Article

Low-Carbon Tour Route Algorithm of Urban Scenic Water Spots Based on an Improved DIANA Clustering Model

Xiao Zhou 1,2, De Zhang 2,3,*, Jiangpeng Tian 4 and Mingzhan Su 4

1 Post-Doctoral Innovation Practice Base of Sichuan Province, Leshan Vocational and Technical College, Leshan 614000, China; zhouxiao@infu.ac.cn
2 State Key Laboratory of Geo-Information Engineering, Xi’an 710054, China
3 Xi’an Research Institute of Surveying and Mapping, Xi’an 710054, China
4 Institute of Geospatial Information, PLA Strategic Support Force Information Engineering University, Zhengzhou 450001, China; tjp_study@infu.ac.cn (J.T.); smzh625@infu.ac.cn (M.S.)

* Correspondence: zhangde01@infu.ac.cn; Tel.: +86-139-9288-9328

Abstract: Aiming at the problems in current research into low-carbon and water scenery tourism, this paper brings forward a low-carbon tour route algorithm of urban scenic water spots based on an improved Divisive Analysis clustering model. Based on the ecological attributes of scenic water spots, the clustering model is set up to create scenic spot clusters. Via the clusters, the low-carbon tour route algorithm of urban scenic water spots based on the optimal energy conservation and emission reduction mode is proposed, and it provides the optimal scenic water spots and low-carbon tour routes for tourists. The model can thus realize the optimization of vehicle exhaust emission in urban travel and reduce exhaust emission damage to urban water bodies and natural environments. In order to verify the advantages of the proposed algorithm, this paper performs an experiment to compare the proposed algorithm with the frequently used route planning methods by tourists. The experimental results show that the proposed algorithm has great advantages in energy conservation, emission reduction and low-carbon travel and can reduce the exhaust emission and the damage to the urban water bodies and the natural environment, realizing low-carbon tourism. The main findings and contributions of the proposed work are as follows. First, an improved clustering algorithm is set up, and the urban scenic water spots are clustered according to attribute data, which could optimize the scenic spot recommendation spatial model. Second, combining with the specific characteristics of scenic water spots, the scenic spot mining and matching algorithm is set up to satisfy tourists’ needs. Third, a method that could reduce emission exhaust by optimizing self-driving tour routes is proposed, which could control and reduce the damage to urban environments and protect water ecosystems. The proposed algorithm could be used as the embedded algorithm of tour recommendation systems or the reference algorithm for planning urban tourism transportation. Especially in peak tourism season, it could be used as an effective method for tourism and traffic management departments to direct traffic flow.

Keywords: urban water bodies; water scenery tourism; low-carbon tour route; DIANA algorithm; ECER mode

1. Introduction

In tourism projects, water scenery tourism is a very popular type. It is a collective name for tourism activities that rely on water resources in scenic spots. Water scenery tourism mainly relies on water resources; thus, scenic spots constructed on water not only have the function of providing sightseeing locations but also aim to maintain urban ecological environments, regulate water circulation, regulate the urban temperature and beautify the urban environment. Thus, it is necessary to protect urban scenic water spots [1]. Urban water scenery tourism resources are relatively scarce. People have hydrophilic
psychology; thus, urban scenic water spots must not only have features of landscape design and sightseeing but also other features. Their multiple attributes determine that the urban scenic water spots can provide recreation and rest places and also provide other tourism functions, such as culture, sports, health, etc. In addition to these attributes, since scenic water spots are distributed in the city geospatial environment, they have geospatial features, namely spatial location, distance relation, water area, etc., which assigns them to the research category of tourism geographic information systems. Some scenic water spots are classical and popular ones, which not only attract local tourists but also attract a large number of out-of-town tourists [2]. When the vast majority of scenic spots in a city are scenic water spots, the city can be identified as a water tourism resource city. Urban water tourism resources should be reasonably developed and effectively protected. As an important component of ecosystems, the water cycle plays an important part in regulating the urban natural environment, in which scenic water spots play a key role [3]. Every year, tens of thousands of tourists come to visit these scenic spots, and their activities will impact the urban ecosystems. Under the condition of huge quantities of tourists, it is important to reduce exhaust emission and the damage to urban ecosystems while meeting tourists’ needs and finally to realize low-carbon travel [4–6].

Currently, the research on urban water scenery tourism and low-carbon tourism mainly focuses on the following aspects. The first is the development of the mode and mechanism of water scenery tourism or the products of water scenery tourism. The second is research on the spatial distribution of water tourism resources. The third is the present situation and mechanism of low-carbon tourism. Li [7] analyzed the problems associated with water tourism resources and proposed development prospects. This study mainly addressed existing problems in rural water tourism resources and proposed new ideas for optimizing spatial patterns, changing the mode of production and adjusting the industrial structure. Zhou [8] classified water tourism resources, used the AHP method to conduct an investigation on Zhangjiajie water tourism resources, and brought forward improvement suggestions. The study reached the following conclusions: Zhangjiajie’s water resources are abundant, but the overall quality is low. The main improvements should include water quality monitoring, developing new modes of water tourism and improving the basic transportation system. Xu [9] studied the usage and development of Chizhou city’s water tourism resources and found that that resource integration, brand building and increasing capital investment are necessary. Li [10] studied the development strategy of urban water ecotourism, identifying the issues with water tourism resources and the corresponding countermeasures. Cao [11] conducted statistical analysis of tourists’ cognition and studied urban water scenery tourism. Cao proposed the principles and ideas behind urban water tourism development, development methods for water tourism activities and purification methods for water environments. Fan [12] conducted data mining on the cultural attributes of water tourism resources in the process of tour planning. By mining the cultural attributes of the scenic water spots, Fan reached the conclusion that cultural attributes are indispensable factors for tourism activities. Li [13] studied the spatial distribution of water tourism resources in megacities and proposed an optimization method for water tourism spatial distribution. Li also determined the key issues that should be addressed in water tourism planning in the future. Song [14] conducted research on the products of water culture and reported the following conclusions: cultural attributes should be considered as critical concerns, and the role of cultural attributes should be emphasized in water tourism development. Dai [15] studied the development mode of water sports tourism, focusing on Hubei province, and proposed the possibility of further developing and promoting Hubei province water sports tourism. Huang [16] used the MCN mode to study the innovation in low-carbon tourism marketing and proposed innovative ideas for Heilongjiang province, including personnel training, establishing supply chains for low-carbon tourism and regulating marketing policies and systems. Tao [17] studied tourists’ low-carbon behaviors and came to the conclusion that the Harbin low-carbon tourism could be enhanced by increasing publicity, improving tourism...
facilities, improving public transportation networks, etc. Tian [18] conducted research on the feasibility and development of low-carbon tourism in Guilin and proposed strategies for promoting the construction of Guilin as a low-carbon tourism city. Wang [19] studied the intrinsic motivation of tourists’ low-carbon traveling activities and found that subjective norms, behavior attitudes and moral norms as well as knowledge of low-carbon options influence and drive low-carbon activities, while perceptual behavior control doesn’t have a significant impact. Ren [20] studied transformation strategies for low-carbon tourism in cities and proposed strategies for resource-exhausted cities to develop low-carbon tourism. Liu [21] conducted research on the construction and application of low-carbon tourism evaluation systems, determining the associated problems and proposing suggestions for improvement. The current studies on water scenery tourism and low-carbon tourism mainly focus on development modes, spatial distributions, low-carbon construction methods, etc., which have certain limitations. First, there has been no targeted research on the feature attributes of scenic water spots and their clustering modes. The clustering of scenic water spots is vital, since it is a precondition for low-carbon tour route research. Second, tourists’ behaviors when choosing scenic water spots in terms of the aspects of low carbon, lowest cost and the most benefits have not been studied. These are also preconditions for low-carbon tour route research, since identifying tourists’ needs is the key to realizing the lowest cost and the greatest benefits. Third, the optimization of the tour route via different transportation choices to ensure energy conservation and emissions reduction (ECER) has not been studied. The choice of transportation method is critical in low-carbon tourism research. When a majority of tourists choose optimal routes, it greatly reduces exhaust emissions. Aiming at the current problems and deficiencies of water scenery tourism, this paper proposes a low-carbon tour route algorithm for urban scenic water spots based on an improved DIANA clustering model. It mainly studies the optimization of scenic water tour routes in terms of scenic feature attributes, tourist choices of scenic spots, low-carbon tour route planning, etc. It aims to reduce exhaust emissions and help to protect urban ecosystems. Compared with the methods from the literature, the proposed algorithm focuses on quantitative study. Compared with [7], the research object of the proposed algorithm is urban water tourism resources, not rural water tourism resources. Urban water resources may be influenced by multiple factors and can be easily contaminated. Thus, they need specific strategies. Compared with [8], the proposed method involves an intensive quantitative study of tour routes and transportation modes and puts forward a specific algorithm. Compared with [9–11], the proposed method designs a specific algorithm for scenic water tour routes. It provides an optimized model of urban water tour routes based on geographic information and algorithm optimization. Compared with [12–15], the proposed method mines water tourism’s cultural attributes and sets them as labels to set up the algorithm. It is a concretization of the ideas presented in the literature. The literature [16–21] has analyzed the existing problems in the water tourism industry in terms of carbon imprint and has put forward optimization countermeasures and development paths for low-carbon tourism spatial optimization. Ref. [22] set up a mode of two-stage supply chain management to reduce greenhouse gases emissions. Ref. [23] proposed a closed-loop supply chain, which connects herbal medicine with biofuels. It can effectively increase waste utilization and reduce environmental pollution. Ref. [24] studied the different production strategies of two kinds of innovative green products and then developed three models to promote the reproduction of the innovative green products to realize environmental protection. Ref. [25] aimed to make pure biofuel with a smaller amount of carbon emission and energy utilization through a smart multi-type biofuel manufacturing framework. By using this method, the percentage of impure biofuel can be decreased through the minimized energy consumption. Compared with the literature [22–25], the proposed algorithm has a similar research idea, in which an optimization model is set up and the aim is to reduce greenhouse gas emissions. Refs. [22–25] mainly focused on biofuels, waste recycling, developing innovative green
products and smart multi-type biofuel manufacturing, while the proposed algorithm reduces vehicle exhaust emission by optimizing tour routes and tourist activities.

The research objectives, innovations and novelties are as follows. First, aiming at the specific attributes of scenic water spots, this research proposes an improved DIANA algorithm, which generates scenic spot clusters on the basis of scenic water spot attributes. Thus, the clustering results are more accurate, conforming to tourists’ needs. Second, based on the clustering results, a tourist interest matching algorithm is set up to precisely mine the scenic water spots and enhance tourist satisfaction. Third, aiming at the issues surrounding low-carbon traveling, a low-carbon tour route planning algorithm based on search optimization is established, which can decrease exhaust emissions from self-driving vehicles to protect urban ecosystems and water resources.

2. Urban Scenic Water Spot Spatial Clustering Based on the Improved DIANA Algorithm

Urban scenic water spots have different feature attributes from other scenic spots. Their main bodies are water areas. They have the functions of sightseeing, recreation, culture transmission, and sports and health activities. When tourists choose the scenic water spots to visit, they usually have an intrinsic motivation related to their specific attributes, such as enjoying the scenery, boating, watching waterfowl, experiencing marine cultures and customs, swimming, diving, tasting seafood, etc. Tourists’ intrinsic motivations will directly determine the choice of scenic water spots. Usually, tourists’ intrinsic motivations match scenic water spots’ attributes [26,27].

According to their attributes, scenic water spots can be divided into several groups: recreation and sightseeing, boating, water culture and custom, and water sports. The recreation and sightseeing group includes all the leisure activities involved when visiting and viewing the water scenery and related supporting facilities. Boating includes sightseeing while on board a boat. The water culture and custom group involves appreciating the historical culture and customs of particular water areas. Water sports includes any physical sport played on, in or around water. Scenic water spots have four attributes: popularity degree, minimum cost, traveling time and spatial distance. The popularity degree is an average value that reflects tourists’ enthusiasm for a particular spot. The minimum cost is the lowest cost of visiting one scenic spot for one tourist (unit: ¥ yuan). The traveling time is the average suitable and comfortable leisure time that could be consumed in visiting one scenic spot (unit: hour). The spatial distance is the geospatial distance between one tour route’s starting point and one scenic spot, which is calculated by the longitude and latitude, and directly determines the traveling cost and exhaust emission. In the modeling of clustering and tour route planning, the groups and attributes should be quantified as a first step.

2.1. The Modeling of the Attribute Quantification Matrix for Scenic Water Spots

According to the principle of the clustering method, dots in space with the close attribute relationships can be classified into one cluster. The tour process is usually constrained by multiple factors. Thus, scenic spot recommendations should consider both classifications and attributes. The key to modeling the scenic water spot spatial clustering algorithm is quantifying the classifications and attributes, as is done in definitions 1.1–1.5 as follows.

**Definition 1.1.** Scenic water spot classification factor $k_{i(0)}$ and attribute factor $k_{2(i)}$. The way in which one scenic water spot $s_i$ is distinguished from another is defined as the scenic water spot classification factor, noted as $k_{i(0)}$, in which $i$ is the footnote of $k_{i(0)}$. Factor $k_{i(0)}$ is an important element to determine the cluster to which the scenic spot belongs. Some factors determine the smart machine’s judgement on whether one scenic spot $s_i$ matches a tourist’s interests and needs and then determine whether to absorb the scenic spot into the cluster. These factors are...
defined as attribute factors, noted as $k_{2(i)}$, in which $i_2$ is the footnote of $k_{2(i)}$. Each scenic water spot has different factors $k_{(i)}$ and $k_{2(i)}$, which form its inner function to match tourists’ needs. According to the definition, scenic water spot $l_{a(i)}$ probably belongs to one type of $m$ factor $k_{(i)}$, while it must have $n$ number of factors $k_{2(i)}$, and $0 < i_s < m$, $0 < i < n$. $i_s$, $i_2 \in Z^+$. 

**Definition 1.2.** Scenic water spot classification factor vector $k_1$ and attribute factor vector $k_{2(i)}$. Store the $m$ number of factors $k_{(i)}$ in the sequence of $i_1$ into the $1 \times m$ dimension row vector $k_{(i)}$. The row vector $k_{(i)}$ is defined as the scenic water spot classification factor vector $k_1$. Store the $n$ number of factors $k_{2(i)}$ in the sequence of $i_2$ into the $n \times 1$ dimension column vector $k_{2(i)}$. The column vector $k_{2(i)}$ is defined as the scenic water spot attribute factor vector $k_{2(i)}$. Vector $k_{(i)}$ and $k_{2(i)}$ meet the following conditions:

1. The dimensions of $k_{(i)}$ and $k_{2(i)}$ are $1 \times m$ and $n \times 1$, respectively.
2. The $k_{(i)}$ and $k_{2(i)}$ are fully ranked, and the ranks are $m$ and $n$, respectively.
3. The $k_{(i)}$ and $k_{2(i)}$ are the basic vectors of classification factors and attribute factors, which can be used to create a topological matrix.
4. When the scenic water spot $l_{a(i)}$ is confirmed, the elements for $k_{(i)}$ and $k_{2(i)}$ are confirmed, namely, one scenic water spot $l_{a(i)}$ always relates to one $k_{(i)}$ and one $k_{2(i)}$.

**Definition 1.3.** Scenic water spot attribute factor topology vector $\Delta k_{2(i)}$. The $1 \times q$ dimension topology vector that is formed by arbitrary one element $\forall k_{2(i)}$ of $k_{2(i)}$, is defined as a scenic water spot attribute factor topology vector, noted as $\Delta k_{2(i)}$. This vector represents the specific values of the factor $k_{2(i)}$. Its element is noted as $\Delta k_{2(i,j)}$, in which $0 < j \leq q$, $j, q \in Z^+$. 

**Definition 1.4.** Scenic water spot attribute matrix $K$. The $(\max i_2 + 1) \times q$ dimension topology matrix that is formed by the basic vector $k_1$ and $k_{2(i)}$ is defined as the scenic water spot attribute matrix, noted as $K$. The matrix row represents the vector $k_{(i)}$ and $\Delta k_{2(i)}$. The first row is $k_{(i)}$, and the other $\max i_2$ number of rows is $\Delta k_{2(i)}$. The matrix column involves the related elements of $k_{(i)}$ and $\Delta k_{2(i)}$, and $\max j$ is the maximum dimension of $k_{(i)}$ or $\Delta k_{2(i)}$. One scenic water spot relates to one specific $K$. According to the definition, the elements of $K$ meet the following conditions:

1. Row rank and column rank: $\text{rank}(K)_{r} \leq \max i_2 + 1$, $\text{rank}(K)_{c} \leq q$.
2. When $i_2$ obtains the maximum value $n$, the row meets $\text{rank}(K)_{c} = n + 1$.
3. Arbitrary row $\forall k_{(i)}$ always has one non-zero element; the other elements are 0.
4. Arbitrary rows $\forall K_{(i)}$ and $\forall K_{(i)}$ or arbitrary columns $\forall K_{(i)}$ and $\forall K_{(i)}$ are nonlinearly correlated.

According to the definition and conditions, the matrix $K$ for one scenic water spot $l_{a(i)}$ is modeled as Equation (1).

$$
K = \begin{bmatrix}
\begin{bmatrix}
k_{1} \\
\Delta k_{2(i)}
\end{bmatrix}, & \text{s.t. } i_2 \in (0,n], i_2 \in Z^+
\end{bmatrix},
\begin{bmatrix}
k_{1(1)} & k_{1(2)} & k_{1(3)} & \ldots & 0 & 0 \\
\Delta k_{2(1,1)} & \Delta k_{2(1,2)} & \ldots & 0 & \ldots & 0 \\
\Delta k_{2(n,1)} & \Delta k_{2(n,2)} & \ldots & \Delta k_{2(n,j)} & \ldots & \Delta k_{2(n,q)}
\end{bmatrix}
\end{bmatrix}
$$
The scenic water spot classification factors and attributes should be quantified to set up the clustering model. Scenic water spot classification factors include \( k_{1(1)} \): recreation and sightseeing; \( k_{1(2)} \): boating experience; \( k_{1(3)} \): culture and custom; \( k_{1(4)} \): water sports. The factors \( k_{1(0)} \) form the vector \( k_1 \). \( k_{1(0)} \) are the contribution factors introduced into DIANA. The definitions 2.1–2.5 are introduced as follows.

**Definition 2.1.** Scenic water spot initial clusters are defined as the scenic water spot initial cluster \( S \). Take one city’s downtown area as the research range. Absorb its urban scenic water spots into one cluster; this cluster is defined as the scenic water spot initial cluster \( S \). The arbitrary one scenic spot in \( S \) is element \( S(i) \).

2.2. Scenic Water Spot Spatial Clustering Model Based on the Improved DIANA Algorithm

DIANA is a clustering algorithm with the mode from top to bottom. Using this process, all the dots are initially gathered in a cluster, and then they are divided into several clusters in the control of clustering objective function. The core step of DIANA is to establish the division criteria. The traditional DIANA algorithm relies on dots’ spatial distance. To avoid deviations, the factor normalization parameter \( \zeta \) is introduced. According to each factor’s value range, the parameters are set as follows.

\[
\begin{align*}
\Delta k_{2(1)}: & \quad \Delta k_{2(1,1)}: p_s \in (0, 0.25]; \quad \Delta k_{2(1,2)}: p_s \in (0.25, 0.50]; \quad \Delta k_{2(1,3)}: p_s \in (0.50, 0.75]; \\
\Delta k_{2(2)}: & \quad \Delta k_{2(2,1)}: c_o \in (0, 100]; \quad \Delta k_{2(2,2)}: c_o \in (100, 200]; \quad \Delta k_{2(2,3)}: c_o \in (200, 300]; \\
\Delta k_{2(3)}: & \quad \Delta k_{2(3,1)}: t_c \in (0, 2.0]; \quad \Delta k_{2(3,2)}: t_c \in (2.0, 4.0]; \quad \Delta k_{2(3,3)}: t_c \in (4.0, 6.0]; \\
\Delta k_{2(4)}: & \quad \Delta k_{2(4,1)}: d_a \in (0, 3.0]; \quad \Delta k_{2(4,2)}: d_a \in (3.0, 5.0]; \quad \Delta k_{2(4,3)}: d_a \in (5.0, 10.0]; \quad \Delta k_{2(4,4)}: d_a \in (10.0, 15.0]; \\
\end{align*}
\]

According to the matrix \( K \) and \( K_{i(\zeta)} \), the scenic water spot spatial clustering model based on the improved DIANA algorithm is set up.
Definition 2.2. Scenic water spot cluster \( S(i) \) and element \( S(i,j) \). The number of elements \( S(i) \) in \( S(0) \) are divided into \( k \) number of clusters, and each cluster is noted as \( S_{(i)} \). Each cluster has \( n(i) \) number of elements, in which \( 0 < i \leq k \), \( 1 < k < n \), \( \sum_{i=1}^{k} n(i) = n \). The element in cluster \( S(i) \) is noted as \( S(i,j), 0 < j \leq n(i), n,k,i,j,n(i) \in N \).

According to the definition, the clusters meet the following conditions:

1. Elements in the same cluster have a relatively strong correlation, while elements in different clusters have a weak correlation.
2. Each cluster is a non-empty set, namely \( \forall S(i) \neq \emptyset \).
3. Arbitrary element \( S(i) \) only belongs to one cluster \( S_{(i)} \), namely \( S(i) \cap \bigcup_{i} S(i) = \emptyset \), and \( S(i,j) \) only relates to one single combination \((i,j)\).
4. The union of all the clusters is \( S(0) \), namely \( S(0) \cap \ldots \cup S(i) = S(0) \).

Definition 2.3. Clustering criterion and the objective function \( f(S(0),S(i)) \). Whether one scenic water spot \( S(i) \) can be absorbed into cluster \( S_{(i)} \) is determined by an objective function, noted as \( f(S(0),S(i)) \). In the improved DIANA algorithm, the objective function is determined by multiple factors in vectors \( k_1 \) and \( k_2(i) \). When the matrix \( K_1(i) \) and \( K_2(i) \) for \( S(i) \) and \( S(i) \) are confirmed, the function \( f(S(0),S(i)) \) between them is confirmed. The modeling process of function \( f(S(0),S(i)) \) is as follows:

1. Confirm the vector \( k_1(k) \) and \( k_2(i) \) for the \( n \) number of elements \( S(i) \) in the cluster \( S_{(i)} \).
2. Set up the topology vector \( \Delta k_2(i) \) for each attribute factor \( k_2(i) \).
3. Form the matrix \( K_1(i) \) and \( K_2(i) \) for element \( S(i) \).
4. Aiming at \( S(i) \) and \( S(i) \), extract the non-zero elements from \( K_1(i) \) and \( K_2(i) \). Store them in the \( 1 \times (n + q) \) dimension vector \( T_i \) from the smaller footnotes to the bigger ones.
5. Set up the norm relation of the Euclidean distance between matrix \( T_{i} \) and \( T_{j} \) as in Equation (3). It is the criterion and objective function for the improved DIANA algorithm.

\[
f(S(i), S(j)) = \| T_{i} - T_{j} \|^{2} = (\sum_{i=1}^{n} (k_{1(i,i)} - k_{1(j,j)})^{2}) + \sum_{j=1}^{n} \Delta k_{2(j,j)} - \Delta k_{2(i,j)}^{1/2}
\]

The improved DIANA algorithm is as follows. Store all scenic spots in the initial cluster \( S(0) \), calculate the distances between each scenic spot, and confirm \( k \) number of dots, with the smallest average dissimilarity as the center points. According to the approach principle, calculate the correlations between each non-center point and center point and absorb points into related clusters. The termination condition for the algorithm is all the scenic spots having been absorbed into clusters. Establish the vectors “splinter group” and “old party”, which are used to store the transition data, noted as \( S_0 \) and \( O_0 \). Establish the transition matrix \( S(k \times \max n(0)) \); this is used to store clusters and meets the following conditions:

1. The dimension is \( k \times \max n(0) \).
2. It contains \( k \times \max n(0) \) number of elements, in which \( n \) number of elements are used to store scenic spots \( S(i) \), others are used to store \( 0 \), \( 0 < n \leq k \times \max n(0) \).
3. The row rank is \( 2 < \text{rank}(S) \leq k \), and column rank is \( 0 < \text{rank}(S) \leq \max n(0) \). The matrix should contains at least 2 clusters.
(4) Row sequence is the footnote $i$ of cluster $S(i)$, and the column sequence is the footnote $j$ of $S(i, j)$. The footnote $j$ is determined by the clustering algorithm. According to the above conditions, the matrix is formed, as in Equation (4).

$$
S = \begin{bmatrix}
S(1) & S(2) & \ldots & \ldots & S(i) \\
S(6) & \ldots & S(12) & \ldots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
S(n) & \ldots & 0 & 0 & \vdots
\end{bmatrix}_{\times \max(n(i))}
$$

**Definition 2.4.** Center point $s(i)\ast$ and non-center point $\sim s(i)\ast$. The scenic spots that have the smallest average dissimilarity calculated by the DIANA algorithm are set as the $k$ number of initial dots for clusters $S(i)$, and they are defined as center points $s(i)\ast$. Other points are defined as non-center points $\sim s(i)\ast$.

**Definition 2.5.** Cluster topology edge $I(S(i), S(j))$, cluster structure tree $T_{r(S(i)\ast)}$, cluster spatial range $R_{x(S(i)\ast)}$. In the process of clustering, each scenic spot is absorbed into a related cluster in a certain spatial sequence. Connecting the adjacent two scenic spots in the sequence will form an edge, which is defined as a cluster topology edge $I(S(i), S(j))$. In one cluster, all the scenic spots as well as all the edges $I(S(i), S(j))$ form a cluster structure tree $T_{r(S(i)\ast)}$. Based on the structure, the tree $T_{r(S(i)\ast)}$ expands to the city’s geospatial range and forms each cluster’s special shaped visualized distribution. It is defined as the cluster spatial range $R_{x(S(i)\ast)}$.

According to the definition and the modeling principle, the improved DIANA algorithm is set up as follows.

Step 1: Set up the objective function storage matrix $F_{\ast \omega}$. The $n \times n$ dimension matrix is set up to store the $f(S(0), S(0))$ values. The row numbers are $S(1), S(2), \ldots, S(n)$; the column numbers are also $S(1), S(2), \ldots, S(n)$. The elements are $f(S(0), S(0))$ values; when the row number equals the column number, the element is 0.

Step 2: Calculate the $f(S(0), S(0))$ values between arbitrary scenic spots $S(i)$ and $S(j)$ in $S(0)$, traversing $i \sim (0, n)$, $j \sim (0, n)$. Store all $C(n, 2)$ values in the matrix $F_{\ast \omega}$.

Step 3: Set up $1 \times k$ dimension vector $S$ and $1 \times (n-k)$ dimension vector $O$. $S$ is used to store the $k$ number of center points $S(i)\ast$, while $O$ is used to store the $n-k$ number of non-center points $\sim S(i)\ast$. Step 1–Step 3 forms the first level of the improved DIANA algorithm.

Step 4: Based on $F_{\ast \omega}$, calculate each scenic spot’s average dissimilarity $t(S(i), S(i))$. Average dissimilarity is used to evaluate the average dispersion extent between one object to others. If in a group, there are $n$ number of objects $X(i)$, one object’s average dissimilarity is calculated by Equation (5). The $d(x(i), x(j))$ represents the spatial distance between objects $x(i)$ and $x(j)$. Then, each scenic spot’s average dissimilarity $t(S(i), S(i))$ in $S(0)$ is calculated as in Equation (6).

$$
t(x(i), x(i)) = \frac{1}{n} \sum_{j=1}^{i-1} d(x(i), x(j)) + \frac{1}{n} \sum_{j=i+1}^{n} d(x(i), x(j)) \quad (5)
$$

$$
t(S(i), S(i)) = \frac{1}{n} \sum_{j=1}^{i-1} f(S(i), S(j)) + \frac{1}{n} \sum_{j=i+1}^{n} f(S(i), S(j)) \quad (6)
$$
Step 5: Search and confirm the steady-state vector $\mathbf{S}$. Vector $\mathbf{S}$ stores center points $s_{(i)}$, and each center point relates to one cluster $S_{(i)}$. The choosing of the center point relies on the smallest average dissimilarity. This is the modeling process of the steady-state vector $\mathbf{S}$. Step 5 forms the second level of the improved DIANA algorithm.

Sub-step 1: Store the $k$ number of scenic spots $s_{(i)}$ into the vector $\mathbf{S}$, $i \sim [1,k] \subset \mathbb{N}$. Store the left $n-k$ number of scenic spots into the vector $\mathbf{O}$, $i \sim [k+1,n] \subset \mathbb{N}$. The vectors $\mathbf{S}$ and $\mathbf{O}$ are fully ranked. The element $s_{(i)}$ of $\mathbf{S}$ relates to the scenic spot $s_{(i)}$ element, $i \sim [1,k]$. The element $O_{(i)}$ of $\mathbf{O}$ relates to the scenic spot $s_{(i)}$ element, $i \sim [k+1,n]$.

Sub-step 2: Calculate the average dissimilarity $t(S_{(i)}, \sim s_{(i)})$ for the $k$ number of scenic spots $s_{(i)}$ in vector $\mathbf{S}$, noted as $t(S_{(1)}, \sim s_{(1)})$, $t(S_{(2)}, \sim s_{(2)})$, ..., $t(S_{(k)}, \sim s_{(k)})$.

Sub-step 3: Take the first element $O_{(1)}$ scenic spot $s_{(k+1)}$ in $\mathbf{O}$, with the average dissimilarity $t(S_{(k+1)}, \sim s_{(k+1)})$. Search the related $t(S_{(k+1)}, \sim s_{(k+1)})$ of scenic spot $s_{(k+1)}$ in $\mathbf{S}$, $i \sim [1,k] \subset \mathbb{N}$. Then make a judgement.

1. If there is an scenic spot $\forall s_{(k+1)}$, its related $t(S_{(k+1)}, \sim s_{(k+1)})$ meets $t(S_{(k+1)}, \sim s_{(k+1)}) < t(S_{(k+1)}, \sim s_{(k+1)})$, and then delete $s_{(i)}$ and $s_{(k+1)}$ from $\mathbf{S}$ and $\mathbf{O}$, and store $s_{(i)}$ into $\mathbf{S}$, store $s_{(k+1)}$ into $\mathbf{S}$.

2. If there is no scenic spot $s_{(i)}$ that meets $t(S_{(k+1)}, \sim s_{(k+1)}) < t(S_{(k+1)}, \sim s_{(k+1)})$, keep the $\mathbf{S}$ and $\mathbf{O}$ elements unchanged.

Sub-step 4: Turn back to the former step. Take the second element $O_{(2)}$ scenic spot $s_{(k+2)}$ in $\mathbf{O}$. Search the related $t(S_{(k+2)}, \sim s_{(k+2)})$ of scenic spot $s_{(i)}$ in $\mathbf{S}$, $i \sim [1,k] \subset \mathbb{N}$. Then make a judgement.

1. If there is an scenic spot $\forall s_{(k+2)}$, its related $t(S_{(k+2)}, \sim s_{(k+2)})$ meets $t(S_{(k+2)}, \sim s_{(k+2)}) < t(S_{(k+2)}, \sim s_{(k+2)})$, and then delete $s_{(i)}$ and $s_{(k+2)}$ from $\mathbf{S}$ and $\mathbf{O}$, and store $s_{(i)}$ into $\mathbf{O}$, store $s_{(k+2)}$ into $\mathbf{S}$.

2. If there is no scenic spot $s_{(i)}$ that meets $t(S_{(k+2)}, \sim s_{(k+2)}) < t(S_{(k+2)}, \sim s_{(k+2)})$, keep the $\mathbf{S}$ and $\mathbf{O}$ elements unchanged.

Sub-step 5: Traverse the vector $\mathbf{O}$. Compare each element $O_{(i)}$ of $\mathbf{O}$ with all elements $S_{(i)}$ in the current $\mathbf{S}$. Delete and change the related elements. The algorithm termination conditions are $S_{(i)}$ and $\mathbf{O}$ are fully ranked. Average dissimilarity $t(S_{(i)}, \sim s_{(i)})$ for each $S_{(i)}$ in $S_{(i)}$ is smaller than that of every element $O_{(i)}$ in $\mathbf{O}$.

Step 6: Continue dividing levels. Absorb the non-center point $\sim s_{(i)}$ into the cluster of $s_{(i)}$, and form matrix $S(k \times \max(n))$. Step 6 forms the third level of the improved DIANA algorithm.

Sub-step 1: Store $k$ number of elements $S_{(i)}$ in the steady-state vector $\mathbf{S}$ into the first row’s elements of $S(k \times \max(n))$.

Sub-step 2: Confirm the cluster for $O_{(1)} \sim S_{(i)}$:

1. Take the first element $O_{(1)}$ in the steady $\mathbf{O}$, calculate the $k$ number of objective function values $f(O_{(1)}, S_{(i)})$ between $O_{(1)}$ and $k$ number of elements $S_{(i)}$, $i \sim [1,k] \subset \mathbb{N}$.

2. Search the minimum value $\min f(O_{(1)}, S_{(i)})$ and its related element $S_{(i)}$ and center point $s_{(i)} \sim s_{(i)}$. Note the center point as $s_{(i)} \sim s_{(i)}$ relating to cluster $S_{(i)}$. 


(3) Store center point \( s_{(i)\star} \sim s_{(i)} \) in the first row element of the first column in \( S(k\times\max\{n_0\}) \). Store the element \( O_{(i)} \) in the second element of the same row. Absorb \( O_{(i)} \sim s_{(i)} \) into \( S_{(i)} \).

(4) Connect center point \( s_{(i)\star} \sim s_{(i)} \) with non-center point \( O_{(i)} \sim s_{(i)} \), and form the first edge \( l(S_{(i)\star}, S_{(i)}) \) for \( S_{(i)} \).

Sub-step 3: Confirm the cluster for \( O_{(2)} \sim s_{(2)} \):

(1) Calculate the \( k \) number of objective function values \( f(O_{(2)}, s_{(i)}) \) between \( O_{(2)} \) and \( k \) number of elements \( S_{(i)}, i \in [1, k] \subset N \).

(2) Take the minimum value \( \min f(O_{(2)}, s_{(i)}) \) and its related center point \( s_{(i)\star} \sim s_{(i)} \). Make a judgement:

(i) If the center point \( s_{(i)\star} \sim s_{(i)} \) is \( s_{(i)} \), the element \( O_{(2)} \sim s_{(2)} \) belongs to \( S_{(i)} \). Store it in the first row element of the third column. Connect the non-center point \( O_{(2)} \sim s_{(2)} \) with \( O_{(i)} \sim s_{(i)} \) and form the second edge \( l(S_{(i)}, S_{(2)}) \).

(ii) If the center point \( s_{(i)\star} \sim s_{(i)} \) is not \( s_{(i)} \). Store it in the second row of the first column element in \( S(k\times\max\{n_0\}) \), and store \( O_{(2)} \) in the same row’s second column element. Absorb \( O_{(2)} \sim s_{(2)} \) into cluster \( S_{(2)} \). Connect the center point \( s_{(2)\star} \sim s_{(i)} \) with non-center point \( O_{(2)} \sim s_{(2)} \) and form the first edge \( l(S_{(2)}, S_{(2)}) \).

Sub-step 4: Use the same method to traverse the left \( n-k-2 \) number of scenic spots \( O_{(i)} \), and search the \( \min f(O_{(i)}, S_{(i)}) \) between \( S_{(i)} \). Confirm the cluster \( S_{(i)} \) of certain center point \( s_{(i)\star} \sim s_{(i)} \) for \( O_{(i)} \). Store \( O_{(i)} \sim s_{(i)} \) in the related row of cluster \( S_{(i)} \) in \( S(k\times\max\{n_0\}) \). Form \( l(S_{(i)}, S_{(i)}) \) when absorbing one \( O_{(i)} \). Each cluster \( S_{(i)} \) relates to one structure tree \( T_{(i)}(S_{(i)\star}) \) and spatial range \( R_{(i)}(S_{(i)\star}) \). The algorithm termination conditions are as follows: (1) All elements \( S_{(i)} \) in \( S \) have been stored into the first row elements in \( S(k\times\max\{n_0\}) \). (2) All elements \( O_{(i)} \) in \( O \) have been stored into the related rows. (3) \( S(k\times\max\{n_0\}) \) is fully ranked both in row and column. (4) There is at least one edge \( l(S_{(i)}, S_{(i)}) \) that connects each scenic spot \( s_{(i)} \), or all scenic spots have been absorbed into the tree \( T_{(i)}(S_{(i)\star}) \).

3. Water Tourism Route Algorithm Based on the Optimal ECER Model

Water tourism should meet tourists’ needs and protect the ecological environments. Based on the clusters and tourists’ needs, the mining algorithm is set up to search for the best matched scenic water spots. Under the constraints of limited traveling time, tourists desire to use as much effective time as possible on visiting scenic spots, not on the ferrying process between scenic spots. Thus, it is necessary for tourists to choose the most convenient and effective route under the condition of a certain transportation mode. Motor vehicle is the most frequently used transportation mode by tourists; this consumes large amounts of energy and produces much exhaust gas, damaging the air, water and other environmental elements. When the exhaust emissions exceed the standard, it causes greenhouse effects, acid rain, etc., influencing air and water quality as well as citizens’ health [28,29]. Thus, reducing energy consumption and exhaust emissions is an effective measure to protect urban water resources and ecosystems. Reducing the ferrying time and the exhaust emissions of motor vehicles is critical to realize low-carbon tourism. From the perspective of spatial analysis, the problem becomes searching for the optimal tour route. The first step is to set up the water tourism space model based on scenic spot mining, making all scenic spots match tourists’ needs. The second step is to set up the optimal ECER tour route model based on the water tourism space model [30–32].
3.1. Water Tourism Space Model Based on Scenic Spot Mining

The precondition to set up the water tourism space and the optimal model on ECER to mine the best matched scenic spots. Thus, it is critical to obtain data on tourists’ interests relating to scenic water spots. The smart machine method provides interest labels for tourists to choose, which rely on scenic water spots’ classifications and attributes. When tourists choose certain labels, the smart machine will recommend a certain number of scenic spots that best match the interest labels. Tourists could confirm a desired traveling sequence according to their schedule [33,34]. Definitions 3.1–3.4 are introduced as follows.

**Definition 3.1.** Tourist interest vector \( \mathbf{v} \). According to the scenic water spot attribute matrix \( \mathbf{K} \), tourists choose interest labels, and then the labels are quantified and stored into the vector \( \mathbf{v} \); this vector is defined as the tourist interest vector \( \mathbf{v} \). It represents a group of interest data. The constraints of vector \( \mathbf{v} \) are as follows:

1. Its dimension is \( 1 \times (m+n) \), in which the \( m \) represents the \( m \) number of classifications and \( n \) represents \( n \) number of attributes, \( m > n \).
2. The former \( m \) number of elements store the classification labels. The latter \( n \) number of elements store attribute labels.
3. This ultimately becomes the fully ranked state, namely \( \text{rank}(\mathbf{v}) = m+n \).
4. The element is noted as \( v_{(i)} \), \( 0 < i \leq m+n \), \( m, n \in \mathbb{N} \).
5. As to the former \( m \) number of elements, there is at least one value of 1; other elements have values of 0.

Tourists choose specific classifications and attributes from the vector.

**Definition 3.2.** Tourist interest matching matrix \( \mathbf{V} \). As to the \( n^* \) number of scenic spots \( s_{(i)} \) in \( S_{B_0} \), their attribute matrix \( \mathbf{K} \) could be vectors \( \mathbf{u}_{(i)} \) with the same dimension and rank as \( \mathbf{v} \). Set the vector \( \mathbf{v} \) as the first row of matrix, and set vectors \( \mathbf{u}_{(i)} \) of the \( n^* \) number of scenic spots \( s_{(i)} \) as other rows of the matrix; the formed matrix is defined as the tourist interest matching matrix, noted as \( \mathbf{V} \). According to the definition, vector \( \mathbf{u}_{(i)} \) and matrix \( \mathbf{V} \) meet the following conditions:

1. Its dimension is \( 1 \times (m+n) \), in which the \( m \) represents the \( m \) number of classifications and \( n \) represents \( n \) number of attributes, \( n > 1 \).
2. The former \( m \) number of elements store the non-zero elements of the first row in matrix \( \mathbf{K} \). The latter \( n \) number of elements store the non-zero elements from the second row in the No. \( n+1 \) row in matrix \( \mathbf{K} \).
3. It must be fully ranked, namely \( \text{rank}(\mathbf{u}_{(i)}) = m+n \). Each \( \mathbf{u}_{(i)} \) represents the unique label set for each scenic spot. The element of \( \mathbf{u}_{(i)} \) is noted as \( u_{(i,j)} \), \( 0 < i \leq n^* \), \( 0 < j \leq m+n \), \( m, n \in \mathbb{N} \).
4. Matrix \( \mathbf{V} \) has \( n^*+1 \) rows and \( m+n \) columns, and they are both fully ranked, namely \( \text{rank}(\mathbf{u}_{(i)}) = n^*+1 \), \( \text{rank}(\mathbf{u}_{(i)}) = m+n \).
5. Two arbitrary rows are nonlinearly correlated, and two arbitrary columns are also nonlinearly correlated.
6. As to the former \( m \) number of elements, there is at least on value of 1; the other elements are values of 0.

According to Definitions 3.1 and 3.2, each row’s elements of matrix \( \mathbf{v} \) are stored according to the attribute labels and geographic locations. The first row stores the interest data and positional relations. Other rows are the same data for each scenic spot. The same column relates to one attribute’s data.
Definition 3.3. Interest matching objective function $g(v, u_{(0)})$. The matching function between the vector $v$ and the vector $u_{(0)}$ is defined as the interest matching objective function, noted as $g(v, u_{(0)})$. It determines the correlation between tourists’ needs and scenic spot functions. The function $g(v, u_{(0)})$ is set up as Equation (7), in which $k_{(1)i}$ is the selected classification factor by tourists, $k_{(1)m}$ is the scenic water spot classification factor, $k_{(2)i}$ is the selected attribute factor by tourists, $k_{(2)m}$ is the scenic water spot attribute factor, and $\zeta_{(2)i}$ is the normalization parameter.

$$
g(v, u_{(0)}) = \|v - u_{(0)}\|_2 = (\sum_{i=1}^{m}(k_{(1)i} - k_{(1)m})^2 + \sum_{j=1}^{n}(\zeta_{(2)i}k_{(2),j} - \zeta_{(2)m})^2)^{1/2}
$$

(7)

Definition 3.4. Objective function tab vector $z$. Define the $1 \times n^*$ dimension vector $z$ as the objective function tab vector. It is used to dynamically store the function values $g(v, u_{(0)})$. It meets the following conditions. The final elements of vector $z$ for the function values represent the correlations between $s_{(i)} = u_{(i)}$ and $v$.

(1) The element $z_{(0)}$ relates to the value $g(v, u_{(0)})$.
(2) The initial state is a zero matrix, and the final state is fully ranked $rank(z) = n^*$.
(3) The transition state is a dynamic state of $g(v, u_{(0)})$ value sequencing.

According to the above definitions, the water tourism space model based on scenic spot mining is set up to confirm the best matched scenic water spots for tourists and finally forms the tourism space of the downtown area.

Step 1: Encode each row’s elements of matrix $v$. The first row is the interest vector $v$, and the other rows are vectors $u_{(0)}$. The same column relates to the same attribute. The basic encoding method is as follows:

(1) The former $m$ number of elements in the first row are the classification factors $k_{(1)i}, k_{(1)2}, ..., k_{(1)m}$, relating to the $m$ number of classifications of the vector $k$. The latter $n$ number of elements in the first row are the selected attribute factors, relating to the $n$ number of attributes $k_{(2)i}, k_{(2)2}, ..., k_{(2)n}$. The latter rows have the same storage mode as the first row.

(2) The second row to the last, No. $n^* + 1$ row, is encoded as $u_{(1)}, u_{(2)}, ..., u_{(n^*)}$. The function values $g(v, u_{(0)})$ do not match the code $u_{(0)}$.

(3) The later $n^*$ number of rows do not match the element $z_{(0)}$ of vector $z$.

(4) Step 2: Calculate and store the No. 1 and No. 2 element of vector $z$.

Sub-step 1: Calculate the objective function value $g(v, u_{(0)})$ between the first row $v$ and row $u_{(0)}$, and store the value into the element $z_{(0)}$.

Sub-step 2: Calculate the objective function value $g(v, u_{(0)})$ between the first row $v$ and row $u_{(2)}$, and store the value in the element $z_{(2)}$.

Sub-step 3: Compare the value $g(v, u_{(0)})$ and $g(v, u_{(2)})$.

(1) If $g(v, u_{(0)}) \leq g(v, u_{(2)})$, keep the element values unchanged.

(2) If $g(v, u_{(0)}) < g(v, u_{(2)})$, delete the values in $z_{(0)}$ and $z_{(2)}$. Store $g(v, u_{(2)})$ in $z_{(0)}$ and store $g(v, u_{(0)})$ in $z_{(2)}$.

Sub-step 4: Calculate the objective function value $g(v, u_{(3)})$ between the first row $v$ and row $u_{(3)}$, and store the value in the element $z_{(3)}$ and then make a comparison.
(1) If \( g(v, u_0) \leq g(v, u_2) \):

- If \( g(v, u_3) < g(v, u_0) \leq g(v, u_2) \), delete the storage values of \( z_0 \) and \( z_2 \), store \( g(v, u_0) \) in \( z_{10} \), store \( g(v, u_0) \) in \( z_2 \) and store \( g(v, u_2) \) in \( z_{11} \);
- If \( g(v, u_0) < g(v, u_3) \leq g(v, u_2) \), keep the \( z_0 \) storage value, delete \( z_2 \) storage value, store \( g(v, u_0) \) in \( z_2 \) and store \( g(v, u_3) \) in \( z_{11} \);
- If \( g(v, u_0) \leq g(v, u_2) < g(v, u_3) \), keep the \( z_0 \) and \( z_2 \) storage value, and store \( g(v, u_0) \) in \( z_2 \).

(2) If \( g(v, u_2) < g(v, u_0) \):

- If \( g(v, u_3) < g(v, u_2) < g(v, u_0) \), delete the storage values of \( z_0 \) and \( z_2 \), store \( g(v, u_0) \) in \( z_{10} \), store \( g(v, u_0) \) in \( z_2 \) and store \( g(v, u_2) \) in \( z_{11} \);
- If \( g(v, u_2) < g(v, u_0) < g(v, u_0) \), keep \( z_0 \) storage value, delete \( z_2 \) storage value, store \( g(v, u_0) \) in \( z_2 \) and store \( g(v, u_0) \) in \( z_{11} \);
- If \( g(v, u_2) < g(v, u_0) < g(v, u_3) \), keep \( z_0 \) and \( z_2 \) storage values, and store \( g(v, u_0) \) in \( z_{11} \).

Sub-step 5: Repeat Sub-steps 1–4. Calculate the objective function value \( g(v, u_{0i}) \) between the first row \( v \) and row \( u_{10} \), traverse the range \( i \sim [4, n^*] \subset \mathbb{N} \). Make a comparison between the current \( g(v, u_{0i}) \) and \( g(v, u_{0j}), g(v, u_{0j}), \ldots, g(v, u_{0j-i}) \). The smaller values are always stored in the smaller footnote elements \( z_0 \), in which the vector \( u_{0i} \) with smaller function value \( g(v, u_{0i}) \) has a closer correlation to the interest vector \( v \).

Sub-step 6: Tourists input the quantity of the scenic spots to be visited in a tour day. The criterion to make the selection is set as follows:

1. Tourists have no requirements on cluster \( S_0 \). Input the scenic spot quantity \( \delta \), the smart machine searches the former \( \delta \) number of elements \( z_0 \), and take the related \( g(v, u_{0i}) \), vector \( u_{0i} \) and scenic spot \( s_0 \).

2. Tourists have requirements on cluster \( S_0 \). Tourists choose \( c \) number of \( S_{10} \)∗ and the quantity \( \delta_{10} \) of scenic spots to be visited in each cluster \( S_0 \). Set counter for each cluster as \( \lambda_{0i} \), the initial value is \( \lambda_{0i} = 0 \), \( 0 < c \leq k_i \), \( c \in \mathbb{N} \), \( k_i \) is the total number of cluster \( S_0 \).
   - Search the No. 1 element \( z_0 \) of the vector \( z \). Confirm the related \( g(v, u_{0i}) \), vector \( u_{0i} \) and scenic spot \( s_{0i0} \).
     - (i) If \( s_{0i0} \in \forall S_{10} \)∗, keep \( s_{0i0} \), \( \lambda = \lambda + 1 \), turn to step 3.
     - (ii) If \( s_{0i0} \notin \forall S_{10} \)∗, delete.
   - Search the No. \( i \) element \( z_0 \) of the vector \( z, i \sim [2, n^*] \subset \mathbb{N} \). Confirm the related \( g(v, u_{0i}) \), vector \( u_{0i} \) and scenic spot \( s_{0i0} \).
     - (i) If \( s_{0i0} \in \forall S_{10} \)∗, keep \( s_{0i0} \), \( \lambda = \lambda + 1 \), turn to step 3.
     - (ii) If \( s_{0i0} \notin \forall S_{10} \)∗, delete.
   - Judge the counter \( \lambda_{0i} \):
     - (i) If \( \exists \forall \lambda_{0i} < \delta_{0i} \), go back to step 2 and continue searching the scenic spots for this cluster \( S_{10} \)∗.
     - (ii) If \( \forall \lambda_{0i} = \delta_{0i} \) and \( \sum_{i=0}^{\lambda_{0i}} \delta_{0i} = \delta \), the algorithm ends. Output scenic spots \( s_{0i} \)∗ of each cluster \( S_{10} \)∗.

According to the clusters and scenic spots confirmed by the tourists’ needs and the proposed algorithm, make visualization maps for the clusters, the selected scenic spots and the tourism space \( \Phi \). Figure 1 shows the process to form the tourism space \( \Phi \).
3.2. Optimal ECER Tour Route Algorithm Based on the Water Tourism Space

When tourists travel in the water tourism space, they should travel along the routes with the lowest costs. Meanwhile, considering the protection of the urban ecosystems and low-carbon requirements, when tourists choose motor vehicles, it is necessary for them to reduce the consumption of energy and exhaust emissions, thus decreasing the damage to the city water resources. Usually, tourists tend to take motor vehicles to save travel time, as they are efficient and fast. However, this also brings exhaust emissions. Thus, in the constraint of the total time, decreasing the whole route distance and reducing the ferrying time could help increase the scenic spot visiting time and reduce exhaust emissions [35–37]. The optimal ECER tour route algorithm based on the water tourism space is set up according to definitions 4.1–4.6.

**Definition 4.1.** First order matrix model $W_0$ of the water tourism space. The scenic water spots distributed in the space $\Phi$ are converted to the matrix lattice according to the quantity $\delta$ of scenic spots to be visited. The matrix lattice that is used to store the scenic water spots is defined as the first order matrix model of the water tourism space, noted as $W_0$. The constraints for $W_0$ are as follows:

1. Scenic water spots $s_{(i)}$ are abstracted as the space lattice with longitude and latitude, according to their spatial distribution.

2. Its dimension is $\lceil \sqrt{\delta} \rceil + 1 \times \lceil \sqrt{\delta} \rceil + 1$, the row and column’s full ranks are both $\text{rank}(W_0) = \lceil \sqrt{\delta} \rceil + 1$.

3. Starting from the first element in the first row and ending at the last element of the No. $\lceil \sqrt{\delta} \rceil + 1$ row, the scenic spots $s_{(i)}$ are stored in spatial sequence.

4. Empty elements are set as value 0.

**Definition 4.2.** Motor vehicle path set $R$ and intersection set $P$. Scenic spots are connected by city roads. According to the definition of city geography, city roads can be divided into several grades. In a city, the roads on which private cars, taxis, and online ride-hailing cars could are classified as motor vehicle accessible roads. Store the accessible roads in a set; this set is defined as the motor vehicle path set $R$. Store the intersections of all the accessible roads in set $P$. The arbitrary intersection $P$ of $R_{(i)}$ and $R_{(j)}$ is an accessible intersection. The effective
intersections $P$ are selected to set up the algorithm; these are those which the tourists will be most likely to pass.

**Definition 4.3.** Second order matrix model $W_{(2)}$ of the water tourism space. Take arbitrary two elements $W_{(i_1,j_1)}$ and $W_{(i_2,j_2)}$ in $W_{(1)}$, and extract a certain number of effective intersections $P$ between $W_{(i_1,j_1)}$ and $W_{(i_2,j_2)}$ in $P$. The effective intersection set $P^*$ forms the ferry space between two scenic spots. Define the set that contains all the effective intersections in $P^*$ as the second order matrix model of the water tourism space, noted as $W_{(2)}$. Set the number of the effective intersections in $W_{(2)}$ as $\mathbf{\Phi}$. The constraints for the $W_{(2)}$ are as follows:

1. Effective intersections $P$ are abstracted as the space lattice with longitude and latitude, according to their spatial distribution.
2. Its dimension is $\left\lfloor \sqrt{|P^*|} \right\rfloor + 1 \times \left\lfloor \sqrt{|P^*|} \right\rfloor + 1$, the row and column’s full ranks are both $\text{rank}(W_{(2)}) = \left\lfloor \sqrt{|P^*|} \right\rfloor + 1$.
3. Starting from the first element in the first row and ending at the last element of the No. $\left\lfloor \sqrt{|P^*|} \right\rfloor + 1$ row, the effective intersections $P$ are stored in spatial sequence.
4. Empty elements are set as value 0.

Figure 2 is the process of abstracting the tourism space $\Phi$ into the model $W_{(1)}$ and $W_{(2)}$. The model is the precondition to set up the optimal ECER tour route algorithm based on the water tourism space. Figure 2a shows the distribution of the scenic water spots, connecting roads and road intersections in the water tourism space $\Phi$. Figure 2b shows the scenic water spots’ distribution. Figure 2c shows the built $W_{(1)}$ model based on Figure 2b. Figure 2d shows the road intersection set in the space $\Phi$ and the effective intersection set between the scenic spots $S(2)$ and $S(3)$, noted by the red points in the dashed frame. Figure 2e shows the built $W_{(2)}$ model based on Figure 2d.

![Figure 2](image-url)
matrix, (d) shows the distribution of the extracted road intersections, and (e) shows the abstracted road intersections in matrix.

**Definition 4.4.** Basic ECER unit $E_n$. When a motor vehicle takes tourists from $W_{(i,j)}$ to $W_{(x,j)}$, the average exhaust emission volume of the motor vehicle per kilometer is defined as the basic ECER unit, noted as $E_n$. In the algorithm, it is measured by the gas weight, unit: kg.

When the car model is confirmed, it could be considered that the $E_n$ is a constant value. When the constant value $E_n$ is confirmed, the total exhaust emission of the motor vehicle is directly proportional to the traveling distance in a tour route. Thus, setting up an optimal route model to reduce the total ferrying distance between scenic spots could effectively decrease the exhaust emission volume.

**Definition 4.5.** Scenic spot path structure tree $T_{(o)}$. Based on the set $W_{(i)}$ between $W_{(i,j)}$ and $W_{(x,j)}$, the travel path connecting $W_{(i,j)}$ and $W_{(x,j)}$ through the effective intersection set $P^*$ is created. The structure tree that is formed by $W_{(i,j)}$, $W_{(x,j)}$ and $P$ in set $P^*$ is defined as scenic spot path structure tree $T_{(o)}$. Tree $T_{(o)}$ should meet the following conditions $C_{(T_{(o)})}$:

1. It is a connected graph that does not contain a simple loop.
2. Vertex $W_{(i,j)}$ or $W_{(x,j)}$ is only connected by one edge. Node $P$ must have two connecting edges.
3. Any two vertexes $u$ and $v$ can only have one connecting edge.
4. It is a directed graph from $W_{(i,j)}$ to $W_{(x,j)}$.
5. The edge weight of the vertexes $u$ and $v$ is the spatial distance.

When a tree $T_{(o)}$ contains $k$ number of vertexes, it must have $k - 1$ number of edges. The total ECER volume $E_{n(o)*}$ of the tree $T_{(o)}$ is modeled as Equation (8).

$$E_{n(o)*} = \sum_{c=1}^{k-1} E_n \cdot d_{(u,v)_{c=1}}$$

(8)

**Definition 4.6.** Total ECER volume storage vector $E_{*}$. When the tree $E_{*}$ is built and the total ECER volume is calculated, store the volumes $E_{n(o)*}$ in a $1 \times q$ dimension vector; this vector is defined as the total ECER volume storage vector $E_{*}$, in which $Q$ is the maximum quantity of the tree $T_{(o)}$.

According to the definitions, the optimal ECER tour route algorithm based on the water tourism space is set up as follows:

Step 1: Set up the ECER model for $W_{(2)}$. The model $W_{(2)}$ is the substructure of the tourism space $\Phi$. The optimal tour route algorithm in $W_{(2)}$ is set up first.

Sub-step 1: Build the tree $T_{(o)}$ and calculate the ECER volume $E_{n(o)*}$. Figure 3 shows the process to build the tree $T_{(o)}$.

1. The initial state is shown in Figure 3a. The space lattice is $W_{(2)}$, containing $W_{(i,j)}$ and $W_{(x,j)}$. The starting point is $W_{(i,j)}$ while the ending point is $W_{(x,j)}$. Absorb $W_{(i,j)}$ into $T_{(o)}$.

2. Judge the edge $d_{(w_{(i,j)}, P)}$, in which the values are 1, 2, 5, and 9. Confirm the smallest edge weight $d_{(w_{(i,j)}, P)}$. Search the $W_{(i,j)}$ and edge $e_{(0)}$ of $P_{(s)}$, shown in Figure 3b. Make a judgement: (1) If $e_{(0)}$ and $P_{(s)}$ meet $C_{(T_{(o)})}$, absorb $e_{(0)}$ and $P_{(s)}$ into $T_{(o)}$; (2) If $e_{(0)}$ and $P_{(s)}$ do not meet $C_{(T_{(o)})}$, delete. Search the other points $P_{(s)}$.
(3) Continue searching the smallest edge weight as in Steps (1)–(2). Find the $P_0$, and form the edge $e(2)$, as shown in Figure 3c. Judge that it meets $C_{(T_0)}$, and absorb $e(2)$ and $P_0$ into $T_0$.

(4) Find the $P_0$, and form the edge $e(3)$, as shown in Figure 3d. Judge that it meets $C_{(T_0)}$, and absorb $e(3)$ and $P_0$ into $T_0$.

(5) Continue searching; when $P_5$ is found, it forms a loop; then delete $P_5$. Find $P_7$ and form the edge $e(4)$, shown as Figure 3e. Judge that it meets $C_{(T_0)}$, and absorb $e(4)$ and $P_7$ into $T_0$.

(6) Find the ending point $W_{(i,j)}$ and absorb it into $T_0$, as shown in Figure 3f.

(7) Calculate the ECER volume $E_{(1)}^*$ of $T_0$, and store it into the first element $E_{(1)}^*$ of $E_{s}^*$. Here, the searching process ends.

Figure 3. The process of searching and building the structure tree $T_0$. (a) shows the initial state of the algorithm. (b) shows the searching process of $P_5$ and $e(1)$. (c) shows the searching process of $P_5$ and $e(2)$. (d) shows the searching process of $P_5$ and $e(3)$. (e) shows the searching process of $P_7$ and $e(4)$. (f) shows the searching process of $W_{(i,j)}$ and $e(5)$.

Sub-step 2: Build the tree $T_1$ and calculate the ECER volume $E_{(2)}^*$; store $E_{(2)}^*$ in the second element $E_{(2)}^*$ of $E_{s}^*$, and compare $E_{(1)}^*$ with $E_{(2)}^*$.

1. If $E_{(1)}^* > E_{(2)}^*$, delete $E_{(1)}^*$ and $E_{(2)}^*$, store $E_{(1)}^*$ in $E_{(2)}^*$, store $E_{(2)}^*$ in $E_{(1)}^*$.

2. If $E_{(1)}^* < E_{(2)}^*$, keep $E_{(1)}^*$ and $E_{(2)}^*$.

Sub-step 3: Repeat the above sub-steps. Build the tree $T_0$ and calculate the ECER volume $E_{(3)}^*$; store $E_{(3)}^*$ in the element $E_{(3)}^*$ of $E_{s}^*$, and compare $E_{(3)}^*$ to $E_{(2)}^*$. The element $E_{(2)}^*$ should meet the following conditions:

1. Vector $E_{s}^*$ is fully ranked, namely $W_{(2)} \text{rank}(E_{s}^*) = q$. 


Sub-step 4: Take the total ECER volume \( E_{n(a)}^* \) of the first element \( E_{d(i)^*} \) in \( E_d^* \). Its related tree \( T_{(a)} \) is the optimal path between \( W_{(i,j)} \) and \( W_{(j,i)} \).

Step 2: Set up the ECER model for \( W_0 \). The \( \delta \) number of scenic spots selected by the smart machine cover different clusters and have specific locations. When tourists travel in the space \( \Phi \), they first choose the favorite sequence on the scenic spots, that is, in the model \( W_0 \), the optimal tour route is searched by the model, which finally accumulates the optimal tour route in the whole \( W_0 \) with the lowest energy consumption and exhaust emission. When a tourist confirms a type of sequence, the ECER model for \( W_0 \) is set up as follows:

Sub-step 1: Build the scenic water spot tour sequence vector \( Q \) based on \( W_0 \). Build a \( 1 \times (\delta+1) \) dimension vector \( Q \) to store the \( \delta \) number of scenic spots and the starting point. The constraints for the vector \( Q \) are as follows:

1. The element is \( Q(i) \), and it is fully ranked, namely \( rank(Q(i)) = \delta \).
2. Element \( Q(i) \) is used to store the starting point \( S_i \). Elements \( Q(2) \sim Q(\delta+1) \) are used to store the \( \delta \) number of scenic spots in the tour sequence.
3. Element \( Q(i) \) and \( Q(i+1) \) are the two adjacent scenic spots, and the spatial interval is \( W_2 \).
4. \( Q_0 \sim Q(\delta+1) \) as well as their spatial intervals \( W_2 \) form a complete tour sequence in the space \( \Phi \).

Sub-step 2: Calculate the total ECER volume \( E_{n(i)}^* \) between \( Q_0 \) and \( Q(2) \), and store the value.

Sub-step 3: Calculate the total ECER volume \( E_{n(i)}^* \) between \( Q_0 \) and \( Q(i+1) \), \( 1 < i \leq \delta, i \in N \), and store the value.

Sub-step 4: Calculate the total ECER volume \( E_{n(0)} \) of the complete tour sequence, as shown in Formula (9).

\[
E_{n(0)} = \sum_{i=1}^{\delta} E_{n(i)}^* \tag{9}
\]

4. Experiment and Result Analysis

In order to verify the feasibility and advantages of the proposed algorithm, an experiment was designed. We chose the tourism city Chengdu and its 24 urban scenic water spots as the research range and objects. They cover multiple tour functions, classifications and attributes, which can satisfy different tourists. The basic principle of the experiment is as follows: Confirm the scenic water spots and quantify their classifications and attributes. Use the proposed improved DIANA algorithm to form scenic spot clusters. Confirm the best matched scenic spots according to tourists’ needs, and then search the optimal ECER tour routes. Finally, the tourists’ frequently used electronic maps are set as the control group to allow comparison and verify that the proposed algorithm is feasible and has advantages. The raw data for the experiment are obtained from the basic geographic information data and traffic information data of Chengdu city. The names and attributes of the scenic water spots are obtained from the official website of Baidu Encyclopedia and the Chengdu City Platform for Common Geospatial Information Services. The traffic information data is used to calculate the results are obtained from the Chengdu City Platform for Common Geospatial Information Services and Chengdu City Public Data Open Platform, as well as Chengdu City’s electronic map.
4.1. The Acquisition of the Scenic Water Spots and Calculation Results

The selected typical scenic water spots in the Chengdu downtown area are: \( s(1) \): Jincheng Lake Park; \( s(2) \): Qinglonghu Reservoir; \( s(3) \): Bailuwan Wetland Park; \( s(4) \): Zhonghe Wetland Park; \( s(5) \): East Lake; \( s(6) \): South Lake; \( s(7) \): North Lake; \( s(8) \): Huanhua Brook and Dufu Cottage; \( s(9) \): Chengdu Happy Valley; \( s(10) \): Shengxian Lake; \( s(11) \): Wangjianglou Park; \( s(12) \): Shahe City Park; \( s(13) \): Xinglong Lake; \( s(14) \): People’s Park; \( s(15) \): Xinhua Park; \( s(16) \): Jiaozhi Park; \( s(17) \): Xinzhendi Golf Club; \( s(18) \): Boya Sports Club; \( s(19) \): International Intangible Cultural Heritage Expo Park; \( s(20) \): Lianghe City Forest Park; \( s(21) \): Shahe Park; \( s(22) \): Sansheng Flower Town; \( s(23) \): Funanhe Flowing Water Park; \( s(24) \): Shibashan Cross-Country Park. Table 1 shows each scenic water spot’s normalized classifications and attributes, in which \( k(2) \) relates to the classification and \( k(3) \) relates to attributes: \( k(1) \): recreation and sightseeing; \( k(2) \): boating experience; \( k(3) \): culture and custom; \( k(4) \): water sports; \( k(5) \): the popularity degree; \( k(6) \): the minimum cost; \( k(7) \): the traveling time; \( k(8) \): the spatial distance. The starting point of the tourist is Chengdu railway station, which is used to calculate the spatial distance. The spatial distance is calculated from the longitude and latitude. Figure 4 shows the scenic water spots map of Chengdu city; Figure 4a shows the scenic water spot geographic spatial distribution, and Figure 4b shows the initial cluster \( S_0 \) formed by the extracted scenic water spots.

Figure 4. The sample scenic water spots’ distribution map and the scenic water spot initial cluster \( S_0 \). (a) Map of the Chengdu sample scenic water spots. (b) Scenic water spot initial cluster for the improved DIANA algorithm.
4.2. The Results of Scenic Water Spot Clustering and Cluster Visualization

Calculate each scenic spot’s average dissimilarity using the data in Table 1; the results are shown in Table 2. When the cluster quantities are set as \( k = 2, \ k = 3, \ k = 4, \ k = 5 \), the clusters and related scenic spots are formed, as shown in Table 3. Figure 5 shows the visualization results of the clusters under different cluster quantity conditions; the small blue ranges are the miniature scenic spots. After connecting the two ranges in the sequence of clustering, the topology edges, structure trees and cluster topology ranges are obtained.
Table 2. The output result of each scenic spot’s average dissimilarity (four decimal places).

<table>
<thead>
<tr>
<th>$S(i)$</th>
<th>$i = 1$</th>
<th>$i = 2$</th>
<th>$i = 3$</th>
<th>$i = 4$</th>
<th>$i = 5$</th>
<th>$i = 6$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t(S(i), S(i))$</td>
<td>0.6015</td>
<td>0.5970</td>
<td>0.5812</td>
<td>0.6138</td>
<td>0.5773</td>
<td>0.6070</td>
</tr>
<tr>
<td>$S(i)$</td>
<td>$i = 7$</td>
<td>$i = 8$</td>
<td>$i = 9$</td>
<td>$i = 10$</td>
<td>$i = 11$</td>
<td>$i = 12$</td>
</tr>
<tr>
<td>$t(S(i), S(i))$</td>
<td>0.6029</td>
<td>1.4168</td>
<td>2.4434</td>
<td>0.5972</td>
<td>1.3556</td>
<td>0.6306</td>
</tr>
<tr>
<td>$S(i)$</td>
<td>$i = 13$</td>
<td>$i = 14$</td>
<td>$i = 15$</td>
<td>$i = 16$</td>
<td>$i = 17$</td>
<td>$i = 18$</td>
</tr>
<tr>
<td>$t(S(i), S(i))$</td>
<td>0.6529</td>
<td>1.3824</td>
<td>1.3784</td>
<td>0.6124</td>
<td>0.5912</td>
<td>0.5895</td>
</tr>
<tr>
<td>$S(i)$</td>
<td>$i = 19$</td>
<td>$i = 20$</td>
<td>$i = 21$</td>
<td>$i = 22$</td>
<td>$i = 23$</td>
<td>$i = 24$</td>
</tr>
<tr>
<td>$t(S(i), S(i))$</td>
<td>1.3778</td>
<td>0.5723</td>
<td>1.1301</td>
<td>0.6029</td>
<td>0.5892</td>
<td>1.1488</td>
</tr>
</tbody>
</table>

Table 3. Each cluster’s output result under different cluster quantity conditions.

<table>
<thead>
<tr>
<th>$k$</th>
<th>Center Point $S(i)$</th>
<th>Cluster $S(i)$</th>
<th>Output Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>$k = 2$</td>
<td>$S(20), S(5)$</td>
<td>$S_{(0)}$: $S(20), S(1), S(2), S(3), S(4), S(6), S(7), S(13), S(17), S(18), S(19), S(24)$</td>
<td>$S_{(0)}$: $S(20), S(2), S(4), S(7), S(17), S(18), S(19), S(24)$</td>
</tr>
<tr>
<td>$k = 3$</td>
<td>$S(20), S(5), S(3)$</td>
<td>$S_{(2)}$: $S(5), S(8), S(9), S(10), S(12), S(14), S(15), S(16), S(21), S(22), S(23)$</td>
<td>$S_{(0)}$: $S(20), S(2), S(4), S(7), S(17), S(18), S(19), S(24)$</td>
</tr>
<tr>
<td>$k = 4$</td>
<td>$S(20), S(5), S(3), S(23)$</td>
<td>$S_{(2)}$: $S(5), S(10), S(16)$</td>
<td>$S_{(3)}$: $S(3), S(1), S(6), S(13), S(22)$</td>
</tr>
<tr>
<td>$k = 5$</td>
<td>$S(20), S(5), S(3), S(23), S(18)$</td>
<td>$S_{(4)}$: $S(23), S(8), S(9), S(10), S(12), S(14), S(19), S(21), S(24)$</td>
<td>$S_{(0)}$: $S(20)$</td>
</tr>
<tr>
<td>$k = 6$</td>
<td>$S(20), S(5), S(3), S(23), S(18)$</td>
<td>$S_{(4)}$: $S(23), S(8), S(9), S(10), S(12), S(14), S(19), S(21)$</td>
<td>$S_{(5)}$: $S(18), S(1), S(2), S(4), S(6), S(7), S(17), S(19), S(24)$</td>
</tr>
</tbody>
</table>
4.3. The Results of the Optimal Scenic Spots and Tour Routes

In a tour day, the tourist confirms the interest labels according to the output clusters and scenic spot attributes. The smart machine calculates and generates the interest vector \( \mathbf{v} \) and interest matching matrix \( \mathbf{V} \) and calculates the objective function values \( g(\mathbf{v}, \mathbf{u}_i) \). Table 4 shows the output results of the objective function values \( g(\mathbf{v}, \mathbf{u}_i) \). Based on the Table 4 data, the smart machine recommends each cluster’s optimal scenic spots for the tourist. The tourist confirms five scenic spots to be visited according to the calculation results and then confirms the tour sequence \( \mathbf{Q} \) according to their interests.

Figure 5. The visualization results of clusters under different cluster quantity conditions. (a1) shows the initial cluster \( S_0 \). (a2–b2) show the clusters \( S_1 \) and \( S_2 \) when the number of clusters is \( k=2 \). (a3–c3) show the clusters \( S_3 \)–\( S_5 \) when the number of clusters is \( k=3 \). (a4–d4) show the clusters \( S_6 \)–\( S_8 \) when the number of clusters is \( k=4 \). (a5–e5) show the clusters \( S_9 \)–\( S_5 \) when the number of clusters is \( k=5 \).
Table 4. The output results of the objective function values $g(v, u(i))$.

<table>
<thead>
<tr>
<th></th>
<th>$S(1)$</th>
<th>$S(2)$</th>
<th>$S(3)$</th>
<th>$S(4)$</th>
<th>$S(5)$</th>
<th>$S(6)$</th>
<th>$S(7)$</th>
<th>$S(8)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$g(v, u(i))$</td>
<td>0.2823</td>
<td>0.3490</td>
<td>0.2317</td>
<td>0.3808</td>
<td>0.1749</td>
<td>0.3640</td>
<td>0.3298</td>
<td>1.5042</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>$g(v, u(i))$</td>
<td>2.5119</td>
<td>0.1513</td>
<td>1.4166</td>
<td>0.0877</td>
<td>0.3966</td>
<td>1.4211</td>
<td>1.4199</td>
<td>0.1393</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>$g(v, u(i))$</td>
<td>0.3202</td>
<td>0.3348</td>
<td>1.5067</td>
<td>0.2110</td>
<td>1.0102</td>
<td>0.1772</td>
<td>0.1552</td>
<td>1.0592</td>
</tr>
</tbody>
</table>

In the experiment, the interest labels confirmed by the tourist are (1) Enjoy the scenery and take photos; (2) do not enjoy boating or splashing; (3) there are no requirements for appreciating water scenery cultures or customs; (4) desire to wander around and do leisure sports. In addition, for one single scenic spot: (5) the popularity degree is set at 0.60; (6) the lowest cost is set at 0; (7) visiting time in a scenic spot is set at 1.50; (8) the maximum distance to the starting point is 20 km, and the nearer the better. Based on the interest labels, the interest vector is output $\mathbf{v} = (1.00, 0.00, 0.00, 1.00, 0.60, 0.00, 0.15, 0.00)$.

4.4. Comparison of Results

Based on the Table 5 data and the tour sequences, the models $\mathbf{w}_1$ and $\mathbf{w}_2$ are set up. In the downtown area, the road network is set up to connect the scenic spots, and then the set $\mathbf{P}$ and effective intersections $\mathbf{R}$ are formed. Collect the basic data of the city roads, and use the optimal tour route ECER algorithm to output the low-carbon tour routes. The experiment takes the frequently used motor vehicle as an example and sets the ECER unit as $E_{\text{ton}}=0.197\text{kg}$, which is the exhaust emission volume of a 1.6 L swept volume motor vehicle that travels 1 km. The optimal emission volume $E_{\text{ton}}*$ for each $\mathbf{W}_{(i)}$ is output, and then the total emission volume $E_{\text{ton}}$ of $\mathbf{W}_{(i)}$ is calculated. Before traveling, tourists usually use electronic maps to plan tour routes. The experiment chooses the Tencent map and Gaode map as the control group; the proposed algorithm, Tencent map and Gaode map are represented as PRA, TCA and GDA. The same experimental performance is carried out on the control group, and the results are shown in Table 6. The $E_{\text{ton}}*$ is the first road interval from the starting point to the first scenic spot. The other $E_{\text{ton}}*$ is the road interval between two scenic spots. The $E_{\text{ton}}*$ is the total emission volume. Figure 6 shows the emission volume $E_{\text{ton}}*$ and total emission volume $E_{\text{ton}}$ of the three algorithms under the conditions of different cluster quantities, and the difference value comparison among the three algorithms for the emission volume. Figure 6a–e are the emission volume $E_{\text{ton}}*$ and total emission volume $E_{\text{ton}}$ of the three algorithms under the conditions of cluster quantities $k=1,2,3,4,5$. The blue columns relate to PRA, the orange columns relate to TCA, and the gray columns relate to GDA. Figure 6f–j are the difference value comparisons among the three algorithms of the emission volume under the conditions of cluster quantities $k=1,2,3,4,5$. The blue columns relate to PRA, the orange columns relate to TCA, and the gray columns relate to GDA.

Table 5. The scenic tour sequence vector $Q$ under the cluster quantity conditions.

<table>
<thead>
<tr>
<th></th>
<th>$Q(1)$</th>
<th>$Q(2)$</th>
<th>$Q(3)$</th>
<th>$Q(4)$</th>
<th>$Q(5)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$k=1$</td>
<td>$S(12)$</td>
<td>$S(16)$</td>
<td>$S(10)$</td>
<td>$S(23)$</td>
<td>$S(5)$</td>
</tr>
<tr>
<td>$k=2$</td>
<td>$S(12)$</td>
<td>$S(16)$</td>
<td>$S(10)$</td>
<td>$S(20)$</td>
<td>$S(3)$</td>
</tr>
<tr>
<td>$k = 3$</td>
<td>$S(12)$</td>
<td>$S(16)$</td>
<td>$S(22)$</td>
<td>$S(20)$</td>
<td>$S(3)$</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
<td>--------</td>
<td>--------</td>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>PRA</td>
<td>1.418</td>
<td>2.620</td>
<td>2.955</td>
<td>1.300</td>
<td>1.497</td>
</tr>
<tr>
<td>TCA</td>
<td>1.497</td>
<td>3.034</td>
<td>3.054</td>
<td>1.399</td>
<td>1.576</td>
</tr>
<tr>
<td>GDA</td>
<td>1.556</td>
<td>2.817</td>
<td>3.014</td>
<td>1.615</td>
<td>1.655</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$k = 4$</th>
<th>$S(12)$</th>
<th>$S(16)$</th>
<th>$S(22)$</th>
<th>$S(20)$</th>
<th>$S(1)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>PRA</td>
<td>1.418</td>
<td>2.620</td>
<td>2.955</td>
<td>2.600</td>
<td>5.516</td>
</tr>
<tr>
<td>TCA</td>
<td>1.497</td>
<td>3.034</td>
<td>3.054</td>
<td>2.660</td>
<td>5.634</td>
</tr>
<tr>
<td>GDA</td>
<td>1.556</td>
<td>2.817</td>
<td>3.014</td>
<td>2.955</td>
<td>5.989</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$k = 5$</th>
<th>$S(12)$</th>
<th>$S(16)$</th>
<th>$S(22)$</th>
<th>$S(20)$</th>
<th>$S(1)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>PRA</td>
<td>1.418</td>
<td>2.620</td>
<td>2.955</td>
<td>3.054</td>
<td>5.595</td>
</tr>
<tr>
<td>GDA</td>
<td>1.556</td>
<td>2.817</td>
<td>2.206</td>
<td>2.305</td>
<td>5.930</td>
</tr>
</tbody>
</table>

Table 6. Comparison of the emission volume of the three algorithms under the different cluster conditions (unit: kg). $E_n(0)^*$ stands for the optimal emission volume in each road interval.

Figure 6. The emission volume $E_n(0)^*$ and total emission volume $E_n(\text{tot})$ of the three algorithms under the conditions of different cluster quantities, and the difference value comparison among the three algorithms of the emission volume.
4.5. Experimental Results Analysis

By confirming the experimental environment and collecting experimental data, the proposed algorithm is used to carried out the experiment and output the results. The experimental results are analyzed for the aspects of clusters and related visualization, optimal scenic spots and tour routes, and algorithm comparison.

(1) The analysis of the clusters and related visualization results

The proposed algorithm is used to form the scenic water spot clusters. Table 2 data show each scenic spot’s average dissimilarity, in which the values have relatively large disparities. This shows that scenic spots have discrepancies in the classification and attribute; in addition, they have different functions to meet tourists’ needs. The smaller the average dissimilarity, the closer the scenic spot will approach the cluster center. Meanwhile, the average dissimilarity also reflects the correlation between two scenic spots. The smaller the average dissimilarity, the closer the scenic spot’s comprehensive attributes will approach other scenic spots. According to the cluster quantity, the related quantity of center points is calculated and output. The experimental results conform to the clustering rules.

Analyze the Table 3 data. When the cluster quantities are different, the output center points are also different. When the cluster quantity and center points are confirmed, and each one has different scenic spots. This shows that when the preconditions are different, scenic spots will be absorbed into different clusters, which will influence the selection of the optimal scenic spots and planning of the optimal tour routes. In Table 3, the scenic spots that are absorbed into the same cluster must have the smallest objective function values with the center point. This shows that when the center point meets tourists’ needs, the scenic spots that are close to the center point have strong correlation to the center point’s functions and are thus are recommended to the tourists.

Analyze the Figure 5 cluster visualization results. When the cluster quantity is \( k = 1 \), the cluster is the initial cluster \( S(0) \). It contains all the scenic spots.

When the cluster quantity is \( k = 2 \), the center points are \( S(20) \) and \( S(5) \). The cluster \( S(1) \) and \( S(2) \) have different distributions and form two kinds of structure trees and spatial ranges. \( S(1) \) is relatively dispersed, while \( S(2) \) is relatively concentrated.

When the cluster quantity is \( k = 3 \), the center points are \( S(20), S(5), S(3) \). The cluster \( S(1), S(2) \) and \( S(3) \) have different distributions and form three kinds of structure trees and spatial ranges. \( S(1) \) is relatively dispersed, while \( S(2) \) and \( S(3) \) are relatively concentrated.

When the cluster quantity is \( k = 4 \), the center points are \( S(20), S(5), S(3), S(23) \). The clusters \( S(1), S(2), S(3) \) and \( S(4) \) have different distributions and form four kinds of structure trees and spatial ranges. \( S(1) \) is relatively dispersed, while \( S(2), S(3) \) and \( S(4) \) are relatively concentrated.

When the cluster quantity is \( k = 5 \), the center points are \( S(20), S(5), S(3), S(23), S(18) \). The clusters \( S(1), S(2), S(3), S(4) \) and \( S(5) \) have different distributions and form five kinds of structure trees and spatial ranges. \( S(5) \) is relatively dispersed, while \( S(1), S(2), S(3) \) and \( S(4) \) are relatively concentrated.

By analyzing the clustering results, the following conclusions could be obtained. First, when the smart machine sets different cluster numbers, the clustering results would be greatly different. The more the cluster number is, the more accurate the clustering results on the scenic water spots’ attributes will be, the more likely to extract the scenic spots that match tourists’ needs. Thus, the confirming of the cluster number is critical, it should not be too large or too small. Second, when the clustering numbers are different, the recommended specific scenic water spots are different. When extracting the scenic spots, the
smart machine will preferentially choose the one that best matches tourists’ needs in each cluster. Thus the recommended scenic spots and the finally confirmed tour routes would be different. Third, according to the visualization results, most of the clusters display the shape of centralized distribution, which interprets that the spatial attributes play a critical role in the clustering process. It conforms to the tourism activity law since the urban geospatial environment provides the basic conditions, and the scenic water spots attributes and spatial attributes are both the decisive factors for planning the low-carbon tour routes.

(2) The analysis of the optimal scenic spots and tour routes results

The results in Table 4 show that the objective function values of the scenic spots are totally different. The minimum value is 0.0877, relating to the scenic spot S(12). This shows that its capacities to meet tourists’ needs are the strongest. The maximum value is 2.5119, relating to the scenic spot S(9). This shows that its capacities to meet tourists’ needs are the weakest. Based on the Table 3 clustering results, each cluster’s scenic spots are sequenced in ascending order. The tourist confirms a tour sequence according to their interests and the recommended optimal scenic spots, as shown in Table 5. As to the Table 5 data, when the clusters are different, the recommended optimal scenic spots and tour sequences are totally different. When the tourist confirms a tour sequence, the smart machine will plan an optimal traveling route, which generates the lowest exhaust emissions.

(3) The analysis of the algorithm comparison results

Under the same condition of the tour sequence, the exhaust emission volumes of the three algorithms are different, as shown in Table 6 and Figure 6. When analyzing the Table 6 data and Figure 6a–e, the exhaust emission volume is directly proportional to the traveling distance. As to the arbitrary cluster quantity condition, for the same algorithm, the exhaust emission volumes are different at each interval. When \( k = 1 \), PRA, TCA and GDA all have the highest exhaust emission volume in \( E_{n(2)} \) and lowest one in \( E_{n(3)} \). When \( k = 2 \), PRA, TCA and GDA all have the highest exhaust emission volume in \( E_{n(4)} \) and lowest one in \( E_{n(0)} \). When \( k = 3 \), PRA, TCA and GDA all have the highest exhaust emission volume in \( E_{n(3)} \) and the lowest one in \( E_{n(0)} \). When \( k = 4 \), PRA, TCA and GDA all have the highest exhaust emission volume in \( E_{n(4)} \) and the lowest one in \( E_{n(0)} \). When \( k = 5 \), PRA, TCA and GDA all have the highest exhaust emission volume in \( E_{n(3)} \) and the lowest one in \( E_{n(0)} \). As to each interval and the whole tour sequence, the exhaust emission volume of the PRA is always lower than TCA and GDA.

Figure 6f–j show the exhaust emission volume difference values between TCA and PRA, GDA and PRA. The difference values fluctuate along the interval \( E_{n(i)} \), namely the algorithms have different performance at different intervals, which creates great differences in the whole tour route. When \( k = 1 \), \( k = 2 \) and \( k = 3 \), the GDA exhaust emission is the highest. When \( k = 4 \) and \( k = 5 \), the TCA exhaust emission is the highest. For arbitrary \( k \), the PRA exhaust emission is always the lowest.

It can be seen from the comparison results that the proposed algorithm has great advantages. First, the proposed algorithm applies the parallel search mode to find the optimal result, which could greatly improve the algorithm’s efficiency and determine the optimal result in a short time. Thus, the proposed algorithm is the optimal one. Second, when searching the vehicle traveling paths in the downtown area, the experimental group algorithms tend to choose the main roads and neglect the secondary roads, since they are convenient for vehicles. Thus, the searched route might not be the optimal path. Third, the proposed algorithm not only has advantages in searching the optimal paths but also produces the lowest exhaust emission for low-carbon traveling, since it provides the shortest traveling distance, resulting in the lowest volume of vehicle energy. Fourth, since the proposed algorithm has the above-mentioned advantages, it could be directly used as the embedded algorithm to develop the smart recommendation system, helping to provide one-stop recommendation services in scenic spots and low-carbon tour routes, while the
experimental group algorithms do not have the functions of smart recommendation of scenic spots and low-carbon tour routes. Tourists have to confirm the scenic spots by themselves and then obtain the planned path between two scenic spots. Thus, in the aspect of application convenience, the proposed algorithm also has advantages.

4) Discussion of water protection

The experimental environment is Chengdu city as well as its urban scenic water spots, and the traveling tool is one 1.6 L swept volume motor vehicle. The experiment shows that the motor vehicle traveling along the route created by the proposed algorithm will generate the lowest exhaust emission, lower than that of the routes created by the electronic maps. According to statistics, over 200 million tourists visit Chengdu every year. If every tourist takes a motor vehicle to travel in the city along the proposed tour sequences designed in the experiment, the algorithm will decrease the exhaust emission volume by more than 100,000 tons. Motor vehicle exhaust contains carbon monoxide, carbon dioxide and nitrogen oxide, which pollute the atmosphere, air, and water resources and can cause greenhouse effects and harm human health. The proposed algorithm provides a method to plan optimal traveling routes, which can effectively reduce the motor vehicle exhaust emission volume, control the pollution of the atmospheric environment and water resources, and realize low-carbon tourism and protect urban ecosystems. As to the proposed algorithm, some managerial insights are provided. First, it could be used to develop a low-carbon driving management APP, which could plan the traveling routes with the lowest exhaust emission when the starting point and the terminal point are confirmed. Meanwhile, it could provide real-time traffic conditions and manage the whole trip for self-driving tourists and thus could reduce environmental pollution. Second, on important festivals and vacation days, it could be used to manage the traffic flow and effectively control the total volume of self-driving vehicles, helping to relieve traffic pressure. It is also useful for increasing the public transportation frequency, to encourage tourists to travel by public transportation and reduce the use of vehicles. Third, the tour routes planned by the proposed algorithm could be highlighted on the electronic maps or published as a paper version and provided to tourism management or traffic管理部门. On important festivals or vacation days, when the tourist travel flow is large, the highlighted routes could be managed to relieve traffic congestion.

4.6. Comparison with the Algorithms in the Literature

Compared with the methods and algorithms in the literature, the proposed algorithm has some novelties. Ref. [5] confirmed the popular urban scenic spots by mining the number of check-ins. In the aspect of data accuracy, the proposed algorithm is superior. It directly confirms tourists’ interest labels to match scenic water spots’ attributes, which is more elaborate and accurate. Ref. [6] studied an ant colony algorithm for better performance of planning the vehicle route, with emphasis on the algorithm optimization and improvement. By comparison, the novelty of the proposed algorithm is different. It combines the shortest path searching with low-carbon traveling, water tourism and environmental protection. Its aim is to set up the mode of low-carbon water tourism and reduce the emission of greenhouse gases. Ref. [9] studied a route planning system for intelligent driving; the proposed algorithm is not used for intelligent driving but for low-carbon driving, whose service targets are ordinary vehicles, not intelligent vehicles. Ref. [17] studied how to choose the proper intelligent driving mode for drivers using intelligent driving systems. Its aim was to reduce exhaust emissions by avoiding bad driving habits. However, intelligent vehicles are not widely produced and used. The vast majority of vehicles cannot reduce the exhaust emissions by controlling the driving modes. Thus, the novelty of the proposed algorithm is to reduce the travel distances, control the energy consumption and finally reduce the exhaust emissions of ordinary vehicles. Ref. [22] set up a scenic spot recommendation system using the visiting frequency of different tourists, which involved scenic spot recommendations. By comparison, the novelty of the proposed
algorithm is recommending scenic spots by matching tourists’ interests. In addition to recommending scenic spots, it also plans low-carbon tour routes.

5. Conclusions

Water tourism is different from other types of tourism. It is closely related to ecosystems. Planning water tourism activities and related tour routes should protect the environment. Based on energy conservation and emission reduction as well as low-carbon tourism, this paper sets up a low-carbon tour route algorithm for urban scenic water spots based on an improved DIANA clustering model. Taking urban water scenery tourism as the research context, the current problems and research status of water tourism are analyzed. An urban scenic water spot clustering model based on an improved DIANA algorithm is proposed. It quantifies the scenic water spot classifications and attributes into specific data based on the ecological tourism mode. The classifications and attributes are set as the parameters to set up the clustering algorithm, and the scenic water spot clusters are obtained. This is the precondition for the smart machine to recommend the optimal scenic spots. Based on the clustering algorithm, by analyzing the requirements of the tourists in ecological tourism, the water tour route algorithm based on the optimal ECER model is set up, in which the tourists’ interest data is mined to set up the water tourism space model. The confirmed scenic spots are arranged by the tourists to form the interested tour sequence and the traveling routes. The experiment shows that the proposed algorithm is feasible and has advantages for energy conservation and emission reduction as well as low-carbon tourism over frequently used electronic maps.

Aiming at low-carbon tourism and ecological travel, this paper proposes a feasible method, which could be embedded into a smart recommendation system to provide optimal scenic spots and plan tour routes. On the aspect of theoretical and methodological contributions, first, this paper proposes an improved DIANA scenic water spot clustering algorithm, which breaks down the limitations of spatial distance and sets the specific attributes of scenic water spots as the key parameters. It also optimizes the clustering algorithm and enhances the algorithm accuracy. Thus, the proposed algorithm has made contributions to the clustering theory. In addition, it combines the clustering method with the low-carbon idea, which contributes to urban ecosystem protection. Second, as to the scenic water spot tourists, this paper sets up a quantitative model for scenic water spot interests. Different from the mode of data mining from big data, the proposed algorithm directly confirms tourists’ interest labels and matches the best scenic spots. Third, this paper innovatively sets up an algorithm model that can reduce the total exhaust emissions by searching the vehicles’ shortest traveling path while meeting the needs of self-driving tourists. Thus, the proposed algorithm is a comprehensive method that combines the tourists’ interests with low-carbon goals, which simultaneously satisfies tourists’ needs and protects urban water ecosystems. Thus, it makes a contribution to ecotourism construction. Fourth, the proposed algorithm can be directly used as the embedded algorithm for the smart recommendation system. Especially for self-driving tourists, the proposed algorithm can reduce the energy consumption and exhaust emission to protect urban water resources. In addition, as an urban tour route planning algorithm, the proposed algorithm’s core aim is to realize a low-carbon footprint and ensure environmental protection. It could be used by tourism management or traffic management departments. During important festivals or vacation days, when the tourist travel flow is large, the low-carbon tour routes could be managed to relieve traffic congestion.

In further research, the following elements of water tourism could be studied. First, the importance of scenic water spots to urban ecosystems is great, and people’s activities have a great impact on scenic water spots. Thus, deep mining the tourists’ activities in scenic water spots and conducting research on the relationship between tourists’ activities and scenic water spots’ impact on urban ecosystems should be considered, with more precise attributes and interest labels designed and more elaborate clusters created. Second, more elaborate research on tourists’ activities could be performed, in which the activities...
could be classified to determine their different impacts on related scenic water spots. This aims to determine a more accurate recommendation algorithm and provide more precise and elaborate scenic spots for specific groups of tourists. Third, in the paper, motor vehicles such as taxis and private cars are set as the research objects and tools. In future study, additional transportation modes could be studied. According to the tourists' needs, multiple collocations of transportation modes can be provided for tourists to further optimize the exhaust emissions, to satisfy different travel demands, and finally to minimize the damage to urban water resources and ecosystems.

**Author Contributions:** Conceptualization, X.Z. and D.Z.; methodology, X.Z. and D.Z.; validation and formal analysis, M.S. and J.T.; investigation, data resources and data processing, J.T.; writing—original draft preparation, X.Z.; writing—review and editing, X.Z., D.Z. and M.S.; visualization, X.Z. and J.T.; supervision, X.Z. and D.Z.; project administration and funding acquisition, D.Z. and X.Z. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the Independent Research Project of State Key Laboratory of Geo-Information Engineering (Grant No. SKLGIE2020-ZZ-3), the founder is the State Key Laboratory of Geo-Information Engineering, and by the Leshan Science and Technology Project (Grant No.20RKX0007 and No.20ZRKX006), the founder is Leshan Science and Technology Bureau.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The data presented in this study are available from the author upon reasonable request.

**Acknowledgments:** The authors would like to thank the postdoctoral innovation practice base of Sichuan province of Leshan Vocational and Technical College and the Computer Science Postdoctoral Mobile Station of Sichuan University. We are also grateful for the support of the State Key Laboratory of Geo-Information Engineering. We also thank the editors and reviewers for their valuable comments.

**Conflicts of Interest:** The authors declare no conflicts of interest.

**References**


