An eigenvector-based hotspot detection

2 authors:

Hadi Fanaee-T
University of Oslo

19 PUBLICATIONS   **86** CITATIONS

João Gama
University of Porto

356 PUBLICATIONS   **5,366** CITATIONS

Some of the authors of this publication are also working on these related projects:

Project   Doctoral View project

Project   Clustering Data Streams View project

An eigenvector-based hotspot detection

Hadi Fanaee-T[1] and João Gama[2]

[1] LIAAD-INESC TEC/FCUP
[2] LIAAD-INESC TEC/FEP
University of Porto, Portugal
{hadi.fanaee@fe, jgama@fep}.up.pt

Abstract. Space and time are two critical components of many real world systems. For this reason, analysis of anomalies in spatiotemporal data has been a great of interest. In this work, application of tensor decomposition and eigenspace techniques on spatiotemporal hotspot detection is investigated. An algorithm called SST-Hotspot is proposed which accounts for spatiotemporal variations in data and detect hotspots using matching of eigenvector elements of two cases and population tensors. The experimental results reveal the interesting application of tensor decomposition and eigenvector-based techniques in hotspot analysis.

Keywords: Hotspot analysis, Tensor decomposition, Spatiotemporal data

1 Introduction

Space and time are two critical components of many real world systems. Nowadays, there is great interest in spatiotemporal data analysis due to the huge amount of data available. Among various analysis tasks that can be carried out on spatiotemporal data, hotspot analysis is recognized as an important tool in security informatics and bio-surveillance. For instance in crime hotspot application, an outcome such as *City Center between hours 8 to 11pm* would be a spatiotemporal hotspot. Outcome like *City Center* or *City Park* would be strict spatial hotspots and *8 to 11pm* and *10 to 12pm* are samples of temporal hotspots. Hotspot analysis goal consists of detecting spatiotemporal regions among data which are showing significant deviations comparing to the rest of the data.

In this paper we propose a novel spatiotemporal hotspot analysis approach based on tensor decomposition [1] and eigenvector elements matching. This approach can be applied for solving two types of problems. Firstly, it independently can detect semi-spatiotemporal hotspots and in the second application can be used as a pre-processing engine for common spatiotemporal methods to reduce a big search space to a few limited regions for post-processing.

The rest of the paper is organized as follow. First in section 2 we outline the related works. In section 3 we explain the proposed algorithm for hotspot detection. In section 4 we describe the experiments and finally in section 5 we end with conclusion and future works.

2 Related Work

Related spatiotemporal techniques can be divided into two main categories: scan statistics and clustering-based techniques. Clustering-based approaches [2–6] are based on this idea that first, thresholds are inferred from the population data and then estimated thresholds are applied on clustering of data points of cases data. Clustering-based approaches have their own limitations and strengths. Their prominent benefit is that they provide exact shape of clusters opposed to the scan statistics-based methods where clusters necessarily should be a regular shape and are not realistic. On the other hand, handling complex data is not straightforward for clustering-based techniques.

Table 1. Comparison between our approach and ST-Scan

	ST-Scan	Our approach
Data input	Spatiotemporal	Multidimensional
Support additional attributes	No	Yes
Support additional dimensions	No	Yes
Spatiotemporal clusters	Exact	Approximate
Automatic trend adjustment	No	Yes
Take into account interdependencies and correlation between dimensions and measurements	No	Yes
Automatically Noise reduction	No	Yes

Scan statistics approaches are based on Kulldorff original work [7] and its variant extensions and are developed for detection of anomalous patterns from spatiotemporal data sets. Table 1 demonstrates a comparison between our approach and Space-time scan statistics (ST-Scan) based approaches. As it can be seen, our approach not only have no restriction for the number of attributes and dimensions but also can discover interdependencies and correlations among different attributes and dimensions. This strength originates from tensor decomposition emphasis which already has shown a great performance in many real-world applications such as chemometrics, econometrics and psychometrics [8]. A comprehensive survey about tensor decomposition techniques and its applications is presented in [1]. Tensor decomposition itself cannot be used for hotspot detection. We adapted some ideas from computer vision such as eigenvector-based model matching which are also shown promising results for face reorganization [9–11]. Another privilege of our approach comparing to the ST-Scan is its ability for reduction of the noises and taking into account the temporal adjustment in an automatic way.

Our approach has some distinct features that make it distinguished from similar works. One of these distinctive features is its strength in handling interdependencies between attributes and dimensions in multilinear and multidimensional

data. To the best of our knowledge, such feature is not taken into account in the existing approaches. This strength originates from tensor decomposition emphasis which already has shown a great performance in many real-world applications in particular areas such as chemometrics, econometrics and psychometrics [8]. However, tensor decomposition itself cannot be used for hotspot detection. We adapt some other ideas from computer vision such as eigenvector-based model matching which are also shown promising results for face reorganization [9–11]. Our work however is different from common eigenvector-based model matching in computer vision. We here, instead of comparing two eigenvectors together, compare their individual eigenvector elements.

2.1 Space-time Scan Statistics (ST-Scan)

There are several variants of ST-Scan approaches. Introducing and comparing of the work with all these dozens of variants is out of scope of this paper. So we confine ourselves to introduce the baseline ST-Scan method [7]. ST-Scan exhaustively moves a varying radius and height cylinder over spatiotemporal space where the height of this cylinder is corresponding to the time dimension and the surface covers the space dimension. It then computes a score F(S) based on the following likelihood ratio statistic [12] (equation 1) for each spatiotemporal cylinder:

$$F(S) = (\frac{C}{B})^C (\frac{C_{total} - C}{B_{total} - B})^{C_{total} - C} \tag{1}$$

where, C is total counts and B is baseline in S and C_{total} and B_{total} are also total counts and baseline of search area respectively. Then all possible cylinders are sorted based on the highest to lowest score and a randomization test is performed for obtaining the cylinder statistical significance. Then spatiotemporal regions whose p-value is lower than a threshold (usually 0.05) are returned as spatiotemporal hotspots.

3 Detection of Spatial and Temporal hotspots

Here we introduce our proposed algorithm for detection of spatial and temporal hotspots. As well as ST-Scan, our algorithm receives three data sets: population data, cases data and geographic data. Outcomes of the algorithm are spatial likely cluster, spatial first priority and second priority hotspot clusters and temporal hotspot intervals. The algorithm is composed of three main parts. The first part is tensor decomposition (lines 1-2). The second part is eigenvector elements matching in lines 4-6 (spatial) and lines 7-9 (temporal) and the rest of the algorithm are allocated to spatial (lines 10-29) and temporal (lines 30-44) hotspot detection.

The required geographic data is a bit different from ST-Scan. In ST-Scan, geographic data includes coordinate of each region. This data need to be processed to be transformed to a neighbor's matrix. As a result we should have a

---

Algorithm 1 SST-Hotspot

Require: Tensor P, Tensor C, Matrix Neighbors, Parameters [R1,R2, ... Rn]

Ensure: LikelyCluster, FirstPriority, SecondPrority, TFirstPriority, TSecondPriority

1: Decompose P: $ES_p \leftarrow 1st\ SEigenvector$, $ET_p \leftarrow 1st\ TEigenvector$
2: Decompose C: $ES_c \leftarrow 1st\ SEigenvector$, $ET_c \leftarrow 1st\ TEigenvector$
3: Perform Sign correction on Pair (ESp,ESc) and Pair (ETp,ETc)
4: **for** each region s **do**   ▷ Spatial Eigenvector elements matching
5:     $DS(s) \leftarrow ESp(s) - ESc(s)$
6: **end for**
7: **for** each time t **do**   ▷ Temporal Eigenvector elements matching
8:     $DT(t) \leftarrow ETp(t) - ETc(t)$
9: **end for**
10: Descending Sort of DS   ▷ Spatial
11: $SL \leftarrow DS(s) \succ 0$
12: $SC \leftarrow Regions\ that\ have\ DS(s) > std(DS)$
13: $ST \leftarrow Remove\ SC\ from\ SL$
14: $S1 \leftarrow Regions\ in\ ST\ that\ have\ DS(s) \geq std(ST)$
15: $S2 \leftarrow Regions\ in\ ST\ that\ have\ DS(s) \prec std(ST)$
16: $LikelyCluster \leftarrow Regions\ in\ S1\ with\ pair\ distance\ lower\ than\ std(ST)$
17: **for** each X in S1 **do**
18:     **for** each H in SC **do**
19:         If Neighbors(X,H)==1 Then Add X to FirstPriority(H)
20:     **end for**
21: **end for**
22: SecondPrority=FirstPriority
23: **for** each X in S2 **do**
24:     **for** each H in SC **do**
25:         **for** each M in FirstPriority(H) **do**
26:             If Neighbors(X,M)==1 Then Add X to SecondPrority(H)
27:         **end for**
28:     **end for**
29: **end for**
30: Descending Sort of DT   ▷ Temporal
31: $TL \leftarrow DT(t) \succ 0$
32: $TC \leftarrow Times\ that\ have\ DT(t) > std(DT)$
33: $TT \leftarrow Remove\ TC\ from\ TL$
34: $T1 \leftarrow Times\ in\ TT\ that\ have\ DT(t) \geq std(TT)$
35: **for** each H1 in TC **do**
36:     **for** each H2 in TC **do**
37:         Add non-repeated [H1-H2] to TFirstPriority
38:     **end for**
39: **end for**
40: **for** each X in T1 **do**
41:     **for** each H in TC **do**
42:         Add [X-H] to TSecondPriority
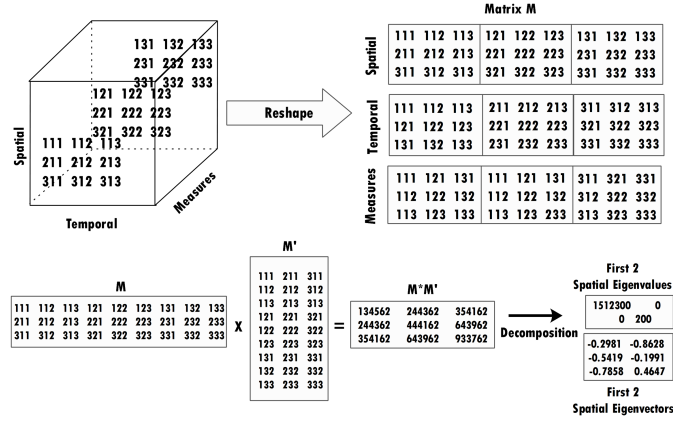43:   **end for**
44: **end for**

Fig. 1. A sample of 3D spatiotemporal tensor decomposition process

matrix of regions-regions which each cell represents the boolean value such that 1 indicates that the region X is neighbor (e.g. has a border) with region Y.

Algorithm 1 demonstrates the algorithm SST-Hotspot. As an input it receives population tensor, cases tensor, geographic data and model parameters. Model parameters Rn by default for each dimension of space and time are 2 and for other dimensions are considered as 1 (R1=2, R2=2, R3=1, Rn=1, ...). It means that tensor decomposition will output 2 eigenvectors for space and time and one for other dimensions.

At lines 1-2 of algorithm we decompose population and cases tensor. *SEigen-vector* and *TEigenvector* represents spatial and temporal eigenvectors respectively. This process is illustrated in Figure 1. A 3D spatiotemporal tensor with dimensions of $3 \times 3 \times 3$ (spatial, temporal and measures) is demonstrated in this figure. We first reshape population and cases tensor in each dimension to the 2D matrix M. Thereafter, we first form a matrix of M×M' and then perform a matrix decomposition to obtain its first 2 eigenvectors. The reason why we retrieve only 2 eigenvectors is that our model parameter for spatial mode has been set to 2 in this example. The First eigenvector represents majority of variance in data and is appropriate for model comparison. Second and rest eigenvectors only include small variations of data and might include noises. We do not limit the algorithm to only one eigenvector. However based on our preliminary observations, on the case study data set, taking the first eigenvector was appropriate for model matching. In other cases one might need to pre-examinate the model on a sample data to see how many eigenvectors are appropriate for model matching.

After decomposition step we need to perform a sign correction [9, 13] at line 3. The reason why we need to apply sign correction is that eigenvectors are not unique; i.e., if ei is an eigenvector, then -ei also can be an eigenvector. Therefore, without sign correction or alignment of the corresponding dimension for the two
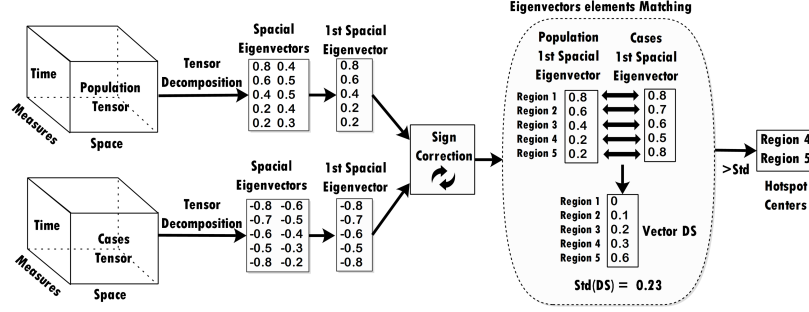
Fig. 2. A sample of Eigenvectors elements matching process

models, a direct comparison of eigenvectors is meaningless [9]. We adapt sign correction technique from [9] to make both sets of eigenvectors have consistent directions.

At lines 4-9 for each spatial region and each temporal point we perform eigenvector elements matching. Note that we do not compute the distance between two eigenvectors rather we match each eigenvector elements corresponding to each region. This process is illustrated in Figure 2. This figure shows two same size tensors of population and cases. As is illustrated, in order to obtain first two eigenvectors, population and cases tensors are decomposed and after sign correction, value of each region eigenvector element in cases first eigenvector is subtracted from the corresponding one in population first eigenvector. This is repeated for all regions and obtained values are allocated in vector DS. Then, the standard deviation of this vector is computed as a threshold. Those regions that their corresponding distances are greater than this threshold are reported as hotpots centers (e.g. in Figure 2, region 4 and 5) which is equivalent to SC in the algorithm.

At line 11 we make another copy of DS saving in variable SL where DS(s) is greater than zero. At line 12 we identify hotspot centers as those regions which DS(s) are greater than standard deviation of DS. At line 13 we make another copy of SL by excluding of hotspot centers into variable ST. We then define standard deviation of ST (std(ST)) as a threshold that distinguish first priority (S1) from second priority (S2) list. Based on this, S1 would include regions whose DS(s) are greater than std(ST) (line 14) and S1 includes regions whose DS(s) are lower than std(ST) (line 15). We define likely cluster at line 16 as hotspots which are shown up in the S1 and their pair distances are very close (lower than std(ST)).

In order to detect hotspot clusters, in the first step (lines 17-21) we connect regions that are appeared in S1 to one of hotspot centers if they are spatial neighborhood. In other words, if for instance region A is identified as a hotspot center and its spatial neighbor B is appeared in S1, likely they form a hotspot

cluster FirstPriority(1)={A,B}. Although, the first priority list includes the most abnormal regions, however those regions that their DS(s) is greater than zero would be interesting for some purposes. For this reason we generate another list called second priority hotspot clusters. We first make a copy of first priority clusters and then connect those regions that are appeared in S2 to one of first priority cluster members. This is done via lines 22-29 in the algorithm. The process is the same as first priority list with this difference that this time we consider the new members of first priorities clusters. For instance in the above example, suppose that region C is not appeared in the S1 but is appeared in S2 and spatially is neighborhood of region B. So second priority list would be SecondPriority(1)={A,B,C}. Now suppose that we have another region D that is appeared in S2 and is a neighborhood of region C but none of regions A and B. This region is not added to second priority list, because it has no connection with the first priority members A and B.

Detection of temporal hotspots is similar to the spatial one. Explanation of lines 30-34 are almost same as lines 10-14. The only difference is this that for temporal detection we do not look for clusters; rather we only need to connect temporal hotspots together to generate hotspot intervals (e.g. '86-89'). At line 35-39 we generate intervals between each temporal hotspot centers together and in lines 40-44 we make an interval between each point in T1 and one of temporal hotspot centers.

4   Experiments

We applied our algorithm on a real data set and tried to compare its performance in comparison to ST-Scan as a basline method to assess the algorithm performance. We used MATLAB running on a personal PC with Intel Core 2 Duo CPU and 3GB Ram. Three MATLAB toolboxes were also used during the experiments: Tensor toolbox [14], ITA toolbox [15] and mapping toolbox [16] for drawing the results on the map. In the following we introduce the used data set and obtained results.

4.1   Data Set

We took the Incidence and Population data from New Mexico brain cancer data set used in [7]. The original source of data is *Surveillance, Epidemiology and End Results (SEER) program* of national cancer institute, collected by New Mexico tumor registry between years of 1973 to 1991 for 32 sub-regions of New Mexico State, United states. There are 1175 reported cases of malignant neoplasm of the brain and nervous system. Each record of cases data includes region of residence, year of diagnosis, age group in 5-year interval (19 groups), race (white, black, other) and sex. Similarly the same attributes are available for populations. Data set is publicly available at [17]. We also used PostgreSQL spatial extension [18] on New Mexico state shape file to to extract the neighborhood matrix. After all we transformed data sets to the tensor format. We made three tensors in

different orders: 2D {D1=Space, D2=Time} where we ignored other attributes such as age, sex and race, 3D tensor {D1=Space, D2=Time, D3= (Age, Sex, Race)} and 5D {D1=Space, D2=Time, D3=Age, D4=Sex, D5=Race}. Each cell in above tensors represents the count (cancer observations in case set and population count in population set).

4.2 Results

Experiments were designed to answer the following questions. In the following subsections we discuss the obtained results regarding each research question.

1. What is the benefit of higher order data modeling? How we can choose the order?
2. What is the validity of the detected Hotspots?
3. How does our approach prune non-significant regions?
4. How does our approach treat with temporal trends in population data?

We applied our algorithm on the generated 2D, 3D and 5D tensors and then compared the first and second priority hotspot clusters to the ST-Scan outputs presented in [7] (which is already obtained via SatScan software). Results for first priority clusters are presented in Figure 3.a and for second priority cluster are presented Figure 3.b. We used standard metrics such as F1 (F-measure), precision and recall as well for comparing output of the algorithm with ST-Scan. This result is shown in Table 2. Likely detected cluster from ST-Scan and SST-Hospot on different tensor orders are also reported in Table 3.

Validation of hotspots by comparing with ST-Scan results. Looking to the Table 3 we easily can observe that how our algorithm detected the important spatial regions "*Bernalillo* and *Chaves*" and two important temporal hotspot centers: *85* and *89*. As it can be seen, this result is similar to ST-Scan. ST-Scan outputs another three clusters "*Chaves*" , "*Curry and Roosevelt*" and "*Los Alamos*". Among these clusters, SST-Hotspot successfully detected "*Chaves*". However, "*Curry* and *Roosevelt*" are neighbors of *Chaves* and *Los Alamos* is almost near to the *Bernalillo*. In other words, SST-Hotspot effectively detected the center of events both spatially and temporally. If we look to the obtained result from ST-Scan (Top left figure in Figures 3.a and Figure 3.b) we find two major affected zones with center of *Bernalillo* and *Chaves*. Both of these centers are detected by SST-Hotspot. As was expected, the most significant anomalous parts of data are appeared in the eigenspace.

Effects of different tensor ordering. ST-Scan by default does not consider the existing trend in population data. The solution provided in [7] was to increase incidence rates 1.2 percent per year based on the available growth rate in population data. Based on this adjustment the only reported cluster from SatScan Software is "*Los Alamos* and *SantaFe*" in years "*86-89*". Our algorithm SST-Hotspot
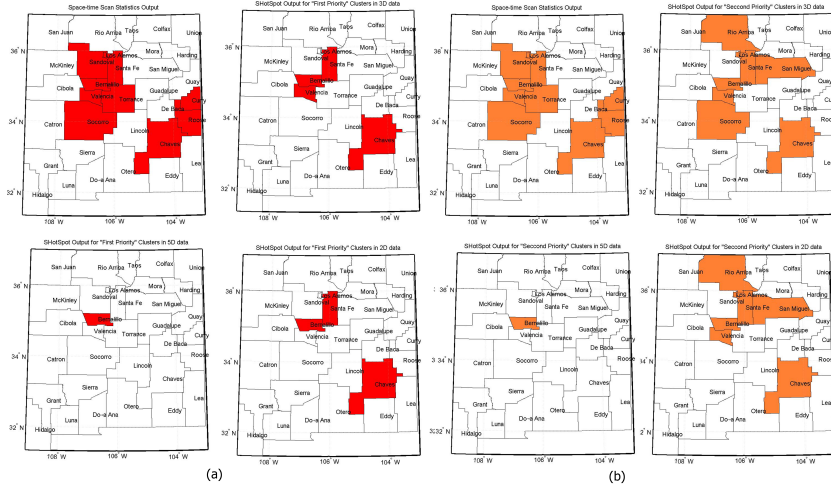
Fig. 3. Hotspot clusters, ST-Scan (Top Left) , SST-Hotspot 2D (Bottom Right), SST-Hotspot 3D (Top Right) and SST-Hotspot 5D (Bottom Left). (a) First priority clusters and (b) Second priority clusters

Table 2. Comparisons of detection power of spatial hotspots with ST-scan results in each tensor order

	SST-Hotspot 2D		SST-Hotspot 3D		SST-Hotspot 5D	
	First Pri-ority	Second Priority	First Pri-ority	Second Priority	First Pri-ority	Second Priority
Precision	100.0	71.43	**100.00**	**75.00**	100.00	100.00
Recall	30.00	50.00	**40.00**	**60.00**	10.00	10.00
F1	46.15	58.82	**57.14**	**66.67**	18.18	18.18

on 3D order also could detect this hotspot. For instance in our experiments on 3D ordering and in S1 list we obtained DS(Los Alamos)=0.005 and DS(Santa Fe)=0.004 where standard deviation of DS vector was equal to Std(DS)=0.004. There were only these two regions in S1 list that their distances together was lower than standard deviation of DS vector (|0.004-0.005|=0.001<0.004) and therefore were reported as most likely hotspot at line 16 of algorithm. This is while in other orders 2D and 5D, we could not detect this hotspot. This shows the importance of tensor order selection.

The reason why this cluster is not detected in 2D and 5D is this that ST-Scan adjusts counts for sex, age and race and thus naturally effects of these attributes are considered in the counts, so, as was expected, the results had to become more similar to ST-Scan. 2D ordering do not take into account other attributes (sex, age and race) and clearly shall not detect similar hotspots as well as ST-Scan. In terms of 5D ordering, the fit of the model for cases data was

Table 3. Detected Hotspot Clusters

Method	ST-Scan	SST-Hotspot 2D	SST-Hotspot 3D	SST-Hotspot 5D
Likely reported spatial clusters	Bernalillo (center of cluster)	Bernalillo	Bernalillo	Bernalillo
Secondary reported spatial clusters	Chaves , Curry and Roosevelt , Los Alamos	Chaves	Chaves	Not Found
Likely reported spatial cluster with trend adjustment	Los Alamos and SantaFe	Not Found	Los Alamos and SantaFe	Not Found
Likely reported temporal clusters	85-89	85,89	85,89	86,89
Secondary reported temporal clusters	88-89 , 82-83 , 86-89	86,90,88,83	86,88,83,90	88,85,91

very low (about 14 percent). For this reason model obtained from cases tensor would not be a good sample to compare with population model with fit of over 89 percent. The reason is this fact that decomposition methods performance drops when face with a sparse tensors. This is while we get 61 percent for cases tensor and 96 percent for population tensor in 3D order. Based on the above findings we conclude that higher order modeling of data improves the detection power, however the ordering needs to be selected carefully. For instance in our case study, the highest tensor order was 5 but presented a lower quality than order 3.

Pruning functionality evaluation. As we earlier mentioned, our algorithm has two functions, first it can be used as a method for spatial and temporal hotspot detection from multidimensional data and second can be used as a pre-processing step in ST-scan process to reduce the search space. It can prune thousands of non-significant spatial or temporal regions in ST-Scan. This means that more cover is required and thus the recall measure would be more significant for us. Because when it is used in ST-Scan it should cover more candidate regions for being tested in ST-Scan process. Therefore the accuracy does not make scene because the final decision about anomaly of the regions should be made by ST-Scan test.

However when SST-Hotspot is going to be applied independently, F-measure need to be taken into account instead. The best F-measure obtained for first priority list of 3D model was 66.67 percent. It means that when the goal is to detect the spatial hotspots from spatiotemporal data we can expect a close result to SST-Scan. This is extremely reasonable when we compare the required extensive computation cost of ST-Scan with light computation cost of SST-Hotspot.

As a result, instead of performing ST-scan test for thousands of cylinders we just need to run the test for a limited regions and temporal periods outputted

from SST-Hotspot. In fact, we have the center of events and suspicious neighbors and suspicious time intervals in SST-Hotsput output. So, we only need to justify the cylinder size based on obtained suspicious places and temporal intervals. For instance, see Table 3. As it can be seen, the main center of most likely cluster (*Bernalillo*) and the two sides of temporal interval of most likely temporal cluster (*85* and *89*) are detected through SST-Hotspot. So a cylinder with circle with center of *Bernalillo* with radius from the most far cluster members (*Los Alamos* and *Socorro*) and height of 85 to 89 (4 years as a size of time window) can be a good primary guess for limiting ST-Scan search space. For other clusters for instance, *Chaves and Roosevelt* and *Curry* we just need to put the center of cylinder surface on the *Chaves* county with each of obtained temporal hotspots from SST-Hotspot first priority list (86-89, 88-89, 88-89 and 83-89). In this case through only four tests we detect *Chaves* cluster in year 88-89.

Trends handling ability. If we configure ST-Scan for temporal trend justification, only one cluster is detected: *Los Alamos and Santa Fe in years 86-89*. ST-Scan in fact increases cases counts equal to 1.2 percent annually because of growing trend in population. For this reason its output is different from when this temporal trend justification is not performed. Our algorithm on 3D data could also detect this spatial hotspot *Los Alamos and Santa Fe* at the same time. However this was not detected in 2D and 5D data. One obvious reason is that this cluster in ST-Scan results is obtained based on adjusted counts for age, sex and race. So, 2D data which ignore all other attributes, definitely shall not detect that. 5D data is also as previously mentioned would not be able to detect that because of high fit difference between its population and cases models. The reason why 3D model could detect that is due this fact that it takes into account the correlation between all variables with time. During the decomposition process when we just select first eigenvector. So noises and other hidden factors are removed from the model. Thus, logically SST-Hostspot 3D shall detect clusters similar to ST-Scan.

5   Conclusion and Future works

In this paper we presented a novel approach for hotspot detection exploiting tensor decomposition and eigenvector elements matching techniques. The experimental results reveal the effectiveness of these techniques. Our approach is not a replacement for ST-Scan and its variants rather can be considered as a helpful method to reduce the search space in ST-Scan process, even though one could adapt this method independently for monitoring spatiotemporal variance in data. We also showed that how multi-way data analysis improves the quality of detection as it already was expected. One major of our approach drawback, is its inability about automatically connecting of the separated spatial and temporal hotspots with aim of identification of spatiotemporal hotspots. Another issue is that the current version is retrospective. Adapting algorithm for online detection would be another feature research direction.

References

1. Kolda, T.G., Baderz, B.W.: Tensor decompositions and application. SIAM Review **51** (2009) 455–500
2. Levine, N.: Crime mapping and the crimestat program. Geographical Analysis **38** (2006) 41–56
3. Zeng, D., Chang, W., Chen, H.: A comparative study of spatio-temporal hotspot analysis techniques in security informatics. In: Proceedings of the 7th International IEEE Conference on Intelligent Transportation Systems. ITSC '04, Tucson, AZ, USA, IEEE (2004) 106–111
4. Jain, A.K., Murty, M.N., Flynn, P.J.: Data clustering: A review. ACM Computing Surveys **31** (1999) 264–323
5. Birant, D., Kut, A.: St-dbscan, an algorithm for clustering spatial-temporal data. Data and Knowledge Engineering **60** (2007) 208–221
6. Kisilevich, S., Mansmann, F., Nanni, M., Rinzivillo, S.: Spatio-temporal clustering. In Maimon, O., Rockach, L., eds.: Data Mining and Knowledge Discovery Handbook. Springer US (2010) 855–874
7. Kulldorff, M., Athas, W.F., Feurer, E.J., Miller, B.A., Key, C.: Evaluating cluster alarms: A space-time scan statistic and brain cancer in los alamos. American Journal of Public Health **88** (1998) 1377–80
8. Johnson, R.A., W., W.D.: Applied Multivariate Statistical Analysis. Pearson Prentice Hall (2007)
9. Park, S.H., Lee, K.M., U., L.S.: A line feature matching technique based on an eigenvector approach. Computer Vision and Image Understanding **77** (2000) 263–283
10. Turk, M., Pentland, A.: Eigenfaces for recognition. Journal of Cognitive Neuroscience **3** (1991) 71–86
11. Turk, M., Pentland, A.: Face recognition using eigenfaces. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. CVPR '91, IEEE (1991) 586–591
12. Neill, D.B.: Detection of spatial and spatio-temporal clusters. PhD thesis, Department of Computer Science, Carnegie Mellon University, Pittsburgh, PA, USA (2006)
13. Shapiro, L.S., Brady, J.M.: Feature-based correspondence: An eigenvector approach. Image and Vision Computing **10** (1992) 283–288
14. Bader, B.W., Kolda, T., et al.: Matlab tensor toolbox version 2.5. `http://www.sandia.gov/~tgkolda/TensorToolbox` (2012) Accessed: December 2012.
15. Sun, J.: Incremental tensor analysis. `http://www.dasfa.net/wiki/index.php?title=Jimeng_Sun` (2012) Accessed: December 2012.
16. Mathworks: Mapping toolbox. `http://www.mathworks.com/products/mapping` (2012) Accessed: December 2012.
17. Kulldorff, M.: Brain cancer incidence in new mexico. `http://www.satscan.org/datasets/nmbrain/index.html` (2012) Accessed: December 2012.
18. Research, R., et al.: Postgis. `http://postgis.refractions.net` (2012) Accessed: December 2012.