VSEM: An open library for visual semantics representation

Elia Bruni
University of Trento
elia.bruni@unitn.it

Ulisse Bordignon
University of Trento
ulisse.bordignon@unitn.it

Adam Liska
University of Trento
adam.liska@unitn.it

Jasper Uijlings
University of Trento
jrr@disi.unitn.it

Irina Sergienya
University of Trento
irina.sergienya@unitn.it

Abstract

VSEM is an open library for visual semantics. Starting from a collection of tagged images, it is possible to automatically construct an image-based representation of concepts by using off-the-shelf VSEM functionalities. VSEM is entirely written in MATLAB and its object-oriented design allows a large flexibility and reusability. The software is accompanied by a website with supporting documentation and examples.

1 Introduction

In the last years we have witnessed great progress in the area of automated image analysis. Important advances, such as the introduction of local features for a robust description of the image content (see Mikolajczyk et al. (2005) for a systematic review) and the bag-of-visual-words method (**BoVW**)[1] for a standard representation across multiple images (Sivic and Zisserman, 2003), have contributed to make image analysis ubiquitous, with applications ranging from robotics to biology, from medicine to photography.

Two facts have played a key role in the rapid advance of these ideas. First, the introduction of very well defined challenges which have been attracting also a wide community of "outsiders" specialized in a variety of disciplines (e.g., machine learning, neural networks, graphical models and natural language processing). Second, the sharing of effective, well documented implementations of cutting edge image analysis algorithms, such as OpenCV[2] and VLFeat.[3]

A comparable story can be told about automatic text analysis. The last decades have seen a long series of successes in the processing of large text corpora in order to extract more or less structured semantic knowledge. In particular, under the assumption that meaning can be captured by patterns of co-occurrences of words, distributional semantic models such as Latent Semantic Analysis (Landauer and Dumais, 1997) or Topic Models (Blei et al., 2003) have been shown to be very effective both in general semantic tasks such as approximating human intuitions about meaning, as well as in more application-driven tasks such as information retrieval, word disambiguation and query expansion (Turney and Pantel, 2010). And also in the case of automated text analysis, a wide range of method implementations are at the disposal of the scientific community.[4]

Nowadays, given the parallel success of the two disciplines, there is growing interest in making the visual and textual channels interact for mutual benefit. If we look at the image analysis community, we discover a well established tradition of studies that exploit both channels of information. For example, there is a relatively extended amount of literature about enhancing the performance on visual tasks such as object recognition or image retrieval by replacing a purely image-based pipeline with hybrid methods augmented with textual information (Barnard et al., 2003; Farhadi et al., 2009; Berg et al., 2010; Kulkarni et al., 2011).

Unfortunately, the same cannot be said of the exploitation of image analysis from within the text community. Despite the huge potential that automatically induced visual features could represent as a new source of perceptually grounded

[1] Bag-of-visual-words model is a popular technique for image classification inspired by the traditional bag-of-words model in Information Retrieval. It represents an image with discrete image-describing features. **Visual words** are identified by clustering a large corpus of lower-level continuous features.

[2] http://opencv.org/

[3] http://www.vlfeat.org/

[4] See for example the annotated list of corpus-based computational linguistics resources at http://www-nlp.stanford.edu/links/statnlp.html.

Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, pages 187–192,
Sofia, Bulgaria, August 4-9 2013.

semantic knowledge,[5] image-enhanced models of semantics developed so far (Feng and Lapata, 2010; Bruni et al., 2011; Leong and Mihalcea, 2011; Bergsma and Goebel, 2011; Bruni et al., 2012a; Bruni et al., 2012b) have only scratched this great potential and are still considered as proof-of-concept studies only.

One possible reason of this delay with respect to the image analysis community might be ascribed to the high entry barriers that NLP researchers adopting image analysis methods have to face. Although many of the image analysis toolkits are open source and well documented, they mainly address users within the same community and therefore their use is not as intuitive for others. The final goal of libraries such VLFeat and OpenCV is the representation and classification of images. Therefore, they naturally lack of a series of complementary functionalities that are necessary to bring the visual representation to the level of semantic concepts.[6]

To fill the gap we just described, we present hereby VSEM,[7] a novel toolkit which allows the extraction of image-based representations of concepts in an easy fashion. VSEM is equipped with state-of-the-art algorithms, from low-level feature detection and description up to the BoVW representation of images, together with a set of new routines necessary to move from an image-wise to a concept-wise representation of image content. In a nutshell, VSEM extracts visual information in a way that resembles how it is done for automatic text analysis. Thanks to BoVW, the image content is indeed discretized and visual units somehow comparable to words in text are produced (the visual words). In this way, from a corpus of images annotated with a set of concepts, it is possible to derive semantic vectors of co-occurrence counts of concepts and visual words akin to the representations of words in terms of textual collocates in standard distributional semantics. Importantly, the obtained visual semantic vectors can be easily combined with more traditional text-based vectors to arrive at a multimodal representation of meaning (see e.g. (Bruni et al., 2011)). It has been shown that the resulting multimodal models perform better than text-only models in semantic tasks such as approximating semantic similarity and relatedness ((Feng and Lapata, 2010; Bruni et al., 2012b)).

VSEM functionalities concerning image analysis is based on VLFeat (Vedaldi and Fulkerson, 2010). This guarantees that the image analysis underpinnings of the library are well maintained and state-of-the-art.

The rest of the paper is organized as follows. In Section 2 we introduce the procedure to obtain an image-based representation of a concept. Section 3 describes the VSEM architecture. Section 4 shows how to install and run VSEM through an example that uses the Pascal VOC data set. Section 5 concludes summarizing the material and discussing further directions.

2 Background

As shown by Feng and Lapata (2010), Bruni et al. (2011) and Leong and Mihalcea (2011), it is possible to construct an image-based representation of a set of target concepts by starting from a collection of images depicting those concepts, encoding the image contents into low-level features (e.g., SIFT) and scaling up to a higher level representation, based on the well-established BoVW method to represent images. In addition, as shown by Bruni et al. (2012b), better representations can be extracted if the object depicting the concept is first localized in the image.

More in detail, the pipeline encapsulating the whole process mentioned above takes as input a collection of images together with their associated tags and optionally object location annotations. Its output is a set of concept representation vectors for individual tags. The following steps are involved: (i) extraction of local image features, (ii) visual vocabulary construction, (iii) encoding the local features in a BoVW histogram, (iv) including spatial information with spatial binning, (v) aggregation of visual words on a per-concept basis in order to obtain the co-occurrence counts for each concept and (vi) transforming the counts into association scores and/or reducing the dimensionality of the data. A brief description of the individual

[5] In recent years, a conspicuous literature of studies has surfaced, wherein demonstration was made of how text based models are not sufficiently good at capturing the environment we acquire language from. This is due to the fact that they are lacking of perceptual information (Andrews et al., 2009; Baroni et al., 2010; Baroni and Lenci, 2008; Riordan and Jones, 2011).

[6] The authors of the aforementioned studies usually refer to words instead of concepts. We chose to call them concepts to account for the both theoretical and practical differences standing between a word and the perceptual information it brings along, which we define its concept.

[7] http://clic.cimec.unitn.it/vsem/

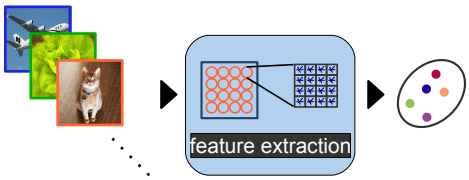

Figure 1: An example of a visual vocabulary creation pipeline. From a set of images, a larger set of features are extracted and clustered, forming the visual vocabulary.

steps follows.

Local features Local features are designed to find local image structures in a repeatable fashion and to represent them in robust ways that are invariant to typical image transformations, such as translation, rotation, scaling, and affine deformation. Local features constitute the basis of approaches developed to automatically recognize specific objects (Grauman and Leibe, 2011). The most popular local feature extraction method is the Scale Invariant Feature Transform (SIFT), introduced by Lowe (2004). VSEM uses the VLFeat implementation of SIFT.

Visual vocabulary To obtain a BoVW representation of the image content, a large set of local features extracted from a large corpus of images are clustered. In this way the local feature space is divided into informative regions (*visual words*) and the collection of the obtained visual words is called *visual vocabulary*. *k*-means is the most commonly used clustering algorithm (Grauman and Leibe, 2011). In the special case of Fisher encoding (see below), the clustering of the features is performed with a *Gaussian mixture model* (GMM), see Perronnin et al. (2010). Figure 1 exemplifies a visual vocabulary construction pipeline. VSEM contains both the *k*-means and the GMM implementations.

Encoding The encoding step maps the local features extracted from an image to the corresponding visual words of the previously created vocabulary. The most common encoding strategy is called *hard quantization*, which assigns each feature to the nearest visual word's centroid (in Euclidean distance). Recently, more effective encoding methods have been introduced, among which the Fisher encoding (Perronnin et al., 2010) has been shown to outperform all the others (Chatfield et al., 2011). VSEM uses both the hard quantization and the Fisher encoding.

Spatial binning A consolidated way of introducing spatial information in BoVW is the use of spatial histograms (Lazebnik et al., 2006). The main idea is to divide the image into several (spatial) regions, compute the encoding for each region and stack the resulting histograms. This technique is referred to as *spatial binning* and it is implemented in VSEM. Figure 2 exemplifies the BoVW pipeline for a single image, involving local features extraction, encoding and spatial binning.

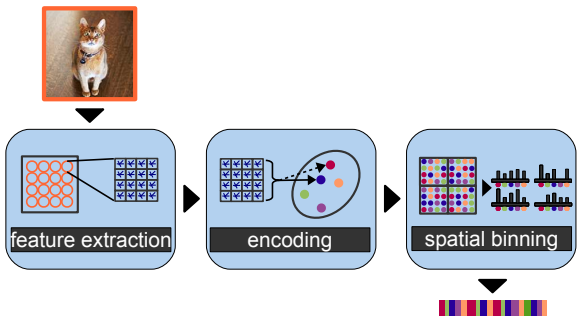

Figure 2: An example of a BoVW representation pipeline for an image. Figure inspired by Chatfield et al. (2011). Each feature extracted from the target image is assigned to the corresponding visual word(s). Then, spatial binning is performed.

Moreover, the input of spatial binning can be further refined by introducing localization. Three different types of localization are typically used: global, object, and surrounding. Global extracts visual information from the whole image and it is also the default option when the localization information is missing. Object extracts visual information from the object location only and the surrounding extracts visual information from outside the object location. Localization itself can either be done by humans (or ground truth annotation) but also by existing localization methods (Uijlings et al., 2013).

For localization, VSEM uses annotated object locations (in the format of bounding boxes) of the target object.

Aggregation Since each concept is represented by multiple images, an aggregation function for pooling the visual word occurrences across images has to be defined. As far as we know, the sum function has been the only function utilized so far. An example for the aggregation step is sketched in

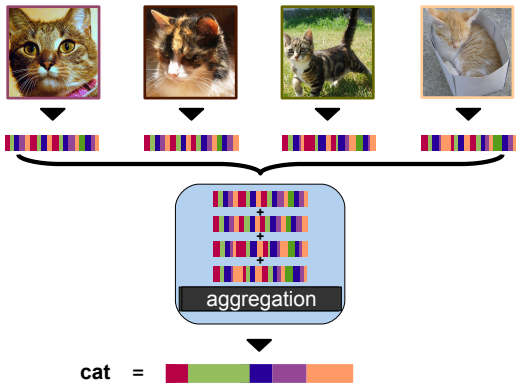

Figure 3: An example of a concept representation pipeline for **cat**. First, several images depicting a cat are represented as vectors of visual word counts and, second, the vectors are aggregated into one single concept vector.

figure 3. VSEM offers an implementation of the sum function.

Transformations Once the concept-representing visual vectors are built, two types of transformation can be performed over them to refine their raw visual word counts: *association scores* and *dimensionality reduction*. So far, the vectors that we have obtained represent co-occurrence counts of visual words with concepts. The goal of association scores is to distinguish interesting co-occurrences from those that are due to chance. In order to do this, VSEM implements two versions of mutual information (pointwise and local), see Evert (2005).

On the other hand, dimensionality reduction leads to matrices that are smaller and easier to work with. Moreover, some techniques are able to smooth the matrices and uncover latent dimensions. Common dimensionality reduction methods are singular value decomposition (Manning et al., 2008), non-negative matrix factorization (Lee and Seung, 2001) and neural networks (Hinton and Salakhutdinov, 2006). VSEM implements the singular value decomposition method.

3 Framework design

VSEM offers a friendly implementation of the pipeline described in Section 2. The framework is organized into five parts, which correspond to an equal number of MATLAB packages and it is written in object-oriented programming to encourage reusability. A description of the packages follows.

- `datasets` This package contains the code that manages the image data sets. We already provide a generic wrapper for several possible dataset formats (`VsemDataset`). Therefore, to use a new image data set two solutions are possible: either write a new class which extends `GenericDataset` or use directly `VsemDataset` after having rearranged the new data as described in **`help`** `VsemDataset`.

- `vision` This package contains the code for extracting the bag-of-visual-words representation of images. In the majority of cases, it can be used as a "black box" by the user. Nevertheless, if the user wants to add new functionalities such as new features or encodings, this is possible by simply extending the corresponding generic classes and the class `VsemHistogramExtractor`.

- `concepts` This is the package that deals with the construction of the image-based representation of concepts. `concepts` is the most important package of VSEM. It applies the image analysis methods to obtain the BoVW representation of the image data and then aggregates visual word counts concept-wise. The main class of this package is `ConceptSpace`, which takes care of storing concepts names and vectors and provides managing and transformation utilities as its methods.

- `benchmarks` VSEM offers a benchmarking suite to assess the quality of the visual concept representations. For example, it can be used to find the optimal parametrization of the visual pipeline.

- `helpers` This package contains supporting classes. There is a general `helpers` with functionalities shared across packages and several package specific `helpers`.

4 Getting started

Installation VSEM can be easily installed by running the file `vsemSetup.m`. Moreover, pascalDatasetSetup.m can be run to download and place the popular dataset, integrating it in the current pipeline.

Documentation All the MATLAB commands of VSEM are self documented (e.g. `help vsem`) and an HTML version of the MATLAB command documentation is available from the VSEM website.

The Pascal VOC demo The Pascal VOC demo provides a comprehensive example of the workings of VSEM. From the demo file `pascalVQDemo.m`multiple configurations are accessible. Additional settings are available and documented for each function, class or package in the toolbox (see Documentation).

Running the demo file executes the following lines of code and returns as output `ConceptSpace`, which contains the visual concept representations for the Pascal data set.

```
% Create a matlab structure with the
% whole set of images in the Pascal
% dataset along with their annotation
dataset = datasets.VsemDataset(
    configuration.imagesPath,'
    annotationFolder',configuration.
    annotationPath);

% Initiate the class that handles
% the extraction of visual features.
featureExtractor = vision.features.
    PhowFeatureExtractor();

% Create the visual vocabulary
vocabulary = KmeansVocabulary.
    trainVocabulary(dataset,
    featureExtractor);

% Calculate semantic vectors
conceptSpace = conceptExtractor.
    extractConcepts(dataset,
    histogramExtractor);

% Compute pointwise mutual
% information
conceptSpace = conceptSpace.reweight();

% Conclude the demo, computing
% the similarity of correlation
% measures of the 190 possible
% pair of concepts from the Pascal
% dataset against a gold standard
[correlationScore, p-value] =
    similarityBenchmark.computeBenchmark
    (conceptSpace,similarityExtractor);
```

5 Conclusions

We have introduced VSEM, an open library for visual semantics. With VSEM it is possible to extract visual semantic information from tagged images and arrange such information into concept representations according to the tenets of distributional semantics, as applied to images instead of text. To analyze images, it uses state-of-the-art techniques such as the SIFT features and the bag-of-visual-words with spatial pyramid and Fisher encoding. In the future, we would like to add automatic localization strategies, new aggregation functions and a completely new package for fusing image- and text-based representations.

References

Mark Andrews, Gabriella Vigliocco, and David Vinson. 2009. Integrating experiential and distributional data to learn semantic representations. *Psychological Review*, 116(3):463–498.

Kobus Barnard, Pinar Duygulu, David Forsyth, Nando de Freitas, David Blei, and Michael Jordan. 2003. Matching words and pictures. *Journal of Machine Learning Research*, 3:1107–1135.

Marco Baroni and Alessandro Lenci. 2008. Concepts and properties in word spaces. *Italian Journal of Linguistics*, 20(1):55–88.

Marco Baroni, Eduard Barbu, Brian Murphy, and Massimo Poesio. 2010. Strudel: A distributional semantic model based on properties and types. *Cognitive Science*, 34(2):222–254.

Tamara Berg, Alexander Berg, and Jonathan Shih. 2010. Automatic attribute discovery and characterization from noisy Web data. In *ECCV*, pages 663–676, Crete, Greece.

Shane Bergsma and Randy Goebel. 2011. Using visual information to predict lexical preference. In *Proceedings of RANLP*, pages 399–405, Hissar, Bulgaria.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022.

Elia Bruni, Giang Binh Tran, and Marco Baroni. 2011. Distributional semantics from text and images. In *Proceedings of the EMNLP GEMS Workshop*, pages 22–32, Edinburgh, UK.

Elia Bruni, Gemma Boleda, Marco Baroni, and Nam Khanh Tran. 2012a. Distributional semantics in Technicolor. In *Proceedings of ACL*, pages 136–145, Jeju Island, Korea.

Elia Bruni, Jasper Uijlings, Marco Baroni, and Nicu Sebe. 2012b. Distributional semantics with eyes: Using image analysis to improve computational representations of word meaning. In *Proceedings of ACM Multimedia*, pages 1219–1228, Nara, Japan.

Ken Chatfield, Victor Lempitsky, Andrea Vedaldi, and Andrew Zisserman. 2011. The devil is in the details: an evaluation of recent feature encoding methods. In *Proceedings of BMVC*, Dundee, UK.

Stefan Evert. 2005. *The Statistics of Word Cooccurrences*. Dissertation, Stuttgart University.

Ali Farhadi, Ian Endres, Derek Hoiem, and David Forsyth. 2009. Describing objects by their attributes. In *Proceedings of CVPR*, pages 1778–1785, Miami Beach, FL.

Yansong Feng and Mirella Lapata. 2010. Visual information in semantic representation. In *Proceedings of HLT-NAACL*, pages 91–99, Los Angeles, CA.

Kristen Grauman and Bastian Leibe. 2011. *Visual Object Recognition*. Morgan & Claypool, San Francisco.

Geoffrey Hinton and Ruslan Salakhutdinov. 2006. Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504 – 507.

Girish Kulkarni, Visruth Premraj, Sagnik Dhar, Siming Li, Yejin Choi, Alexander C. Berg, and Tamara L. Berg. 2011. Baby talk: Understanding and generating simple image descriptions. In *Proceedings of CVPR*, Colorado Springs, MSA.

Thomas Landauer and Susan Dumais. 1997. A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological Review*, 104(2):211–240.

Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce. 2006. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *Proceedings of CVPR*, pages 2169–2178, Washington, DC.

Daniel D. Lee and H. Sebastian Seung. 2001. Algorithms for non-negative matrix factorization. In *In NIPS*, pages 556–562. MIT Press.

Chee Wee Leong and Rada Mihalcea. 2011. Going beyond text: A hybrid image-text approach for measuring word relatedness. In *Proceedings of IJCNLP*, pages 1403–1407.

David Lowe. 2004. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2), November.

Chris Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to Information Retrieval*. Cambridge University Press, Cambridge, UK.

K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. V. Gool. 2005. A Comparison of Affine Region Detectors. *International Journal of Computer Vision*, 65(1).

Florent Perronnin, Jorge Sanchez, and Thomas Mensink. 2010. Improving the fisher kernel for large-scale image classification. In *Proceedings of ECCV*, pages 143–156, Berlin, Heidelberg.

Brian Riordan and Michael Jones. 2011. Redundancy in perceptual and linguistic experience: Comparing feature-based and distributional models of semantic representation. *Topics in Cognitive Science*, 3(2):1–43.

Josef Sivic and Andrew Zisserman. 2003. Video Google: A text retrieval approach to object matching in videos. In *Proceedings of ICCV*, pages 1470–1477, Nice, France.

Peter Turney and Patrick Pantel. 2010. From frequency to meaning: Vector space models of semantics. *Journal of Artificial Intelligence Research*, 37:141–188.

J.R.R. Uijlings, K.E.A. van de Sande, T. Gevers, and A.W.M. Smeulders. 2013. Selective search for object recognition. *IJCV*.

Andrea Vedaldi and Brian Fulkerson. 2010. Vlfeat – an open and portable library of computer vision algorithms. In *Proceedings of ACM Multimedia*, pages 1469–1472, Firenze, Italy.