Low-Cost Visual Feature Representations For Image Retrieval

Ramon F. Pessoa, William Robson Schwartz, Jefersson A. dos Santos
Department of Computer Science
Universidade Federal de Minas Gerais (UFMG)
Belo Horizonte - Minas Gerais, Brazil, 31270-901
Email: {ramon.pessoa, william, jefersson}@dcc.ufmg.br

***Abstract*—This work addressed two research issues in order to investigate and to propose effective solutions for image retrieval on mobile devices: 1) low-cost representation for mobile image search and 2) spatial visual feature extraction. First, we test twenty mid-level representations of binary descriptors, ten color descriptors, five texture descriptors and two shape descriptors in ten datasets, considering the trade-off configuration regarding effectiveness, efficiency, and compactness of visual features. Finally, we propose two approaches of spatial bags of visual words called BOBGrid (spatial Bag Of BIC Grid) and BOBSlic (spatial Bag Of Slic) and compare them with our baselines. In statistical analyzes, BOBGrid and BOBSlic achieved processing results and performed better than our baselines WSA and BOSSANova.**

***Keywords*-Mobile Image Search; Global Descriptors; Binary Descriptors; Bag of Visual Words; Spatial Bag of Visual Words.**

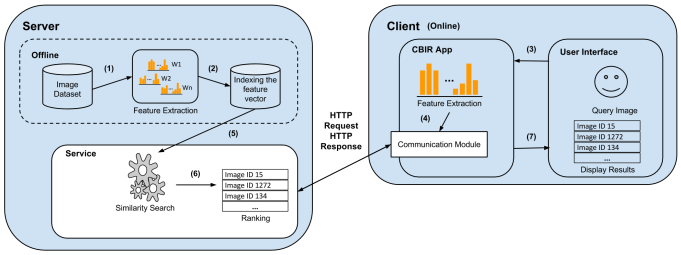

Fig. 1. Mobile visual search architecture used in this work. Low-cost image feature extraction is performed on mobile devices and the search step is processed on the server the server side. Only the query feature vectors are required to be transferred. In image retrieval on mobile devices, we need compact and fast, but also accurate image representations.

I. Introduction

Mobile Visual Search (MVS) is a new research area in Content-Based Image retrieval (CBIR) which provides the services of search and retrieval of visual information specifically for mobile devices. Besides the traditional challenges, such as, translation, rotation and changes in scale and illumination, image processing in mobile devices is limited by many other constraints. For instance, the memory and computing resources may be very limited. Regarding feature extraction from images, those constraints configure a *trade-off* among effectiveness, efficiency and compactness [1]. Therefore, it is important to perform a thorough evaluation regarding such aspects, as has been done in the literature in other domains (web image retrieval, remote sensing image classification, mobile visual recognition and machine learning, mobile augmented reality).

Objectives and Contributions: In this work[1], we investigate feature representation strategies for allowing real-time content-based image retrieval using mobile devices as query interfaces. In this model, the feature extraction step is performed on the device and the search step is processed on the server, as illustrated in Fig. 1. Thus, only the query feature vectors are required to be transferred. In image retrieval on mobile devices, we need compact and fast, but also accurate image representations. We achieve our objective by evaluating and developing robust, fast and efficient feature extraction algorithms that fit mobile device constraints.

The main contributions of this work are the following. 1) A comparative study of binary descriptors using mid-level representation and global descriptors (color, texture, and shape) in an approach of image retrieval on mobile devices. 2) We propose two new bag of visual word representations that include spatial information to improve the quality of image representation on mobile devices, which could be crucial to distinguish types of objects and scenes.

Some results obtained in this work were published in XX Iberoamerican Congress on Pattern Recognition (CIARP 2015) and in XXVIII Conference on Graphics, Patterns and Images (SIBGRAPI Work in Progress 2015). In [1], we performed an extensive study on low-cost representations for image feature extraction on mobile devices and in [3], we performed an experimental comparison of feature extraction and distance metrics for image retrieval.

II. Background and Concepts

In this section, we present some background concepts necessary to understand the approaches we have analyzed and proposed in this work.

A. Feature Extraction

In CBIR, local features are tipically employed with mid-level representation in order to encode the global visual content of the images [4]. Global features can represent an

[1]This work relates to a M.Sc. dissertation [2]. Final master thesis available in http://homepages.dcc.ufmg.br/~ramon.pessoa/master_thesis/20151218-final-ramon-pessoa_master-thesis.pdf

image with only one vector leading to reduced computation cost[2]. Mid-level features are also good alternatives since they provide suitable representation for the amount of local features extraction.

Local Binary Features: As an alternative, low-complexity binary descriptors have recently emerged. There are three main advantages of this kind of descriptor: (1) the time required for extracting, (2) the small size of extracted feature vector and (3) low-cost matching. Therefore, in this work we use binary descriptors instead of non-binary descriptors. In this work, we use five binary descriptors: 1) Binary Robust Independent Elementary Features (BRIEF), 2) Oriented FAST and Rotated BRIEF (ORB), 3) Binary Robust Invariant Scalable Keypoints (BRISK), 4) Fast REtinA Keypoint (FREAK), 5) Boosting binary keypoint descriptors (BinBoost).

Global Feature Descriptors: Global features, which describe an image as a whole, can represent image content efficiently. They provide an overall spatial organization of scale and orientation information of the image. If by one side the literature indicates that, in general, global features are less effective than local ones, by the other side, they are more efficient since they are not dependent on mid-level representation. In this work, we use seventeen global descriptors: ten color descriptors (Auto-Correlogram Color (ACC), Border/Interior Pixel Classification (BIC), Cumulative Global Color Histogram (CGCH), Color Bitmap, Color Structure (CSD), Color Wavelet HSV (CWHSV), Color Wavelet LUV (CWLUV), Global Color Histogram (GCH), Joint Auto-Correlogram (JAC), Local Color Histogram (LCH)), five texture descriptors (Local Activity Spectrum (LAS), Local Binary Pattern (LBP), Quantized Compound Change Histogram (QCCH), Statistical Analysis of Structural Information (SASI), UNSER) and two shape descriptors (Edge Orientation Autocorrelogram (EOAC), Spherical Pyramid Technique (SPYTEC)).

B. Mid-level Representation

The Bag of Visual Words model converts the set of local descriptors into the final image representation vector by a succession of four steps: 1) sampling strategy (selection of regions (patches) into the image), 2) local feature descriptor (non-binary or binary image representation for each patch in the image), 3) coding (assigning each local descriptor to visual words) and 4) pooling (summarizing the local descriptor projections using average or maximum operations, for example). In the following two subsection, we detail each of the four steps.

Sampling Strategies: According to [5], the patch selection can be based on two approaches: (i) using points of interest (sparse sampling): in this case an algorithm is applied to find such a region to be described; or (ii) dense sampling, where fixed-size regions are allocated on a regular grid size. In this thesis we use six sparse sampling: 1) FAST (Features from Accelerated Segment Test), 2) GFTT (Good Features to Track), 3) GFTTHarris (Good Features to Track using Harris), 4) Maximally Stable Extremal Regions (MSER), 5) Oriented FAST and Rotated BRIEF (ORB Detector) and 6) Speeded-Up Robust Features (SURF Detector)[3].

Bag of Visual Words: The mid-level representation is useful to convert a set of local features into an unique global representation for each image, which is called Bag of Visual Words (BoW). This process is divided into offline and online steps. In the offline phase, after local feature descriptor are obtained on the image database, the features are clustered to create the vocabulary of visual words (also known as codebook or dictionary). In the online phase, the same process (dense sampling, local feature descriptor, clustering features into visual words) is done using the codebook as a dictionary to do assignment and pooling steps. Assignment (or coding) is a step that associates the feature vector of a point detected in the image with the visual words in the dictionary. A pooling strategy is used for summarizing/selecting the assignment values from the coding/assignment step, generating the image feature vector [6]. In this work, we have evaluated two word assignment strategies: 1) Hard assignment and 2) Soft assignment. We used two pooling strategies (average or maximum) to summarize the assignment vectors. After the assignment and pooling steps, the Bag of Visual Words final vector representation is created and can be used in visual pattern recognition tasks, such as content-based image retrieval or categorization/classification of images.

C. Image Segmentation

Recently, superpixels have become an essential tool to the vision community. These algorithms group pixels into perceptually meaningful regions, which can be used to replace the rigid structure of the pixel grid. In this work, we use a recent superpixel algorithm called SLIC. Simple Linear Iterative Clustering (SLIC) [7] algorithm simply performs k-means clustering approach in the 5D space of color information (the $CIElab$ color space) and image location to efficiently generate superpixels. In summary, SLIC is an adaptation of k-means for superpixel generation where the search space is dramatically reduced and a weighted distance measure combines color and spatial proximity. The main parameters of SLIC are n (number of approximately equally-sized superpixels) and their compactness (c).

D. Evaluation Metrics

We use three measures in this work to evaluate several algorithms: 1) Mean Average Precision (MAP), 2) Precision at Top N images. As we are studying representations in the CBIR context, achieving a high precision on the initial images retrieved is important. Therefore, most of the time we considered the top 10 images to calculated the $P@N$, and 3) Compression Ratio (CR). CR is defined as the ratio between the uncompressed size and compressed size.

We have used the $P@5$, $P@10$, $P@15$ metric to evaluate **effectiveness**. To give an overall precision, we report

[2]All these local and global descriptors are described in the master thesis [2].

[3]All these sampling strategy are described in the master thesis [2].

the **effectiveness** using the MAP (Mean Average Precision) metric. The **efficiency** was evaluated by computing the feature extraction and representation time, in seconds. Finally, we have used the representation size (in bytes) and the Compression Ratio (CR) as measures for evaluating the **compactness**.

E. Benchmark Datasets

In this work, we use several benchmark image datasets to compare our methods of image retrieval. These benchmarks provide a common ground for researchers to compare their methods. The datasets are splitted in four categories: 1) Scenes, 2) Mobile Visual Search, 3) Single-label, 4) Multi-label. The datasets names are Fifteen Scene Categories (15Scenes), Oxford Buildings (OxBuild11), Paris Landmarks (Paris), Zurich Building (ZuBuD), WANG, Caltech 101, Caltech 256, PASCAL Visual Object Classes 2007 (VOC2007), University of Washington dataset (UWdataset)[4].

III. Low-Cost Representation for Mobile Image Search

In this work, we deal with the *feature extraction triple trade-off problem* (efficiency, effectiveness and compactness) in mobile devices by evaluating low-cost feature representations. We concentrate our efforts in four main fronts: (1) binary low-level descriptor selection; (2) mid-level representation; (3) low-level global representation analysis and (4) feasibility analysis of data compression techniques.

We are interested in balancing computational cost, precison, and feature representation size. In this sense, binary descriptors are considerable options because they provide effective and compact representation. Mid-level representations based on Bag of Visual Words (BoVW, or just BoW) are good alternatives since it provides effective features and compacted in comparison with the amount of local features extracted. Finally, we present global descriptors (color, texture, and shape) analysis as an alternative for mid-level representation, as well as, image features compression techniques.

Analyzes of Low-Cost Representations: We tested twenty mid-level representations of binary descriptors, ten color descriptors, five texture descriptors and two shape descriptors in ten datasets, considering the trade-off configuration regarding effectiveness, efficiency, and compactness of visual features. To learn the codebooks and create mid-level representations, we apply a k-medians clustering algorithm with Hamming distance over all sampled descriptors, as [8]. We created one dictionary for each binary descriptor (BRIEF, BRISK, FREAK, ORB, BinBoost). The parameters for dictionary generation and image representation are the same: dense sampling (6 pixels) as in [9], [8], and 1024 visual words, as in [8]. For all the experiments we use statistical analysis and the results are reported with a confidence of 95% (α=0.05). Fig. 2 shows a summary of all analyzes of low-cost representations for mobile image search done in this master thesis using ten datasets. All analyzes can be found in the master's thesis [2].

[4]All these datasets are described in the master thesis [2].

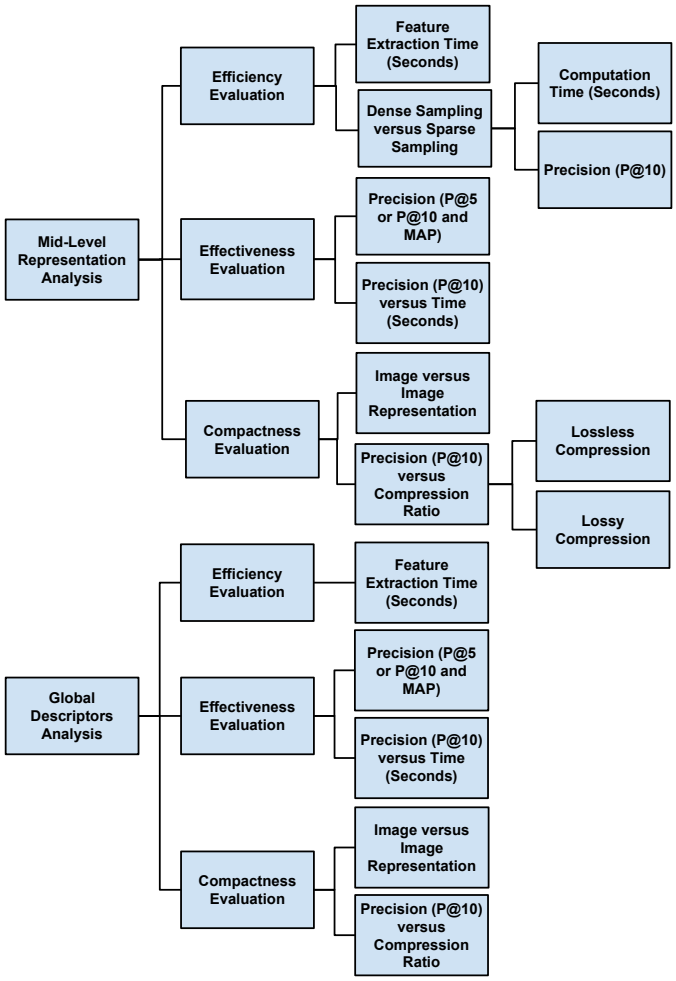

Fig. 2. Analyzes of low-cost representations for mobile image search.

Discussion: Considering the analysis of effectiveness, efficiency and compactness of visual features presented in the master thesis [2], we can draw our conclusions regarding the best global descriptors and bag of words representations to be used in the mobile image search scenario: 1) BIC (Border/Interior Pixel Classification) [10] and 2) DEOBSM (Bag of Words using Dense Sampling, ORB descriptor, Soft assignment and Maximum pooling), respectively. We note that the BIC descriptor seems to outperform DEOBSM in almost cases. Paired statistical tests (Table I) show in which scenary BIC can be considered better then DEOBSM. In Table I, we use P@5 just to SVMS692 and ZuBuD, because this two datasets have just 5 images per class.

Image retrieval experiments on mobile devices: We developed a prototype system using Android platform[5] for retrieving the content of general image by using mobile devices as query interface. We did some preliminary tests in the smartphone LG Nexus 5 using the best descriptors pointed out in this thesis (BIC and Bag of Words using Dense Sampling, ORB descriptor, Soft assignment and Maximum

[5]Android platform - http://developer.android.com/

TABLE I
STATISTICAL TEST PAIRED T-TEST, WITH 95% OF CONFIDENCE, BETWEEN THE BIC DESCRIPTOR AND THE DEOBSM (BAG OF WORDS USING DENSE SAMPLING, ORB DESCRIPTOR, SOFT ASSIGNMENT AND MAXIMUM POOLING) DESCRIPTOR.

Dataset	BIC	DEOBSM	Best, IC(95%)
15Scenes (MAP)	12.99 ± 0.02	$\mathbf{21.76 \pm 0.05}$	DEOBSM
caltech101 (MAP)	6.68 ± 0.02	$\mathbf{10.62 \pm 0.03}$	DEOBSM
caltech256 (MAP)	2.86 ± 0.01	$\mathbf{5.32 \pm 0.01}$	DEOBSM
OxBuild11 (MAP)	20.57 ± 0.06	$\mathbf{33.53 \pm 0.06}$	DEOBSM
Paris (MAP)	$\mathbf{11.65 \pm 0.03}$	11.38 ± 0.03	BIC
SMVS692 (MAP)	24.14 ± 0.01	$\mathbf{35.98 \pm 0.01}$	DEOBSM
UWdataset (MAP)	$\mathbf{36.24 \pm 0.08}$	18.89 ± 0.05	BIC
VOC2007 (MAP)	$\mathbf{25.37 \pm 0.04}$	24.31 ± 0.04	BIC
WANG (MAP)	$\mathbf{51.8 \pm 0.12}$	37.28 ± 0.14	BIC
ZuBuD (MAP)	$\mathbf{78.99 \pm 0.02}$	71.15 ± 0.03	BIC
15Scenes (P@10)	30.2 ± 0.05	$\mathbf{43.88 \pm 0.08}$	DEOBSM
caltech101 (P@10)	20.37 ± 0.03	$\mathbf{26.51 \pm 0.04}$	DEOBSM
caltech256 (P@10)	$\mathbf{15.31 \pm 0.01}$	13.99 ± 0.01	BIC
OxBuild11 (P@10)	28.43 ± 0.1	$\mathbf{46.29 \pm 0.13}$	DEOBSM
Paris (P@10)	32.79 ± 0.07	32.90 ± 0.05	Not Different
SVVM692 (P@5)	23.08 ± 0.01	$\mathbf{34.27 \pm 0.01}$	DEOBSM
UWdataset (P@10)	$\mathbf{59.74 \pm 0.1}$	35.05 ± 0.08	BIC
VOC2007 (P@10)	$\mathbf{21.05 \pm 0.03}$	20.83 ± 0.04	BIC
WANG (P@10)	$\mathbf{77.73 \pm 0.1}$	56.12 ± 0.13	BIC
ZuBuD (P@5)	$\mathbf{72.62 \pm 0.02}$	0.7001 ± 0.03	BIC

pooling = DEOBSM). As a result, for images dimension of 300×500 using *LG Nexus 5*, the feature extraction of BIC takes about 300 milliseconds and DEOBSM takes around 500 milliseconds. Using bag of words with size of 128, the feature extraction takes about 300 milliseconds. It is a difficult task to be less than 300 milliseconds because of the Java Naming and Directory Interface (JNDI) overhead.

IV. SPATIAL FEATURE REPRESENTATION FOR MOBILE IMAGE SEARCH

In the last years, bag-of-visual words representations have been successfully used in many applications of computer vision such as image retrieval and classification. However, the traditional pooling methods usually discard the spatial configuration for visual words in the image and this kind of information is important to distinguish types of object and arrangements in the image. Therefore, the research community has been very active proposing new approaches of bag of visual words to encode the spatial information of visual words to improve image semantics and distinguish different classes of scenes or objects [6].

In section III, we point out the BIC (Border/Interior Pixel Classification) as one of the best descriptors analyzed. Thus, we use this descriptor to create representations of part of images and the vectors representation are used on mid-level strategies. In this work, we have proposed two approaches: (1) BOBGrid (spatial Bag Of BIC Grid) and (2) BOBSlic (spatial Bag Of Slic). We have conducted experiments by comparing the two proposed spatial representations against Word spatial arrangement (WSA) [4] and BossaNova [11].

Word spatial arrangement (WSA): Proposed by [4], [12], the WSA is an approach to represent the spatial arrangement of visual words under the bag-of-visual-words model and it is based on the idea of dividing the image space into quadrants using each point as the origin of the quadrants and counting the number of words that appear in each quadrant. In WSA, first, the image space is divided and each point p_i is detected in the image by a sparse sample, for example. Then, the space is divided into 4 quadrants, putting the point p_i in the quadrant's origin. For every other detected point p_j , WSA increments the counters of the visual word associated with p_j in the position that corresponds to the position of p_j in relation to p_i. For instance, if w_j is the visual word associated with p_j and p_j is at top-left from p_i, the counter for top-left position of w_j is incremented. After all points are analyzed in relation to p_i, the quadrant's origin goes to the next point $p_i + 1$, and the counting in relation to p_i+1 begins. When all points have been the quadrant's origin, the counting finishes and each 4-tuple is normalized by its sum [4].

BossaNova Representation: Althout is not a spatial mid-level representation, the BossaNova representation [11], [13] is an extension of BoW model which provides an improvement in the pooling stage, to preserve a more rich way the information obtained during the encoding step. BossaNova differentiates from the BoW approach at the coding/pooling stage, resulting in a new representation that better preserves the information from the encoded local descriptors by using a density-based pooling step. Their coding function activates the closest codewords to the descriptor, which corresponds to a localized soft coding over the visual codebook. The pooling step estimates the distribution of the descriptors around each codeword, while the BoW estimates the distribution around one or determined number of codewords.

BOBGrid Representation: To encode spatial information on bag-of-visual words representation, we propose to split the image in nine similar quadrants. For each tile, we compute visual features by using the BIC descriptor [10] – Border/Interior Pixel Classification, which were the most suitable descriptor as presented in section III. We encode the spatial information by creating a graph with edges starting from the the center quadrant. In summary, we use a directed graph with eight edges. The BOBGrid is divided into two steps: Offline and Online. In the offline step, we used the nine splited parts of all images on the dataset to create a dictionary (or codebook) using k-means algorithm. On the Online step, we use the dicionary created to generate BOBGrid representations. A dictionary of 128 visual words was constructed selecting points in the feature space to create BOBGrid representations. Fig. 3 (2) presents the process of splitting the image in nine quadrants and the creation of edges generating a graph in the imagem. Fig. 3 presents the offline and online processes to create a codebook based on nine quadrants in the image and the process which uses the dictionary created to generate BOBGrid representations.

BOBSlic Representation: The second spatial algorithm proposed in this work is the BOBSlic representation. BOBSlic use the SLIC algorithm [7] – Simple Linear Iterative Clustering – to separate the image in parts segmented. As shown before, SLIC has two parameters: n = Number of superpixels and their compactness (c). We tested several values

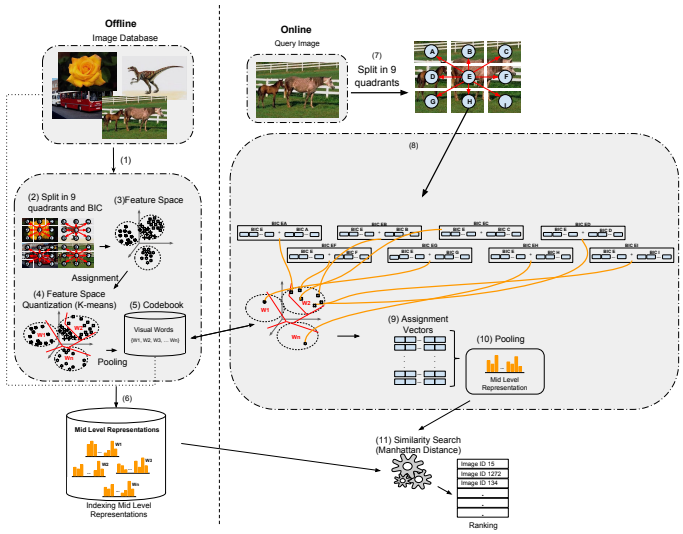

Fig. 3. BoBGrid Representation: In the offline step, eight edges of nine quadrants are clustered and a codebook is created. On the online step, given a query image a BOBGrid representation is created and then assign to the dictionary to create a bag of words representation.

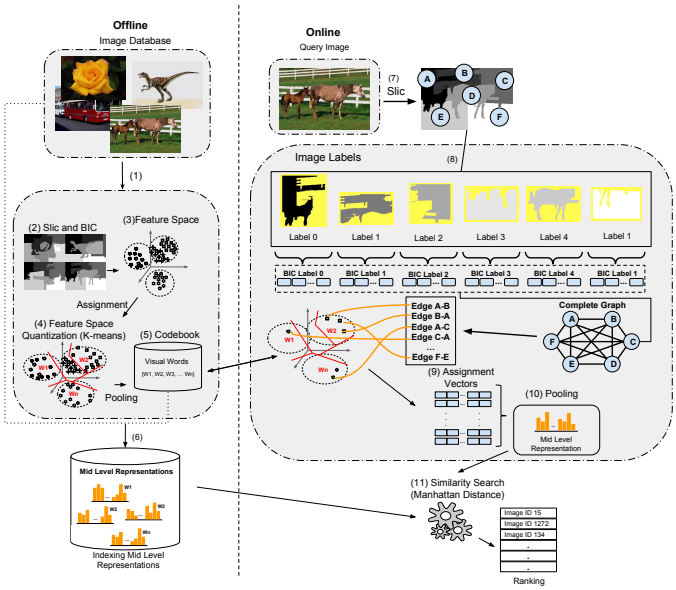

Fig. 4. BOBSlic Representation: In the offline step, the segmented regions created by the superpixel algorithm SLIC are clustered and a codebook is created. On the online step, given a query image, a BOBSlic representation is created and then assign to the dictionary to create a bag of words representation.

for the parameters n (n = 10, 20 or 30) and c (c = 0, 5, 20, 50). The best results were obtained with n = 10 and c = 50. Therefore, we use this setting to segment images with SLIC. After SLIC, we compute BIC descriptors [10] for each segmented part. We have created a complete graph where the edge is a concatenation of two BIC vectors. The graph is undirected, therefore we have two edges round trip (edge = $[BIC_a + BIC_b]$ and edge = $[BIC_b + BIC_a]$). Again, we have offline and online steps where a codebook (dictionary) is generated and used to create spatial bag of words from images on the dataset and the query image. Fig. 4 presents the offline and online processes to create a codebook based on SLIC-BIC in the image and the process which uses the dictionary created to generate BOBSlic representations.

Experiments: To evaluate the proposed approaches considering a CBIR scenario, we have used the WANG dataset. This dataset can be classified as different types of images with scenes (like monuments), object (like buses) and high intra-class variation like (africa). For each category, the same object appears in different rotation and viewpoints.

In our experimental setup, we compare BOBGrid and BOBSlic with our baseline WSA. We also compare with the BossaNova approach. We use the best configuration described on the baselines papers of WSA [4] and BossaNova [11]. For WSA, we have used the standard version (WSA) available on WSA info page [6]. For BossaNova, we have used the version (BossaNova) available on BossaNova info page [7] and modified in the work [8] to be used with binary descriptors.

Results: Table II shows the experimental results in WANG database [14]. Note that the proposed approaches (BOBGrid and BOBSlic) outperformed our baseline WSA, and also BossaNova and the tradicional Bag-of-Words models (BoW). In Table II, DE = Dense, FT = FAST, HS = GFTT using HARRIS, OB = ORB detector / ORB descriptor, BB = BinBoost, BF = BRIEF, BK = BRISK, SA = Soft-AVG, SM = Soft-MAX, HA = Hard-AVG, HM = Hard-MAX, BN = BossaNova, BOBGrid = Bag Of BIC Grid, BOBSlic = Bag Of Slic BIC.

TABLE II
PRECISION RESULTS OF BOBGRID AND BOBSLIC, OUR BASELINE (WSA), BOSSANOVA APPROACHES AND TRADITIONAL BAG OF WORDS (BOW'S) STRATEGIES ON WANG DATASET. DE = DENSE, FT = FAST, HS = GFTT USING HARRIS, OB = ORB DETECTOR / ORB DESCRIPTOR, BB = BINBOOST, BF = BRIEF, BK = BRISK, SA = SOFT-AVG, SM = SOFT-MAX, HA = HARD-AVG, HM = HARD-MAX, BN = BOSSANOVA, BOBGRID = BAG OF BIC GRID, BOBSLIC = BAG OF SLIC BIC.

Approach	P@5(%)	P@10(%)	P@15(%)	MAP
DEBBSA	70.92	64.37	60.35	41.29
DEBBHA	73.32	65.71	61.35	37.93
DEBFSM	62.94	54.65	50.55	34.65
DEOBSM	63.52	56.12	52.51	37.28
FTBKSM	70.74	63.47	59.65	41.63
DEOBBN	66.02	58.67	54.99	38.93
HSOBBN	55.02	46.95	43.11	29.33
OBOBBN	48.18	40.31	37.17	27.43
HSOBHMWSA	27.22	18.60	15.59	11.96
HSOBSMWSA	32.04	23.16	20.01	13.67
OBOBHMWSA	27.08	18.45	15.53	11.90
OBOBSMWSA	29.40	20.54	17.58	17.58
BOBGrid	**77.90**	**71.57**	**67.73**	**48.06**
BOBSlic	**78.20**	**71.43**	**68.07**	**48.81**

It is important to point out that the proposed algorithms provide more compact representations than our baseline WSA and BossaNova, which are suitable for applications in mobile devices. Both vectors of BOBGrid and BOBSlic have length of 128, while WSA has size of $4K$, BossaNova has size of

[6]http://www.recod.ic.unicamp.br/~otavio/pr_wsa/index.htm
[7]https://sites.google.com/site/bossanovasite/

$K \times (B + 1)$, where B is a BossaNova parameter which indicates the local histogram number of bins. Traditional BoW's has feature vector size of K, where K is the dictionary size. For retrieval experiments, which are generally based on computing distances between vectors, with the Euclidean or Manhattan distance, for example, vectors should be compact, or embedded in an index structure, to avoid the curse of the dimensionality [15].

To determine the statistical significance of results, a statistical test for differences between means was done using paired t-test, paired about the classes of the database. In conclusion, BOBGrid and BOBSlic performs statistically better than our baseline WSA with a confidence of 95% on WANG dataset. BOBGrid and BOBSlic also showed greater accuracy in relation to BossaNova on WANG dataset.

V. Conclusion and Future Work

Considering the analysis regarding effectiveness, efficiency and compactness presented, we can draw our conclusions about the best global descriptors and bag of words representations to be used in the mobile image search scenario: 1) BIC (Border/Interior Pixel Classification) [10] and 2) DEOBSM (Bag of Words using Dense Sampling, ORB descriptor, Soft assignment and Maximum pooling), respectively. Paired statistical test in ten datasets showed that BIC can be considered better then DEOBSM. Therefore, for mobile visual retrieval, we may consider use BIC descriptor as the best option considering the triple trade-off problem regarding efficiency, effectiveness and compactness.

We proposed two approaches of extracting spatial information on images to improve the quality of image representation. These approaches are called BOBGrid (Spatial Bag of BIC Grid) and BOBSlic (Spatial Bag of Slic BIC). We compare them against WSA (Visual Word Spatial Arrangement) [6] and BossaNova (Bag Of Statistical Sampling Analysis) [11]. In statistical analyzes, both BOBGrid and BOBSlic outperform our baseline of spatial bag of visual words WSA and the BossaNova algorithm in the WANG dataset. In addition, descriptors are more compact, which make them more suitable for mobile devices applications. Our approaches have vectors of length 128, while the baselines have size 1024. As aforementioned, for retrieval experiments, vectors should be compact to avoid the curse of the dimensionality. The experiments were evaluated using several precision metrics (P@5, P@10, P@15 and MAP) and statistical analysis. The results presented indicate the importance of using image parts and segmentations on images to create more robust bag of words. We could observe that BOBSlic seems to be better than BOBGrid because it uses a segmentation approach to represent bag of words.

As future works, we propose to perform more experimental analysis on mobile devices, to evaluate algorithms of text processing and analyze a multimodal approach using text and image features together to improve the similarity search, to exploit more algorithms which use semantic or spatial information and to propose a method to select the best descriptors to use in an average rank aggregation approach.

Acknowledgment

This work was partially financed by Brazilian National Research Council – CNPq (Grant 477457/2013-4 and 449638/2014-6), Minas Gerais Research Foundation – FAPEMIG (Grants APQ-01806-13, APQ-00567-14 and APQ-00768-14), and Coordination for the Improvement of Higher Education Personnel – CAPES.

References

[1] R. F. Pessoa, W. R. Schwartz, and J. A. dos Santos, "A study on low-cost representations for image feature extraction on mobile devices," in *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications - 20th Iberoamerican Congress, CIARP 2015, Montevideo, Uruguay, November 9-12, 2015, Proceedings*, ser. Lecture Notes in Computer Science. Springer International Publishing, 2015, vol. 9423, pp. 424–431. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-25751-8_51

[2] R. F. Pessoa, "Low-cost visual feature representations for image retrieval," Master's thesis, Universidade Federal de Minas Gerais, 12 2015. [Online]. Available: http://homepages.dcc.ufmg.br/~ramon.pessoa/master_thesis/20151218-final-ramon-pessoa_master-thesis.pdf

[3] R. F. Pessoa, W. R. Schwartz, and J. A. d. Santos, "An experimental comparison of feature extraction and distance metrics for image retrieval," in *XXVIII Conference on Graphics, Patterns and Images (SIBGRAPI). Salvador, Brazil, August 26-29, 2015*, 2015.

[4] O. A. Penatti, E. Valle, and R. d. S. Torres, "Encoding spatial arrangement of visual words," in *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*, ser. Lecture Notes in Computer Science, C. San Martin and S.-W. Kim, Eds. Springer, 2011, vol. 7042, pp. 240–247.

[5] T. Tuytelaars, "Dense interest points," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2010*. IEEE, 2010, pp. 2281–2288.

[6] O. A. B. Penatti, F. B. Silva, E. Valle, V. Gouet-Brunet, and R. da S. Torres, "Visual word spatial arrangement for image retrieval and classification," *Pattern Recognition*, vol. 47, no. 2, pp. 705–720, 2014.

[7] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk, "Slic superpixels compared to state-of-the-art superpixel methods," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 11, pp. 2274–2282, Nov. 2012.

[8] C. Caetano, S. Avila, S. Guimarães, and A. d. A. Araújo, "Representing local binary descriptors with bossanova for visual recognition," in *Proceedings of the 29th Annual ACM Symposium on Applied Computing*, ser. SAC '14. ACM, 2014, pp. 49–54.

[9] K. van de Sande, T. Gevers, and C. Snoek, "Evaluating color descriptors for object and scene recognition," *TPAMI*, vol. 32, no. 9, pp. 1582–1596, Sept 2010.

[10] R. O. Stehling, M. A. Nascimento, and A. X. Falcão, "A compact and efficient image retrieval approach based on border/interior pixel classification," in *Proceedings of the Eleventh International Conference on Information and Knowledge Management*, ACM. New York, NY, USA: ACM, 2002, pp. 102–109.

[11] S. Avila, N. Thome, M. Cord, E. Valle, and A. De A. AraúJo, "Pooling in image representation: The visual codeword point of view," *Computer Vision and Image Understanding (CVIU)*, vol. 117, no. 5, pp. 453–465, 2013.

[12] O. A. B. Penatti, "Image and video representations based on visual dictionaries," Ph.D. Thesis, Universidade Estadual de Campinas, 11 2012.

[13] S. E. F. de Avila, "Extended bag-of-words formalism for image classification," Ph.D. Thesis, Universidade Federal de Minas Gerais, 6 2013.

[14] J. Z. Wang, J. Li, and G. Wiederhold, "Simplicity: Semantics-sensitive integrated matching for picture libraries," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 9, pp. 947–963, 2001.

[15] H. Kang, M. Hebert, and T. Kanade, "Image matching with distinctive visual vocabulary," in *IEEE Workshop on Applications of Computer Vision (WACV), 2011*. IEEE, 2011, pp. 402–409.