

ARTISTIC VISUALIZATION OF TRAJECTORY DATA USING CLOUD MODEL

Tao Wu^{a,b,*}, Yingchun Zhou^b, Limin Zhang^b

^a School of Information Engineering, Lingnan Normal University, Guangdong Zhanjiang 524048, China - wutao@whu.edu.cn

^b Guangdong Engineering and Technological Development Center for E-learning, Guangdong Zhanjiang 524048, China

Commission IV, WG IV/3

KEY WORDS: Data Visualization, Spatiotemporal Data Analysis, Trajectory Data, Visual Analytics, Artistic Style

ABSTRACT:

Rapid advance of location acquisition technologies boosts the generation of trajectory data, which track the traces of moving objects. A trajectory is typically represented by a sequence of timestamped geographical locations. Data visualization is an efficient means to represent distributions and structures of datasets and reveal hidden patterns in the data. In this paper, we explore a cloud model-based method for the generation of stylized renderings of trajectory data. The artistic visualizations of the proposed method do not have the goal to allow for data mining tasks or others but instead show the aesthetic effect of the traces of moving objects in a distorted manner. The techniques used to create the images of traces of moving objects include the uncertain line using extended cloud model, stroke-based rendering of geolocation in varying styles, and stylistic shading with aesthetic effects for print or electronic displays, as well as various parameters to be further personalized. The influence of different parameters on the aesthetic qualities of various painted images is investigated, including step size, types of strokes, colour modes, and quantitative comparisons using four aesthetic measures are also involved into the experiment. The experimental results suggest that the proposed method is with advantages of uncertainty, simplicity and effectiveness, and it would inspire professional graphic designers and amateur users who may be interested in playful and creative exploration of artistic visualization of trajectory data.

1. INTRODUCTION

With the rapid development of positioning services and location acquisition techniques, massive spatiotemporal trajectory data can be easier obtained, and mining or analysing trajectory data is very important in many different applications, such as transportation management (Wasesa et al, 2017), mobility study (Ting, 2016), path discovery, movement behaviour analysis (Tang et al, 2016), and mobile communication management. A trace of object is the continuous moving in geographical space, while the corresponding trajectory is a sample of location points that the moving object passes. The visual analytics solutions show great potential as they allow users to explore the data and improve mining process and results (Schreck, 2009), and there are various methods to manage, process and mine trajectory data (Leonardi et al, 2017).

However, we believe that, there are several uncertainties in the sampling process of trajectory data, on the other hand, map creation, as well as geolocation in the trajectory data, has historically involved much artistry. In fact, the GPS doodle pioneered by Stephen Lund (Stephen, 2017) has attracted considerable attention.

Inspired by artworks and other sources, we try to handle the uncertainties of trajectory data and then focus on its artistic visualization, that is, the aesthetic effect of the traces of moving objects. As the previous references (Tobias, 2013 and Sungye et al, 2013) done, we are thus not interested in precise and correct depictions of the world but instead in altering maps to satisfy a specific personal aesthetic.

There are many other works related to this topic in non-photorealistic rendering and computational aesthetics, including image-guided Voronoi aesthetic patterns (Wu and Zhang, 2016), stroke-based technique (Hertzmann, 2003), image-

guided fracture (David, 2005) and frayed cell (Michael et al, 2014). Nevertheless, none of these methods have been involved into trajectory data, and how to generate an aesthetic effect of trajectory data is still a challenging problem.

In this context, we proposed an artistic visualization algorithm for trajectory data based on cloud model (VTC for short). Our intentions are twofold: (1) how can the uncertain trajectory be produced using a cloud model-based algorithm? (2) how different types of strokes can affect the aesthetic qualities of the rendered results?

Cloud model is a cognitive model (Li et al, 2009), which can realize the bidirectional cognitive transformation between qualitative concept and quantitative data based on probability statistics and fuzzy set theory. There are various successful applications using cloud model (Wang, 2014, Wu et al, 2015a, Wu et al, 2015b). This novel model has been attracted several attentions from researchers, which is different from the traditional statistical methods and fuzzy set methods, and cannot be considered as randomness compensated by fuzziness, fuzziness compensated by randomness, second-order fuzziness or second-order randomness.

The cloud model-based rendering consists of three main steps, including uncertain line representation using the extended cloud model, uncertain trajectory approximation with varying shadow, and then generating an aesthetic image with the artistic style.

Our method satisfies the property of uncertainty, simplicity, and effectiveness: (1) The proposed method is involved as a novel technique with uncertainty using cloud model. (2) The proposed method is very simple, and only the uncertain representation of lines and the drawing of shadows are introduced into the traditional method. (3) From the artist perspective, the focus is on creativity and iterative process, rather than the result image. The proposed method is a dynamic generation one by one, and

* Corresponding author

correspondingly various styles of result images can be obtained by using various types of strokes.

The rest of the paper is organized as following: Section 2 firstly provides an introductory explanation of cloud model, then proposes the extended cloud model for uncertain line and uncertain trajectory. Section 3 describes the VTC algorithm. Section 4 shows the experimental results and discussion. Finally, the conclusion is drawn in Section 5.

2. UNCERTAIN TRAJECTORY USING CLOUD MODEL

2.1 Cloud Model

Let U be a quantitative universe described by precise numbers, and C is a qualitative concept related to U , Given a number $u \in U$, and u is a random realization of the qualitative concept C , $\mu(u)$ is the membership degree of u to C , $\mu(u) \in [0,1]$, it is a random number with a stable tendency :

$$\mu : U \rightarrow [0,1], \quad u \in U, \quad \mu(u) \quad (1)$$

$$\mu(u) = e^{-\frac{(u - Ex)^2}{2En^2}} \quad (2)$$

The distribution of u in domain is as a normal cloud model, each u is called a cloud drop.

There are three numerical characters of cloud model, that is, expectation Ex , entropy En and hyper-entropy He , and a qualitative concept C is denoted by $C(Ex, En, He)$ uniquely. Ex is the expectation of the cloud drops distribution in domain, and is the position representing the qualitative concept best, corresponding to the center of gravity of the cloud. En is the uncertainty measurement of the qualitative concept, decided by the randomness and the fuzziness of the concept. En reflects the discrete degree of cloud drops which can represent the concept, so-called randomness. On the other hand, it reflects the interval of cloud drop accepted by the concept, so-called fuzziness. He is the uncertain measurement of entropy, i.e. the entropy of En , decided by the randomness and the fuzziness of En .

The kernel of normal cloud model is the transform between qualitative concept and quantitative data (Li et al, 2009), and it is realized by normal cloud generators. On one hand, forward normal cloud generator is the mapping from qualitative concept to quantitative values, and it produces the cloud drops to describe a concept when three numerical characters and the number of cloud drops are input. On the other hand, backward normal cloud generator provides the transformation from quantitative numerical values to quality concept, and a normal cloud model with three numerical characters is defined by computing mean, absolute central moment with the first order, and variance of sample data. Essentially, the normal cloud generators are two algorithms based on probability measure space. These processes are uncertain and cannot be expressed by a precise function. More information about normal cloud model can be obtained from (Li and Du, 2007).

As can be seen from Figure 1, a cloud model is possible to measure the deviation of a random phenomenon from a normal distribution when the random phenomenon does not satisfy strictly a normal distribution.

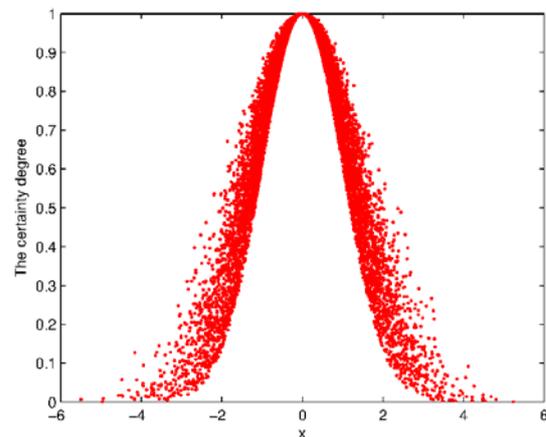


Figure 1. Normal cloud model

2.2 Uncertain Line

Inspired by the cloud model, we believe that a line segment can also be represented by an extended definition, if considering it as a qualitative concept with the numerical characters. Any line is a part of a continuous domain in 2D plane, and an uncertain transformation from the quantitative value and the qualitative concept can be easily achieved using an extended cloud model. Based on the traditional normal cloud model, we define the uncertain line and the dataset of sampling points.

Given two endpoints of a line segment, that is, $P_1 = (px_1, py_1)$ and $P_2 = (px_2, py_2)$, where px_1, py_1, px_2 and py_2 are the coordinates, we can take the slope k and the vertical intercept b as a key concept to be expressed by cloud model, determined by (Ex_k, En_k, He_k) and (Ex_b, En_b, He_b) . Consequently, an uncertain line segment would be simply defined by the parameter set $\{k, b\}$ as below:

$$C \left(\begin{matrix} Ex_k \\ Ex_b \end{matrix}, \begin{matrix} En_k \\ En_b \end{matrix}, \begin{matrix} He_k \\ He_b \end{matrix} \right) \quad (3)$$

where Ex_k, Ex_b are the real values of slope and vertical intercept, calculated by the following equations,

$$\begin{aligned} Ex_k &= \frac{py_2 - py_1}{px_2 - px_1} \\ Ex_b &= py_1 - px_1 \frac{py_2 - py_1}{px_2 - px_1} \end{aligned} \quad (4)$$

Therefore, we can extend the definition of cloud model. Let D , consisting of quantitative values, be a continuous domain in the plane, and let a line L be the qualitative concept related to D . For a line l on L , its certainty degree is as below:

$$m = \mu_k \mu_b \quad (5)$$

where μ_k, μ_b are determined by Eq.(2).

A given line segment can be approximated by discrete cloud drops using a generation algorithm, and then the quantitative value and the qualitative concept can be transformed to each other uncertainly. Naturally, these cloud drops with certainty degrees can be called as an uncertain line, and we can consider

these cloud drops as a representation of the uncertain line. The detailed algorithm is as follows.

Step 1, Given \mathbf{P}_1 and \mathbf{P}_2 , calculate the expected values $[Ex_k, Ex_b]$ according to Eq.(4), and set the entropy $[En_k, En_b]$ and hyper-entropy $[He_k, He_b]$.

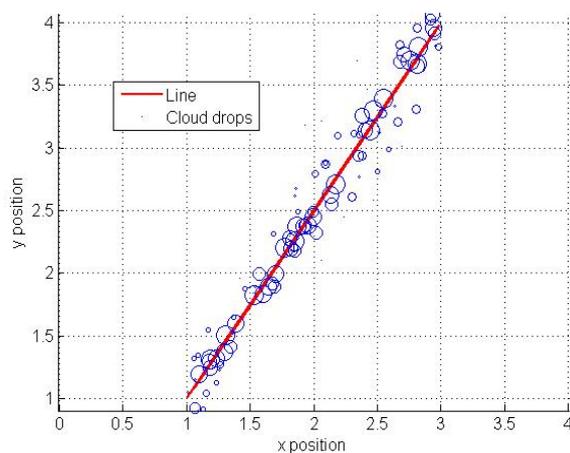
Step 2, Generate a cloud drop (k_i, μ_{k_i}) with numerical characters (Ex_k, En_k, He_k) .

Step 3, Generate a cloud drop (b_i, μ_{b_i}) with numerical characters (Ex_b, En_b, He_b) .

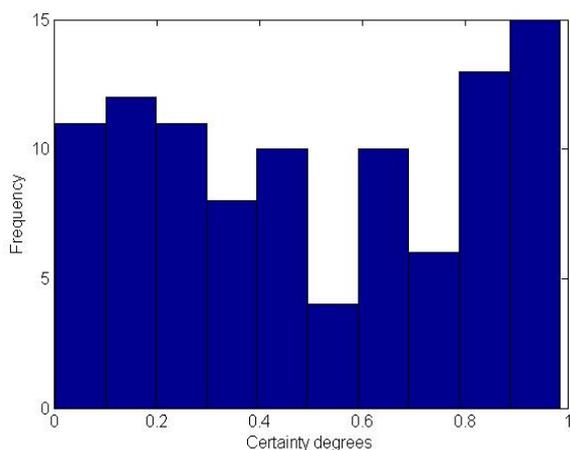
Step 4, For the point on the i th random realization l_i , obtain the abscissa value $lx_i = rand(1)(px_2 - px_1) + px_1$, where $rand(1)$ denotes a random number in $[0, 1]$, and then calculate the ordinate value $ly_i = k_i lx_i + b_i$. The certainty degree μ_i is determined by Eq.(5).

Step 5, Repeat steps 2 to 4 until the number of cloud drops N reaches the predefined value.

For the two endpoints $\mathbf{P}_1 = (1,1)$ and $\mathbf{P}_2 = (3,4)$, the expected values can be calculated as $Ex_k = 1.5$ and $Ex_b = -0.5$, and we fix the other numerical characters $En_k = 0.1Ex_k = 10He_k$ and $En_b = 0.1Ex_b = 10He_b$. Then we generate $N=100$ cloud drops of the uncertain line, as shown in Figure 2.



(a) The uncertain line



(b) The distribution of certainty degrees

Figure 2. The uncertain line and its certainty degree

The location relations between the sample points and the line are as shown in Figure 2(a). The circles with various sizes denote cloud drops of sample points, whose size is related with its certainty degree. As can be seen from Figure 2(a), these sample points are around the line segment with a nonuniform distribution. Intuitively, most of offset values between the

sample points and the line segment are acceptable, and then a concept “near or around the line segment” is achieved. In this point of view, an uncertain representation of a line segment is feasible and reasonable using the extended cloud model. Furthermore, the distribution of certainty degrees is also listed in Figure 2(b), and from another way it is proved a nonuniform distribution rather than a Gaussian distribution.

2.3 Uncertain Trajectory

A trace of a moving object in geographical space is continuous while a trajectory is only a sample of location points that the moving object passes. A spatial trajectory, as a simplest case of trajectory data, is represented by a sequence of timestamped locations. Although trajectory data is collected from various sources, real-captured data is typically raw, erroneous, and contain uncertainties, outliers, missing values, or mismatched items. Thus, there are several uncertainties in the sampling process of trajectory data. In other words, the traditional visualization of trajectory data in a precise manner may be questionable, and no aesthetic enjoyment.

In this subsection, we define uncertain trajectory based on uncertain line, and then process the trajectory data in an uncertain manner with the aesthetic effect.

For each pair of sample points of trajectory data, $\mathbf{P}_1 = (px_1, py_1)$ and $\mathbf{P}_2 = (px_2, py_2)$, where px_1, py_1, px_2 and py_2 are the coordinates of them, we can generate the cloud drops of the trace line, whose number denoted by N .

Then we employ densely drawn coloured lines, placed a shadow line (blue) to the main (red) lines on two sides, as shown in Figure 3(a), two cloud drops ($N=2$) and the corresponding shadow lines are generated. These lines have a length that varies randomly and they fade out toward the end, its length is denoted by r . This would create a visual effect that is reminiscent of watercolour painting and we strive to render trajectory data in a similar way.

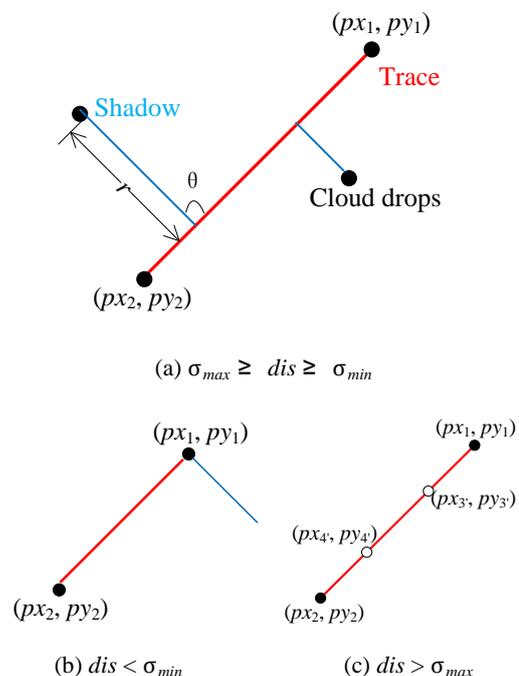


Figure 3. Schematic depiction of how the coloured lines (blue) are created with respect to the base line (red) from the trajectory

Unlike the style of the existing methods (Wu et al, 2016 and Tobias, 2013), an additional parameter θ is introduced into controlling the orientation of the shadow line, that is, the angle between the shadow line and the main trace, as shown in Figure 3(a). This would create a more interesting effect than the existing methods.

According to the above method, a trace line from each cloud drop to the main trace is drawn one by one, and all the lines in each pair of sample points from trajectory data are drawn with the same colour, while different lines are rendered with various colours. Then the aesthetic patterns with different styles can be produced.

If the two trajectory points are closed enough to each other, that is, the length of the trace line segmentation from P_1 to P_2 is smaller than a predefined thresholding σ_{min} , we generate none of cloud drops to avoid too crowded rendering, which appears very boring. Instead, only a shadow line crossing one endpoint is placed perpendicular to the main trace on one of its sides, as shown in Figure 3(b).

Conversely, once the trace line between two trajectory points is enough long, greater than a predefined thresholding σ_{max} , we calculate the detailed sub-sample points with a certain distance d , and each pair of two sample points are processed by the above method. As shown Figure 3(c), we divide the line into three parts, and $P_3 = (px_3, py_3)$ and $P_4 = (px_4, py_4)$ are the sub-sample points. In other words, the original two trajectory points are evolved to four new points to be processed in this step.

In this way, we generate novel trajectory data with another style, and we can call this result as uncertain trajectory since the original trajectory data are deleted and updated, as well as new data added.

In addition, we allow the reduction sampling with a certain step size s because of the aesthetic effect. Also for this purpose, we introduce a user-specified colour palette to create various visual effects.

The detailed algorithm of uncertain trajectory based on uncertain line using cloud model is as follows.

Step 1, For the pair of trajectory data (px_1, py_1) and (px_2, py_2) , calculate the distance dis between them.

Step 2, According to the above distance dis , make three kinds of processes respectively.

Case 1: If $\sigma_{max} \geq dis \geq \sigma_{min}$, generate N cloud drops of uncertain line, and draw the shadow lines crossing (px_1, py_1) at an angle θ with the main trace, which can be fixed a random value or a constant with $\theta=90^\circ$. In addition, the length of the shadow lines is also consisted of a constant part dis and a random part less than $r-dis$.

Case 2: If $dis < \sigma_{min}$, draw a shadow line crossing the endpoint and being perpendicular to the main trace on one of its sides, whose length is a random number less than dis .

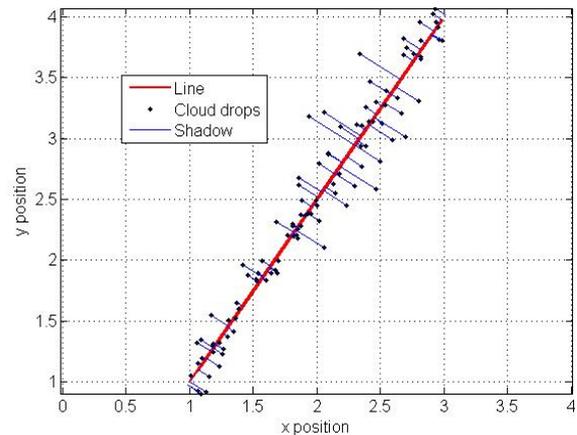
Case 3: If $dis > \sigma_{max}$, divide the line from (px_1, py_1) and (px_2, py_2) into several parts with the equal length d . Then for each part, repeat steps 1 to 2 until each part of sub-sample points being processed.

Step 3, Repeat steps 1 to 2 until each pair of sample points of trajectory data being processed.

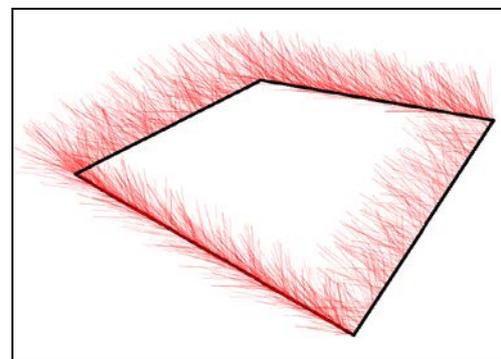
We consider the line from (1,1) to (3,4) in Figure 2(a) as a synthetic trajectory, and generate the uncertain trajectory with a constant value ($\theta=90^\circ$), the result image is shown in Figure 4(a), there are some relatively rough boundaries instead of the original accurate ones.

Additionally, another synthetic trajectory consisted of four lines is rendered with a random θ , the result is shown in Figure 4(b), which appears an aesthetic effect, and is more interesting than

the line style that the traditional visualization of map and trajectory data done. In the following section, we will demonstrate more visual results and evaluate them using aesthetic measures.



(a) The rendering result of a line trajectory



(b) The uncertain trajectory of a convex quadrangle

Figure 4. The uncertain trajectory based on uncertain line

2.4 Rendering Strokes and Implementation

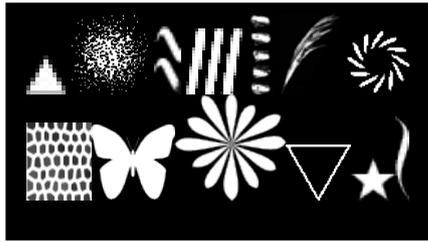
In this subsection, we further improve the visual effect more interesting, since the line drawing of shadow is still lack of artist conception. Inspired by the stroke-based NPR methods (Hertzmann, 2003), we introduced rendering strokes into the proposed technique rather than pure lines. In other words, we replace the line drawing with stroke rendering in step 2 of the algorithm mentioned in the previous section.

For the purposes of test and verification, a rendering system is developed using the Processing, which is a flexible software sketchbook and a language for learning how to code within the context of the visual arts. The open source, free software can be publicly downloaded from <https://www.processing.org/>.

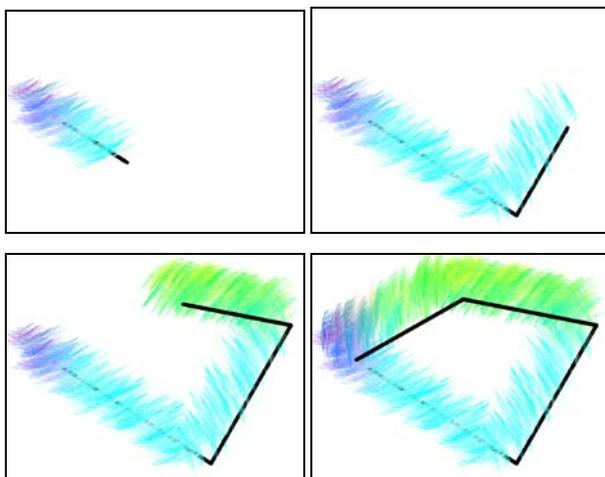
We firstly collect the material images on the rendering strokes, which should be a PNG file format because of the necessary of the alpha channel. Considering the colouring mechanism of Processing, the original images are pre-processed by GIMP using the whitening method. Some example strokes are shown in Figure 5(a). Certainly, this is only depended on our experience. It is still an open problem, and some other methods and stroke images are also compatible.

For an uncertain trajectory, a shadow line from the cloud drop to the main trajectory line is added using the function of `line()` provided by Processing. While for an artistic trajectory, a stroke image is placed on the location of each cloud drop, which can be easily implemented by the function of `image()` in Processing.

Additionally, combined with *tint()* and *blendMode()*, the stroke images can be tinted to specified colours or made transparent by including an alpha value. An example result is shown in Figure 5(b), and one can observe that this result is pleasing with a colourful style. For comparative purpose, the real line is also drawn in the result, which would be removed in the final rendering result.



(a) The used stroke images



(b) The result images of generations 200, 800, 1000 and 2000, from left to right, up to bottom

Figure 5. The rendering results using a stroke image

Furthermore, the journey of rendering iteration by iteration may be just as or more interesting than the result images. Thus, we generate the rendering images with Processing, and then record the generation process as a video, as the previous work done (Döllner, 2012 and Buschmann et al, 2016). In addition, we can save each frame images and convert them into an animation using the Movie Maker attached on Processing. For the reference, we show four frame images in Figure 5(b), generated by 200, 800, 1000 and 2000 generations. To some extent, this demonstrates the possibility of producing video or animation by our method.

3. THE PROPOSED METHOD

In this section, we described the proposed method in detail, and provided a brief analysis on the computational complexity of the proposed approach.

To sum up, the proposed VTC algorithm includes five steps, and it is described as the following.

Step 1 (Initialization). The related parameters are initialized including the numerical characteristics of cloud model $[En_k, En_b]$ and $[He_k, He_b]$, the number of cloud drops N , the predefined threshold σ_{min} , σ_{max} , the inclination angle of shadow

lines θ , step size of reduction sampling s and distance increment for subsampling d .

Step 2 For a given point set of trajectory data, denoted by $\{(px_1, py_1), (px_2, py_2), \dots, (px_n, py_n)\}$, two consecutive points with the step size s are selected as endpoints of the current line, and fix $i=1$.

Step 3 (Uncertain line). For the i th pair of trajectory data (px_i, py_i) and (px_{i+s}, py_{i+s}) , calculate the distance dis between them. If necessary, the cloud drops for uncertain line are generated by the method in Section 2.2.

Step 4 (Uncertain trajectory). Trajectory is approximated and added the shadows using the method in Section 2.3.

Step 5 (Loop). The counter i is incremented by $i=i+1$. Repeat steps 3 and 4 until all the trajectory lines are processed; that is, $i=n$.

The above algorithm is easy to implement and the time complexity is acceptable. The key time cost of the VTC algorithm lies in steps 3 and 4. For the trajectory data consisted of n points, each pair of them are processed using sub-sample in the worst case. The number of new points is $[dis/d]-1$, thus the total number of processed lines is $[dis/d] + 1$. Each uncertain line includes N cloud drops in the worst case, and the drawing of shadows of each line can be achieved in a constant time $O(1)$. Therefore, the time cost is approximately $O(Nn[dis/d])$ in the worst case. Generally, $[dis/d] \ll Nn$ and $N < n$. Altogether, the VTC algorithm costs time $O(Nn)$, or $O(n)$ more roughly. In other words, the time complexity of the VTC algorithm is approximately linear with the number of trajectory data, which indicates that our method is of high efficiency.

4. EXPERIMENTAL RESULTS AND DISCUSSION

4.1 Experiment Setup

In this section, we present a detailed analysis on experimental results and discussion, as well as the choice of the parameters of the proposed method. To demonstrate the efficiency and feasibility of the proposed method, we conduct six groups of experiments, and all the result images are involved into quantitative comparisons using four aesthetic measures.

The proposed algorithm has been implemented under the Processing environment, and performed on a 2.5GHz Core PC i7-6500U CPU with 8GB RAM.

The proposed algorithm includes five parameters, which can be divided into three groups. The first is the angle θ between the shadow line and the main trace, and we only provide a choice between a random and constant value. The second includes the numerical characteristics of cloud model $[En_k, En_b]$ and $[He_k, He_b]$ the predefined threshold σ_{min} , σ_{max} , we use the default settings of $En_k = 0.1Ex_k = 10He_k$, $En_b = 0.1Ex_b = 10He_b$, $\sigma_{min}=0.1$, $\sigma_{max}=1$, but this should be varied with different maps. The last contains the number of cloud drops N , step size of reduction sampling s and distance increment for subsampling d , which can be adjusted by users' interaction.

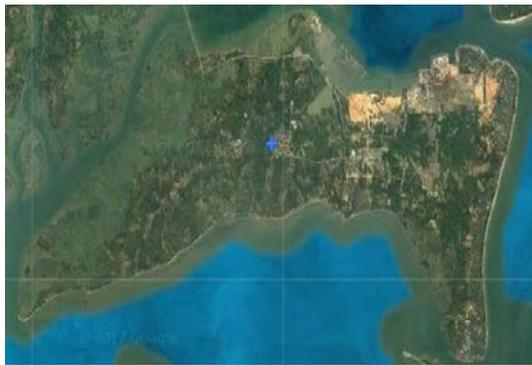
The used real trajectory dataset is collected at Donghai island of Zhanjiang, China using GPS. Another two real datasets are download from Geolife project (<https://www.microsoft.com/en-us/download/details.aspx?id=52367>) and mobile computing project of transportation research and analysis computing center (https://www.cs.uic.edu/~boxu/mp2p/gps_data.html).

4.2 Results and Analyses

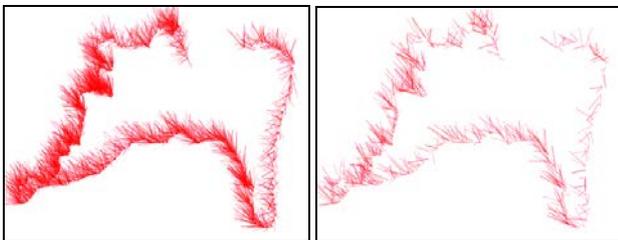
In this subsection, we demonstrate and investigate the visual and quantitative influence of different parameters or settings on

the aesthetic qualities of visualization, including real and synthetic trajectory data, step sizes, length of shadow, types of strokes, colour modes, filling opacity, background styles, as well as quantitative comparisons.

4.2.1 The Number of Cloud Drops: In this group, we investigate the number of cloud drops, which determines the intensity of the shadow lines. We use the real trajectory dataset of a car trip at Zhanjiang, whose map is shown as Figure 6(a). The rendering results are shown in Figure 6(b), with a small number of cloud drops, the final image is with deep shadow. Conversely, there appears little shadows. Therefore, a moderate value is more appropriate to the number of cloud drops from the perspective of aesthetic effect.



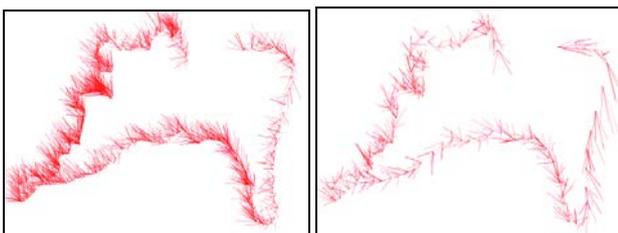
(a) The Google map of used dataset



(b) The small number (c) The large number

Figure 6. Rendering results with various number of cloud drops

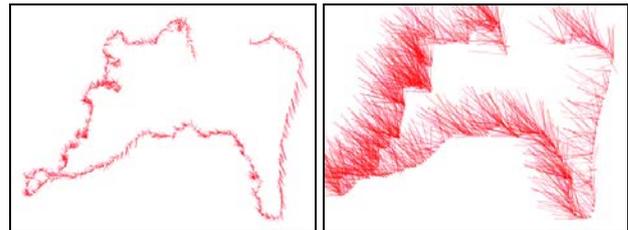
4.2.2 Step Size: In this group, we investigate the step size, which determines the number of the removed trajectory data. The rendering results are shown in Figure 7. Compared with Figure 6(b), there appears the similar effect, since both the number of cloud drops and step size control the intensity of shadow lines, the difference is that, the former is related with the number of sample points, while the latter with the trajectory data. But there still exists slight differences between them from a detailed view. Clearly, it is nice to fix a medium step size as the number of cloud drops.



(a) The small size (b) The large size

Figure 7. Rendering results with various step sizes

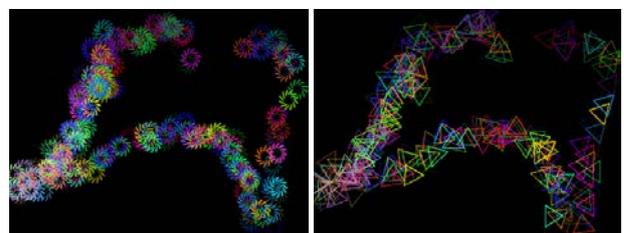
4.2.3 Length of Shadow: In this group, we investigate the length of shadow, which determines the style of shadow lines. The rendering results are shown in Figure 8. With a small value for length of shadow, there appears the narrow shadow, otherwise, the wide shadow. As shown in Figure 8, too narrow shadow would lead to less efficient, even invalid results by the proposed method, while too wide shadow results in boring images. Thus, a moderate length of shadow is beneficial for the proposed method. Nevertheless, we provide this parameter for users' interaction.



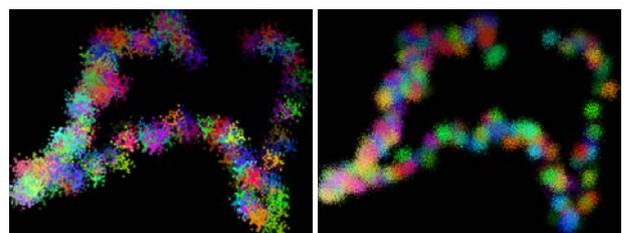
(a) The narrow shadow (b) The wide shadow

Figure 8. Rendering results with various length of shadow

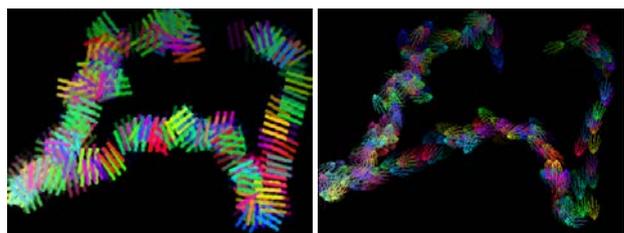
4.2.4 Types of Strokes: The different shapes of strokes can enrich the visual outcome of visualization. We investigate six types of strokes. The rendering results are shown in Figure 9. Clearly, various strokes would appear various styles of result images, and different types of strokes can affect the aesthetic qualities. For the quantitative comparisons in the next section, we name these results as No 1 to No 6.



(a) No 1 (b) No 2



(c) No 3 (d) No 4



(e) No 5 (f) No 6

Figure 9. Rendering results with various strokes

As shown in Figure 9, the visual effect is similar to the blinking of neon light, and these results can be divided into three groups. The results in Figures 9(a) and (b) demonstrate an empty stroke-based rendering, which has a visual effect with geometrical style. Those images of No 3 and No 4 (see Figures 9(c) and (d)) appear particle texture with varied size, and simulate the sand painting art. As shown in Figures 9(e) and (f), there have the wonderful rhythmical flow of lines and curves. The results indicate the effectiveness of the proposed method. Even so, the stroke-based rendering for artistic visualization of trajectory data is still an open problem under the proposed framework using cloud model, and various stroke images can be introduced into the system and generate beautiful and interesting results.

4.2.5 Colour Modes with Various Dataset: The artistic visualization can be enriched by the use of colours because the colour schema plays an important role in aesthetic perception. In this group, we try to provide a study on the colour schemas for the proposed method. All the above results used the colour schema of the HSV map and the pure background, which is default for the proposed algorithm. Besides that, the RGB colormap and the image background are also tested. The rendering results are shown in Figure 10. For the quantitative comparisons in the next section, we name these results as No 7 to No 10.

The results in Figure 10(c) is generated by a synthetic dataset of trajectory, while the other three are used real trajectory data from a driving at Zhanjiang, Geolife project and mobile computing project respectively. The black ground is a default setting, white or sand background are also used. In addition, we present a rendering map from filling opacity to certainty degrees and locations, as shown in Figure 10(a) to (c).

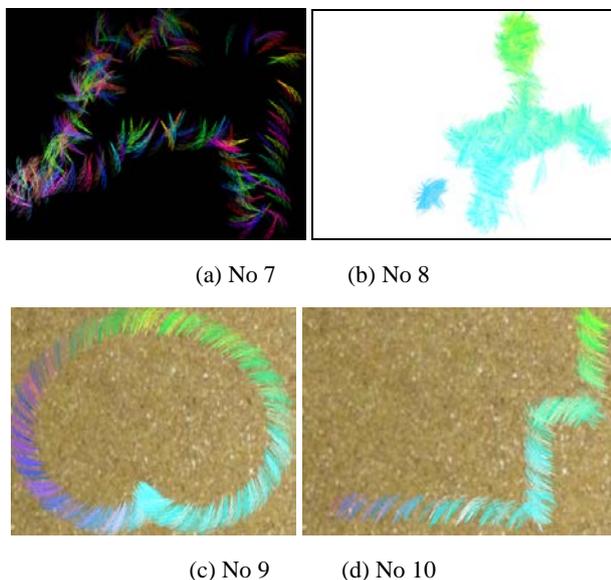


Figure 10. Rendering results with various colour modes

4.2.6 Quantitative Comparisons: The aesthetic performance is a key problem to practical engineering applications, and it is desired that the visualization of trajectory data can be generated with a good visual effect. Although the aesthetics problem has a strong subjectivity, the computational aesthetics is one of the most challenging in image processing and computer graphics, we still introduce four measures as a quantitative evaluation. The aesthetic measures used in our experiments include Benford's Law (Jolion, 2001) (BL for short), Fractal Dimension

(Wannarumon, 2008) (FD for short), Global Contrast Factor (Matkovic, 2005) (GCF for short) and Shannon's Entropy (Rigau, 2008) (SE for short). Full details of these measures refer to the original papers.

Ten images in Figure 9 and Figure 10 are involved into the quantitative comparison. The aesthetic evaluation scores of various result images are listed in Table 1. BL is a measure on the distribution of intensity of pixels. The result images generate not so high BL values, thus they approximately obey Benford's Law. Most of images obtain lower FD values, which are considered simple. Except for Figure 10(b) to (d), almost all of images produce higher GCF scores, and then they are more colourful and interesting from the point of view of the GCF aesthetics. SE provides a probabilistic method for comparing pixels in an image. Images shown in Figure 10(c) and (d) gain the higher SE score than others, which seem to satisfy a uniform distribution of brightness values.

Images	Measures			
	BL	FD	GCF	SE
Figure 9(a)	0.581	0.572	10.956	2.801
Figure 9(b)	0.465	0.572	10.444	2.147
Figure 9(c)	0.721	0.574	12.612	3.655
Figure 9(d)	0.652	0.572	11.849	3.182
Figure 9(e)	0.792	0.576	13.495	3.999
Figure 9(f)	0.514	0.571	9.285	2.463
Figure 10(a)	0.556	0.571	9.791	3.231
Figure 10(b)	0.472	0	3.205	1.959
Figure 10(c)	0.825	0.942	2.781	7.175
Figure 10(d)	0.801	0.949	2.162	7.064

Table 1. Aesthetic evaluation of various result images

4.3 Discussion

In this subsection, we provide a brief discuss on the proposal. Relating to research question (1) in Section I, we define the extended cloud model of uncertain line, and introduce it into the visualization of trajectory data, then provide a cloud model-based algorithm for artistic visualization of trajectory data. Relating to research question (2), we demonstrate different artistic styles using different types of strokes, as well as colour modes. The result images achieve the interesting and pleasing effect, and various types of strokes can be applied into various fields.

Overall, the cloud model-based method performs a good performance, and it can be applied to the visualization of trajectory data. In addition, the proposed method provides to users a potential and interesting solution, in which the trajectory data is as an interactive method of artistic visualization, as the pioneer done (Stephen, 2017).

5. CONCLUSION

In the paper, we extend the definition of cloud model, then the VTC algorithm is proposed. Furthermore, we analyse the visual effect and performance influence of the proposed method, and investigate the aesthetic qualities with various parameters, including types of strokes, colour modes, as well as quantitative comparisons on the result images. The experiment results verify the efficiency and feasibility of the proposed method.

There are a couple of issues that will be considered in the future research: (1) How to fix the parameters to fit for the feature of trajectory data is currently under investigation, and will be reported later. (2) The art work also includes various styles,

besides that has been involved into the proposed method. Introducing other novel forms of cloud models into the proposed method handling the uncertainty of trajectory data is well worth further studying on.

ACKNOWLEDGEMENTS

This work was partially supported by the National Natural Science Foundation of China under Grant (No. 61402399), and by Foundation for Distinguished Young Teachers in Higher Education of Guangdong, China (No. YQ2014117).

REFERENCES

- Buschmann S., Trapp M., and Döllner J., 2016. Animated visualization of spatial-temporal trajectory data for air-traffic analysis. *The Visual Computer*, 32(3), pp. 371-381.
- Döllner J., 2012. Animated visualization of spatial-temporal trajectory data for air-traffic analysis. In: International Conference on Geoinformatics, Hong Kong, China, pp. 1-7.
- David M., 2005. Image-guided fracture, In: *Proceedings of the Graphics Interface Conference*, Canadian Human Computer Communications Society, Victoria, Canada, pp. 219-226.
- Hertzmann, A., 2003. A Survey of Stroke-Based Rendering. *IEEE Computer Graphics and Applications*, 23(4), pp. 70-81.
- Jolion J., 2001. Images and Benford's law. *Journal of Mathematical Imaging and Vision*, 14(1), pp.73-81.
- Leonardi L., Orlando S., Raffaetà A., et al., 2014. A general framework for trajectory data warehousing and visual OLAP. *GeoInformatica*, 18(2), pp. 273-312.
- Li D. and Du Y., 2007. Artificial intelligence with uncertainty, Boca Raton:Chapman and Hall/CRC.
- Li D., Liu C., Gan W., 2009. A new cognitive model: Cloud model. *International Journal of Intelligent Systems*, 24 (3), pp. 363-375.
- Matkovic K., Neumann L., Neumann A., et al., 2005. Global contrast factor - a new approach to image contrast. In: *Proceeding of the first Eurographics Conference on Computational Aesthetics in Graphics, Visualization and Imaging*, pp.159-168.
- Michael B., Corinna V., and Daniel W., 2014. Frayed cell diagrams, In: *Proceedings of the ACM Workshop on Computational Aesthetics (CAe '14)*, British Columbia, Canada, pp. 93-96.
- Rigau J., Feixas M., and Sbert M., 2008. Informational aesthetics measures, *IEEE Computer Graphics and Applications*, 28(2), pp.24-34.
- Schreck T., Bernard J., Von Landesberger T., Kohlhammer J., 2009. Visual cluster analysis of trajectory data with interactive Kohonen maps. *Information Visualization*, 8(1), pp. 14-29.
- Stephen L., 2017. Strava art. Sketchbook of a GPS Artist, Victoria, BC, Canada. <https://gpsdoodles.com/>.
- Sungye K., 2013. Bristle Maps: A Multivariate Abstraction Technique for Geovisualization. *IEEE Transactions on Visualization and Computer Graphics*, 19 (9), pp.1438-1454.
- Tang J., Jiang H., Li Z., et al., 2016. A two-layer model for taxi customer searching behaviours using GPS trajectory data. *IEEE Transactions on Intelligent Transportation Systems*, 17, pp. 1-7.
- Ting F., 2016. Mining spatiotemporal patterns of urban dwellers from taxi trajectory data. *Frontiers of Earth Science*, 10(2), pp. 205-221.
- Tobias I., Ross M., Abish M., Yun J., David S. E., et al., 2013. Visual Abstraction and Stylisation of Maps. *The Cartographic Journal*, 50 (1), pp.8-18.
- Wang G., Xu C., Li D., 2014. Generic normal cloud model. *Information Sciences*, 280(0):1-15.
- Wannarumon S., Bohez E. and Kittinan A., 2008. Aesthetic evolutionary algorithm for fractal-based user-centered jewellery design. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 22(0), pp.19-39.
- Wasesa M., Stam A. and Heck E. V., 2017. The seaport service rate prediction system: using drayage truck trajectory data to predict seaport service rates. *Decision Support Systems*, 95, pp. 37-48.
- Wu T., Xiao J., Qin K., and Chen Y., 2015a, Cloud model-based method for range-constrained thresholding, *Computers & Electrical Engineering*, 42(0), pp. 33-48.
- Wu T. and Zhang L., 2016, Image-Guided Voronoi Aesthetic Patterns with an Uncertainty Algorithm Based on Cloud Model, *Mathematical Problems in Engineering*, 2016(0), pp. 9837123-18.
- Wu X., Guo G., and Bai Z., 2015b. Cloud Model-Based Energy Management Strategy for Parallel Hybrid Vehicles, *Journal of Control Science and Engineering*, 2015(0), pp.141654-7.