# Early burst detection for memory-efficient image retrieval – Supplementary material –

Miaojing Shi *
Peking University

Yannis Avrithis
University of Athens, NTUA

Hervé Jégou
Inria

## Abstract

*This supplementary material provides additional experimental results, discussion and interpretations that complement our paper. The additional results refer to more parameters and options in the underlying retrieval models, more parameters and options in our burst detection and aggregation, more datasets and more large scale experiments. We also include a comparison to the state-of-the-art.*

## 1. Introduction

We follow the same experimental setup as outlined in section 4.1 in the paper, and separately present results on VLAD and SMK*/ASMK*, as in Sections 4.1 and 4.2 of our the paper, respectively. As promised in our paper, the content is as follows.

1. Additional results for varying vocabulary size $k$;
2. Analysis of the effect different parameters depending on the retrieval model;
3. Results for more datasets, including large scale ones.

We adopt connected components for burst detection by default in all our experiments. On *Holidays*, we employ the proposed descriptor kernel and scale kernel only; referring to (2) in the paper, we use kernel $k_u k_s$. On *Oxford* and *Paris* on the other hand, we employ all three kernels, *i.e*. descriptor, scale and orientation; that is, *i.e*. $k_u k_s k_\theta$ referring to (2). This setting always gives the best performance.

In all graphs, the baseline is always the rightmost point.

## 2. More results with VLAD

We begin by providing a quantitative comparison of connected components *vs*. hierarchical spectral clustering (HSC) for burst detection and aggregation, justifying our choice of algorithm. We then complement our evaluation on VLAD given in section 4.2 in the paper, by providing results on varying vocabulary size $k$, on the effect of power law normalization and on large scale experiments. Unless otherwise stated, we use a vocabulary of $k = 16$, no power law normalization and symmetric aggregation.

**Burst detection.** Fig. 1 illustrates the performance of VLAD on *Holidays-L* under varying aggregation% for two different burst detection methods, namely connected components and HSC. As shown by the results of the preliminary study in Fig. 4 and 5 in the paper, HSC is a spectral method that produces both consistent groups of features and a reasonable distribution of group sizes, that is actually quite similar to that of connected components. Albeit more simple, the approach based connected components turns out to always be superior. The situation is even more favorable for connected components *vs*. other methods. This is why, after this preliminary analysis, we focus on connected components in our experiments.

**Vocabulary size.** Fig. 2 shows the performance on *Holidays-L* as a function the vocabulary size $k$. There is an improvement of 2-3% over the baseline for a large range of aggregation%. The relative improvement is higher for smaller vocabularies. A possible explanation is that bursts are split into different cells when the vocabulary is larger.

**Power law.** Fig. 2 also shows the effect of power law normalization by repeating the measurements for $\alpha = 1$ and $\alpha = 0.7$, where $\alpha$ is the power law parameter of [11]. It can be seen that, by aggregating bursty features at an early stage, we obtain at least as much gain as with power law on baseline VLAD (with aggregation% = 1). Furthermore, the two methods are complementary as power law yields a further improvement of 1-1.5% on the highest performance of our early aggregation. With $k = 64$ for instance, mAP is 63.0 and 64.3 for $\alpha = 1$ and $\alpha = 0.7$, respectively. This result is already very close to the highest value of 65.8 ob-

*Miaojing Shi has worked on this paper while he was a visiting student at INRIA Rennes.

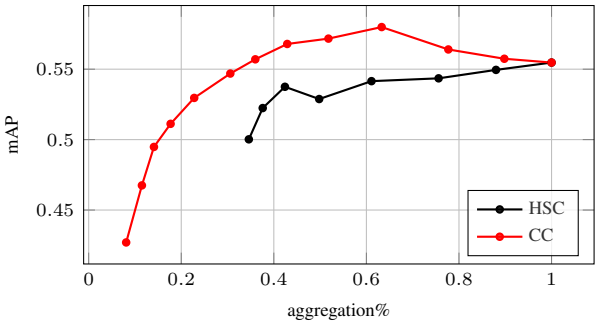


Figure 1. VLAD performance *vs*. aggregation% on *Holidays-L* for connected components (CC) and hierarchical spectral clustering (HSC). Vocabulary size $k = 16$; baseline power law parameter $\alpha = 1$.

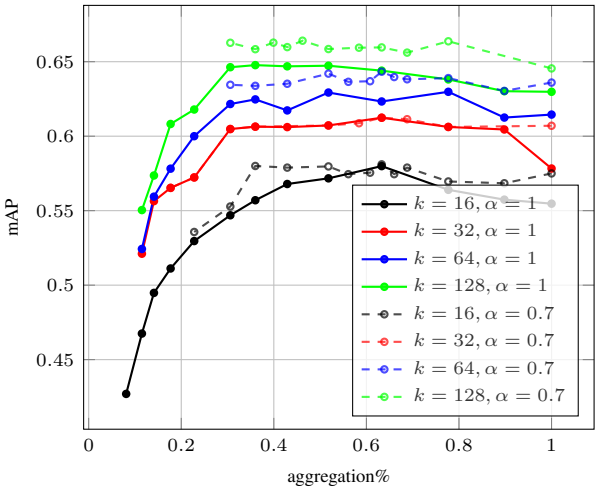


Figure 2. VLAD performance *vs*. aggregation% on *Holidays-L* for different vocabulary sizes $k$ and different power law parameter $\alpha$.

| aggregation% | 1.000 | 0.764 | 0.638 | 0.556 |
|---|---|---|---|---|
| $k = 16$ | 41.3 | 42.7 | 44.1 | 45.0 |
| $k = 64$ | 46.3 | 47.5 | 48.3 | 48.8 |

Table 1. VLAD mAP performance *vs*. aggregation% on *Holidays-L* plus 100K distractors for two vocabulary sizes.

tained in [6] using residual normalization (RN) and local coordinate system (LCS).

**Large scale.** Following Table 1 in the paper, Table 1 provides more results on large scale experiments on *Holidays-L* plus 100K distractors. With a further decrease of aggregation%, we still get further absolute performance gain over baseline VLAD (aggregation% = 1) at even more reduced memory and query time. As in Fig. 2, the improvement is higher for smaller vocabularies.

## 3. More results with SMK*/ASMK*

We extend here the evaluation on SMK*/ASMK* given in section 4.3 in the paper, by providing results on varying vocabulary size $k$, on the impact of SMK*/ASMK* selectivity on *Holidays-L*, on the *Oxford* and *Paris* datasets, on multiple assignment and on large scale experiments. We also include a comparison to the state-of-the-art. Unless otherwise stated, we use a vocabulary of $k = 65$K, selectivity parameter $\alpha = 3$ and asymmetric aggregation. This corresponds to aggregation% $= 1$ for SMK*, and less than 1 for ASMK* because ASMK* provides its own form of aggregation. We measure the total aggregation%, *i.e.*, the ratio of encoded descriptors over initial descriptors.

**Vocabulary size.** Fig. 3 compares different vocabulary sizes on *Holidays-L*. By adopting our early burst aggregation, we get absolute improvement over all sizes. We observe that the improved trade-off holds for all tested vocabulary sizes. The improvement is more significant for SMK*: performance remains constant or improves even for 60% aggregation before starting to drop. This is expected because ASMK* includes another form of aggregation.

**Selectivity.** The impact of SMK*/ASMK* selectivity is evaluated in Fig. 4. It appears that the previous observations hold under varying selectivity and our method maintains good performance over a wide range of aggregation%.

**More datasets.** We also compare symmetric to asymmetric aggregation on *Oxford* and *Paris* as well, as shown in Fig. 5,6 respectively. These datasets are notoriously more difficult for methods operating under limited memory. Still, our method maintains a good memory efficiency. In *Paris* for instance, we can keep only 50% features for a 4% drop in mAP. Asymmetric aggregation is still superior for low aggregation%.

**Multiple assignment.** We further evaluate our burst detection and aggregation by adopting multiple assignment (MA) in the ASMK* framework. We apply it on the query side only as in [9], which is consistent with not aggregating query descriptors in our asymmetric strategy. We use the five nearest visual words as in [17]. Fig. 7 shows ASMK* results for two initial feature sets. *Holidays-L* still maintains a gain of 6-7% over *Holidays-S* at the same memory, so we still offer a significant performance boost at no cost. Comparing to Fig. 9 in the paper, observe that single and multiple assignment give similar performance when aggregating; this means we can have this boost at a fraction of query time without multiple assignment. Finally, Fig. 8 shows ASMK* results on *Oxford* and *Paris*; under multiple assignment, we even get absolute performance improvement in this case.

| dataset | *Holidays-L* 101K | | | *Oxford* 105K | | |
|---|---|---|---|---|---|---|
| aggregation% | 0.65 | 0.52 | 0.28 | 0.90 | 0.76 | 0.55 |
| mAP | 85.1 | 84.5 | 77.6 | 68.9 | 68.9 | 63.6 |

Table 2. ASMK* mAP performance on *Holidays-L* and Oxford plus 100K distractors. Vocabulary size: 65K. The first column of each dataset is the baseline mAP and aggregation% in ASMK*.

**Large scale.** Following Table 2 in the paper, Table 2 provides more results on large scale experiments on *Holidays-L* and Oxford plus 100K distractors. It is interesting that *e.g.* in *Oxford*, the result is more promising than at small scale: we can save 15% of memory at no performance cost while increasing efficiency, or keep only 50% features for a 5% drop in mAP.

**Comparison to the state of the art.** Table 3 shows state-of-the-art results compared to our best results on ASMK*. We only compare to methods relating to vocabularies and descriptor representation and not *e.g.* spatial matching [13],[3], query expansion [5],[18], feature augmentation [19],[1] or nearest neighbor re-ranking [15],[7].

The first group of methods in Table 3 relies on a large vocabulary (1M or more) and in general does not include any descriptor signature. Note that [13] differs from [14] mainly in the quality of local descriptors; the descriptors of [13] are used in most other works, including ours. One way to improve performance in this setting is to learn an even larger vocabulary (16M) on a much larger training set [12], which is a costly off-line process. Another way is to use the extremely fine partition of a multi-index [4], as in [2], which cannot be fully inverted so there are four additional bytes as a descriptor signature.

The second group relies on a smaller vocabulary (100K or less) and embeds a descriptor signature, *e.g.* a Hamming code [9], as in [8],[17] and this work, or product quantization code [10], as in [16]. This is in most cases superior to the first group with an only slightly slower query, but requires 64 additional bits for the signature.

The third group includes ASMK* [17] and this work. Here the descriptor signature is still 64 bits, but the number of descriptors is reduced compared to all previous methods, as indicated by aggregation%, which is different for each dataset. Despite the lower memory and faster query, these methods are in general superior to all previous ones. Additionally, we get a significant performance gain of 7% in *Holidays* over [17] and a lower gain of up to 1% in the remaining datasets. The former is obtained by starting from the larger feature set and aggregating such that the total number of features is not higher than in [17], as shown in Fig. 7; the latter is due to the absolute performance improvement shown in Fig. 8.

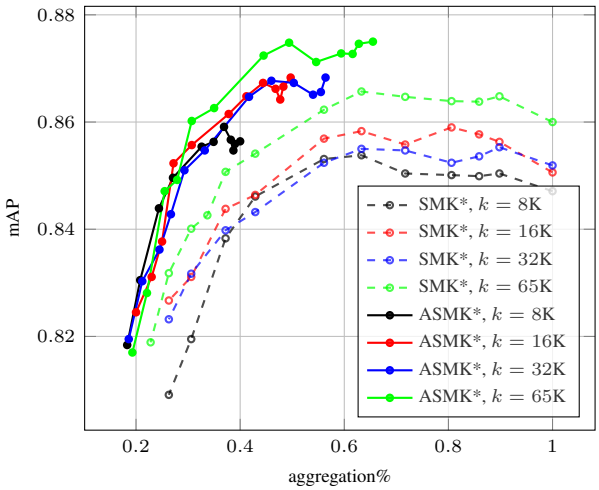


Figure 3. SMK*/ASMK* performance *vs.* aggregation% on *Holidays-L* for different vocabulary sizes $k$. Selectivity exponent $\alpha = 3$. Asymmetric aggregation.

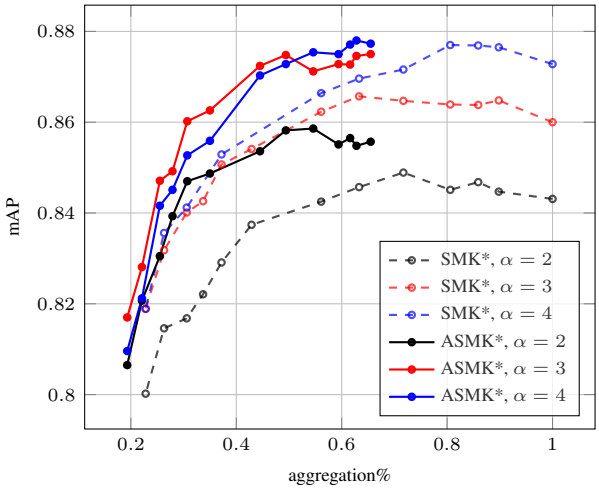


Figure 4. SMK*/ASMK* performance *vs.* aggregation% on *Holidays-L* for different values of selectivity exponent $\alpha$. Vocabulary size $k = 65$K. Asymmetric aggregation.

## References


[1] R. Arandjelovic and A. Zisserman. Three things everyone should know to improve object retrieval. In *CVPR*, 2012. 3

[2] Y. Avrithis. Quantize and conquer: A dimensionality-recursive solution to clustering, vector quantization, and image retrieval. In *ICCV*. 2013. 3, 4

[3] Y. Avrithis and G. Tolias. Hough pyramid matching: Speeded-up geometry re-ranking for large scale image retrieval. *IJCV*, 107(1):1–19, 2014. 3

[4] A. Babenko and V. Lempitsky. The inverted multi-index. In *CVPR*, 2012. 3

[5] O. Chum, A. Mikulik, M. Perdoch, and J. Matas. Total recall

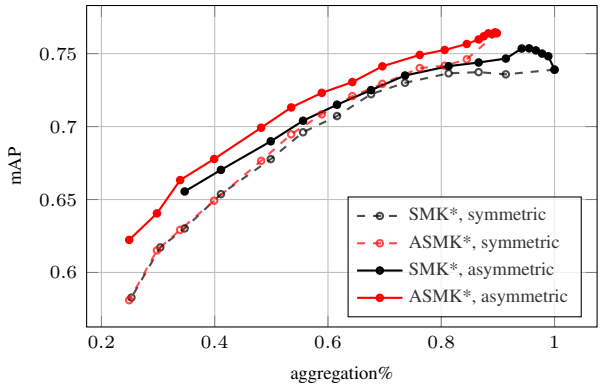


Figure 5. SMK*/ASMK* performance *vs*. aggregation% on *Oxford* for symmetric and asymmetric aggregation. Vocabulary size $k = 65$K; selectivity exponent $\alpha = 3$.

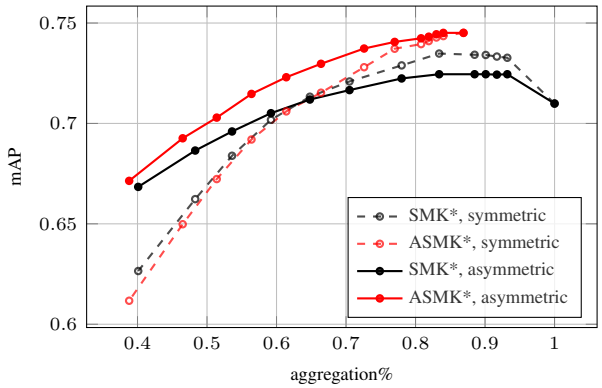


Figure 6. SMK*/ASMK* performance *vs*. aggregation% on *Paris* for symmetric and asymmetric aggregation. Vocabulary size $k = 65$K; selectivity exponent $\alpha = 3$.

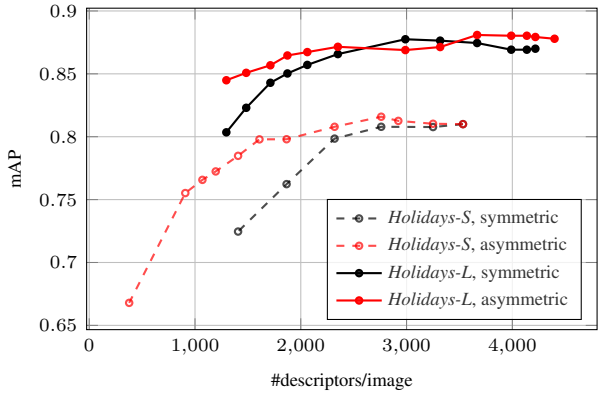


Figure 7. ASMK* performance *vs*. average number of aggregated descriptors per image on *Holidays* with multiple assignment for two initial feature sets and for symmetric and asymmetric aggregation. Vocabulary size $k = 65$K; selectivity exponent $\alpha = 3$.

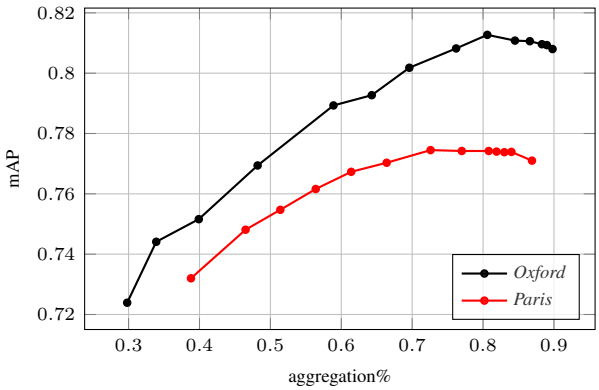


Figure 8. ASMK* performance *vs*. aggregation% on *Oxford* and *Paris* with multiple assignment. Vocabulary size $k = 65$K; selectivity exponent $\alpha = 3$. Asymmetric aggregation.

| Dataset | MA | Hol. | Paris | Oxf. |
|---|---|---|---|---|
| BoW [14] | | - | - | 40.3 |
| BoW [14] | ✓ | - | - | 49.3 |
| BoW [13] | | - | - | 55.8 |
| Fine vocab. [12] | | 74.9 | 74.9 | 74.2 |
| Multi-index [2] | ✓ | - | 69.6 | 70.3 |
| HE [9] | | 74.5 | - | 51.7 |
| HE [9] | ✓ | 77.5 | - | 56.1 |
| AHE+burst [8] | | 79.4 | - | 66.0 |
| AHE+burst [8] | ✓ | 81.9 | - | 69.8 |
| Query ad. [16] | | 81.4 | 70.3 | 73.9 |
| Query ad. [16] | ✓ | 82.1 | 73.6 | 78.0 |
| *aggregation%* | | 78% | 86% | 89% |
| ASMK* [17] | | 80.0 | 74.4 | 76.4 |
| ASMK* [17] | ✓ | 81.0 | 77.0 | 80.4 |
| This work | ✓ | **88.1** | **77.5** | **81.3** |

Table 3. Comparison of our best mAP result to state-of-the-art using inverted files as in BoW or also local descriptors as in HE. We only report results for the initial large-scale ranking (no spatial re-ranking).

II: Query expansion revisited. In *CVPR*, 2011. 3

[6] J. Delhumeau, P. Gosselin, H. Jégou, and P. Perez. Revisiting the VLAD image representation. In *ACM Multimedia*, 2013. 2

[7] A. Delvinioti, H. Jégou, L. Amsaleg, and M. E. Houle. Image retrieval with reciprocal and shared nearest neighbors. In *VISAPP*, 2014. 3

[8] M. Jain, H. Jégou, and P. Gros. Asymmetric hamming embedding. In *ACM Multimedia*, 2011. 3, 4

[9] H. Jégou, M. Douze, and C. Schmid. Improving bag-of-features for large scale image search. *IJCV*, 87(3):316–336, 2010. 2, 3, 4

[10] H. Jégou, M. Douze, and C. Schmid. Product quantization for nearest neighbor search. *PAMI*, 33(1):117–128, 2011. 3

[11] H. Jégou, F. Perronnin, M. Douze, J. Sanchez, P. Perez, and C. Schmid. Aggregating local image descriptors into com-

pact codes. *PAMI*, 34(9):1704–1716, 2012. 1

[12] A. Mikulik, M. Perdoch, O. Chum, and J. Matas. Learning a fine vocabulary. In *ECCV*, 2010. 3, 4

[13] M. Perdoch, O. Chum, and J. Matas. Efficient representation of local geometry for large scale object retrieval. In *CVPR*, 2009. 3, 4

[14] J. Philbin, O. Chum, J. Sivic, M. Isard, and A. Zisserman. Lost in quantization: Improving particular object retrieval in large scale image databases. In *CVPR*, 2008. 3, 4

[15] D. Qin, S. Gammeter, L. Bossard, T. Quack, and L. Van Gool. Hello neighbor: Accurate object retrieval with k-reciprocal nearest neighbors. In *CVPR*, 2011. 3

[16] D. Qin, C. Wengert, and L. Van Gool. Query adaptive similarity for large scale object retrieval. In *CVPR*, 2013. 3, 4

[17] G. Tolias, Y. Avrithis, and H. Jégou. To aggregate or not to aggregate: Selective match kernels for image search. In *ICCV*, 2013. 2, 3, 4

[18] G. Tolias and H. Jégou. Visual query expansion with or without geometry: refining local descriptors by feature aggregation. *Pattern Recognition*, 47(10):3466–3476, 2014. 3

[19] P. Turcot and D. Lowe. Better matching with fewer features: the selection of useful features in large database recognition problems. In *ICCV*, 2009. 3