Higher-order Occurrence Pooling for Bags-of-Words: Visual Concept Detection

Piotr Koniusz, Fei Yan, Philippe-Henri Gosselin, Krystian Mikolajczyk

Higher-order Occurrence Pooling for Bags-of-Words: Visual Concept Detection

Piotr Koniusz, Fei Yan, Philippe-Henri Gosselin, Krystian Mikolajczyk

Abstract—In object recognition, the Bag-of-Words model assumes: i) extraction of local descriptors from images, ii) embedding the descriptors by a coder to a given visual vocabulary space which results in mid-level features, iii) extracting statistics from mid-level features with a pooling operator that aggregates occurrences of visual words in images into signatures, which we refer to as First-order Occurrence Pooling. This paper investigates higher-order pooling that aggregates over co-occurrences of visual words. We derive Bag-of-Words with Higher-order Occurrence Pooling based on linearisation of Minor Polynomial Kernel, and extend this model to work with various pooling operators. This approach is then effectively used for fusion of various descriptor types. Moreover, we introduce Higher-order Occurrence Pooling performed directly on local image descriptors as well as a novel pooling operator that reduces the correlation in the image signatures. Finally, First-, Second-, and Third-order Occurrence Pooling are evaluated given various coders and pooling operators on several widely used benchmarks. The proposed methods are compared to other approaches such as Fisher Vector Encoding and demonstrate improved results.

Index Terms—Bag-of-Words, Mid-level features, First-order, Second-order, Co-occurrence, Pooling Operator, Sparse Coding

1 Introduction

Bag-of-Words [1], [2] (BoW) is a popular approach which transforms local image descriptors [3], [4], [5] into image representations that are used in retrieval and classification. To date, a number of its variants have been developed and reported to produce state-of-the-art results: Kernel Codebook [6], [7], [8], [9] a.k.a. Soft Assignment and Visual Word Uncertainty, Approximate Locality-constrained Soft Assignment [10], [11], Sparse Coding [12], [13], Local Coordinate Coding [14], and Approximate Locality-constrained Linear Coding [15]. We refer to this group as standard BoW. A second group improving upon BoW approaches includes Super Vector Coding [16], Vector of Locally Aggregated Descriptors [17], Fisher Vector Encoding [18], [19], and Vector of Locally Aggregated Tensors [20]. The main hallmarks of this group, in contrast to standard BoW, are: i) encoding of descriptors relative to the centres of their clusters, ii) extraction of second-order statistics from mid-level features to complement the first-order cues, iii) pooling with Power Normalisation [19], [21] which counteracts so-called *burstiness* [11], [22].

Several evaluations of various BoW models [11], [23], [24], [25], [26] address multiple aspects of BoW. A recent review of coding schemes [24] includes Hard Assignment, Soft Assignment, Approximate Locality-constrained Linear Coding, Super Vector Coding, and Fisher Vector Encoding. The role of pooling operators has been studied in [11], [25], [26] which lead to improvements in object category recognition. A detailed comparison of BoW [11] shows that the choice of pooling for various coders affects the classification performance. All evaluations highlight that the second group of methods, *e.g.* Fisher Vector Encoding perform significantly better than the standard BoW approaches.

The pooling step in standard BoW aggregates only first-order occurrences of visual words in the mid-level features and max-pooling [13] is often combined with various coding schemes [7], [13], [14]. In this paper, we study the BoW model according to the coding and pooling techniques and present ideas that improve the performance. The analysis of First-, Second-, and Third-order Occurrence Pooling in the BoW model constitutes the main contribution of this work. In more detail:

1) We propose Higher-order Occurrence Pooling that aggregates co-occurrences rather than occurrences of visual words in mid-level features, which leads to more discriminative representation. It is also presented as a novel approach for fusion of various descriptor types.
2) We introduce a new descriptor termed *residual* in the context of higher-order occurrence pooling that improves the accuracy of mid-level features as well as a novel pooling operator based on Higher Order Singular Value Decomposition [27], [28] and Power Normalisation [21].
3) We present an evaluation of First-, Second-, and Third-order Occurrence Pooling combined with several coding schemes and produce state-of-the-art BoW results on several standard benchmarks. We outperform Fisher Vector Encoding [19], [29] (FV), Vector of Locally Aggregated Tensors [20] (VLAT), Spatial Pyramid Matching [13], [30], and coder-free Second-order Pooling from [31].

Our method is somewhat inspired by Vector of Locally Aggregated Tensors [20] (VLAT) that also models the co-occurrences of features. Note that VLAT differs from VLAD [17] by employing second-order statistics. In con-

- *P. Koniusz graduated from CVSSP, Guildford, UK, he is currently with CVG, NICTA, Canberra, Australia. See http://claret.wikidot.com*
- *F. Yan is at Centre for Vision, Speech and Signal Processing (CVSSP), University of Surrey, Guildford, UK.*
- *P. H. Gosselin is with ETIS at Ecole Nationale Superieure de l'Electronique et de ses Applications, Cergy, France.*
- *K. Mikolajczyk is with ICVL at Imperial College London, UK.*

trast to VLAT, our approach allows to incorporate arbitrary coders and pooling operators. It also differs from the recently proposed Second-order Pooling applied to the low-level descriptors for segmentation [31]. In contrast, we perform pooling on the mid-level features thus still preserve the benefits from the coding step in BoW. In addition, we propose and evaluate Higher-order Occurrence Pooling. Moreover, unlike 2D histogram representation [32], which is another take on building rich statistics from the mid-level features, our approach results from the analytical solution to the kernel linearisation problems.

Recently, a significant change in the approach and performance of image recognition has been observed. Methods based on Deep Neural Networks [33] have started to dominate the field and little attention is now given to the BoW model. This paper discusses the latest achievements in BoW and demonstrates the performance on standard benchmarks. It also compares the results from BoW techniques to DNN methods.

The reminder of this section first introduces the standard model of Bag-of-Words in section 1.1. The coders and pooling operators used are presented in sections 1.2 and 1.3. The rest of this paper is organised as follows. Section 2 describes the BoW with Higher-order Occurrence Pooling. Section 3 proposes a new fusion method for various descriptors based on Higher-order Occurrence Pooling. Section 4 discusses Third-order Occurrence Pooling on low-level descriptors. Section 5 presents our experimental evaluations.

1.1 Bag-of-Words Model

Let us denote low-level descriptor vectors, such as SIFT [3], as $\boldsymbol{x}_n \in \mathbb{R}^D$ such that $n = 1, ..., N$, where N is the total descriptor cardinality for the entire image set $\mathcal{I}$, and D is the descriptor dimensionality. Given any image $i \in \mathcal{I}$, $\mathcal{N}^i$ denotes a set of its descriptor indices. We drop the superscript for simplicity and use $\mathcal{N}$. Therefore, $\{\boldsymbol{x}_n\}_{n\in\mathcal{N}}$ denotes a set of descriptors for an image $i \in \mathcal{I}$. Next, we assume $k = 1, ..., K$ visual appearance prototypes $\boldsymbol{m}_k \in \mathbb{R}^D$ a.k.a. visual vocabulary, words, centres, atoms, or anchors. We form a dictionary $\boldsymbol{\mathcal{M}} = \{\boldsymbol{m}_k\}_{k=1}^K$, where $\boldsymbol{\mathcal{M}} \in \mathbb{R}^{D\times K}$ can also be seen as a matrix with its columns formed by visual words. This is illustrated in figure 1. Following the formalism of [11], [25], we express the standard BoW approaches as a combination of the mid-level coding and pooling steps, followed by the ℓ_2 norm normalisation:

$$\boldsymbol{\phi}_n = f(\boldsymbol{x}_n, \boldsymbol{\mathcal{M}}), \quad \forall n \in \mathcal{N} \tag{1}$$

$$\hat{h}_k = g\left(\{\phi_{kn}\}_{n\in\mathcal{N}}\right) \tag{2}$$

$$\boldsymbol{h} = \hat{\boldsymbol{h}}/\|\hat{\boldsymbol{h}}\|_2 \tag{3}$$

A mapping function $f : \mathbb{R}^D \to \mathbb{R}^K$ in Equation (1), *e.g.* Sparse Coding [12] embeds each descriptor $\boldsymbol{x}_n$ into the visual vocabulary space $\boldsymbol{\mathcal{M}}$ resulting in mid-level features $\boldsymbol{\phi}_n \in \mathbb{R}^K$. Pooling operation $g : \mathbb{R}^{|\mathcal{N}|} \to \mathbb{R}$ in Equation (2), *e.g.* Average or Max-pooling, aggregates occurrences of visual words in mid-level features, and therefore in an image. It uses all coefficients ϕ_{kn} from visual word $\boldsymbol{m}_k$ for image i to obtain k^{th} coefficient in vector $\hat{\boldsymbol{h}} \in \mathbb{R}^K$. Note that $\boldsymbol{\phi}_n$ denotes an n^{th} mid-level feature vector while ϕ_{kn} denotes its k^{th} coefficient. This formulation can also be extended to pooling over cells of Spatial Pyramid Matching as in [11]. Finally, signature $\hat{\boldsymbol{h}}$ is normalised in Equation (3). Signatures $\boldsymbol{h}_i, \boldsymbol{h}_j \in \mathbb{R}^K$ for $i, j \in \mathcal{I}$ form a linear kernel $Ker_{ij} = (\boldsymbol{h}_i)^T \cdot \boldsymbol{h}_j$ used by a classifier. This model of BoW assumes First-order Occurrence Pooling and often uses SC, LLC, and LcSA coders that are briefly presented below.

1.2 Mid-level Coders

The mid-level coders f are presented in this section. Note that in $\boldsymbol{x}_n$ and $\boldsymbol{\phi}_n$ we skip index $_n$ to avoid clutter.

Sparse Coding (SC) [12], [13] expresses each local descriptor $\boldsymbol{x}$ as a sparse linear combination of the visual words from $\boldsymbol{\mathcal{M}}$. The following problem is solved:

$$\begin{aligned} &\boldsymbol{\phi} = \arg\min_{\bar{\boldsymbol{\phi}}} \left\|\boldsymbol{x} - \boldsymbol{\mathcal{M}}\bar{\boldsymbol{\phi}}\right\|_2^2 + \alpha\|\bar{\boldsymbol{\phi}}\|_1 \\ &s.\,t.\ \ \bar{\boldsymbol{\phi}} \geq 0 \end{aligned} \tag{4}$$

A low number of non-zero coefficients in $\boldsymbol{\phi}$, referred to as sparsity, is induced with the ℓ_1 norm and adjusted by constant α. We impose a non-negative constraint on $\boldsymbol{\phi}$ for compatibility with Analytical pooling [11], [26].

Approximate Locality-constrained Linear Coding (LLC) [15] addresses the non-locality phenomenonthat can occur in SC and is formulated as follows:

$$\begin{aligned} &\boldsymbol{\phi}^* = \arg\min_{\bar{\boldsymbol{\phi}}} \left\|\boldsymbol{x} - \boldsymbol{\mathcal{M}}\left(\boldsymbol{x}, l\right)\bar{\boldsymbol{\phi}}\right\|_2^2 \\ &s.\,t.\ \ \bar{\boldsymbol{\phi}} \geq 0,\ \ \mathbf{1}^T\bar{\boldsymbol{\phi}} = 1 \end{aligned} \tag{5}$$

Descriptor $\boldsymbol{x}$ is coded with its l-nearest visual words $\boldsymbol{\mathcal{M}}\left(\boldsymbol{x}, l\right) \in \mathbb{R}^{D\times l}$ found in dictionary $\boldsymbol{\mathcal{M}}$. Constant $l \ll K$ controls the locality of coding. Lastly, the resulting $\boldsymbol{\phi}^* \in \mathbb{R}^l$ is re-projected into the full length nullified mid-level feature $\boldsymbol{\phi} \in \mathbb{R}^K$.

Approximate Locality-constrained Soft Assignment (LcSA) [7], [10] is derived from Mixture of Gaussians [34] $G(\boldsymbol{x}; \boldsymbol{m}_k, \sigma)$ with equal mixing weights. The component membership probability is used as a coder:

$$\phi_k = \begin{cases} \frac{G(\boldsymbol{x};\boldsymbol{m}_k,\sigma)}{\sum_{\boldsymbol{m}'\in\boldsymbol{\mathcal{M}}(\boldsymbol{x},l)} G(\boldsymbol{x};\boldsymbol{m}',\sigma)} & \text{if } \boldsymbol{m}_k \in \boldsymbol{\mathcal{M}}\left(\boldsymbol{x}, l\right) \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

where ϕ_k is computed from the l-nearest Gaussian components of $\boldsymbol{x}$ that are found in dictionary $\boldsymbol{\mathcal{M}}$.

Fisher Vector Encoding (FV) [18], [19] uses a Mixture of K Gaussians as a dictionary. It performs descriptor coding w.r.t. to Gaussian components $G(w_k, \boldsymbol{m}_k, \boldsymbol{\sigma}_k)$ which are parametrised by mixing probability, mean, and on-diagonal standard deviation. The first- and second-order features $\boldsymbol{\phi}_k, \boldsymbol{\psi}_k \in \mathbb{R}^D$ are :

$$\boldsymbol{\phi}_k = (\boldsymbol{x} - \boldsymbol{m}_k)/\boldsymbol{\sigma}_k, \quad \boldsymbol{\psi}_k = \boldsymbol{\phi}_k^2 - 1 \tag{7}$$

Concatenation of per-cluster features $\bar{\boldsymbol{\phi}}_k \in \mathbb{R}^{2D}$ forms the mid-level feature $\boldsymbol{\phi} \in \mathbb{R}^{2KD}$:

$$\boldsymbol{\phi} = \left[\bar{\boldsymbol{\phi}}_1^T, ..., \bar{\boldsymbol{\phi}}_K^T\right]^T, \quad \bar{\boldsymbol{\phi}}_k = \frac{p\left(\boldsymbol{m}_k|\boldsymbol{x}, \theta\right)}{\sqrt{w_k}} \begin{bmatrix} \boldsymbol{\phi}_k \\ \boldsymbol{\psi}_k/\sqrt{2} \end{bmatrix} \tag{8}$$

where p and θ are the component membership probabilities and parameters of GMM, respectively. Note that this formulation is compatible with equation (1) except that $\boldsymbol{\phi}$ is $2KD$

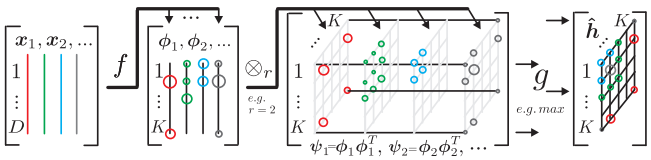


Fig. 1. Bag-of-Words with Second-order Occurrence Pooling (order $r=2$). Local descriptors $\boldsymbol{x}$ are extracted from an image and coded by f that operates on columns. Circles of various sizes illustrate values of mid-level coefficients. Self-tensor product $\otimes_r$ computes co-occurrences of visual words for every mid-level feature $\boldsymbol{\phi}$. Pooling g aggregates visual words from the co-occurrence matrices $\boldsymbol{\psi}$ along the direction of stacking.

rather than K dimensional. Also, $\boldsymbol{\phi}$ contains second-order statistics unlike codes of SC, LLC, and LcSA.

Vector of Locally Aggregated Tensors (VLAT) [20] also has a coding step that yields the first- and second-order features $\boldsymbol{\phi}_k \in \mathbb{R}^D$ and $\boldsymbol{\Psi}_k \in \mathbb{R}^{D\times D}$ per cluster:

$$\boldsymbol{\phi}_k = \boldsymbol{x} - \boldsymbol{m}_k, \quad \boldsymbol{\Psi}_k = \boldsymbol{\phi}_k \boldsymbol{\phi}_k{}^T - \boldsymbol{C}_k \tag{9}$$

In contrast to Vectors of Locally Aggregated Descriptors [17] that employs first-order occurrences, only the second-order matrices $\boldsymbol{\Psi}_k$ are used to form the mid-level features after normalisation with covariance matrices $\boldsymbol{C}_k$ of k-means clusters. Each $\boldsymbol{\Psi}_k$ is symmetric, thus the upper triangles and diagonals are extracted and unfolded by operator $u{:}$ to form vector $\boldsymbol{\phi}$:

$$\boldsymbol{\phi} = \left[u{:}(\boldsymbol{\Psi}_1)^T, ..., u{:}(\boldsymbol{\Psi}_K)^T \right]^T \tag{10}$$

This formulation is also compatible with equation (1) except that $\boldsymbol{\phi}$ is $KD(D+1)/2$ dimensional.

1.3 Pooling Operators

In BoW, pooling operators aggregate occurrences of visual words represented by the coefficients of mid-level feature vectors. The typically used pooling operators are presented below.

Average pooling [2] counts the number of descriptor assignments to each cluster k or visual word $\boldsymbol{m}_k$ and normalises such counts by the number of descriptors in the image. It is used with SC, LLC, LcSA, FV, VLAT and is defined as:

$$\hat{h}_k = \operatorname{avg}\left(\{\phi_{kn}\}_{n\in\mathcal{N}}\right) = \frac{1}{|\mathcal{N}|}\sum_{n\in\mathcal{N}} \phi_{kn} \tag{11}$$

Max-pooling [10], [13], [25], [26] selects the largest value from $|\mathcal{N}|$ mid-level feature coefficients corresponding to visual word $\boldsymbol{m}_k$:

$$\hat{h}_k = \max\left(\{\phi_{kn}\}_{n\in\mathcal{N}}\right) \tag{12}$$

To detect occurrences of visual words, Max-pooling is often combined with SC, LLC, and LcSA coders. It is not applicable to FV or VLAT, as their mid-level feature coefficients do not represent visual words.

MaxExp [26] represents a *theoretical expectation of Max-pooling* inspired by a statistical model. The mid-level feature coefficients for a given $\boldsymbol{m}_k$ are presumed to be drawn at random from Bernoulli distribution under the i.i.d. assumption. From binomial distribution, given exactly $\bar{N} = |\mathcal{N}|$ trials, the probability of at least one visual word $\boldsymbol{m}_k$ present in an image is:

$$\hat{h}_k = 1 - (1 - h_k^*)^{\bar{N}}, \quad h_k^* = \operatorname{avg}\left(\{\phi_{kn}\}_{n\in\mathcal{N}}\right) \tag{13}$$

This operator aggregates $\bar{N}$ independent features but number $\bar{N} \leq |\mathcal{N}|$ has to be found by cross-validation. MaxExp is typically used with SC, LLC, and LcSA as constraint $0 \leq h_k^* \leq 1$ does not hold for FV or VLAT.

Power Normalisation a.k.a. Gamma [19], [21], [22] approximates the statistical model of MaxExp as shown in [11] and is used by SC, LLC, LcSA, FV, VLAT :

$$\hat{h}_k = \operatorname{sgn}(h_k^*)\,|h_k^*|^\gamma, \quad h_k^* = \operatorname{avg}\left(\{\phi_{kn}\}_{n\in\mathcal{N}}\right) \tag{14}$$

The influence of statistical dependence between features is controlled by $0 < \gamma \leq 1$.

@n pooling [11] was designed to suppress the low values of mid-level feature coefficients that are considered a noise in SC, LLC, and LcSA. This operator is based on MaxExp and considered a trade-off between Max-pooling and Analytical pooling [11], [26]:

$$\hat{h}_k = 1 - (1 - h_k^*)^{\bar{N}}, \quad h_k^* = \operatorname{avg}\operatorname{srt}\left(\{\phi_{kn}\}_{n\in\mathcal{N}}, @n\right) \tag{15}$$

The $@n$ largest mid-level features are selected by partial sort algorithm srt and averaged by avg. $\bar{N}$ is the number of averaged features such that $1 \leq \bar{N} \leq @n \leq |\mathcal{N}|$. The mid-level feature coefficients for any $\boldsymbol{m}_k$ are presumed to be drawn from a Bernoulli distribution under the i.i.d. assumption. Given exactly $\bar{N} = @n$ trials, equation (15) yields the probability of *at least one visual word $\boldsymbol{m}_k$ present amongst the $@n$ largest mid-level feature coefficients*. Assuming that large ϕ_{kn} correspond to visual words $\boldsymbol{m}_k$, $@n$ cuts off small ϕ_{kn} that originate from noise. This does not hold for FV or VLAT as their small ϕ_{kn} are not considered noise.

2 Bag-of-Words with Higher-Order Occurrence Pooling

In this section, we introduce the Higher-Order Occurrence Pooling for BoW with its derivation in sections 2.1 and 2.2.

Bag-of-Words typically uses First-order Occurrence Pooling with the coding and pooling operators discussed in section 1. In contrast, FV and VLAT benefit from the second-order statistics. We therefore equip BoW with the second- or higher-order statistics. Figure 1 illustrates the proposed BoW with Second-order Occurrence Pooling. First, we perform coding (cf. equation (1)), then embed the second- or higher-order statistics by replacing equation (2) with:

$$\boldsymbol{\psi}_n = u{:}\left(\otimes_r \boldsymbol{\phi}_n\right) \tag{16}$$

$$\hat{h}_k = g\left(\{\psi_{kn}\}_{n\in\mathcal{N}}\right) \tag{17}$$

Equation (16) represents self-tensor product $\otimes_r$ performed on every mid-level feature vector $\boldsymbol{\phi}_n$, where $r \geq 1$ is a chosen order. This computes higher-order occurrences (or co-occurrences) of visual words in every mid-level feature. For $r=1$, the above formulation is reduced to the standard BoW as $\boldsymbol{\psi}_n = \boldsymbol{\phi}_n = \otimes_1(\boldsymbol{\phi}_n)$. For $r>1$, the resulting $\boldsymbol{\psi}_n$ are symmetric matrices, therefore only half of the coefficients are retained and unfolded into vectors with operator $u{:}$. Specifically, one can extract the upper triangle and diagonal

for $\otimes_2$ or the upper pyramid and diagonal plane for $\otimes_3$. Therefore, the dimensionality mid-level features based on self-tensor product is $K^{(r)} = \binom{K+r-1}{r}$.

Equation (17) performs pooling similar to equation (2), however, this time g aggregates co-occurrences or higher-order relations of visual words in mid-level features. Function $g : \mathbb{R}^{|\mathcal{N}|} \rightarrow \mathbb{R}$ takes k^{th} higher-order coefficients ψ_{kn} for all $n \in \mathcal{N}$ in an image to produce a k^{th} coefficient in vector $\hat{\boldsymbol{h}} \in \mathbb{R}^{K^{(r)}}$, where $k = 1, ..., K^{(r)}$.

Lastly, the normalisation from equation (3) is applied to $\hat{\boldsymbol{h}}$. The resulting signatures $\boldsymbol{h}$ are of dimensionality $K^{(r)}$ that depends on the dictionary size K and rank r. Note that sizes of FV and VLAT signatures depend on K and D (descriptor dimensionality).

2.1 Linearisation of Minor Polynomial Kernel

BoW with Higher-order Occurrence Pooling can be derived analytically by performing the following steps: i) defining a kernel function on a pair of mid-level features $\boldsymbol{\phi}$ and $\bar{\boldsymbol{\phi}}$, referred to as Minor Kernel, ii) summing over all pairs of mid-level features formed from a pair of images, iii) normalising sums by the feature counts and, iv) normalising the resulting kernel. First, we define Minor Polynomial Kernel:

$$ker\left(\boldsymbol{\phi}, \bar{\boldsymbol{\phi}}\right) = \left(\beta \boldsymbol{\phi}^T \bar{\boldsymbol{\phi}} + \lambda\right)^r \tag{18}$$

We chose $\beta = 1$ and $\lambda = 0$, while $r \geq 1$ denotes the polynomial degree and the order of occurrence pooling. Equation (18) reduces to the dot product $ker\left(\boldsymbol{\phi}, \bar{\boldsymbol{\phi}}\right) = \left\langle \boldsymbol{\phi}, \bar{\boldsymbol{\phi}} \right\rangle^r$ of a pair of mid-level features. The mid-level features result from $\mathcal{N}$ and $\bar{\mathcal{N}}$ descriptors in two images. We assume $\boldsymbol{\phi}$ and $\bar{\boldsymbol{\phi}}$ are the ℓ_2 normalised. We define a kernel function between two sets $\boldsymbol{\Phi} = \{\boldsymbol{\phi}_n\}_{n \in \mathcal{N}}$ and $\bar{\boldsymbol{\Phi}} = \left\{\bar{\boldsymbol{\phi}}_{\bar{n}}\right\}_{\bar{n} \in \bar{\mathcal{N}}}$:

$$\begin{aligned} Ker\left(\boldsymbol{\Phi}, \bar{\boldsymbol{\Phi}}\right) &= \frac{1}{|\mathcal{N}|} \sum_{n \in \mathcal{N}} \frac{1}{|\bar{\mathcal{N}}|} \sum_{\bar{n} \in \bar{\mathcal{N}}} \left\langle \boldsymbol{\phi}_n, \bar{\boldsymbol{\phi}}_{\bar{n}} \right\rangle^r \\ &= \frac{1}{|\mathcal{N}|} \sum_{n \in \mathcal{N}} \frac{1}{|\bar{\mathcal{N}}|} \sum_{\bar{n} \in \bar{\mathcal{N}}} \left(\sum_{k=1}^{K} \phi_{kn} \bar{\phi}_{k\bar{n}} \right)^r \end{aligned} \tag{19}$$

The rightmost summation in equation (19) can be expressed as a dot product of two self-tensor products of order r:

$$\begin{aligned} \left(\sum_{k=1}^{K} \phi_{kn} \bar{\phi}_{k\bar{n}} \right)^r &= \sum_{k^{(1)}=1}^{K} ... \sum_{k^{(r)}=1}^{K} \phi_{k^{(1)}} \bar{\phi}_{k^{(1)}} \cdot ... \cdot \phi_{k^{(r)}} \bar{\phi}_{k^{(r)}} \\ &= \left\langle u_{:}(\otimes_r \boldsymbol{\phi}_n), u_{:}\left(\otimes_r \bar{\boldsymbol{\phi}}_{\bar{n}}\right) \right\rangle \end{aligned} \tag{20}$$

By combining equations (19) and (20) we have:

$$\begin{aligned} Ker\left(\boldsymbol{\Phi}, \bar{\boldsymbol{\Phi}}\right) &= \frac{1}{|\mathcal{N}|} \sum_{n \in \mathcal{N}} \frac{1}{|\bar{\mathcal{N}}|} \sum_{\bar{n} \in \bar{\mathcal{N}}} \left\langle u_{:}(\otimes_r \boldsymbol{\phi}_n), u_{:}\left(\otimes_r \bar{\boldsymbol{\phi}}_{\bar{n}}\right) \right\rangle \\ &= \left\langle \frac{1}{|\mathcal{N}|} \sum_{n \in \mathcal{N}} u_{:}(\otimes_r \boldsymbol{\phi}_n), \frac{1}{|\bar{\mathcal{N}}|} \sum_{\bar{n} \in \bar{\mathcal{N}}} u_{:}\left(\otimes_r \bar{\boldsymbol{\phi}}_{\bar{n}}\right) \right\rangle \\ &= \left\langle \underset{n \in \mathcal{N}}{\text{avg}} \left[u_{:}(\otimes_r \boldsymbol{\phi}_n) \right], \underset{\bar{n} \in \bar{\mathcal{N}}}{\text{avg}} \left[u_{:}\left(\otimes_r \bar{\boldsymbol{\phi}}_{\bar{n}}\right) \right] \right\rangle \end{aligned} \tag{21}$$

Finally, $Ker\left(\boldsymbol{\Phi}, \bar{\boldsymbol{\Phi}}\right)$ is normalised to ensure self-similarity $Ker\left(\boldsymbol{\Phi}, \boldsymbol{\Phi}\right) = Ker\left(\bar{\boldsymbol{\Phi}}, \bar{\boldsymbol{\Phi}}\right) = 1$, by :

$$Ker\left(\boldsymbol{\Phi}, \bar{\boldsymbol{\Phi}}\right) := \frac{Ker\left(\boldsymbol{\Phi}, \bar{\boldsymbol{\Phi}}\right)}{\sqrt{Ker\left(\boldsymbol{\Phi}, \boldsymbol{\Phi}\right)} \sqrt{Ker\left(\bar{\boldsymbol{\Phi}}, \bar{\boldsymbol{\Phi}}\right)}} \tag{22}$$

The model derived in equation (21) is the BoW defined in equations (1), (16), and (17).

2.2 Beyond Average Pooling of Higher-order Occurrences

This section provides an extension of the proposed Higher-order Occurrence Pooling such that it can be combined with Max-pooling, which was reported to outperform the Average pooling in visual recognition [11], [13], [26]. We note that Average pooling counts all occurrences of a given visual word in an image, thus it quantifies areas spanned by repetitive patterns while max-pooling only detects the presence of a visual word in an image. Max-pooling was shown to be a lower bound of the likelihood of *at least one visual word* $\boldsymbol{m}_k$ *being present in an image* [10].

First, we define max operators on mid-level features: i) $\max_{n \in \mathcal{N}} \phi_n = \max\left(\{\phi_n\}_{n \in \mathcal{N}}\right)$ and ii) $\max_{n \in \mathcal{N}} \boldsymbol{\phi}_n$ as a vector formed from element-wise $\max\left(\{\phi_{1n}\}_{n \in \mathcal{N}}\right), \max\left(\{\phi_{2n}\}_{n \in \mathcal{N}}\right), ..., \max\left(\{\phi_{Kn}\}_{n \in \mathcal{N}}\right)$. Note that $\boldsymbol{\Phi} = \{\boldsymbol{\phi}_n\}_{n \in \mathcal{N}}$ and $\bar{\boldsymbol{\Phi}} = \left\{\bar{\boldsymbol{\phi}}_{\bar{n}}\right\}_{\bar{n} \in \bar{\mathcal{N}}}$ are two sets of mid-level features formed by $\mathcal{N}$ and $\bar{\mathcal{N}}$ descriptors from a pair of images. BoW with Max-pooling and Polynomial Kernel of degree r is given in equation (23) which is then expanded in equation (24) and simplified to a dot product between two vectors in equation (25) such that it forms a linear kernel. A lower bound of this kernel is proposed in equation (26), which represents Higher-order Occurrence Pooling with the Max-pooling operator. We further express it as a dot product between two vectors in equation (27).

$$Ker\left(\boldsymbol{\Phi}, \bar{\boldsymbol{\Phi}}\right) = \left\langle \hat{\boldsymbol{h}}, \hat{\bar{\boldsymbol{h}}} \right\rangle^r, \; and \; \begin{cases} \hat{h}_k = \max\left(\{\phi_{kn}\}_{n \in \mathcal{N}}\right) \\ \hat{\bar{h}}_k = \max\left(\{\bar{\phi}_{kn}\}_{\bar{n} \in \bar{\mathcal{N}}}\right) \end{cases}$$

$$= \left(\sum_{k=1}^{K} \max_{n \in \mathcal{N}}(\phi_{kn}) \cdot \max_{\bar{n} \in \bar{\mathcal{N}}}\left(\bar{\phi}_{k\bar{n}}\right) \right)^r \tag{23}$$

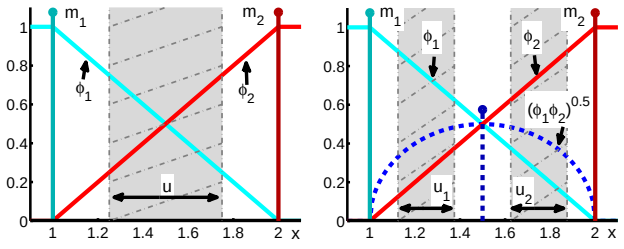
$$= \sum_{k^{(1)}=1}^{K} ... \sum_{k^{(r)}=1}^{K} \left(\begin{array}{c} \max_{n \in \mathcal{N}}(\phi_{k^{(1)}n}) \cdot ... \cdot \max_{n \in \mathcal{N}}(\phi_{k^{(r)}n}) \cdot \\ \cdot \max_{\bar{n} \in \bar{\mathcal{N}}}\left(\bar{\phi}_{k^{(1)}\bar{n}}\right) \cdot ... \cdot \max_{\bar{n} \in \bar{\mathcal{N}}}\left(\bar{\phi}_{k^{(r)}\bar{n}}\right) \end{array} \right) \tag{24}$$

$$= \left\langle u_{:}\left[\otimes_r \max_{n \in \mathcal{N}} \left(\boldsymbol{\phi}_n\right) \right], u_{:}\left[\otimes_r \max_{\bar{n} \in \bar{\mathcal{N}}} \left(\bar{\boldsymbol{\phi}}_{\bar{n}}\right) \right] \right\rangle \tag{25}$$

$$\geq \sum_{k^{(1)}=1}^{K} ... \sum_{k^{(r)}=1}^{K} \left(\begin{array}{c} \max_{n \in \mathcal{N}}(\phi_{k^{(1)}n} \cdot ... \cdot \phi_{k^{(r)}n}) \cdot \\ \cdot \max_{\bar{n} \in \bar{\mathcal{N}}}\left(\phi_{k^{(1)}\bar{n}} \cdot ... \cdot \bar{\phi}_{k^{(r)}\bar{n}}\right) \end{array} \right) \tag{26}$$

$$= \left\langle \max_{n \in \mathcal{N}}\left[u_{:}(\otimes_r \boldsymbol{\phi}_n)\right], \max_{\bar{n} \in \bar{\mathcal{N}}}\left[u_{:}\left(\otimes_r \bar{\boldsymbol{\phi}}_{\bar{n}}\right)\right] \right\rangle \tag{27}$$

The formulation with Average pooling, which preserves bi-linearity in equation (21), is convenient for the linearisation but breaking bi-linearity leads to improvements as demonstrated in [20]. Equation (27) introduces max-pooling that breaks bi-linearity and enables the use of other suitable operators for Higher-order Occurrence Pooling such as the $@n$ operator. An interesting probabilistic difference between the BoW models in equations (25) and (27) can be shown. We first consider Max-pooling in regular BoW with a linear kernel. If mid-level feature coefficients ϕ_{kn} are drawn from a feature distribution under the i.i.d. assumption given a visual word $\boldsymbol{m}_k$, the likelihood of *at least one visual word*



(*a*) first-order (*b*) second-order

Fig. 2. Uncertainty in Max-pooling. Mid-level feature coefficients ϕ_1 and ϕ_2 are produced for descriptors $1 \leq x \leq 2$ given visual words $m_1 = 1$ and $m_2 = 2$. (*a*) First-order Occurrence Pooling results in the pooling uncertainty u (the grey area). See text for explanations. (*b*) Second-order statistics produce co-occurrence component $(\phi_1\phi_2)^{0.5}$ that has a maximum for x indicated by the dashed stem. This component limits the pooling uncertainty. The square root is applied to preserve the linear slopes, *e.g.* $(\phi_1\phi_1)^{0.5} = \phi_1$.

$\boldsymbol{m}_k$ *being present in an image* [10] is an upper bound of Max-pooling:

$$1 - \prod_{n \in \mathcal{N}} (1 - \phi_{kn}) \geq \max\left(\{\phi_{kn}\}_{n \in \mathcal{N}}\right) \tag{28}$$

We now derive upper bounds of Max-pooling for the BoW models in equations (25) and (27). We denote the image signature from equation (25) as tensor $\boldsymbol{T} = \otimes_r \max_{n \in \mathcal{N}}(\boldsymbol{\phi}_n) \in \mathbb{R}^{K^r} = \{T_{k^{(1)}}, ..., T_{k^{(r)}}\}$. Coefficient-wise, it is:

$$T_{k^{(1)},...,k^{(r)}} = \prod_{s=1}^{r} \max\left(\{\phi_{k^{(s)}n}\}_{n \in \mathcal{N}}\right) \tag{29}$$

Each coefficient of an image signature obtained with Max-pooling and Polynomial Kernel is upper bounded by the probability of *visual words* $\boldsymbol{m}_{k^{(1)}}, ..., \boldsymbol{m}_{k^{(r)}}$ *jointly occurring at least once* ***after*** *pooling*:

$$T_{k^{(1)},...,k^{(r)}} \leq \prod_{s=1}^{r} \left(1 - \prod_{n \in \mathcal{N}} (1 - \phi_{k^{(s)}n})\right) \tag{30}$$

The image signature from equation (27) forms tensor $\boldsymbol{T}' = \max_{n \in \mathcal{N}}(\otimes_r \boldsymbol{\phi}_n) \in \mathbb{R}^{K^r}$. Coefficient-wise, this is:

$$T'_{k^{(1)},...,k^{(r)}} = \max\left(\left\{\prod_{s=1}^{r} \phi_{k^{(s)}n}\right\}_{n \in \mathcal{N}}\right) \tag{31}$$

Again, we note that each coefficient of an image signature is upper bounded by the probability of *visual words* $\boldsymbol{m}_{k^{(1)}}, ..., \boldsymbol{m}_{k^{(r)}}$ *jointly occurring in at least one mid-level feature* $\boldsymbol{\phi}_n$ ***before*** *pooling*:

$$T'_{k^{(1)},...,k^{(r)}} \leq 1 - \prod_{n \in \mathcal{N}} \left(1 - \prod_{s=1}^{r} \phi_{k^{(s)}n}\right) \tag{32}$$

Unlike the joint occurrence after pooling in equation (30), the joint occurrence of visual words computed in equation (32) before pooling can be interpreted as a new auxiliary element in the visual vocabulary.

Intuitive illustration. We argue that the joint occurrence of visual words in the mid-level features benefits Max-pooling (and other related operators) by limiting the uncertainty about the presence of descriptors. Figure 2 illustrates the mid-level coefficients given one dimensional visual words $m_1 = 1$ and $m_2 = 2$. It shows two linear slopes representing coding values ϕ_1 and ϕ_2 for any descriptor from range $1 \leq x_n \leq 2$. If all $x_n = 1.5$ then $\phi_{1n} = \phi_{2n} = 0.5$ for all n. Applying Max-pooling would result in $max(\{\phi_{1n}\}_{n \in \mathcal{N}}) = max(\{\phi_{2n}\}_{n \in \mathcal{N}}) = 0.5$. This signature uniquely indicates the presence of $x_n = 1.5$, therefore uncertainty of x_n location is $u = 0$. However, if x_n have different values from the given range, due to Max-pooling, the two largest coefficients ϕ_{1n} and ϕ_{2n} for the two descriptors closest to m_1 and m_2 would mask the presence of other descriptors, *i.e.* the mid-level signature would not contain any information about other descriptors. Thus, as $max(\{\phi_{1n}\}_{n \in \mathcal{N}}) \to 1$ and $max(\{\phi_{2n}\}_{n \in \mathcal{N}}) \to 1$, the uncertainty in location of other descriptors x_n is $u \to 1$. We argue that the role of pooling is to aggregate the mid-level features into a signature that preserves information about all descriptors.

Figure 2(b) extends the above example with the second-order statistics which, in addition to ϕ_1 and ϕ_2, introduces $\phi_1\phi_2$. Its maximum occurs for descriptor $x_n = 1.5$. We applied the square root $(\phi_1\phi_2)^{0.5}$ to preserve the linear slopes of ϕ_1 and ϕ_2 in the plot. Note that Max-pooling is applied to the individual $max(\{\phi_{1n}\}_{n \in \mathcal{N}})$ and $max(\{\phi_{2n}\}_{n \in \mathcal{N}})$ as well as to the joint term $max(\{\phi_{1n}\phi_{2n}\}_{n \in \mathcal{N}})$. This term indicates of the presence of descriptors in the mid-range between m_1 and m_2. The second-order statistics limit the uncertainty of Max-pooling such that $u_1 + u_2 \leq u$, thus increase the resolution of the visual dictionary.

3 Mid-level Feature Fusion with Higher-Order Occurrence Pooling

Shape, texture and colour cues are often combined for object category recognition [5], [19], [35], [36], [37], [38] and visual concept detection [11], [39], [40], [41], [42]. Some approaches employ so-called early fusion on the low-level descriptors [5], [35], [43]. Other methods apply coding and pooling on various modalities first, followed by so-called late fusion of multiple kernels [35], [36], [37], [38], [41].

We first formalise the early and late fusions, which are used as a baseline for comparisons to our fusion method based on Higher-order Occurrence Pooling. It captures the co-occurrences of visual words in each mid-level feature as shown in equation (27) of section 2.2. This is extended here to multiple descriptor types via linearisation of Minor Polynomial Kernel.

3.1 Early and Late Fusion

Early fusion is typically referred to when different types of low-level descriptors are concatenated. Such a fusion of various descriptors with their spatial coordinates was introduced in [43] as Spatial Coordinate Coding. It was presented as a low dimensional alternative to Spatial Pyramid Matching [30]. A similar fusion was used by others, *e.g.* in Joint Sparse Coding [44].

Low-level descriptor vector $\boldsymbol{x}$ and dictionary $\boldsymbol{\mathcal{M}}$ can be formed by concatenating Q descriptor types:

$$\begin{aligned} \boldsymbol{x} &= \left[\sqrt{\beta^{(1)}}\boldsymbol{x}^{(1)^T}, ..., \sqrt{\beta^{(Q)}}\boldsymbol{x}^{(Q)^T}\right]^T \\ \boldsymbol{\mathcal{M}} &= \left[\sqrt{\beta^{(1)}}\boldsymbol{\mathcal{M}}^{(1)^T}, ..., \sqrt{\beta^{(Q)}}\boldsymbol{\mathcal{M}}^{(Q)^T}\right]^T \end{aligned} \tag{33}$$

Weights $\beta^{(1)}, ..., \beta^{(Q)}$ determine the contribution of descriptor types $\boldsymbol{x}^{(1)}, ..., \boldsymbol{x}^{(Q)}$ given dictionaries $\boldsymbol{\mathcal{M}}^{(1)}, ..., \boldsymbol{\mathcal{M}}^{(Q)}$ and are typically learnt from the data.

Spatial Coordinate Coding (SCC) [43] is an example of early fusion which extends descriptors $\boldsymbol{x}$ with their spatial locations $\boldsymbol{x}^s = [c^x/w, c^y/h]^T$ normalised by the image width and height. Thus $\boldsymbol{x} := [\sqrt{\beta^s}\boldsymbol{x}^{s\,T}, \sqrt{1-\beta^s}\boldsymbol{x}^T]^T$.

Opponent SIFT [5] extracts gradient histograms at locations $\boldsymbol{x}^s$ from the grey and colour maps and forms vectors $\boldsymbol{x}$ and $\boldsymbol{x}^c$. All three terms are then concatenated $\boldsymbol{x} := [\sqrt{\beta^s}\boldsymbol{x}^{s\,T}, \sqrt{1-\beta^s-\beta^c}\boldsymbol{x}^T, \sqrt{\beta^c}\boldsymbol{x}^{c\,T}]^T$.

Late Fusion [36], [42] of multiple descriptor types is performed by independently coding and pooling each type and combining their kernels:

$$Ker_{ij} = \sum_{q=1}^{Q} \beta^{(q)} Ker_{ij}^{(q)} \tag{34}$$

There are various approaches to learn kernel weights $\beta^{(q)}$ [35], [38], [41]. However, for a small number of kernels, cross-validation can be used.

3.2 Linearisation of Minor Polynomial Kernel

The kernel linearisation for multiple descriptor types follows the approach detailed in section 2.1. We define Minor Polynomial Kernel:

$$ker\left(\{(\boldsymbol{\phi}^{(q)}, \bar{\boldsymbol{\phi}}^{(q)})\}_{q=1}^{Q}\right) = \left(\sum_{q=1}^{Q} \beta^{(q)} \boldsymbol{\phi}^{(q)\,T} \bar{\boldsymbol{\phi}}^{(q)} + \lambda\right)^r \tag{35}$$

There is one pair of mid-level features $(\boldsymbol{\phi}_n^{(q)}, \bar{\boldsymbol{\phi}}_{\bar{n}}^{(q)})$ per each descriptor type q with weights $\beta^{(q)}$ adjusting the contributions. Similarly to equation (19), we obtain:

$$ker\left(\{(\boldsymbol{\phi}^{(q)}, \bar{\boldsymbol{\phi}}^{(q)})\}_{q=1}^{Q}\right) = \left(\sum_{q=1}^{Q} \beta^{(q)} \left\langle \boldsymbol{\phi}^{(q)}, \bar{\boldsymbol{\phi}}^{(q)} \right\rangle\right)^r \tag{36}$$

The kernel function is evaluated between two images represented by two sets of mid-level features $\boldsymbol{\Phi} = \{\{\boldsymbol{\phi}_n^{(q)}\}_{n\in\mathcal{N}}\}_{q=1}^{Q}$ and $\bar{\boldsymbol{\Phi}} = \{\{\bar{\boldsymbol{\phi}}_{\bar{n}}^{(q)}\}_{\bar{n}\in\bar{\mathcal{N}}}\}_{q=1}^{Q}$ from $\mathcal{N}$ and $\bar{\mathcal{N}}$ descriptors and Q modalities:

$$\begin{aligned} Ker\left(\boldsymbol{\Phi}, \bar{\boldsymbol{\Phi}}\right) &= \frac{1}{|\mathcal{N}|}\sum_{n\in\mathcal{N}} \frac{1}{|\bar{\mathcal{N}}|}\sum_{\bar{n}\in\bar{\mathcal{N}}} \left(\sum_{q=1}^{Q} \beta^{(q)} \left\langle \boldsymbol{\phi}^{(q)}, \bar{\boldsymbol{\phi}}^{(q)} \right\rangle\right)^r \\ &= \frac{1}{|\mathcal{N}|}\sum_{n\in\mathcal{N}} \frac{1}{|\bar{\mathcal{N}}|}\sum_{\bar{n}\in\bar{\mathcal{N}}} \left(\sum_{q=1}^{Q} \beta^{(q)} \sum_{k=1}^{K} \phi_{kn}^{(q)} \bar{\phi}_{k\bar{n}}^{(q)}\right)^r \end{aligned} \tag{37}$$

Bi-modal Second-order Occurrence Pooling is first derived by linearising the above kernel for $Q=2$ (two coders) and $r=2$ (second-order). We denote $\beta^{(1)} = \beta$ and $\beta^{(2)} = 1-\beta$. Thus, by expanding the square term in equation (38), we obtain three dot products:

$$\left(\beta\sum_{k=1}^{K} \phi_{kn}^{(1)} \bar{\phi}_{k\bar{n}}^{(1)} + (1-\beta)\sum_{k=1}^{K} \phi_{kn}^{(2)} \bar{\phi}_{k\bar{n}}^{(2)}\right)^2 \tag{38}$$

$$= \beta^2 \left\langle u{:}(\boldsymbol{\phi}_n^{(1)} \boldsymbol{\phi}_n^{(1)\,T}), u{:}(\bar{\boldsymbol{\phi}}_{\bar{n}}^{(1)} \bar{\boldsymbol{\phi}}_{\bar{n}}^{(1)\,T}) \right\rangle \tag{39}$$

$$+ 2\beta(1-\beta) \left\langle u{:}(\boldsymbol{\phi}_n^{(1)} \boldsymbol{\phi}_n^{(2)\,T}), u{:}(\bar{\boldsymbol{\phi}}_{\bar{n}}^{(1)} \bar{\boldsymbol{\phi}}_{\bar{n}}^{(2)\,T}) \right\rangle \tag{40}$$

$$+ (1-\beta)^2 \left\langle u{:}(\boldsymbol{\phi}_n^{(2)} \boldsymbol{\phi}_n^{(2)\,T}), u{:}(\bar{\boldsymbol{\phi}}_{\bar{n}}^{(2)} \bar{\boldsymbol{\phi}}_{\bar{n}}^{(2)\,T}) \right\rangle \tag{41}$$

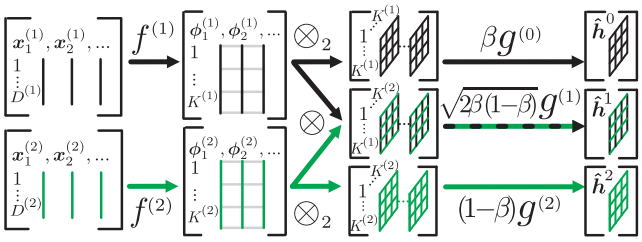


Fig. 3. Bi-modal Bag-of-Words with Second-order Occurrence Pooling. Two types of local descriptors $\boldsymbol{x}^{(1)}$ and $\boldsymbol{x}^{(2)}$ are extracted from an image and coded by coders $f^{(1)}$ and $f^{(2)}$. Self-tensor product $\otimes_2$ computes co-occurrences of visual words in every mid-level feature $\boldsymbol{\phi}^{(1)}$ and $\boldsymbol{\phi}^{(2)}$, respectively. Moreover, tensor product $\otimes$ captures co-occurrences of visual words between $\boldsymbol{\phi}^{(1)}$ and $\boldsymbol{\phi}^{(2)}$ (cross-term operation). Pooling g aggregates co-occurring visual words.

Combining these terms with equation (37) yields:

$$\begin{aligned} &Ker\left(\boldsymbol{\Phi}, \bar{\boldsymbol{\Phi}}\right) = \\ &= \beta^2 \Big\langle \operatorname*{avg}_{n\in\mathcal{N}} \left[u{:}(\boldsymbol{\phi}_n^{(1)} \boldsymbol{\phi}_n^{(1)\,T})\right], \operatorname*{avg}_{\bar{n}\in\bar{\mathcal{N}}} \left[u{:}(\bar{\boldsymbol{\phi}}_{\bar{n}}^{(1)} \bar{\boldsymbol{\phi}}_{\bar{n}}^{(1)\,T})\right] \Big\rangle \\ &+ 2\beta(1-\beta) \Big\langle \operatorname*{avg}_{n\in\mathcal{N}} \left[u{:}(\boldsymbol{\phi}_n^{(1)} \boldsymbol{\phi}_n^{(2)\,T})\right], \operatorname*{avg}_{\bar{n}\in\bar{\mathcal{N}}} \left[u{:}(\bar{\boldsymbol{\phi}}_{\bar{n}}^{(1)} \bar{\boldsymbol{\phi}}_{\bar{n}}^{(2)\,T})\right] \Big\rangle \\ &+ (1-\beta)^2 \Big\langle \operatorname*{avg}_{n\in\mathcal{N}} \left[u{:}(\boldsymbol{\phi}_n^{(2)} \boldsymbol{\phi}_n^{(2)\,T})\right], \operatorname*{avg}_{\bar{n}\in\bar{\mathcal{N}}} \left[u{:}(\bar{\boldsymbol{\phi}}_{\bar{n}}^{(2)} \bar{\boldsymbol{\phi}}_{\bar{n}}^{(2)\,T})\right] \Big\rangle \end{aligned} \tag{42}$$

Note that the first and the last terms represent Second-order Occurrence Pooling for independent coders $q=1$ and $q=2$ and correspond to equation (21) in section 2.1. However, the middle dot product represents the cross-term that captures additional information in form of the co-occurrences between visual words of mid-level features from two coders. Figure 3 illustrates this model.

Bi-modal Higher-order Occurrence Pooling can also be derived by expanding Minor Polynomial Kernel in equation (36). For order $r \geq 2$ and two coders $Q=2$, by substituting $a = \langle \boldsymbol{\phi}^{(1)}, \bar{\boldsymbol{\phi}}^{(1)} \rangle$ and $b = \langle \boldsymbol{\phi}^{(2)}, \bar{\boldsymbol{\phi}}^{(2)} \rangle$, one can expand Minor Polynomial Kernel in equation (36) using Binomial theorem:

$$\Big(\beta a + (1-\beta) b\Big)^r = \sum_{s=0}^{r} \binom{r}{s} \Big(\beta a\Big)^{r-s} \Big((1-\beta) b\Big)^s \tag{43}$$

The derivations follow the same steps as for Bi-modal Second-order Occurrence Pooling. We skip that for clarity and define Bag-of-Words with Bi-modal Higher-order Occurrence Pooling:

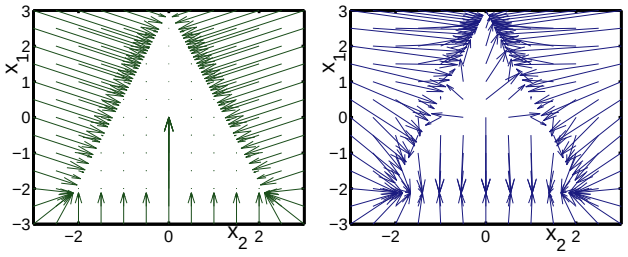
$$\begin{aligned} \boldsymbol{\phi}_n^{(1)} &= f^{(1)}\left(\boldsymbol{x}_n^{(1)}, \boldsymbol{\mathcal{M}}^{(1)}\right) \\ \boldsymbol{\phi}_n^{(2)} &= f^{(2)}\left(\boldsymbol{x}_n^{(2)}, \boldsymbol{\mathcal{M}}^{(2)}\right) \end{aligned}, \quad \forall n \in \mathcal{N} \tag{44}$$

$$\boldsymbol{\psi}_n^s = u{:}\Big[\left(\otimes_{r-s} \boldsymbol{\phi}_n^{(1)}\right)\left(\otimes_s \boldsymbol{\phi}_n^{(2)}\right)\Big], \; s = 0, ..., r \tag{45}$$

$$\hat{h}_k^s = \binom{r}{s}^{\frac{1}{2}} (1-\beta)^{\frac{s}{2}} \beta^{\frac{r-s}{2}} g^{(s)}\left(\{\psi_{kn}^s\}_{n\in\mathcal{N}}\right), \; k = 1, ..., K^{(r,s)} \tag{46}$$

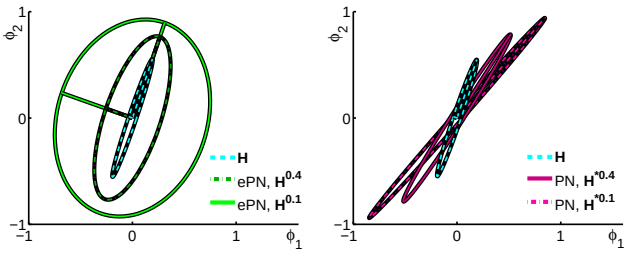
$$\boldsymbol{h} = \hat{\boldsymbol{h}}/\|\hat{\boldsymbol{h}}\|_2, \; \hat{\boldsymbol{h}} = \left[\hat{\boldsymbol{h}}^{\mathbf{0}\,T}, ..., \hat{\boldsymbol{h}}^{\boldsymbol{r}\,T}\right]^T \tag{47}$$

Equations (44) and (45) follow the terminology from equations (1) and (16) and represent the coding step for two coders. The coders can be of different types and their dictionary sizes $K^{(1)}$ and $K^{(2)}$ may differ. Equation (45) represents the joint occurrence of visual words in $\boldsymbol{\phi}_n^{(1)}$ or $\boldsymbol{\phi}_n^{(2)}$, or the cross-modal joint occurrence of visual words



(*a*) SC, α=1 (*b*) LcSA, σ^2=4, l=2

Fig. 4. Illustration of Residual Descriptors. Quantisation loss of the descriptors from their original positions $\boldsymbol{x}$ denoted by the grid points, to the corresponding reconstructed positions $\hat{\boldsymbol{x}}$ indicated by the arrows. (*a*) SC: optimal reconstruction (no displacement) within the triangle. (*b*) LcSA: poor reconstruction within the triangle due to low $l=2$.



(*a*) (*b*)

Fig. 5. Whitening of the autocorrelation matrix $\boldsymbol{H}$. (*a*) The eigenvalue- and (*b*) coefficient-wise Power Normalisation steps (*ePN*) and (*PN*) are shown. See $\boldsymbol{H}^{0.4}$, $\boldsymbol{H}^{0.1}$, $\boldsymbol{H}^{*0.4}$, and $\boldsymbol{H}^{*0.1}$, (*) is the element-wise power.

per mid-level pair $(\boldsymbol{\phi}_n^{(1)}, \boldsymbol{\phi}_n^{(2)})$. It results from an expansion of Minor Polynomial Kernel in equation (36) according to Binomial theorem in a similar way to equations (38-41). The dimensionality of $\boldsymbol{\psi}_n^s$ after removing repeated coefficients and unfolding is $K^{(r,s)} = K^{(r-s)}K^{(s)}$. Equation (46) represents pooling that aggregates the joint occurrences or the cross-modal joint occurrences of visual words. Function $g^{(s)} : \mathbb{R}^{|\mathcal{N}|} \rightarrow \mathbb{R}$ uses the k^{th} joint occurrence to produce the k^{th} coefficient in vector $\hat{\boldsymbol{h}}^s \in \mathbb{R}^{K^{(r,s)}}$. The weighting factors preceding $g^{(s)}$ result from Binomial expansion (cf. equation (43)). Equation (47) concatenates and normalises the joint occurrence statistics.

Multi-modal Higher-order Occurrence Pooling can be derived in the same way by using Multinomial instead of Binomial theorem and expanding Minor Polynomial Kernel in equation (36). The fusion can be performed by concatenating the mid-level features from Q coders:

$$\boldsymbol{\phi}_n = \left[\sqrt{\beta^{(1)}}\boldsymbol{\phi}_n^{(1)T}, \sqrt{\beta^{(2)}}\boldsymbol{\phi}_n^{(2)T}, ..., \sqrt{\beta^{(Q)}}\boldsymbol{\phi}_n^{(Q)T}\right]^T \quad (48)$$

Thus, mid-level features $\boldsymbol{\phi}_n$ form tensors (cf. equation (16)) and can form Bi- or Multi-modal Second- and Higher-order Occurrence Pooling.

3.3 Residual Descriptors

In this section, we introduce an approach based on Bi-modal Second-order Occurrence Pooling that further improves the accuracy of coding. Various coding approaches such as SC and LLC optimise a trade-off between the quantisation loss (defined below) and a chosen regularisation penalty, *e.g.* sparsity or locality as in equations (4) and (5). The quality of quantisation in coders can be measured based on the theory of Linear Coordinate Coding [14]. The linear approximation of descriptor $\boldsymbol{x}$ given dictionary $\boldsymbol{\mathcal{M}}$ and coder f that produces mid-level feature $\boldsymbol{\phi}$ is $\hat{\boldsymbol{x}} = \boldsymbol{\mathcal{M}}f(\boldsymbol{x}) = \boldsymbol{\mathcal{M}}\boldsymbol{\phi}$. The quantisation loss a.k.a quantisation error is then defined as:

$$\xi^2 = \|\boldsymbol{x} - \hat{\boldsymbol{x}}\|_2^2. \quad (49)$$

However, ξ^2 quantifies only its magnitude.

Residual Descriptor (RD) is therefore defined as:

$$\boldsymbol{\xi} = \boldsymbol{x} - \boldsymbol{\mathcal{M}}\boldsymbol{\phi} \quad (50)$$

and illustrated in figure 4. Descriptors $\boldsymbol{x} \in [-3, 3]^2$ are coded with three atoms $\boldsymbol{m}_1 = [0, 3]^T$, $\boldsymbol{m}_2 = [-2, -2]^T$, and $\boldsymbol{m}_3 = [2, -2]^T$ by SC and LcSA. The mid-level features $\boldsymbol{\phi}$ are projected back to the descriptor space: $\hat{\boldsymbol{x}} = \boldsymbol{\mathcal{M}}\boldsymbol{\phi}$. The resulting quantisation loss, *i.e.* RD, are visualised by displacements between each descriptor $\boldsymbol{x}$ and its approximation $\hat{\boldsymbol{x}}$. Figure 4(a) shows low quantisation loss for SC with regularisation $\alpha = 1$ and figure 4(b) shows large quantisation errors for LcSA due to low $l=2$.

To better represent the original $\boldsymbol{x}$, both the mid-level feature $\boldsymbol{\phi}$ and the Residual Descriptor $\boldsymbol{\xi}$ are incorporated into the signature as two different descriptors:

$$\boldsymbol{\phi}_n^{(1)} = f\left(\boldsymbol{x}_n, \boldsymbol{\mathcal{M}}\right), \quad \boldsymbol{\phi}_n^{(2)} = \boldsymbol{x}_n - \boldsymbol{\mathcal{M}}\boldsymbol{\phi}_n^{(1)} \quad (51)$$

With this formulation, the cross-term of the Bi-modal Second-order Occurrence Pooling can capture co-occurrences between mid-level feature $\boldsymbol{\phi}^{(1)}$ encoding original descriptor $\boldsymbol{x}$ and the corresponding residual error $\boldsymbol{\phi}_n^{(2)}$. Thus, it associates the error with the descriptor and improves the coding accuracy.

4 Pooling Low-level Descriptors

Recently, a coder-free approach was proposed in [31] in the context of semantic segmentation. This method avoids mid-level coding and employs the autocorrelation matrix formed by Average pooling of the outer products of local image descriptors. The matrix is then normalised with the log operator. We go beyond the second-order and generalise this approach to Higher-order Occurrence Pooling as well as propose a two stage normalisation based on eigenvalue decomposition and Power Normalisation.

Eigenvalue Power Normalisation. Corrections such as Power Normalisation (cf. section 1.3) are known to improve the Average pooling [19], [21], [22]. This is related to the problem of burstiness which was defined in [22] as "*the property that a given visual element appears more times in an image than a statistically independent model would predict*". The Analytical pooling operators [11], [26] have been advocated as a remedy to the burstiness phenomenon. They act similarly to the MaxExp operator (cf. section 1.3) which approximates the probability of at least one particular visual word being present in an image. They are applied to each coefficient in mid-level features, which are assumed to be i.i.d., therefore they can also be interpreted as whitening of the i.i.d. coefficients. We argue that the i.i.d. assumption does not always hold in real images. For instance, local descriptors extracted from repetitive texture patterns

frequently co-occur and their coefficients are correlated. Thus, the burstiness of these descriptors can be addressed effectively by decorrelating along the principal components of the signal rather than coefficient-wise normalisations.

We thus propose to perform Power Normalisation, MaxExp, or a similar correction on the eigenvalues of the higher-order tensor coefficients. Figures 5 (a) and (b) show the difference between the eigenvalue (ePN) and coefficient-wise (PN) Power Normalisation. Autocorrelation matrix $\boldsymbol{H}$ was built from correlated $2D$ features $\boldsymbol{\phi}$. The principal components of $\boldsymbol{H}^{0.4}$ and $\boldsymbol{H}^{0.1}$ show the data being whitened with ePN to a significant extent. On the contrary, element-wise Power Normalisation (*PN*) fails to whiten the correlated data.

Higher-order Pooling with eigenvalue Power Normalisation (ePN). Second-order Occurrence Pooling can be performed by applying Power Normalisation to the eigenvalues of the autocorrelation matrix. For the second-order matrix, we use Singular Value Decomposition. For the Third-order Occurrence Pooling, we employ Higher Order Singular Value Decomposition (HOSVD) [27], [28]. Power Normalisation is then performed on the eigenvalues from so-called core tensor and the autocorrelation tensor is reassembled:

$$\boldsymbol{\phi}_n = \boldsymbol{x}_n,\ \forall n \in \mathcal{N} \tag{52}$$

$$\boldsymbol{H} = \underset{n\in\mathcal{N}}{\operatorname{avg}}(\boldsymbol{\Phi}_n)\,,\ \boldsymbol{\Phi}_n = \otimes_r \boldsymbol{\phi}_n \tag{53}$$

$$(\boldsymbol{E}; \boldsymbol{A}_1, ..., \boldsymbol{A}_r) = \mathrm{HOSVD}(\boldsymbol{H}) \tag{54}$$

$$\hat{\boldsymbol{E}} = \mathrm{sgn}(\boldsymbol{E})\,|\boldsymbol{E}|^{\gamma_e} \tag{55}$$

$$\hat{\boldsymbol{H}} = \hat{\boldsymbol{E}} \times_1 \boldsymbol{A}_1 \cdots \times_r \boldsymbol{A}_r \tag{56}$$

$$\hat{\boldsymbol{h}} = \mathrm{sgn}(\boldsymbol{h}^*)\,|\boldsymbol{h}^*|^{\gamma}\,,\ \boldsymbol{h}^* = u_:(\hat{\boldsymbol{H}}) \tag{57}$$

Coder-free image signatures are represented in equation (52), however, to reduce their size, we apply PCA $\boldsymbol{\phi}_n = \mathrm{pcaproj}\,(\boldsymbol{x}_n)$ and obtain $\boldsymbol{\phi}_n \in \mathbb{R}^K$. We investigate three variants of these features based on the use of spatial information: i) no spatial information, ii) appended on the descriptor level, *i.e.* Spatial Coordinate Coding, iii) appended according to equation (48) where $\boldsymbol{\phi}_n^{(1)} = \mathrm{pcaproj}\,(\boldsymbol{x}_n)$ and $\boldsymbol{\phi}_n^{(2)}$ is a binary vector, type of SPM [30], obtained by assigning 1 for each spatial window containing descriptor $\boldsymbol{x}_n$, 0 otherwise.

Average pooling is performed in equation (53) as discussed in section 2.1. In detail, the higher-order autocorrelation tensor $\boldsymbol{H} \in \mathbb{R}^{K^r}$ (an r^{th}-order equivalent of the autocorrelation matrix) is computed by averaging over tensors $\boldsymbol{\Phi}_n \in \mathbb{R}^{K^r}$.

Equations (54-56) and (57) represent two stage pooling with eigenvalue- and coefficient-wise corrections such as Power Normalisation, respectively. In Equation (54), HOSVD denoted by operator $\mathrm{HOSVD} : \mathbb{R}^{K^r} \rightarrow \left(\mathbb{R}^{K^r}; \mathbb{R}^{K\times K}, ..., \mathbb{R}^{K\times K}\right)$ decomposes the higher-order autocorrelation tensor $\boldsymbol{H}$ into core tensor $\boldsymbol{E}$ of eigenvalues and orthonormal factor matrices $\boldsymbol{A}_1, ..., \boldsymbol{A}_r \in \mathbb{R}^{K\times K}$, which can be interpreted as the principal components in r modes. Element-wise corrections are then applied to eigenvalues $\boldsymbol{E}$ by Power Normalisation (cf. equation (55)). The higher-order autocorrelation tensor $\hat{\boldsymbol{H}} \in \mathbb{R}^{K^r}$ is reassembled in equation (56) by r-mode product $\times_r$ (detailed in [28]) of normalised tensor $\hat{\boldsymbol{E}}$ and $\boldsymbol{A}_1$, ..., $\boldsymbol{A}_r$. Operator $u_:$ is used in equation (57) to remove the redundant coefficients from symmetric tensor $\hat{\boldsymbol{H}}$ and coefficient-wise correction is applied to $\boldsymbol{h}^*$. Coefficient-wise correction, *i.e.* MaxExp from equation (13), can be also applied here. Note that the ℓ_2 normalisation (cf. equation (3)) is always applied at the end.

5 Experimental Results

We first introduce our experimental settings in section 5.1. First-, Second-, and Third-order Occurrence Pooling are compared to FV and VLAT in section 5.2. Various descriptor fusion techniques with Higher-order Occurrence Pooling are evaluated in section 5.3. Higher-order Occurrence Pooling variants for the low-level descriptors are compared in section 5.4 and other pooling techniques are evaluated in section 5.5.

5.1 Experimental Settings

Eight widely used image recognition benchmarks were used in our experiments. The datasets, descriptor parameters, various experimental details and state-of-the-art results are summarised in table 1. Other settings are discussed below.

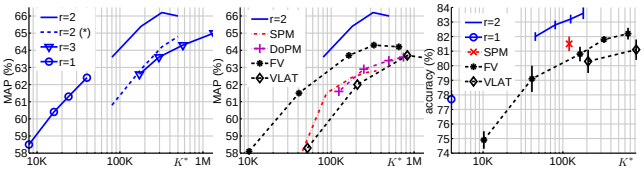
Descriptors. Opponent SIFT was extracted on dense grids. Either grey scale only (128D) or grey and colour components (128D+144D) were used as detailed in table 1. PCA was applied to reduce descriptor dimensionality to 80D for the grey and 120D opponent components in FV and VLAT.

Spatial bias. Spatial relations in images were exploited mainly by Spatial Coordinate Coding [43] described in section 3.1. Spatial Pyramid Matching (**SPM**) [30] and Dominant Angle Pyramid Matching (**DoPM**) [43] were additionally used for comparison to the standard BoW with first-order occurrences. Multiple spatial grids such as 1x1, 1x3, 3x1 or 1x1, 2x2, 3x3, and 4x4 were used in SPM [30]. DoPM [43] was used to exploit dominant gradient orientations with 5 quantisation levels of 1, 3, 6, 9, and 12 angular grids. BoW with first-order occurrences employed either SCC or SPM with/without DoPM as detailed later. As recommended in [29], we use FV and VLAT with SCC rather than SPM.

Dictionaries. Online Dictionary Learning [58] was used to train dictionaries for Sparse Coding. Dictionary learning proposed in [15] was shown to outperform k-means, we therefore used it for Locality-constrained Linear Coding and adapted it to work with Approximate Locality-constrained Soft Assignment. Dictionary size was varied from 4K to 40K for First-, 300 to 1600 for Second-, and 100 to 200 for Third-order Occurrence Pooling. Fisher Vector Encoding [18] and Vector of Locally Aggregated Tensors [20] were used in comparisons, GMM and k-means dictionaries with 64 to 4096 and 64 to 512 atoms were employed, respectively.

Coding and Pooling. Unless stated otherwise, Sparse Coding SC and our $@n$ pooling operator were used in the experiments. Additional results for Max-pooling, MaxExp, Power Normalisation and the proposed eigenvalue based normalisation are also provided. FV and VLAT were combined with Power Normalisation only as other operators are not directly applicable.

Classifiers. Kernel Discriminant Analysis [59] with linear kernels $Ker_{ij} = (\boldsymbol{h}_i)^T \cdot \boldsymbol{h}_j$ was applied in all experiments



(*a*) VOC07 (*b*) VOC07 (*c*) Caltech101

Fig. 6. Performance of BoW with Higher-order Occurrence Pooling reported for several signature lengths K^*. (*a*) Occurrence Pooling for order $r=1,2,3$ with Spatial Coordinate Coding (SCC); (*) denotes $r=2$ without SCC. (*b*, *c*) BoW with order $r=2$ compared to SPM ($r=1$), DoPM ($r=1$), FV as well as VLAT.

(unless stated otherwise), where $\boldsymbol{h}_i, \boldsymbol{h}_j \in \mathbb{R}^K$ are signatures for images i and j. Mean Accuracy (`acc.`) and Mean Average Precision (`MAP`) are reported by us (see table 1).

5.2 Bag-of-Words with First-, Second-, and Third-order Occurrence Pooling

This section compares the performance of the proposed Higher-order Occurrence Polling to the state-of-the art approaches, *e.g.* FV and VLAT, on PascalVOC07 and Caltech101 benchmarks. The results are reported for BoW (cf. section 2) with Sparse Coding of grey scale SIFT and occurrence orders $r=1$, 2, and 3. Note that the BoW model with $r=1$ is equivalent to the standard BoW (cf. section 1.1).

Figure 6(a) compares performance for various orders r. BoW with Second-order Occurrence Pooling $r=2$ outperforms orders $r=1$ and $r=3$, and achieves 65.4%, 66.2%, and 66.0% MAP for signature lengths $K^*=180300$, 320400, and 500500, respectively. These K^* correspond to dictionary sizes $K=600$, 800, and 1000. BoW with $r=1$ scores 3.8% less, that is 62.4% for $K=K^*=40000$. Note that for larger visual dictionaries the coding step is computationally prohibitive, *i.e.* it takes 815s to code 1000 descriptors for $K=40000$ on a single core of 2.3GHz AMD Opteron (1.5s for $K=800$). BoW of order $r=3$ yields 65% MAP ($K=200$ and $K^*=1353400$). The top score of 66.2% is attained by Second-order Occurrence Pooling with Spatial Coordinate Coding (SCC) [43]. Ignoring spatial information (*i.e.* no SCC) decreases MAP by 1.4%.

Figure 6(b) compares BoW ($r=2$ with SCC) to FV, VLAT, and to BoW ($r=1$) based on SPM (spatial) [30] and DoPM (dominant angle) [43] pyramids. With 66.2%, Second-order Occurrence Pooling outperforms FV by 1.9%. BoW ($r=1$) with SPM or DoPM as well as VLAT attain lower scores of 62.8%, 63.6%, and 63.7%, respectively. The reported results are the top scores w.r.t. varying signature size.

The classification performance on Caltech101 is presented in figure 6(c). The settings are identical to those in figure 6(b). Second-order Occurrence Pooling scores $83.6 \pm 0.4\%$ accuracy for signature length $K^*=180300$ ($K=600$ atoms). This is a 2.8% improvement over FV ($80.8 \pm 0.5\%$) for the comparable signature length $K^*=163840$. BoW ($r=1$) with SPM ($K^*=120000$) yields $81.5 \pm 0.4\%$ accuracy. FV and VLAT obtain their top scores of $82.2 \pm 0.4\%$ and $81.1 \pm 0.7\%$ for large signature sizes. Reducing the number of training images per class from 30 to 15 lowers the scores by around 8%. Otherwise, all trends remain consistent.

The main observations from figure 6 are that the proposed model with second-order occurrences yields the highest performance and provides an attractive trade-off between the tractability of coding and increasing signature lengths. Also, Spatial Coordinate Coding [43] attains better results than the model without spatial information for the same K^*.

5.3 Descriptor fusion with Second-order Occurrence Pooling

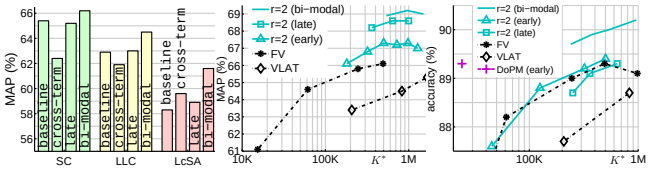
This section evaluates our novel approach to descriptor fusion proposed in section 3 and illustrated in figure 3. The fusions are demonstrated for three types of descriptors, namely the grey SIFT, colour components of SIFT, and the Residual Descriptor (RD) proposed in section 3.3. The following fusion schemes are presented: a) early and b) late fusions explained in sections 3.1, c) Bi-modal Second-order Occurrence Pooling ($r=2$) outlined in section 3.2, c) Multi-modal Second-order Occurrence Pooling, *e.g.* fusion of all three types of descriptors. We also report results for FV and VLAT, both using the early fusion. Comparison of fusion schemes with different coders to baseline approach are presented in figure 7(a).

Dataset		No. of classes	Training samples	Test samples	Eval. measure	State-of-the-art non-CNN	ours	CNN
PascalVOC07	[45]	20	5011	4952	MAP	[29] 66.3	69.2	[46] 82.4
Caltech101	[47]	102	15/30 (per class)	7614/6084	acc.	[37] 84.3	83.9	[46] 93.4
Flower102	[36]	102	2040	6149	acc.	[48] 80.3	90.2	[49] 91.3
ImageCLEF11	[39]	99	8K (x2 flip)	10K	MAP	[42] 38.8	41.2	-
15Scenes	[30]	15	100 (per class)	2985	acc.	[50] 89.8	90.1	[51] 91.6
PascalVOC10AR	[45]	9	608 (x2 flip)	613	MAP	[52] 65.1	66.5	[53] 70.1
MITIndoors67	[54]	67	5360 (x2 flip)	1340	acc.	[55] 64.6	68.9	[51] 70.8
SUN397	[56]	397	19850	19850	acc.	[57] 47.2	49.0	[51] 53.8

	Descr. type	Descr. per img.	Location samp. (px)	Radii (px)	Dict. size	Coding	Spatial inform.	Order
PascalVOC07	Opp. SIFT	19420	4:2:16	12, 16:8:56	100-1600	{SC/LLC/ LcSA/raw	{none/SCC/ SPM*/DoPM*	1*,2,3
Caltech101	SIFT	5200	4:2:10	16:8:40	300-800	SC/raw	SCC/SPM*	1*,2
Flower102	Opp. SIFT	14688	6:3:15	16:8:40	300-1600	SC	SCC/DoPM*	1*,2
ImageCLEF11	Opp. SIFT	19642	4:2:16	12, 16:8:56	800	SC	SCC	2,3
15Scenes	SIFT	12650	3, 4:2:16	10,12, 16:8:56	400-800	SC/raw	{none/SCC/ SPM	2,3
PascalVOC10AR	Opp. SIFT	5660	4:2:16	12, 16:8:56	400-800	SC/raw		2,3
MITIndoors67	Opp. SIFT	65284	3, 4:2:16	10,12, 16:8:56	800-1000	SC/raw	SCC	2,3
SUN397	Opp. SIFT	49986	4:2:16	12, 16:8:56	800	SC	SCC	2

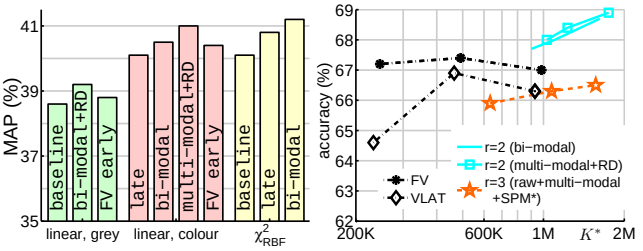
(*) the first-order BoW with SPM/DoPM is used for comparisons - it was proposed in [11], [43]

TABLE 1
Summary of the datasets with corresponding state-of-the-art results, as well as experimental settings for the results presented in this section.



(*a*) VOC07, RD (*b*) VOC07, fusion (*c*) Flower102, fusion

Fig. 7. Descriptor fusion with Second-order Occurrence Pooling. (*a*) `baseline` show results for SC, LLC, and LcSA coders ($r=2$, $K=600$) without fusion. Residual Descriptors from section 3.3 were combined by cross-term (cf. section 2.1), `late` fusion (cf. section 3.1), and `bi-modal` fusion with Second-order Occurrence Pooling. FV, VLAT, and DoPM ($r=1$) use early fusion of grey and colour SIFT.



(*a*) ImageCLEF11 (*b*) MITIndoors67

Fig. 8. Evaluation of proposed fusion schemes based on Second-order Occurrence Pooling on (*a*) ImageCLEF11 and (*b*) MITIndoors67. Plots include `late`, `bi-modal`, and `multi-modal` fusions of grey and colour SIFT components as well as Residual Descriptors for the linear and χ^2_{RBF} kernels. FV and VLAT use `early` fusion for grey and colour components.

Baseline results for Second-order Occurrence Pooling with SC, LLC, and LcSA ($K=600$, $K^*=180300$) are presented without any fusion. The best MAP scores for baseline are obtained with SC (65.4%) followed by LLC (62.9%) and LcSA (58.3%). This is due to the different quantisation loss (cf. equation (49)) of the coders. We measured ξ^2 according to equation (49) over a large number of descriptors, averaged the error scores, and observed the same ranking $\xi^2_{SC} < \xi^2_{LLC} < \xi^2_{LcSA}$. We also note that the gap in performance between SC and LcSA is 7.1% for $r=2$. The gap is smaller for the standard BoW ($r=1$) with SPM. This shows that the quantisation noise is amplified by the higher order occurrences of visual words. The undesired effects of the quantisation loss are addressed by our fusion with Residual Descriptor.

Residual Descriptor (RD) is combined with SC, LLC, and LcSA by bi-modal and late fusions. Compared to the baseline, the late fusion with the RD in figure 7(a) does not make a significant difference for any of the coders. This is expected as the late fusion does not associate the residual codes with the corresponding descriptors or with the mid-level features (cf. section 3.3). The cross-term is also insufficient without the self-tensors in equations (39) and (41). However, capturing the co-occurrences of RD with the corresponding features by using bi-modal fusion results in a significant gain for all coders, *i.e.* 0.8%, 1.6%, and 3.3% MAP for SC, LLC, and LcSA, respectively. The benefits from using RD are larger for the coders with high quantisation loss. Note that SC attains 66.2% MAP with the overall signature length $K^*=265356$, which is reduced from $K^*=320400$ in section 5.2 (both variants yield the same score).

Grey and colour components of SIFT are also fused by the proposed schemes. In figure 7(b), the proposed bi-modal fusion scores 69.2% MAP ($K=800$), which improves upon grey SIFT by 3%. Note that separate dictionaries are used for the grey and colour descriptors resulting in signatures of length $K^*=960400$. The best scores for late and early fusions are 68.6% and 67.3%, respectively, both below the bi-modal fusion. Finally, FV and VLAT with the early fusion perform worse than the proposed methods and score 65.6% and 64.8%. The results of a similar experiment performed on the Flower102 set are presented in figure 7(c). The observations from PascalVOC07 in figure 7(b) hold in this experiment except for the late fusion scoring worse than the early fusion. The results for both datasets show that the proposed bi-modal fusion offers a good trade-off between the performance and the length of signatures.

Multi descriptor fusion. Figure 8 presents the comparison of several fusion schemes with Second-order Occurrence Pooling of grey, colour, and Residual Descriptors on ImageCLEF11 which includes many abstract topics, *e.g. party life*. We therefore compare the classification performance of linear and χ^2_{RBF} kernels. Evaluations w.r.t. K^* are studied on MITIndoors67.

In addition to the fusion approaches evaluated above, *i.e.* early, late, bi-modal, we also introduce multi-modal fusion that combines all grey, colour, and Residual Descriptors. Results for fusion with Fisher Vectors were obtained by early fusion as in figure 7. Results for linear kernels confirm the observations from figure 7, *i.e.* the bi-modal fusion with RD improves upon baseline and FV. Adding colour further improves the results in particular with the multi-modal combination of grey, colour, and Residual Descriptors. Signatures of comparable size were used for all methods. The results also show a small gain when χ^2_{RBF} is chosen over linear kernels.

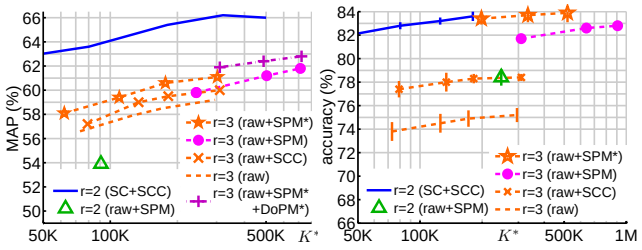
Evaluations on MITIndoors67 show a marginal difference for bi-modal fusion with or without Residual Descriptor. We obtain 68.9% accuracy which is close to 70.8% of CNN approach [51].

Evaluations on the SUN397 dataset resulted in 44.5% and 49.0% accuracy obtained for our baseline and bi-modal Second-order Occurrence Pooling, respectively, which improves upon FV with 43.0% and 47.2% [57]. These results are below 53.5% from [51], which was obtained with a more complex CNN architecture trained on the Places and ImageNet datasets with use of multiple dataset augmentations.

5.4 Low-level Descriptor Pooling

We present the results for Second- and Third-order Occurrence Pooling ($r=2$ and $r=3$) on the low-level grey SIFT descriptors (`raw`), that is coder-free techniques from section 4. The `raw` methods are compared to our Second-order Occurrence Pooling on the mid-level features. Our third-order method uses the two stage pooling, *i.e.* eigenvalue- and coefficient-wise Power Normalisation. The Second-order Pooling on SIFT corresponds to the approach from [31], which uses the matrix logarithm and coefficient-wise Power Normalisation $0<\gamma\leq 1$. For best performance, SIFT (`raw`) was combined with SCC or SPM and DoPM.

Figure 9 shows the results for PascalVOC07 and Caltech101. Our Second-order Occurrence Pooling ($r=2$) with



(*a*) VOC07 (*b*) Caltech101

Fig. 9. Second- and Third-order Occurrence Pooling for fusion of low-level descriptors. Grey SIFT (`raw`) with no spatial cues was fused with SCC (`raw+SCC`), SPM (`raw+SPM`) [31], `bi-modal` fused SPM (`raw+SPM*`) and `multi-modal` fused SPM and DoPM (`raw+SPM*+DoPM*`). Our Second-order Occurrence Pooling ($r = 2$) of mid-level features uses Sparse Coding and Spatial Coordinate Coding (`SC+SCC`).

the SC coder (`SC+SCC`) outperforms coder-free methods (`raw+...`). For instance, coder-free Third-order Occurrence Pooling ($r = 3$) results in a somewhat lower performance except for Caltech101 which suggests that datasets with little clutter and well aligned objects of fixed scale can be reliably classified without the coding step and mid-level features. However, Second-order Pooling ($r = 2$) with SPM (`raw+SPM`) from [31] yields 54.0% MAP for PascalVOC07 and 78.5% for Caltech101. These scores are nearly 10% and 6% lower than results on our coder-free approaches ($r = 3$). Another interesting observation is the improvement introduced by our bi-modal fusion. For comparable signature lengths, descriptor fusion with either SPM only or SPM and DoPM, *e.g.* (`raw+SPM*`) or (`raw+SPM*+DoPM*`), outperformed regular SPM showing the benefit of our tensor level fusion. Finally, the results for FV and VLAT confirm the observations from figures 6 and 7.

Additional evaluation on MITIndoors67 is presented in figure 8(b) where our multi-modal fusion of grey SIFT, colour, and SPM components (`raw+multi-modal+SPM*`) attained 66.5% accuracy. Moreover, our Second-order Occurrence Pooling (SC with SCC) produced state-of-the-art results of 65.0% on PascalVOC10AR (not in plots). This was further improved to 66.5% by fusion with colour. Third-order Occurrence Pooling of low-level descriptors fused with colour and SPM components yields 66.0% MAP.

5.5 Pooling Operators

In this section we compare our pooling operators to other state-of-the-art methods on PascalVOC07.

In figure 10, we first compare the impact of the proposed two stage eigenvalue Power Normalisation (`ePN+Gamma`) which achieves 60.0% MAP and outperforms the coefficient-wise Power Normalisation (`Gamma`) scoring only 57.5% MAP. This experiment was performed on image signatures from the low-level descriptors with coder-free Third-order Occurrence Pooling ($r = 3$) as discussed in section 4.

Similar comparison is carried out for Second-order Occurrence Pooling of mid-level features obtained with Sparse Coding. The best result of 65.4% MAP is obtained by our $@n$ operator, that is 4% higher than the Max-pooling. The proposed two stage pooling based on eigenvalue- and coefficient-wise pooling improves upon single stage techniques (Max, Gamma, MaxExp). In particular, eigenvalue

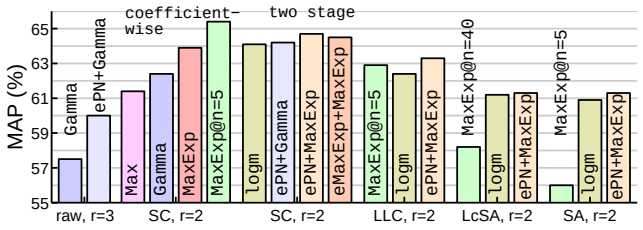
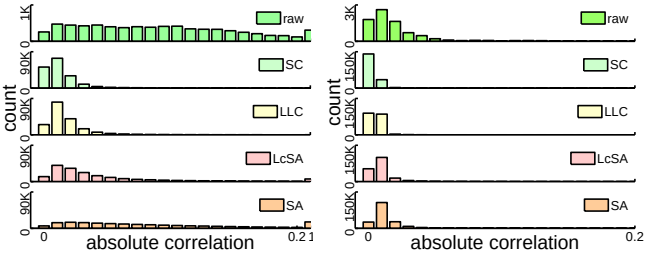


Fig. 10. Comparison of pooling operators on PascalVOC07. Second-order Occurrence Pooling ($r = 2$, $K = 600$), SCC, grey SIFT, and linear kernels were applied except for the coder-free case ($r = 3$). For the SC coder, `Max`-pooling, `Gamma` Power Normalisation, `MaxExp`, and our $@n$ (`MaxExp@n=5`) were combined with Second-order Occurrence Pooling. Variants of two stage pooling with the first stage eigenvalue Power Normalisation or MaxExp (`ePN` or `eMaxExp`) followed by second stage coefficient-wise Power Normalisation or MaxExp are compared to the matrix logarithm (`logm`). Results on LLC, LcSA, and SA (Soft Assignment [7], [11]) employ a subset of pooling operators to show their varied ability to decorrelate the features.

Power Normalisation (`ePN+Gamma`) improves by 1.8% upon (`Gamma`).

Figure 10 reveals a vital difference between the coders used in the experiments. We note that Sparse Coding with the $@n$ operator outperforms the geodesic distance, *i.e.* matrix logarithm (`logm`) used in [31] as well as the eignevalue Power Normalisation combined with the MaxExp operator (`ePN+MaxExp`). The higher scores of SC suggest that it generates low-correlated mid-level features, which benefit from coefficient-wise pooling such as $@n$. However, in contrast to SC and LLC, LcSA with coefficient-wise pooling (`MaxExp@n=40`) scores 3% less than `logm`. This significant difference can be attributed to the correlation between dimensions of mid-level features obtained with LcSA. It is demonstrated in [60] that LcSA does not take into account correlation between visual words selected for low-level descriptors whilst LLC does. Soft Assignment [7], which is closely related to LcSA [10], exhibits even larger correlations.

The above observations are supported by the simulations in figure 11. SIFT descriptors from 100 randomly selected images (PascalVOC07) were coded with the SC, LLC, LcSA, and SA coders. The same coding parameters and dictionaries were used as in experiments from figure 10. Histograms of the absolute values of Pearson's correlation coefficients were computed. Figure 11(a) shows that for SC and LLC most of the coefficients of mid-level features are weakly correlated. In contrast, low-level descriptors (`raw`), LcSA,



(*a*) various coders (*b*) various coders+ePN

Fig. 11. Histogram of the absolute values of Pearson's correlation coefficients for raw (no coder) and SC, LLC, LcSA, and SA coders (*a*) without pooling and (*b*) with eigenvalue pooling (ePN). Compact histograms show that $K = 600$ dimensions of mid-level features are decorrelated better by SC and LLC than LcSA or SA. Also, ePN further decorrelates the data.

Grey SIFT	VOC 07	CLEF 11	Calt. 101 (15 img.)	Calt. 101 (30 img.)	VOC 10AR	MIT In67	SUN 397	15 Scenes
SC r=2	Uni-modal							Bi-modal RD
	66.2	**40.1**	76.6±.5	83.6±.4	**65.0**	**64.2**	**44.5**	**90.1±.6**
raw r=3	Multi-modal SPM*+DoPM*		Bi-modally fused SPM*					
	62.7	38.1	**77.2±.2**	**83.9±.8**	63.5	-	-	88.4±.6
FV	64.3	38.8	75.7±.5	82.2±.4	-	-	-	-
VLAT	63.7	-	74.2±.6	81.1±.7	-	-	-	-
SCC, r=1	62.4	-	72.0±.3	77.7±.7	-	-	-	-
SPM, r=1	62.8	-	74.9±.4	81.5±.5	-	-	-	-
DoPM, r=1	63.6	-	-	-	-	-	-	-

Grey + Opp. SIFT		VOC 07	CLEF 11	Flower 102	VOC 10AR	MIT In67	SUN 397
Bi-modal	SC, $r=2$	**69.2**	**41.2**	**90.2**	-	68.7	**49.0**
Early	SC, $r=2$	67.3	-	89.4	-	-	-
Late	SC, $r=2$	68.6	40.8	89.3	**66.5**	-	-
Multi-m.	SC, $r=2$, RD	-	-	-	-	**68.9**	-
Late	raw, $r=3$	-	-	-	66.0	-	-
Multi-m.	raw, $r=3$, SPM*	-	-	-	-	**66.5**	-
FV		65.6	40.4	89.3	-	67.4	-
VLAT		64.8	-	88.7	-	66.9	-
DoPM	$r=1$	-	-	89.3	-	-	-

TABLE 2
Summary of the best results from this study. See figures 6-9 for detailed comparisons.

and SA produce more correlated coefficients. Figure 11(b) shows that the ePN operator reduces coefficient correlation for all methods.

6 Conclusions

This paper proposes a theoretically derived framework that extends Bag-of-Words with higher-order occurrences computed on mid-level features. According to our results, Second-order Occurrence Pooling offers the best trade-off between the complexity of coding, the length of signatures, and the classification performance. It outperforms the first-order variants of BoW, Fisher Vector Encoding, and Vector of Locally Aggregated Tensors. Sparse Coding and the $@n$ pooling operator are highlighted as the best performers. Evaluations were conducted in a common testbed on several standard benchmarks. The best results obtained by our methods are listed in table 2 as well as compared to state-of-the-art results in table 1.

A coder-free Third-order Occurrence approach with a novel two stage pooling is also proposed. It challenges coder-based BoW on simple datasets. We emphasise that the coder-based BoW performs better if its quantisation loss is low as shown in comparisons to Higher-order Occurrence Pooling with Sparse Coding. The importance of feature decorrelation in image representations is also demonstrated.

To benefit from multiple types of descriptors, bi- and multi-modal fusions are formulated based on cross-modal occurrences. Their contribution is demonstrated with fusion of the grey and colour features, and our Residual Descriptor.

Finally, we believe that our ideas of exploiting higher-order occurrences are also applicable to CNN features. The research in this area has been reported in [61], [62], [63].

Acknowledgments

This work has been supported by EU Chist-Era EPSRC EP/K01904X/1 Visual Sense project and the BBC. The authors would also like to thank Prof. Adrian Hilton (CVSSP) and Dr. Julien Mairal (INRIA LEAR) for their support in preparing this manuscript.

References

[1] J. Sivic and A. Zisserman, "Video Google: A Text Retrieval Approach to Object Matching in Videos," *ICCV*, vol. 2, pp. 1470–1477, 2003.
[2] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual Categorization with Bags of Keypoints," *ECCV Workshop on Statistical Learning in Computer Vision*, pp. 1–22, 2004.
[3] D. G. Lowe, "Object Recognition from Local Scale-Invariant Features," *CVPR*, vol. 2, pp. 1150–1157, 1999.
[4] K. Mikolajczyk and C. Schmid, "A Performance Evaluation of Local Descriptors," *PAMI*, vol. 27, no. 10, pp. 1615–1630, 2005.
[5] K. E. A. van de Sande, T. Gevers, and C. G. M. Snoek, "A Comparison of Color Features for Visual Concept Classification," *CIVR*, pp. 141–149, July 2008.
[6] J. C. van Gemert, J. M. Geusebroek, C. J. Veenman, and A. W. M. Smeulders, "Kernel Codebooks for Scene Categorization," *ECCV*, vol. 5304, pp. 696–709, 2008.
[7] J. C. van Gemert, C. J. Veenman, A. W. M. Smeulders, and J. M. Geusebroek, "Visual Word Ambiguity," *PAMI*, 2010.
[8] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Lost in Quantization: Improving Particular Object Retrieval in Large Scale Image Databases." *CVPR*, 2008.
[9] P. Koniusz and K. Mikolajczyk, "Soft Assignment of Visual Words as Linear Coordinate Coding and Optimisation of its Reconstruction Error," *ICIP*, 2011.
[10] L. Lingqiao, L. Wang, and X. Liu, "In Defence of Soft-assignment Coding," *ICCV*, 2011.
[11] P. Koniusz, F. Yan, and K. Mikolajczyk, "Comparison of Mid-Level Feature Coding Approaches And Pooling Strategies in Visual Concept Detection," *CVIU*, 2012.
[12] H. Lee, A. Battle, R. Raina, and A. Y. Ng, "Efficient Sparse Coding Algorithms," *NIPS*, pp. 801–808, 2007.
[13] J. Yang, K. Yu, Y. Gong, and T. S. Huang, "Linear Spatial Pyramid Matching using Sparse Coding for Image Classification," *CVPR*, pp. 1794–1801, 2009.
[14] K. Yu, T. Zhang, and Y. Gong, "Nonlinear Learning using Local Coordinate Coding," *NIPS*, 2009.
[15] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong, "Locality-constrained Linear Coding for Image Classification," *CVPR*, 2010.
[16] X. Zhou, K. Yu, T. Zhang, and T. S. Huang, "Image Classification using Super-Vector Coding of Local Image Descriptors," *ECCV*, pp. 141–154, 2010.
[17] H. Jegou, M. Douze, C. Schmid, and P. Pérez, "Aggregating Local Descriptors into a Compact Image Representation," *CVPR*, pp. 3304–3311, 2010.
[18] F. Perronnin and C. Dance, "Fisher Kernels on Visual Vocabularies for Image Categorization," *CVPR*, vol. 0, pp. 1–8, 2007.
[19] F. Perronnin, J. Sánchez, and T. Mensink, "Improving the Fisher Kernel for Large-Scale Image Classification," *ECCV*, pp. 143–156, 2010.
[20] R. Negrel, D. Picard, and P.-H. Gosselin, "Compact Tensor Based Image Representation for Similarity Search," *ICIP*, 2012.
[21] S. Boughorbel, J.-P. Tarel, and N. Boujemaa, "Generalized Histogram Intersection Kernel for Image Recognition," *ICIP*, pp. 161–164, 2005.
[22] H. Jégou, M. Douze, and C. Schmid, "On the Burstiness of Visual Elements," *CVPR*, pp. 1169–1176, 2009.
[23] J. Yang, Y.-G. Jiang, A. G. Hauptmann, and C.-W. Ngo, "Evaluating Bag-of-Visual-Words Representations in Scene Classification," *MIR*, pp. 197–206, 2007.
[24] K. Chatfield, V. Lempitsky, A. Vedaldi, and A. Zisserman, "The Devil is in the Details: An Evaluation of Recent Feature Encoding Methods," *BMVC*, 2011.
[25] Y. Boureau, F. Bach, Y. LeCun, and J. Ponce, "Learning Mid-Level Features for Recognition," *CVPR*, 2010.
[26] Y. Boureau, J. Ponce, and Y. LeCun, "A Theoretical Analysis of Feature Pooling in Vision Algorithms," *ICML*, 2010.
[27] L. D. Lathauwer, B. D. Moor, and J. Vandewalle, "A multilinear singular value decomposition," *SIAM J. Matrix Analysis and Applications*, vol. 21, pp. 1253–1278, 2000.
[28] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM Review*, vol. 51, no. 3, pp. 455–500, 2009.
[29] J. Sánchez, F. Perronnin, and T. E. de Campos, "Modeling the Spatial Layout of Images Beyond Spatial Pyramids." *PRL*, 2012.
[30] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories," *CVPR*, vol. 2, pp. 2169–2178, 2006.

[31] J. Carreira, R. Caseiro, J. Batista, and C. Sminchisescu, "Semantic Segmentation with Second-Order Pooling." *ECCV*, 2012.

[32] X. YU and Y.-J. ZHANG, "A 2-D Histogram Representation of Images for Pooling." *SPIE*, 2011.

[33] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," *NIPS*, pp. 1097–1105, 2012.

[34] J. Bilmes, "A Gentle Tutorial of the EM Algorithm and its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models," ICSI, Tech. Rep. TR-97-021, 1997.

[35] P. Koniusz and K. Mikolajczyk, "On a Quest for Image Descriptors Based on Unsupervised Segmentation Maps," *ICPR*, vol. 0, pp. 762–765, 2010.

[36] M.-E. Nilsback and A. Zisserman, "Automated Flower Classification over a Large Number of Classes," *ICVGIP*, Dec 2008.

[37] J. Yang, Y. Tian, L.-Y. Duan, T. Huang, and W. Gao, "Group-Sensitive Multiple Kernel Learning for Object Recognition," *TIP*, vol. 21, no. 5, pp. 2838–2852, 2012.

[38] F. Yan, K. Mikolajczyk, M. Barnard, H. Cai, and J. Kittler, "Lp Norm Multiple Kernel Fisher Discriminant Analysis for Object and Image Categorisation," *CVPR*, 2010.

[39] S. Nowak, K. Nagel, and J. Liebetra, "The CLEF 2011 Photo Annotation and Concept-based Retrieval Tasks," *CLEF*, 2011.

[40] M. J. Huiskes and M. S. Lew, "The MIR Flickr Retrieval Evaluation," *MIR*, pp. 39–43, 2008.

[41] M. A. Tahir, F. Yan, M. Barnard, M. Awais, K. Mikolajczyk, and J. Kittler, "The University of Surrey Visual Concept Detection System at ImageCLEF 2010: Working Notes," *ICPR*, 2010.

[42] A. Binder, W. Samek, and M. Kawanabe, "The joint Submission of the TU Berlin and Fraunhofer FIRST (TUBFI) to the ImageCLEF 2011 Photo Annotation Task: Working Notes," *CLEF*, 2011.

[43] P. Koniusz and K. Mikolajczyk, "Spatial Coordinate Coding to Reduce Histogram Representations, Dominant Angle and Colour Pyramid Match," *ICIP*, 2011.

[44] J. Yang, Z. Wang, Z. Lin, X. Shu, and T. Huang, "Bilevel Sparse Coding for Coupled Feature Spaces," *CVPR*, 2012.

[45] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL Visual Object Classes Challenge 2007-2012 Results," http://pascallin.ecs.soton.ac.uk/challenges/VOC, 2012.

[46] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition," *ECCV*, 2014.

[47] L. Fei-fei, R. Fergus, and P. Perona, "Learning Generative Visual Models from Few Training Examples: An Incremental Bayesian Approach Tested on 101 Object Categories," *CVPR Workshop on Generative-Model Based Vision*, 2004.

[48] M. Awais, F. Yan, K. Mikolajczyk, and J. Kittler, "Novel Fusion Methods for Pattern Recognition," *ECML*, 2011.

[49] H. Azizpour, A. S. Razavian, J. Sullivan, A. Maki, and S. Carlsson, "From Generic to Specific Deep Representations for Visual Recognition," *arXiv*, 2014.

[50] S. Gao, I. W. Tsang, L. Chia, and P. Zhao, "Local Features Are Not Lonely - Laplacian Sparse Coding for Image Classification," *CVPR*, 2010.

[51] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva, "Learning Deep Features for Scene Recognition using Places Database," *NIPS*, 2014.

[52] B. Yao, X. Jiang, A. Khosla, A. L. Lin, L. J. Guibas, and L. Fei-Fei, "Action recognition by learning bases of action attributes and parts," *ICCV*, 2011.

[53] F. Khan, J. van de Weijer, R. M. Anwer, M. Felsberg, and C. Gatta, "Semantic pyramids for gender and action recognition," *TIP*, vol. 23, no. 8, pp. 3633–3645, 2014.

[54] A. Quattoni and A. Torralba, "Recognizing indoor scenes," *CVPR*, 2009.

[55] B. Yao, A. Khosla, and L. Fei-Fei, "Combining randomization and discrimination for fine-grained image categorization," *CVPR*, 2011.

[56] J. Xiao, K. A. Ehinger, J. Hays, A. Torralba, and A. Oliva, "SUN Database: Exploring a Large Collection of Scene Categories," *IJCV*, 2014.

[57] J. Sánchez, F. Perronnin, T. Mensink, and J. Verbeek, "Image classification with the fisher vector: Theory and practice," *IJCV*, vol. 105, no. 3, pp. 222–245, 2013.

[58] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online Learning for Matrix Factorization and Sparse Coding," *JMLR*, 2010.

[59] M. Tahir, J. Kittler, K. Mikolajczyk, F. Yan, K. van de Sande, and T. Gevers, "Visual Category Recognition using Spectral Regression and Kernel Discriminant Analysis," *ICCV Workshop on Subspace Methods*, 2009.

[60] P. Koniusz, "Novel Image Representations for Visual Categorisation with Bag-of-Words," Ph.D. dissertation, Centre for Vision, Speech and Signal Processing, University of Surrey, March 2013.

[61] J. Mairal, P. Koniusz, Z. Harchaoui, and C. Schmid, "Convolutional Kernel Networks," *NIPS*, 2014.

[62] F. Yan and K. Mikolajczyk, "Leveraging High Level Visual Information for Matching Images and Captions," *ACCV*, 2014.

[63] F Yan and K. Mikolajczyk, "Deep Correlation for Matching Images and Text," *CVPR*, 2015.

Piotr Koniusz is a researcher in Computer Vision Group at NICTA, Canberra, Australia. Previously, he worked as a post-doctoral researcher in the team LEAR, INRIA, Rhone-Alpes, France. He received his B.Sc. degree in Telecommunications and in Architecture and Design of Microcontroller Systems in 2004 from the Warsaw University of Technology, Poland. He completed his PhD degree in Computer Vision in 2013 at CVSSP, University of Surrey, Guildford, UK. His interests include visual concept detection, action recognition, feature and representation learning, invariance learning, as well as tensor, kernel and deep learning methods.

Fei Yan is a senior research fellow at University of Surrey in the United Kingdom. His research interests focus on machine learning, in particular kernel methods, structured learning, and deep neural networks. He is also interested in the application of machine learning to computer vision and natural language processing, such as object recognition, object tracking, natural language analysis and generation, and joint modelling of vision and language. He has publications in major machine learning and computer vision conferences and journals.

Philippe Henri Gosselin received the PhD degree in image and signal processing in 2005 (Cergy, France). After 2 years of post-doctoral positions at the LIP6 Lab. (Paris, France) and at the ETIS Lab. (Cergy, France), he joined the MIDI Team in the ETIS Lab as an assistant professor, and then promoted to full professor in 2012. His research focuses on machine learning for online multimedia retrieval. He developed several statistical tools for dealing with the special characteristics of content-based multimedia retrieval. This includes studies on kernel functions on histograms, bags and graphs of features, but also weakly supervised semantic learning methods. He is involved in several international research projects, with applications to image, video and 3D objects databases.

Krystian Mikolajczyk is an Associate Professor at Imperial College London. He completed his PhD degree at the Institute National Polytechnique de Grenoble and held a number of research positions at INRIA, University of Oxford and Technical University of Darmstadt, as well as faculty positions at the University of Surrey, and Imperial College London. His main area of expertise is in image and video recognition, in particular methods for image representation and learning. He has served in various roles at major international conferences co-chairing British Machine Vision Conference 2012 and IEEE International Conference on Advanced Video and Signal-Based Surveillance 2013. In 2014 he received Longuet-Higgins Prize awarded by the Technical Committee on Pattern Analysis and Machine Intelligence of the IEEE Computer Society.