LARGE-SCALE SEMI-SUPERVISED LEARNING BY APPROXIMATE LAPLACIAN EIGENMAPS, VLAD AND PYRAMIDS

Eleni Mantziou, Symeon Papadopoulos, Yiannis Kompatsiaris

CERTH- ITI, Thessaloniki, Greece

ABSTRACT

The paper builds upon recent advances in feature representation and dimensionality reduction to propose a semi-supervised image annotation framework that achieves state-of-the-art accuracy at substantial gains in computation cost. More specifically, the framework combines the VLAD feature aggregation method with spatial pyramids and PCA for image representation, and proposes the use of Approximate Laplacian Eigenmaps (ALEs) for learning concepts in time linear to the number of images (labeled and unlabeled) available at training. A set of thorough experiments on MIR-Flickr and ImageCLEF 2012 ground truth annotations explore the impact of PCA and pyramids on the attained accuracy, and demonstrate that the proposed framework achieves virtually the same accuracy with a state-of-the-art manifold learning approach, while at the same time offering substantial speedup (in the order of ×80), making possible the completion of a training/testing run for a set of 25k images in less than 3 minutes in a commodity workstation.

1. INTRODUCTION

Despite continuous research efforts in recent years, image annotation remains a challenging problem that is tackled by a variety of computer vision approaches. In particular, the increasing availability of online content has given rise to a host of approaches based on semi-supervised learning that can benefit from the inclusion of unlabeled samples in the training process. More specifically, *manifold learning* approaches rely on the assumption that there is an underlying image manifold, wherein semantically similar images are placed close to each other and semantically dissimilar images are positioned far from each other. Typically, manifold learning is implemented by means of constructing a similarity graph between labeled and unlabeled images and leveraging the graph to estimate the labels of the unlabeled images by considering the labels of neighboring labeled images.

In such settings, a sparse similarity graph is built to encode the visual similarities between images and the graph Laplacian is computed to extract a new learning representation. However, the methods using this technique, have the drawback of being highly dependent on the choice of the neighborhood size in the graph and on the manipulation of a $n \times n$ Laplacian matrix; the manipulation of such large matrices is computationally costly, which is impractical for large datasets. A solution to this problem, proposed by [1], incorporates the aggregation of the eigenvectors and the reduction of the associated complexity, by taking the limit as the number of points go to infinity. We will refer to this technique as Approximate Laplacian Eigenmaps (ALE). In this paper, we construct features, which are easy to manipulate especially in large scale problems, by combining the ALE with the VLAD feature aggregation, PCA and spatial pyramids. As a result, we built a competitive approach both in efficiency and time complexity. We also present a thorough experimental study, for validating our approach. Through extensive experiments, we explore the trade-off between feature size and accuracy; a high competitive accuracy is achieved, even with a low feature size. The evaluation of the proposed methodology demonstrates an exceptional gain in speed and scalability, while the performance decreases only marginally.

2. RELATED WORK

In this section we present the state-of-the art in the area of encoding methods, spatial histograms and manifold learning, which are the main approaches this paper is based on.

Vector of Locally Aggregating Descriptors: In [2], the authors propose a simplified non-probabilistic version of the Fisher Vector for feature aggregation, the so-called Vector of Locally Aggregated Descriptors (VLAD). VLAD uses a codebook μ_K computed using k-means. By applying nearest neighbor (NN) search, each local descriptor x_t is associated with its nearest centroid. Then, the differences between the descriptors $\mathbf{x}_t$ and the centroid μ_i are accumulated to u_i. Finally, the d-dimensional vector of each feature is concatenated with u_i to construct the Kd dimensional VLAD vector.

Spatial Binning: Spatial pyramids were first introduced by Lazebnik et. al to take into account the weak geometry of the Bag-of-Words (BoW) representation by utilizing spatial histograms [3]. An image is repeatedly partitioned and histograms of local features are computed by pooling descriptor-level statistics. The pooling step aggregates the mapped descriptors into a single signature. There are two proposed pooling methods: Sum and Max pooling, with Sum pooling being the most commonly used strategy [4]. In [4], the combination of Gausian Mixture Model (GMM) coding and Fisher Vector (FV) encoding with Sum pooling give satisfactory results. Recent experiments show that Max pooling combined with sparse coding, an alternative approach of computing the coding vector, is a very competitive combination [5].

Manifold Learning: The main idea of manifold learning is to treat the class label of unlabeled images as a missing variable that is derived based on an assumed manifold between labeled and unlabeled images. Specifically, a sparse similarity graph is built in order to identify the most similar images and to compute the graph Laplacian [6]. Laplacian Eigenmaps (LE) is a recent non-linear dimensionality reduction technique that aims to preserve the local structure of the data, thus accurately encoding the underlying graph manifold. A recent approach based on LEs is the Graph Structure Features (GSF) [7], which is used as a state-of-the art competitor in the experimental section.

Unfortunately, the LE-based methods have to construct similarity matrices and subsequently the graph Laplacians, which are quadratic to the size of labeled and unlabeled data. Thus, as the size of data increases, the use of manifold learning becomes prohibitively expensive. Many methods have been proposed to efficiently calculate the graph Laplacian. Most of them are based on building a smaller graph by randomly subsampling a subset of the points [8]. The drawback of these methods is that their output can change dramatically according to the sampling points. Recently, the ALE approach was presented that reduces complexity by using the convergence of the eigenvectors of the normalized graph Laplacian to eigenfunctions [1]. Our proposed framework builds upon this approach.

3. PROPOSED FRAMEWORK

SMaL is a **S**calable **Ma**nifold **L**earning framework on top of ALE, which is linear to the number of images, making possible to use the graph Laplacian in large-scale problems. SMaL makes use of VLAD vectors of different descriptors, and reduces their dimensionality by PCA. Furthermore, the framework computes the spatial pyramids of each image to increase the performance while making features amenable to dimensionality reduction. An overview of the framework is illustrated in Figure 1, including different variants using both global and local descriptors, and imposing different order in the application of PCA, pyramids and Sum pooling.

As mentioned in [7] in order to construct the LE we need to build a $n \times n$ similarity matrix between labeled and unlabeled images. A matrix like this is very costly to compute in large collections. In SMaL, we tackle this problem based on an approximate computation of LEs by estimating a smaller covariance matrix, as suggested in [1], where it is hypothesized that the data $x_i \in \Re^d$ are samples from a distribution $p(x)$. Rotating the data to be as independent as possible, $s = Rx$, can result in a $B \times B$ histogram of bins that approximates the density $p(s)$ of the rotated data. Then, instead of computing the eigenvectors of the similarity matrix between the original data ($n \times n$), one can define eigenfunctions g corresponding to the eigenvalues of the rotated data s ($B \times B$), which can be seen as approximations of the LEs of the original data when $n \to \infty$. This is considerably faster, since typically $B \ll n$. These are recovered by solving the following equation:

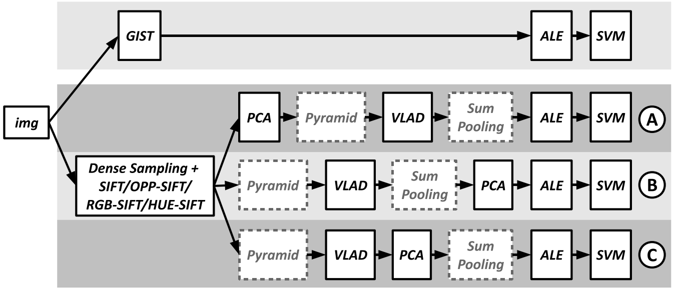

Fig. 1. Overview of SMaL framework. The dashed lines denote optional processing steps.

$$\left(\tilde{D} - P\tilde{W}P\right) g = \sigma P\hat{D}g \qquad (1)$$

where $\tilde{W}$ is the affinity between the B discrete points, P is a diagonal matrix whose diagonal elements give the density at the discrete points, $\tilde{D}$ is a diagonal matrix whose diagonal elements are the sum of the columns of $P\tilde{W}P$, and $\hat{D}$ is a diagonal matrix whose diagonal elements are the sum of the columns of $P\tilde{W}$. An interpolation step follows to the target dimension C_D (described in [1]) and in the end, the $n \times C_D$ approximate LE vectors are derived.

In the final step, a linear classifier is trained using the approximate vectors of the labeled items as input. In our implementation, we opted for the use of linear SVM.

4. EVALUATION

In this section, we evaluate and compare SMaL and GSF as described in Section 3 and in [7] respectively. The GSF is based on the construction of a similarity graph between the images, which is used to obtain the first eigenvectors and manage them as features. We analyze how approximation affects accuracy compared to the execution time.

GSF approximately optimizes the values from the top-*k* [100, 200, 500, 1000, 1500, 2000] NN values, computing the corresponding LE vectors for six different dimensions C_D= [10, 50, 100, 200, 400, 500] repeatedly for every feature and concept, seeking the best parameter set (top-*k*, C_D). In SMaL, no variable needs optimization, since it was observed that different values of B and C_D did not affect accuracy. Thus, we choose to set $B = 50$ and C_D = 500.

The MIR-Flickr (MIRF) [9] image collection was chosen for benchmarking, using two different ground truth annotations. The first one has a set of 24 concepts [9] and the second is the ImageCLEF 2012 (ICLEF12) annotation [10] that has 94 concepts. Both contain annotations for all 25K images of MIRF. For accuracy the mean Interpolated Average Precision (MiAP)[1] measure is used. We have used GIST [11] as global descriptor and SIFT, HUE-SIFT, OpponentSIFT and RGB-SIFT as local descriptors [12]. We use a dense regular grid with spacing of 6 pixels and we perform k-means clustering with typical vocabulary size to 64 centroids as proposed in [2] for better performance. The dataset was divided in three parts: *train*, *validation* and *test*. We set the SVM parameter c =5 in all experiments.

MIRF								
				No PCA	Early PCA		Late PCA	
Descriptor	K	D	#	D	*80*	*40*	*1024*	*512*
GIST	-	480	SMaL	31.8	**35.8**	34.0	-	-
			GSF	**34.7**	34.4	33.7	-	-
SIFT	64	128	SMaL	41.5	41.7	41.5	**45.8**	45.7
			GSF	41.2	41.3	41.8	**45.3**	45.2
HUE-SIFT	64	165	SMaL	30.4	36.3	38.7	**47.3**	47.2
			GSF	46.0	44.0	43.8	**46.2**	46.1
OPP-SIFT	64	384	SMaL	39.3	43.1	44.2	**48.0**	48.0
			GSF	47.5	47.5	47.2	**47.7**	47.6
RGB-SIFT	64	384	SMaL	40.0	44.0	44.0	**48.5**	48.4
			GSF	48.3	**48.4**	47.5	47.8	48.0

ICLEF12								
				No PCA	Early PCA		Late PCA	
Descriptor	K	D	#	D	*80*	*40*	*1024*	*512*
GIST	1	480	SMaL	16.6	**18.0**	17.0	-	-
			GSF	**21.0**	20.1	20.6	-	-
SIFT	64	128	SMaL	20.0	21.6	21.3	24.0	**24.1**
			GSF	22.4	22.2	22.3	**25.0**	**25.0**
HUE-SIFT	64	165	SMaL	15.0	17.7	19.0	**25.6**	**25.6**
			GSF	25.0	24.0	23.7	**25.3**	25.2
OPP-SIFT	64	384	SMaL	20.0	22.5	23.2	25.6	**26.0**
			GSF	26.0	**26.4**	26.2	26.0	26.0
RGB-SIFT	64	384	SMaL	19.5	23.2	23.2	**26.3**	26.2
			GSF	26.3	**26.7**	26.3	26.5	26.4

Table 1. Performance of SMaL and GSF in relation to the position of PCA and to the reduced feature size.

4.1. Dimensionality Reduction in SMaL

Our objective in this experiment was to analyze the impact of dimensionality reduction through Principal Component Analysis (PCA) in two different steps of the processing pipeline.

- the local descriptors are reduced from D to $D' = 80$ and 40 components (Early PCA, A)[2].
- the final VLAD vectors are reduced from $D = Kd$ to D'=1024 and 512 components (Late PCA, B/C)[2].

Table 1 shows the performance comparison between SMaL and GSF. The main observation is that while GSF remains largely unaffected by PCA, SMaL reaps significant benefits by moving from No PCA to Early PCA and even further to Late PCA. Moreover, the choice of D (512, 1024) only marginally affects accuracy; thus, for efficiency reasons, it is recommended to use the more compact features.

4.2. SMaL vs GSF including Spatial Pyramids

Another set of experiments is performed when combining the VLAD vector and spatial pyramids (B^2). In the case of spatial pyramids, VLAD vectors are extracted for each bin. We use Sum pooling and extract 8 VLAD vectors per image: one for the whole image, three for the top, middle and bottom bins and four for each of the four quadrants. We first L2 normalize each of the 8 VLAD vectors independently and after the concatenation we perform the power and L2 normalization. The final VLAD vector after the concatenation of spatial pyramids is reduced from $D = 8Kd$ to D'= 1024 and 512 components using PCA and is L2-normalized before the SVM to make it suitable for use on the linear SVM.

As expected, spatial pyramids with PCA have a positive impact on the accuracy of both approaches. The improvement is about 1-2% for each feature. Table 2 summarizes the performance of two methods in MIRF and ICLEF12. The best performance is on RGB-SIFT when combining spatial pyramids with PCA. For instance, in ICLEF12, the best MiAP for SMaL is 28.1% with PCA at 512, while the best GSF MiAP is 28.6% with PCA at 1024. Also, spatial binning does not help accuracy when HUE-SIFT is used; instead, in some cases the performance decreases by little: for example in SMaL the MiAP with PCA at 512 is 25.6% and with pyramids is 25.1%.

We also have to report that we conducted experiments applying PCA on local descriptors (A^2) or at each bin separately (C^2), but with less improvement compared to the ones of Table 2 (B^2). For example, using the A^2 configuration with PCA to 80-d at SIFT the MiAP was 21.3% compared to 27.4%. In addition, the order of PCA ans Sum pooling (B^2 vs C^2), was found to significantly affect the framework accuracy (27.4% for B vs 20.8% for C).

4.3. Accuracy vs Time Trade-Off

This subsection discusses the issue of balancing between accuracy and time. Table 3 presents the measured execution times for GSF and SMaL versus the attained accuracy. For completing the image annotation, SMaL needs only some minutes, in a commodity workstation with four cores and 12 GB RAM; GSF provides marginally better results in classification, but needs hours for the annotation. A typical execution time for SMaL is 10 mins (60 msec per test image), whereas GSF needs about 200. Thus, SMaL is a very fast approach, practical in large-scale datasets and real-time applications.

Discussion: According to Tables 1 and 2, GSF appears to perform slightly better. In case we do not combine late

[1] Also known as 11-points interpolated average precision. Computed with the vl_pr() method of the VLFeat library, www.vlfeat.org/mdoc/vl_pr.html

[2] A, B & C refer to the three different variants of SMaL in Fig. 1

MIRF							
				No Pyramids		Pyramids	
Descriptor	*K*	*D*	#	*1024*	*512*	*1024*	*512*
SIFT	64	1024	SMaL	45.8	45.7	**48.3**	48.1
			GSF	45.3	45.2	**48.7**	48.4
HUE-SIFT	64	1320	SMaL	**47.3**	47.2	46.4	46.5
			GSF	**46.2**	46.1	46.7	46.0
OPP-SIFT	64	3072	SMaL	48.0	48.0	49.3	**49.4**
			GSF	47.7	47.6	**49.9**	49.5
RGB-SIFT	64	3072	SMaL	48.5	48.4	**50.3**	**50.3**
			GSF	47.8	48.0	**50.8**	50.4

ICLEF12							
				No Pyramids		Pyramids	
Descriptor	*K*	*D*	#	*1024*	*512*	*1024*	*512*
SIFT	64	1024	SMaL	24.0	24.1	26.4	**26.5**
			GSF	25.0	25.0	**27.4**	27.2
HUE-SIFT	64	1320	SMaL	25.6	**25.6**	25.3	25.1
			GSF	25.3	25.2	**25.7**	25.1
OPP-SIFT	64	3072	SMaL	25.6	26.0	27.3	**27.5**
			GSF	26.0	26.0	**27.7**	27.4
RGB-SIFT	64	3072	SMaL	26.3	26.2	27.9	**28.1**
			GSF	26.5	26.4	**28.6**	28.4

Table 2. Performance of SMaL and GSF with and without the use of pyramids at two different dimensions.

PCA with ALE, we cannot achieve very competitive results against GSF. More specifically, in the raw features GSF performs much better than SMaL: when using RGB-SIFT, SMaL achieves 19.5%, while GSF 26.3%. However, when late PCA is applied on VLAD, SMaL achieves competitive results to GSF. Finally, the best performance in SMaL is achieved when an RGB-SIFT VLAD vector is used in tandem with spatial pyramids and PCA at 1024 (Table 2), offering a speedup of ×53.5 compared to GSF. SMaL also performs better in MIRF (in a 50-50 train-test split) when directly compared with Multiple Kernel Learning (MKL) [13], in which 18 visual features are used and the performance metric is the mean Average Precision (mAP). In MKL the mAP is 53%, while SMaL with only one feature, OPP-SIFT in combination with Spatial Pyramids and PCA at 1024 achieves 57.56% mAP.

5. CONCLUSIONS

In this paper, we proposed an approximate semi-supervised learning approach, using VLAD vectors with spatial information. Our representation significantly decreases the time complexity, enabling the use in large-scale settings. Our extensive experiments in MIRF and ICLEF12 show that the proposed framework gives results similar to state-of-the art methods, achieving large computational gains. In the future we plan to investigate the performance on textual features and their fusion with visual. The next challenge is to explore the performance in larger datasets and to use incremental methods, to render their application even easier in real-time scenarios.

Acknowledgements: This work was supported by the SocialSensor project, partially funded by the European Commission, under contract number FP7-287975.

		MiAP		**Times (min.)**		
Descriptor	*Dims*	*SMaL*	*GSF*	*SMaL*	*GSF*	*Speedup*
SIFT	64×128	20.0	22.4	16	154	9.6
SIFT	64×80	21.6	22.2	11	187	17.0
SIFT	64×40	21.3	22.3	7	181	25.9
SIFT	1024	24.0	25.0	7	124	31.0
SIFT	512	24.1	25.0	2.5	137	49.6
HUE-SIFT	64×165	15.0	25.0	19	181	9.5
HUE-SIFT	64×80	17.7	24.0	16	180	11.3
HUE-SIFT	64×40	19.0	23.7	5	152	30.4
HUE-SIFT	1024	25.6	25.3	5	153	30.6
HUE-SIFT	512	25.6	25.2	3	250	83.3
OPP-SIFT	64×365	20.0	26.0	56	360	6.4
OPP-SIFT	64×80	22.5	26.4	10	226	22.6
OPP-SIFT	64×40	23.2	26.2	6	192	32.0
OPP-SIFT	1024	26.0	26.0	4	191	47.8
OPP-SIFT	512	26.0	26.0	3	254	84.7
RGB-SIFT	64×365	19.5	26.3	89	373	4.2
RGB-SIFT	64×80	23.2	26.7	40	206	5.15
RGB-SIFT	64×40	23.2	26.3	7	191	27.3
RGB-SIFT	1024	26.3	26.5	4	214	53.5
RGB-SIFT	512	26.2	26.4	3	255	85.0

Table 3. Accuracy vs Time Trade-Off between SMaL & GSF.

6. REFERENCES

[1] R. Fergus, Y. Weiss, and A. Torralba, "Semi-supervised learning in gigantic image collections," in *Advances in Neural Information Processing Systems 22*, pp. 522–530. 2009.

[2] H. Jégou, M. Douze, C. Schmid, and P. Pérez, "Aggregating local descriptors into a compact image representation," in *IEEE Conf. on, CVPR*, 2010, pp. 3304–3311.

[3] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *IEEE Conf. on CVPR*, 2006, vol. 2, pp. 2169 – 2178.

[4] F. Perronnin, J. Sánchez, and T. Mensink, "Improving the fisher kernel for large-scale image classification," in *ECCV*, 2010.

[5] J. Yang, K. Yu, Y. Gong, and T. Huang, "Linear spatial pyramid matching using sparse coding for image classification," in *IEEE Conf. on CVPR*, 2009.

[6] Y. Bengio, O. Delalleau, N.L. Roux, J.F. Paiement, P. Vincent, and M. Ouimet, "Learning eigenfunctions links spectral embedding and kernel pca," *Neural Computation*, vol. 16, no. 10, pp. 2197–2219, 2004.

[7] S. Papadopoulos, C. Sagonas, I. Kompatsiaris, and A. Vakali, "Semi-supervised concept detection by learning the structure of similarity graphs," in *19th Intern Conf. on MMM*, January 2013.

[8] A. Talwalkar, S. Kumar, and H. Rowley, "Large-scale manifold learning," in *IEEE Conf. on CVPR, 2008.*, pp. 1 –8.

[9] M. J. Huiskes and M. S. Lew, "The mir flickr retrieval evaluation," in *Proceedings of the 2008 ACM MIR '08:*, New York, NY, USA, ACM.

[10] B. Thomee and A. Popescu, "Overview of the clef 2012 flickr photo annotation and retrieval task. in the working notes for the clef 2012 labs and workshop," Italy, 2012.

[11] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *International Journal of Computer Vision*, vol. 42, pp. 145–175, 2001.

[12] K. E. A. van de Sande, T. Gevers, and C. G. M. Snoek, "Empowering visual categorization with the gpu," *IEEE Transactions on Multimedia*, vol. 13, no. 1, pp. 60–70, 2011.

[13] M. Guillaumin, J. Verbeek, and C. Schmid, "Multimodal semi-supervised learning for image classification," in *IEEE Conference on Computer Vision & Pattern Recognition*, 2010, pp. 902 – 909.