# Filtering Keypoints with ORB Features

Thaisa Aline Correia Garcia [1], Antonio Maria Garcia Tommaselli [1], Letícia Ferrrari Castanheiro [1], Mariana Batista Campos [2]

1 Department of Cartography, Faculty of Sciences and Technology, São Paulo State University (UNESP), São Paulo 19060-900, Brazil - (thaisa.correia, a.tommaselli, leticia.ferrari)@unesp.br
2 Department of Remote Sensing and Photogrammetry, Finnish Geospatial Research Institute (FGI), National Land Survey of Finland, Espoo, Finland - mariana.campos@nls.fi

**Keywords:** ORB features, keypoints, machine learning, fisheye lenses.

**Abstract**

Keypoint detectors and descriptors are essential for identifying points and their correspondences in overlapping images, being fundamental inputs for many subsequent processes, including Pose Estimation, Visual Odometry, vSLAM, Object Detection, Object Tracking, Augmented Reality, Image Mosaicking, and Panorama Stitching. Techniques like SIFT, SURF, KAZE, and ORB aim to identify repeatable, distinctive, efficient, and local features. Despite their robustness, some keypoints, especially those detected in fisheye cameras, do not contribute to the solution, and may introduce outliers or errors. Fisheye cameras capture a broader view, leading to more keypoints at infinity and potential errors. Filtering these keypoints is important to maintain consistent input observations. Various methods, including gradient-based sky region detection, adaptive algorithms, and K-means clustering, addressed this issue. Semantic segmentation could be an effective alternative, but it requires extensive computational resources. Machine learning provides a more flexible alternative, processing large data volumes with moderate computational power and enhancing solution robustness by filtering non-contributing keypoints already detected in these vision-based approaches. In this paper we present and assess a machine learning model to classify keypoints as sky or non-sky, achieving an accuracy of 82.1%.

## 1. Introduction

Detectors and descriptors of interest points enable the identification of keypoints and correspondences between overlapping images. This is used in many vision-based approaches, including Pose Estimation, Visual Odometry, Visual Simultaneous Localization and Mapping (vSLAM), Object Detection, Object Tracking, Augmented Reality, Image Mosaicking, and Panorama Stitching. There are many keypoints detectors and feature descriptors, such as SIFT (Scale-Invariant Feature Transform) (Lowe, 2004), SURF (Speeded-Up Robust Features) (Bay et al., 2008), KAZE (Alcantarilla et al., 2012), and ORB (Oriented FAST and Rotated BRIEF) (Rublee et al., 2011), which identify features in images that have the following properties: repeatability, distinctiveness, efficiency, and locality (Gao and Zhang, 2021; Tareen and Saleem, 2018). Despite the robustness of those methods, some detected keypoints in the scene do not contribute to the solution and can even introduce outliers or gross errors into the processes. The challenge of obtaining consistent point observations increases when working with fisheye images, which can capture a broader view and consequently identify a larger number of keypoints compared to perspective cameras. In particular, the wide Field of View (FoV) can increase the mapping of sky areas and distant objects in outdoor scenarios (Zhao et al., 2022) and result in a higher number of keypoints being generated at infinity. These keypoints generate parallel projecting rays (e.g., points detected at cloud edges) that can lead to significant errors in subsequent processing steps, such as pose estimation. Therefore, it is desirable to previously filter out keypoints to maintain a consistent network of input observations.

There are various methodologies in the literature aimed at identifying and filtering areas in images that can be discarded, such as detecting the sky and other elements in the image. For example, Shen and Wang (2013) proposed a sky region detection method based on gradient information and energy function optimization. Nice et al. (2020) developed a methodology to refine edge results in sky region detection using an adaptive algorithm and machine learning for fisheye images. Additionally, Kato et al. (2016) segmented the sky using K-means clustering. Moreover, semantic segmentation can be incorporated into the process to detect, segment, and filter areas of the image that are not of interest or may introduce errors in the map (Shao et al., 2020; Yu et al., 2018). This approach is also recommended for identifying dynamic objects in the scene, such as cars, motorcycles, and pedestrians, which can negatively impact the solution estimation (Gao and Zhang, 2021).

Although semantic segmentation has shown promising results, the networks are built on deep learning architectures, requiring extensively trained models that demand substantial computational and storage resources, as well as complex sampling processes (Shao et al., 2020; Wang et al., 2021; Yu et al., 2018). Additionally, since many methodologies in Computer Vision and Photogrammetry rely on keypoints to find corresponding points between images, it would be suitable to define a fast and efficient technique capable of validating this type of data. This would ensure that filtering does not significantly impact the computational performance of the solution, thus improving the effectiveness of the developed approach, using the features already extracted in the process. In this work, we hypothesize that keypoint filtering using a machine learning algorithm trained to identify undesirable points, based on pre-extracted feature descriptors, can be a viable alternative to reduce the computational demands of outlier filtering in navigation applications, especially using fisheye images.

Machine learning is also a field of artificial intelligence capable of processing large volumes of structured data to analyse patterns and make predictions. A significant advantage of machine learning is its ability to make low-complexity decisions, requiring only a moderate amount of computer processing power (Braga-Neto, 2020; Géron, 2019). Machine learning is flexible and allows for the manipulation of keypoints, enhancing solution robustness by removing non-contributing elements from the scene (Khan and Al-Habsi, 2020). Another factor to consider is the need of efficiency,

when incorporating machine learning techniques into embedded platforms with limited and low-cost computational resources.

To validate the hypothesis, here we trained the machine learning algorithm Random Forest (RF) and assessed the resulting model for binary classification (sky, non-sky keypoint) using ORB features to remove sky points in images from fisheye lens camera. This aims to eliminate gross errors in input observations of subsequent process that requires feature matching (e.g. pose estimation) by removing these points in the sky. Avoiding the processing of non-contributory data, this approach can optimize computational efficiency and enhance the overall performance of the solution. It will avoid outliers while enabling the use of lighter and faster prediction models suitable for implementation on compact mobile platforms. Experimental assessments were performed using ORB features automatically detected in fisheye image dataset acquired in an outdoor scenario.

## 2. Materials and Methods

### 2.1 ORB Features

Features are generally detected in corners, blobs, edges, junctions, and lines. The detected features are described by unique patterns based on their neighbouring pixels. This process, known as feature description, assigns a unique identity to each feature. Some techniques employ both feature detectors and feature description algorithms, while others operate independently. However, these independent feature detectors can be paired with many appropriate feature descriptors. Using these descriptors, matching algorithms can be applied to identify keypoints with similar features, as illustrated in Figure 1 (Forstner, 1986; Tareen and Saleem, 2018).

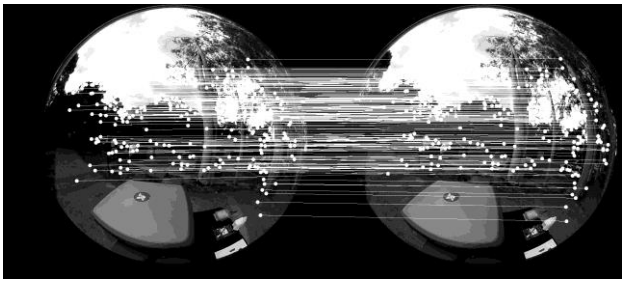


Figure 1. Keypoints and correspondences between two overlapping images.

In the literature, several keypoint detectors and descriptors have been proposed. Tareen and Saleem (2018) conducted a comparative study of the prominent detectors and descriptors currently in use. They concluded that while SIFT is the most accurate and robust, it has high computational cost. They considered ORB to be the most efficient method, balancing robustness, and computational performance, and thus, it stands out as a high-performance alternative. This conclusion motivated the use of ORB in this work.

ORB is a fusion of the FAST (Features from Accelerated Segment Test) detector and a fast binary descriptor based on BRIEF (Binary Robust Independent Elementary Feature). FAST identifies interest points in images by detecting local grayscale changes. Unlike other interest points detection algorithms, FAST only requires comparing the brightness levels of neighbouring pixels, making the approach faster. Originally, FAST did not include directional information, so ORB employs a scaled pyramid (downsampling images at some levels to achieve different resolutions) and the intensity centroid method to address scale and rotation information. BRIEF describes the features of the surrounding area around the points detected in the previous step. It is a binary descriptor, typically a 128–512-bit string, encoding the size relationship between two random pixels near the keypoint. BRIEF efficiently compares these randomly selected points. However, the original BRIEF lacks rotation invariance, making it more susceptible to errors when the image is rotated. With the previously calculated directional information, ORB can use BRIEF to create features that perform well under translation, rotation, and scaling. Meanwhile, the combination of FAST and BRIEF is highly efficient, making ORB features popular in real-time scenarios (Gao and Zhang, 2021; Rublee et al., 2011; Tareen and Saleem, 2018).

### 2.2 Machine Learning: Random Forest algorithm

Machine learning (ML) algorithms have been widely explored across countless applications, particularly those involving large volumes of data and complex datasets (structured objects and their attributes) (Braga-Neto, 2020; Khan and Al-Habsi, 2020). ML algorithms are employed to train models based on parameters that link input and output data through a learning process that is continuously refined using the dataset. This learning process can be supervised, unsupervised, semi-supervised, or reinforcement-based (Shobha and Rangaswamy, 2018). Supervised learning makes predictions from unknown data by mapping the relationship between inputs and known outputs. It uses classification algorithms when the label value is discrete and regression techniques when the label value is continuous to develop these predictive models (Géron, 2019; Shobha and Rangaswamy, 2018). A significant advantage of ML is its ability to make low-complexity decisions, requiring a moderate amount of computer processing power, and it can make predictions based on a dataset that is not excessively voluminous or complex.

Given the large number of available ML algorithms, a preliminary analysis and assessment were conducted to select a suitable model based on the nature of our problem. Considering our dataset, which includes multiple features for the same keypoint and potential overlaps in sky regions (e.g., intensity changes between surfaces), decision tree supervised learning algorithm, such as RF, are often recommended.

A decision tree is a kind of supervised learning algorithm used to solve classification and regression problems. It is based on a multistage or hierarchical decision-making method, breaking down the decision into several more straightforward and more interpretable decisions in a series of binary nodes. RF, an ensemble of different decision trees, can also be used to solve classification and regression problems. RF incorporates randomness in the selection of attributes (Breiman, 2001; Shobha and Rangaswamy, 2018). In RF approach, multiple trees are trained using different subsets of attributes and data, ensuring that each tree creates a different model. These models depend on values from a randomly sampled vector, independently and identically distributed for all trees. The final output is the average of the predictions from all trees. Due to the randomness in data samples and variables considered at each decision node, RF often reduces the problem of overfitting (Breiman, 2001). Moreover, combining multiple decision trees

results in more accurate and stable predictions (Géron, 2019; Khan and Al-Habsi, 2020).

To validate this assumption and the effectiveness of a decision tree supervised learning algorithm within our specific context, we conducted preliminary tests using RF. These tests involved training RF with a dataset and analysing the metrics achieved on the validation sets. Our objective was to compare the performance of RF with Support Vector Machine (SVM), a widely used machine learning methodology for classification (Géron, 2019; Khan and Al-Habsi, 2020; Shobha and Rangaswamy, 2018). The results indicated that RF outperformed SVM in our context, demonstrating superior performance on the validation set. Therefore, we selected RF to train our sky keypoints detection model in fisheye images, leveraging its ability to make accurate discrete predictions, handle overlapping features, maintaining robustness against overfitting.

### 2.3 Training Dataset

For training purposes, we selected the WoodScape dataset, distinguished by including fisheye image data and a comprehensive range of annotation types. Moreover, designed specifically for research in autonomous driving and computer vision, the dataset features outdoor images that include a wide variety of scenes and environments commonly encountered in urban settings, within complex, real-world environments (Yogamani et al., 2019). WoodScape comprises four surround-view cameras and involves nine image processing steps, including segmentation, depth estimation, and 3D bounding box detection, among others. Semantic annotation of 40 classes at the instance level is provided for over 10,000 images, also containing the sky, as indicated in Figure 2.

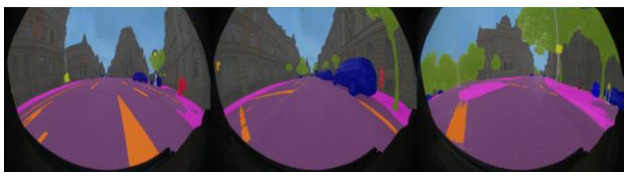


Figure 2. Semantic segmentation from WoordScape (Yogamani et al., 2019)

Outdoor images with a wider FoV, such as those collected by fisheye cameras, have an increased probability of detecting keypoints in areas that are not useful for vision-based approaches. Keypoints at the infinite, such as sky keypoints, or on moving objects, can negatively affect the estimation based on the mathematical model used. Considering that a large part of the outdoor image may contain these sky keypoints, we developed a model to determine whether to use a keypoint, by inferring if it belongs to the sky area (Braga-Neto, 2020; Khan and Al-Habsi, 2020). We used the fisheye images from the WoodScape dataset, along with semantic boundaries, to generate masks that separate the areas in the images that belong to the sky from those that do not. We considered two classes: (1) keypoints in the sky and (0) keypoints out of the sky area, as illustrated in Figure 3.

(a)

(b)

Figure 3. WoodScape dataset example: (a) RGB fisheye image; (b) Mask obtained through semantic segmentation of the sky class.

### 2.4 Machine Learning Training Model

The workflow for training the sky-point classification model using ORB features is depicted in Figure 4. First, we used the WoodScape dataset, which contains outdoor images from fisheye lens cameras and their corresponding semantic data, to generate a mask. This mask was applied to the images to label and extract the pixels that belong to the sky and those out of this area. Next, we detected the keypoints and extracted their features from the areas identified as sky and non-sky in the previous step. We then saved the keypoints and all related information to a structured file, including their position, the image in which they were detected, their descriptor, and their binary class: 0 (non-sky) or 1 (sky).

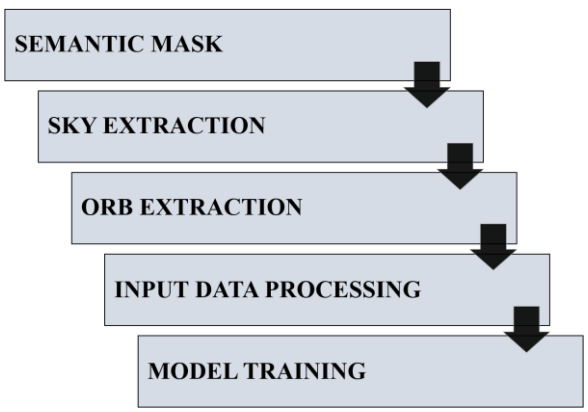



Figure 4. Machine Learning Training Model workflow for binary classification of keypoints as sky and non-sky points.

For training the proposed RF model, we used the keypoints and their respective information saved in a file, applying a data processing step to structure the input data for training. During the ORB extraction stage, we obtained a set of 9,000,000 samples, each one consisting of a keypoint and its 128-dimensional descriptor. The majority of samples were from keypoints not belonging to the sky, so we opted to apply a data balancing step. Balancing is an important step to prevent bias towards a dominant class with more samples in training, ensuring a fair distribution among classes (Géron, 2019). In our context, we performed undersampling by randomly selecting 1,000,000 samples of sky keypoints and 1,000,000 samples of non-sky keypoints. Random selection aims to capture the most representative samples from the dataset possible. This approach ensures a more equitable distribution of data for training our model. Among the selected samples, we allocated 80% for model training and 20% for validation. After completing the training, we obtained the trained RF model for prediction, specifically to classify the keypoints as either belonging to the sky or not, as illustrated in Figure 5. An arbitrary image is processed for keypoint detection and ORB feature extraction. These extracted features are then classified by the model. Based on this decision, we filter out the keypoints that belong to the sky, which can be discarded for the subsequent processing steps.

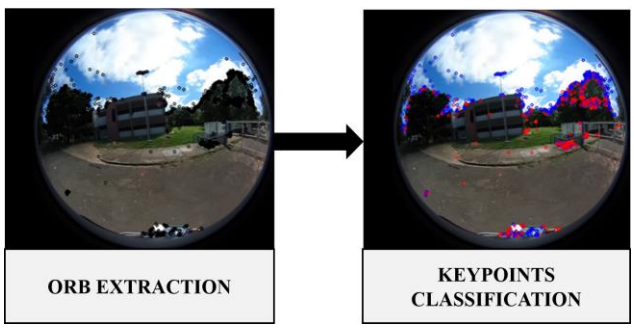



Figure 5. Prediction flow of the sky classification.

## 3. Experiments and Results

### 3.1 Filter detection performance

Initially, we applied the trained machine learning model to the validation dataset, which consists of 20% of the data extracted from the WoodScape dataset. The validation dataset was separated to ensure it accurately represents the diversity and complexity of the entire dataset, providing a robust set for evaluating the model's performance. Our model, designed for sky keypoint classification based on binary decision-making, demonstrated an accuracy rate of 82.1% on this validation dataset. We employed the accuracy metric, as depicted in Equation 1, to assess the 20% of samples designated for model validation (Géron, 2019). Here, TP stands for true positives, TN for true negatives, FP for false positives, and FN for false negatives.

$$accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

The RF model accuracy achieved on the validation dataset, validates its effectiveness and robustness in distinguishing sky features from other elements within the images. To achieve this level of accuracy, we utilized a preprocessing step in the model training, known as randomized hyperparameter tuning (Géron, 2019). This method randomly combines different hyperparameters within a specified range and selects the model that achieves the best results based on the tested hyperparameters, aiming to optimize and enhance its predictive capabilities. The WoodScape dataset itself is known for its challenging and diverse set of fisheye images, which include varying weather conditions, lighting scenarios, and outdoor contexts. Successfully achieving an 82.1% accuracy on such a dataset reinforces the robustness of our model, even with a reduced number of samples. One of our key considerations was to develop a lightweight and simple model to ensure its feasibility for use in complex outdoor scenarios that require portable platforms, such as forest and narrow corridors, which typically involve low-cost sensors and embedded platforms with limited resources. In some situations, requiring real-time response, the efficiency of the processes is essential.

### 3.2 Test on an independent dataset

We applied the model to our dataset consisting of fisheye images captured with the Ricoh Theta S dual-fisheye hyperhemispherical camera. Examples of Ricoh Theta S images are shown in Figures 6(a) and 6(b). Figure 6(b) depicts the predictions on an image from this dataset, with keypoints classified as sky shown in blue, while non-sky keypoints are represented in red. We followed the procedure previously described and illustrated in Figure 5. From the fisheye image, we extracted the ORB features and then submitted the keypoints and their features to prediction using our RF model.

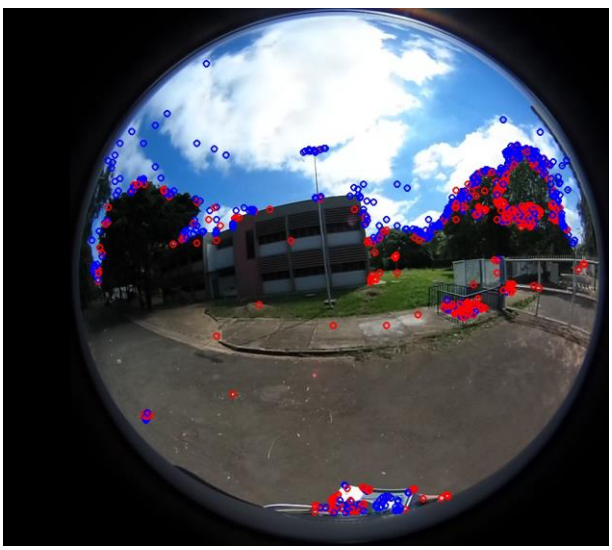
It was possible to verify that false positives or false negatives still occur, but the model generally labelled most of the sky keypoints in the image from a dataset different from the one used to train the model. It is important to note that the trained model is of reduced size, making it easy to store, and it also demonstrates good computational performance in response time during the prediction stage. The proposed filtering approach can contribute to many vision-based applications, as a preprocessing step. Regarding vSLAM solution, the inclusion of a point filtering step can contribute to a more robust and stable solution. Further improvements of the model are possible by adding new samples, both from the Woodscape dataset and from other cameras, including the Ricoh Theta S itself.

## 4. Conclusion

Based on our feasibility study results, we concluded that ML offers the flexibility to develop simple, practical approaches that can be used to label keypoints based on their already extracted features and which can be applied to remove outliers for different fisheye image applications. To validate our hyphothesis, in this work, we focused on developing a ML model to support vision-based applications that rely on keypoint detection and feature extraction. Our results showed that keypoints (particularly keypoints in the sky in outdoor scenarios) that could significantly degrade further solutions, such as pose estimation, can be efficiently removed by including the proposed filtering step based on ML models. The proposed RF model for classifying sky keypoints achieved an

accuracy of 82.1%, demonstrating its effectiveness in detecting the majority of sky points in an independent set of images captured by the Ricoh Theta S fisheye camera. Despite occasional false positives and false negatives, the model generally succeeds in removing most sky keypoints from the images. We chose to train the model with a smaller number of samples aiming to develop a lightweight model that consumes minimal computational resources and does not impact the workflow of any solution it is integrated into. A key advantage of our approach to be mentioned is utilising features already extracted in various vision-based approaches, providing an improvement with low computational complexity. By focusing on the efficient removal of sky keypoints, we can enhance the overall performance of vision-based applications, ensuring they remain reliable and effective even in challenging outdoor environments.

(a)



(b)

Figure 6. Classification of keypoints: (a) Ricoh Theta S image RGB; (b) Keypoints belonging to the sky (blue) and non-sky (red).

**Acknowledgements**

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) (88887.310313/2018-00), by the National Council for Scientific and Technological Development, CNPq (grant n. 303670_2018-5), by the São Paulo Research Foundation, FAPESP (Grant: 2021/06029-7) and by the Academy of Finland (AKA) (353264 - Digital technologies, risk management solutions and tools for mitigating forest disturbances (MULTIRISK));.

**References**

Alcantarilla, P.F., Bartoli, A., Davison, A.J., 2012. KAZE Features, in: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (Eds.), Computer Vision – ECCV 2012, *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 214–227. https://doi.org/10.1007/978-3-642-33783-3_16

Bay, H., Ess, A., Tuytelaars, T., Van Gool, L., 2008. Speeded-Up Robust Features (SURF). Comput. Vis. Image Underst. 110, 346–359. https://doi.org/10.1016/j.cviu.2007.09.014

Braga-Neto, U., 2020. *Fundamentals of Pattern Recognition and Machine Learning*. Springer International Publishing, Cham. https://doi.org/10.1007/978-3-030-27656-0

Breiman, L., 2001. *Random Forests*. Mach. Learn. 45, 5–32. https://doi.org/10.1023/A:1010933404324

Forstner, W., 1986. A feature based correspondence algorithm for image matching. *Int Arch Photogramm*. 26, 150–166.

Gao, X., Zhang, T., 2021. *Introduction to Visual SLAM: From Theory to Practice*. Springer Singapore, Singapore. https://doi.org/10.1007/978-981-16-4939-4

Géron, A., 2019. *Hands-On Machine Learning with Scikit-Learn, Keras, and Tensorflow: Concepts, Tools, and Techniques to Build Intelligent Systems*, 2rd ed. O’Reilly Media, Canada.

Kato, S., Kitamura, M., Suzuki, T., Amano, Y., Waseda University, Electronic Navigation Research Institute, 2016. NLOS Satellite Detection Using a Fish-Eye Camera for Improving GNSS Positioning Accuracy in Urban Area. J. Robot. Mechatron. 28, 31–39. https://doi.org/10.20965/jrm.2016.p0031

Khan, A.I., Al-Habsi, S., 2020. *Machine Learning in Computer Vision*. Procedia Comput. Sci. 167, 1444–1451. https://doi.org/10.1016/j.procs.2020.03.355

Lowe, D.G., 2004. Distinctive image features from scale invariant keypoints. Int. J. Comput. Vis. 60, 91–110.

Nice, K.A., Wijnands, J.S., Middel, A., Wang, J., Qiu, Y., Zhao, N., Thompson, J., Aschwanden, G.D.P.A., Zhao, H., Stevenson, M., 2020. Sky pixel detection in outdoor imagery using an adaptive algorithm and machine learning. Urban Clim. 31, 100572. https://doi.org/10.1016/j.uclim.2019.100572

Rublee, E., Rabaud, V., Konolige, K., Bradski, G., 2011. ORB: An efficient alternative to SIFT or SURF, in: I*EEE International Conference on Computer Vision*. pp. 2564–2571.

Shao, C., Zhang, C., Fang, Z., Yang, G., 2020. A Deep Learning-Based Semantic Filter for RANSAC-Based Fundamental Matrix Calculation and the ORB-SLAM System. IEEE Access 8, 3212–3223. https://doi.org/10.1109/ACCESS.2019.2962268

Shen, Y., Wang, Q., 2013. Sky Region Detection in a Single Image for Autonomous Ground Robot Navigation. Int. J. Adv. Robot. Syst. 10, 362. https://doi.org/10.5772/56884

Shobha, G., Rangaswamy, S., 2018. *Machine Learning, in: Handbook of Statistics*. Elsevier, pp. 197–228. https://doi.org/10.1016/bs.host.2018.07.004

Tareen, S.A.K., Saleem, Z., 2018. A comparative analysis of SIFT, SURF, KAZE, AKAZE, ORB, and BRISK, in: 2018 *International Conference on Computing, Mathematics and Engineering Technologies* (iCoMET), IEEE, Sukkur, pp. 1–10. https://doi.org/10.1109/ICOMET.2018.8346440

Wang, Y., Duan, X., Sun, Y., Wang, J., 2021. A Visual SLAM Algorithm Based on Image Semantic Segmentation in Dynamic Environment, in: 2021 *4th International Conference on Pattern Recognition and Artificial Intelligence* (PRAI), IEEE, Yibin, China, pp. 401–405. https://doi.org/10.1109/PRAI53619.2021.9550800

Yogamani, S., Hughes, C., Horgan, J., Sistu, G., Varley, P., O’Dea, D., Uricar, M., Milz, S., Simon, M., Amende, K., Witt, C., Rashed, H., Chennupati, S., Nayak, S., Mansoor, S., Perroton, X., Perez, P., 2019. WoodScape: A multi-task, multi-camera fisheye dataset for autonomous driving. https://doi.org/10.48550/ARXIV.1905.01489

Yu, C., Liu, Z., Liu, X., Xie, F., Yang, Y., Wei, Q., Fei, Q., 2018. DS-SLAM: A Semantic Visual SLAM towards Dynamic Environments. https://doi.org/10.48550/ARXIV.1809.08379

Zhao, X., Hu, Q., Zhang, X., Wang, H., 2022. An ORB-SLAM3 Autonomous Positioning and Orientation Approach using 360-degree Panoramic Video, in: 2022 2*9th International Conference on Geoinformatics*, IEEE, Beijing, China, pp. 1–7. https://doi.org/10.1109/Geoinformatics57846.2022.9963855