Master in
Computer Vision
Barcelona

C3 PROJECT - WEEK 1

Image Classification using Bag of Visual Words (BoVW)

Team 4:

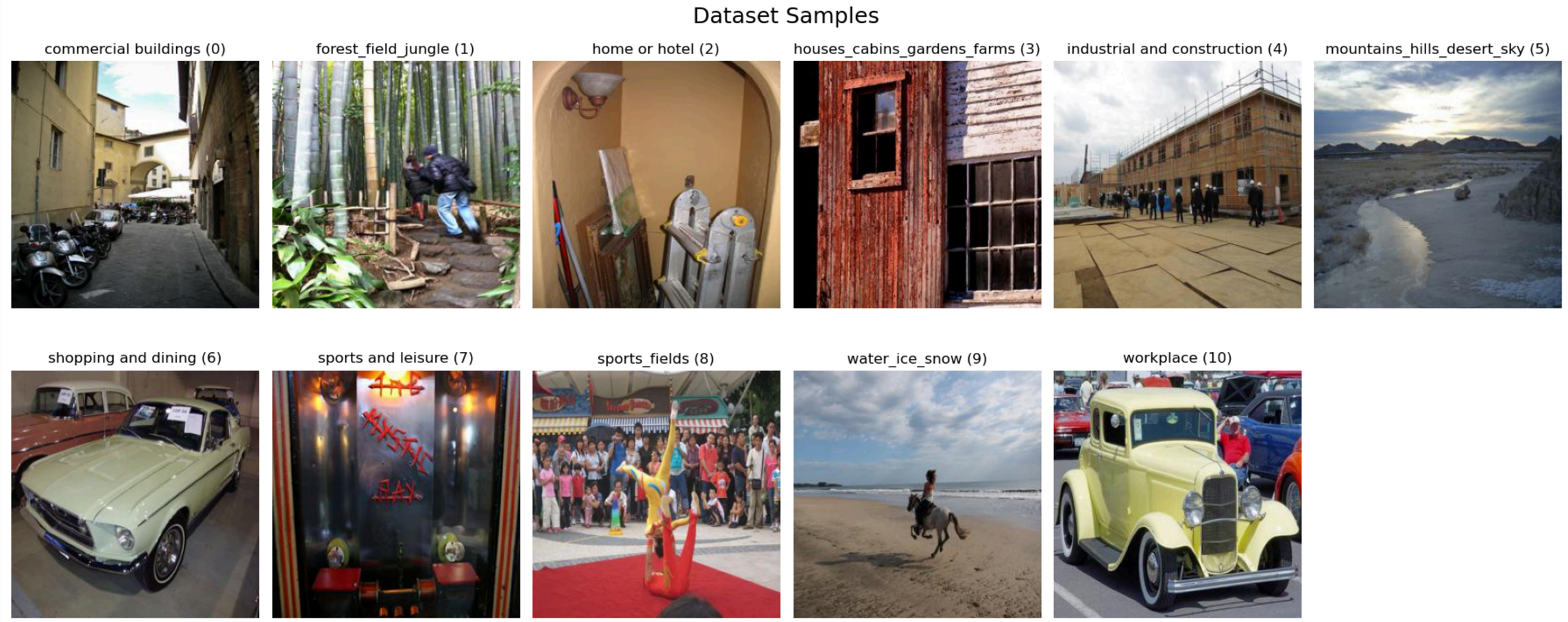
Alvaro Javier Diaz Laureano
Benet Ramió i Comas
Marina Rosell Murillo
Mohammed Oussama Ammouri

The Task & Dataset

Data: 11-Scene Dataset
Split: 8700 Training / 2200 Testing

800 Train / 200 Test per class *(Exception: 700 Train for 'Industrial & Construction')*

Goal: given a new image, classify into the right scene using BoVW



Approach: Experimental Strategy & Evaluation

Optimization Strategy: *Iterative "Greedy" Search*

- **Constraint**: A full Grid Search across all hyperparameters was infeasible due to high computational costs and time constraints.
- **Approach**: We adopted a step-wise iterative approach. We optimized one parameter at a time, selected the best-performing configuration (based on CV mean Accuracy), and fixed it as the "Base Model" for subsequent experiments.

Evaluation Protocols:

- **Model Selection**: We utilized 5-Fold Cross-Validation (CV) for every experiment.
  - *Metric*: We analyzed the Mean CV Accuracy and Standard Deviation to select stable hyperparameters that generalize well.
  - *Overfitting Check*: We monitored the gap between Train and CV mean accuracy.
- **Final Testing**: The absolute final model is evaluated on the held-out Test Set.
  - *Quantitative*: Calculate Accuracy, Precision, Recall, F1 Score, and ROC AUC.
  - *Qualitative*: Visual inspection of the Confusion Matrix and analise specific misclassified examples to understand semantic failures.

Approach: Hyperparameter Search Space

1- Feature Descriptors: Evaluated SIFT, ORB, and AKAZE to find the most robust extractor.

2- Number of features: Tested limits on maximum features per image to balance density and noise.

3- Dense SIFT: Tuned step_size and scale_factor to balance resolution vs. compute load.

4- Vocabulary: Tuned Codebook Size (K) for K-Means to find the optimal "dictionary" size.

5- Dimensionality reduction: Tested PCA to reduce descriptor dimensions without losing critical variance.

6- Normalization and scaling: Tested L1/L2 Normalization and Min-Max/Robust Scaling for statistical comparability.

7- Spatial Pyramids (SPM): Tested levels L=0,1,2 to assess the impact of spatial layout.

8- Classifiers:

- *Logistic Regression*: Tuned Regularization (C) to test the efficacy of a linear baseline.
- *SVM*: Compared Linear, RBF, and Histogram Intersection kernels with distinct C values.
- *SVM-RBF*: Tweaked Gamma (γ) parameter.

Feature Descriptors

Hypothesis: ***"SIFT will outperform binary descriptors due to superior texture handling."***

- *Rationale:* Scene classification relies on global texture (e.g., grass vs. mountains), not just specific interest points. Binary descriptors often lose this fine-grained detail.

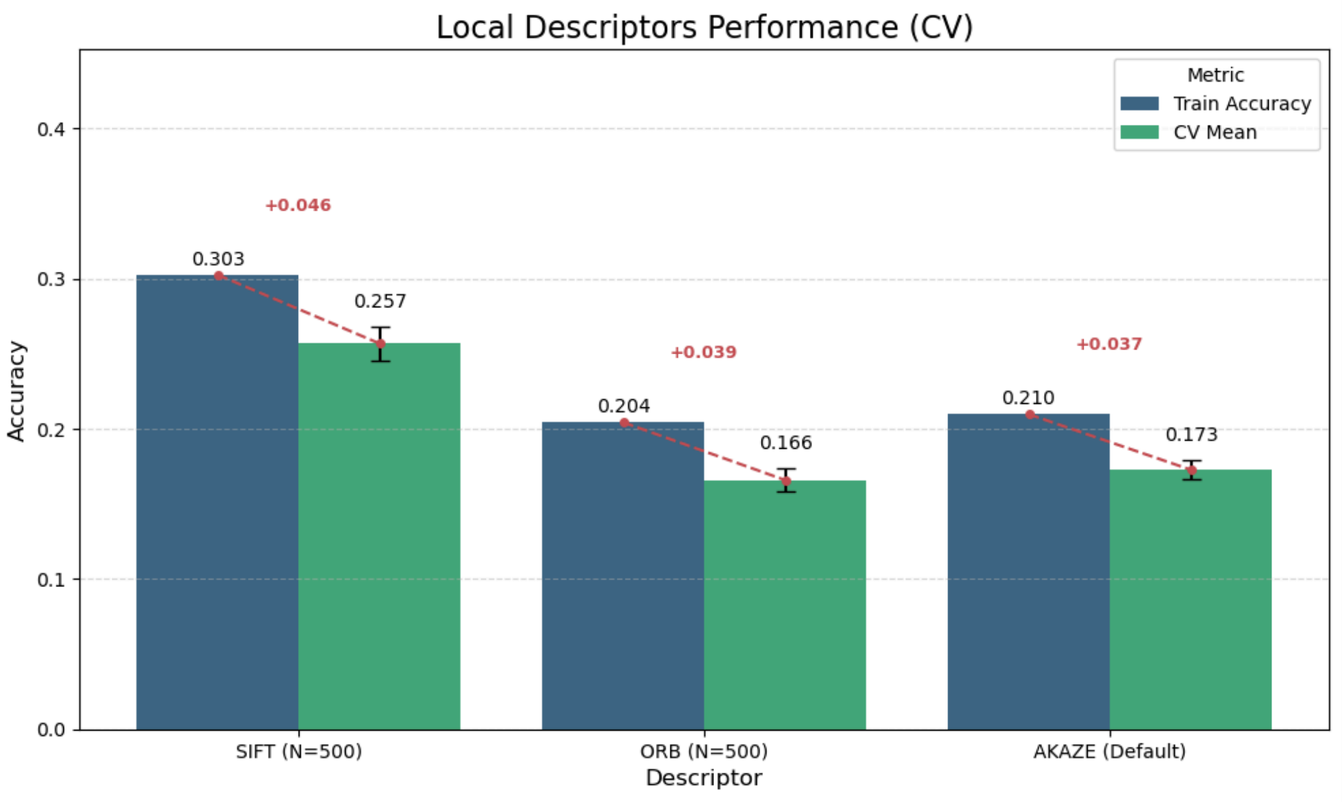
Results: ***Validated***

- SIFT achieved the highest accuracy, effectively capturing distinctive scene textures.
- *Trade-off*: SIFT is computationally heavier than ORB.

Decision: Prioritizing accuracy over speed, we selected SIFT as the fixed descriptor for all subsequent experiments.

Experimental Setup:

- *Baseline*: Codebook k=50, capped at 500 features per image.
- *Comparison*: SIFT (Gradient-based) vs. ORB/AKAZE (Binary).



Number of Features

Hypothesis: ***"Increasing feature count improves accuracy by reducing histogram sparsity, up to a saturation point."***

- *Rationale*: To correctly classify a scene, the visual word histogram must be statistically representative. If n_features is too low, the histogram is too sparse to be discriminative. If too high, we expect diminishing returns as we capture noise rather than structure.

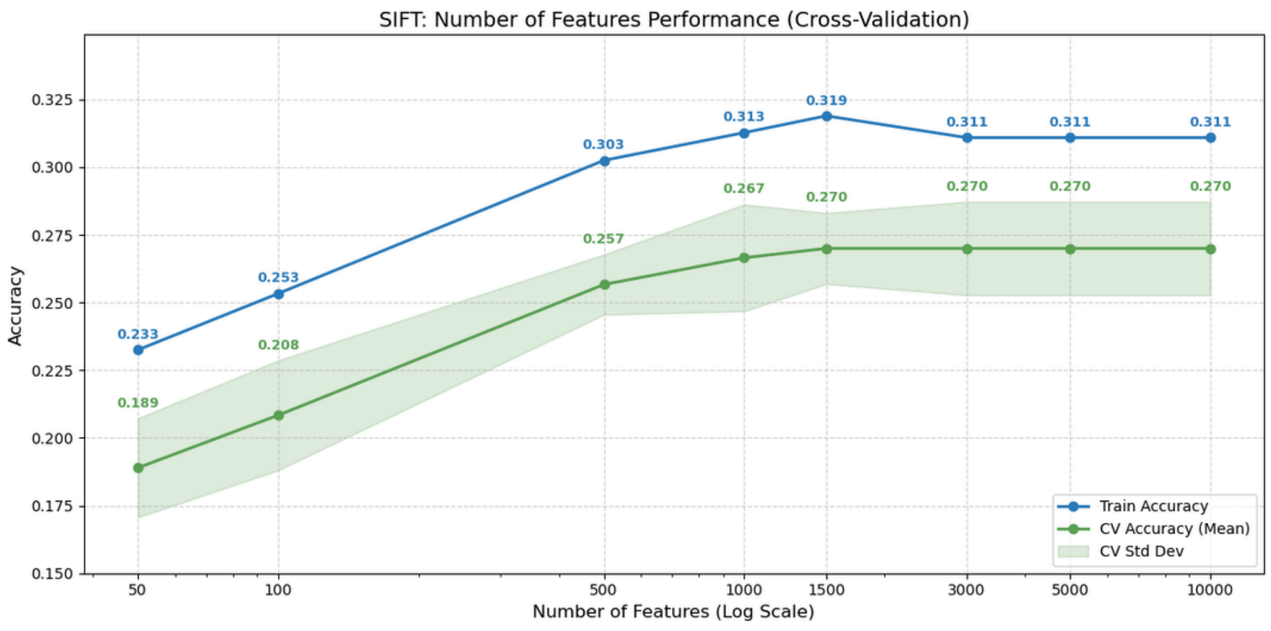
Results:***Validated***

- Accuracy rises consistently and plateaus at ~1000.
- *Analysis*: The specific scenes in the dataset rarely contain more than 1000 distinct, high-contrast keypoints. Setting limits higher (e.g., 5000) adds no new information.

Decision: We fixed n_features = 1000 to maximize information density without computational waste.

Experimental Setup:

- *Variable*: Maximum SIFT features retained per image (n_features: 50 to 10000).
- *Constant*: Codebook Size K=50.



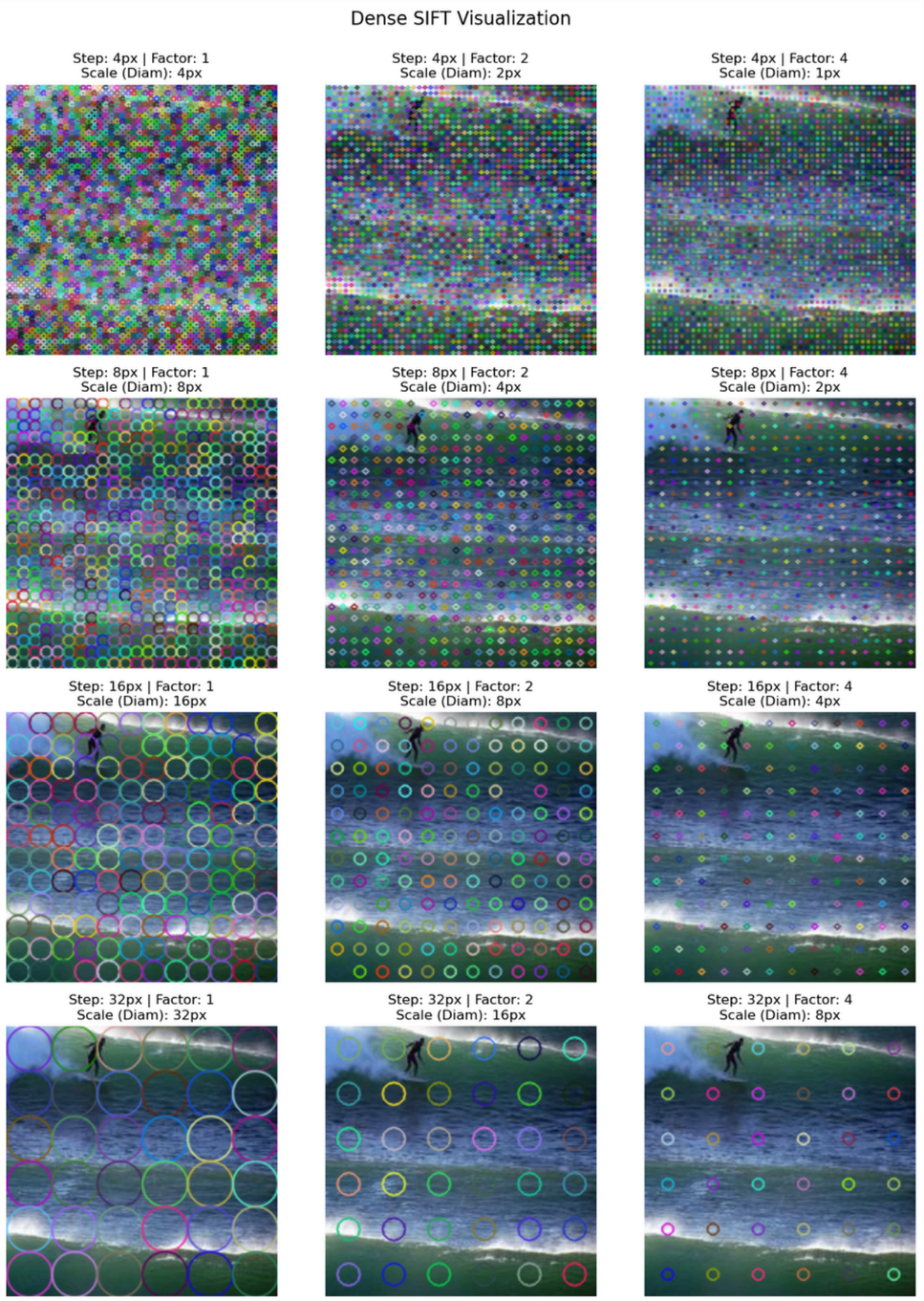
Role of Descriptors: Local descriptors are the fundamental building blocks of the BoVW model. They transform raw pixels into invariant vectors.

Dense SIFT

Concept: Regular Grid Sampling Instead of detecting "interesting" points, we sample features on a fixed grid to capture uniform textures (sky, road) often missed by standard SIFT.

Key Parameters:

- *Step Size*: Distance (in pixels) between consecutive grid centers.
  - Effect: Smaller steps = Denser grid (more features).
- *Scale*: The radius of the feature descriptor.
  - Constraint: Defined as a fraction of the Step Size to control overlap.



Dense SIFT

Hypothesis: ***"Dense SIFT will outperform Keypoint-SIFT, and denser grids (smaller Step Size) will yield the highest accuracy."***

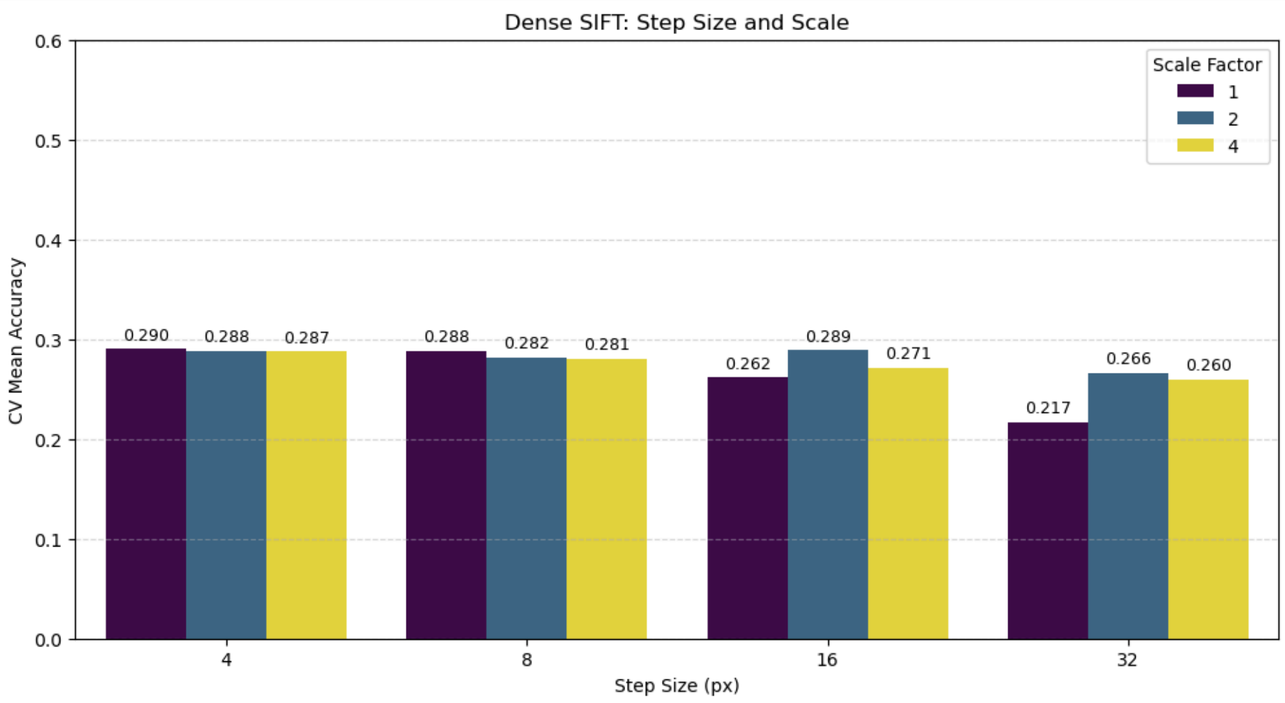
- *Rationale*: Scene classification relies on global uniform textures (sky, grass, road). Standard keypoint detectors focus only on high-contrast corners, often missing this context. Denser grids ensure total coverage.

Results: ***Partially validated***

- *Dense vs. Keypoint*: Dense SIFT consistently outperforms the standard Keypoint baseline, confirming grid sampling is superior for scenes.
- *Saturation*: Steps 4, 8, and 16 yielded statistically similar results. Sampling every 4 pixels adds redundancy rather than new information.
- *Under-sampling*: Step 32 is too sparse to capture necessary texture details, causing accuracy drop.

Experimental Setup:

- *Variables*: Step Sizes [4, 8, 16, 32] and Scale Factors [1, 2, 4].
- *Constant*: Codebook Size k=50.



Dense SIFT

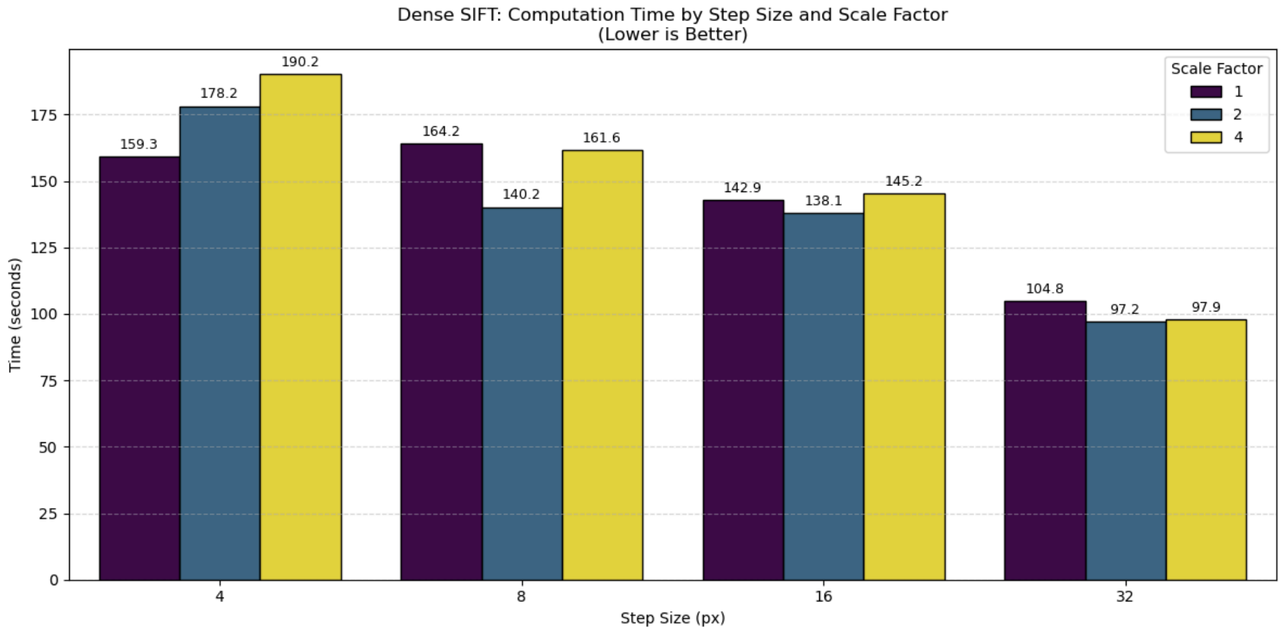
Analysis: As shown in the time plot, the computational cost grows as the step size decreases.

- *Step Size 4*: Extremely expensive (high execution time) with only marginal accuracy gains over Step 8.
- *Step Size 8*: Offers a balanced compromise between feature density and processing speed.

Decision: For subsequent experiments, we will proceed with two configurations:

1. *Step Size 8 (Scale Factor 1)*: Selected for high accuracy.
2. *Step Size 16 (Scale Factor 2)*: Selected for efficiency.

Reasoning: These configurations provide a robust accuracy baseline while keeping the pipeline computationally feasible for more complex downstream tasks like PCA or Spatial Pyramids.



Role of Scale: critical determinant of performance. Unlike standard SIFT, which automatically estimates keypoint size (scale invariance), Dense SIFT operates at fixed, predefined scales. This means the feature size must be explicitly matched to the texture granularity of the dataset to avoid capturing noise or missing details.

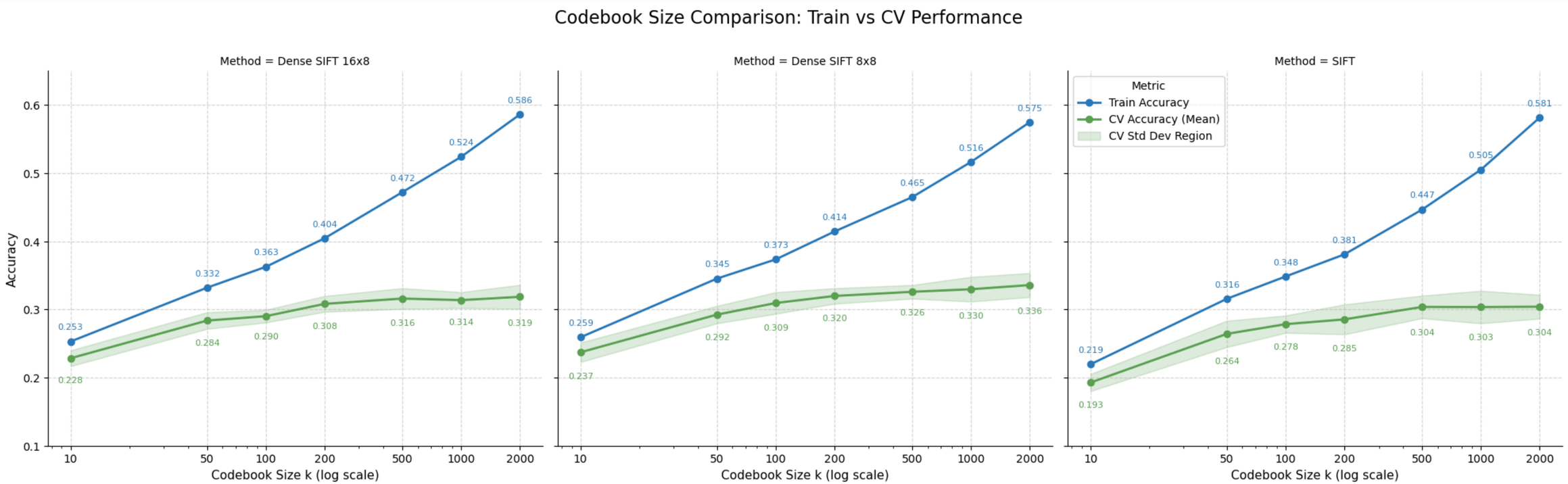
Codebook Size

Hypothesis: ***"Dense SIFT (Step 8) will outperform others as k increases, as it requires a larger vocabulary to map its massive feature count ."***

- *Rationale*: Since the Step 8 configuration generates a significantly higher volume of descriptors, it requires a larger "dictionary" to effectively quantize the feature space without losing information. We expect Baseline SIFT to saturate earlier.

Experimental Setup:We evaluate the system's performance by varying the vocabulary size (k) across three distinct feature extraction configurations:

1. *Baseline SIFT*: (n_features=1000)
2. *Dense SIFT A*: (step_size=16, scale_factor=2)
3. *Dense SIFT B*: (step_size=8, scale_factor=1)



Codebook Size

Results & Analysis: ***Validated***

- Dense SIFT (Step 8) consistently yields better Cross-Validation (CV) accuracy compared to Step 16 or regular SIFT. The denser sampling captures more fine-grained details necessary for discriminating between scene categories.

Final Decision: We select Dense SIFT (Step 8, Scale Factor 1) with a codebook size of k=200

- *Overfitting Avoidance*: k=200 represents the "sweet spot." Increasing k further to 500 or 1000 results in significant overfitting, where the model memorizes the training data (high train accuracy) without generalizing better to unseen data (stable CV accuracy).
- *Efficiency*: k=200 provides the most competitive accuracy before the computational cost and risk of overfitting surpasses the marginal performance gains.

Note: We will retain standard SIFT with k=200 as a comparative baseline for the subsequent experiments.

The Impact of Codebook Size (k):

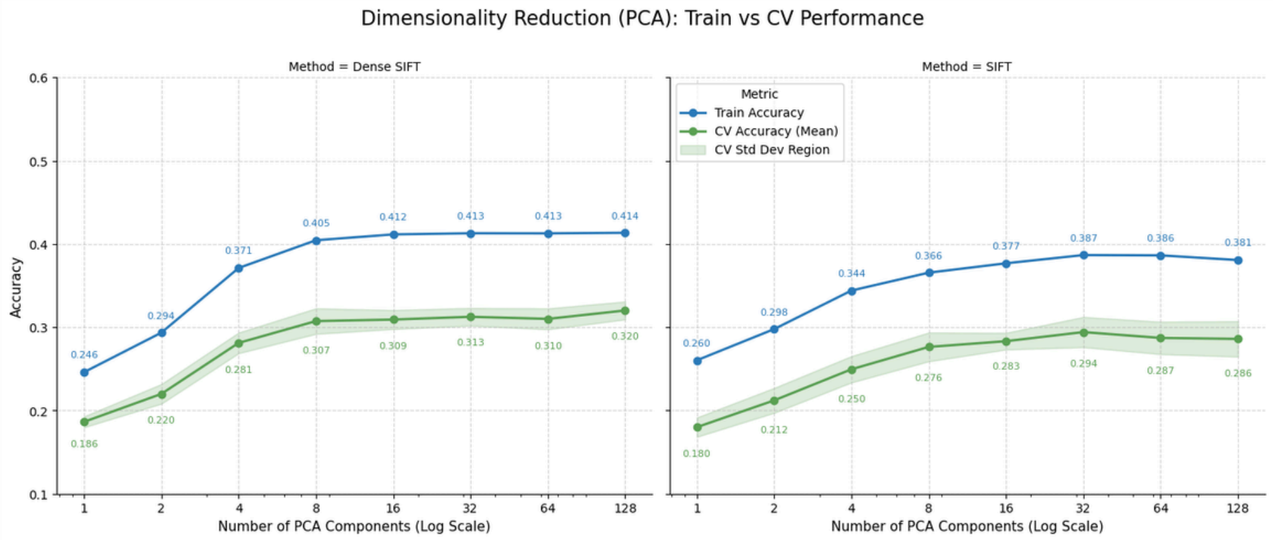
- *Small k (Underfitting)*: Visual words become too generic. Distinct visual features (e.g., a wheel vs. a clock) are forced into the same cluster, losing discriminative power.
- *Large k (Overfitting)*: Creates a very high-dimensional histogram. If k is too large relative to the feature count, the system clusters noise, leading to sparse histograms and overfitting to training data.

Dimensionality Reduction

Hypothesis: ***"PCA filters noise and redundancy, potentially improving performance. However, we anticipate a 'tipping point' where excessive reduction discards essential structure, causing a drastic drop in accuracy."***

Experimental Setup:

- *Method*: Principal Component Analysis (PCA) applied to descriptors before K-Means clustering.
- Standard SIFT vs. Dense SIFT.
- *Components:* [16, 32, 64, 128].



Results & Analysis:

- *Standard SIFT (Sparse)*: ***Validated***. Accuracy peaks at 32 components (0.294) vs. baseline (0.285).
  - Sparse keypoints often overlap; PCA effectively removes this redundancy and noise, resulting in a cleaner signal.
- *Dense SIFT (Grid)*: ***Refuted***. Performance stabilizes at 8-D but never outperforms the raw 200-D input.
  - Dense descriptors capture distinct texture across a grid. This data is less redundant than overlapping keypoints, so reducing dimensions deletes valid information.

Decision:

- *Dense SIFT*: Proceed with full 200-D (No PCA) to maximize texture retention.
- *Standard SIFT*: Apply PCA (32-D) for the comparative baseline.

Normalization and Scaling

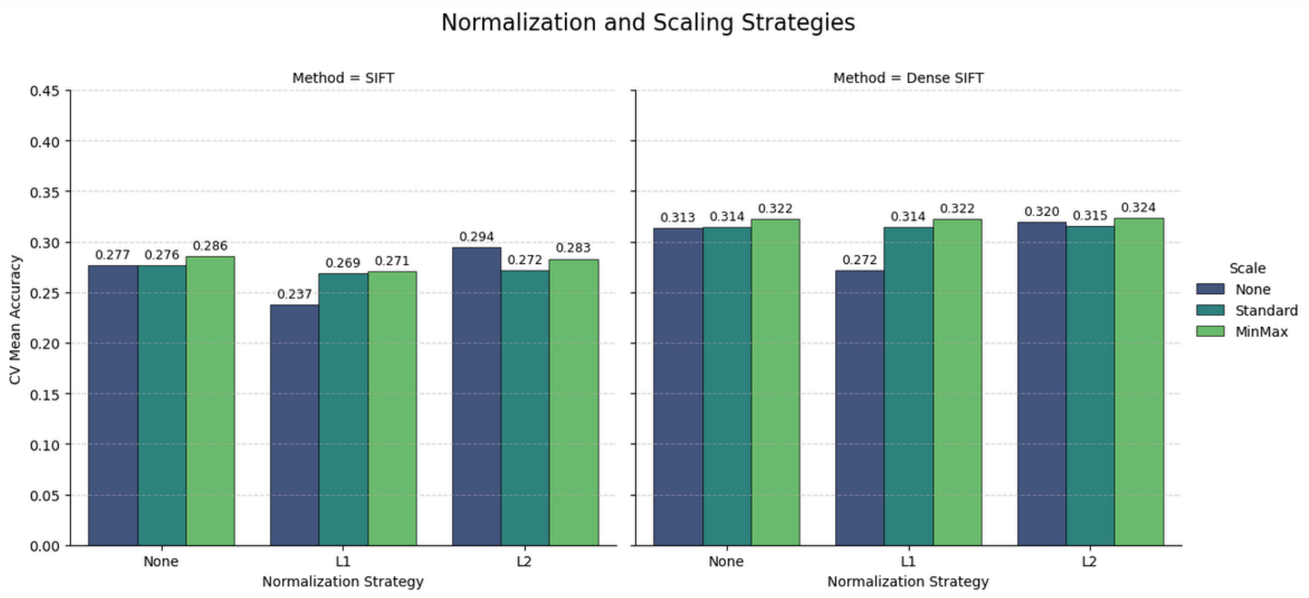
Hypothesis: ***"Raw histogram counts are biased by image complexity. Normalization ensures that classification relies on the distribution's 'shape' rather than its raw 'magnitude'."***

Experimental Setup:

- *Normalization*: None, L1 (Manhattan), L2 (Euclidean).
- *Scaling*: None, Standard Scaling (z-score), MinMax Scaling (0-1 range).

Results & Analysis: ***Validated***

- Standard SIFT (Sparse): L2 Normalization only is best (0.294).
  - SIFT histograms are sparse. Aggressive scaling (like MinMax) amplifies noise in empty bins, distorting the data structure.
- Dense SIFT (Grid): L2 Norm + MinMax Scaling is best (0.324).
  - Dense sampling creates fully populated histograms. Scaling to [0, 1] prevents high-frequency visual words (common textures) from dominating the classifier.



Decision:

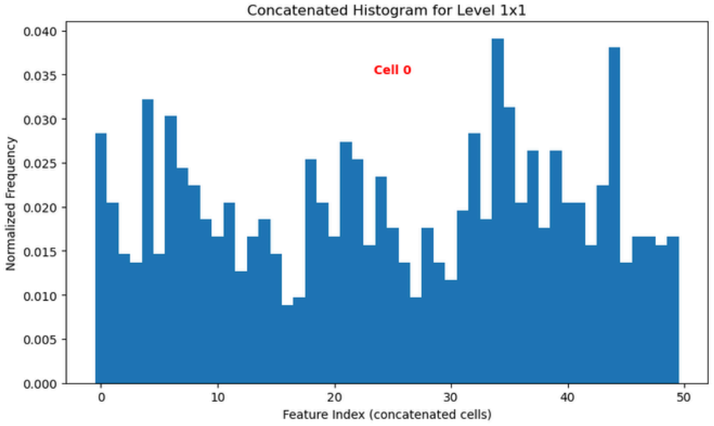
- *Standard SIFT*: Apply L2 Normalization (No Scaling).
- *Dense SIFT*: Apply L2 Normalization + MinMax Scaling.

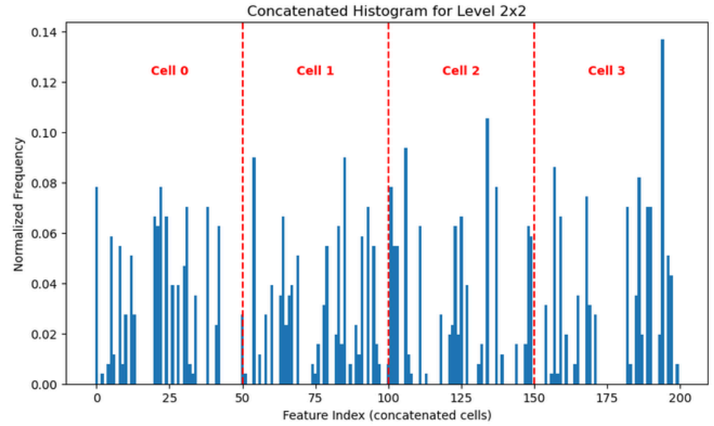
Spatial Pyramid

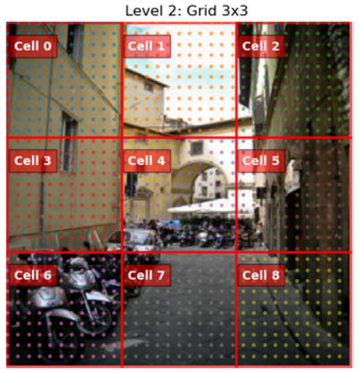
Hypothesis: ***"Standard BoVW is 'orderless' and discards all spatial context. We hypothesize that partitioning the image into increasingly fine sub-regions will capture scene layout (e.g., sky at the top, ground at the bottom), thereby improving discrimination between texture-similar classes."***

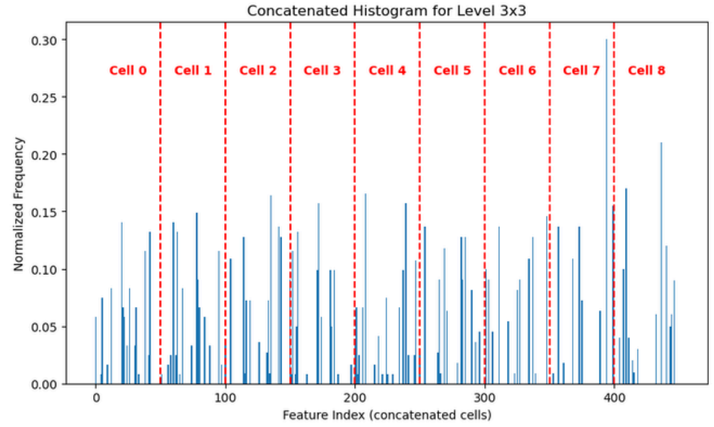
Experimental Setup:

- *Method*: We divide the image into grids at three levels and concatenate the histograms from all regions into a single feature vector.
  - Level 0: 1x1 (Global image).
  - Level 1: 2x2 (4 regions). Total dimension: 1 + 4 = 5 x original vocabulary size.
  - Level 2: 4x4 (16 regions). Total dimension: 1 + 4 + 16 = 21 x original vocabulary size.
- *Tested Configurations*:
  a. Baseline SIFT: (k=200, PCA-32).
  b. Dense SIFT: (Step 8, k=200, MinMax Scaling).









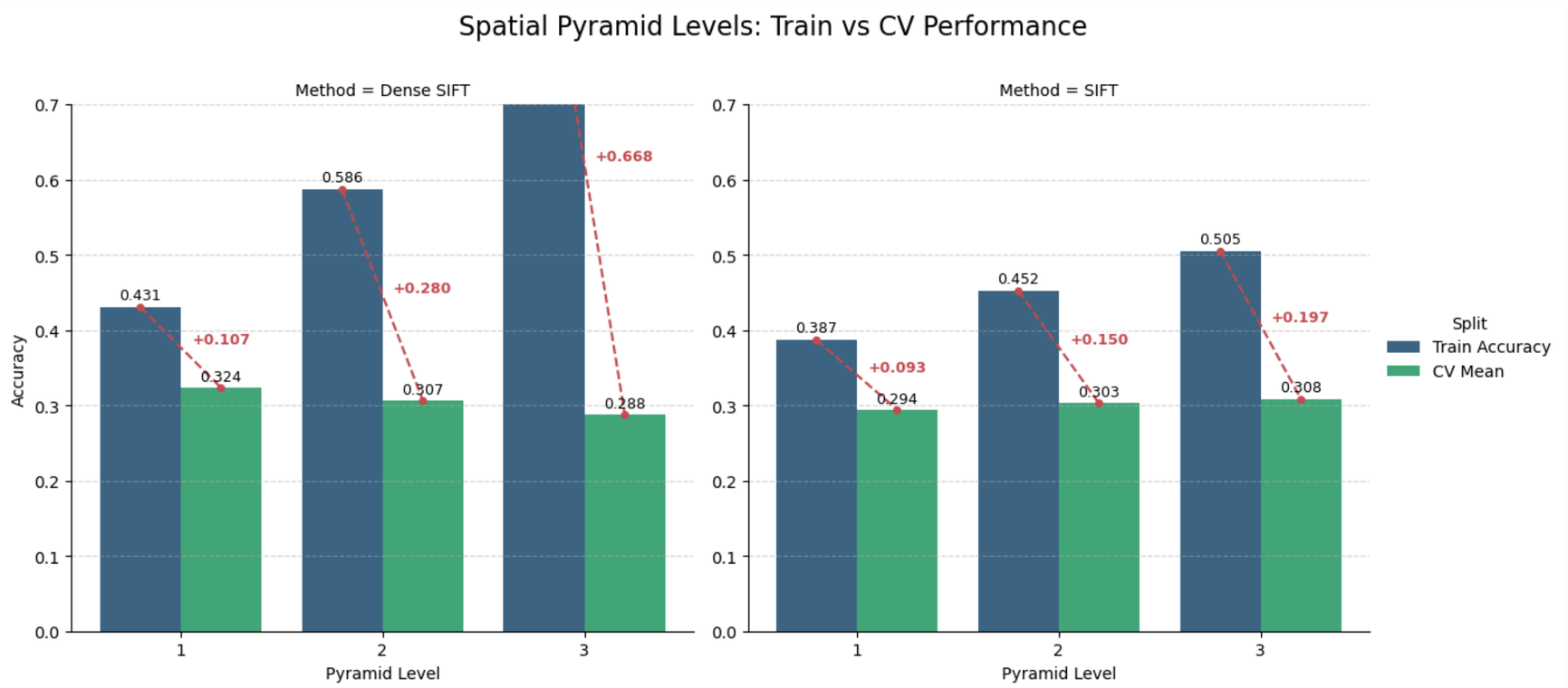
Spatial Pyramid

Quantitative Results:

- *SIFT*: Observed a slight improvement (~3% increase) in CV Accuracy.
- *Dense SIFT*: CV Accuracy stagnated or decreased, despite Training Accuracy rising sharply.

Analysis:

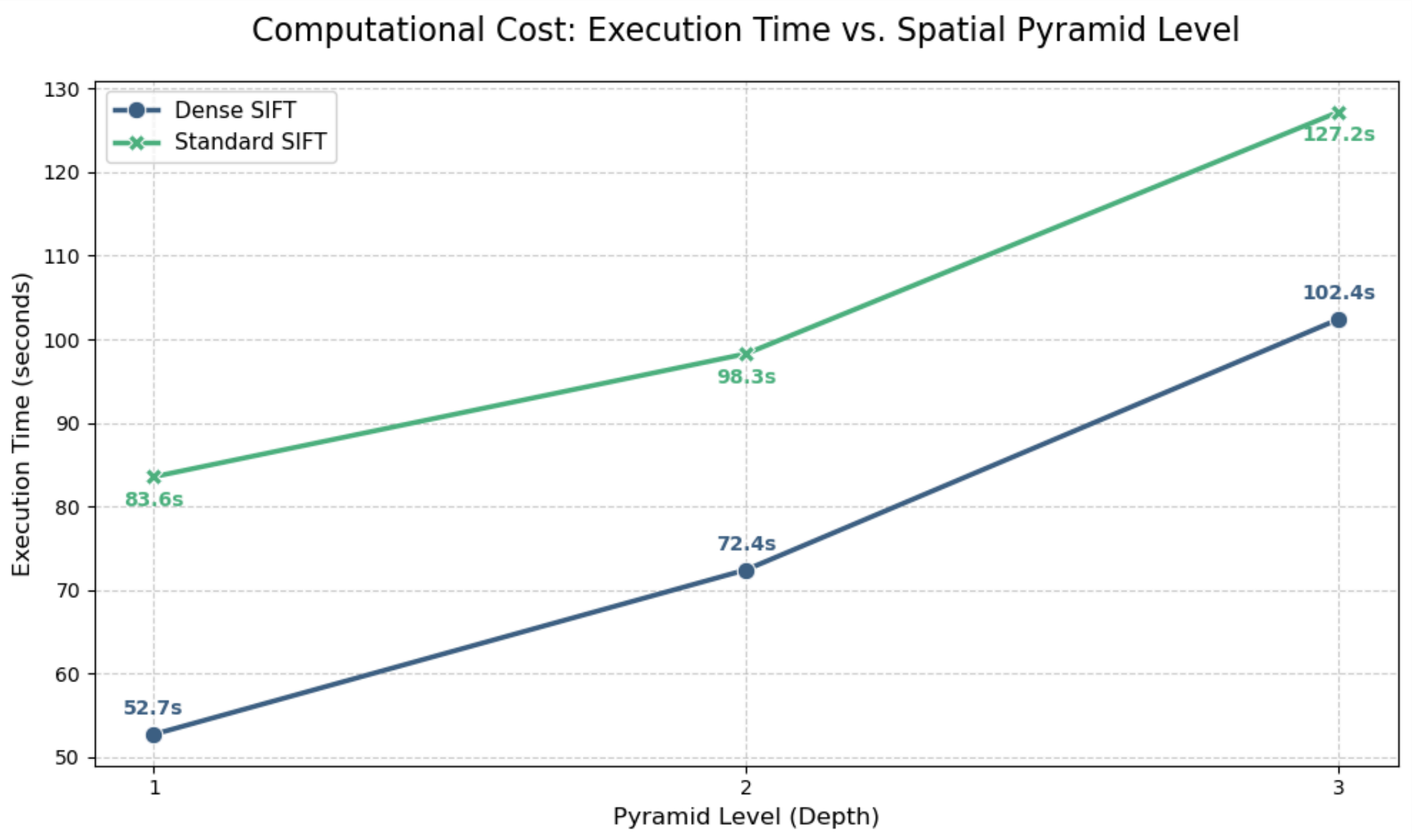
- *Curse of Dimensionality*: For a vocabulary k=200, Level 2 results in a feature vector of 4,200 dimensions (200 x 21).
- *Generalization Gap*: The massive increase in dimensionality allowed the classifier to memorize the training data (overfitting) rather than learning generalizable spatial rules. The Dense SIFT configuration, already rich in information, saturated the model's capacity given the dataset size.



Spatial Pyramid

Execution Time Analysis:

- *Computational Bottleneck*: Execution time and memory usage grow linearly with the concatenated feature vector significantly slowing down training without yielding proportional accuracy gains.



Decision: We will DISCARD Spatial Pyramids for the final system.

1. *Overfitting*: The lack of improvement on the held-out set indicates the model is fitting noise as levels added.
2. *Efficiency*: The marginal gains (in Standard SIFT) do not justify the 21 x increase in feature dimensionality.

Next Step: We proceed with the Global (Level 0) Dense SIFT representation, which offers the best balance of accuracy, generalization, and efficiency for this specific dataset to test the classifiers.

Classifiers - Logistic Regression Regularization

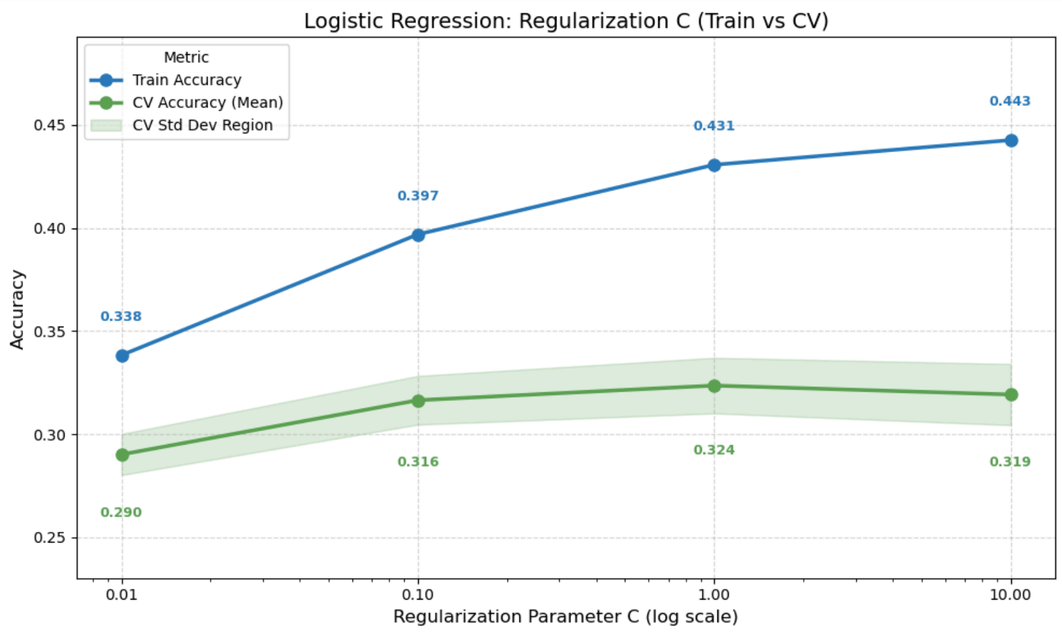
Hypothesis: ***"C governs the bias-variance trade-off. We hypothesize that extreme values will degrade performance (underfitting at low C, overfitting at high C), while an intermediate value will maximize generalization."***

Results & Analysis: ***Validated***

- *Underfitting (C=0.01)*: Strong penalization suppresses weights too heavily, resulting in poor accuracy on both Train and CV sets.
- *Optimal Range (C=0.1 and 1.0)*: This range represents the stable "sweet spot." Both values show solid convergence, with manageable gaps between Training and CV scores.
- *Overfitting (C=10.0)*: Training accuracy climbs significantly (0.443), but CV accuracy drops (0.319). The widening gap indicates the model is overfitting to training noise.

Experimental Setup:

- *Fixed Pipeline*: Dense SIFT, k=200, L2, MinMax Scale.
- *Model*: Logistic Regression (Linear).
- *Variable*: Regularization Parameter C



Decision: We choose C=1.0.

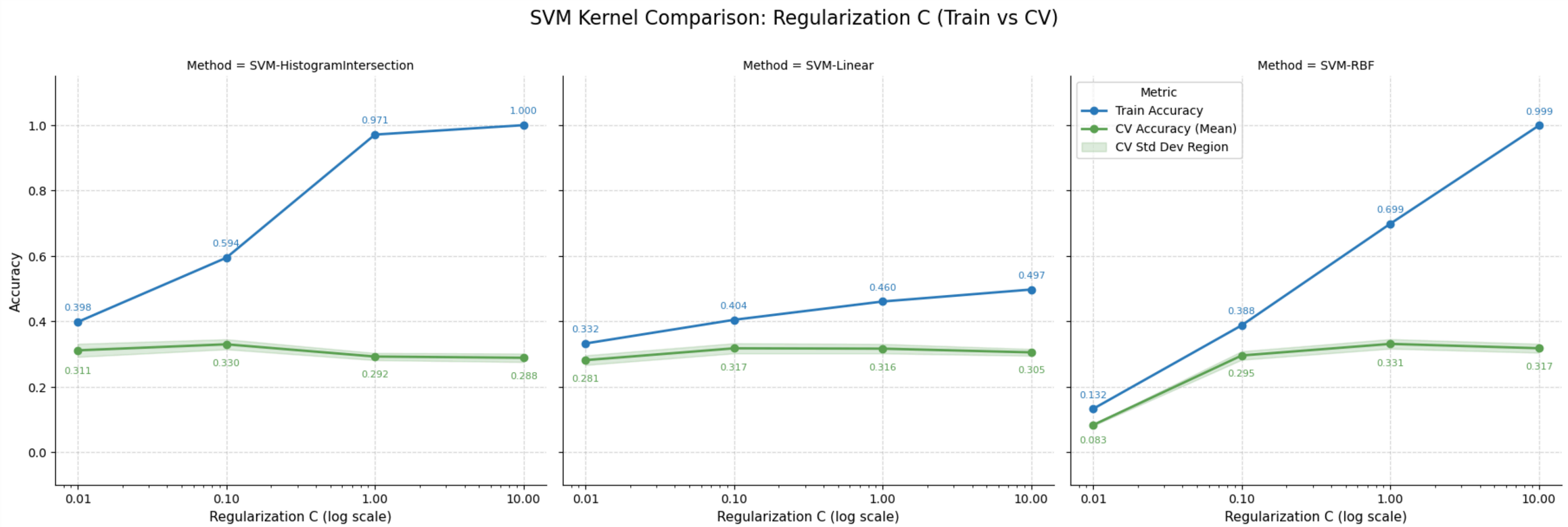
- We prioritize the peak CV accuracy (0.324) of C=1.0, accepting a marginal increase in overfitting compared to the tighter fit of C=0.1.

Classifiers - SVM Regularization

Hypothesis: ***"Non-linear kernels will outperform the linear baseline by capturing complex relationships between visual words. Specifically, the Histogram Intersection kernel is expected to be optimal, as it mathematically aligns with the nature of histogram comparison"***

Experimental Setup:

- *Fixed Pipeline*: Dense SIFT, k=200, L2, MinMax Scale.
- *Model*: SVM with Linear, RBF (Euclidean) and Histogram Intersection kernels
- *Variable*: Regularization Parameter C



Classifiers - SVM Regularization

Kernel	Best C	CV Acc.	Test Acc.	Insight
Linear	*0.1*	*0.317*	*0.404*	**Too Simple**: Linear boundaries cannot separate complex scene histograms. Fails to beat Logistic Regression baseline.
RBF	*1.0*	*0.331*	*0.699*	**Overfitted**: Highest score, but massive 37% generalization gap.
Hist. Int.	*0.1*	*0.330*	*0.594*	**Optimal**: Measures explicit bin overlap. Matches RBF peak but generalizes far better. High sensitivity to C.

Decision: SVM with Histogram Intersection (C=0.1) as the best kernel to compare against our linear baseline

- While RBF yielded a marginal numeric gain, we prioritize generalization.

Classifiers - SVM RBF Gamma

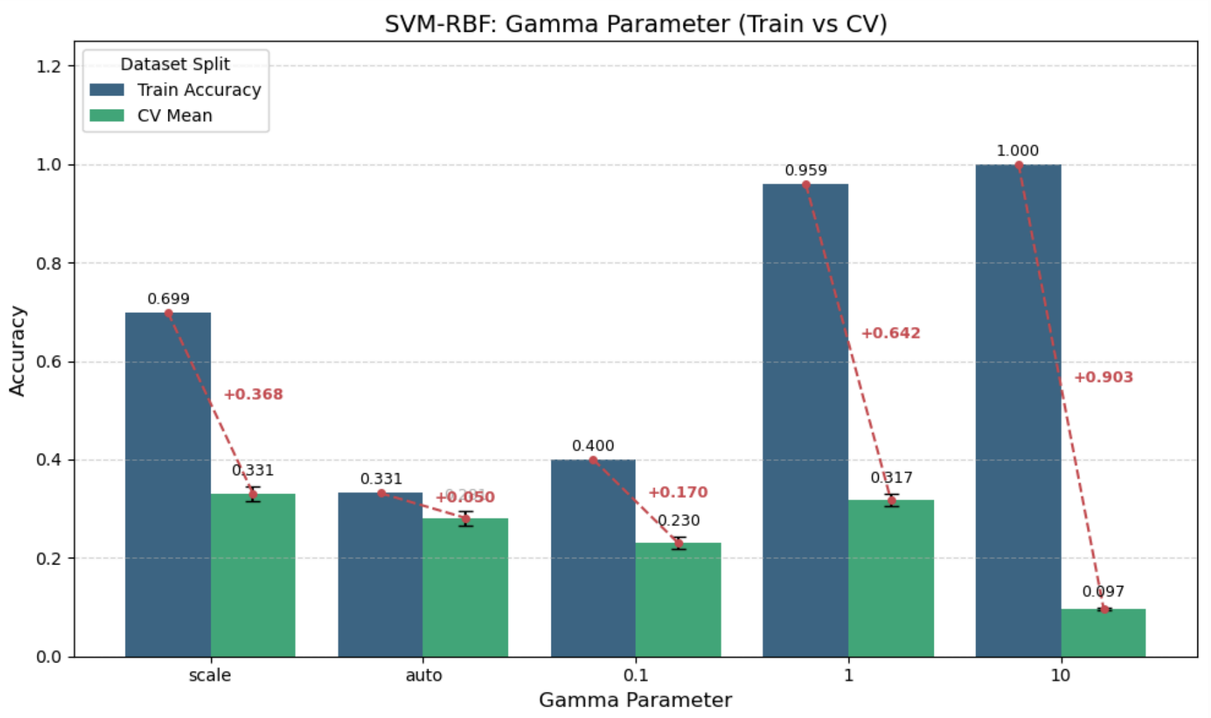
Hypothesis: ***"Gamma controls the "reach" of training points. We expect high γ to cause overfitting (tight boundaries) and low γ to underfit, while the adaptive 'scale' heuristic should offer the best balance "***

Experimental Setup:

- *Fixed Pipeline*: Dense SIFT, k=200, L2, MinMax Scale.
- *Model*: SVM (RBF Kernel) with fixed C=1.0.
- *Variable*: Gamma (γ) [scale, auto, 0.1, 1, 10].

Results & Analysis: ***Validated***

- *Optimal Setting ('scale')*: Peaks at 0.331 but remains unstable. The large gap between Training (0.699) and CV (0.331) confirms the kernel's high variance.
- *Severe Overfitting (γ >= 1)*: Model collapses into memorization. At γ=10, Training hits 100% while CV plummets to 0.097 (worse than random guessing).
- *Underfitting ('auto', 0.1)*: Performance drops; boundaries are too smooth to separate classes.



Decision: This reinforces our preference for the Histogram Intersection Kernel (C=0.1), which matches RBF's peak accuracy without this extreme sensitivity.

Final Evaluation: Test Set Performance

We conclude our study by evaluating the optimized pipeline (Dense SIFT, k=200, L2 + MinMax Scaling) on the Test Set for both of our selected classifiers.

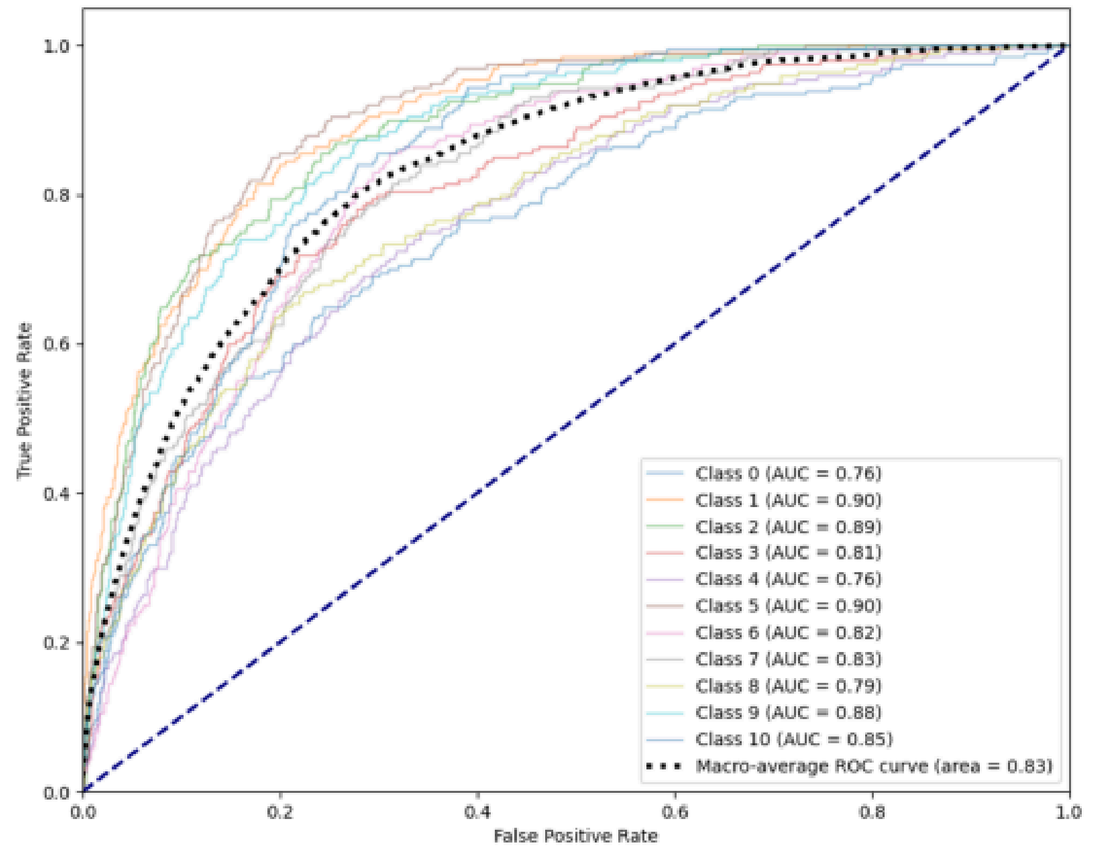
- **Logistic Regression (C=1.0)**
- **SVM with Histogram Intersection Kernel (C=0.1)**

Key Metrics	Accuracy	AUC
Baseline (Random)	9% approx.	-
Logistic Regression	39.64%	0.835
SVM Histogram Intersection	38.18%	0.833

ROC Curve Analysis:

- *Top Performers (AUC >= 0.90)*: Classes 1, 2, and 5 show high sensitivity and specificity, indicating clear feature separability.
- *Weaker Performers (AUC <= 0.76)*: Classes 0 and 4 hug the diagonal, suggesting significant feature overlap with other categories.

Linear Regression ROC



Very similar ROC curve for the SVM with Hist. Int

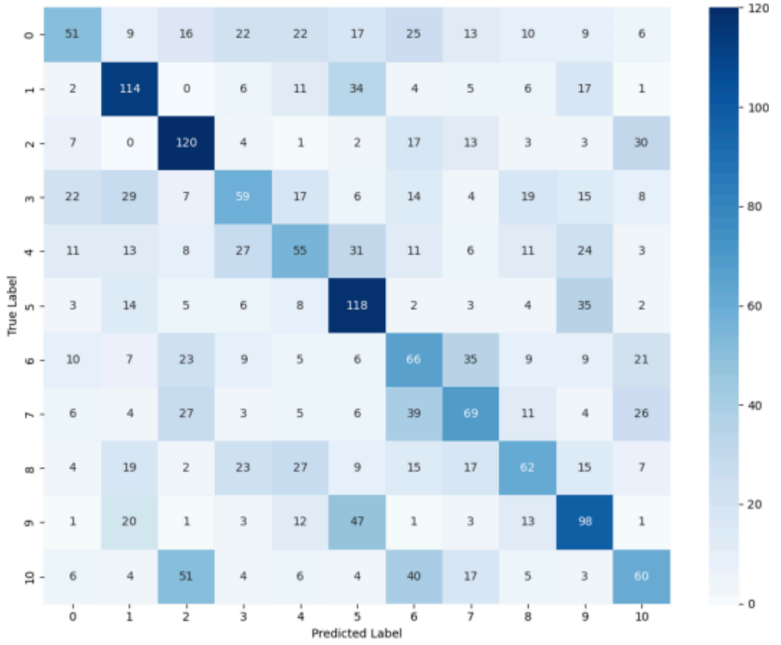
Error Analysis: Confusion Matrix Insights

We will make insights based on both confusion matrix since both are pretty similar.

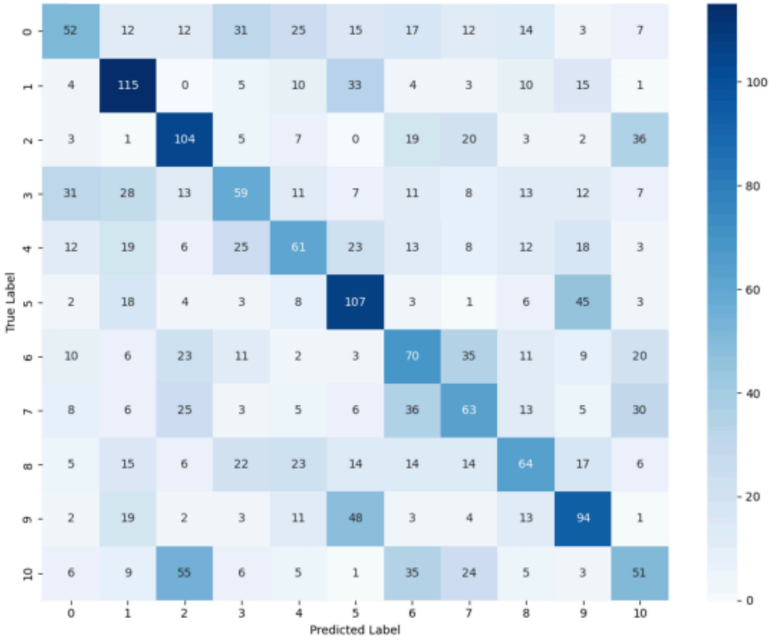
Key Points:

- **Diagonal Dominance:** The matrix is diagonal-dominant, confirming correct classification is the most frequent outcome.
- **The "Sink" Class (Class 6):** Class 6 acts as a 'gravitational sink,' frequently absorbing instances from Class 7 and 10.
- **Specific Confusion:** We observe a major cluster of errors between Class 9 (Water/Ice) and Class 5 (Mountains/Desert) as well as Class 2 (Home/Hotel) and Class 10 (Workplace)

LR Confusion Matrix



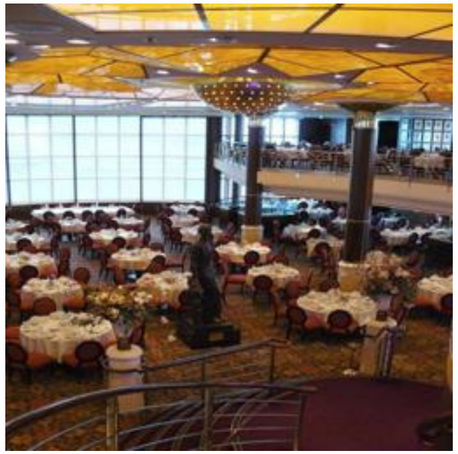
SVM Confusion Matrix



Visual Example: Why the Model Fails

"Sink" Class

Confused as Class 6



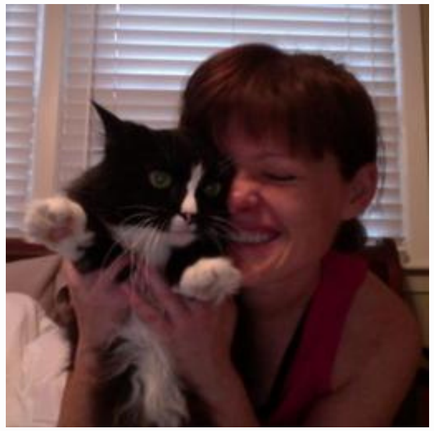
Sports and leisure
7

Shopping and dining
6

Workplace
10

Visual Example: Why the Model Fails

Home or Hotel
2



Specific Confusion

Workplace
10

Conclusions

1. The "Semantic Gap" of Local Descriptors

- *Limitation*: Bag of Visual Words with local descriptors (SIFT) is fundamentally designed for Image Retrieval (matching specific points), not Scene Classification.
- *Problem*: The descriptors rely on low-level texture matching, which fails to capture the high variance within semantic classes (e.g., a "Workplace" and a "Home" often share identical textures like furniture).

Final Best Configuration:

- Descriptor: Dense SIFT (Step Size 8px, Scale Factor 1)
- Codebook: k=200.
- Preprocessing: L2 Normalization + MinMax Scaling.
- Classifier: Logistic Regression (C=1.0)
- Final Performance: 39.64% Accuracy / 0.835 AUC.

2. Performance Verdict

- *Relative Success*: Despite these inherent limitations, our optimized system achieved ~40% accuracy compared to a ~9% random baseline. We successfully maximized the signal extraction possible with handcrafted features.

3. Future Work: The Deep Learning Necessity

- *Solution*: To achieve high accuracy and overcome intra-class variance, Convolutional Neural Networks (CNNs) are required. Unlike BoVW, CNNs can learn hierarchical, global semantic features necessary to distinguish complex scenes.