

Modeling and Comparing Spatiotemporal Events

Kristin Eickhorst, Peggy Agouris, Anthony Stefanidis
Department of Spatial Information Science and Engineering
University of Maine
348 Boardman Hall
Orono, ME 04469-5711
{snoox, peggy, tony}@spatial.maine.edu
<http://dipa.spatial.maine.edu/>

Abstract

Spatiotemporal helixes constitute a novel method we developed for modeling changes in an object over time. Changes in both an object's trajectory and its outline can be represented with these helixes. In addition, spatiotemporal helixes can be compared to determine whether multiple objects have undergone similar changes. In this paper, we present a method by which these helixes may be compared, and then discuss possible actions to take after results have been obtained. Helixes of objects with similar behaviors may be generalized into a single helix that represents the behavior of the group of objects. In addition, helixes of objects that stop behaving in a similar manner over time can be split into discrete helixes. These aspects of helix definition and behavior are potentially useful to digital government related activities involving large geospatial datasets including objects moving or phenomena evolving over time.

1. Introduction

In the image processing community, it is becoming increasingly important to store only that data which is absolutely essential to the application at hand. This is especially true when dealing with large datasets of multiple images collected over time. Storing all of the data that is collected by a sensor would be cost-prohibitive. In object-tracking applications, there is thus a need for modeling methods that extract only the most important information about an object's movement. *Spatiotemporal helixes* are one such method of spatiotemporal modeling that was introduced in [1, 2] and which has been useful in isolating specific information about an object's movements and deformations over time.

The concept of spatiotemporal helix can be potentially quite useful for a variety of applications, including digital government related activities that deal with geospatial data. For instance, while not originally developed with this purpose in mind, helixes have been of interest to researchers at NASA, who are looking for a way to model the movement of invasive plant species throughout the Colorado Plateau. Species like cheatgrass, Canada thistle, and tamarisk, that are not native to the area, have for a number of years been taking over the habitat of endemic species like cottonwood. The challenge currently faced by the Invasive Species Forecasting System team (<http://bp.gsfc.nasa.gov/index.html>) is to come up with a way of tracking species movements and making predictions about future areas of infestation. Helixes are useful for this effort as they model relevant object information extracted from satellite imagery.

While there are already methods for tracking and representing objects' behaviors over time, such as lifelines [3, 4] and video summarization programs [5, 6], spatiotemporal helixes contribute in a different manner. They are integrated models of an object's centroid trajectory and its outline's expansion/contraction over time. Helixes are not simply a visualization mechanism, but also incorporate database and data storage issues in order to allow the user to query for particular object behaviors of interest.

The members of the Invasive Species Forecasting System have been interested in helices because they offer the potential to model species movements in a more detailed and helpful manner than is currently possible with available technologies. The most interesting aspect of the helices for many of these researchers is that they are able to model splitting and merging of polygons that represent a given species. This helps them see when behaviors are similar enough that conclusions can be drawn about a group of objects instead of a single entity. These behaviors can then be modeled in a more detailed fashion using linear regression and kriging techniques [7]. There is potential for other government agencies to use these helices as well if they wish to model any events over time, such as the spread of crime events over a geographic area or the epidemiology of a disease.

In order to demonstrate the effectiveness of helices for this purpose and other similar endeavors, this paper first details the spatiotemporal helix itself. It discusses how the components that are extracted can be useful in generalizing detail for storage in a database system. Once the helices have been discussed, metrics for comparing multiple helices are presented. The helices that have been compared are then looked at in a group to see whether aggregation or splitting is recommended. Finally, new experiments are presented in order to demonstrate the effectiveness of the process, and recommendations are made for future work.

2. The Spatiotemporal Helix

The spatiotemporal helix model is constructed in a four-step process: 1) find the center of mass for the object at each time instance, 2) detect changes in the object's outline in each cardinality quadrant, 3) construct a self-organizing map (SOM) that follows the trajectory of the object while picking out only those nodes which are necessary to generalize the object's behavior and forming a "spine" for the helix, and 4) add information about outline changes to the spine with "prongs" that show expansion or contraction. The remainder of this section will detail the performance of these tasks.

As an example, the following dataset has been used, consisting of five polygons that have been created synthetically on a 400x400 pixel grid. This dataset is representative of the kind of data acquired by the satellite sensors that NASA uses in its Invasive Species project, and spatiotemporal data that other governmental agencies may obtain for their GIS applications (Figure 1). The polygons in these five frames represent temporal snapshots in the spatial evolution of an object or a phenomenon over time. In our example, this would give the user an idea of the size and shape of a tamarisk polygon in each frame. The sensor data would need to undergo an object extraction process before the polygons could be used as input for our process, but this is outside the scope of the current paper. For more information on our relevant activities in automated object extraction, the reader is referred to [8-10].

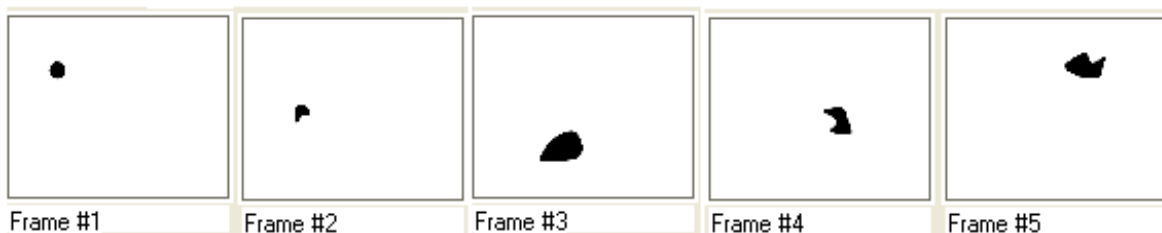


Figure 1: Five sample frames used for input in spatiotemporal helix extraction

In the first stage of helix construction, the object’s center of mass is extracted in each frame and plotted on a three-dimensional grid. Each asterisk stands for the location of the object at a given time instance. For example, assume that each of the frames used in this example was taken after a ten-minute delay. The first frame is linked to time $t=10$ and the fifth frame is linked to $t=50$. In this stage, a rudimentary trajectory is also constructed by linking the centers of mass for each frame (Figure 2). This gives us a detailed view of the positions of the tamarisk polygon at each time instance as well as its trajectory.

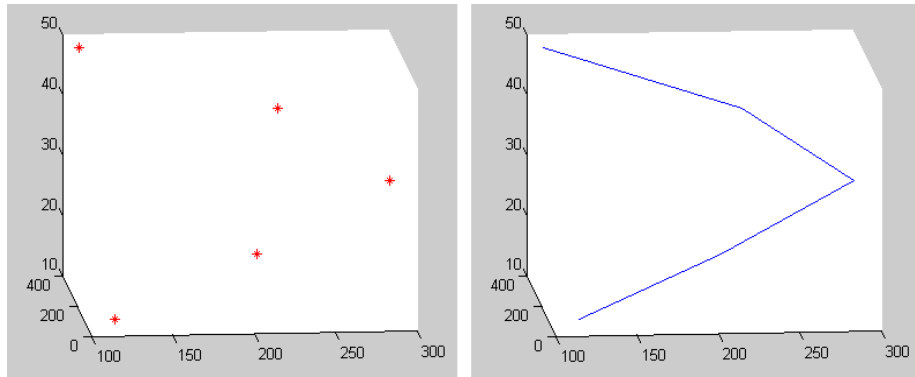


Figure 2: Object’s center of mass and rough trajectory

The second stage divides the object into four quadrants, based roughly on the cardinal directions of north, south, east, and west. This is done at each time instance, and the center of mass found in the first step is used as the origin for each division (Figure 3). Once this has been accomplished, the object is then compared to itself in order to discover whether there has been an expansion or contraction during each time interval. For instance, the object grows significantly between frames 2 and 3, and this leads to an increase in area for all four quadrants. This may occur in our example if the tamarisk polygon encounters additional resources that it needs for growth and thus expands in all directions. This change will be quantified in the final step of helix construction, which will be discussed later in this section.



Figure 3: Object divided into cardinality quadrants in each frame

The third stage is concerned with construction of a Self-Organizing Map (SOM) that generalizes the trajectory or “spine” of the helix by picking out specific (x, y, t) locations where changes such as rapid acceleration, deceleration, or rotation occur and marking them with nodes. A SOM is a neural network solution that organizes nodes into an ordered sequence through competitive learning [11]. In this example, the object is moving at a fairly uniform pace, so it does not experience much acceleration or deceleration. The major change is in rotation, which happens most notably at frame 3, the apex of the object’s trajectory.

When we ask for a generalized picture of the spine with 4 nodes we can see that the nodes for frames 1 and 2 from Figure 2 are merged into a single node to save space (Figure 4). When we ask for 3 nodes, only frame 3 retains its original node. This step decreases the number of nodes used to define the tamarisk polygon's location within the time interval of interest, and consequently leads to a reduction in the amount of space needed to store this data while maintaining the most important characteristics of the object's spatiotemporal behavior. For more detailed information on our relevant work on SOMs please see [10].

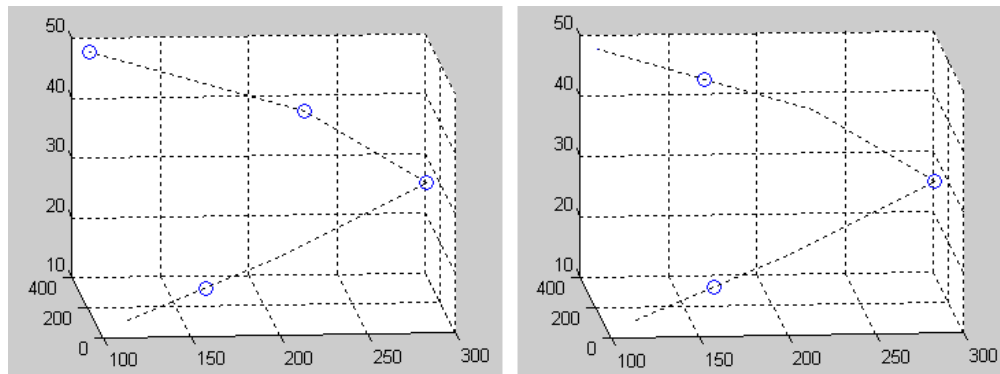


Figure 4: SOMs constructed with 4 (L) and 3 (R) nodes

The final stage in helix construction is to move each extracted node to the closest position recorded in the frames. For example, when four nodes are extracted in the SOM process, three of the nodes are located at the object's position in frames 3, 4, and 5. The fourth node is located between the object's positions in frames 1 and 2, but is closer to frame 2 (Figure 4). Thus, when constructing the helix, our algorithm places the final nodes in frames 2, 3, 4, and 5 (Figure 5). When examining the SOM of 3 nodes, we end up with final node placement in frames 1, 3, and 5. In our example, this would select specific frames from the original dataset, and use only them to define the placement of the tamarisk polygon over time. It is more accurate than using the node placements from the third step, because it does not create interpolated positions of the polygon, but uses locations that were already part of the dataset.

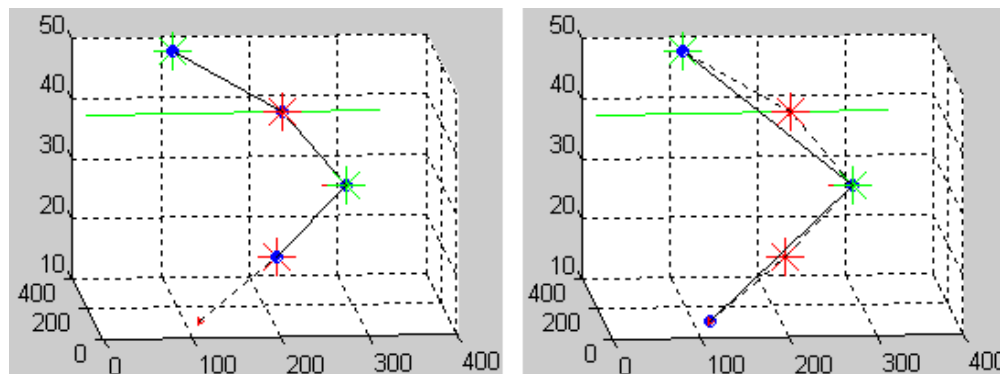


Figure 5: Complete helices for 4(L) and 3(R) nodes with spines and prongs

In addition to selecting the most important object instances that should be recorded in our database, the fourth stage in our helix construction process also examines the changes in expansion or contraction that were recorded between frames and compares them to a user-defined threshold. In this example, the threshold has been set at 20%. The largest change that was found in our example dataset occurs between frames 3 and 4, where there is a large reduction of area in the west quadrant and a slightly smaller reduction in the east quadrant. This is represented in our helices by a long line emerging from the “west” side of the node at frame 4 and a shorter line on the “east” side of the same node. This would indicate that the tamarisk polygon has undergone the most significant change in outline between these two frames.

In our spatiotemporal helix model the lines that are added in this stage are represented as “prongs,” which are indicated as arrows either pointing out of the helix spine (for expansion) or into it (for contraction) as can be seen in Figure 6. Each prong has its own time stamp, which enables cross-referencing in the database to determine its spatial location. This allows for storage of a single variable (t) instead of a set of variables (x, y, t) and reduces the amount of space needed in the database. A single time instance can have more than one prong attached to it, as expansion and/or contraction may occur in each of the four quadrants of interest.

Once the derivation of helices is complete, the next step is to compare them to each other in order to determine similarities or differences that exist between them and which refer to the behavior of the objects or phenomena they represent. Similarity metrics have been developed for this purpose, and they look at both the nodes and prongs within the helix in order to draw conclusions.

3. Similarity Metrics

There are two types of similarity metrics that can be used to compare helices. The first of these is an abstract method, in which presence or absence of specific node and prong qualifiers is compared. Two helices are compared at each node in order to determine whether both are exhibiting similar behaviors. If both are accelerating or decelerating at a given instant, then a value of 0 is given to that pair. If one is accelerating and the other is decelerating, a value of 2 is assigned (Table 1). In this way, helices with many dissimilar pairs will have a higher final comparison value, and two helices that are exactly alike will have a value of zero for their comparison. The same holds true for comparisons of rotation (clockwise v. counterclockwise) and prong types (expansion v. contraction).

Helix 1 ↓	Helix 2 →	Accelerate	Constant	Decelerate
Acceleration		0	1	2
Constant		1	0	1
Deceleration		2	1	0

Table 1: MST Cost metrics for comparing qualifier attributes of acceleration and deceleration

The other type of comparison is quantitative in nature, with specific differences computed between nodes and prongs. In this case, instead of a somewhat arbitrary value of “2” assigned to a pair of dissimilar nodes, the angle of acceleration and the angle of deceleration are compared by taking the absolute value of the difference between the two angles. Similar differences are found between angles of rotation and the magnitudes of expansion or contraction. In this case, the following equation is utilized:

$$Sim_r^c = a_n \sum (n_r^i - n_c^j) + a_p \sum (p_r^i - p_c^j) + a_q \sum (q_r^i - q_c^j) + a_r \sum (r_r^i - r_c^j) + a_a \sum (a_r^i - a_c^j)$$

where:

- $(n_r^i - n_c^j)$ expresses the Euclidean spatiotemporal distance among a reference and a corresponding candidate node, aggregated across all nodes,
- $(p_r^i - p_c^j)$ expresses the Euclidean spatiotemporal distance among a reference and a corresponding candidate prong, aggregated across all prongs,
- $(q_r^i - q_c^j)$ expresses the difference in velocity gradient or rotation among corresponding nodes, aggregated across all nodes,
- $(r_r^i - r_c^j)$ expresses the difference in deformation magnitude among corresponding prongs, aggregated across all prongs,
- $(a_r^i - a_c^j)$ expresses the difference in deformation angle among corresponding prongs, aggregated across all prongs,
- a_n, a_p, a_q, a_r, a_a are the corresponding relative weights for each component, with $a_n + a_p + a_q + a_r + a_a = 1$

We normalize all quantities by dividing their actual values by their range, so that in this case, a value of 1 is assigned to the most dissimilar helix pairs and a value of 0 is given to pairs that are exactly the same. Once a degree of similarity has been determined, whether by abstract or quantitative methods, we can decide whether these helixes belong in a group or should be kept separate. Work related to similarity that may be of interest includes [12-14].

4. Grouping Helixes

If the helixes in question are sufficiently similar, then it may be useful to group them together into a single entity and to express the behavior of their component objects with an “aggregate helix.” This could be useful in our invasive species scenario if several polygons of tamarisk are evolving in a similar manner and we want to draw more general conclusions about their overall behavior. The aggregate helix that is created could then be used for predictions about the future behavior of all tamarisk polygons that begin in a similar way to the first few nodes and prongs of the aggregate (Figure 6). The user selects the level of similarity that must be reached in order to justify this decision, with more detailed applications needing helixes with comparison values approaching zero.

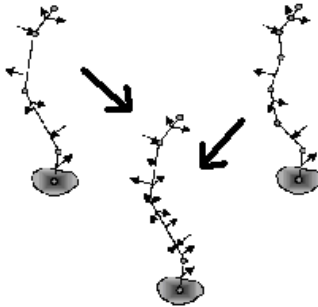


Figure 6: Aggregate helix as formed from individual helixes

When low enough values are found, corresponding node and prong locations of the individual helixes can be averaged so that the new aggregate helix is a composite of the original helixes. Attributes that are associated with each node and prong can be calculated in a variety of ways, including simply taking the

average of all values for each instance, looking for the minimum or maximum value, or using categorical rules in order to choose the best value for a given application. For instance, when making predictions about future growth of tamarisk, the user might want to take the maximum values of expansion and contraction so that there is no underestimation of overall growth rates. On the other hand, the minimum values for acceleration might be desired, in order to specify at least how far downriver the tamarisk groves will be likely to move in any given year. This could be useful for conservation planning purposes.

Helixes may not always need to be grouped together in order to get the most information out of a dataset. There may be occasions in which the helixes start out in a similar fashion, thus making it useful to construct an aggregate helix in order to express their common traits. However, over time they may diverge in value enough to merit individual treatment once again. This could happen in our tamarisk example if one part of the larger polygon crosses a sandbar in a river and moves to the opposite shore. This new polygon would be an offshoot of the previous instance and would retain some characteristics, but would be better expressed by a new helix that splits off from the aggregate. Key instances in which splitting should occur can be calculated by comparing the deviation of node and prong values from the aggregate average and splitting when a user specified threshold is crossed.

5. Additional Considerations and Current Work

Another factor that we are currently exploring and which would be useful in deciding whether to merge or split helixes is visualization of the node and prong values with colors, various levels of shading, fuzziness, or other overlays. This information would be intended to supplement the quantitative values of the helix components, and could be useful for quick decision making with regards to helix activities. For instance, if one wanted to be visually alerted to nodes where acceleration was occurring at a very fast rate, these nodes could be colored with red or another bright color. This might help conservationists see which polygons of tamarisk are moving the fastest and concentrate their attention and efforts in that location in order to stop the rapid spread of the invasive species. If metadata about accuracy is present, this could be expressed in the relative crispness or fuzziness of the node and prong representations [15].

In addition to visual representations of helixes, we are also working on utilizing aggregation and splitting techniques when dealing with text and other data formats that may be associated with the helixes. These different types of data can complement and enhance the helix capabilities. As an example, we might have metadata about accuracy, such as that mentioned in our discussion of visualization. The level of metadata that deals with specific horizontal and vertical accuracies of angles is quite detailed. The FGDC lists a set of standard metadata that is set up in a hierarchical fashion. By exploiting this construction we can gradually zoom in to the level of detail that is required, both within the data itself and within the corresponding metadata [16, 17]. Thus, the user would not need to see metadata on accuracy or fuzzy visualizations until a very detailed view of the helix itself is desired.

6. Experiments

In order to test our spatiotemporal helix approach and its usefulness for applications like NASA's invasive species project, we have performed experiments on helix generation, similarity metrics, and aggregation of helixes. These experiments have yielded results that confirm that our system can be beneficial to the broader digital government community. For helix generation, we have constructed datasets of 700 frames in addition to the sample dataset of five frames that was presented earlier, and have used differing thresholds to determine the number of nodes and prongs that define the helix. Figure 7

shows two helixes that were constructed during this phase. Both have 17 nodes, but helix “a” has more prongs than helix “b.” Their respective prong thresholds are 10% and 20%.

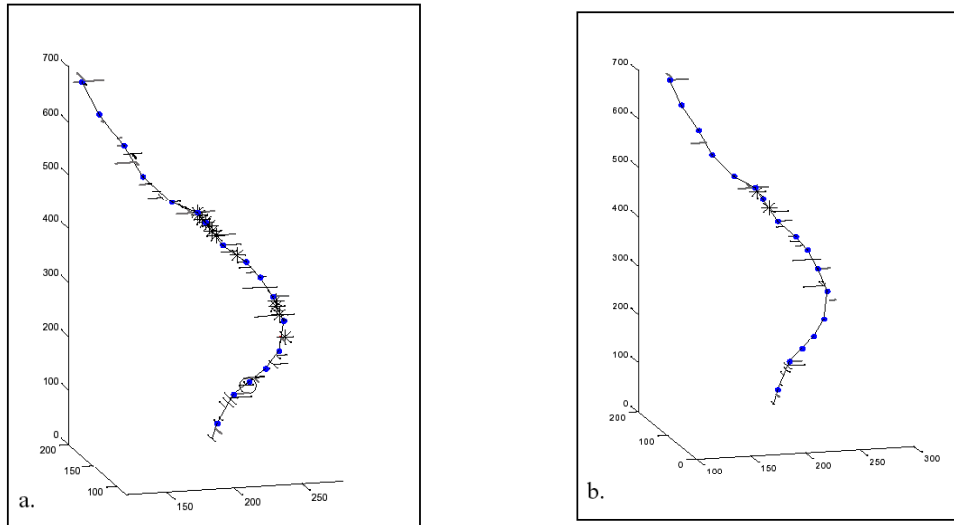


Figure 7: Helixes constructed from differing thresholds

One of the most interesting experiments that we performed was to use the prong information that we had gathered to reconstruct the boundaries of an object as it should appear at $t=700$ sec. We used only the image of the object at $t=0$ sec, and the expansions and contractions as indicated by the prong magnitudes and angles, and then compared these results to the actual object boundaries in frame 700. We found that with our dataset, we were able to reconstruct the object with 83% accuracy when using a prong threshold of 20% and with 94% accuracy when using any prong threshold below 15%. There seems to be a definite level of prong definition beyond which no additional benefit is gained in storing the extra information. See [14] for a more detailed discussion of this topic.

Another type of experiment that we have conducted involves the computation of similarity indices using the metrics discussed earlier. When comparing two helixes, we have one reference helix and one candidate helix (Figure 8). When examining a node on the reference helix, we do not simply look for a match at the same time instance on the candidate helix, but expand our time window to account for variations that may have occurred while obtaining the dataset. Thus we are not looking solely at time t_2 for a match, but anywhere between t_1 and t_3 .

With this in mind, we created a dataset of 100 helixes, consisting of an average of 19 nodes and 7 prongs each, and used both the abstract and quantitative metrics to compare reference helixes to a pool of candidate helixes. We noted the time that it took to run each of these queries, and found that on average, the abstract query took 2 seconds, while the more mathematically intensive quantitative query took 4 seconds. These are very encouraging results as digital government applications in the geospatial realm are often large-scale efforts where computational times matter, and we anticipate working with even larger helixes in the invasive species project.

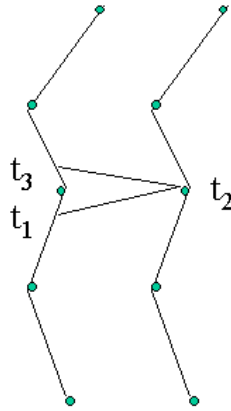


Figure 8: Comparison of reference helix (right) to candidate (left)

Finally, the aggregation and splitting of helixes is our current focus, and we are developing an environment that calculates average node and prong values and notifies the user when an aggregation or split may be beneficial for representing the behavior of objects of interest. At this point we have developed a way for two objects to be extracted from a sequence of images and their helixes computed separately. The database of nodes and prongs that is developed in this step can then be queried for similarity of attributes and if the similarity values are lower than a threshold of the user's choice, then these helixes can be combined into a single helix with a combination of attributes. This stage of the research is still in development.

7. Conclusions

Spatiotemporal helixes represent a new and promising theory that we have developed for modeling and analyzing change in geospatial applications and other large-scale data-intensive projects in various fields like epidemiology and biodiversity informatics. These projects all share a need to condense the amount of significant data being maintained in their databases and facilitate the tracking, detection of similarities, and/or discovery of behavioral patterns in the evolution of seeming unrelated phenomena. The spatiotemporal helix enables the accomplishment of these goals, and helix construction and generalization, as well as similarity comparisons, and aggregation or splitting are all key aspects of the process as presented in this paper.

Acknowledgements

NSF has supported this work through Digital Government Award #9983432. Kristin Eickhorst's work has also been funded in part by NASA through a Maine Space Grant Fellowship and the generous support of Goddard Space Flight Center.

References

1. Stefanidis, A., P. Agouris, and P. Partsinevelos, *SpatioTemporal Helixes for Event Modeling*, in *National Conference on Digital Government Research*, 2002, Los Angeles. p. 219-224.
2. Agouris, P. and A. Stefanidis, *Efficient Summarization of Spatiotemporal Events*, *Communications of the ACM*, 2003, **46**(1): p. 65-66.
3. Plaisant, C., B. Milash, A. Rose, S. Widoff, and B. Shneiderman, *LifeLines: Visualizing personal histories*, in *CHI '96*, 1996, New York: ACM. p. 221-227.
4. Hornsby, K. and M. Egenhofer, *Modeling Moving Objects over Multiple Granularities*, *Annals of Mathematics and Artificial Intelligence*, 2002, **36**(1-2): p. 177-194.
5. Pope, A., R. Kumar, H. Sawhney, and C. Wan, *Video Abstraction: Summarizing Video Content for Retrieval and Visualization*, in *Proc. 32nd Asilomar Conference of Signals, Systems & Computers*, 1998. p. 915-919.
6. Zhou, J.Y., E.P. Ong, and C.C. Ko, *Video Object Segmentation and Tracking for Content-Based Video Coding*, in *IEEE International Conference on Multimedia and Expo (III)*, 2000. p. 1555-1558.
7. Diggle, P.J., J.A. Tawn, and R.A. Moyeed, *Model-based Geostatistics*, *Applied Statistics*, 1998, **47**(3): p. 299-350.
8. Agouris, P., A. Stefanidis, and S. Gyftakis, *Differential Snakes for Change Detection in Road Segments*, *Photogrammetric Engineering & Remote Sensing*, 2001, **67**(12): p. 1391-1399.
9. Agouris, P., K. Beard, G. Mountrakis, and A. Stefanidis, *Capturing and Modeling Geographic Object Change: A SpatioTemporal Gazetteer Framework*, *Photogrammetric Engineering & Remote Sensing*, 2000, **66**(10): p. 1241-1250.
10. Doucette, P., P. Agouris, A. Stefanidis, and M. Musavi, *Self-Organized Clustering for Road Extraction in Classified Imagery*, *ISPRS Journal of Photogrammetry and Remote Sensing*, 2001, **55**(5-6): p. 347-358.
11. Kohonen, T., *Self-Organizing Maps*. 1997: Springer-Verlag.
12. Vlachos, M., D. Gunopulos, and G. Kollios, *Robust Similarity Measures for Mobile Object Trajectories*, in *DEXA 2002, 5th International Workshop on Mobility in Databases and Distributed Systems*, 2002, Aix-en-Provence, France. p. 721-728.
13. Stefanidis, A., P. Agouris, M. Bertolotto, J. Carswell, and C. Georgiadis, *Scale and Orientation-Invariant Scene Similarity Metrics for Image Queries*, *International Journal of Geographical Information Science*, 2002, **16**(8): p. 749-772.
14. Stefanidis, A., K. Eickhorst, P. Agouris, and P. Partsinevelos, *Modeling and Comparing Change using Spatiotemporal Helixes*, in *ACM-GIS 2003*, 2003, New Orleans, LA: ACM Press. p. 86-93.
15. Pang, A., C. Wittenbrink, and S. Lodha, *Approaches to Uncertainty Visualization*, *The Visual Computer*, 1997, **13**(8): p. 370-390.
16. Eickhorst, K., *Hierarchical Structures for Video Query Systems*, in *DG.O 2002 Proceedings, National Conference on Digital Government Research*, 2002, Los Angeles. p. 475-479.
17. Eickhorst, K. and P. Agouris, *On the Use of Hierarchies and Feedback for Intelligent Video Query Systems*, in *Proceedings of ISPRS 2002 Symposium of Commission IV [CD-ROM]*, 2002, Ottawa.