Towards Optimal VLAD for Human Action Recognition from Still Images

Lei Zhang¹, Changxi Li¹, Peipei Peng¹, Xuezhi Xiang^{1*}, Jingkuan song²

¹ Harbin Engineering University, Harbin, China

² University of Trento, Trento, Italy

Abstract

Human action recognition from still image has recently drawn increasing attention in human behavior analysis and also poses great challenges due to the huge inter ambiguity and intra variability. Vector of locally aggregated descriptors (VLAD) has achieved state-of-the-art performance in many image classification tasks based on local features. The great success of VLAD is largely due to its high descriptive ability and computational efficiency. In this paper, towards optimal VLAD representations for human action recognition from still images, we improve VLAD by tackling three important issues including empty cavity, ambiguity and pooling strategies. The empty cavity limits the performance of VLAD and has long been overlooked. We investigate the empty cavity and provide an effective solution to deal with it, which improves the performance of VLAD; we enhance the codewords with middle level of assignments which are more reliable and can provide more useful information for realistic activity; we propose incorporating the generalized max pooling to replace sum pooling in VLAD, which is more reliable for the final representation. We have conducted extensive experiments on four widely-used benchmarks to validate the proposed method for human action recognition from still images. Our method produces competitive performance with state-of-the-art algorithms.

^{*}Partly supported by National Science Foundation of China (61571147,61401113), National Science Foundation of Heilongjiang (F2015027,LC201426) and the Fundamental Research Funds for the Central Universities (HEUCF150801)

^{*}Corresponding author (xiangxuezhi@hrbeu.edu.com)

Keywords: VLAD, empty cavity, ambiguity, generalized max pooling, human activity recognition

1. Introduction

Human action recognition from video can make the full advantage of structure and motion information [1, 2, 3]. Furthermore, together with temporal information, the system can be more robust to noisy background [4, 5]. While
5 the same task from still image poses even bigger challenges due to the limited useful information which deduces large intra-class variation and inter-class ambiguity.

Human action recognition from still images plays an important role in human behavior analysis. The bag-of-feature (BoF) [6, 7] is one of the most effective
10 framework in image and video representations based on local features. Recently, Fisher vector (FV) [8, 9] and its non-probabilistic version, i.e., vector of locally aggregated descriptor (VLAD) [10, 11] have been extensively used due to their high performances in different tasks including image retrieval, scene/object classification and human activity recognition[12, 13, 14, 15].

BoF encodes local features by calculating the histogram of local feature descriptors on a pre-learned codebook, and the histogram can be viewed as an approximate of the distribution of local features [16]. The Fisher vector, which is derived from the Fisher kernel, describes a signal by gradient vectors calculated from its probability density function [9]. Normally, the probability density
20 function is modeled by Gaussian mixture models (GMMs). In contrast to BoF using the hard assignment, FV with GMMs can be regarded as using soft assignment [17], which has shown better performance. Moreover, FV concerns not only on which codeword is activated but also on the gradient indicating the direction in which parameters should be adjusted to best fit the data. In
25 contrary to FV, VLAD avoids the estimation of the complex GMM model which is extremely time-consuming and achieves competitive and even better performance than FV [9]. In this work, we will build our method on VLAD to take

advantage of its high performance and computational efficiency.

BoF, FV and VLAD can be viewed in a unified framework which describes
30 images with local features by two main components: codebook creation and
feature encoding. Most researchers focus on these two aspects to generate good
codebooks and achieve discriminative encodings. However, the empty cavity
issue always happens in these methods and would compromise the performance
significantly, which will be thoroughly investigated in this work.

35 As for codebook creation, apart from the traditional k-means clustering algo-
rithm, hierarchical clustering [18] and spectral clustering [19] can also be applied,
avoiding to determine the size of codebook beforehand. With recent prevalence
of sparse coding [20], supervised [21, 22, 23] and unsupervised [24, 25, 26] code-
book generation approaches on sparse coding are proposed. The main distinc-
40 tion among these approaches on sparse coding rests on the constraint conditions,
such as locality-constrained linear coding (LLC) [27] limits the local descriptor
only to be encoded by the codewords in its neighbor, and local coordinate coding
(LCC) [25] adds the locality constraint from a different way.

With regards to assignment, extensive work including soft assignment [17],
45 triangle assignment [28], localized soft assignment [29] has been conducted to
avoid the limitation of hard assignment. Hard assignment assigns each local
descriptor to its closest codeword. However, soft assignment describes a local
descriptor by multiple codewords using a kernel function (e.g., Gaussian func-
tion) of the distance between local descriptor and codewords. Applying the
50 distance kernel function involved multiple codewords may be the key reasons of
the accuracy enhancement of probability density estimation [30] in soft assign-
ment. Triangle assignment and localized soft assignment are kinds of tradeoff
between hard and soft assignments. The former one only counts the distance
to the codeword whose distance is closer than the mean of the distances to all
55 codewords. Yet the latter one combines the idea of localization and the soft
assignment, where only the distance to the codeword is meaningful if the local
descriptor belongs to the k-nearest neighbors of this codeword.

However, no matter which approach is adopted to generate the codebook

and how the local descriptors are assigned to codewords, the problem of empty
60 cavity and imbalance of local descriptor assignments do exist. Although [31]
discussed the negative effect of empty cavity in BoF model and [29] gave the
analysis about ambiguity of codeword, the negative effects of empty cavity and
the imbalance influence of assignment in VLAD has long been neglected.

Towards optimal VLAD, in this paper, we improve VLAD by dealing with
65 three key issues in VLAD: empty cavity, ambiguity and pooling. We make the
following three major contributions:

- 1) By exploiting the negative effect of empty cavity theoretically and ex-
perimentally, we propose an effective method to tackle the empty cavity issue,
which can significantly improve the performance.
- 70 2) By investigating the distribution of the number of assignment in each
codeword in one image, and rebuilding the relations of the imbalance assignment
and codeword ambiguity, we propose selecting the reliable codewords to enhance
the weight of the pooled vector related to those codewords in VLAD.
- 3) We for the first time incorporate the generalized max pooling into the
75 construction of VLAD, which shows better performance in ordinary VLAD using
the sum pooling.

2. Still image datasets for human action recognition

Human action recognition from still images has played increasing important
role in human behavior analysis and it poses even more challenges than video-
80 based action recognition. Before introducing our method, we would like to
describe four challenging and widely-used image datasets that are used in this
work. As shown in Fig.1, the most difficult factor in each dataset is listed. In
PPMI, confusing examples of holding and playing instruments are the hardest
factor to be distinguished, as shown in sub-figure (a). Similarly, for actions from
85 the same class in Stanford 40 Action, the poses and backgrounds are different
such as brushing and cooking in sub-figure (b). Moreover, sub-figure (c) in
Fig.1 gives the examples of small object hard to be detected in UIUC sports,

and similar pose and background examples cross-class in Sports 6 are listed in sub-figure (d).

- 90 • PPMI [32] contains images of humans interacting with twelve different musical instruments. They are: bassoon, cello, clarinet, erhu, flute, French horn, guitar, harp, recorder, saxophone, trumpet and violin. For each instrument, there are images that contain a person playing the instrument (PPMI+), as well as images that contain a person holding the instrument without playing
95 (PPMI-). Here we treat it as a classification task with 24 classes. We use 100 images in each class for training and the rest 100 images for testing.
- Stanford 40 Action [33] contains images of 40 diverse daily human action, such as ‘brushing teeth’, ‘cleaning the floor’, ‘reading book’, ‘throwing a frisbee’, etc. All images are obtained from Google, Bing and Flickr. There are 9532
100 images in total with 180~300 images per action class. The images within each class have large variations in human pose, appearance and background clutter. We use 100 images in each class for training and all the rest images for testing as in [33].
- UIUC Sports [34] contains 8 sports categories collected from internet: bocce,
105 croquet, polo, rowing, snowboarding, badminton, sailing and rock climbing. The number of images in each category varies from 137 (bocce) to 250 (rowing). The background of each image is highly cluttered and human poses largely diverse. Furthermore, as shown in Fig. 1, the foreground human poses are too small to be detected in some images. We follow the experimental setting in [35] by randomly selecting 70 images as training set and 60
110 images as testing set, respectively.
- Sports 6 [36] contains images from six different sports as ‘tennis-forehand’, ‘tennis-serve’, ‘volleyball-smash’, ‘cricket-defensive-shot’, ‘cricket-bowling’ and ‘croquet-shot’ with 50 images per class. This dataset is with significant con-
115 fusion due to similar pose in different action. For example, the poses in ‘volleyball-smash’ are with high similarity to those in ‘tennis-serve’ and the

(a) Confusing examples of holding and playing in PPMI

(b) Different pose and background within-class in Stanford 40 Action

(c) Small object in UIUC sports

(d) Similar pose and background cross-class in Sports 6

Figure 1: The typical examples in four datasets demonstrating challenges.

Figure 2: Empty cavity rate changed with different codebook size over four datasets

backgrounds from ‘tennis-forehand’ and ‘tennis-serve’ are exactly the same shown in Fig. 1. We use fixed 30 images in each class for training and the other 20 images for testing.

3. Exploiting negative impact for empty cavity in VLAD

3.1. Empty cavity

Empty cavity means during the assignment of local descriptors in one image, there are always some codewords with no local descriptor assigned to, leaving empty in the final representation. As a result, the obtained representation tends to be less discriminative and therefore compromises the recognition performance. The phenomenon of empty cavity does widely exist in both BoF and VLAD, which severely compromises the overall performance. To show the empty cavity problem, Fig.2 gives the empty cavity rate for above four datasets, where the empty cavity rate means average number of empty cavity over training data divided by the size of codebook.

It can be seen that for some datasets, the empty cavity rate is extremely high. For example, the empty cavity rate in PPMI dataset is up to 19.36% with 512 of the codebook size, which means nearly 100 codewords with no local descriptor assigned to. Additionally, the ascending trend of the empty cavity

135 rate with increasing of the codebook size is obvious in this figure. That means,
if the codebook size is further increased as 1024 or 2048, the empty cavity
phenomenon is much more serious.

3.2. Negative impact of empty cavity

3.2.1. Revisit of VLAD

140 Let $\{\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_K\}$ be the codebook learned by k-means. For one image
 X , each local descriptor $\mathbf{x}_t \in R^D$ is assigned to the nearest codeword as $\mathbf{u}_i =$
 $NN(\mathbf{x}_t)$. Moreover, the *pooled vector* $\mathbf{v}_i$ for codeword i is computed by :

$$\mathbf{v}_i = \sum_{\mathbf{x}_t: NN(\mathbf{x}_t)=\mathbf{u}_i} (\mathbf{x}_t - \mathbf{u}_i) \quad (1)$$

which is sum pooling strategy of the *residual vectors*, *i.e.*, the subtraction of
local descriptor $\mathbf{x}_t$ and its belonging codeword. Finally, K pooled vectors are
145 concatenated as a single $K \times D$ dimensional vector.

By deeply analyzing the pooled vector $\mathbf{v}_i$, it can be seen that the distances of
local descriptors to codewords are encoded. Thus the similarities relative to the
codewords during matching are all incorporated which increases the accuracy of
measuring the relationship between local descriptors. While for BoF, only the
150 number of pairs of local descriptors assigned to the same codeword in two images
are counted, with no consideration about the similarity of local descriptors. This
is the main reason for the better performance of VLAD.

3.2.2. Negative effect of empty cavity from kernel aspect

Traditional methods including the standard VLAD does not provide any way
155 to handle empty cavity phenomenon, which means the *pooled vector* $\mathbf{v}_i$ is zero
for empty cavity codeword. We will demonstrate the negative effect of this zero
vector from the kernel aspect.

In both classification and retrieval tasks, the essential part is to compute the
similarity between two images X and Y . Kernel is one way to fulfill this aim.

160 Let $\mathbf{X} = \{\overbrace{x_1, \dots, x_D}^{\mathbf{X}_1}, \dots, \overbrace{x_{D \times (K-1)+1}, \dots, x_{D \times K}}^{\mathbf{X}_K}\}$ be the VLAD representations

Figure 3: Part examples of empty cavity index between two randomly selected images in PPMI, where pairs in black rectangles show the codewords solely missed condition and pairs in green ellipses present the codewords jointly missed condition. Both conditions are empty cavity cases, which are harmful for final performance.

for image X and $\mathbf{Y} = \{\overbrace{y_1, \dots, y_D}^{\mathbf{Y}_1}, \dots, \overbrace{y_{D \times (K-1)+1}, \dots, y_{D \times K}}^{\mathbf{Y}_K}\}$ is VLAD for image Y . Then we can kernelize the match between $\mathbf{X}$ and $\mathbf{Y}$ as:

$$K(\mathbf{X}, \mathbf{Y}) = \sum_i K_i(\mathbf{X}_i, \mathbf{Y}_i) \quad (2)$$

where $K_i(\mathbf{X}_i, \mathbf{Y}_i)$ presents the *part kernel* in cluster i . The essence of Eq. (2) is to decompose the whole kernel function into several independent elements, and each element only focuses on the part generated by one cluster during VLAD.

Without loss of generality, we consider a linear kernel widely used for recognition, and therefore we have

$$K(\mathbf{X}, \mathbf{Y}) = \mathbf{X}^T \mathbf{Y} = \sum_i \mathbf{X}_i^T \mathbf{Y}_i \quad (3)$$

Where $\mathbf{X}_i$ is part representation build on cluster i , namely, *pooled vector* above, and $K_i(\mathbf{X}_i, \mathbf{Y}_i) = \mathbf{X}_i^T \mathbf{Y}_i$.

From Eq. (3), it is obvious that if $\mathbf{X}_i$ happens to be a zero vector caused by empty cavity in cluster i , and no matter what value $\mathbf{Y}_i$ is, the *part kernel* K_i is the same. This kind of matching has misleading result under two conditions:

- 175 1) When the codeword is mutually missed in two images, *i.e.*, both $\mathbf{X}_i$ and $\mathbf{Y}_i$ are zero vectors, the codeword that can give more information will receive higher weights in the similarity measurement. 2) When the codeword is missed in only one image, that is, either $\mathbf{X}_i$ or $\mathbf{Y}_i$ is a zero vector, the final similarity measurement should vary with non-zero vector left, rather than fixed zero.

180 Fig.3 is part figure of the assignments for each codeword between two randomly selected images in the PPMI dataset, which shows the above two conditions. Firstly, it can be seen that the number of points from image 1 or image on the horizontal axis occupies a certain proportion, which means the empty cavity is not an accidental phenomena. Secondly, the pairs in green ellipses express the condition of codeword jointly missed in two images, corresponding to zero assignment for those codewords. Additionally, the pairs in black rectangles present the condition that the empty cavity codeword index in images X and Y does not happen at the same location. For example, for the highest rectangle, the assignments of local descriptors is eight for image 2 and zero for image 1.

190 Another thing worth to notice is that the above two conditions of empty cavity do not happen occasionally.

3.3. Solutions to empty cavity

In order to solve the empty cavity problem, a non-zero reference vector should be found for pooled vector $\mathbf{v}_i$ by intuition. To obtain a better insight into reference vector selection, Fig.4 illustrates the procedure of VLAD and shows the meaning of a reference vector in our approach. In Fig.4, the real blue arrows and black dashed arrows represent the residual vector as $\mathbf{x}_t - \mathbf{u}_i$ in VLAD from two images. Moreover, the real red arrow and the dashed red arrows represent the pooled vectors from different images. For a certain codeword which local descriptors from both images are assigned to, as the dashed rectangles parts, the *part kernel* is to compute the inner product between point a and point b where

200

Figure 4: Toy illustration for VLAD and empty cavity solution

the origin is the related codeword. However, there are also two other conditions for codewords as shown by dashed ellipses and dashed triangle in Fig.4.

The dashed ellipse case is to show the mismatched empty cavity condition
 205 between two images (for example, for codeword i , where only two local descriptors in image 2 are assigned and no points in image 1 is in this cavity). The other case is for co-missing codeword j in both two images, which is shown by dashed triangle part. The standard VLAD is to neglect the effect of this codeword under these two conditions. No matter how many local descriptors in
 210 image 2 are assigned to the cluster i , the part kernel has no difference, which is zero. Similar, for the dashed triangle case, the part kernel is also zero. To fix this problem, we aim to find a point as the reference to keep the pooled vector nonzero. For empty cavity from distinct image, this reference point should keep unchanged. b_1 , b_2 and b_3 are three strategies of reference point which is independent to different image itself. They are type I to type III cases to replace
 215 the pooled vector as follows.

1) Type I:

$$\hat{\mathbf{v}}_i = \frac{1}{K} \sum_{k=1}^K \mathbf{u}_k - \mathbf{u}_i \quad (4)$$

This is to treat the mean of all codewords (blue four-points star) as the reference point in Fig. 4. Then the pooled vector for image 1 is the vector starting from codeword i and ending to this blue four-points star. The main idea for this strategy is to find the point with highest probability as the reference point in R^D space, since the average of all codewords is the real center of training data points.

2) Type II:

$$\hat{\mathbf{v}}_i = \frac{1}{\text{card}(\mathcal{S})} \sum_{k \in \mathcal{S}} \mathbf{u}_k - \mathbf{u}_i \quad (5)$$

where $\mathcal{S}$ is the set that codeword is close to codeword i and at the same time, the number of points assigned to this codeword is smaller than a threshold. The nearby codewords with smaller assignments are similar to the codeword with no assignment. This is the case of the red arrow to five-points star in Fig. 4.

3) Type III:

$$\hat{\mathbf{v}}_i = \mathbf{u}_i \quad (6)$$

For this case, it just treats the codeword itself as the pooled vector if empty cavity happens.

4. Ambiguity in assignment

Different from text retrieval systems, where the document vector space is discrete itself, the local descriptor space R^D is a continuous one. When we use limited visual words to represent the whole continuous space, there will be an ambiguity problem. The ambiguity between codewords will be influenced by the number of words in the codebook. On the one hand, a big size of codebook allows a rich selection of codewords, while the small vocabulary essentially leads to different image parts represented by the same vocabulary element with more serious ambiguity. On the other hand, big size of vocabulary will aggravate the sparse situation and increase computational complexity.

Figure 5: Variation of local descriptor number assigned to different codewords over different images, where the left figure is from *playing bassoon* class and right is from *playing clarinet* in the PPMI dataset

The ambiguity in codeword can be reflected in two-field. One is the misrepresentation situation, that is, the codeword could not represent the characteristic of local descriptors. The other case corresponds to the codewords with higher assignments, which is similar as the useless *function word* in document vector space. Although the *function word* as ‘is’ or ‘are’ may happen in one document many times, it is still meaningless in semantic analysis for classification or recognition.

In this section, we aim to build the relations between the number of assignment and ambiguity, and then give the proper solutions to enhance the codeword with less ambiguity.

4.1. Analysis of ambiguity

Fig. 5 presents the variation of local descriptor assignment number in the PPMI dataset, where left sub-figure and right sub-figure are from distinct categories. It can be seen that except several high peaks in both sub-figures, others are flat among images and different codeword index. It is also worth to notice that for different classes, the locations of peaks in Fig. 5 are different.

For the number of local descriptors assigned to different codewords, we divides the corresponding ranges into three parts: higher assignments as the peak or close to peak values in Fig. 5, lower assignments close to zero, and middle assignments which are neither higher nor lower parts.

Fig. 6 gives the location information of local descriptors with different num-

Figure 6: The locations of local descriptors of PPMI image with higher and lower assignments (left) and with middle assignments in Fig.5 (right)

ber of assignments to one codeword. The blue circles in left sub-figure represent
 265 the locations of local descriptors in the cavity with highest assignments, and
 each red stars are the locations for those cavities with only one assignment.
 Moreover, stars of four different colors (pink, red, yellow and green) in right
 sub-figure in Fig. 6 show the locations of local descriptors in the cavity with
 middle level of assignments.

270 It is clear to see that the locations in the left sub-figure could not grasp the
 essence of the image. In fact, the blue circles in Fig. 6 express the burstness
 phenomenon, *i.e.*, the property that a given visual element appears more times
 in an image than a statistically independent model would predict [37]. Since
 the visual word with burstness mostly provides the background information, it
 275 should be attenuated during later processing. These peak or near peak assign-
 ments correspond to the ambiguity situation similar to that *function words* in
 the document vector space.

However, the red stars carry similar indiscriminate information for recogni-
 tion in the left sub-figure of Fig. 6, which is related to the misrepresentation
 280 of the ambiguity. It is interesting to find that this kind of one assignment ap-
 pearance does not occasionally happen and may exist many times in one image
 (e.g., over 20 times in left sub-figure of Fig. 6). It also should be scaled down

during recognition.

Moreover, most stars related to the middle level assignments in the right
sub-figure of Fig. 6 can represent the key information which is crucial to the
recognition performance. The information grasped by the middle level assign-
ments should be further enhanced.

4.2. Enhancing codewords with middle level assignment

Burstness means that during assignment, some codewords are much more
frequently selected than others. Inspired from text retrieval [38], most existing
approaches use an inverse document frequency (idf) weighting scheme to allevi-
ate the codeword effect which happens in all images. In fact, the traditional idf
could not directly attenuate the burstness, especially when the size of codebook
is large. In Fig. 5, it can be seen the peaks do not happen at fixed locations
along codeword index axis and image index axis, which means the codeword
with peak assignment in an image may have zero assignment in other images.

Moreover, [37] proposed the modified idf with the score normalization. The
normalization factor is to sum the score over the descriptors in the same code-
word in the image for intra-image burstness or to sum the score over all descrip-
tors in all images for inter-image burstness. The shortcomings of this modified
idf lie in two aspects: 1) The normalization factor is computed over all local
descriptors in all images, which will increase the computational burden. 2) It
is more suitable for a retrieval task rather than a recognition task, and it does
not consider the situation of lower assignments problem.

For the unbalance assignment to each codeword, we propose to enhance the
codewords with the middle level assignment in VLAD instead of penalizing the
codewords with the higher or lower level assignments, since the middle level
assignment has less ambiguity. In order to determine the range of middle level
assignment, we assume the probability distribution function of the number of
assignments as Gaussian distributions. Then the range can be determined by

the mean and standard variance as follow:

$$range = [\max(c_1, N_\mu - k_1 N_\sigma), \min(c_2, N_\mu + k_2 N_\sigma)] \quad (7)$$

where N_μ is the mean number of assignments in one image, and σ is corresponding standard variance. k_1, k_2, c_1 and c_2 are constants determined by cross validation in following experiments.

315 Then the pooled vector in VLAD is changed as:

$$v_i = \begin{cases} w \sum_{\mathbf{x}_t: NN(\mathbf{x}_t)=i} (\mathbf{x}_t - \mathbf{u}_i) & \text{if } Card(\{\mathbf{x}_t : NN(\mathbf{x}_t) = i\}) \in range \\ \sum_{\mathbf{x}_t: NN(\mathbf{x}_t)=i} (\mathbf{x}_t - \mathbf{u}_i) & \text{if } Card(\{\mathbf{x}_t : NN(\mathbf{x}_t) = i\}) \notin range \end{cases} \quad (8)$$

where w is bigger than one, and we vary this weight from 1 to 2.4 with 0.2 step increment.

5. Generalized max pooling in VLAD

Pooling technique has played an important role in image representation
320 based on local descriptors. Max pooling has shown its great advantages over other pooling strategies including average pooling and sum pooling. However, max pooling has not yet been investigated in VLAD due to non-trivial use in it. In this paper, we propose a new method to incorporate the newly proposed generalized max pooling strategy into VLAD.

325 It has been indicated in [39] that max pooling is particularly well-suited to the separation of features with high sparsity. However, the conventional max pooling can not be directly incorporated into VLAD. We investigate and adopt the recently proposed generalized max pooling (GMP) [40] into VLAD, which shows improved performance for all the different tasks.

330 The generalized max pooling involves equalizing the similarity between local descriptors and the pooled representation, which is formulated as:

$$\varphi_t^T \varphi^{gmp} = m \quad (9)$$

where φ_t is the local descriptor and φ^{gmp} is the vector after generalized max pooling. When applying this rule into VLAD and setting $m = 1$ since this constant has no influence for final classification, this equation is changed into:

$$(\mathbf{x}_i^t - \mathbf{u}_i)^T \varphi^{gmp} = 1 \quad (10)$$

335 Expanding Eq. (10) into matrix form for cluster i , it is can be rewritten as:

$$[(\mathbf{x}_i^1 - \mathbf{u}_i) (\mathbf{x}_i^2 - \mathbf{u}_i) \cdots (\mathbf{x}_i^{N_i^X} - \mathbf{u}_i)]^T \varphi^{gmp} = \mathbf{R}^T \varphi^{gmp} = \mathbf{1}_{N_i^X} \quad (11)$$

where $\mathbf{1}_{N_i^X}$ denotes the N_i^X dimensional vector of all ones and $\mathbf{R}$ corresponds to the residual matrix. In practice, we can optimize the φ^{gmp} by following equation as:

$$\varphi^{gmp} = \arg \min_{\varphi} \|\mathbf{R}^T \varphi^{gmp} - \mathbf{1}_{N_i^X}\|^2 + \lambda \|\varphi\|^2 \quad (12)$$

This is a ridge regression problem whose solution is

$$\varphi^{gmp} = (\mathbf{R}\mathbf{R}^T + \lambda \mathbf{I})^{-1} \mathbf{R} \mathbf{1}_{N_i^X} \quad (13)$$

340 The regularization parameter λ can be determined during experiments. When λ is large enough, it backs to sum pooling as $\varphi^{gmp} \approx \mathbf{R} \mathbf{1}_{N_i^X} / \lambda$.

6. Experiments and results

We have evaluated our method on the four challenging datasets extensively, provided in-depth analysis on the performance, and compared with existing
345 algorithms on the datasets. The proposed method has significantly improved the original VLAD and largely outperformed state-of-the-art algorithms.

6.1. Experimental setting

For local descriptors, we extract dense SIFT descriptors with 3×3 grids for the better performance of dense sampling on image classification than sparse
350 interest points [41]. It has been shown that directly applying on raw SIFT leads to suboptimal performance. Therefore, we follow existing work [10] to

conduct PCA on the raw SIFT to reduce the dimensionality to 100, which shows improved performance in previous work.

The codebook is learned by the k-means clustering algorithm with randomly
 355 selecting 20% training data from each class for computational efficiency. In most cases, the codebook size is selected as 512 unless specified.

For VLAD, L_2 normalization is conducted on the SIFT descriptor, pooled vectors and final aggregated vectors. It has been shown that L_2 normalization on SIFT is helpful for background suppression and accurate assignment in high
 360 dimensional space. Moreover, L_2 normalization on pooled vector can further alleviate the burstiness of visual words which heavily influences the final performance [37]. At last, L_2 normalization on final aggregated vector is applied for eliminating the magnitude influence among different images from distinct classes.

A linear SVM classifier [42] is adopted for final recognition and classification
 365 task to benchmark with the compared algorithms. The parameters k_1 , k_2 in Eq. (7) are varied from 1 with ± 0.2 steps. c_1 and c_2 are initially selected as $\{2, 70\}$ on the analysis of Fig. 5, and then are varied from the initial value with $\{\pm 2, \pm 5\}$ steps respectively. The final value are set as 0.6, 1.8, 5 and 80 by cross
 370 validation in following experiments.

6.2. In-depth analysis on four still image datasets

In this section we investigate the benefits of the proposed solutions to the empty cavity and ambiguity issues in codewords together with generalized max pooling both separately and jointly.

375 6.2.1. Impact of different solutions to empty cavity

In Fig. 7, we report the comparisons of different solutions to empty cavity on four datasets for (a) sum pooling and (b) GMP in VLAD and show the advantages of the proposed solutions over the standard VLAD.

Sub-figure (a) in Fig. 7 reflects the effectiveness of different solutions to
 380 empty cavity in standard VLAD. It can be observed that all types of solutions

(a) Sum pooling in VLAD

(b) GMP in VLAD

Figure 7: Comparison of different solutions to empty cavity in four datasets with sum pooling and generalized max pooling (GMP) in VLAD

outperform the standard VLAD. The improvements are significant on UIUC sports and Sport 6 datasets. On PPMI and Stanford 40 Action datasets, although the enhancement is not as obvious as shown in top right sub-figure, there are still some effectiveness on overall performance. Secondly, it is noticed that the strategy with highest performance is different on these four datastes.

However, when GMP is applied to VLAD in sub-figure (b) of Fig. 7, the effectiveness of different solutions to empty cavity are not as obvious as those in sub-figure (a). As for the comparison of sum pooling and GMP, from the final performance, except the Sports 6 dataset, the performance on other three datasets is largely enhanced by GMP. The performance of GMP in Sports 6 dataset may be restricted by the limited training data. Additionally, comparing the first dark green bar in each sub-figure, the effect of GMP separately can be shown, where the improvements of GMP on Stanford 40 and UIUC are impressive.

Figure 8: Accuracy changing with different weight added to the pooled vectors of *sum pooling* in VLAD with middle level assignments on four datasets

6.2.2. Effect of different middle level assignments

We tune the weight from 1 to 2.4 with 0.2 step increment on four datasets. The tuning results are shown in Fig. 8 for *sum pooling* and Fig. 9 for *GMP* in VLAD. In fact, Fig. 8 gives the separate influence of ambiguity solution. The trend is flattened in the upper sub-figure, while there is some oscillating in the lower sub-figure in both Fig. 8 and Fig. 9. From these results, it is clear that the enhancement of the codewords with middle level assignment is effective by properly selecting the weights.

6.2.3. Joint evaluations on four datasets

In order to compare with the state-of-the-art performance on these four datasets, in this section, both accuracy and mean average precision (MAP) are considered. According to the results in Fig. 9, we list the best performance of joint solution of proposed approach and the comparison to VLAD with *sum pooling* and *GMP* in Table 1. We discuss the performance of joint solution of empty cavity and ambiguity together with *GMP* on the four datasets one by one.

- 1) *PPMI*: No matter for accuracy or MAP, the improvement of joint solution

Figure 9: Accuracy changing with different weight added to the pooled vectors of *GMP* in VLAD with middle level assignments on four datasets

over standard VLAD with sum pooling is more than 1%. While together with *GMP* pooling strategy, the enhancement is weakened, which is less than 0.5%. Additionally, *GMP* instead of sum pooling in standard VLAD can boost system performance on this dataset.

2) *Stanford 40 Action*: Three observations can be made from the results on this dataset in Table 1. First, although the performance difference for joint solution is not so obvious as that on PPMI, there are still some improvements over VLAD regardless of pooling strategies. Second, effectiveness of *GMP* has no consistency in accuracy and MAP. Although it will improve the performance with respect to accuracy, there is a little decline for MAP. Last, it is worth to notice that the best MAP is achieved by combination of *GMP* and joint solution approach.

3) *UIUC sports*: Among these four datasets, this is the most suitable one with the remarkable improvement of proposed scheme. The improvements of joint solution for both accuracy and MAP are more than 2.5% over VLAD with sum pooling strategy. On the contrary, the final performances descend about 0.5% in conjunction with *GMP*. The best result of accuracy is 88.12% and highest MAP achieves 88.94%.

4) *Sports 6*: Since the limited training data in this dataset, *GMP* has no

Dataset	Pooling strategy	Accuracy		MAP	
		VLAD	joint solution (weight/type)	VLAD	joint solution (weight/type)
PPMI	sum pooling	49.33%	50.50% (1.8/Type II)	47.07%	48.60% (1.8/Type II)
	GMP	50.17%	50.67% (1.8/Type II)	48.27%	48.60% (1.8/Type I)
Stanford 40 Action	sum pooling	39.79%	40.37% (1.4/Type II)	36.72%	36.78% (1.4/Type II)
	GMP	40.42%	40.93% (1.6/Type II)	36.51%	37.14% (1.6/Type II)
UIUC sports	sum pooling	85.42%	88.12% (1.8/Type I)	86.22%	88.94% (1.8/Type I)
	GMP	86.04%	87.50% (1.6/Type I)	86.52%	88.39% (1.6/Type II)
Sports 6	sum pooling	89.17%	90.83% (2.4/Type I)	90.23%	91.55% (2.4/Type I)
	GMP	87.50%	90.83% (2.4/Type I)	88.75%	91.55% (2.4/Type I)

Table 1: Accuracy and MAP comparisons among joint solution to empty cavity and ambiguity in codewords in VLAD (sum pooling and GMP) on four datasets

PPMI	method	MAP	Sports 6	method	MAP
	VLAD [10]	47.07%		HMI [43]	83.00%
	SPM [44]	39.10%		SFC [36]	79.00%
	LLC [27]	41.80%		MMC [45]	83.00%
	Grouplets [32]	36.70%		HOI [46]	87.00%
	BoW	22.70%		Semantic Pyramids [47]	92.50%
	our scheme	48.06%		our scheme	91.55%
UIUC sports	method	Accuracy	Stanford 40 Action	method	MAP
	LDA [34]	73.40%		OB [35]	32.50%
	ScSPM [24]	83.46%		SPM [44]	34.90%
	LLC [27]	83.46%		LLC [27]	35.20%
	Salient coding [48]	81.83%		Sparse bases [33]	45.70%
	HMP [49]	85.70%		EPM [50]	42.20%
	our scheme	88.12%		our scheme	37.14%

Table 2: Comparison of our scheme with the state-of-the-art on four still image datasets on human activity

positive influence on this dataset. However, considering the empty cavity and ambiguity in codewords can enhance the performance compared with standard VLAD, and it can also compensate the degrade performance caused by GMP.

6.3. Comparison with the state-of-the-art

435 We compare our method with existing algorithms in Table 2. It can be seen that our method produces a much better performance than most of the state-of-the-art algorithms. Our method has great superior performance on the PPMI and UIUC sports datasets and the competitive results on the Sports 6 and Stanford 40 Action datasets compared to other approaches.

440 7. Conclusion

We have presented an improved VLAD for human action recognition from still images based on local features. Towards optimal VLAD, we propose improving three key components in the construction of VLAD, which includes empty cavity, codeword ambiguity and generalized max pooling (GMP). We

445 successfully tackled the empty cavity and codeword ambiguity problems and for
the first time incorporated the generalized max pooling into VLAD. Our method
significantly improved the performance of VLAD and achieved competitive and
even better performance than that of state-of-the-art algorithms on four widely
used benchmark datasets for human action recognition from still images.

450 References

- [1] X. Zhen, L. Shao, D. Tao, X. Li, Embedding motion and structure features
for human action recognition, *IEEE Transactions on Circuits and Systems
for Video Technology* 23 (7) (2013) 1182–1190.
- [2] X. Zhen, L. Shao, X. Li, Action recognition by spatio-temporal oriented
455 energies, *Information Sciences* 281 (2014) 295–309.
- [3] M. Yu, L. Liu, L. Shao, Structure-preserving binary representations for
rgb-d action recognition, *PAMI*.doi:10.1109/TPAMI.2015.2491925.
- [4] L. Shao, L. Liu, M. Yu, Kernelized multiview projection for robust action
recognition, *IJCV*.doi:10.1007/s11263-015-0861-6.
- 460 [5] X. Zhen, D. Tao, X. Li, Spatio-temporal laplacian pyramid coding for action
recognition, *IEEE Transactions on Cybernetics* 44 (6) (2014) 817–827.
- [6] J. Sivic, A. Zisserman, Video google: A text retrieval approach to object
matching in videos, in: *CVPR.*, 2003.
- [7] H. Jegou, M. Douze, C. Schmid., Improving bag-of-features for large scale
465 image search, *International Journal of Computer Vision* 87 (3) 316336.
- [8] F. Perronnin, J. Sanchez, T.Mensink, Improving the fisher kernel for large-
scale image classification, in: *ECCV.*, 2010.
- [9] J. Snchez, F. Perronnin, T. Mensink, J. Verbeek, Image classification with
the fisher vector: Theory and practice, *International Journal of Computer
470 Vision* 105 (3) 222–245.

- [10] H. Jegou, M. Douze, C. Schmid, P. Perez, Aggregating local descriptors into a compact image representation, in: CVPR., 2010.
- [11] H. Jegou, F. Perronnin, M. Douze, J. Sanchez, Aggregating local image descriptors into compact codes, IEEE Transactions on pattern analysis and machine intelligence 34 (9) (2012) 1704–1716.
- [12] K. E. van de Sande, C. G. Snock, A. W. Smeulders, Fisher and vlad with flair, in: CVPR 2014., 2014.
- [13] G. Philippe-Henri, M. Naila, J. Herve, P. Florent, Inria+xerox@fgcomp: Boosting the fisher vector for fine-grained classification, Tech. Rep. 0, INRIA (December 2013).
- [14] E. Spyromitros-Xioufis, S. Papadopoulos, I. Y. Kompatsiaris, G. Tsoumakas, I. Vlahavas, A comprehensive study over vlad and product quantization in large-scale image retrieval, IEEE Transactions on Multimedia 16 (6) (2014) 1713–1728.
- [15] R. Arandjelovic, A. Zisserman, All about vlad, in: CVPR., 2013.
- [16] Y. Boureau, F. Bach, Y. Lecun, J. Ponce, Learning mid-level features for recognition, in: CVPR., 2010.
- [17] L. Liu, L. Wang, X. Liu, In defense of soft-assignment coding, in: ICCV., 2011.
- [18] W. Zhang, D. Zhao, X. Wang, Agglomerative clustering via maximum incremental path integral, Pattern Recognition 46 (2013) 3056–3065.
- [19] F. Bach, M. Jordan, Learning spectral clustering, in: NIPS., 2004.
- [20] J. Mairal, F. Bach, J. ponce, G. Sapiro, Online dictionary learning for sparse coding, in: ICML., 2009.
- [21] J. Yang, K. Yu, T. Huang, Supervised translation-invariant sparse coding, in: CVPR., 2010.

- [22] D.M.Bradley, J.A.Bagnell, Differential sparse coding, in: NIPS., 2008.
- [23] J.Mairal, F. bach, J.Ponce, G. Sapiro, Supervised dictionary learning, in: NIPS., 2008.
- 500 [24] J.Yang, K. Yu, Y.Gong, T. Huang, Linear spatial pyramid matching using sparse coding for image classification, in: CVPR., 2009.
- [25] K. Yu, T. Wang, Y. Gong, Nonlinear learning using local coordinate coding, in: NIPS., 2009.
- [26] Z. Jiang, G. Zhang, L. Davis, Submodular dictionary learning for sparse
505 coding, in: CVPR., 2012.
- [27] J. Wang, J.Yang, K. Yu, F. Lv, T. Hunag, Y.Gong, Locality-constrained linear coding for image classification, in: CVPR., 2010.
- [28] A. Coates, H. Lee, A. Y. Ng, An analysis of single-layer networks in unsupervised feature learning, in: AISTATS., 2011.
- 510 [29] J. Hemert, C. Veenman, A. Smeulders, J. Geusebroek, Visual word ambiguity, IEEE Transactions on pattern analysis and machine intelligence 32 (7) (2010) 1271–1283.
- [30] Y. Huang, Z. Wu, L. Wang, T. Tan, Feature coding in image classification: a comprehensive study, IEEE Transactions on pattern analysis and machine
515 intelligence 36 (3) (2014) 493–506.
- [31] O. C. Herv Jgou, Negative evidences and co-occurences in image retrieval: The benefit of pca and whitening, in: ECCV., 2012.
- [32] B. Yao, L. Fei-Fei, Grouplet: A structured image representation for recognizing human and object interactions, in: CVPR., 2010.
- 520 [33] B. Yao, X. Jiang, A. Khosla, A. L. Lin, L. J. Guibas, L. Fei-Fei, Human action recognition by learning bases of action attributes and parts, in: ICCV, Barcelona, Spain, 2011.

- [34] L. Li, L. Fei-Fei, What, where and who? classifying events by scene and object recognition, in: CVPR., 2007.
- 525 [35] L. Li, H. Su, E. P. Xing, L. Fei-Fei, Object bank: a high-level image representation for scene classification and semantic feature sparsification, in: NIPS., 2010.
- [36] A. Gupta, A. Kembhavi, L. Davis, Observing human-object interactions: using spatial and functional compatibility for recognition, IEEE Transactions on Pattern Analysis 31 (10) (2009) 1775–1789.
- 530 [37] H. Jegou, M. Douze, C. Schmid, On the burstiness of visual elements, in: CVPR., 2009.
- [38] G. Salton, C. Buckley, Term-weighting approaches in automatic text retrieval, Information Processing and Management 24 (5) (1998) 513–523.
- 535 [39] Y.-L. Boureau, J. Ponce, Y. LeCun, A theoretical analysis of feature pooling in visual recognition, in: ICML., 2010.
- [40] N. Murray, F. Perronnin, Generalized max pooling, in: CVPR., 2014.
- [41] E. Nowak, F. Jurie, B. Triggs, Sampling strategies for bag-of-features image classification, in: ECCV, 2006.
- 540 [42] R. Fan, K. Chang, C. Hsieh, X. Wang, C. Lin, Liblinear: a library for large linear classification, JCLR 9 (2008) 1871–1874.
- [43] A. Prest, C. Schmid, V. Ferrari, Weakly supervised learning of interactions between humans and objects, PAMI 34 (3) (2012) 601–614.
- [44] S. Lazebnik, C. Schmid, J. Ponce, Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories, in: CVPR., 2006.
- 545 [45] B. Yao, F.-F. Li, Modeling mutual context of object and human pose in human-object interaction activities, in: CVPR., 2010.

- [46] B. Yao, F.-F. Li, Recognizing human-object interactions in still images by modeling the mutual context of objects and human poses, PAMI 34 (9) (2012) 1691–1703.
- 550 [47] F. S. Khan, J. van de Weijer, R. M. Anwer, M. Felsberg, C. Gatta, Semantic pyramids for gender and action recognition, TIP 23 (8) (2014) 3633–3645.
- [48] Y. Huang, K. Huang, Y. Yu, T. Tan, Salient coding for image classification, in: ICCV., 2011.
- 555 [49] L. Bo, X. Ren, D. Fox., Hierarchical matching pursuit for image classification: architecture and fast algorithms, in: NIPS., 2011.
- [50] G. Sharma, F. Jurie, C. Schmid, Expanded parts model for human attribute and action recognition in still images, in: CVPR., 2013.