Technical Report

TR-FSU-INF-CV-2014-01

Seeing through bag-of-visual-word glasses: towards understanding quantization effects in feature extraction methods

Alexander Freytag, Johannes Rühle, Paul Bodesheim, Erik Rodner, and Joachim Denzler

Computer Vision Group, Friedrich Schiller University Jena, Germany
http://www.inf-cv.uni-jena.de

Abstract. Vector-quantized local features frequently used in bag-of-visual-words approaches are the backbone of popular visual recognition systems due to both their simplicity and their performance. Despite their success, bag-of-words-histograms basically contain low-level image statistics (e.g., number of edges of different orientations). The question remains how much visual information is lost in quantization when mapping visual features to code words? To answer this question, we present an in-depth analysis of the effect of local feature quantization on human recognition performance. Our analysis is based on recovering the visual information by inverting quantized local features and presenting these visualizations with different codebook sizes to human observers. Although feature inversion techniques are around for quite a while, to the best of our knowledge, our technique is the first visualizing especially the effect of feature quantization. Thereby, we are now able to compare single steps in common image classification pipelines to human counterparts[1].

1 Introduction

Traditionally, standard image classification systems follow a typical architecture: (1) pre-processing, (2) feature extraction, and (3) training and classification. A significant number of current image categorization methods still follows the bag-of-visual-words (BoW) approach for feature extraction: local features on a dense grid (*e.g.*, SIFT) are extracted and grouped by (un)supervised clustering for codebook creation (*e.g.*, k-Means), which then allows for assigning local features to groups and for forming histograms that can be used as image representations [1,2,3,4,5,6]. The popularity of the BoW strategy is also apparent when looking at the list of Pascal VOC submissions [7], where the majority of recognition systems can be perfectly mapped to the above outlined pipeline.

Clustering of local features has most often been motivated by the analogy of words for text categorization [1]. However, the discovered clusters usually correspond to blob-like objects being semantically poor, and the power of resulting image representations stems from informative statistics rather than from interpretable semantic parts.

[1] An abstract version of this paper was accepted for the ICPR FEAST Workshop.

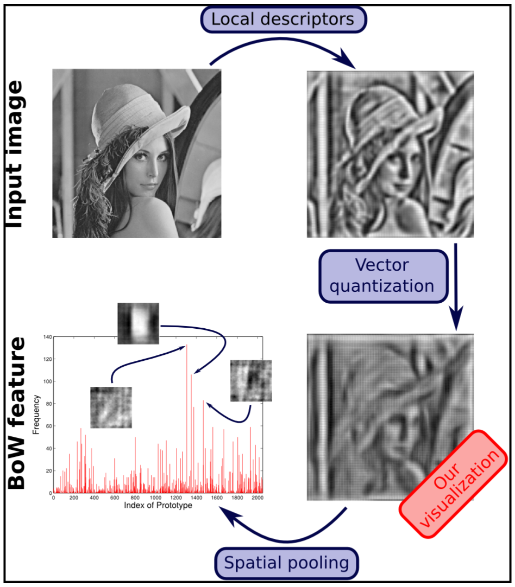


Fig. 1. The long way from an image to its bag-of-visual-words representation. In this paper, we aim at visualizing the loss of information during vector quantization.

In the following paper, we like to see behind the BoW curtain by inspecting how much information is usually lost in vector quantizing local features to precomputed codebooks. We believe that our analysis is valuable for researchers trying to improve BoW models as well as for developers who try to build a good image recognition system. Therefore, we present a new method that inverts quantized local features and visualizes the information loss during this process (see Fig. 1 for a vizualisation thereof). Our inversion method is easy to implement and builds upon the work of [8], where histogram of oriented gradient (HOG) features are inverted to study the visual information loss occurring during feature extraction. In contrast to previous work in this area [9,8], we focus on the effects of vector-quantization within the BoW model and to our knowledge, we are the first qualitatively and quantitatively studying this aspect by asking human observers about estimated image content after inversion.

2 Related work

A brief history of bag-of-visual-words The bag-of-visual-words model goes back to [1], where it was first shown that histograms built on vector quantized local features are highly suitable for object recognition tasks. Forming a single histogram by spatially pooling quantized features over the whole image and discarding any spatial information was regarded as efficient option for obtaining occlusion-invariant features. It was the common understanding at this time that interest point detectors are necessary to select only local features at certain positions which are also suitable for image matching. Later on, [10] showed that this is not the case and that dense and random selection of local features allows for larger BoW descriptors and also for higher recognition rates. The evaluation paper of [11] further studied the influence of detector invariance properties and demonstrated that, for example, invariance with respect to rotations and affine transformations explicitly hurts recognition performance.

The authors of [12] and [2] showed how pyramid matching and simple spatial pooling allows further performance boosting by re-incorporating rough spatial information of local features. The importance of a proper feature encoding with a given codebook was highlighted in [13], where it was shown that the actual choice of cluster method does not influence recognition results significantly and even a random clustering is sufficient. The most important contributions in the area of feature encoding are the work of [14], where soft quantization was first proposed, as well as the paper of [15], where the authors developed an encoding method based on Fisher vectors, and the locality-constrained linear coding (LLC) method of [3]. The key idea of the LLC method is to use only the k nearest neighbors in a codebook for encoding.

Image reconstruction from local features Reconstructing an image from a given feature recently gained attention within our community to better understand learned models. In one of the first works within this area, the authors of [9] propose a technique to reconstruct an image from densely extracted SIFT descriptors. How to invert local binary patterns used for the task of face identification was introduced by [16]. Noteworthy, the authors have not been directly interested in inspecting feature capacity or learned models, but pointed to the problem that local features still contain lots of visual information and thus can be critical from a juristic point of view for face identification systems. A visualization technique for popular HOG features was given in [8] which allowed for insights why object detectors sometimes fire at unexpected positions. The work most similar to the current paper was recently published in [17], which aims at inverting a given bag-of-visual-word histogram by first randomly arranging prototypes and then optimize their positions based on adjacency costs. Since [17] measures inversion quality only in terms of reconstruction error to the original images, it would be interesting to combine their inversion technique and our evaluation method based on asking human observers.

Outline of this paper The remainder of the paper is structured as follows: Our simple yet insightful inversion technique is presented in Sect. 3. We then provide a detailed comparison between machine learning performance and human performance for the task of scene categorization in Sect. 4, and present our webpage for accessing the evaluation server and participating in the large-scale study. A summary of our findings in Sect. 5 conclude the paper.

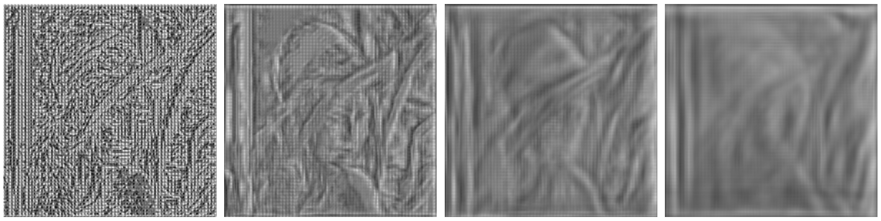


Fig. 2. How does the size of extracted local features influence the visual quality when using our inversion technique? From left to right: patch sizes are 16 px, 32 px, 64 px, and 128 px, respectively.

3 Unbagging bag-of-visual words: visualizing quantization effects

Our technique is simple and in line with current trends for image reconstruction from local features [9,16,8]. For an unseen image, we extract local features on a dense grid and follow the bag-of-words paradigm by quantizing them using a pre-computed codebook. Based on inversion techniques for local features (see [9,16,8] for impressive results), we can compute the most probable image patch for every prototype, *i.e.*, we can visually inspect the quantization quality for a given codebook. Thus, for any local feature x, we vector-quantize it with a codebook and draw the inverted prototype into the reconstruction image with position and size according to the support of the extracted local feature. The complete pipeline of BoW-computation is visualized in Fig. 1. In contrast to previous works for feature inversion, which aim at inspecting the image in the top-right corner, out techniques is designed for inspecting the following step and consequently aims at visualizing quantization effects. It should be noted that for the simplicity of demonstration, we chose HOG-features where inversion was successfully presented in [8] and source code is publicly available. However, our method is not restricted to HOG-features and can be applied to any local feature type where inversion techniques are known for (*e.g.*, [9,16]). Our source code is available at `http://www.inf-cv.uni-jena.de/en/image_representation`.

Effect of local patch sizes With the inversion method at hand, we can investigate the dependencies between several variables during feature extraction and the resulting visual quality. Let us first look at the effect of feature extraction support, *i.e.*, sizes of patches local features are extracted from, on the resulting visual quality when using our inversion technique. In Fig. 2, inversion results are displayed for increasing patch sizes for local feature extraction. For region sizes too small, extracted features can hardly capture any high-level statistics, and thus the resulting inversion looks heavily 'cornered'. On the other hand, for regions too large, extracted local features are highly diverse, and negative aspects of quantization become visible.

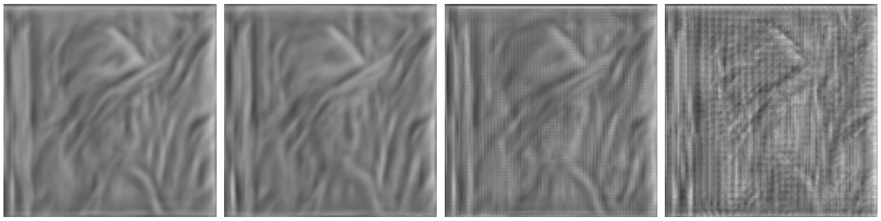


Fig. 3. How does the stride width for local feature extraction influence the visual quality when using our inversion technique? From left to right: stride of 2 px, 4 px, 8 px, and 16 px, respectively.

Effect of patch stride As a second aspect, we investigate the effect of stride width during local feature extraction on visual quality of reconstructed images. For different strides between 2 px and 16 px, reconstruction results are given in Fig. 3. As can be seen, small stride widths tend to average out high-frequency parts, whereas higher strides result in edge artifacts at the boundaries of extracted local patches.

In summary, depending on the local features and inversion method at hand, we can easily inspect dependencies of parameter settings on the visual quality. Thus, we can implicitly estimate which parameter configuration preservers or neglects certain kinds of information present in original images.

4 Experimental evaluation

Our experiments are based on a scene classification task and in particular, we use the 15 Scenes dataset of [18]. This task was chosen, because it is also difficult for human observers due to the moderate number of classes (see human performance on original images in Sect. 4.2) and the fine-grained details that are necessary to distinguish between different scene categories, *e.g.*, street *vs.* highway.

4.1 Machine learning baseline

Local features are extracted from overlapping 64×64 image patches on a dense grid with a stride of 8 pixel and zero padding on image borders. As underlying representations, we choose the commonly used variant of HOG-features as presented in [19]. In general, 4×4 HOG blocks are computed resulting in $D = 512$ dimensional features. Clustering is done using the k-Means implementation of VLFeat [20]. All classes are learned with 100 images during training for the 15 Scenes dataset. Classification is performed using LibLinear [21] and explicit kernel maps [5] to increase model complexity (χ^2-approximation with $n = 3$ and $\gamma = 0.5$ as suggested by [5]). All classification results presented are averaged over 25 random data splits. Regularization parameters are optimized using 10-

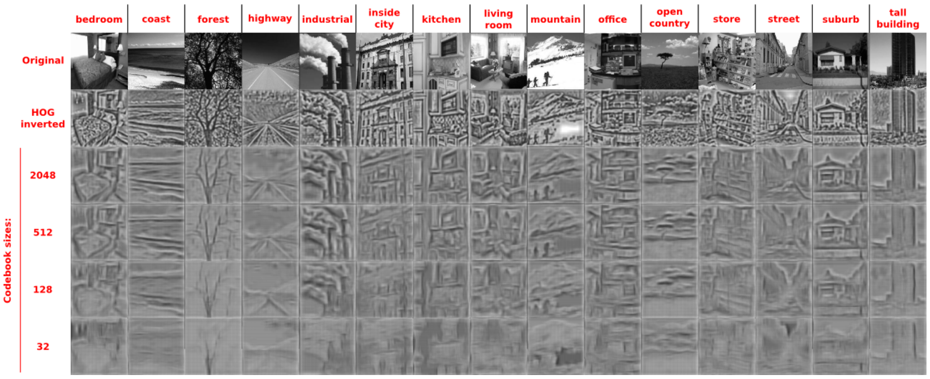


Fig. 4. Overview of the images presented to human observers during the experiment (15 Scenes dataset).

fold cross-validation. For further details, we point to the source code released on the project page[2].

4.2 Experimental setup for human experiments

Since the scene recognition task was unknown to most of our human observers, we showed them example images for each category in the beginning similar to Fig. 6. Afterward, human subjects needed to classify new images and we randomly sampled visualizations with different quantization levels (original image, inverted HOG image without quantization, inverted HOG image with codebook size k; $k \in \{32, 128, 512, 2048\}$). There was no time restriction during the test phase and human observers were allowed to see example images of the categories throughout the whole experiment. In total, we had 20 participants in our study by the time this paper was written, and most of them were colleagues from our group. Note that this can of course not be considered as a representative group of human subjects and results will be definitely biased. However, the conclusions we can draw from this limited amount of data are still interesting and a large-scale study is currently running.

4.3 Evaluation: are we lost in quantization?

The main results of our experiments obtained by humans as well as machine learning techniques are given in Fig. 5. The plot shows the human scene recognition performance measured in terms of the average recognition rate depending on the type of quantization (original image, no quantization, quantization with a specific size of the codebook).

As can be seen, the human recognition performance increases with the number of codebook elements, which is not a surprising fact. However, it is surprising

[2] `http://www.inf-cv.uni-jena.de/en/image_representation`

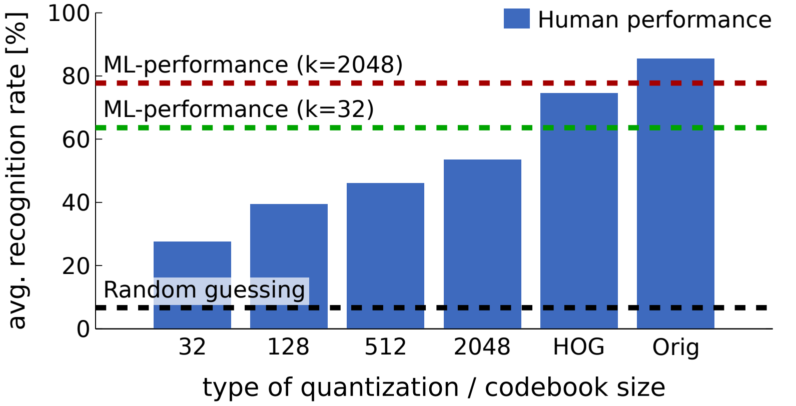


Fig. 5. Human classification result in comparison to machine performance.

that for a codebook size of 32 human performance is significantly worse than machine learning performance (marked with a green line in the plot). This gap becomes smaller when we increase the codebook size but it is still existing for $k = 2{,}048$ and even when no quantization is used at all. Only when the original images are shown to human subjects, the machine learning bag-of-visual words method is not able to beat human performance. It has to be noted here that the small gap between human and machine performance in this case is still surprising given the fact that the machine learning method is not provided with any spatial information.

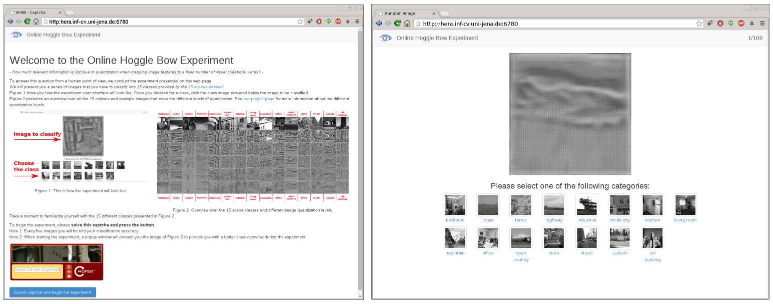
4.4 Web interface to the evaluation server

Our current results are based on only a small set of human subjects. However, we already prepared a large-scale web-based study and a corresponding web interface[3] for our human studies. Some screenshots are displayed in 6. The web interface can be access under `http://hera.inf-cv.uni-jena.de:6780`.

5 Conclusions

In this paper, we analyzed the influence of quantization in the bag-of-visual-words approach on the recognition performance of human observers and compared it to the performance of an automatic visual recognition system. Throughout our analysis, we tried to establish a fair comparison between human and machine performance as much as possible by providing each of them with the same local features. In particular, we inverted quantized local features and presented them to observers in a human study, where the task was to perform scene recognition.

[3] The authors would like to acknowledge Clemens-Alexander Brust for writing an excellent `flask` application for the web interface.



Welcome to the Online Hoggle Bow Experiment

- How much relevant information is lost due to quantization when mapping image features to a fixed number of visual codebook words? -

To answer this question from a human point of view, we conduct the experiment presented on this web page.
We will present you a series of images that you have to classify into 15 classes provided by the 15 scenes dataset.
Figure 1 show you how the experiment user interface will look like. Once you decided for a class, click the class image provided below the image to be classified.
Figure 2 presents an overview over all the 15 classes and example images that show the different levels of quantization. See our project page for more information about the different quantization levels.

Figure 1: This is how the experiment will look like.

Figure 2: Overview over the 15 scene classes and different image quantization levels.

Take a moment to familiarize yourself with the 15 different classes presented in Figure 2.

To begin the experiment, please **solve this captcha and press the button**.
Note 1: Every five images you will be told your classification accuracy.
Note 2: When starting the experiment, a popup window will present you the image of Figure 2 to provide you with a better class overview during the experiment.

Please select one of the following categories:

Fig. 6. Screenshots showing the web experiment. After an introduction (*left image*), the user has to classify the presented inverted image into the given 15 scenes (*right image*).

Our results showed that (i) humans perform significantly worse than machine learning approaches when being restricted to the visual information present in quantized local features rather than having access to the original input images, and (ii) that early stages of low level local feature extraction seem to be most crucial with respect to achieving human performance on original images. Finally, we demonstrated (iii) that large codebook sizes in the order of thousands of prototypes are essential not only for good machine learning performance, but more interestingly, also for human image understanding.

References

1. G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray, “Visual categorization with bags of keypoints,” in *Workshop on statistical learning in computer vision (ECCV-WS)*, vol. 1, 2004, p. 22.
2. S. Lazebnik, C. Schmid, and J. Ponce, “Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories,” in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2006, pp. 2169–2178.
3. J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong, “Locality-constrained linear coding for image classification,” in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010, pp. 3360–3367.
4. A. Kapoor, K. Grauman, R. Urtasun, and T. Darrell, “Gaussian processes for object categorization,” *International Journal of Computer Vision (IJCV)*, vol. 88, pp. 169–188, 2010.
5. A. Vedaldi and A. Zisserman, “Efficient additive kernels via explicit feature maps,” *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 34, no. 3, pp. 480–493, 2012.
6. E. Rodner, A. Freytag, P. Bodesheim, and J. Denzler, “Large-scale gaussian process classification with flexible adaptive histogram kernels,” in *European Conference on Computer Vision (ECCV)*, vol. 4, 2012, pp. 85–98.
7. M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, “The pascal visual object classes (voc) challenge,” *International Journal of Computer Vision (IJCV)*, vol. 88, no. 2, pp. 303–338, Jun. 2010.

8. C. Vondrick, A. Khosla, T. Malisiewicz, and A. Torralba, "Hog-gles: Visualizing object detection features," in *International Conference on Computer Vision (ICCV)*, 2013.
9. P. Weinzaepfel, H. Jégou, and P. Pérez, "Reconstructing an image from its local descriptors," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011, pp. 337–344.
10. E. Nowak, F. Jurie, and B. Triggs, "Sampling strategies for bag-of-features image classification," in *European Conference on Computer Vision (ECCV)*, 2006.
11. J. Zhang, M. Marszałek, S. Lazebnik, and C. Schmid, "Local features and kernels for classification of texture and object categories: A comprehensive study," *International Journal of Computer Vision (IJCV)*, vol. 73, no. 2, pp. 213–238, 2007.
12. K. Grauman and T. Darrell, "The pyramid match kernel: Discriminative classification with sets of image features," in *International Conference on Computer Vision (ICCV)*, 2005, pp. 1458–1465.
13. A. Coates and A. Ng, "The importance of encoding versus training with sparse coding and vector quantization," in *International Conference on Machine Learning (ICML)*, 2011, pp. 921–928.
14. T. Deselaers, D. Keysers, and H. Ney, "Improving a discriminative approach to object recognition using image patches," in *Pattern Recognition – Annual Symposium of the German Association for Pattern Recognition (DAGM)*. Springer, 2005, pp. 326–333.
15. F. Perronnin and C. Dance, "Fisher kernels on visual vocabularies for image categorization," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007, pp. 1–8.
16. E. d'Angelo, L. Jacques, A. Alahi, and P. Vandergheynst, "From bits to images: Inversion of local binary descriptors," *CoRR*, vol. abs/1211.1265, 2012.
17. H. Kato and T. Harada, "Image reconstruction from bag-of-visual-words," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.
18. 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, no. 3, pp. 145–175, 2001.
19. P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part based models," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 32, no. 9, pp. 1627–1645, 2010.
20. A. Vedaldi and B. Fulkerson, "VLFeat: An open and portable library of computer vision algorithms," http://www.vlfeat.org, 2008.
21. R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin., "Liblinear: A library for large linear classification," *Journal of Machine Learning Research (JMLR)*, vol. 9, pp. 1871–1874, 2008.