GNSS-Based Scene Recognition by Means of Machine Learning

Yuting Yang, Di He(✉), and Wenxian Yu

Shanghai Key Laboratory of Navigation and Location-Based Services, School of Sensing Science and Engineering, School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, Shanghai 200240, People's Republic of China
dihe@sjtu.edu.cn

Abstract. Scene recognition has attracted considerable attention in the field of navigation in recent years since the signals can be used for navigation depending on the specific scene in which the receiver is located. Scene recognition can assist in developing a scene-adaptive navigation system that selects different positioning algorithms or sensors for navigation, enabling accurate positioning and navigation in different environments. In this study, we propose a supervised machine learning model. Based on the multi-satellite GNSS observation, 16 features are designed, and the most effective 8 features are selected from them as model inputs through the Chi-square test. Support Vector Machine and Random Forest are utilized as classification models with the strategy of result fusion to divide the outdoor navigation scenes into four types: Highway, Suburb, Urban Canyon, and Avenue. The test results show that our proposed scene recognition algorithm achieves an overall recognition accuracy of 94.37%, with each of the four types of scenes having a recognition accuracy exceeding 93%, which shows better performances than other existing methods.

Keywords: Scene recognition · GNSS · Machine Learning

1 Introduction

The Global Navigation Satellite System (GNSS) is a space-based radio-navigation and positioning system that provides users with all-weather 3D coordinates, speed, and time information at any point on the Earth's surface or near-Earth space [1]. In recent years, with the increasing number of satellite constellations and the continuous updating of positioning algorithms, GNSS-based positioning and navigation are becoming more and more precise and efficient, showing vigorous vitality. However, current GNSS positioning algorithms still have limitations in different scenes. A particular GNSS positioning algorithm is

Supported by the National Natural Science Foundation of China under Grant Nos. 61971278, and 62231010.

F. Gao et al. (Eds.): Chinacom 2023, LNICST 590, pp. 528–546, 2024.
https://doi.org/10.1007/978-3-031-67162-3_33

not well adapted to all scenes and provides robust positioning results. This is because the navigation signals that the receiver can receive and their quality depend on the scene in which the receiver is located. For example, in an open environment such as a highway, there is no obstruction between the satellite and the receiver, most of the signals received are Line-of-Sight (LOS) signals. The number of visible satellites is enough, and the signals are of high quality, so a simple positioning algorithm such as Single Precise Positioning or Loosely Coupled GNSS/Inertial Navigation System (INS) Integration can be used to achieve high-precision positioning. However, in urban areas with dense high buildings, due to multipath interference and None-Line-of-Sight (NLOS) reception caused by occlusion, more complex positioning algorithms such as tightly Coupled GNSS/INS Integration, shadow matching, or NLOS signal filtering are required to achieve high-precision positioning [2]. Scene recognition refers to deriving the scene in which the receiver is located based on the analysis of the information received by the sensor. The scene here refers to the environment, such as indoor/outdoor, open sky, city, etc. In recent years, scene recognition has attracted considerable attention in the field of navigation, because it can provide information about the potential quality of navigation signals, and can help build a scene adaptive navigation system that selects different positioning algorithms or sensors in different scenes so that the system can achieve precise positioning and navigation in different scenes. In addition, scene recognition can also assist with other tasks, such as path planning and so on.

At present, many researchers have proposed a variety of GNSS positioning algorithms for different scenes. For example, [3] proposes a shadow matching algorithm for urban canyon areas. In [4], the author proposes a precise point positioning (PPP)/INS/Vision/Light Detection and Ranging (LiDAR) tightly coupled algorithm for urban scenes. [5] develops a deep learning-based wireless positioning technology for indoor scenes. However, these algorithms are useful only if we know which environment the receiver is located in. Currently, according to the sensors used, scene recognition methods can be broadly categorized as GNSS-based scene recognition, vision-based scene recognition, depth map-based scene recognition, etc. [2]. Scene recognition based on GNSS signals uses the signals received by the GNSS receiver to establish some features to determine the scene. Vision-based scene recognition uses a camera such as a fisheye camera or a front camera combined with certain algorithms to achieve scene recognition. Depth map-based scene recognition uses LiDAR and RGB-D cameras to obtain depth maps and further perform scene recognition. [6] quantitatively measures the Carrier-to-Noise Ratio (C/N0) of the GNSS signal outside residential areas, in urban canyons, and within residential areas using a highly sensitive GNSS receiver. It is demonstrated that the strength of occluded signals can be reduced by $20dB \cdot Hz$ compared to LOS signals, and it is possible to recognize different scenes by C/N0 of GNSS signal. [7] achieves classification of urban, rural, and off-road scenes by performing color extraction and texture analyses on images obtained from a low-cost camera placed at the front of the vehicle. However, this method was only applicable during the daytime and the classification accuracy

was low. [8] proposes an indoor-outdoor scene recognition algorithm based on ResNet, which can address the problem of indoor-outdoor scene recognition in both color and depth images. The algorithm achieves an average accuracy of 96% for both indoor and outdoor scenes. However, only two types of scenes are defined in this research. Besides, since it uses both color and depth images as data, the model training consumes a lot of time and is highly demanding on computational power.

Compared with other sensors for scene recognition, the one based on GNSS signals has obvious advantages. First, GNSS-based scene recognition can save costs. The reason why we need scene recognition is that it may be possible to build an adaptive navigation system that can be applied to various scenes. Other sensors such as LiDAR and fisheye cameras can only recognize the scene but can't perform navigation, so choosing them will introduce a lot of unnecessary costs. Secondly, scene recognition based on GNSS signals has low arithmetic requirements and fast recognition speed, so it can meet the demand for real-time recognition. Finally, scene recognition based on GNSS signals is not affected by the alternation of day and night which is an obvious advantage over vision-based ones.

Since scene recognition based on GNSS signals has the above three advantages, this study proposes a supervised machine learning model. Based on the multi-satellite GNSS observation, 16 features are designed, and the most effective 8 features are selected from them as model inputs through the Chi-square test. We use Support Vector Machine (SVM) and Random Forest (RF) as classification models with the idea of result fusion to divide the outdoor navigation scenes into four classes: Highway, Suburb, Urban Canyon, and Avenue. The reason why we do not consider indoor scenes in this study is that indoor scenes are almost impossible for GNSS positioning. Due to the obstruction of ceilings, the number of visible satellites is very small, and the signal quality is low. The article is structured as follows: Sect. 2 introduces the state of art on scene recognition based on GNSS signals; Sect. 3 introduces our proposed scene recognition algorithm, including three parts: feature extraction, data preprocessing, and classifier; the experiments and results are recorded in Sect. 4; and finally the conclusions are given in Sect. 5.

2 Related Work

Since GNSS-based scene recognition has the unique advantages mentioned above, many researches have been focused on it and have obtained satisfying results. For example, [9] develops a scene recognition framework based on GNSS signals to categorize different terrestrial environments into indoor, urban, and open-sky. The framework first utilizes a Hidden Markov Model to recognize indoor and outdoor environments based on the availability and C/N0 of GNSS signals. The outdoor environment is then further categorized into urban and open sky using a fuzzy inference system. Test results in different environments show an overall recognition accuracy of 88.2%. However, the data used in this study is static so it

does not meet the dynamic requirements of navigation. [10] classifies the scenes into four types: deep indoor, shallow indoor, semi-outdoor, and open outdoor. Based on the Hidden Markov Model, the author analyzes the results of multi-constellation satellite signals on scene recognition performance in detail and proves that the use of multi-constellation signals can significantly improve the accuracy of scene recognition. In addition, the author proposes a Recurrent Neural Network (RNN)-based scene recognition method with an overall recognition accuracy of 98.65%. However, deep learning is computationally intensive, uninterpretable, and dependent on data, which may not be effective when the data is insufficient. [11] proposes a supervised machine learning model that can robustly classify four types of scenes: canyons, urban, trees, and open-sky, with an overall classification accuracy of 82.40%. However, the feature vector proposed in the algorithm is not rigorously analyzed, such as the correlation between Hacc/Vacc and the scene.

In order to solve a series of problems in the above algorithms, we propose a scene recognition algorithm based on machine learning using dynamic data. The correlation between the features and the scene and the importance of features for scene recognition are evaluated when extracting the features. Then the features are filtered to ensure that the proposed features can effectively recognize the scene.

3 Methodology

This section describes our proposed scene recognition algorithm. We classify the outdoor scenes into four categories: Highway, Urban Canyon, Suburb, and Avenue. Section 3.1 provides the details of feature extraction and the analysis of features. Section 3.2 describes the process of data preprocessing, and Sect. 3.3 describes the classifiers used for scene recognition.

3.1 Feature Extraction

This section describes the features extracted from the signals received by the GNSS receiver. These features will be fed into the classifier that we design for subsequent scene recognition. The information used to construct the features can be directly obtained from the GNSS receiver. We employ the concept of exhaustive enumeration to generate a comprehensive list of features for scene recognition. Subsequently, we assess the significance of these features using a Chi-square test and identify the top eight features proved to be the most influential for scene recognition. These eight features were then incorporated into our feature vector. Note that the use of information from multiple constellations can improve the recognition results by reducing random outliers. Therefore, in the following feature extraction, all features are computed using full constellation information, including GPS, GLONASS, Galileo, and Beidou. The only exception is the Position Dilution of Precision (PDOP), which uses only GPS constellations.

In various scenes, the visibility of satellites can vary due to different obstructions. Therefore, it is expected that the number of visible satellites and their elevation ranges will differ. In addition, the reception of LOS and NLOS signals varies in different environments. NLOS signals experience attenuation due to obstacles blocking the signal transmission, resulting in a decrease in signal strength and degradation of signal quality. So, it can be assumed that the C/N0 of the received satellite signals will vary across different scenes. The above three features have also been shown to be effective in other scene recognition studies [10,11]. To ensure that they are also applicable to the dataset used, we analyze the number of visible satellites, the distribution of satellite elevation, and C/N0 for each scene.

Figure 1 shows the distributions of the number of visible satellites (NVS) for different scenes. The black solid line in the boxplot represents the median value, while the black diamond point represents the mean value. It can be observed that the distributions of the number of visible satellites vary among the four types of scenes we have defined. Additionally, the mean and median values also differ. However, the distribution of visible satellites in the Urban Canyon and the Avenue is relatively similar. This similarity may be attributed to the presence of not only trees but also short buildings on both sides of the road.

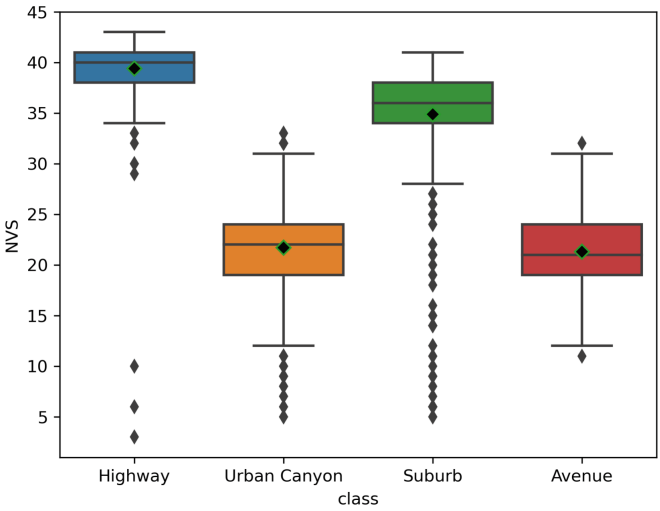

Fig. 1. Distributions of the number of visible satellites (NVS) for different scenes. The solid black line in the boxplot indicates the median, and the black diamond point marks the mean value.

The distributions of satellite elevation for each scene is plotted in Fig. 2. The distributions of satellite elevation for the four types of scenes overlap considerably, but the mean, median, and lower quartile are able to distinguish them.

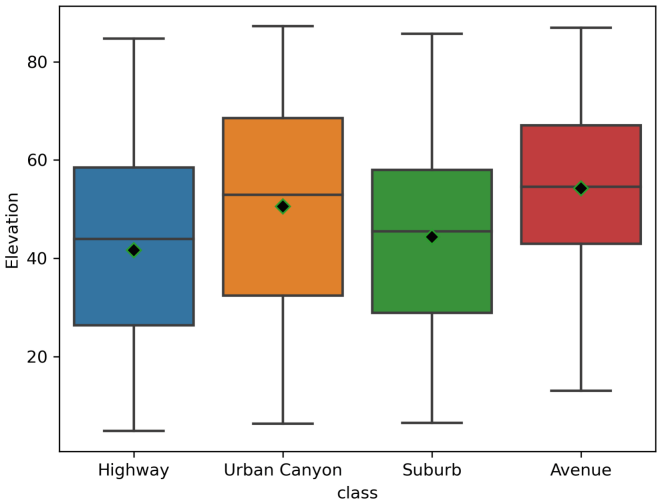

Fig. 2. Distributions of satellite elevation for different scenes. The solid black line in the boxplot indicates the median, and the black diamond point marks the mean value.

The distributions of C/N0 of the received satellite signals for different scenes are plotted in Fig. 3. Compared with the distribution of the number of visible satellites and that of satellite elevation, the distribution of C/N0 has a weaker ability to distinguish four types of scenes. However, it can be observed that the mean C/N0 values of urban canyon and avenue are lower compared with highway and suburb, which is due to the NLOS reception in urban canyon and the shading caused by trees in avenue. In addition, it can be observed that when C/N0 is less than $30dB \cdot Hz$, it can be considered as an outlier for all four types of scenes. Therefore, the threshold for C/N0 is set to $30dB \cdot Hz$ in the subsequent feature extraction.

Based on the NVS, satellite elevation, and C/N0, we propose thirteen features.

The first is the NVS. However, what is different is that we set a C/N0 threshold ζ to consider only the visible satellites with a C/N0 value greater than the threshold. This is to minimize the impact of NLOS signals and highlight the differences in features among the four types of scenes. According to the previous analysis in Fig. 3, we set the threshold ζ to 30 dB/Hz. The feature is denoted as NVS_F as follows:

$$Mask(i) = \begin{cases} 1 & , C/N0_i > \zeta \\ 0 & , otherwise \end{cases} \tag{1}$$

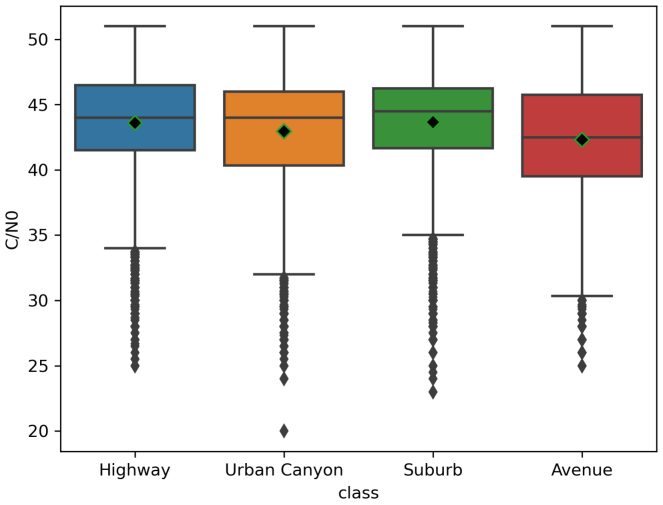

Fig. 3. Distributions of C/N0 of received satellite signals in different scenes. The black solid line in the boxplot indicates the median and the black diamond point marks the mean value.

$$NVS\_F = \sum_{i \in S_v} Mask(i) \tag{2}$$

Next, we hypothesize that the NVS would vary among the four types of scenes within the low (0–30°), medium (30–60°), and high (60–90°) elevation ranges. For example, on a highway, where there are no obstructions such as buildings or trees, there are more visible satellites in the low, medium, and high elevation ranges. In suburban areas, due to the presence of short buildings, there are fewer visible satellites within low elevation range compared to the highway. And in an urban canyon, there are fewer visible satellites in the low and medium elevation ranges due to the obstruction of tall buildings. The NVS is only high in the high elevation range. Therefore, we propose categorizing the elevation into three levels: low (0–30°), medium (30–60°), and high (60–90°). We will use the NVS under the low, medium, and high levels as three features, which will be denoted as NVS_F(0-30), NVS_F(30-60) and NVS_F(60-90) as follows:

$$Mask'(i) = \begin{cases} 1 & ,(C/N0_i > \zeta)\,\&(ELE\_on <= ELE < ELE\_off) \\ 0 & ,otherwise \end{cases} \tag{3}$$

$$NVS\_F(ELE\_on\text{-}ELE\_off) = \sum_{i \in S_v} Mask'(i) \tag{4}$$

where ELE_on represents the lower elevation limit and ELE_off represents the upper elevation limit. For example, when calculating NVS_F(0-30), $ELE\_on = 0$ and $ELE\_off = 30$.

Further, we consider that the ratio of the NVS in the low, medium, and high elevation ranges to the total NVS may be different for the four types of scenes. Therefore, we propose three ratios as the features, which are denoted as NVS_L_prop, NVS_M_prop, and NVS_H_prop, respectively, as follows:

$$NVS\_L\_prop = NVS\_F(0\text{-}30)/NVS\_F \tag{5}$$

$$NVS\_M\_prop = NVS\_F(30\text{-}60)/NVS\_F \tag{6}$$

$$NVS\_H\_prop = NVS\_F(60\text{-}90)/NVS\_F \tag{7}$$

For satellite elevation, previous studies have shown that the average elevation of visible satellites can be utilized as a distinguishing feature for different scenes [11]. We also selecte it as a feature, denoted as ELE_avg as follows:

$$ELE\_avg = \frac{1}{NVS} \sum_{i \in S_v} ELE_i \tag{8}$$

where ELE_i refers to the elevation of the ith satellite in the set S_v.

In addition, we consider that when the receiver is on a highway without any building obstructions, it can receive signals from both high elevation satellites and low elevation satellites. However, the averaging operation may weaken its ability to receive signals from low elevation satellites and may cause confusion with the other three types of scenes. Therefore, we propose sorting the satellites by elevation from low to high and selecting only the first 40% satellites to calculate the average elevation. This feature will help amplify the differences between the four types of scenes. The feature is denoted as ELE_avg(40%).

Similar to the idea of constructing features by the NVS, we construct five features for the C/N0. The first one is the average value of C/N0. The C/N0 is averaged over all the satellite signals that can be received at the current moment. The feature is denoted as $\mu_{C/N0}$ as follows:

$$\mu_{C/N0} = \frac{1}{NVS} \sum_{i \in S_v} C/N0_i \tag{9}$$

where $C/N0_i$ refers to the $C/N0$ of the ith satellite in the set S_v.

Next, we implement the same approach of categorizing the elevation and utilize the average value of C/N0 at low (0-30°), medium (30-60°), and high (60-90°) levels as the three features, which are denoted as $\mu_{C/N0}$(0-30), $\mu_{C/N0}$(30-60), and $\mu_{C/N0}$(60-90), respectively.

Since the analysis of Fig. 3 concludes that the mean value of the $C/N0$ may not effectively distinguish these four types of scenes, it is evident that the minimal value of $C/N0$ in the four types of scenes exhibits a more pronounced difference. So, we propose to use the minimal value of $C/N0$ among all the visible satellites at each moment as a feature for scene recognition. This feature is denoted as $C/N0\_min$ as follows:

$$C/N0\_min = \min_{S_v} C/N0 \tag{10}$$

In addition to the NVS, satellite elevation, and C/N0, the receiver's positioning solution and the geometric distribution of the satellites can also provide scene-relevant information. In an urban canyon with severe occlusion, the positioning error of the receiver is greater than that in a highway or suburban area. The positioning solution of the receiver is closely related to the pseudorange. Therefore, the pseudorange residual, which is the difference between the measured pseudorange and the estimated distance between the receiver and the satellite, can reflect the positioning error. Therefore, the mean pseudorange residual of all visible satellites is also selected as a feature, denoted as Pr_res as:

$$Pr\_res = \frac{1}{NVS} \sum_{i \in S_v} |l_{m,i} - l_{e,i}| \tag{11}$$

where $l_{m,i}$ represents the measured pseudorange of the ith satellite in the S_v set, and $l_{e,i}$ represents the estimated pseudorange of the ith satellite in the S_v set.

PDOP reflects the geometric distribution of visible satellites. Under the same measurement error condition, a smaller PDOP value indicates a more favorable satellite distribution, resulting in a reduced positioning error. Therefore, PDOP can also provide information for scene recognition, so we also select it as a feature.

Till now we have proposed a total of 16 features. It is important to note that some of these features are derived from the NVS, elevation, and C/N0, and they may contain similar information. Thus, there may exist redundant features. Conducting feature selection can reduce the number of features to prevent the curse of dimensionality, decrease training time, improve the model's generalization ability, and mitigate overfitting. Therefore, we aim to evaluate the importance of the proposed 16 features and eliminate unnecessary ones in order to reduce computational costs and enhance the overall efficiency of the scene recognition algorithm. Here the Chi-square test is used to calculate the correlation between a single feature and the classification result. The Chi-square test returns two important values: the Chi-square score($\mathcal{X}^2$) and the P-value.

$$\mathcal{X}^2 = \sum_{i=1}^{4} \frac{(O_i - E_i)^2}{E_i} \tag{12}$$

where O_i represents observed value of feature in the scene i while E_i represents estimated value of feature. The P-value is obtained from the Chi-square distribution table. We propose the hypothesis H_0: the feature is independent of the scene type. When the P-value is less than 0.05, H_0 will be rejected and the feature is considered correlated with the label. Besides, the higher the Chi-square score, the more important the feature is. We calculate the Chi-square score between each feature and labels and then rank them from high to low. The result is given in Fig. 4. While all features have a P-value less than 0.05, we want to select features with a large Chi-square score.

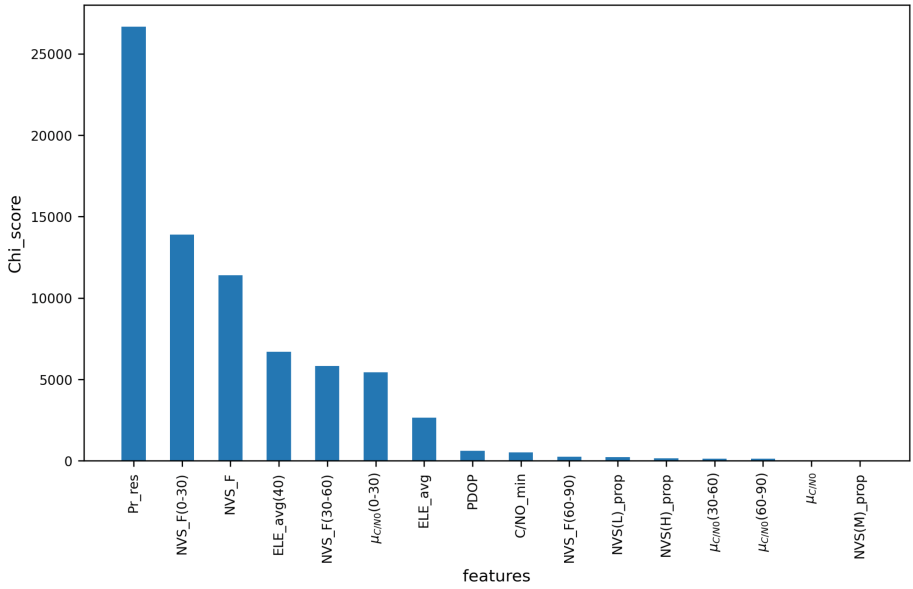

Fig. 4. Importance assessment of 16 features. The vertical coordinate represents the Chi-square score, and each of the 16 bars corresponds to one of the 16 features.

We can observe that the three most important features for scene recognition are Pr_res, NVS_F(0-30) and NVS_F while the least important feature is NVS_M_prop. Compared with ELE_avg, our proposed feature ELE_avg(40%) is more important for scene recognition. The top 9 features make a significant contribution to scene recognition. Eventually, we select eight features as inputs for the subsequent classifier, which are represented as the feature vector v. The reason for excluding ELE_avg is that it duplicates the information carried by ELE_avg(40%).

$$\begin{aligned} v = [&Pr\_res, NVS\_F(\text{0-30}), NVS\_F, NVS\_F(\text{30-60}), \\ &\mu_{C/N0}(\text{0-30}), ELE\_avg(40\%), PDOP, C/N0\_min] \end{aligned} \tag{13}$$

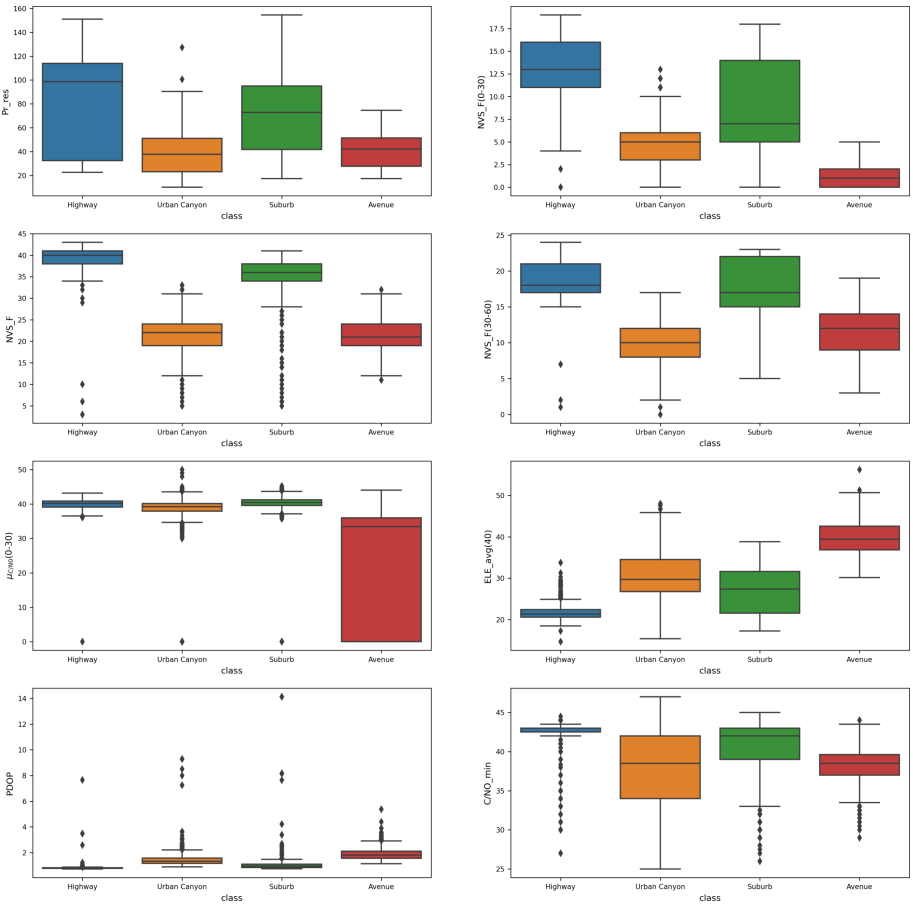

Fig. 5. Distributions of eight features across four types of scenes. The solid black line in the boxplot represents the median, while the black diamond point marks the mean.

The eight subplots in Fig. 5 give the distribution of our proposed eight-dimensional features across four types of scenes. The distributions show that there are significant differences across different scenes, demonstrating that the 8 selected features can be utilized to train the model in distinguishing four types of scenes.

3.2 Pre-processing

Directly inputting the features into the classifier may not yield the desired result, as the scales of the features may not be uniform. In order to accelerate the convergence of the model and improve the classification accuracy, we standardize the feature vectors using Min-Max Normalization, which ensures that the value of each feature falls within the interval [0,1].

$$X' = \frac{X - X_{min}}{X_{max} - X_{min}} \tag{14}$$

3.3 Classifier

In this study, we chose to use SVM and RF as classifiers and fuse their outputs for scene recognition. The reasons are as follows: the SVM is a binary classification model that can be extended to multi-classification using the One-Versus-One (OVO) strategy, One-Versus-Rest (OVR) strategy, or Error-Correcting Output Codes (ECOC). Since SVM tries to find the optimal hyperplane that maximizes the distance from the support vector to the hyperplane between two classes, SVM exhibits stability with constant parameters, and it has a clear advantage in Few-shot learning. However, implementing SVM for large-scale training samples can be challenging. RF, on the other hand, is a multi-classification model that integrates multiple decision trees using the Bagging technique. It is designed to handle high-dimensional data and exhibits strong generalization ability and robustness. However, it is random and unstable during each training session. Since SVM and RF have different principles and are complementary, fusing their results can reduce bias, variance, and improve accuracy.

We fuse the prediction results of SVM and RF. The trained SVM model and RF model will calculate the probability of the sample belonging to each scene when making predictions on the test set. We sum the four probabilities predicted by SVM and the four probabilities predicted by RF correspondingly. Finally, we select the class with the highest probability as the final recognition result.

4 Experiments and Results

This section introduces the experiments and results. Section 4.1 introduces the dataset used in the simulations. Section 4.2 describes the settings and flow of the experiments. Section 4.3 shows the results and gives the corresponding analyses.

4.1 Dataset

In this study, data collection is carried out in Shanghai. The GNSS data is obtained by an onboard NovAtel high-precision GNSS/INS device (PP7+SPAN-ISA-100C), with a sampling frequency of 1 Hz. We label the scene using the fisheye camera data collected synchronously with GNSS. The scenes are divided into four types: Highway, Urban Canyon, Suburb, and Avenue as shown in Fig. 6. The Highway resembles an open environment with almost no obstructions. The Urban Canyon is located in a busy downtown area, surrounded by tall buildings that block many of the GNSS signals. Some of the tall buildings are covered with glass, which can easily cause multipath interference. The Suburb is more open, surrounded by a few short buildings and trees, and a small number of low elevation GNSS signals may be blocked. The sky in the Avenue is almost completely covered by foliage, and there are short buildings on both sides of the road. As a result, some low and medium elevation GNSS signals may be blocked.

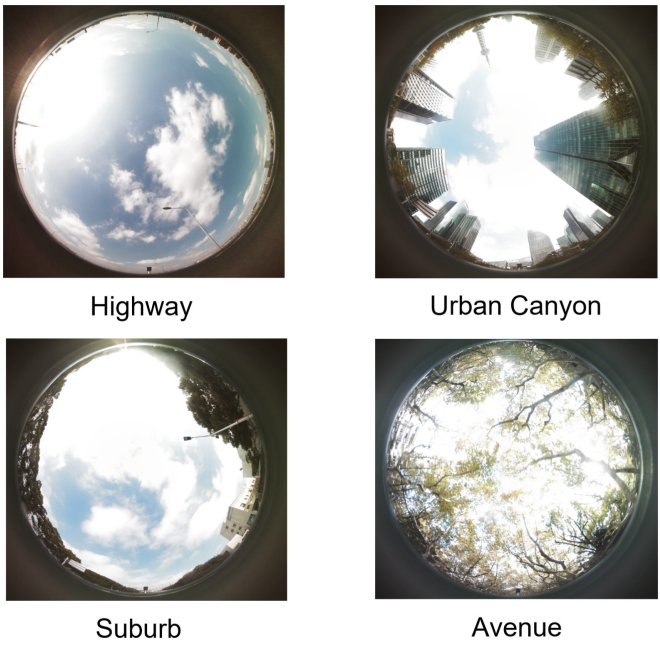

Fig. 6. Examples of the four types of scenes

Figure. 7 presents the route map for data collection in different scenes. The orange route is the total route of data collection, from which we extracted certain sections as the data for scene recognition. The Highway sections are shown in blue; the Urban Canyon sections are shown in red, the Suburb sections are shown in pink, and the Avenue sections are shown in green. The number of sample points in each scene is given in Table 1.

Table 1. The number of sample points for each type of scene

class	Highway	Urban Canyon	Suburb	Avenue
samples	1358	1623	1414	962
percentage	25.3%	30.3%	26.4%	18.0%

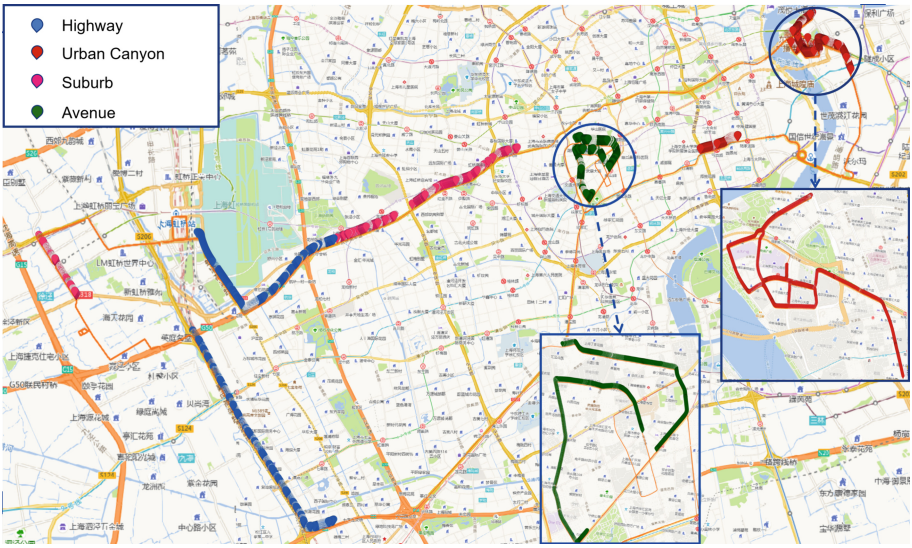

Fig. 7. Data collection routes

4.2 Experiment

Before the experiment, we first adjust the parameters of SVM and RF. Since we only have eight features, which is much less than the number of samples, and considering multiple tests, it is finally determined that the Gaussian kernel is the most appropriate. For the SVM model with a Gaussian kernel, the parameters C and γ are crucial. C is the penalty coefficient, which determines the model's tolerance to errors. γ is a parameter that affects the range of the Gaussian function associated with each support vector, thereby influencing the generalization performance of the corresponding model. We determine that the trained model performs best when $C = 9$, $\gamma = 4$ using the cross-validation method. The parameter optimization for the RF can be categorized into two types: parameter optimization for the Random Forest framework and parameter optimization for the Decision Tree. For the Random Forest framework, we tune the number of Decision Trees ($n\_estimators$). Additionally, we tune the parameters of the Decision Tree, including the maximum number of features considered in constructing the Decision Tree ($max\_features$), and the maximum depth of the Decision Tree (max_depth). Also, through cross-validation, it is ultimately determined that the model yields the best results when $n\_estimator = 70$, $max\_features = 6$, and $max\_depth = 20$.

Since Deep Learning has achieved extraordinary results these days, we also conduct comparison experiments with two modern Deep Learning-based approaches, Multi-Layer Perceptron (MLP) and Long Short Term Memory (LSTM). The MLP we build has 4 hidden layers, each output uses the ReLU function as the activation function, and the output layer uses the softmax func-

tion to calculate the recognition probability. The LSTM we establish has 2 layers with a hidden size of 32.

We first compare the scene recognition results of SVM based on the One-Versus-One strategy (SVM-OVO-v), SVM based on OVR strategy (SVM-OVR-v), and SVM based on ECOC (SVM-ECOC-v), and select the classifier with the best results for the subsequent experiments. The predicted probabilities of the SVM classifier are then fused with the predicted probabilities of RF using the result fusion method proposed in Sect. 3.3. In order to demonstrate the effectiveness and high efficiency of our proposed 8-dimensional feature vector v for scene recognition and the superiority of the SVM, RF result fusion method (SRRF-v), we conduct scene recognition experiments using the following algorithms: SVM combined with the proposed 8-dimensional feature vector v (SVM-OVO-v); RF combined with the feature vector v (RF-v); MLP combined with the feature vector v; LSTM combined with the feature vector v; SRRF combined with the feature vector v (SRRF-v); SRRF combined with all 16-dimensional vectors (SRRF-all); SRRF combined with the feature vector r proposed in [11].

$$\begin{aligned} r = [&NVS_{gps}, NVS_{gal}, NVS\_filtered_{gps}, NVS\_filtered_{gal}, H_{acc}, \\ &V_{acc}, PDOP, \mu_{C/N0_{gps}}, \mu_{C/N0_{gal}}, elev_{gps}, elev_{gal}, res_{gps}, res_{gal}] \end{aligned} \tag{15}$$

Finally, we divided the train set and test set in a 1:2 ratio for scene recognition experiments. We present the confusion matrices of the recognition results of SVM-v, RF-v, and our proposed SRRF-v algorithm. We also calculate the recognition accuracy for each type of scene to further prove the superiority of SRRF-v.

To better analyze the recognition results, we use the following evaluation metrics: Accuracy, Precision, Recall, and F1-score, which are defined as follows, respectively:

$$Accuracy = \frac{True\,positive + True\,negative}{True\,positive + True\,negative + False\,positive + False\,negative} \tag{16}$$

$$Precision = \frac{True\ positive}{True\ positive + False\ positive} \tag{17}$$

$$Recall = \frac{True\ positive}{True\ positive + False\ negative} \tag{18}$$

$$F1\text{-}score = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{19}$$

4.3 Results

Table 2 displays the performance of the three SVM-based algorithms in the five-fold cross-validation method. It can be observed that the average recognition accuracy of all three SVM-based classifiers exceeds 92%. Comparing the three SVM-based multi-classifiers, it can be seen that SVM-OVO yields the best result,

while SVM-ECOC performs the worst. The accuracy, precision, recall, and F1-score of SVM-OVO are all higher than those of SVM-OVR and SVM-ECOC. However, overall, there is not much difference in the performances of the three classifiers. Rather surprisingly, the ECOC encoding (4 bits) does not improve the accuracy of classification.

Table 2. Performance of SVM-based algorithms in five-fold cross-validation experiment (A: Accuracy; P: Precision; R: Recall; F: F1-score)

Classifier	Performance			
SVM-OVO-v	A:0.9287	P:0.9244	R:0.9309	F:0.9269
SVM-OVR-v	A:0.9279	P:0.9235	R:0.9309	F:0.9262
SVM-ECOC-v	A:0.9257	P:0.9215	R:0.9287	F:0.9240

Table 3 presents the performance of seven algorithms: SVM-v, RF-v, MLP-v, LSTM-v, SRRF-v, SRRF-all, and SRRF-r in the five-fold cross-validation experiment. The recognition accuracy of SVM-v is 92.87%, RF-v achieves 93.47%, MLP-v achieves 91.71%, LSTM-v achieves 93.12% and our proposed SRRF-v achieves 95.05%. It can be observed that SRRF-v is higher than the rest of the algorithms in all of the evaluation metrics.

In addition, the recognition accuracy, precision, recall, and F1-score of SRRF-v are nearly identical to those of SRRF-all. This proves that our filtered eight-dimensional feature vector v is as effective as using the entire 16-dimensional feature vector. This proves that feature selection eliminates redundant features. Besides, by comparing the training time and prediction time of different algorithms, we can draw a conclusion that Deep Learning methods consume much time in training. SRRF-v consumes 0.236 s on training and 0.093 s on prediction, which is significantly less than SRRF-all, proving our feature selection method

Table 3. Performance of each algorithm in five-fold cross-validation experiment (A: Accuracy; P: Precision; R: Recall; F: F1-score; TT: Training Time; PT: Prediction Time)

Classifier	Performance					
SVM-OVO-v	A:0.9287	P:0.9244	R:0.9309	F:0.9269	TT:0.274	PT:0.142
RF-v	A:0.9347	P:0.9320	R:0.9316	F:0.9317	TT:0.021	PT:0.002
MLP-v	A:0.9171	P:0.9114	R:0.9181	F:0.9147	TT:8.133	PT:0.018
LSTM-v	A:0.9312	P:0.9259	R:0.9328	F:0.9293	TT:101.725	PT:0.252
SRRF-v	A:0.9505	P:0.9487	R:0.9483	F:0.9484	TT:0.236	PT:0.093
SRRF-all	A:0.9510	P:0.9493	R:0.9495	F:0.9489	TT:0.295	PT:0.151
SRRF-r	A:0.9352	P:0.9330	R:0.9329	F:0.9327	TT:0.255	PT:0.110

improves the efficiency of the SRRF. The recognition accuracy, precision, recall, and F1-score of SRRF-v are higher than those of SRRF-r. This demonstrates that our proposed eight-dimensional feature vector is more effective in scene recognition compared to the 16-dimensional feature vector r proposed in [11].

To further compare the performances of SVM-OVO-v, RF-v, and SRRF-v in scene recognition, the confusion matrix and recognition accuracy of each method is presented in Table 4, Table 5, and Table 6, respectively. These tables provide a detailed analysis of the performances of SVM-OVO-v, RF-v, and SRRF-v in recognizing the four types of scenes.

According to Table 4, the total recognition accuracy of SVM-OVO-v is 92.61%, and its recognition accuracy exceeds 90% in all four types of scenes. Its recognition accuracy for both Highway and Avenue exceeds 95% but is relatively low for Urban Canyon and Suburb.

Table 4. Confusion matrix of SVM-OVO-v

True class	Predicted class					
	Highway	Urban canyon	Suburb	Avenue	T	F
Highway	871	1	34	0	96.14%	3.86%
Urban canyon	0	974	1	108	90.00%	10.00%
Suburb	53	27	860	3	90.88%	9.12%
Avenue	0	24	2	616	95.95%	4.05%
Overall Accuracy	92.61%					

Table 5 shows that RF-v achieves a total recognition accuracy of 93.62%. Additionally, its recognition accuracies for the four types of scenes are over 91%, with its accuracy in recognizing Highways exceeding 95%. Compared to SVM-OVO-v, RF-v has more average recognition accuracies for the four types of scenes, but it shows a significantly lower recognition accuracy specifically for Avenue.

Table 5. Confusion matrix of RF-v

True class	Predicted class					
	Highway	Urban canyon	Suburb	Avenue	T	F
Highway	874	1	31	0	96.47%	3.53%
Urban canyon	0	1007	11	65	92.98%	7.02%
Suburb	41	15	877	10	93.00%	7.00%
Avenue	0	50	4	588	91.59%	8.41%
Overall Accuracy	93.62%					

Table 6 shows that the total recognition accuracy of SRRF-v reaches 94.38%, which is higher than that of SVM-OVO-v and RF-v. In addition, its recognition accuracy for each type of scene exceeds 93%. Compared with SVM-v and RF-v, the recognition accuracies of SRRF-v for the four types of scenes are not only more average, but also higher. This demonstrates that SRRF-v yields better results than individual SVM and RF model.

Table 6. Confusion matrix of SRRF-v

True class	Predicted class					
	Highway	Urban canyon	Suburb	Avenue	T	F
Highway	872	1	33	0	96.25%	3.75%
Urban canyon	0	1006	3	74	92.89%	7.11%
Suburb	31	21	889	2	94.27%	5.73%
Avenue	0	34	4	606	94.10%	5.90%
Overall Accuracy	94.38%					

The main errors are the misjudgment of Highway and Suburb, as well as the misjudgment of Urban Canyon and Avenue. This aligns with the significant similarity in features between Highway and Suburb, as well as between Urban Canyon and Avenue, which we analyze in the previous feature extraction part.

5 Conclusion

In this study, a GNSS-based scene recognition method, SRRF-v, is proposed. The SRRF-v method provides a framework for feature extraction and evaluation, as well as a strategy for fusing the outputs of SVM and RF to enhance recognition results. Furthermore, the SRRF-v method can be applied to dynamic data. The method utilizes GNSS signals to propose 16 features through exhaustive enumeration. It then analyzes the validity of these features using the Chi-square test and ultimately determines an 8-dimensional feature vector. The 8-dimensional feature vector is input into SVM and RF to train two models that can be used for scene recognition. Finally, the output probabilities are fused to determine the type of scene. From the recognition results, the five-fold cross-validation accuracy of SRRF-v reaches 95.05%. The overall recognition accuracy reaches 94.38% when the training and testing sets are divided by 1:2. Furthermore, the recognition accuracies for Highway, Urban Canyon, Suburb, and Avenue all exceed 93%. The main errors are the confusion between Highway and Suburb, as well as the misjudgment of Urban Canyon and Avenue. From an occlusion perspective, Highway is similar to Suburb, and Urban Canyon is similar to Avenue. These similarities may be the reason for the misjudgment. We will try to find solutions for this problem in future research.

Acknowledgement. This work is supported by the National Natural Science Foundation of China (NFSC) under Grant Nos. 61971278 and Grant 62231010.

References

1. Zaidi, A., Suddle, M.: Global navigation satellite systems: a survey. In: 2006 International Conference on Advances in Space Technologies, pp. 84–87 (2006). https://doi.org/10.1109/ICAST.2006.313803
2. Feriol, F., Vivet, D., Watanabe, Y.: A review of environmental context detection for navigation based on multiple sensors. Sensors **20**(16) (2020). https://doi.org/10.3390/s20164532, https://www.mdpi.com/1424-8220/20/16/4532
3. Groves, P.D.: Shadow matching: a new GNSS positioning technique for urban canyons. J. Navig. **64**(3), 417–430 (2011). https://api.semanticscholar.org/CorpusID:18769611
4. Li, S., Li, X., Wang, H., Zhou, Y., Shen, Z.: Multi-GNSS PPP/ins/vision/lidar tightly integrated system for precise navigation in urban environments. Inf. Fusion **90**, 218–232 (2023)
5. Ayyalasomayajula, R., Arun, A., Wu, C., Sharma, S., Bharadia, D.: Deep learning based wireless localization for indoor navigation. In: MobiCom 2020: The 26th Annual International Conference on Mobile Computing and Networking, pp. 1–14 (2020)
6. Macgougan, G., et al.: Performance analysis of a stand-alone high-sensitivity receiver. GPS Solutions **6**(3), 179–195 (2002)
7. Tang, I., Breckon, T.P.: Automatic road environment classification. IEEE Trans. Intell. Transp. Syst. **12**(2), 476–484 (2011)
8. Kumari, S., Jha, R.R., Bhavsar, A., Nigam, A.: Indoor–outdoor scene classification with residual convolutional neural network. In: Chaudhuri, B.B., Nakagawa, M., Khanna, P., Kumar, S. (eds.) Proceedings of 3rd International Conference on Computer Vision and Image Processing. AISC, vol. 1024, pp. 325–337. Springer, Singapore (2020). https://doi.org/10.1007/978-981-32-9291-8_26
9. Gao, H., Groves, P.D.: Environmental context detection for adaptive navigation using GNSs measurements from a smartphone. Navig.-J. Inst. Navig. **65**(1), 99–116 (2018)
10. Xia, Y., Pan, S., Gao, W., Yu, B., Gan, X., Zhao, Y., Zhao, Q.: Recurrent neural network based scenario recognition with multi-constellation GNSS measurements on a smartphone. Measurement **153**, 107420 (2020)
11. Feriol, F., Watanabe, Y., Vivet, D.: GNSS-based environmental context detection for navigation. In: 2022 IEEE Intelligent Vehicles Symposium (IV), pp. 888–894 (2022). https://doi.org/10.1109/IV51971.2022.9827023