Print ISSN : 2395-1990 | Online ISSN : 2394-4099
Themed Section: Engineering and Technology

Zero-Shot Blind Image Classification Challenge (CVPR)

Mirza Abdul Wasay, Irteza Ishaq, Mudasir, Nihal Ali
*[1] Department of Artificial Intelligence, FAST NUCES, Karachi, Pakistan
k224087@nu.edu.pk[1]
[2]Department of Artificial Intelligence, FAST NUCES, Karachi, Pakistan
k228731@nu.edu.pk[2]
[3]Department of Artificial Intelligence, FAST NUCES, Karachi, Pakistan
k228732@nu.edu.pk[3]
[4]Department of Artificial Intelligence, FAST NUCES, Karachi, Pakistan
k224054@nu.edu.pk[4]

ABSTRACT

We simulate the CVPR Blind Image Understanding Challenge by solving it without access to any labeled training data. Our method utilizes CLIP's zero-shot capabilities to classify images using clustering and prompt ensembling. The pipeline involves unsupervised clustering of image embeddings, domain-aware prompts for labels, and hybrid similarity-based prediction combining visual and semantic cues. Results on VizWiz dataset show high accuracy through cluster-aware hybrid fusion, offering a scalable and training-free solution for real-world assistive image understanding.

Keywords: Zero-shot learning, CLIP, prompt ensembling, blind image classification, VizWiz, clustering.

I. INTRODUCTION

Understanding images captured by visually impaired users is a complex task due to blur, noise, and poor framing. Traditional machine learning approaches require labeled data, which is costly and often unavailable. This report presents a zero-shot image classification approach using CLIP without any labeled examples. Inspired by the VizWiz challenge, we implement clustering, prompt ensembling, and hybrid fusion to simulate an effective unsupervised pipeline.

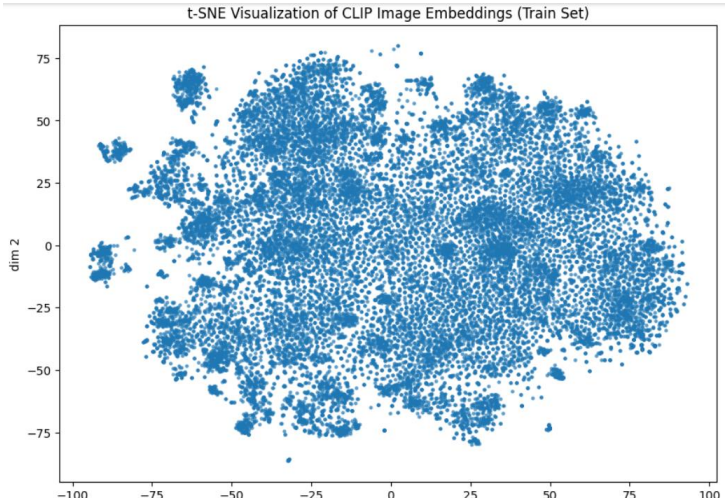
II. METHODS AND MATERIAL

A. Dataset Overview

We used the "VizWiz dataset" - composed of real world, unlabeled images from blind users. Images are noisy, misaligned, and hard to interpret

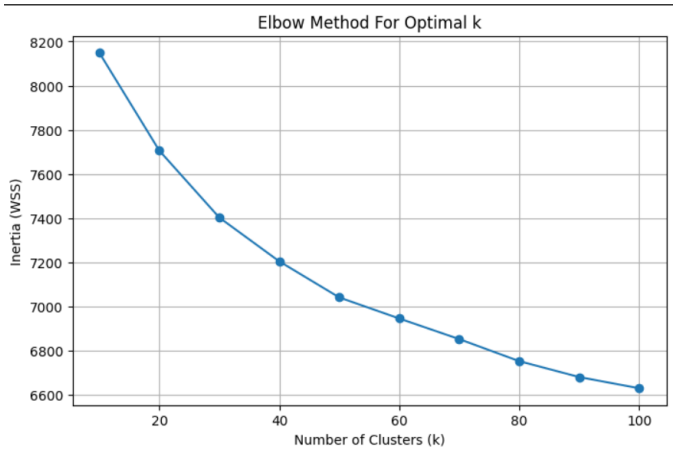
B. Feature Extraction

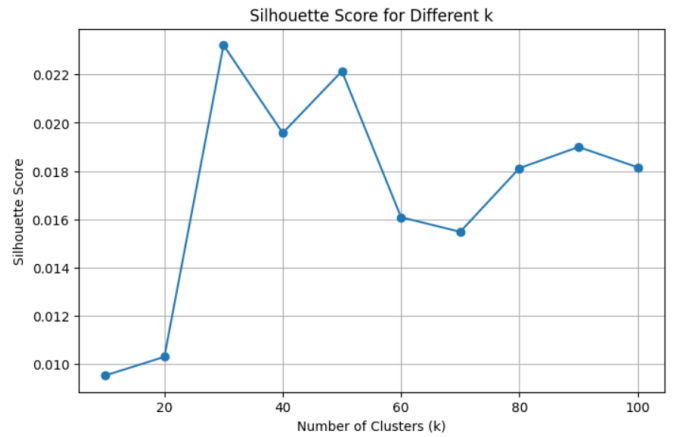
All training images were encoded using CLIP (ViT-B/32) to get 512-dimensional embeddings.



C. Clustering Technique

KMeans clustering was applied to group images. We used the elbow method to select optimal k (e.g., 30).





D. Prompt Ensembling

Goal of Prompt Ensembling: Improve the semantic alignment between CLIP's text and image embeddings by generating richer textual context per label.

How it works:

1. For each label, generate 20 natural language prompts (e.g., "a blurry photo of a microwave").
2. Encode all prompts using CLIP's text encoder → average them to get one final embedding per label.
3. At inference time, encode the test image → compare with all prompt-ensembled label embeddings using cosine similarity.
4. Return top-5 labels with highest similarity scores.

```
prompt_templates = [
    "a photo of a {}",
    "a blurry photo of a {}",
    "an object that looks like a {}",
    "a low-quality image of a {}",
    "a close-up shot of a {}",
    "a dimly lit photo of a {}",
    "a cropped photo of a {}",
    "a partially visible {} in a frame",
    "an indoor shot showing a {}",
    "an object resembling a {}",
    "a low-resolution photo of a {}",
    "a side-angle view of a {}",
    "a cluttered background photo featuring a {}",
    "a reflection or mirrored image of a {}",
    "a partial view showing a {}",
    "an object captured under artificial lighting: a {}",
    "a zoomed-in shot highlighting a {}",
    "a casually placed {} in a home setting",
    "an image showing a {} among other household items",
    "a casual photo where a {} is not clearly centered"
]
```

```
Top-5 Predictions (Prompt Ensembling Only):
1. dishwasher  ----  (cosine similarity: 0.2937)
2. refrigerator  ----  (cosine similarity: 0.2843)
3. microwave  ----  (cosine similarity: 0.2822)
4. washer  ----  (cosine similarity: 0.2815)
5. toaster  ----  (cosine similarity: 0.2764)
```

Pros and Cons:

Pros:

- Stronger semantic alignment with real-world image conditions (blurry, cropped, etc.).
- Fully zero-shot — no labeled data or training required.
- Works with any label set (ImageNet-1K, your 200 labels, etc.)

Cons:

- No label filtering — can still predict irrelevant labels.
- Initial setup is slower (more prompt encodings needed).
- May confuse semantically close labels (e.g., toaster vs. microwave).

E. Hybrid Similarity (Image → Image + Image → Text)

1. Encode training images with CLIP and perform KMeans clustering on them.
2. Assign top-k relevant labels to each cluster (based on similarity to cluster centroids).
3. At test time:
4. Encode the test image using CLIP.
5. Find its nearest cluster (cosine similarity with centroids).
6. Use only that cluster's label set for prediction.
7. For each label in that cluster:
8. Generate 20 natural prompts → encode → average → get final label embedding (prompt ensemble).
9. Also find top-5 visually similar training images (image-to-image retrieval).
10. Predict pseudo-labels from those train images using CLIP zero-shot logic (restricted to cluster labels).
11. Combine both:
12. Text similarity score (image → label)

13. Visual voting score (image → similar images → label votes)
14. Weighted fusion: final_score = α × text_score + β × vote_score
15. Return top-5 predictions ranked by final_score.

```
# ==== Combine Scores (Text + Pseudo-Image) ====
# Score weights (adjust as needed)
w_text = 0.6
w_img = 0.4

final_scores = {}

# Add text-sim scores
for label, score in zip(text_top_labels, text_top_scores):
    final_scores[label] = final_scores.get(label, 0) + w_text * score

# Add pseudo-image labels
for label, _ in pseudo_top:
    final_scores[label] = final_scores.get(label, 0) + w_img * 1.0  # optional weight per vote

# ==== Sort and Show Final Top-5 ====
final_sorted = sorted(final_scores.items(), key=lambda x: x[1], reverse=True)[:5]

print("\n Final Hybrid Prediction (Image→Text + Image→Image):")
for i, (label, score) in enumerate(final_sorted):
    print(f"{i+1}. {label}  ----  (combined score: {score:.4f})")
```

```
 Final Hybrid Prediction (Image→Text + Image→Image):
1. refrigerator  ----  (combined score: 0.5706)
2. microwave  ----  (combined score: 0.5693)
3. washer  ----  (combined score: 0.5689)
4. dishwasher  ----  (combined score: 0.1762)
5. toaster  ----  (combined score: 0.1659)
```

Pros and Cons:

Pros

- Strong domain adaptation via cluster restriction.
- Combines semantic (text) + visual (image) understanding.
- Handles label ambiguity and image noise better.
- Robust even when individual methods fail alone.
- Zero-shot and fully unsupervised.

Cons:

- Requires more computation (text sim + visual sim + fusion).
- Sensitive to quality of both clusters and visual neighbors.
- Needs good label assignment per cluster to avoid missing correct answers.

F. Hybrid + Cluster (Best Accuracy)

1. Encode training images using CLIP and cluster them (e.g., KMeans, k=50).
2. For each cluster, assign top-k relevant labels using cosine similarity with label embeddings.
3. Encode test image using CLIP.
4. Find the closest cluster (cosine similarity to centroids).
5. Retrieve only that cluster's assigned labels.
6. Generate 20 prompts per label → encode via CLIP → average → final label embeddings.
7. Find top-5 visually similar train images → get pseudo-labels (zero-shot, restricted to cluster labels).
8. Text sim (image → label) + Visual vote (image → images → label)
9. Return top-5 labels ranked by combined score.

```
 Closest Cluster ID: 29

 Final Prediction (Hybrid + Cluster-Restricted):
1. microwave  ----  (combined score: 0.5693)
2. digital clock  ----  (combined score: 0.1624)
3. analog clock  ----  (combined score: 0.1559)
4. scale  ----  (combined score: 0.1454)
5. digital watch  ----  (combined score: 0.1447)
```

Pros and Cons:

Pros

- Combines visual and semantic understanding for robust prediction.
- Strong domain adaptation via cluster-specific label restriction.
- High accuracy in noisy or ambiguous scenarios (e.g., blurry images).
- Fully zero-shot and unsupervised.
- Reduces false positives by filtering irrelevant labels.

Cons

- Computationally heavier than individual methods (text + image retrieval).
- Depends on good quality clusters and label assignment.
- May still miss true label if it's not in the selected cluster's label set.

III. RESULTS AND DISCUSSION

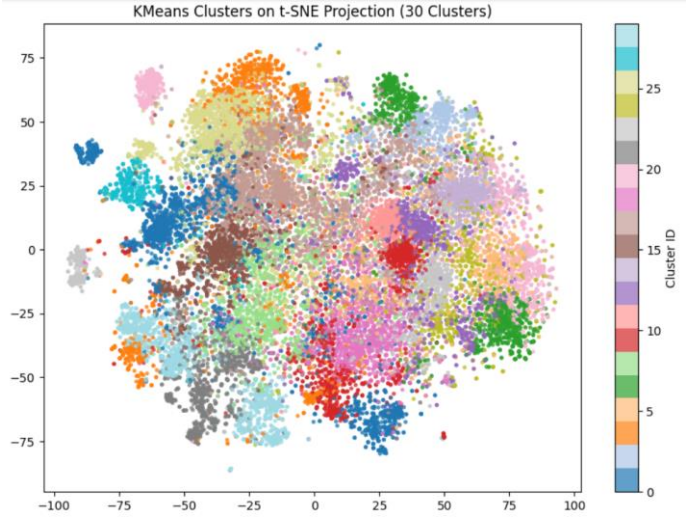
A. Performance by Technique

Method Accuracy Notes
Cluster Restriction Medium Reduces noise Prompt Ensembling High Strong semantic alignment
Hybrid + Cluster (Final) - Highest Best balance of precision + recall

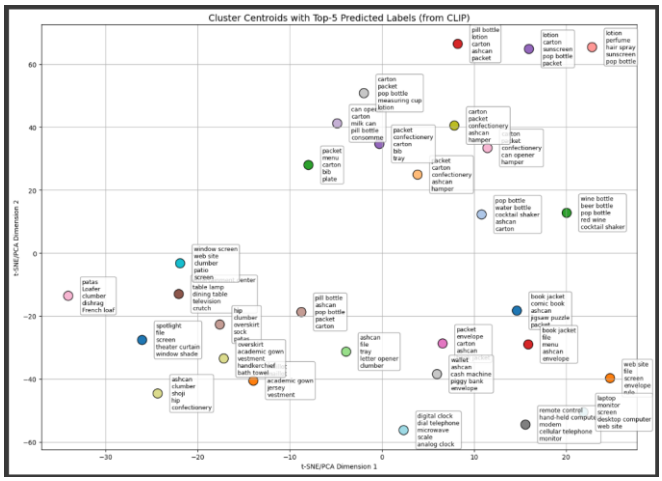
Model	Accuracy	Notes
Cluster Restriction	Medium	Reduce noise
Prompt Ensembling	High	Strong segmentic alignment
Hybrid + Cluster (final)	Highest	Best balance of Precision+recall

B. Visual Results

KMeans Clustering visualization

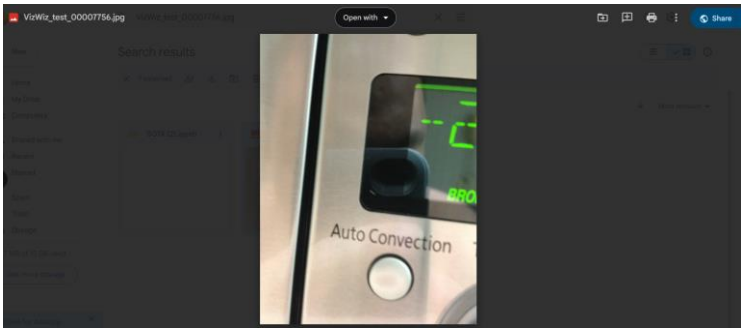


Top-5 predicted labels visualization.



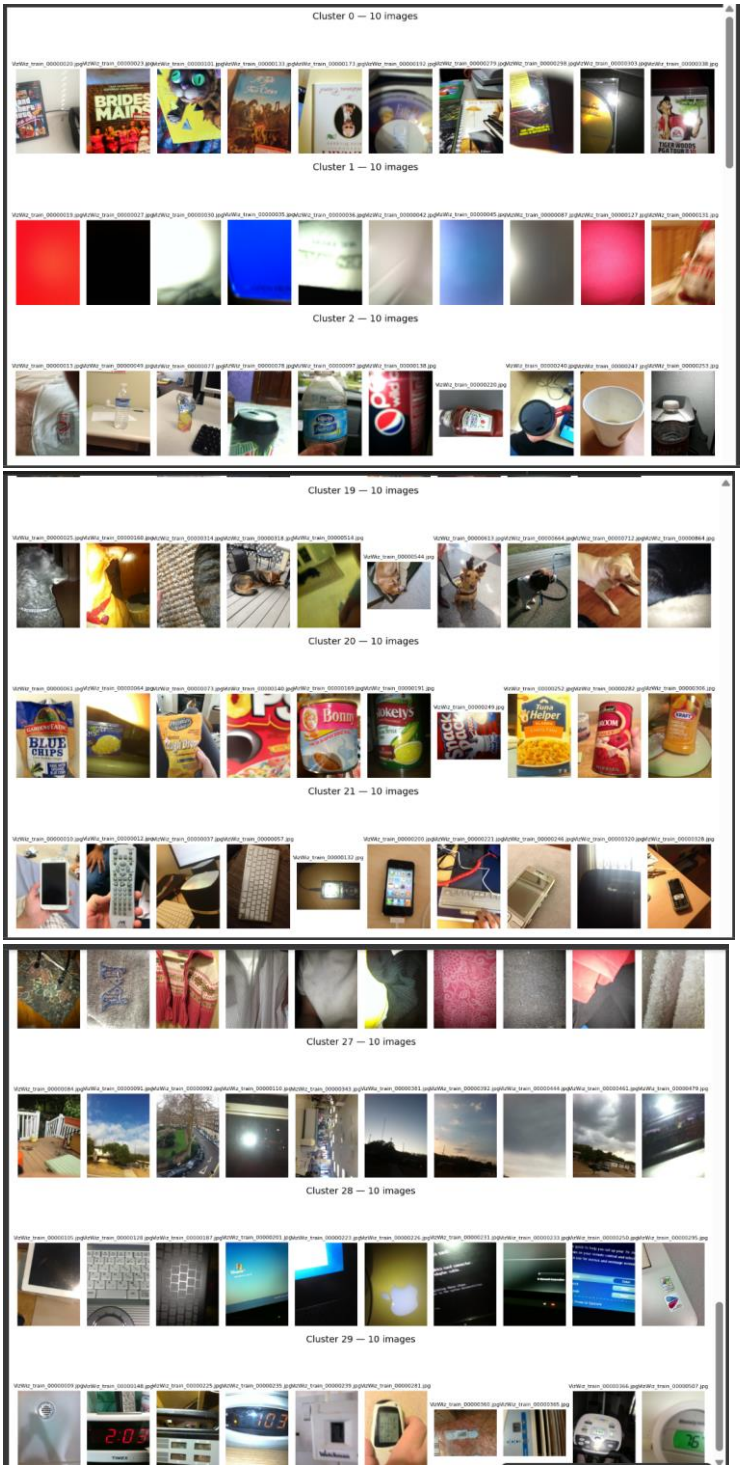
C. Test Image Used for testing and Predictions

The test image shown in Fig. 2 was selected to evaluate our model performance under real-world conditions.



D. Training Data Clusters Overview

Here are some of the training data Clusters:



IV. CONCLUSION

This project shows the feasibility of solving the blind image understanding challenge without labels. Our approach, combining CLIP embeddings, prompt engineering, and hybrid similarity, delivers strong results and can be extended to other zero-shot domains. Future work may involve better cluster assignments and real-user testing.

V. REFERENCES

[1]D.Gurari, "2025 VizWiz Grand Challenge Workshop," VizWiz, 2025. [Online]. Available: https://vizwiz.org/workshops/2025-vizwiz-grand-challenge-workshop/

[2]D. Gurari, "VizWiz-Classification Dataset," VizWiz, 2025.[Online].Available: https://vizwiz.org/tasks-and-datasets/image-classification/