Contour Generalization Based Nearest Neighbor Approximation for Mining Uncertain Data

M. Kalavathi
Head, Department of Computer Science
Government Arts and Science College
Komarapalayam.- 638 183.
Email : Kalavathisakthi@gmail.com

Dr. P. Suresh
Head, Department of Computer Science
Salem Sowdeswari College
Salem - 636 010.
Email sur_bh0071@rediffmail.com

Abstract-Recently, data uncertainty has developed into an accepted topic in database and data mining area due to the enormous amount of uncertainty involved. The previous methods extend traditional spatiotemporal tolerance for coping with data uncertainty. The methods used Continuous Range Queries and Superseding Nearest Neighbor search, thus rely on conventional multidimensional index. Such methods cannot handle uncertain objects to NN classifier based continuous queries. At the same time, Nearest Neighbor, which is essential characteristics of uncertain objects, has not been considered in measuring contour segments in uncertain data. In this paper, a Contour Generalization based NN approximation (CG-NN Approximation) technique is developed, with the objective of solving spatiotemporal tolerance problem. Contour Generalization technique reduces the overhead count on analyzing the NN based continuous queries and uses a distance function to produce high dimensionality on spatiotemporal queries. An optimal Contour Generalization algorithm produces an optimal Contour Generalization for every input query points.CG-NN Approximation considers various types of queries on uncertain database, determines the query type, and distance function to provide solution for queries on uncertain data. Contour Generalization approximates a polygonal contour, so that it is sufficiently close (i.e.,) nearest neighbor and has less contour segments. The lesser contour segments takes less storage-space and thus minimizing the computational overhead. Experimental evaluation is measured in terms of computation overhead, spatiotemporal tolerance, storage space and query processing efficiency. Experimental analysis shows that CG-NNA technique is able to reduce the computation overhead for continuous queries on uncertain data by 19.35% and reduce the average operation cost by21.24% compared to the state-of-the-art works.

Keywords-Continuous Range Queries, Superseding Nearest Neighbor, Contour Generalization, NN approximation, Spatiotemporal, Polygonal Contour

I. INTRODUCTION

Due to the intrinsic property of uncertainty, numerous interesting queries have been raised for diverse purposes. Among diverse approaches for modeling uncertain data, many research papers have been contributed in this domain. Processing Continuous Range with Spatiotemporal Tolerance (PCR-ST) [1] relaxed query's accuracy requirements through well-defined query semantics. Superseding Nearest Neighbor on Uncertain Spatial Database (SNN-USD) [2] used multi-dimensional index aiming at producing SNN core without deriving the whole superseding graph.

Reverse Nearest Neighbor (RNN) [3] improved the pruning efficiency using sampling-based approximate algorithm. Voronoi diagrams and R-Tree index [4] was applied on uncertain data with the objective of reducing pruning overheads. In [5], Probability Distribution Similarity was applied on uncertain data aiming at improving the efficiency and scalability using Gauss transform technique. With the objective of efficiently evaluating the trajectory queries in [6], u-bisector was applied on imprecise location data. Though efficiency was improved in all the above said methods, but computation overhead was compromised. To address issues related to computation overhead, CC-NNA technique used Contour Generalization technique.

II. RELATED WORK

Location-based application uses the position of mobile device to identify the current user location and customizes the results of the query to include neighboring points of interests. In [7], a service architecture based on user centric was designed with the objective of improving privacy between the users and the service providers. In

[8], dynamic programming techniques were applied aiming at reducing the error. Clique Tree Propagation [9] algorithm was introduced to reduce the computation cost by considering query workload.

In [10], Chebyshev method was applied to similarity measurement for time series data improving the noise. To reduce the computational cost, another method based on uncertain graph streams called Node Neighbor Tree [11] was designed. Rough set theory Clustering Technique (RCT) [12] was applied to minimize the forecasting period using Echo State Network (ESN). The above said methods though minimized the computation cost the query processing efficiency remained unsolved. To address this, CC-NNA technique used Optimal Polygon Contour.

Many research works were conducted on efficiently addressing range queries over uncertain objects. In [13], pivot-based indexing technique was applied aiming at improving the speed of range query computation. A review of data mining techniques on uncertain data was presented in [14]. Another method called Probabilistic Range Query [15] was introduced aiming at minimizing the preprocessing time in a significant manner. Feature extraction issue related to query handling was introduced in [16] using sparse coding resulting in improving the predictive performance of the query results being retrieved.

Segmentation and sampling methods were used in [17] for Moving Object Databases (MOD) aiming at improving the query similarity rate. In [18], optimal set of sub-queries were retrieved in an efficient manner using cost-based query planning model. Effective index was performed on uncertain objects using U-Quad tree [19] resulting in the improvement of cost. Top-k queries were addressed in [20] using totally and partially explained sequences.

In this paper, we formalize the problem of contour generalization based NN approximation on uncertain data using the polygon contour. We present a new probabilistic contour generalization based NN approximation technique that employs (i) Contour Generalization technique exploiting the query type and distance function (ii) Optimal Polygon Contour technique based on if-then-condition. Our contributions in this work are two-fold. First, we propose a novel technique Contour Generalization that identifies the Nearest Neighbors aiming at reducing the computation overhead and spatiotemporal tolerance. Second, an optimal polygon contour algorithm is designed based on the ‘intersect’ between the users and queries issued by them to minimize the storage space and therefore the query processing efficiency.

The rest of the paper is organized as follows: Section 2 defines the problem of spatiotemporal and illustrates its characteristics by developing an algorithm that computes the optimal contour generalization for every input query points on mining uncertain data. Section 3 experimentally evaluates our solutions with the aid of census income dataset. Section 4 discusses in detail using table and graph form. Finally, Section 8 concludes the paper.

III. DESIGN OF CONTOUR GENERALIZATION BASED NN APPROXIMATION TECHNIQUE

Uncertain data is used in several real applications, such as sensor network monitoring, object recognition, location-based services (LBS), and moving object tracking. In this section, a Contour Generalization based NN approximation (CG-NNA) technique is presented. For improving the query processing efficiency and reducing execution time per iteration and percentage of tolerance level, CG-NNA is designed. The architecture of our proposed technique is presented in Fig. 1.

As shown in the figure 1, architecture diagram of Contour Generalization based NN approximation technique comprises of two parts. The first part Contour Generalization technique is designed at aiming to reduce the computation overhead and spatiotemporal tolerance based on the Euclidean Distance function and Query type. The second part Optimal Polygon Contour using optimal contour algorithm results in the improvement of query efficiency.

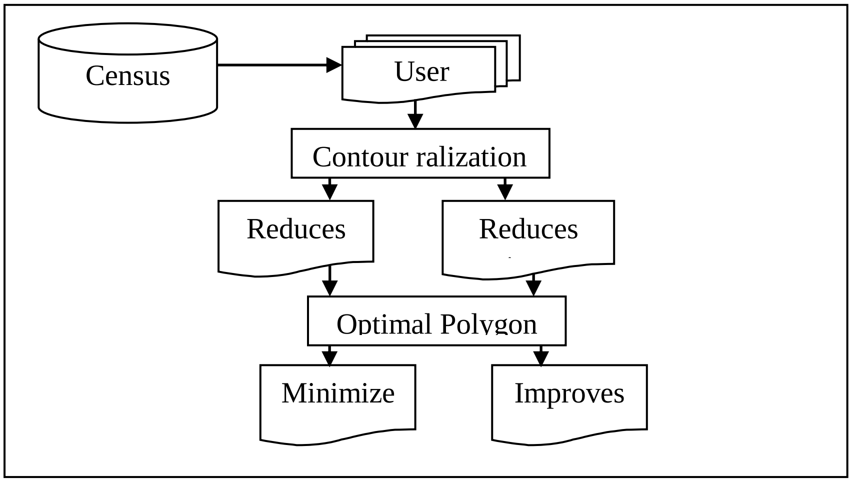

Figure 1. Architecture diagram of Contour Generalization based NN approximation technique

A. Design of Contour Generalization technique

A spatiotemporal query is a time-stamped sequence of queries representing the information related to space and/or time. In this section, the design of Contour Generalization technique to reduce computation overhead on spatiotemporal queries for every input query points on uncertain data is studied. Fig. 2 shows the block diagram of Contour Generalization technique.

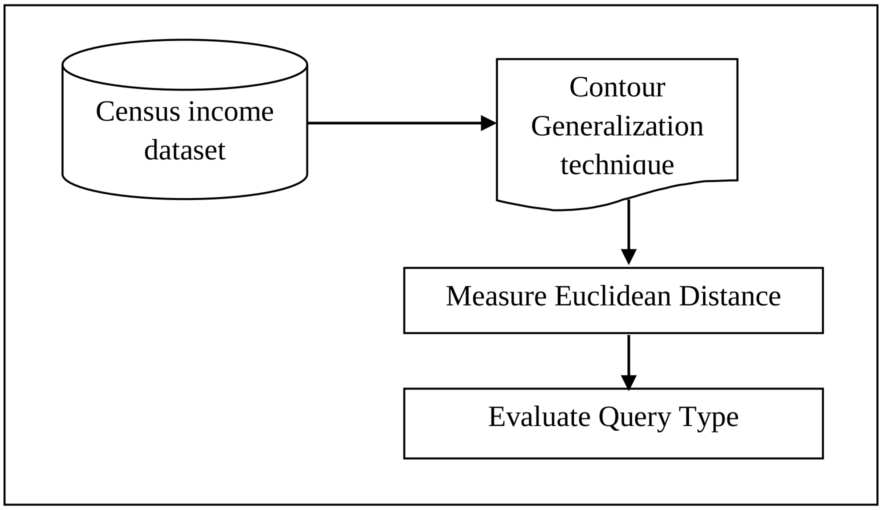

Figure 2. Block diagram of Contour Generalization technique

As shown in Fig. 2, the Contour Generalization technique initially measures the Euclidean distance between the user queries for every input query points. Next, the query type is further analyzed based on the spatiotemporal tolerance factor '∂' aiming at reducing the computation overhead and spatiotemporal tolerance on uncertain data. Let us consider a '$d\ dimensional$' query that is an ordered collection 'C' be formulated as given below

$$C = \{(T_1, Q_1), (T_2, Q_2), \dots, (T_n, Q_n)\} \quad (1)$$

In (1), 'n' is the length of the query 'Q' is an ordered collection, 'C' is uncertain data with time stamps '$T_1, T_2, \dots, T_n$' respectively. The objective of the proposed method CG-NNA is by using Contour Generalization technique, NN based continuous queries are analyzed, aiming at reducing the computation overhead over sliding window.

The Contour Generalization technique uses time-based sliding window that contains all queries that arrives within a fixed interval of time ‘T_i’. The advantage of using time-based sliding window in Contour Generalization technique is that it follows first come first go deletion format on uncertain data. This in turn ensures faster NN monitoring and therefore reduces the computation overhead.

Using time-based sliding window, the problem of retrieving similar queries using Contour Generalization technique on uncertain data is formulated as follows. Given a reference census income dataset extracted from UCI repository, ‘DS’, measure for distance ‘$Dist$’, a query ‘q’, the set of ‘Q_i’ queries is formulated as in (2).

$$Q_i = \{a \in DS \mid Dist\ (a, q\ )\} \tag{2}$$

For every input query points, the Contour Generalization technique identifies the Nearest Neighbors (NN) of ‘Q_i’ on uncertain data aiming at reducing the computational overhead on spatiotemporal queries.

In this paper we address the problem of managing spatiotemporal queries issued from the side of the user of the form ‘(i, j, T)’ on uncertain data. This indicates that a query ‘q’ will be at location with coordinates ‘(i, j)’ at time ‘T’. One of the key observations to produce high dimensionality on spatiotemporal queries is that ‘(i, j, T)’ is reduced and its computational overhead saved by approximating ‘(i, j, T)’ by interpolating the nearest neighbor (i.e., before and after) queries.

The contour generalization technique uses Euclidean distance (i.e. to identify the nearest neighbor) between the original queries issued by the user and the approximation is determined using simplification referred to as the error-tolerance. This in turn solves the spatiotemporal tolerance problem. CG-NN Approximation technique determines the distance function and query type for providing solution for queries on uncertain data.

By applying Euclidean Distance function, Contour Generalization technique reduces the overhead count on analyzing the NN based continuous queries.CG-NN Approximation uses a distance function to produce high dimensionality on spatiotemporal queries. Let ‘T_1, T_2’ be two timestamps of length ‘N’ and let ‘$User_1, User_2$’ be the corresponding queries issued by two users. Suppose ‘$User_1$’ issues ‘$x_0, x_1, \dots, x_n$’ and ‘$User_2$’ issues ‘$y_0, y_1, \dots, y_n$’, then the results of the queries retrieved using Euclidean Distance function is given as in (3).

$$Dist = (User_1, User_2) = \left(\sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}\right) \tag{3}$$

From (3), based on the query type and distance function, the proposed CC-NNA technique provides solution for queries on uncertain data in an efficient manner. The query type in CC-NNA technique is formulated with a distance function ‘$Dist$’, for the respective query ‘Q_i’, for every spatiotemporal tolerance factor ‘∂’ and is given as in (4).

$$QT = (Q_i, \partial, Dist) \tag{4}$$

From (3) and (4), CG-NN Approximation technique determines the query type ‘QT’, and distance function ‘$Dist$’ for providing solution for the queries on uncertain data. Fig. 3 shows the algorithmic description of Optimal Contour Generalization.

```
Input : Query ‘$Q_i = Q_1, Q_2, .., Q_n$’, Timestamp ‘$T_i = T_1, T_2, .., T_n$’, Dataset ‘$DS$’,
Output: Minimized computation overhead and spatiotemporal tolerance
Step 1: Begin
Step 2:     For each ordered Collection ‘$C$’
Step 3:         For each Query ‘$Q_i$’
Step 4:             Measure Euclidean Distance function using (3)
Step 5:             Evaluate query type using (4)
Step 6:         End for
Step 7:     End for
Step 8: End
```

Figure 3. Optimal Contour Generalization

The Optimal Contour Generalization algorithm includes two steps. For each ordered collection and query issued by the user, the first step measures the Euclidean Distance function to identify the common spatiotemporal query types. Based on the Euclidean Distance function, the second step evaluates the query type on uncertain data for every input query points. This in turn reduces the computation overhead and spatiotemporal tolerance problem in a significant manner.

B. *Design of Optimal Polygon Contour*

Besides time-based sliding window, the CC-NNA technique uses an Optimal Polygon Contour that approximates a polygonal contour, so that it is sufficiently close (i.e.,) nearest neighbor and has less contour segments. The lesser contour segments takes less storage-space and thus minimizing the computational overhead. Fig. 4 shows the block diagram of Optimal Polygon Contour.

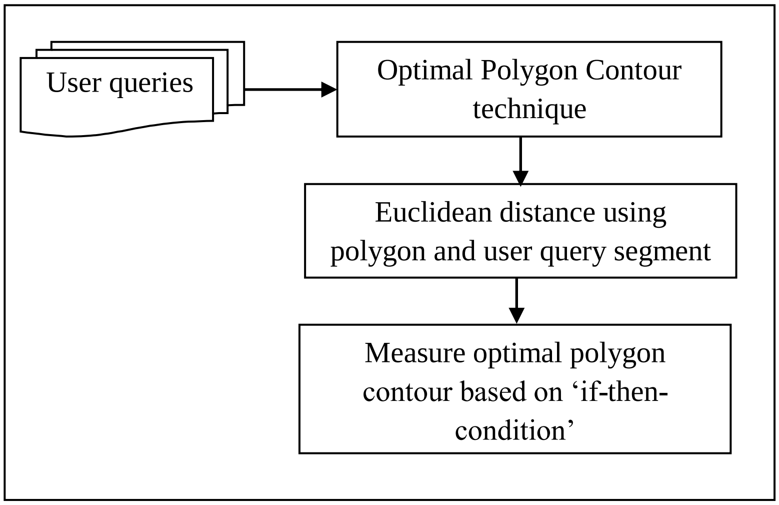
User queries
Optimal Polygon Contour technique
Euclidean distance using polygon and user query segment
Measure optimal polygon contour based on 'if-then-condition'

Figure 4. Block diagram of Optimal Polygon Contour

As shown in the above figure, the block diagram of Optimal Polygon Contour technique measures two factors. With the objective of minimizing the contour segment, the Optimal Polygon Contour technique initially obtains the Euclidean distance based on the polygon and user query segment. Next, 'if-the-condition' is used to approximate the polygon contour for the queries on uncertain data.

The Optimal Polygon Contour in CC-NNA technique is designed in such a manner that for any polygon 'P' if intersect '(Q', ∂, T_1, T_2)' is true, then there exists a time '$T_i \in (T_1, T_2)$' such that the expected location of the original query 'Q' at time 'T_i' is no further than '∂' from interior of 'Q'. On the other hand, if intersect '(Q', ∂, T_1, T_2)' is false, then for every '$T_i \in (T_1, T_2)$', the expected location of the original query 'Q' at time 'T_i' is outside of 'Q'.

Let '$P_i = (x_i, y_i, T_i)$ denote a query, and let '$User_i, USer_j$' denote the user query segment between the segments '$s_i = (x_i, y_i, T_i)$' and '$s_j = (x_j, y_j, T_j)$' of query 'Q_i' on uncertain data. Then the Euclidean distances between the polygon 'P' and the user query segment '$User_i, USer_j$' are defined as follows:

$$Dist\left(User_i,\ USer_j\right) = \sqrt{(x_i - y_i)^2 + \left(x_j - y_j\right)^2} \quad (5)$$

The Optimal Polygon Contour is then formulated as given below.

$$(Q \cap P) = Dist\left(User_i,\ USer_j\right) \quad (6)$$

$$if\ (Q \cap P) = True\ ,then\ Intersect\ (Q'\ ,\partial, T_1, T_2)\ \in True \quad (7)$$

$$if\ (Q \cap P) \neq True\ ,then\ Intersect\ (Q'\ ,\partial, T_1, T_2)\ \in False \quad (8)$$

From (7) and (8), the optimal polygonal contour is sufficiently close (i.e.,) nearest neighbor and has less contour segments. This in turn reduces the storage-space with minimal computational overhead. Fig. 5 shows the algorithmic description of optimal polygon contour model.

Algorithm shows the optimal polygon contour algorithm. The optimal polygon contour includes three important steps. In first step, for each user query at different time intervals, the distance between the user queries is measured. So that sufficiently close (i.e.,) nearest neighbor is identified and it has less contour segments. Then the optimal polygon contour is measured based on the ‘if-then-condition’. When the condition is true, the expected location of the original query ‘Q’ at time ‘T_i’ no more than spatiotemporal tolerance factor ‘∂’ from the interior of ‘Q. This helps to minimize storage space and improves the query processing efficiency. Otherwise, the location of the original query ‘Q’ at time ‘T_i’ is outside of Q. This helps to identify the uncertain data.

```
Input: User‘$User_i = User_1, User_2, \dots, User_n$’,polygon‘$P$’,
Query ‘$Query_i = Query_1, Query_2, \dots, Query_n$’, Timestamp ‘$T_i = T_1, T_2, \dots, T_n$’
Output:
Step 1: Begin
Step 2:     For each user $User_i$
Step 3:         For each query $Query_i$
Step 4:             For each timestamp $T_i$
Step 5:                 Evaluate the distance between the users (i.e. queries) using (5)
Step 6:                 Measure optimal polygon contour using (6)
Step 7:                 If $(Q \cap P) = True$ then
Step 8:                     $Intersect\ (Q'\ ,\partial, T_1, T_2)\ \in True$
Step 9:                 End if
Step 10:                If $(Q \cap P) \neq True$ then
Step 11 :                   $Intersect\ (Q'\ ,\partial, T_1, T_2)\ \in False$
Step 12:                End if
Step 13:            End for
Step 14:        End for
Step 15:    End for
Step 16 : End
```

Figure 5. Optimal Polygon Contour Algorithm

For example: we use census income dataset, the various types of persons on uncertain database with attributes details are selected for predict whether the user income exceeds 50 dollars/yr. For each user, the various attributes are taken into consideration for experimental. The optimal Polygon Contour is measured using Euclidean distances between the users. Due to, the nearest neighbors are identified and it has less contour segments. Next, ‘if-then condition’ is used to estimate the polygon contour for predict the user income. If the user is encircled within the polygon contour, then the user income is not exceeds 50 dollars/yr (i.e. less than or equal to 50 dollars/year). Otherwise, it predicts the uncertain data (i.e. user income exceed 50 dollars/yr).

CG-NN Approximation considers various types of queries on uncertain database, determines the query type, and distance function to provide solution for queries on uncertain data. Contour Generalization approximates a polygonal contour, so that it is sufficiently close (i.e.,) nearest neighbor and has less contour segments. The lesser contour segments takes less storage-space and thus minimizing the computational overhead.

IV. EXPERIMENTAL RESULTS

For efficient analysis of NN based continuous queries, the CC-NNA technique used census income dataset extracted from UCI repository. The following experiments are based on census income datasets, each consisting of a set of 14 attributes of both categorical and integer characteristics in nature. Additionally, we also applied our technique to two census income datasets where the prediction of uncertain data, income is measured to see whether it exceeds '50 $dollar/year$'.

The extraction of census income dataset was done by Barry Becker from the 1994 Census database. A set of reasonably clean records was extracted using the following conditions: ((AAGE>16) && (AGI>100) && (AFNLWGT>1) && (HRSWK>0)). The data was extracted from the census bureau database found at http://www.census.gov/ftp/pub/DES/www/welcome.html.

Prediction task is to determine whether a person makes over 50K a year. The fourteen attributes included in the experiments are age, workclass, fnlwgt, education, education-num, marital status, occupation, relationship, race, sex, capital-gain, capital-loss, hours-per-week and native-country.

In order to perform the experiments, our technique CC-NNA technique used six important attributes like, age, work-class, education, occupation, hours-per-week and native country to determine whether a person makes over 50K a year. Seven iterations were conducted with different age ranging between 40 years and 50 years conducted at different time periods.

V. DISCUSSION

In this section, we analyze our proposed technique CC-NNA with respect to computation overhead, spatiotemporal tolerance with respect to average operation cost, query efficiency and storage space. We also compare our proposed technique with well-known mining methods on uncertain data known as Processing Continuous Range with Spatiotemporal Tolerance (PCR-ST) [1] and Superseding Nearest Neighbor on Uncertain Spatial Database (SNN-USD) [2].

A. *Scenario 1: Analysis of computation overhead*

In this section to check the efficiency of CC-NNA technique, the metric computation overhead is evaluated and compared with the state-of-the-art works, PCR-ST [1] and SNN-USD [2].The computation overhead is the time taken to process the query for every input query points. The computation overhead is the product of the number of queries 'Q_i' and the time taken to process the queries '$Time\ (Q_i)$' respectively. The mathematical formulation for computation overhead is as given below.

$$CO = \sum_{i=1}^{n} Q_i * Time\ (Q_i) \tag{9}$$

From (9), the computation overhead 'CO' is evaluated and is measured in terms of milliseconds (ms). Lower the computation overhead, more efficient the method is said to be. To deliver with a detailed performance, in Table 1 we apply the number of queries and time taken to retrieve the query to obtain the computation overhead and comparison is made with two other existing methods, PCR-ST and SNN-USD respectively. Lower computation overhead results in the improvement of the technique.

TABLE 1. TABULATION FOR COMPUTATION OVERHEAD

No. of queries (Q)	**Computation overhead (ms)**		
	CC-NNA	**PCR-ST**	**SNN-USD**
5	1.82	2.48	2.91
10	3.56	4.16	4.46
15	5.87	6.47	6.77
20	7.85	8.45	8.75
25	4.32	5.02	5.32
30	6.55	7.15	7.45
35	8.13	9.03	9.33

A comparative analysis for computation overhead with respect to different number of queries was performed with the existing PCR-ST and SNN-USD is shown in Fig. 6. The increasing number of queries in the range of 5 to 35 is considered for experimental purpose on mining uncertain data. As illustrated in figure, comparatively while considering more number of queries, the computation overhead also increases, though betterment achieved using the proposed technique CC-NNA.

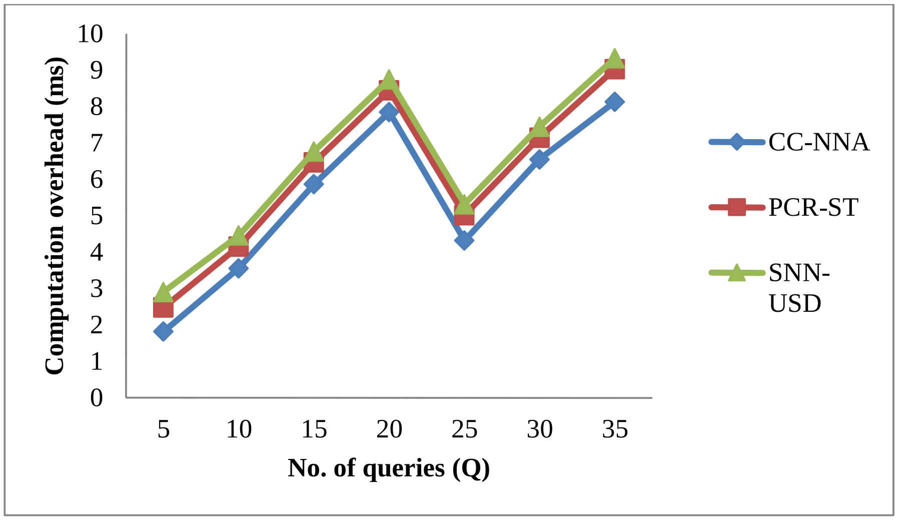

Figure 6. Measure of Computation Overhead

The measurement of computation overhead is comparatively reduced using the CC-NNA technique when compared to two other existing methods [1, 2]. This improvement in computation overhead is because of the application of Contour Generalization based on the Euclidean distance and the query type on analyzing the NN based continuous queries. Furthermore, a spatiotemporal tolerance factor '∂' introduced in CC-NNA technique is an ordered collection that in turn reduces the computation overhead by 15.34% and 23.37 % compared to PCR-ST and SNN-USD respectively.

B. Scenario 2: Analysis of spatiotemporal tolerance

Table 2 shows the average operation cost for seven different iterations with varied spatiotemporal tolerance rate '∂'. Based on the spatiotemporal tolerance, the average operation cost (i.e. for queries) 'AOC' is mathematically formulated as given below.

$$AOC = ST * QA \qquad (10)$$

From (10), the average operation cost 'AOC' is evaluated using the spatiotemporal tolerance 'ST' and the queries addressed 'QA'. The average operation cost is measured in terms of percentage (%).

The second evaluation metric considered for evaluating the effectiveness of the technique CC-NNA is average operation cost with respect to different spatiotemporal tolerance rate. Fig. 5 shows the elaborate comparison made with the existing two state-of-the-art works.

TABLE 2. TABULATION FOR AVERAGE OPERATION COST

Spatiotemporal tolerance ()	Average operation cost (%)		
	CC-NNA	**PCR-ST**	**SNN-USD**
0.1	71.35	60.89	51.35
0.2	75.87	63.82	52.75
0.3	79.35	67.30	59.25
0.4	68.45	56.40	49.35
0.5	72.87	60.82	53.75
0.6	75.33	63.28	56.23
0.7	81.35	69.30	62.25

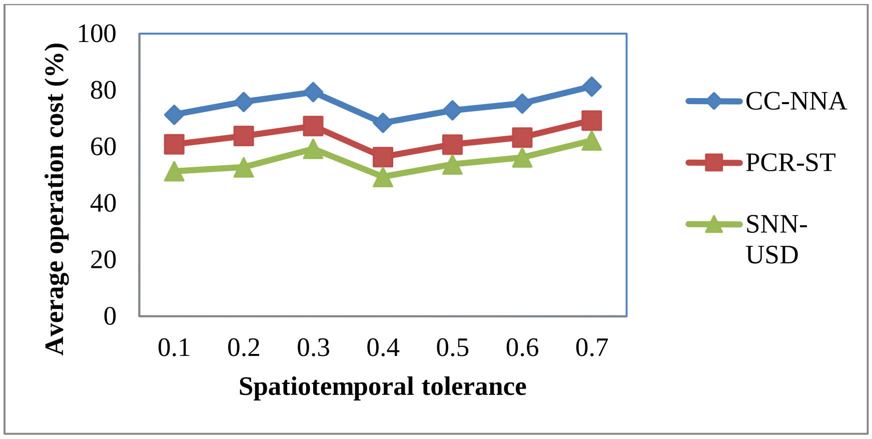

Figure 7. Measure of average operation cost

From Fig. 7 it is clear that CC-NNA performs better than PCR-ST [1] and SNN-USD [2]. In CC-NNA technique, with an increase in the spatiotemporal tolerance, the average operation cost also increases. As shown in Figure 7, the average operation cost is reduced using the proposed CC-NNA technique. With the construction of Optimal Contour Generalization algorithm, where interpolation is made based on the nearest neighbor, the average operation cost is reduced using the proposed CC-NNA technique. By constructing distance function and query type, whenever a query is being issued by the user regarding the income of the customer, the Optimal Contour Generalization algorithm provide spatiotemporal tolerance factor '∂' resulting in minimizing the average operation cost. This integration of distance function, query type and optimal selection of spatiotemporal tolerance factor results in the improvement of average operation cost by 15.81% compared to PCR-ST. Besides, different queries on uncertain data are addressed for different users. As a result, better performance is provided and therefore the average operation cost is reduced by 26.68% compared to SNN-USD.

C. Scenario 3: Analysis of query processing efficiency

The efficiency of query processing using three techniques namely, CC-NNA, PCR-ST and SNN-USD is provided in table 3. The Query processing efficiency was performed based on the queries addressed 'QA' to the total number of queries 'Q' issued by the user with different timestamps 'T' respectively. The query processing efficiency is mathematically formulated as given below.

$$QPE = \left(\frac{QA}{Q}\right) * T * 100 \tag{11}$$

The query processing efficiency is measured in terms of percentage (%). Higher the query processing efficiency, more efficient the method is said to be.

TABLE 3. TABULATION FOR QUERY PROCESSING EFFICIENCY

No. of queries (Q)	**Query processing efficiency (%)**		
	CC-NNA	**PCR-ST**	**SNN-USD**
5	74.19	64.32	59.42
10	71.35	61.48	52.37
15	75.32	63.20	57.15
20	79.21	67.08	60.03
25	84.35	72.23	67.18
30	76.13	64.01	57.01
35	86.52	74.30	67.25

Fig. 8 shows the measure of query processing efficiency with respect to different number of queries in the range of 5 to 35. As shown in the figure, the query processing efficiency is observed to be improved using CC-NNA technique than when compared to PCR-ST and SNN-USD respectively. This is because of the application of optimal polygon contour algorithm. By applying optimal polygon contour algorithm, optimal polygon contour is measured based on the 'if-then-condition' for queries on uncertain data. This in turn improves the query processing efficiency by 14.70% compared to PCR-ST and 23.22% compared to SNN-USD respectively.

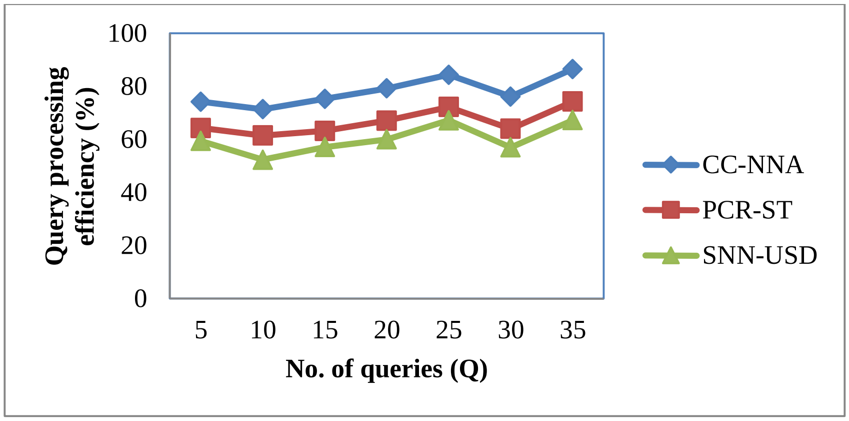

Figure 8. Measure of query processing efficiency

D. *Scenario 4: Analysis of storage space*

In order to measure the storage space based on the users query information through the CC-NNA technique, the contour segments with polygon contour is considered during the experiments on uncertain data. Storage space on uncertain data refers to the rate at which the contour segments are evaluated on uncertain data. Lower the amount of storage space, more reliable the technique is said to be.

TABLE 4. TABULATION FOR STORAGE SPACE

Methods	Storage space (MB)
CC-NNA	155
PCR-ST	185
SNN-USD	235

The storage space using CC-NNA, PCR-ST and SNN-USN technique is provided in an elaborate manner in table 4 with different users' query with varied attributes are implemented using MATLAB.

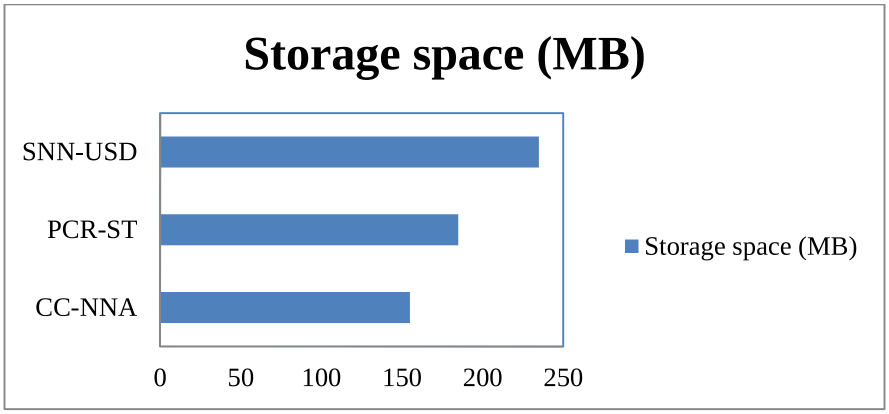

Figure 9. Measure of storage space

Fig. 9 shows the storage space ratio based on census income dataset extracted from the UCI repository with respect to 35 different queries during implementation settings at different time intervals. As depicted in the figure with the increase in the number of queries, the storage space is also increased. But when compared to the state-of-the-art works, the storage space is comparatively better than the two other methods. This is because the proposed CC-NNA uses polygon contour in addition to time-based sliding window while mining uncertain data. So the storage space is reduced by 19.35% compared to PCR-ST and 27.02% compared to SNN-USD respectively.

VI. CONCLUSION

This work presents a novel technique called Contour Generalization based NN approximation for mining uncertain data with the objective of solving the spatiotemporal problem. The performance of the proposed technique is compared with other mining methods on uncertain data (namely, PCR-ST and SNN-USD). The proposed technique has the following advantages. (i) Reduce the computation overhead by applying Contour Generalization technique with the aid of Euclidean distance and query type, (ii) provides lesser average operation cost using Optimal Contour Generalization algorithm which significantly reduces the spatiotemporal problem, (iii) representation of Optimal Polygon Contour algorithm for improving the query processing efficiency. Finally with the construction of optimal polygon contour based on the 'if-then-condition', the storage space is reduced in a significant manner. Experiments were conducted to measure the performance of CC-NNA technique and evaluated the performance in terms of different metrics, such as computation overhead, average operation cost, query efficiency and storage space for effective mining of uncertain data. The results show that CC-NNA technique offers better performance with an improvement of query processing efficiency by 14.70% and reducing the storage space by 46.38% compared to PCR-ST and SNN-USD respectively.

REFERENCES

[1] Tobias Farrell, Kurt Rothermel and Reynold Cheng, “Processing Continuous Range Queries with Spatiotemporal Tolerance”, IEEE Transactions on Mobile Computing, Volume 10, Issue 3, March 2011, Pages 320 – 334.

[2] Sze Man Yuen, Yufei Tao, Xiaokui Xiao, Jian Pei and Donghui Zhang, “Superseding Nearest Neighbor Search on Uncertain Spatial Databases”, IEEE Transactions on Knowledge and Data Engineering, Volume 22, Issue 7, July 2010, Pages 1041 – 1055.

[3] Muhammad Aamir Cheema, Xuemin Lin, Wei Wang, Wenjie Zhang and Jian Pei, “Probabilistic Reverse nearest Neighbor Queries on Uncertain Data”, IEEE Transactions on Knowledge and Data Engineering, Volume 22, Issue 4, April 2010, Pages 550 – 564.

[4] Ben Kao, Sau Dan Lee, Foris K.F. Lee, David Wai-lok Cheung and Wai-Shing Ho, “Clustering Uncertain Data Using Voronoi Diagrams and R-Tree Index”, IEEE Transactions on Knowledge and Data Engineering, Volume 22, Issue 9, September 2010, Pages 1219 – 2133.

[5] Bin Jiang, Jian Pei, Yufei Tao and Xuemin Lin, “Clustering Uncertain Data Based on Probability Distribution Similarity”, IEEE Transactions on Knowledge and Data Engineering, Volume 25, Issue 4, April 2013, Pages 751 – 763.

[6] Xike Xie, Man Lung Yiu, Reynold Cheng and Hua Lu, “Scalable Evaluation of Trajectory Queries over Imprecise Location Data”, IEEE Transactions on Knowledge and Data Engineering, Volume 26, Issue 8, August 2014, Pages 1 – 14.

[7] Rinku Dewri and Ramakrisha Thurimella, “Exploiting Service Similarity for Privacy in Location Based Search Queries”, IEEE Transactions on Parallel & Distributed Systems, Volume 25, Issue 2, February 2014, Pages 1 – 10.

[8] Graham Cormode and Minos Garofalakis, “Histograms and Wavelets on Probabilistic Data”, IEEE Transactions on Knowledge and Data Engineering, Volume 22, Issue 8, August 2010, Pages 1142 – 1157.

[9] Shaoxu Song, Lei Chen and Jeffrey Xu Yu, “Answering Frequent Probabilistic Inference Queries in Databases”, IEEE Transactions on Knowledge and Data Engineering, Volume 23, Issue 4, April 2011, Pages 512 – 526.

[10] WeiWang, Guohua Liu and Dingjia Liu, “Chebyshev Similarity Match between Uncertain Time Series”, Hindawi Publishing Corporation, Mathematical Problems in Engineering, April 2015, Pages 1 – 14.

[11] Lei Chen and Changliang Wang, “Continuous Sub graph Pattern Search over Certain and Uncertain Graph Streams”, IEEE Transactions on Knowledge and Data Engineering, Volume 22, Issue 8, August 2010, Pages 1093 – 1109.

[12] Dongxiao Niu, Ling Ji, QingguoMa and Wei Li, “Knowledge Mining Based on Environmental Simulation Applied to Wind Farm Power Forecasting”, Hindawi Publishing Corporation, Mathematical Problems in Engineering, Volume 2013, Article ID 597562, 8 pages.

[13] Fabrizio Angiulli and Fabio Fassetti, “Indexing Uncertain Data in General Metric Spaces”, IEEE Transactions on Knowledge and Data Engineering, Volume 24, Issue 9, September 2012, Pages 1 – 18.

[14] Feng Chen, Pan Deng, Jiafu Wan, Daqiang Zhang, Athanasios V. Vasilakos and Xiaohui Rong, “Data Mining for the Internet of Things: Literature Review and Challenges”, Hindawi Publishing Corporation, International Journal of Distributed Sensor Networks, Volume 2015, Article ID 431047, 14 pages.

[15] Zhi-Jie Wang, Dong-Hua Wang, Bin Yao and Minyi Guo, “Probabilistic Range Query over Uncertain Moving Objects in Constrained Two-Dimensional Space”, IEEE Transactions on Knowledge and Data Engineering, Volume 27, Issue 3, March 2015, Pages 866 – 879.

[16] Brian Quanz, Jun (Luke) Huan and Meenakshi Mishra, “Knowledge Transfer with Low-Quality Data: A Feature Extraction Issue”, IEEE Transactions on Knowledge and Data Engineering, Volume 24, Issue 10, October 2012, Pages 1789 – 1802.

[17] Costas Panagiotakis, Nikos Pelekis, Ioannis Kopanakis, Emmanuel Ramasso and Yannis Theodoridis, “Segmentation and Sampling of Moving Object Trajectories Based on Representativeness”, IEEE Transactions on Knowledge and Data Engineering, Volume 24, Issue 7, July 2012, Pages 1328 – 1343.

[18] Rajeev Gupta and Krithi Ramamritham, “Query Planning for Continuous Aggregation Queries over a Network of Data Aggregators”, IEEE Transactions on Knowledge and Data Engineering, Volume 24, Issue 6, June 2012, Pages 1065 – 1079.

[19] Ying Zhang, Wenjie Zhang, Qianlu Lin and Xuemin Lin, “Effectively Indexing the Multi-Dimensional Uncertain Objects for Range Searching”, IEEE Transactions on Knowledge and Data Engineering, Volume 26, Issue 3, March 2014, Pages 1 – 12.

[20] Massimiliano Albanese, Cristian Molinaro, Fabio Persia, Antonio Picariello and V.S. Subrahmanian, “Discovering the Top-k Unexplained Sequences in Time-Stamped Observation Data”, IEEE Transactions on Knowledge and Data Engineering, Volume 26, Issue 3, March 2014, Pages 577 – 594.

M. Kalavathi is a Head, Department of Computer Science, Govt. Arts & Science College, Komarapalayam. She received the M.Sc., Degree from Bharathiar University in 2004, M.Phil Degree from Annamalai University in 2006, respectively in Computer Science. Her research interest includes Data Mining and Digital Image Processing.

Dr. P. Suresh is a Head, Department of Computer Science, Salem Sowdeswari College [Govt. Aided], Salem. He received the M.Sc., Degree from Bharathidasan University in 1995, M.Phil Degree from Manonmaniam Sundaranar University in 2003, M.S (By Research) Degree from Anna University, Chennai in 2008, PGDHE Diploma in Higher Education and Ph.D., Degree from Vinayaka Missions University in 2010 and 2011 respectively in Computer Science. He is an Editorial Advisory Board Member of Elixir Journal. His research interest includes Data Mining and Natural Language Processing. He is a member of Computer Science Teachers Association, New York.