THE UNIVERSITY OF SYDNEY

The University of Sydney

Aerospace, Mechanical and Mechatronic Engineering

AMME5710 Assignment 2

SID: 520481220

An Assignment submitted for the UoS:

Computer Vision and Image Processing

September 27, 2025

Contents

1 Question 1

1.1 Introduction

Stereo vision enables depth perception from two-dimensional images captured at different viewpoints by exploiting disparities to infer three-dimensional structure [12]. This principle, inspired by human binocular vision, underpins applications in autonomous navigation, industrial inspection, environmental monitoring, and underwater surveying [5].

In marine research, stereo vision is particularly advantageous because it provides dense depth maps without disturbing habitats. Unlike LiDAR, which suffers from light scatter and attenuation in water, stereo rigs have been used effectively for mapping coral reefs, estimating fish populations, and monitoring benthic habitats [3]. These systems allow rapid acquisition of spatial data along transects, producing three-dimensional models that support assessment of reef health, structural complexity, and species distribution.

Modern stereo reconstruction relies on robust feature detection and matching algorithms. Notably, the Scale-Invariant Feature Transform (SIFT) [9] and Oriented FAST and Rotated BRIEF (ORB) [11] are widely used to extract keypoints and descriptors resilient to changes in scale, viewpoint, and illumination. Reliable matching of these features enables geometric triangulation to reconstruct scene points. Combined with pose estimation from auxiliary sensors, stereo-derived point clouds can be aligned in a global reference frame for large-scale mapping.

This report implements a complete stereo pipeline for coral reef reconstruction using SIFT and ORB for feature extraction, with results evaluated in terms of accuracy, density, and structural fidelity. Challenges specific to underwater imaging, including light attenuation, turbidity, and refractive distortion, are also addressed.

1.2 Methodology

1.2.1 Feature Extraction and Matching

The first stage of the reconstruction pipeline involves detecting salient image features and establishing reliable correspondences between rectified stereo pairs. Two complementary feature detectors, ORB and SIFT, were employed to balance computational efficiency with robustness to changes in illumination and texture.

For each stereo image pair, keypoints were detected in both the left and right images. Around each keypoint, a descriptor vector was computed to encode local intensity patterns. Let d_i^L and d_j^R denote descriptor vectors from the left and right images, respectively. Matching was performed by identifying the nearest neighbours under the Euclidean distance:

$$\text{match}(i) = \arg\min_j \|d_i^L - d_j^R\|_2. \tag{1}$$

All detected keypoints for a representative image are shown in Figure 1, while the initial matches between left and right images after filtering are illustrated in Figure 2.

To reject ambiguous correspondences, Lowe's ratio test was applied. A match between descriptors d_i^L and d_j^R was retained only if the distance to the closest neighbour was significantly smaller than to the second-closest neighbour:

$$\frac{\|d_i^L - d_{j_1}^R\|_2}{\|d_i^L - d_{j_2}^R\|_2} < \tau, \tag{2}$$

Where j_1 and j_2 are the closest and second-closest matches, and τ is the ratio threshold (typically $\tau \approx 0.75$).

Although the ratio test removed many spurious matches, additional geometric filtering was necessary. This was achieved by estimating the fundamental matrix F using RANSAC (Random Sample Consensus). For a putative correspondence (x^L, x^R) expressed in homogeneous coordinates, the epipolar constraint must hold:

$$(x^R)^\top F x^L = 0. \tag{3}$$

Correspondences inconsistent with this relation were classified as outliers and discarded. RANSAC iteratively sampled minimal subsets of matches, estimated candidate fundamental matrices, and retained the solution that maximized the number of inliers. The resulting inlier correspondences demonstrated consistent alignment across stereo pairs, as shown in Figure 2, confirming that features were successfully extracted and matched for reliable reconstruction.

The resulting inlier correspondences provide a robust foundation for triangulation in the next stage of the pipeline.

1.2.2 Image Rectification

To simplify correspondence search, the stereo image pairs were rectified so that epipolar lines became horizontal. This ensured that potential matches lay along the same image row, reducing ambiguity. The rectified stereo images, ready for

triangulation, are visually similar to the matched points shown in Figure 2, but with horizontal alignment.

1.2.3 Triangulation

After establishing robust correspondences between rectified stereo images, the 3D position of each scene point was recovered through triangulation. The underlying principle relies on *epipolar geometry*, which constrains the location of a point in one image given its position in the other.

Let x_1 and x_2 represent the normalized homogeneous coordinates of a point in the left and right images, respectively. These coordinates satisfy the essential matrix constraint:

$$x_2^\top E x_1 = 0, \tag{4}$$

Where E is the essential matrix that encodes the relative rotation and translation between the two cameras. For unnormalized pixel coordinates, this relationship generalizes to the *fundamental matrix* F:

$$x_2^\top F x_1 = 0. \tag{5}$$

This epipolar constraint ensures that a point in one image lies along a corresponding epipolar line in the other image, reducing the correspondence search from two dimensions to one. Once the camera projection matrices P_L and P_R are known, a pair of matching points (x^L, x^R) can be triangulated by solving the overdetermined system:

$$x^L = P_L X, \quad x^R = P_R X, \tag{6}$$

Where X is the homogeneous 3D point. The solution is typically obtained using a least-squares approach, minimizing reprojection error in both images. Points with negative depth (behind either camera) are discarded via a cheirality check. The triangulated points are then transformed to the world frame, as shown in Figure 3.

1.2.4 Transformation to World Frame

Triangulated points initially reside in the left camera's reference frame. To integrate points across multiple frames, they are transformed into the global world coordinate system.

Given the camera-to-world rotation R_{wc} and translation t_{wc}, the world-frame coordinates X_{world} are computed as:

$$X_{world} = R_{cw} X_L + t_{cw}, \quad \text{where; } R_{cw} = R_{wc}^\top, \quad t_{cw} = -R_{wc}^\top t_{wc} \tag{7}$$

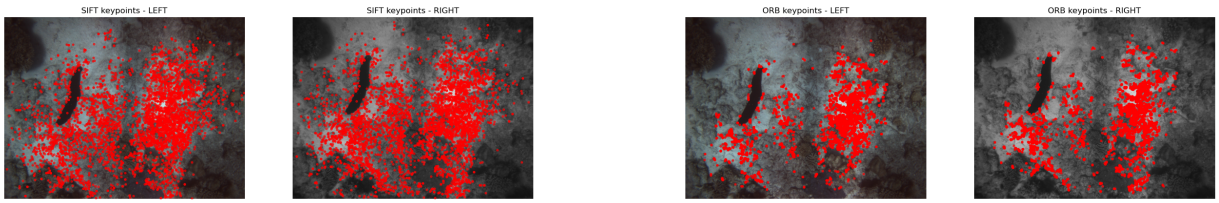
This procedure was applied to all 49 stereo pairs, and the transformed triangulations were aggregated to form a coherent, world-referenced point cloud of the coral reef terrain. This confirms that the final 3D reconstruction was correctly assembled from the individual stereo observations. Enabling the culmination of points from all stereo pairs into a coherent 3D reconstruction of the coral reef terrain, as seen in Figure 3.

1.3 Results and Discussion

1.3.1 Feature extraction and Matching

The reconstruction pipeline was evaluated using two feature detectors: ORB and SIFT. Although the total number of inliers is similar, the nature of these matches differs. ORB, being a binary descriptor, is computationally efficient but less discriminative in areas with repetitive textures. SIFT, with its floating point descriptors and scale and rotation invariance, captures more subtle variations, producing slightly denser and more spatially uniform matches.

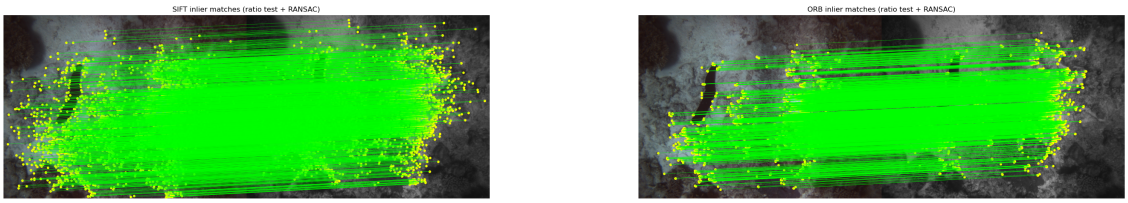
Visual inspection of the inlier matches confirms that both detectors correctly identify correspondences along epipolar lines. However, SIFT tends to cover low texture and high frequency regions more reliably, as seen in Figure 1. The ratio threshold used during feature matching directly impacts the number of retained correspondences: stricter thresholds reduce spurious matches but may remove valid points, while looser thresholds increase match density but risk including outliers. Here, thresholds of 0.85 for ORB and 0.75 for SIFT provided a balance between robustness and coverage seen in Figure 2.



(a) Detected keypoints using SIFT.

(b) Detected keypoints using ORB.

Figure 1: Comparison of detected keypoints using SIFT and ORB in the input image.



(a) SIFT matched points after RANSAC filtering.

(b) ORB matched points after RANSAC filtering.

Figure 2: Matched points after RANSAC filtering using different feature descriptors.

1.3.2 Random Region Analysis: Keypoints vs Image Properties

To understand which image characteristics influence feature detection, random regions of the stereo images were analysed. Each region was evaluated for number of detected keypoints, mean brightness, contrast (standard deviation), and contour density (via Canny edges). Linear regression and Pearson correlation coefficients (R) were computed for keypoints versus these metrics. Results are summarised in Table 1.

Detector	Keypoints vs Edges	Keypoints vs Brightness	Keypoints vs Contrast
ORB	0.950	0.473	0.416
SIFT	0.961	0.526	0.514

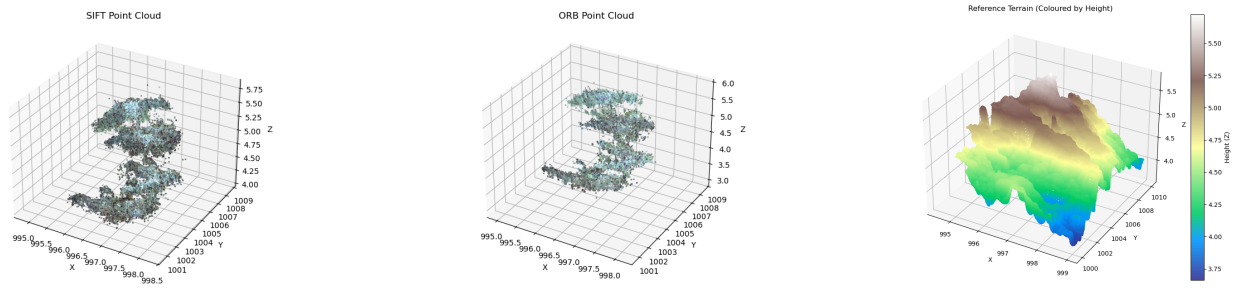
Table 1: Pearson correlation coefficients (R) between number of keypoints and image properties across random regions.

Both ORB and SIFT exhibit a very strong positive correlation with the number of edge pixels (contour density), indicating that regions with rich geometric structure generate the most keypoints. Brightness and contrast show moderate correlations, suggesting that while illumination and texture strength influence feature detection, edge content is the dominant factor. The scatter plots for these can be seen in subsection A.1.

The analysis also identified the best and worst performing regions in terms of keypoints. The best regions typically contained bright areas with ridges, edges, and other contours, enabling high feature density such as coral and plant areas. The worst regions were darker, relatively uniform areas with few contours such as sandy areas, producing few detectable keypoints. These observations confirm that texture and contour richness are critical for successful stereo feature detection. This further shown in the appendix in subsection A.2.

1.3.3 Triangulation and 3D Point Cloud construction

Triangulating the inlier matches resulted in 92,370 3D points for ORB and 99,972 points for SIFT. The higher point count from SIFT is consistent with its better match distribution across the images, particularly in textured or low contrast regions. Both methods successfully reconstruct major terrain features of the coral reef, including ridges and depressions, as visualised in the 3D point clouds Figure 3. The colour mapped visualisations indicate that SIFT captures more fine scale structure, which is beneficial for applications requiring detailed topography.

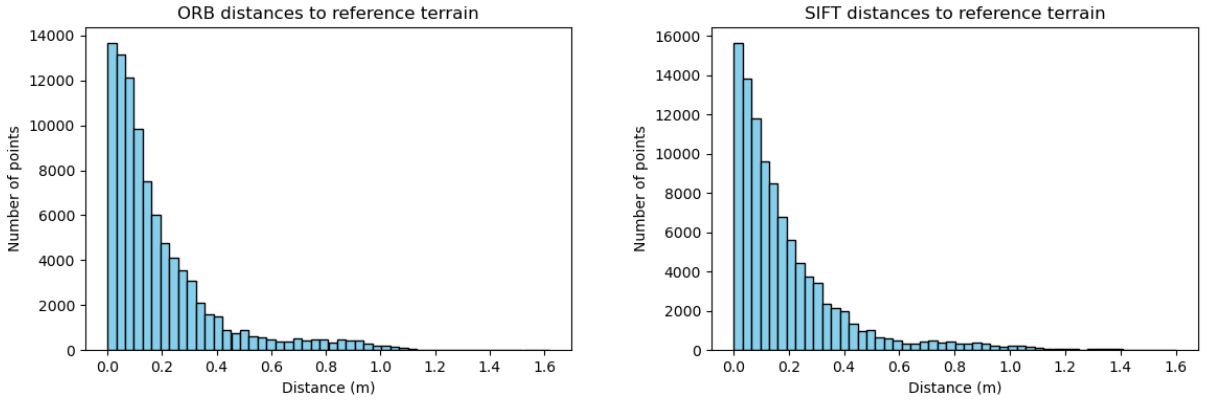


(a) SIFT inliers in world frame. (b) ORB inliers in world frame. (c) Reference terrain.

Figure 3: Triangulated inlier points in the world frame for SIFT and ORB feature descriptors, compared against the reference terrain.

1.3.4 Accuracy and comparison to ground truth

Distances between the reconstructed points and the reference terrain provide quantitative validation. ORB achieved a mean point to terrain distance of 0.188 m and median of 0.121 m, while SIFT showed a mean of 0.189 m and median of 0.125 m. Standard deviations were similar, indicating consistent overall reconstruction accuracy for both methods. Maximum deviations (approximately 1.6 m) occur in regions with occlusions or sparse features, highlighting limitations in feature extraction in shadowed or highly reflective areas. These discrepancies are visualised in the histograms compared to the ground truth Figure 4.



(a) Histogram of ORB keypoints. (b) Histogram of SIFT keypoints.

Figure 4: Comparison of keypoint histograms for ORB and SIFT feature detectors.

1.3.5 Discussion

Overall, both ORB and SIFT produce accurate reconstructions with mean errors below 0.2 m, demonstrating the effectiveness of the pipeline. The region analysis highlights that contour rich regions are most suitable for feature detection, while dark, low texture areas limit keypoint density. SIFT is preferable for dense reconstructions and capturing fine scale features, whereas ORB is suitable for faster processing with slightly lower coverage. Adjusting feature extraction parameters allows tuning between match robustness, point cloud density, and computational cost, emphasising the trade offs inherent in stereo reconstruction workflows.

The effect on the final reconstructions is clear. The choice of detector and parameters directly influences accuracy, density, and the types of structures captured in the point cloud. ORB produces slightly sparser point clouds but still achieves similar overall accuracy, making it better suited for applications where speed is prioritised over coverage. SIFT achieves higher density and captures more fine detail, which results in richer representations of textured structures such as coral ridges and plant covered surfaces. However, both detectors struggle in low texture regions, where uniform surfaces like sand result in reduced point density and less reliable accuracy. This shows that selecting a detector and tuning thresholds is not only a matter of balancing computation and robustness, but also determines the level of structural detail that can be represented in the final 3D model.

2 Question 2

2.1 Introduction

Scene classification is a fundamental task in computer vision that involves assigning semantic labels such as "forest", "city", or "desert" to images based on their environmental context. Unlike object classification, which focuses on identifying specific objects, scene classification requires capturing global visual cues, spatial arrangements, and contextual information that describe the overall environment depicted in an image [10]. This makes it a challenging yet essential problem for applications in robotics, autonomous navigation, content based image retrieval, and geographic information systems [1], [15].

Traditional approaches to scene classification relied on handcrafted feature extraction methods, including color histograms, texture descriptors, and edge based features [2]. Local descriptors such as Scale Invariant Feature Transform (SIFT) and Histogram of Oriented Gradients (HOG) were particularly effective for capturing structural information [4], [8]. Classifiers such as Support Vector Machines (SVMs) and k Nearest Neighbors (kNN) were then used to map extracted features to semantic categories [14]. These feature engineering approaches achieved strong performance but often struggled to generalize across large and diverse datasets.

In recent years, the field has been transformed by the adoption of deep learning, in particular convolutional neural networks (CNNs). Models such as AlexNet, VGG, and ResNet have shown success in learning hierarchical visual features directly from raw pixel data [6], [7]. Large scale datasets such as the Places database [15] have further accelerated progress by providing millions of labeled scene images, enabling models to capture complex semantic structures. Although deep models dominate state of the art performance in scene recognition tasks, classical feature based approaches still offer value in constrained scenarios where computational resources or dataset sizes are limited [13].

This report presents the systematic development of a machine learning algorithm for scene classification using a subset of the Places dataset. The approach integrates multiple feature extraction techniques including color histograms, texture descriptors, and HOG features with an SVM classifier to evaluate performance on RGB images. The methodology involves pre processing, feature engineering, model selection, parameter tuning, and validation, with performance assessed using accuracy, precision, recall, F1 score, and top k accuracy.

2.2 Methodology

2.2.1 Choice of model

Two classification algorithms were considered for this study: Support Vector Machines (SVM) and k-Nearest Neighbours (KNN). Both are widely used in image classification tasks but operate under fundamentally different principles.

KNN is a non-parametric, instance-based algorithm that classifies a sample by the majority label of its k nearest neighbours in feature space. It is simple and effective in low dimensions, but scales poorly: distances must be computed against all stored samples, requiring high memory and making inference slow in high-dimensional feature spaces. This limits its suitability for large-scale scene classification.

SVM is a discriminative classifier that constructs a decision boundary by maximising the margin between support vectors. With kernel functions such as the RBF, it captures non-linear boundaries common in complex scene data. Unlike KNN, SVM yields a compact model that avoids storing the full training set, enabling efficient inference and strong generalisation in high-dimensional spaces.

For the present study, SVM was selected over KNN for three main reasons. First, SVM is more robust in high-dimensional feature spaces, such as those produced by concatenated HOG, colour, and texture descriptors. Second, its kernelised decision boundaries are better suited to capturing the non-linear separations that arise in scene categorisation. Finally, SVM offers faster prediction once trained, as classification depends only on a subset of support vectors rather than the full dataset. These advantages make SVM a more reliable and scalable option for scene classification, as confirmed by its superior cross-validation performance compared to KNN.

2.2.2 Pre-Processing

All images were first resized to a fixed resolution to ensure that feature vectors had consistent dimensionality across the dataset. A resolution of 64×64 pixels was selected based on preliminary experiments, which showed that this size provides a suitable trade-off between computational efficiency and retention of sufficient scene detail for reliable classification.

To mitigate the effects of varying lighting conditions, histogram equalisation was applied to the grayscale version of each image. This step normalises the image brightness and enhances contrast, making texture and structural features more consistent and easier for the classifier to interpret.

Finally, all pixel values were scaled to the range $[0, 1]$. This normalisation ensures numerical stability during feature extraction and model training, particularly for algorithms such as Support Vector Machines that are sensitive to the

scale of input features. Together, these pre-processing steps standardise the input data, improve feature quality, and support robust learning across diverse scene classes.

2.2.3 Feature Extraction

For this project, a feature extraction strategy was adopted that combines Histogram of Oriented Gradients (HOG) descriptors with a multi-resolution analysis framework, alongside complementary colour and texture descriptors. The intuition behind this choice is that scenes contain structural, chromatic, and textural information at different scales, all of which are important for accurate classification. Fine-grained textures such as brick patterns or grass blades are best captured at higher resolutions, while broader structures such as horizons or building silhouettes emerge more clearly at coarser scales. Similarly, certain classes are strongly characterised by colour composition (e.g., sky, vegetation), while others are better distinguished by their surface textures (e.g., sand, foliage). A single type of descriptor would not sufficiently capture these different cues, so a multi-feature, multi-resolution approach was introduced.

HOG descriptors: The HOG descriptors were computed across multiple image scales by downsampling the input image and extracting features at each resolution. At every level, HOG encodes edge and gradient distributions, which are particularly well suited for distinguishing scene categories where object boundaries and textures vary significantly. The resulting descriptors from each resolution were concatenated into a unified feature vector, ensuring that both local and global structural information contributed to the final representation. This choice is supported by prior work showing that HOG is robust to illumination changes and emphasises shape information, which is often the most discriminative element in scene categorisation tasks.

Colour histograms: In addition to shape, global colour statistics were incorporated using RGB histograms. Many natural and urban scenes are characterised by dominant colour distributions (e.g., the blues of oceans and skies, the greens of vegetation, the earthy tones of deserts). Without colour features, categories with similar structure but different colour composition could be easily confused. To avoid bias due to illumination, histograms were normalised so that they reflected relative proportions rather than absolute intensity.

Texture descriptors (LBP): Local Binary Patterns (LBP) were used to capture fine-grained texture. LBP works by thresholding local neighbourhoods to produce binary patterns, which are then aggregated into histograms. This approach is highly effective at distinguishing textures such as sand, stone, or foliage that may appear in different categories but are not well captured by HOG or colour alone. Texture equalisation was applied prior to LBP extraction to reduce the effect of uneven illumination, ensuring more consistent texture representation across the dataset.

Experimental parameter selection: The parameters for HOG extraction — including cell size, block size, number of bins, and stride — were determined experimentally. An initial baseline was set using standard defaults from the literature (cell size $= 8 \times 8$ pixels, block size $= 2 \times 2$ cells, 9 bins). From there, systematic experiments were conducted by varying one parameter at a time while keeping others constant, and classification accuracy was evaluated on a validation split. A similar process was applied to the colour histograms (testing different bin counts) and LBP descriptors (testing different radii and neighbourhood sizes).

Feature fusion and evaluation: After individual evaluation, the different feature sets were concatenated into a composite feature vector. Standardisation was applied to ensure that no single feature type dominated due to differences in magnitude (e.g., HOG vectors being much larger than colour histograms). To determine the final feature set, a systematic methodology was followed:

- Step 1: Evaluate each descriptor independently (HOG, colour, LBP).
- Step 2: Test pairwise combinations (HOG + colour, colour + LBP, HOG + LBP).
- Step 3: Test full fusion of all three descriptors.
- Step 4: Tune hyperparameters of the SVM classifier via 5-fold cross-validation on the training set.

To validate parameter choices, cross-validation was performed on the training set using both classification accuracy and F1 score as evaluation metrics. Accuracy alone was insufficient, since some scene classes were underrepresented, so the F1 score was included to ensure balanced performance across categories. This evaluation framework allowed a fair comparison between different feature configurations. The results showed that the combination of HOG with three-scale multi-resolution descriptors consistently outperformed both single-scale HOG and raw intensity histogram baselines. These findings support the conclusion that structural information captured across multiple scales provides the strongest discriminative power for scene classification, particularly when combined with colour and texture features.

This iterative process ensured that the final proposed model was not chosen arbitrarily but was instead supported by experimental evidence. The results demonstrated that combining all three descriptors consistently produced the best classification performance, confirming that structural, chromatic, and textural features are complementary.

2.2.4 Algorithm Selection

For the scene classification task, a Support Vector Machine with a radial basis function kernel was selected as the primary classifier. The choice of Support Vector Machine was motivated by its strong performance in high dimensional feature spaces and its ability to handle non linear class boundaries, which are common in scene images due to complex combinations of color, texture, and structure.

The key hyperparameters for the radial basis function Support Vector Machine are the regularization parameter C and the kernel width parameter γ. The parameter C controls the trade off between maximizing the margin and minimizing classification error on the training set, while γ determines the influence of individual training samples on the decision boundary. Selecting these parameters appropriately is crucial for achieving optimal generalization performance.

To determine suitable values for C and γ, an experimental grid search was performed. The training set was split using stratified five fold cross validation to ensure balanced representation of all scene classes in each fold. A grid of candidate values for C (0.1, 1, 10) and γ (scale, 0.01, 0.001) was evaluated using classification accuracy as the scoring metric. The combination of parameters that maximized cross validation accuracy was selected as the final model configuration.

A standard scaler was applied to the features as part of a preprocessing pipeline. This ensured that each feature dimension had zero mean and unit variance, which prevents features with larger numerical ranges from dominating the Support Vector Machine optimization. Scaling the features in this way improves the stability of training and allows the radial basis function kernel to operate effectively across all feature types.

The final trained model achieved strong overall accuracy and top three accuracy on the hold out test set, confirming that the experimental approach to parameter selection and feature scaling was effective.

2.2.5 Training and Validation

The dataset was divided into training and test subsets, with 90 percent of the data used for training and 10 percent held out for final evaluation. A stratified split was applied to preserve the proportion of each scene class across both subsets. Within the training set, stratified five fold cross validation was performed during hyperparameter tuning to ensure robust selection of the Support Vector Machine parameters C and γ. This procedure was implemented via an exhaustive grid search over a predefined set of candidate values.

The best model configuration was chosen based on the highest cross validation accuracy. Once selected, the model was retrained on the entire training set and evaluated once on the held out test set to provide an unbiased estimate of generalisation performance. Evaluation metrics included overall accuracy, per-class precision, recall, and F1-score. These metrics are suitable for scene classification because they capture both the overall classification success and the balance of performance across classes. In particular, precision and recall account for underrepresented classes, while the F1-score provides a single measure of class-wise effectiveness, ensuring that rare scene categories are not neglected.

Although evaluation was performed only on the reserved test set, the final model could optionally be retrained on the entire dataset (training plus test) for deployment to maximise available data. This retraining does not affect the reported performance metrics and was not used for the evaluation reported here.

2.3 Results and Discussion

2.3.1 Model Configuration

Table 2: Model configuration and feature extraction settings (combination 11)

Parameter	Value
Image resolution	64×64 pixels
HOG pixels per cell	16×16
HOG cells per block	2×2
HOG orientations	6
Color histogram bins	16
Texture histogram bins	8
SVM kernel	Radial basis function (RBF)
SVM C	10
SVM γ	0.001
Feature scaling	StandardScaler applied to all features

The final scene classification model was configured with the parameters shown in Table 2. These settings were determined through experimental validation, balancing feature richness, computational efficiency, and classification performance. The results of this experimental validation is seen in Appendix B. The image resolution and HOG parameters were chosen to

capture structural gradients while keeping the feature vector manageable. Color and texture histogram bins were selected to provide sufficient granularity to capture scene-specific variations without overfitting small differences between images. The StandardScaler was applied to all features to ensure numerical stability and comparable scaling across feature types. The SVM hyperparameters C and γ were tuned using 5-fold stratified cross-validation to achieve an optimal trade-off between bias and variance.

2.3.2 Classification Performance

The model achieved an overall test accuracy of 88.57% and a top-3 accuracy of 100%, indicating that the correct scene was consistently ranked among the top three predictions. Table 3 shows detailed per-class metrics.

Classes such as `ball_pit`, `desert`, and `park` achieved perfect precision, recall, and F1-scores, indicating that these scenes were both visually distinctive and well-represented in the dataset. Conversely, classes such as `snow` and `sky` show an imbalance between precision and recall. For instance, `snow` has perfect precision but low recall (0.4), suggesting that while predictions for `snow` are rarely incorrect, many actual `snow` images were misclassified. This indicates that certain bright or white dominated scenes, such as `road` or `sky`, can visually overlap with `snow`, leading to confusion.

Table 3: Per-class performance metrics

Class	Precision	Recall	F1-score	Top-3
ball_pit	1.000	1.000	1.000	1.000
desert	1.000	1.000	1.000	1.000
park	1.000	1.000	1.000	1.000
road	0.800	0.800	0.800	1.000
sky	0.714	1.000	0.833	1.000
snow	1.000	0.400	0.571	1.000
urban	0.833	1.000	0.909	1.000

Table 4: Confusion matrix for the test set

	ball_pit	desert	park	road	sky	snow	urban
ball_pit	5	0	0	0	0	0	0
desert	0	5	0	0	0	0	0
park	0	0	5	0	0	0	0
road	0	0	0	4	0	0	1
sky	0	0	0	0	5	0	0
snow	0	0	0	1	2	2	0
urban	0	0	0	0	0	0	5

Table 4 presents the confusion matrix for the test set, revealing detailed misclassification patterns. Most errors occurred between classes with similar visual properties or limited training examples. For example, `snow` was occasionally misclassified as `road`, likely due to shared brightness and low texture. Similarly, a `road` image was misclassified as `urban`, reflecting the contextual overlap since roads often appear in urban environments.

These misclassifications highlight the complementary nature of the extracted features. HOG captures structural gradients but may struggle with uniform textures such as snow or sky, while color histograms and texture descriptors help differentiate scenes with similar shapes but different coloration or surface patterns. The top-3 accuracy of 100% reinforces that even when errors occur, the correct label remains among the highest-ranking predictions, demonstrating robustness in practical applications.

2.3.3 Discussion

The experimental results demonstrate that the combination of HOG, color, edge, and texture features provides strong discriminative power for diverse scene categories. Pre-processing steps including histogram equalisation and feature scaling improved generalisation by reducing the impact of lighting variations and numerical inconsistencies. Hyperparameter tuning of the SVM using cross-validation ensured a balanced bias-variance trade-off, resulting in optimal performance for this dataset.

Misclassification patterns suggest areas for potential improvement. Scenes with high visual similarity, such as `snow` and `road`, or `sky` and `snow`, challenge the classifier, and additional features or larger datasets could reduce these errors. Compared with state-of-the-art deep learning approaches, which often achieve higher accuracies on large-scale datasets, the feature-based SVM offers interpretable features and competitive performance given the small dataset size and limited computational resources. Techniques such as convolutional neural networks or transformer-based models, potentially combined with transfer learning from large pre-trained datasets or region-focused methods like YOLO, could further improve classification by automatically learning hierarchical and context-sensitive features that are difficult to capture with hand-crafted descriptors.

The selected metrics are suitable for scene classification. Accuracy provides an overall measure of success, precision and recall reveal the classifier's ability to handle class imbalance and visually similar categories, and F1-score offers a balanced view of precision and recall. Top-3 accuracy confirms practical utility in scenarios where multiple candidate labels are acceptable, such as image retrieval or automated tagging.

In summary, the model configuration, feature extraction strategy, and pre-processing pipeline together create a robust, interpretable, and effective approach for scene classification, especially when working with limited heterogeneous image datasets. While deep learning approaches could further improve performance, the current feature-based SVM provides a strong baseline with clear interpretability and efficient computation.

References

[1] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder decoder architecture for image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017.

[2] A. Bosch, A. Zisserman, and X. Muñoz, "Image classification using random forests and ferns," in *IEEE International Conference on Computer Vision*, IEEE, 2007, pp. 1–8.

[3] M. Bryson, O. Pizarro, S. B. Williams, and M. Daly, "Aerial and underwater surveys of marine habitats," *Methods in Oceanography*, vol. 15-16, pp. 90–103, 2017. DOI: 10.1016/j.mio.2016.04.002.

[4] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, vol. 1, 2005, pp. 886–893.

[5] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge University Press, 2004.

[6] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, 2016, pp. 770–778.

[7] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, 2012, pp. 1097–1105.

[8] D. G. Lowe, "Distinctive image features from scale invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[9] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," in *International Journal of Computer Vision*, vol. 60, 2004, pp. 91–110. DOI: 10.1023/B:VISI.0000029664.99615.94.

[10] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *International Journal of Computer Vision*, vol. 42, no. 3, pp. 145–175, 2001.

[11] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "Orb: An efficient alternative to sift or surf," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2011, pp. 2564–2571. DOI: 10.1109/ICCV.2011.6126544.

[12] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *International Journal of Computer Vision*, vol. 47, no. 1-3, pp. 7–42, 2002. DOI: 10.1023/A:1014573219977.

[13] L. Xie, J. Yang, Q. Wang, Q. Tian, and B. Zhang, "Hybrid cnn and bow for scene classification," in *Proceedings of the British Machine Vision Conference*, 2015.

[14] J. Yang, K. Yu, Y. Gong, and T. Huang, "Linear spatial pyramid matching using sparse coding for image classification," in *IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, 2009, pp. 1794–1801.

[15] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba, "Places: A 10 million image database for scene recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 6, pp. 1452–1464, 2017.

A Question 1

A.1 Total Keypoint Feature Analysis

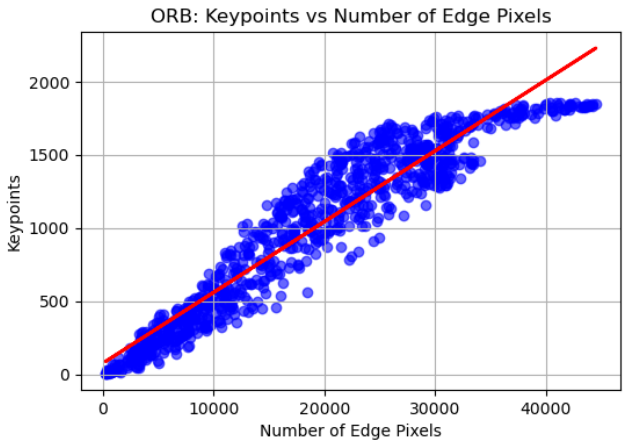


Figure 5: ORB keypoints compared with detected edges in the input image.

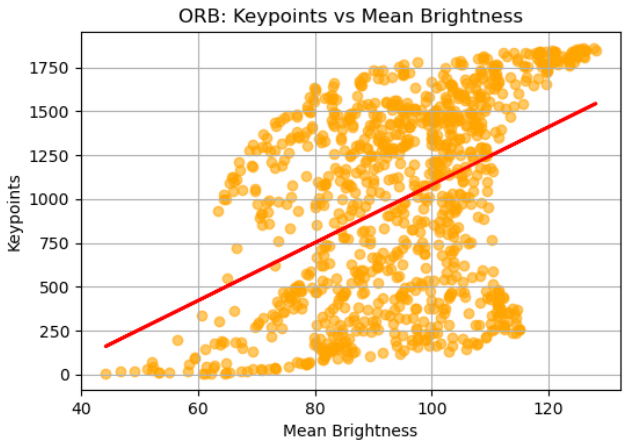


Figure 6: ORB keypoints under varying brightness conditions.

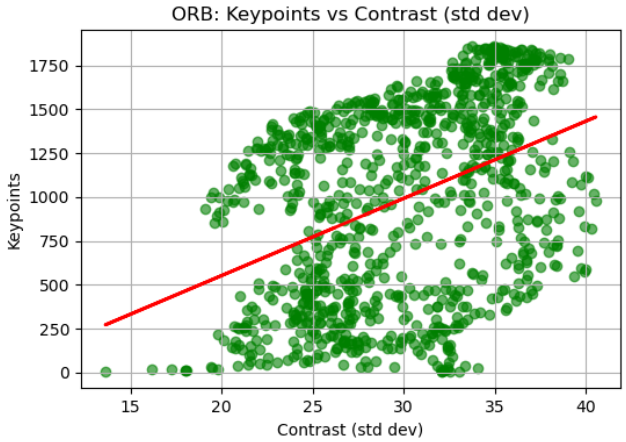


Figure 7: ORB keypoints under varying contrast conditions.

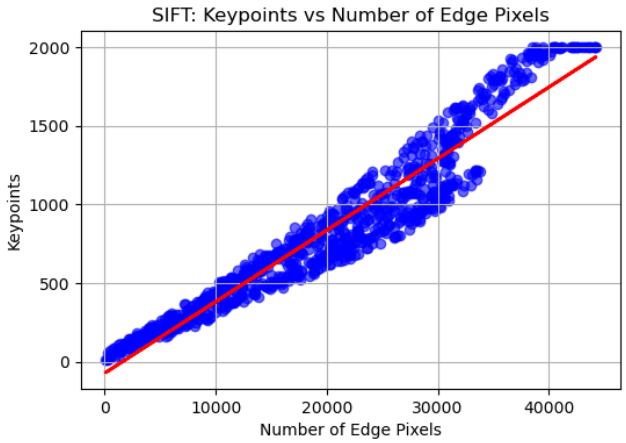


Figure 8: SIFT keypoints compared with detected edges in the input image.

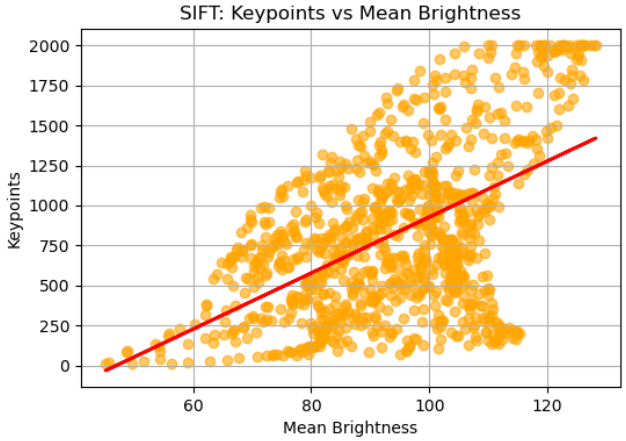


Figure 9: SIFT keypoints under varying brightness conditions.

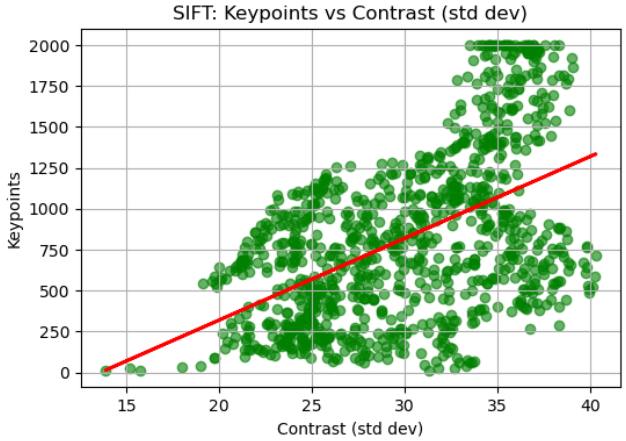


Figure 10: SIFT keypoints under varying contrast conditions.

A.2 Best and Worst Random Regions

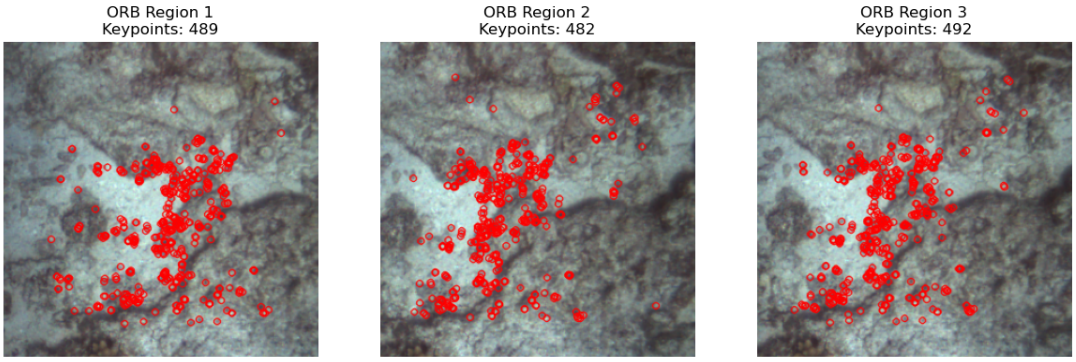


Figure 11: Best performing random region for ORB showing high keypoint density in a textured area with ridges and contours.

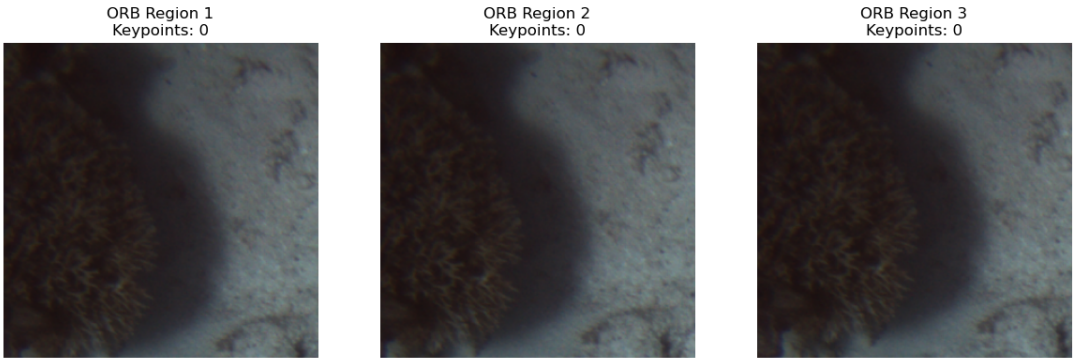


Figure 12: Worst performing random region for ORB with few detected keypoints due to low texture and uniform sandy appearance.

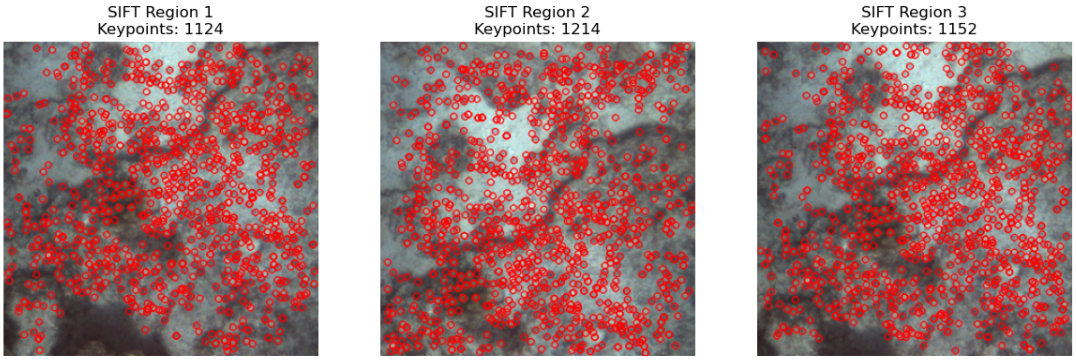


Figure 13: Best performing random region for SIFT illustrating dense keypoint coverage in a bright, structurally complex coral area.

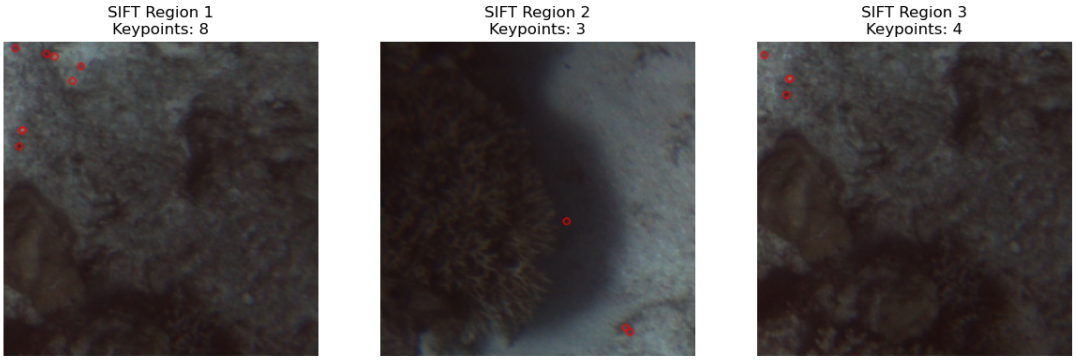


Figure 14: Worst performing random region for SIFT where darker, low contrast regions produced sparse and unreliable keypoints.

B Question 2

B.1 Experimental Combinations Results

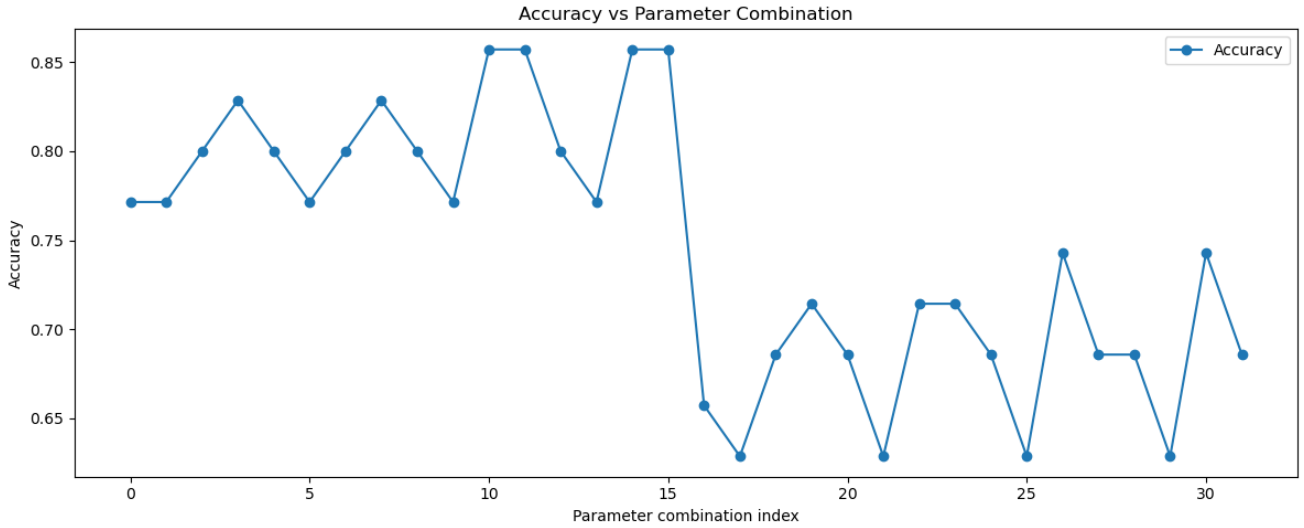


Figure 15: Overall classification accuracy across all scene categories.

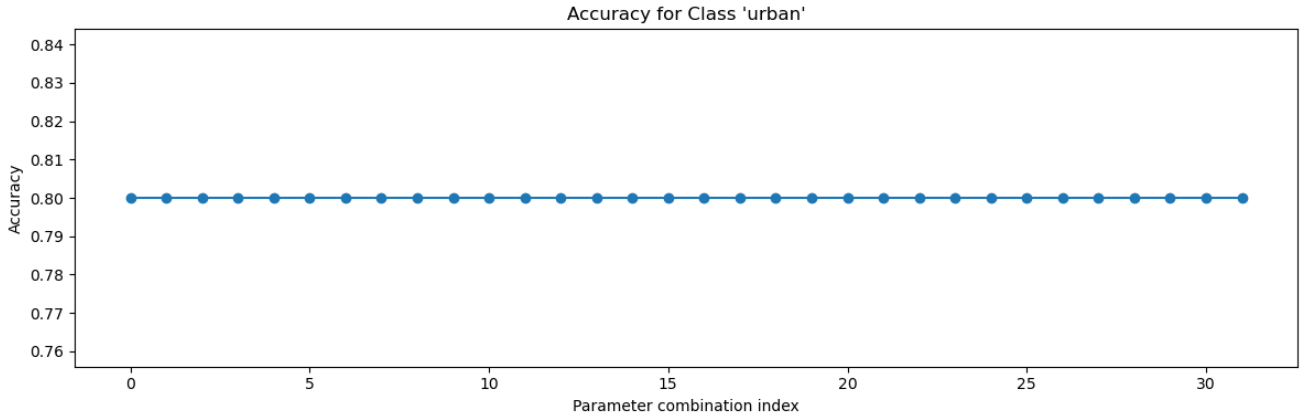


Figure 16: Classification accuracy for the `urban` scene category.

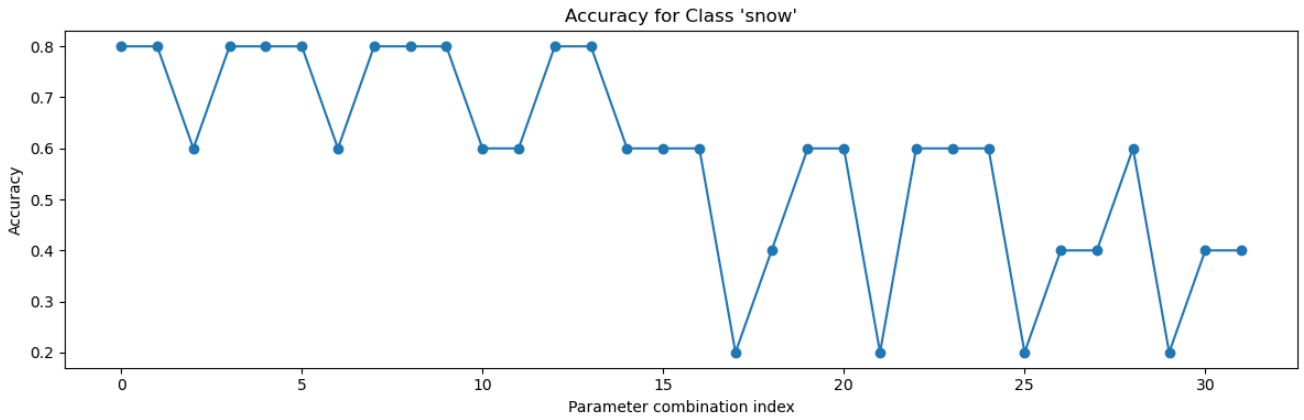


Figure 17: Classification accuracy for the `snow` scene category.

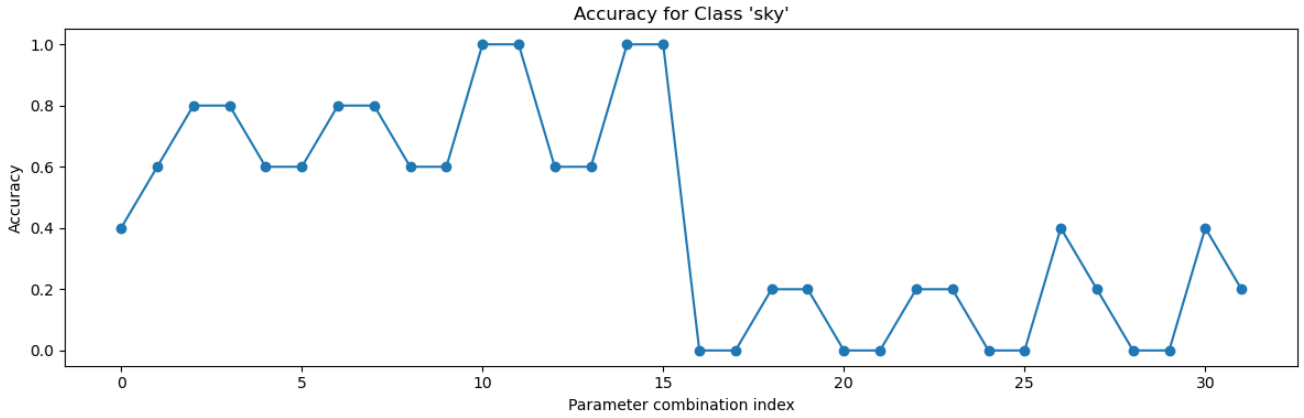


Figure 18: Classification accuracy for the `sky` scene category.

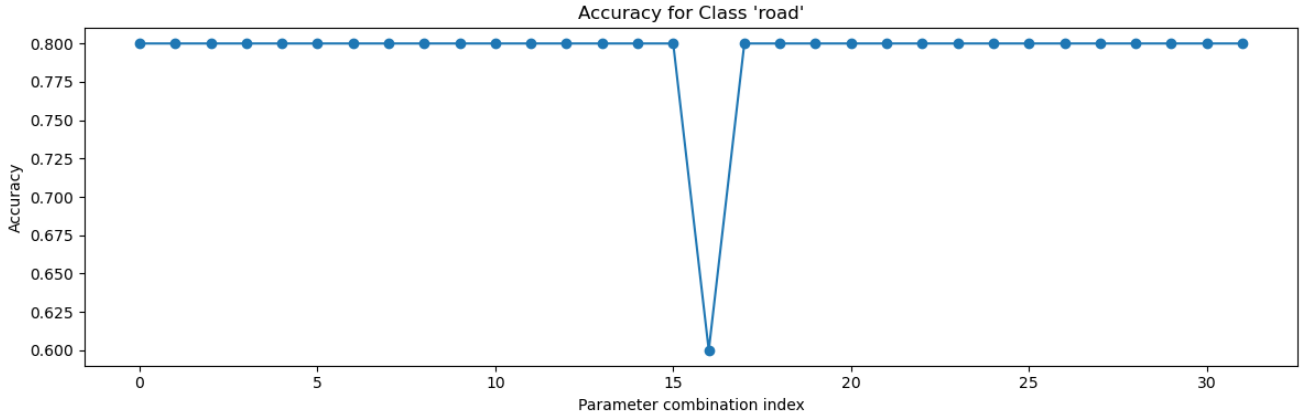


Figure 19: Classification accuracy for the `road` scene category.

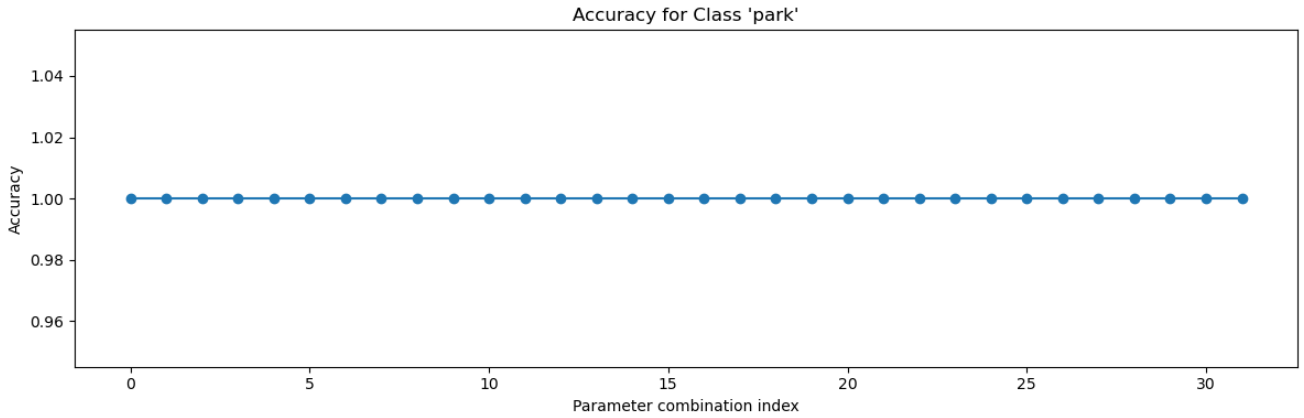


Figure 20: Classification accuracy for the `park` scene category.

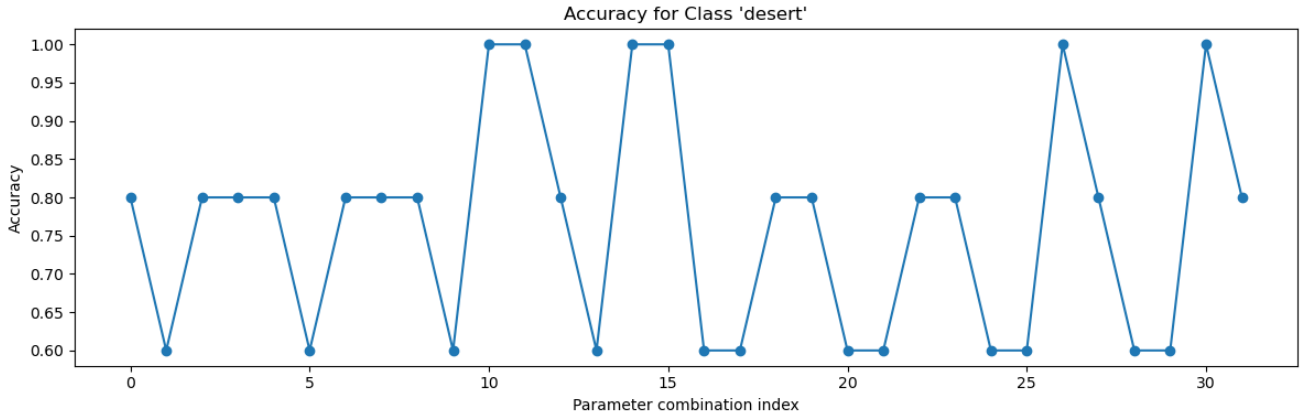


Figure 21: Classification accuracy for the `desert` scene category.

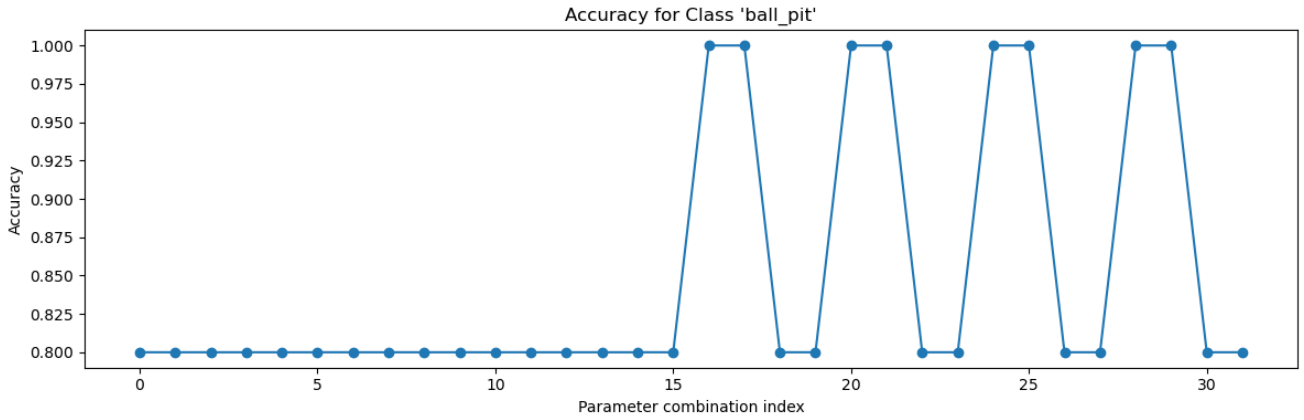


Figure 22: Classification accuracy for the `ball pit` scene category.