AKULA – Adaptive Cluster Aggregation for Visual Search

Abhishek Nagar[*], Zhu Li[*], Gaurav Srivastava[*], and Kyungmo Park[+]

[*]*Samsung Research America*
1301 E. Lookout Dr.
Richardson, TX 75082, USA
{a.nagar, zhu1.li, srivastava.g}@samsung.com

[+]*Samsung Electronics*
416 Maetan 3-dong, Youngtong-gu. Suwon,
Gyeonggi-do, Korea 442-600
kyungmo.park@samsung.com

Abstract: Key point features are very effective tools in image matching and key point feature aggregation is an effective scheme for creating a compact representation of the images for visual search. This solution not only achieves compression, but also offers the benefits of better accuracy in matching and indexing efficiency. Research is active in this area and recent results on Fisher Vector based aggregation have shown to be very effective in a number of application scenarios. In this paper, we present a new direct aggregation scheme that is adaptive to the descriptor distributions from individual images and does not enforce a single generative model such as GMM in the Fisher Vector type aggregation. Moreover, it achieves better compression as well as image matching accuracy. Simulation results with the image identification data set from MPEG Compact Descriptor for Visual Search (CDVS) effort demonstrate the effectiveness of this approach.

Introduction

Image search involves identifying images with similar objects. The main challenges of image search include differences in illumination, geometric transformations, and occlusion of the objects in the images. Scale Invariant Feature Transform (SIFT) [Lowe04] is currently one of the most successfully used representation for image search. SIFT features are essentially a set of salient points identified on the image that can be reliably detected despite imaging variations. However, despite their high accuracy in image search, their use in applications like mobile visual search, is handicapped by the fairly large bit rate of the uncompressed SIFT features. For an VGA sized image, around 1000 SIFT key points are generated on average, requiring 128 bytes each, leading to a visual query representation 128k bytes in size. This may sounds trivial but for large number of mobile users and applications like augmented reality and smart glasses, the traffic can quickly grow and put a heavy burden on the mobile networks.

To address this, direct compression of the SIFT descriptors can be a viable solution. In fact, solutions like Product Quantization [Wang12], low memory transform coding [m25929], and Laplacian Embedding based transform coding [Xin13], have shown that SIFT descriptor can be compressed 10x times to approximately 80~120 bits per descriptor, without losing too much key point matching performance. This brings the bit rate for sending 1000 SIFT key points down to 10~15K bytes range.

To achieve even further compression efficiency, directly coding key points is becoming difficult. Instead, a key point aggregation scheme is introduced. Aggregation does not

seek to preserve the key points information for reconstruction, instead it finds alternative representation of a collection of key points. This not only achieves better compression, but also makes the matching between two images more efficient. Furthermore as aggregation typically yields a vectorized representation of the image, it makes efficient indexing and hashing scheme possible, to help large scale image data base retrieval.

In this paper, we present an Adaptive KLUster Aggregation, or AKULA aggregation solution that is extremely compact compared with the state of art in aggregation, and the simulation with the data set and ground truth of the MPEG Compact Descriptors for Visual Search (CDVS) [Reznik11], demonstrated that the new AKULA aggregation is also performing better in image matching. The AKULA descriptor is also integrated into the MPEG CDVS Test Model [W13564], and the end-to-end performance of AKULA vis-a-vis Fisher Vector based aggregation is quite favorable.

Aggregation in Visual Search

Aggregation takes a collection of key points as input, and output a new descriptor that is compact, offers better image matching capability, and can also be stateless and leads to better indexing efficiency. A number of techniques have been proposed in the literature to aggregate the local key point descriptors into an efficiently search-able Global Descriptor (GD). One technique is the bag-of-words approach (BoW) [Csurka04], [Sivic03] which originated in the text-based information retrieval research. A BoW representation computes the zeroth order statistic of the local descriptors of an image, which is a histogram of local descriptor counts corresponding to a set of representative visual words generated by quantizing the feature space. A major limitation of the BoW representation is its low discriminability, which makes its use prohibitive for large databases with millions of images.

GDs such as Vectors of Locally Aggregated Descriptors (VLAD) [Jegou10] and Residual Enhanced Visual Vectors (REVV) [Chen11] compute the first-order statistic by first computing the errors of the local descriptors from their nearest visual words and then aggregating them into a mean vector for each visual word. The mean error vectors for all the visual words are concatenated to form the GD.

Fisher Vector [Perronnin10] and Scalable Compressed Fisher Vector (SCFV) [Duan13] techniques lead to enhanced matching accuracy by performing second-order aggregation i.e. computing the mean and variance of local descriptor errors with respect to cluster centers of a Gaussian mixture model. In these methods, the dimension of each local descriptor is first reduced to d using principal component analysis (PCA). Then, a Gaussian mixture model (GMM) is trained with k component Gaussian functions using a training database. For each of the local descriptors in the image, a d dimensional Fisher information vector is computed with respect to each of these k Gaussian components. The Fisher information vectors are averaged for each of the k components separately to form a k x d dimensional vector, which is then binarized to form the compressed Fisher vector representation. The scalability is incorporated by masking the bits corresponding to a set of Gaussian components separately for different images.

Of the two aggregation schemes REVV and SCFV that were adopted into the MPEG CDVS visual object identification pipeline, a drawback is that they both requires the storage of a global model, that can have a large memory cost to the mobile

implementation, and the dependence on this global model, actually limits the degree-of-freedom (dof) of the aggregation, that yields sub-optimal performance in both compression efficiency and matching accuracy.

Adaptive Local Visual Feature Aggregation

The Adaptive KLUster Aggregation (AKLUA), or AKULA with a slight abuse of the order of the acronym, is trying to capture the aggregating information from visual key points without any dependence on a global generative model, as the *k*-means model in VLAD and the GMM model in the Fisher Vector cases. This eliminates the cases of poor coding of the information of images whose generated key points deviate significantly from the global model. For compact aggregation that is to be sent over a bandwidth limited channel, the number of GMM components allowed is limited and this directly affects the performance of the Fisher Vector type aggregation as outlined in the original paper [Jegou10].

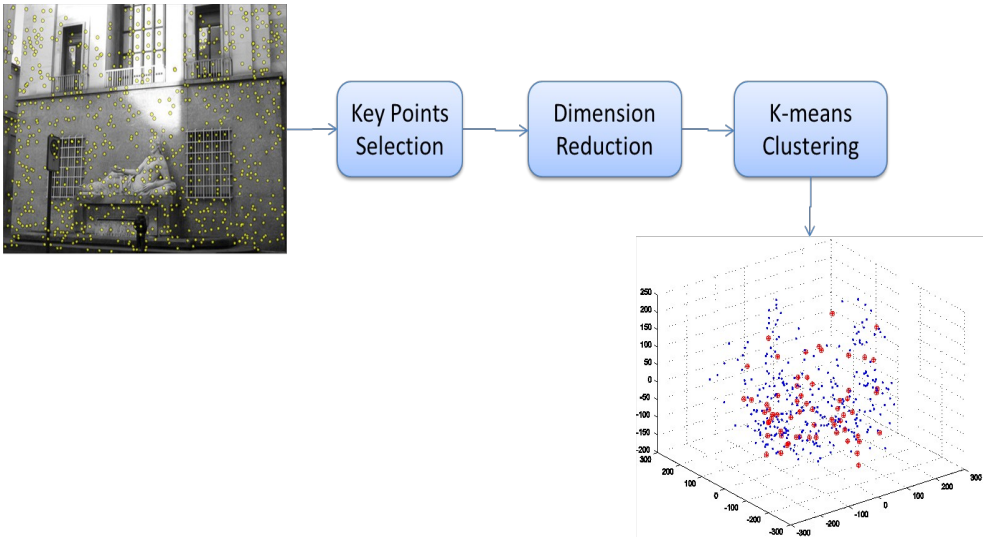


Figure 1. AKULA aggregation process

The process of AKULA description generating is illustrated by a diagram in the Figure 1. First the dimensionality of the key point feature space, in this case, the SIFT space, is reduced via a Principal Component Analysis (PCA), then the aggregation is obtained by directly computing the k-means centroids of key points in some properly obtained subspaces, and their associated key points count. Let a collection of key points be $S = \{s_1, s_2, ..., s_n\}$, and its *m*-cluster AKULA aggregation is represented by the centroids x_k, and key points counts p_k as,$A = \{x_1, x_2, ..., x_m; p_1, p_2, ..., p_m\}$, obtained through k-means clustering, which finds cluster centers and its assignment q_k, such that the following distortion is minimized,

$$\min_{x_1,x_2,...,x_m,q_1,q_2,..,q_n \in [1..m]} \sum_k dist(s_k, x_{q_k}) \qquad (1)$$

This is typically computed though the well-known Lloyd-Max algorithm [Lloyd82] which has many fast implementations. It is worth noting that not all the key point

generated are used for the AKULA aggregation, and the selection of key points for aggregation is not a trivial process. A typical 640x480 pel VGA sized image can generate 1100 SIFT key points in average and not all of them having good repeatability in key point matching. Therefore a key point selection process that based on a probabilistic modeling of the likelihood of repeatability in key point detection is used. Here, the repeatability likelihood functions are obtained conditioned on a set of observed key point features: peak strength, scale, orientation and distance to the center of image. Details can be found in [m31396].

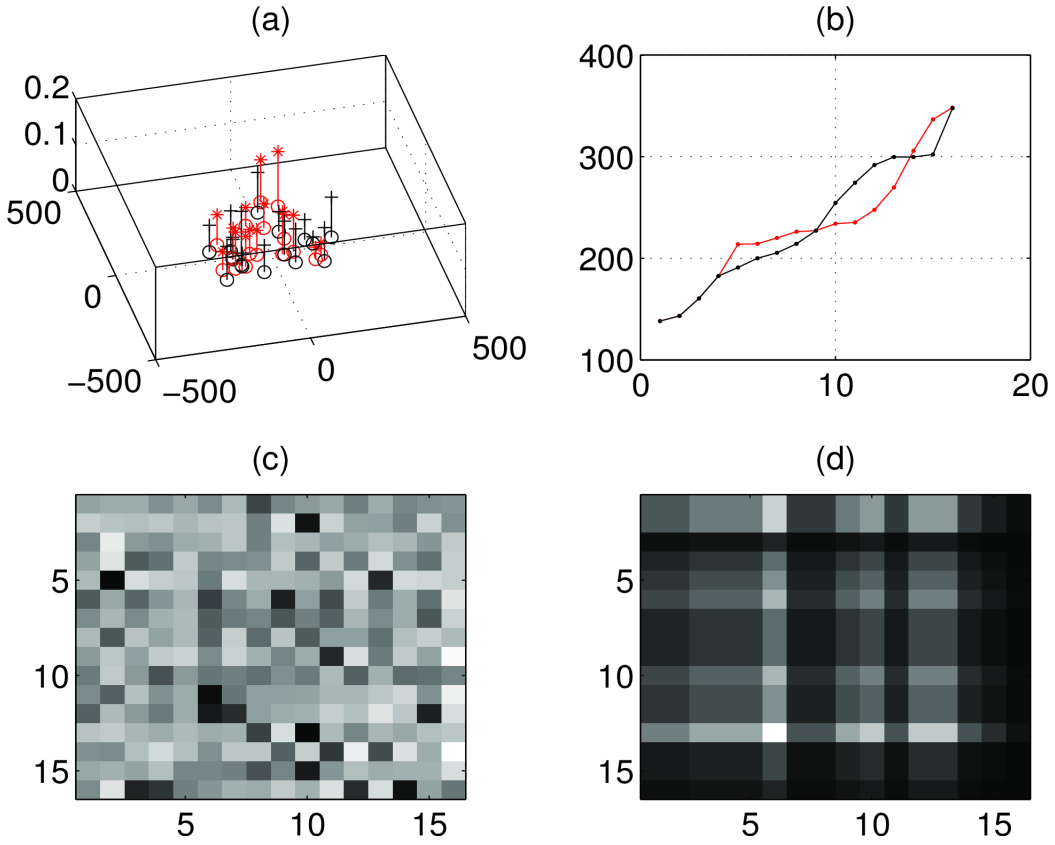


Figure 2. AKULA distance metric

AKULA is not a state-less descriptor. Its structure is more like the minutia points based representation for fingerprint verification. The distance metric for AKULA should reflect the fact that any permutation of the centroids, should still be a representation of the same aggregation. The distance between two AKULA descriptors, A^1, and A^2 , is therefore computed as,

$$d(A^1, A^2) = \frac{1}{m}\sum_{k=1}^{m} d^f_{min}(k) w_f(k) + \frac{1}{m}\sum_{j=1}^{m} d^b_{min}(j) w_b(j) \quad (2)$$

Here the forward and backward minimum distances are computed as,

$$d^f_{min}(k) = \min_j d(j,k) \quad (3)$$

$$d^b_{min}(j) = \min_k d(j,k) \quad (4)$$

where the AKULA centroids distances are $d(j,k) = \sqrt{(x_j - x_k)^2}$. The weights associated with the forward/backward minimum distances, are computed as,

$$w_f(k) = p(j^*) + p(k), s.t. j^* = arg\min_k d(j,k) \quad (5)$$

$$w_b(j) = p(j) + p(k^*), s.t., k^* = arg\min_j d(j,k) \quad (6)$$

The AKULA distance metric is very intuitive to understand. The forward and backward matching process finds the nearest neighbors pairs of the AKULA centroids, and this is weighted by the number of key points counts associated with them. The Dominant Color Descriptor [Manjunath01] from the MPEG-7 standardization effort has a similar structure, but having a different distance metric which computes affinity from all distance pairs involved.

An example of AKULA distance computing is illustrated in Fig. 2, where Fig. 2a is a plot of two AKULA descriptors with m=16 clusters, plotted on the first 2 dimensional plane of the feature space, and the z-axis represents the normalized key point counts. The $d^f_{min}$and d^b_{min} are illustrated in Fig. 2b, and the pair-wise centroid distance matrix and weight matrix are shown in Fig. 2c and Fig. 2d respectively.

Overall, the AKULA aggregation bit rate is very compact. For a PCA dimension of d_0=8, and m=16, it requires only 128 bytes to encode. In fact our AKULA used in benchmarking has a bit rates of 64, 128, and 256 bytes. Similarly performing state-of-art aggregation schemes like Fisher Vector, in contrast, requireing 256 to 1024 bytes. AKULA is a much better aggregation scheme in terms of compression efficiency.

Simulation Results

The MPEG working group under the ISO is developing an international standard on image query compression for mobile visual search over the wireless channels, it is called Compact Descriptor for Visual Search (CDVS) [W13564]. A large data set is collected for this standardization effort, it consists of approximately 36000 labeled images and more than 1 million unlabeled images from various sources including CD/DVD cover, books, building and landmarks, video clips, paintings and prints. Some samples from the CDVS data set are shown in Fig. 3.

The performance of the AKULA in image matching is benchmarked against the current state of the art, i.e, the Fisher Vector approach. There are approximately a total of 186K labeled image matching ground truth items, organized into five data sets shown in Table I.

Table I. CDVS data set

Data Set	1a, 1b	2	3	4	5
Content	Graphics, CD/Books	Paintings	Video Frames	Buildings/ Landmark	Common Objects

The True Positive Rate (TPR) and the False Positive Rate (FPR), are defined as,

$$TPR = \frac{tp}{tp + fp}$$

$$FPR = \frac{fp}{tn + fp}$$

The image matching performance as TPR(FPR) are plotted in the Figs 4a ~4f below, for the data sets 1a, 1b, 2, 3, 4, and 5. For the data set 1a and 1b, which consist of book and CD covers taken at different angles and illumination conditions, the AKULA starts to outperform the Fisher Vector for FPR greater than 10%, which for data set 2, 3, which are museum paintings, randomly selected video frames, AKULA consistently outperforms Fisher Vector throughout all FPR operating ranges. For data set 4 and 5, which include images of 3D objects that having much larger variations in object position and extrinsic/intrinsic camera parameters in image formation, the performance of AKULA is even better. This points to the strength of the AKULA aggregation scheme, which not only achieve better compactness in aggregation, but also delivers better performance in aggregation-descriptor only image matching performances.

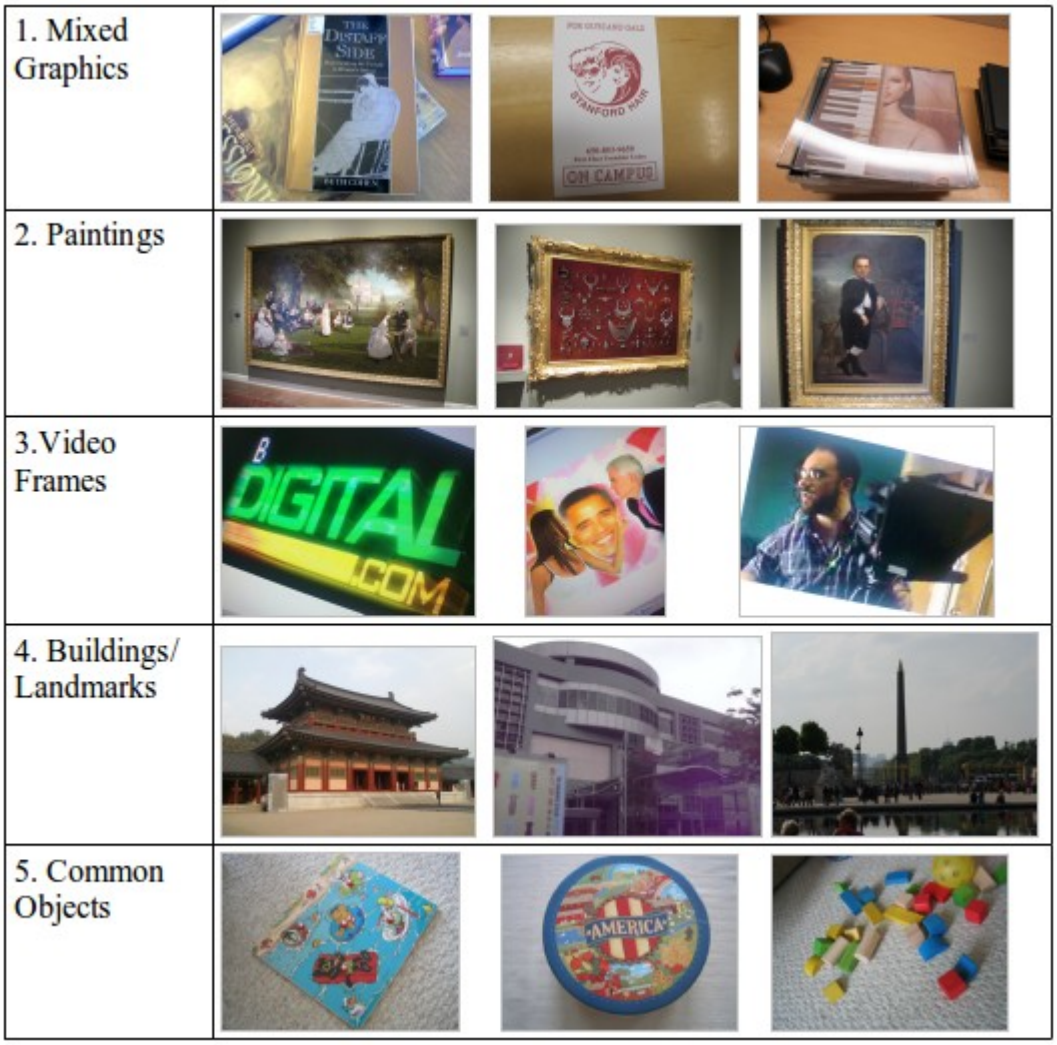
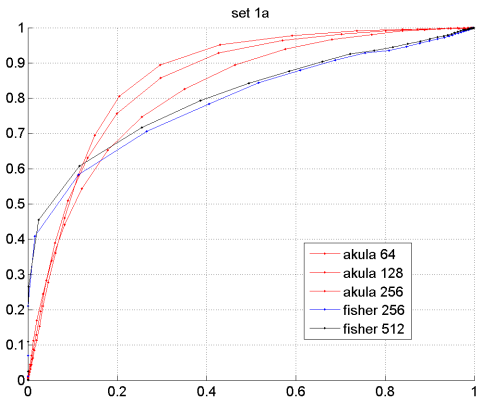


Figure 3. CDVS Data set

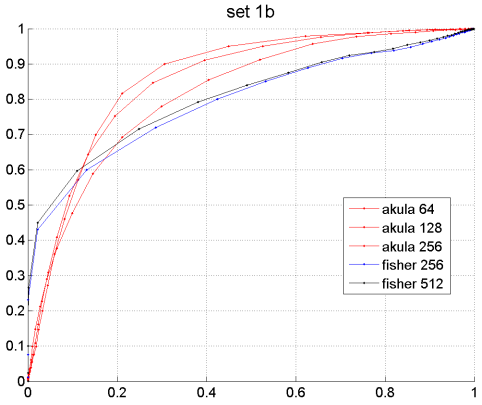
In Fig. 4, the red dotted plots are the AKULA TPR-FPR curves at bit rate 64, 128 and 256 bytes, for d_0=8, and m=8, 16 and 32. The blue and black curves are the Fisher Vector

TPR-FPR performances at 256 and 512 bytes, for the same dimensionality and with 32 and 64 GMM components. The Fisher Vector is implemented with the VL_FEAT package [Vedaldi10]. To have a fair comparison, the Fisher Vector and AKULA aggregation schemes are fed with exactly the same set of SIFT key points from the image. After initial detection of SIFT points in an image, the key points are sorted by the peak strength, and only the top k=300 SIFT points are selected for the aggregation.

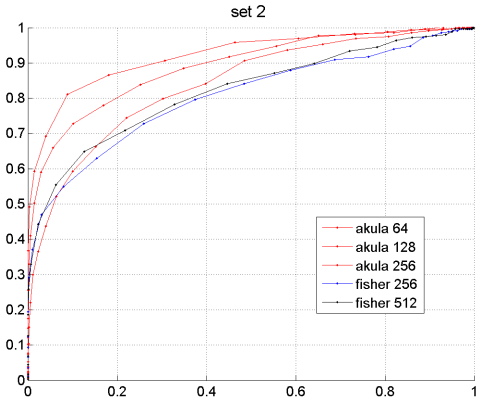
Notice that AKULA is not a state-less descriptor, like Fisher Vector, where the description can be easily binarized for fast comparison. Any permutations of AKULA centroids and associated key point counts, are still valid AKULA description of the same image, and should return zero-distance among them.



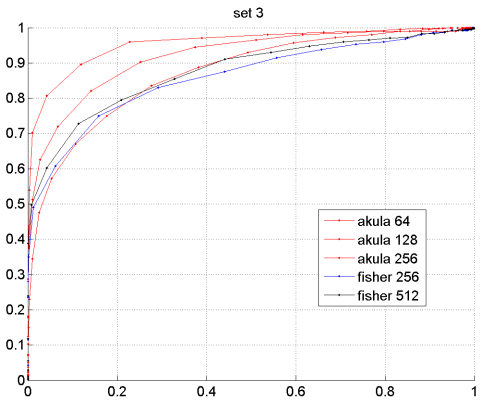
(a)



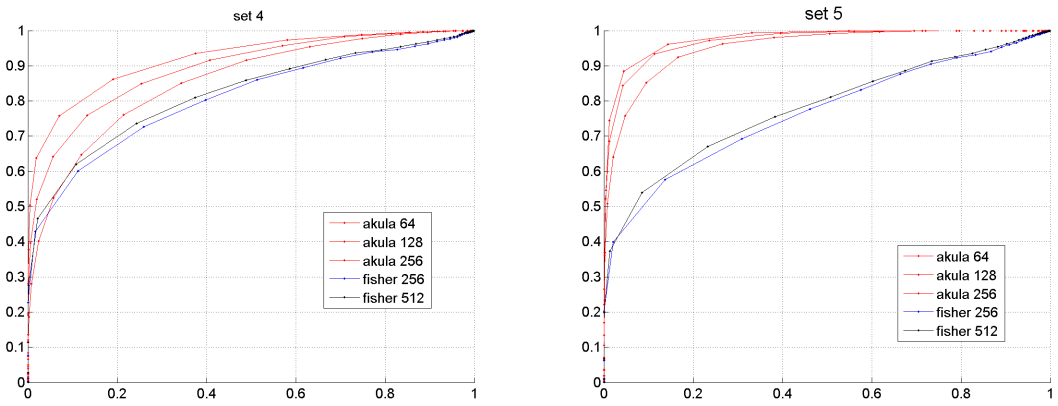
(b)



(c)



(d)



(e) (f)

Figure 4: AKULA vs Fisher Vector in Image Matching

The AKULA performs well in data sets 2,3,4 and 5 across all FPR ranges, while for data set 1, the Fisher Vector performs better at below 10% FPR range. Overall, AKULA is offering a better image matching performance while operates at much smaller bit rate.

Notice that the key points are not involved in the image matching, only the aggregated information is used. This puts the compression ratio of the visual query for a typical 60KB VGA size JPEG image at 250~1000 times.

The AKULA aggregation scheme is also integrated into the MPEG CDVS test model [m31491], it operates at 64, 128 and 256 bytes as discussed, and performs well compared with a much larger bit rate (280~980 bytes) binarized Fisher Vector type aggregation scheme [Duan13]. The GD operating rates are summarized in Table II, for CDVS rates between 0.5 ~ 16K bytes:

Table II. Aggregation Rates

Rate	512	1K	2K	4K	8K	16K
Fisher Vec Rate	276	292	319	621	731	804
AKULA Rate	64	128	128	128	256	256

The overall image matching performance is matched up, while the saving in bit rates significantly improved the object localization performance, especially at the lower bit rates, as illustrated in the Fig. 5,

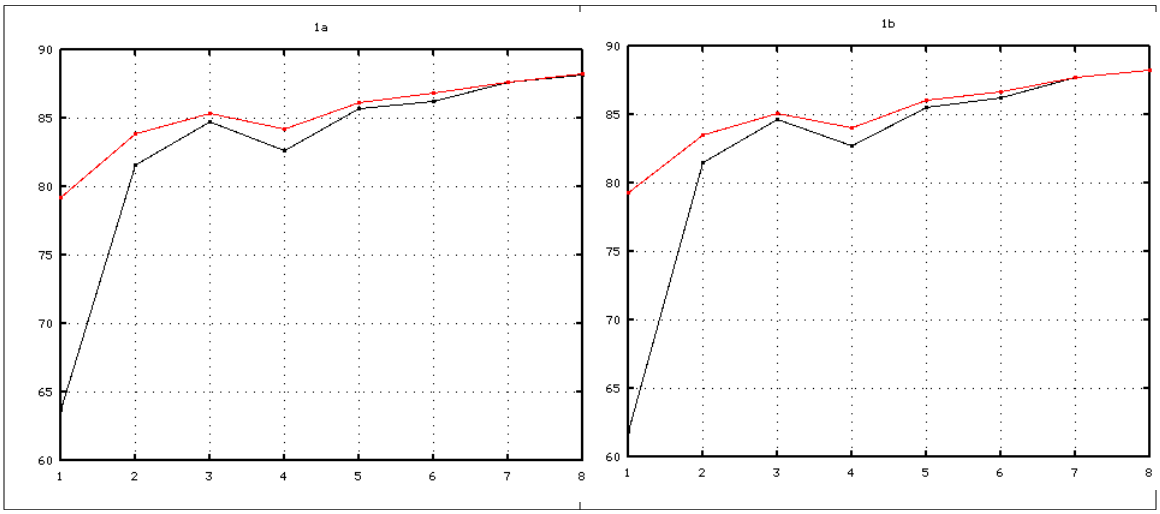


Figure 5. Localization Accuracy

The red curve is the localization accuracy for AKULA integrated solution, while the black curve is that of the Fisher Vector. The significant gains at the lower rates of matching, can be very valuable for applications where low rates operation are mandated.

Conclusion and Future Work

Visual identification and search is becoming a key enabler for a variety of important applications in mobile computing. In this work, we presented a novel local key point feature aggregation scheme that is compact in resulting visual query bit rates, while offering state-of-art performance in image matching. In the future further optimization will be performed on the distance metric tuning, introducing heat kernel mappings that can adapt the metric to better reflects the true potential of this approach.

Furthermore, an outer loop of optimization is to be introduced to optimizes the subspaces where this aggregation is performed. Ideas and frameworks from the Complimentary Hashing work [Xu11] and Grassmannian Hashing [Wang11] are being adopted for this purpose, and further performance gains are expected.

References

[Chen11] Chen, David, et al. "Residual enhanced visual vectors for on-device image matching.", *Proc of the 45th Asilomar Conf on Signals, Systems and Computers* (ASILOMAR), 2011, Asilomar.

[Csurka04] G. Csurka et al., ‘‘Visual Categorization with Bags of Keypoints,’’ *Proc. Workshop on Statistical Learning in Computer Vision*, ECCV, vol. 1, 2004, p. 22.

[Duan13] Duan, Ling-Yu, et al. "Compact descriptors for mobile visual search and MPEG CDVS standardization.", *IEEE International Symposium on Circuits and Systems* (ISCAS), 2013, Beijing, China.

[Jegou10]H. Jegou, M. Douze, C. Schmid, and P. Perez, “Aggregating local descriptors into a compact image representation”, *Proc of IEEE CVPR* 2010, pp. 3304–3311.

[Lloyd82] S.Lloyd. Least square quantization in PCM. *IEEE Trans. on Information Theory*, 28(2), 1982.

[Lowe04] D.G. Lowe, ‘‘Distinctive Image Features from Scale-Invariant Keypoints,’’ *Int’l J. Computer Vision*, vol. 60, no. 2, 2004, pp. 91-110.
[m25929] Stavros Paschalakis, Karol Wnukowicz, Miroslaw Bober, Alessandra Mosca, Massimo Mattelliano, Gianluca Francini, Skjalg Lepsoy, Massimo Ballestri, “CDVS CE2: Local Descriptor Compression Proposal”, *ISO/IEC JTC1/SC29/WG11/m25929*, July., 2012, Stockholm, Sweden.
[m31396] Gianluca Francini, Massimo Balestri, Skjalg Lepsoy, “CDVS: extended dataset for feature selection training”, *ISO/IEC JTC1/SC29/WG11/m31396*, Oct., 2013, Geneva, Switzerland
[m31491] A. Nagar, Z. Li, G. Srivastava, and K. Park, “CDVS CE2: AKULA - Adaptive Cluster Aggregation ”, *ISO/IEC JTC1/SC29/WG11/m31491*, Oct., 2013, Geneva, Switzerland.
[Manjunath01] B. S. Manjunath, Jens-Rainer Ohm, Vinod V. Vasudevan, Akio Yamada: Color and texture descriptors. *IEEE Trans. Circuits Syst. Video Techn*. 11(6): 703-715 (2001).
[Perronnin10] F. Perronnin, J. Sánchez, and T. Mensink. Improving the fisher kernel for large-scale image classification. In *Proc. ECCV*, 2010.
[Reznik11] Y.A. Reznik, ‘‘On MPEG Work Towards a Standard for Visual Search,’’ *Proc. SPIE 8135, Applications of Digital Image Processing XXXIV*, SPIE, 2011.
[Sivic03] J. Sivic and A. Zisserman, ‘‘Video Google: A Text Retrieval Approach to Object Matching in Videos,’’ *Proc. 9th IEEE Int’l Conf. Computer Vision*, IEEE CS, 2003, pp. 1470-1477.
[W13564] M. Bober et.al, “Test Model 6: Compact Descriptor for Visual Search”, *ISO/IEC JTC1/SC29/WG11/W13564*, Apr., 2013, Inchoen, Korea.
[Vedaldi10] A. Vedaldi, B. Fulkerson: Vlfeat: an open and portable library of computer vision algorithms. *ACM Multimedia* 2010: 1469-1472.
[Wang11] X. Wang, Z. Li, L. Zhang, J. Yuan, “Grassmann Hashing for approximate nearest neighbor search in high dimensional space”, *Proc. IEEE Intl Conf on Multimedia & Expo* (ICME), Barcelona, Spain, 2011.
[Wang12] C. Wang, L.-Y. Duan, Y. Wang, and W. Gao, “PQ-WGLOH: A bit-rate scalable local feature descriptor”, *Proc of IEEE ICASSP*: pp.941-944, Vancouver, Canada, 2012.
[Xin13] X. Xin, Z. Li, A. K. Katsaggelos, "Laplacian embedding and key points topology verification for large scale mobile visual identification", *Signal Processing: Image Communication*, vol. 28(4): pp. 323-333, 2013.
[Xu11] H. Xu, J. Wang, Z. Li, G. Zeng, S. Li, N. Yu, “Complementary hashing for approximate nearest neighbor search”, *IEEE Intl Conf on Computer Vision* (ICCV), Barcelona, Spain, 2011: 1631-1638.