the 2012 IEEE Pacific Visualization Symposium*, Songdo, South Korea, Feb 2012, pp. 41–48.

[19] F. Zhou, W. Huang, Y. Zhao, Y. Shi, X. Liang, and X. Fan, "Entvis: A visual analytic tool for entropy-based network traffic anomaly detection," *IEEE Computer Graphics and Applications*, vol. 35, no. 6, pp. 42–50, Nov 2015.

[20] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2008, no. 10, p. 10008, 2008.