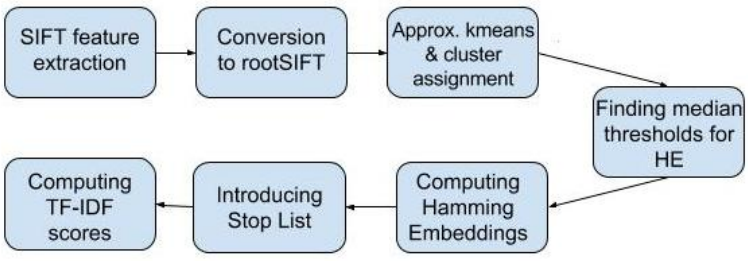
# CS698): Assignment 1
# Exact Instance Retrieval

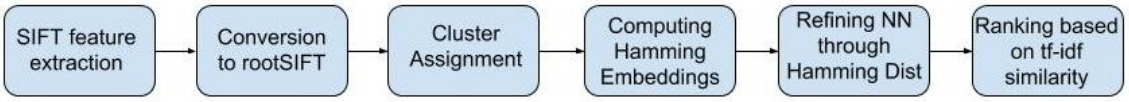
Rishabh Goyal, 14549
Priyank Pathak, 13514

## Techniques Used

1. **Feature Extraction:** We have used RootSIFT features, which have been shown to give better performance than, SIFT features in [1]. These features can be obtained from SIFT features by taking L1 norm, followed by taking element-wise square root.
   *Note:* SIFT features were extracted using David Lowe's MATLAB code.
2. **Approximate K-Means:** Approximate K-Means was implemented using kd-trees. It is computationally much cheaper than vanilla K-Means, which enabled us to train our model for a comparatively much larger number of visual words. In [2], it has also been shown to perform better than hierarchical K-Means.
3. **Hamming Embeddings:** If comutationally feasible, the most appropriate approach would have been to find the kNN of the computed SIFT features and use them to compute similarity score. However finding Euclidean distance between a large number of points disallows this. Hamming embeddings enable us to learn binary representation for each feature, such that the distance between any teo features in the Euclidaen space is closely approximated by distance in this Hamming Embedding space. Therefore computation of distance is reduced to a bitwise XOR which is much cheaper [3]. Use of Hamming mbeddings helps us draw fine grained boundaries within a cluster and improves performance substantially.
4. **Stop-List:** A stop list is basically removing a percentage of the most popular visual words from the vocabulary. This helps us to deal sith the problem of clutter like chessboard, table components etc. which are abundantly visible in all images. This idea was used in [4]
5. **TF-IDF:** The TF-IDF scores are then computed for the pruned vocabulary. This gives appropriate weights to visual words based on their frequency in the document and the images.




Training Pipeline: SIFT, approximate k-means, Hamming Embeddings and TF-IDF score were main techniques used




Test Pipeline

# Results

After appropriately tuning the hyperparameters, the following results were obtained for the sample dataset provided for testing.

| **k** | **5** | **10** | **30** | **72** | **6048** |
|---|---|---|---|---|---|
| **MAP** | 0.76 | 0.56 | 0.26 | 0.12 | 0.13 |

Table 1: MAP scores for differently values of $k$, 30% of popular words removed and hamming threshold of 30, for *testSample* dataset

| **k** | **5** | **10** | **30** | **72** | **6048** |
|---|---|---|---|---|---|
| **MAP** | 1.0 | 0.9 | 0.72 | 0.45 | 0.50 |

Table 2: MAP scores for differently values of $k$, 30% of popular words removed and hamming threshold of 45, for *testSample2* dataset

# Possible Improvements

The following improvements were thought of, but were not implemented due to the lack of time:

1. SIFT does not use colour information. Performance can be greatly impoved if we incorporate colour information somehow.
2. Use of spatial information. Spatial re-ranking using the RANSAC algorithm could be done for further improvement.
3. Increasing the number of visual words is another trivial improvement, that was limited by computational resources.

# References


[1] R. Arandjelovic and A. Zisserman, "Three things everyone should know to improve object retrieval," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 2911–2918.

[2] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Object retrieval with large vocabularies and fast spatial matching," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2007, pp. 1–8

[3] H. Jegou, M. Douze, and C. Schmid, "Hamming embedding and weak geometric consistency for large scale image search," in European conference on computer vision , 2008, pp. 304–317.

[4] J. Sivic and A. Zisserman, "Video google: A text retrieval approach to object matching in videos," in Proceedings of the IEEE International Conference on Computer Vision, 2003, pp. 1470–1477.