UNIVERSITÀ DEGLI STUDI DI TRENTO

Department of Information Engineering and Computer Science

Master thesis on
Telecommunication Engineering

Remote Sensing Image Description Methods in JPEG Compressed Domain for Large-Scale Remote Sensing Image Retrieval

Advisor	Candidate
Begüm Demir	Francesco Quiri
Co-Advisor	
Lorenzo Bruzzone	

Academic year 2016/2017

Aknowledgement

Thanks to all, my sincere companions

...laughing,
in the mechanism.
Agrippa

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

In the last decades analysis and processing of Remote Sensing (RS) data have become a very important tool for monitoring the Earth and human activities, due to the intrinsic ability of aircrafts or spacecrafts to have a large scale view of the Earth surface. RS image are useful in fields such as environmental monitoring, land processes, atmospheric science, hydrology, oceanography etc.Nowadays the advancements in RS and computer technology and the ever-increasing number of civilian satellites give birth to the so-called RS big data era [1, 2]. Big data as explained in [3], is a concept that mainly is described by the known three V, that are Volume, Variety and Velocity. *Volume* , refers to the quantity of data. Thinking for example that Sentinel mission alone is capable to collects more than 20 TBs of data per day. *Variety* is a notion about the heterogeneity of data. In fact today exists a wide range of sensors and platforms [4]. In the past satellite missions were focused on a single sensor, while currently trend is following a cooperative sensing approach. A clear example is given by constellations like Sentinel missions. The approach to use multiple sensors in a mission leads to the wide variety of data. Paradigm like tandem mission always like Sentinel satellites is related do the speed of acquisition, in fact the third big data V stands for *Velocity* . For example Sentinel-2 has a revisit time of 10 days at equator if only one satellite is considered or 5 days, always at the equator, if both satellites are considered [5].During the years big data analysts add other terms to define big data, such as *Veracity* (uncertainty of data), *Variability* and *Complexity* , *Visualization* and *Value* . It is clear that big data environment is challenging and difficulties lie in all sub-system of data handling pipeline, such as managing, processing, storing, loading, *etc*. Then without an effective solution risk that most of the information remains buried and never accessed is real. Researches to solve this problem have proposed various information retrieval paradigms, one of the most famouse for visual informations and images is the so-called Content-Based Image Retrieaval(CBIR). Generally a CBIR system is composed by two stages [6]: the first is concerning feature extraction, that is the process which find a discriminative description of image content. Second procedure is about image matching, in which query image descriptors are compared with those of all other images present in the archive and therefore the most relevant are retrieved.The main issue to process a huge amount of images is the computational load of processing algorithms and consequently computing heaviness directly affects processing time. For these reasons, in various type of applications, is not advantageous to processing the entire information enclosed in images, but is enough to exploit only a part of it. For example it is not uncommon to perform investigations on the so called quick look (QL) products, that is a subsampled and/or lossy compressed version of the original data. QL products are mainly used by the expert user, in order to visually inspects the image before downloading it, but in the last years automatic analysis methods for QL imagery were proposed for lightening processing chain and obtaining acceptable results without employing the full resolution products [7–9]. These new methodologies could resolve in part the new problems of management, processing and interpretation of RS data arisen from massive availability of the latter. The accessibility of large volume of RS data is largely attributed to some new policies introduced on free access and distribution of satellites data, as the one of Landsat products from USGS [10].Obviously the analysis of QL imagery have an enormous advantage in terms of data volume reduction, but still is not enough when the analysis are done at global scale [7], or simply when is necessary to execute advanced image processing in acceptable time, due to the fact that the images need to be decompressed and then processed. It is known that decompression is an overhead time-consuming operation, that is not related to the final aim of the image investigation, so is clear that a strong needing in development of compressed domain processing tools exist also for RS imagery.

For these reasons in this thesis, it has been proposed and studied the effectiveness of RS CBIR techniques based on JPEG compressed images. JPEG is a lossy compression standard widely used on internet, famous due to its good compression rate and image quality. JPEG compression scheme is composed by a series of operation: image block partition, block based Discrete Cosine Transform (DCT), that is a Fourier-related transform, quantization and final entropy coding. Decompression chain is realized by performing inverse steps in inverse order, the critical and most time consuming operation is the inverse DCT (IDCT) step, due to number of operation required. Taking in account this fact, in the past years researchers and practitioners proposed and studied methods for manipulating and/or extracting information without fully decompressing the image. Usually this result is achieved by partially decompressing the image and then apply processing techniques directly in DCT domain. JPEG-related compressed techniques and in general lossy compression schemes are rarely used in RS data applications. In fact sometimes they are used in compression systems mounted on-board of satellites, while in data-centers images are stored with lossless compression techniques. However in geographic web-centric services lossy techniques are widely used, as in the case of QL imagery, and, how it was proven in [7–9], the analysis of lossy compressed products is an effective solution for resolving, with acceptable efficiency, the problem of information retrieval from the massive RS EO archive.

The remaining part of the thesis is structured as follows. Chapter 2 gives an overview of JPEG standard, in order to understand how the compression chain works, and introduction to extracted features from compressed-domain image. Chapter 3 regards test of proposed features in the context of basic k-NN CBIR architecture. Chapter 4 explains how to exploit this kind of features in a supervised CBIR environment. Chapter 5 shows the benefits in terms of processing time of proposed descriptors. Finally, Chapter 6 presents the conclusions of the methods proposed in this work.

Chapter 2

Related works

How already introduced in the first chapter, main goal of this thesis is to understand the effectiveness of compressed domain features. Then before explaining proposed methods is necessary to illustrate JPEG compression standard and give a review concerning principal JPEG-based features. Thereafter bag of visual words representation and Multiple Kernel Learning (MKL) with its correlated arguments are introduced.

2.1 JPEG Compression

JPEG standard is referred to a family of specifications that define encoding/decoding algorithms for images and data-stream architecture for generating and describing compressed data. The first version of JPEG, the so-called JPEG Baseline, was standardize in 1992 [11] from the joint ISO/CCITT committee known as Joint Photographic Experts Group (JPEG). JPEG compression standard were designed to be a generic algorithm to supports a broad variety of applications and during the years become the most famous compression method for grayscale and color still image, largerly due to its good compression performance and image quality. From now on baseline JPEG is referred simply as JPEG and for sake of simplicity only grayscale images are considered. The JPEG encoder consists of a forward DCT transform step, a quantizer and a Huffman encoder. All components of encoder/decoder stage are feed with image patch large 8x8 pixels, so before starting the compression procedure the image is splitted in non-overlapping 8x8 blocks and pixels' value is subtracted by 128. Pre-processing steps are done in order to reduce size of final coded image stream as much as possible. Blocks are analized from left to right, top to bottom. Clearly decoding stages are done by the inverting operations and done in inverse order. In the next subsections a explanation of each compression step is given.

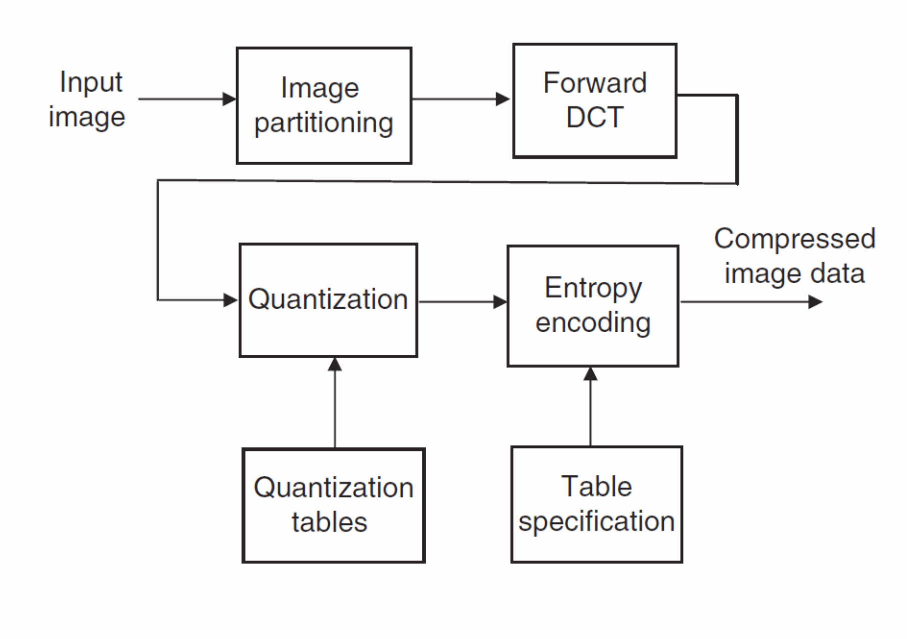


Figure 2.1: Block diagram of JPEG encoding process

2.1.1 Block Discrete Cosine Transform

The first stage of compression process is the conversion of each block from pixel domain to DCT domain. Forward DCT transform estimates a total of 64 real-valued coefficients from 8x8 block. Mathematically DCT transform is a lossless transform, but in real world, due to limit number of bits, it is the first source of lossiness of JPEG compression chain. Technically DCT is a Fourier-related transform, in fact is similar to the Discrete Fourier Transform (DFT), except that involve only real numbers, therefore fast algorithms variations, like FFT, or other properties are in common to both. Mathematical description of DCT trasform is given below:

Forward:

$$F(u,v) = \alpha(u)\alpha(v)\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)\cos\left[\frac{(2x+1)u\pi}{2N}\right]\cos\left[\frac{(2y+1)v\pi}{2N}\right] \tag{2.1.1}$$

Inverse:

$$f(x,y) = \sum_{u=0}^{N-1}\sum_{v=0}^{N-1} \alpha(u)\alpha(v)F(u,v)\cos\left[\frac{(2x+1)u\pi}{2N}\right]\cos\left[\frac{(2y+1)v\pi}{2N}\right] \tag{2.1.2}$$

In which: $\alpha(u) = \begin{cases} \sqrt{1/N} & \text{if } u = 0 \\ \sqrt{2/N} & \text{if } u = 1, 2, ..., N-1 \end{cases}$ and $N = 8$

From the above forward transform equation 2.1.1 is clear that pixels are expressend in terms of sum of cosine oscillating at various frequencies. In JPEG 64 orthogonal basis are used and goes from frequency $(0, 0)$ to frequency $(7, 7)$. Generally frequency $(0, 0)$ is called DC component, because represent the mean pixel's color, instead the other 63 coefficients are called AC coefficients, because they express frequency behaviour, or texture variation, of pixels.

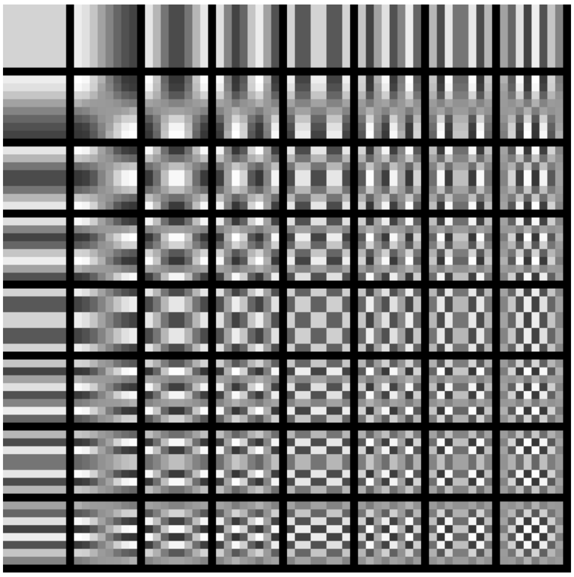


Figure 2.2: 8x8 DCT basis

DCT is used for compression scheme, because has the capabilities of concentrating most of pixel energy in few coefficients, so shortly after transform fewer values with smaller magnitude are stored, instead of 64 pixels with a larger bit-depth. Another important property of DCT transform for compression purposes is the decorrelation of coefficients, which means that coefficient can be optimally compressed.

2.1.2 Quantization

After DCT transform step, frequency coefficients are quantized in order to achieve an higher compression rate. This step introduce a strong distortion and for this is the main source of lossiness of the entire compression algorithm. Quantization consists in rounded to the nearest integer division of DCT coefficients and could be more or less harsh in function of the quality factor (QF). JPEG standard does not specify how much a coefficient should be scaled, then the so-called ***Quantization Matrix*** (QM), that is a 8 by 8 matrix that contains all quantizer step-size parameters, is decided by the user and each coefficient is tuned by setting a QF. Specification indicates that quantizer step-size coefficient could be any integer from 1 to 255. Since human visual system is very sensitive to small distortions in low frequencies, rather than distorsions in higher frequencies, accordingly the optimal trade off between high compression rate and image quality preservation, that means with ideally imperceptible visual artifacts, is achieved by a greater quantization of high frequencies with respect to lower ones. Mathematically speaking each coefficients is quantized by a linear quantizer, then the maximum distortion that can be reached is equal to one half of quantizer step-size. Formally quantization is described as:

$$F_q(u,v) = round\left(\frac{F(u,v)}{Q(u,v)}\right) \tag{2.1.3}$$

Where $Q(u,v)$ is a quantizer step-size coefficient and $round()$ will round to the nearest integer. Then reconstruction step follows the straight inverse operation:

$$\tilde{F}(u,v) = F_q(u,v) \times Q(u,v) \tag{2.1.4}$$

In which $\tilde{F}(u,v)$ is the reconstructed coefficient. As already mentioned JPEG does not specify a precise quantization table, but only range of possible value for the quantization coefficients. Then design of quantization tables can be done either based on rate-distortion theory, so by minimizing image visual distortion, or on human visual system characteristics. An example of quantization table is given below:

Table 2.1: Example of quantization table

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

2.1.3 Huffman Encoding

After quantization step, in order to optimize as more as possible effectiveness of entropy encoder, DC and AC coefficients are treated separately.

DC coefficients Encoding

Considering the fact that adjacent blocks have similar DC components, is straight to encode the difference between subsequent DC values instead their raw value independently. So for obtaining the optimal compression of DC coefficients a predictive encoding approach is necessary. Mathematically the difference $DIFF$ is defined as:

$$DIFF = F_q(0,0) - PRED \tag{2.1.5}$$

In which $PRED$ is the difference value of the previous block. Entropy coding is done by means of the so-called Huffman Tables (HTs). HTs contain binary codes and class numbers that correspond to a specific $DIFF$ magnitude value. Precisely $DIFF$ value, that could be either positive or negative, is used to select a class, or category. Noticed that a class correspond to a subset of possible $DIFF$ values, so once the class is selected additional bits are added and they represent the exact value of $DIFF$. Below the two HTs for DC components are shown:

Table 2.2: Category symbols and bit code

Category (CAT)	**Bit Code Word**
00	00
01	010
02	011
03	100
04	101
05	110
06	1110
07	11110
08	111110
09	1111110
0A	11111110
0B	111111110

Each category of 2.2 specify how many additional bits are needed to encode $DIFF$ value. Additional bits stream are defined by the second HT.

SSS	Size	Additional Bits	DC Value
00	0	~	0
01	1	0, 1	-1,1
02	2	00, 01, 10, 11	-3,-2, 2, 3
03	3	000, 001, 010, 011, 100, 101, 110, 111	-7, -6, -5,-4, 4, 5, 6, 7
04	4	0000,..., 0111, 1000,..., 1111	-15,...,-8, 8,...,15
05	5	0 0000,..., 1 1111	-31,...,-16, 16,...,31
06	6	00 0000,..., 11 1111	-63,...,-32,32,...,63
07	7	000 0000,..., 111 1111	-127,...,-64, 64,...,127
08	8	0000 0000,..., 1111 1111	-255,...,-128, 128,...,255
09	9	0 0000 0000,..., 1 1111 1111	-511,...,-256, 256,...,511
0A	10	00 0000 0000,..., 11 1111 1111	-1023,...,-512, 512,...,1023
0B	11	000 0000 0000,..., 111 1111 1111	-2047,...,-1024, 1024,...,2047

Table 2.3: DC additional bits

Huffman tables were proposed as integrating part of JPEG standard, because they work well in generic purpose environment, but is possible to define ab optimize version of these tables by using Huffman's algorithm on the specific image.

AC coefficients Encoding

While for encoding DC coefficients is necessary to exploit correlation between adjacent blocks, for optimally encode AC coefficients is require to take advantage of the energy distribution within DCT block. As already mentioned after quantization is highly probable that most of the non-zero coefficients lie on the low frequencies basis. Therefore AC coefficients are reorganized taking into account of this property. JPEG standard reorganized coefficients with a zig-zag order.

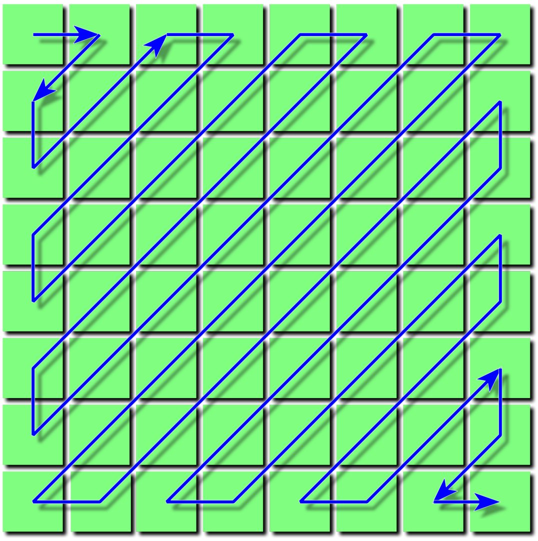


Figure 2.3: Zig-zag reordering

How is clear from figure 2.3 final coefficients vector is formed by keeping near contiguos low frequency coefficients, thus exploiting energy distribution after quantization step. Now coefficients stream is encoded by means of Run-Length(RL) encoding technique. As for DC coefficients, also AC coefficients are encoded through two HT, but they work in slightly different way. In this case most significat bits represent the run-length of zeros between two non-zero coefficients, while least significat bits are reserved for encoding the magnitude of the non-zero coefficient that interrupts zeros sequence. As for DC HTs, also tables of AC coefficients can be optimize in function of image content and distribution.

Finally after encoding step data stream is enclosed in a structure delimeted by headers that contains all the informations needed at decoding stage.

2.2 JPEG Compression for Remote Sensing Images

Usually RS applications need high quality data, in order to do reliable analysis on specific characteristic of investigated scene. Therefore studies on RS imagery compressed in JPEG format are few. In [12] different compression algorithms, including JPEG, have been compared, to study the effects of compression on automatic land classification. Then JPEG is used sometimes on board satellite compression system, in fact in [13] a partial list of the compression algorithms used in different space missions is shown. Obviously JPEG applied in space mission is severly limited due to image quality constraints. Then in [14] JPEG and JPEG200 have been compared to study classification capabilities at different compression rates. Also [15] follows the same direction, in which classification accuracy is measured on JPEG compressed images at different compression ratios. More recently JPEG was used also in the context of big data processing as in [7], in which a composite global scale image is produced by selecting cloud-free images from QL products. While in [8] a cloud detection is tested on JPEG QL imjages of LANDSAT 7/ETM+ scenes. In [9] an automatic image processing architecture based on JPEG compressed QL produts is proposed and tested. Therefore recent trends indicate that JPEG is preferred in web environment and working with web products, like QL, could become the standard method for analyzing huge amount of data.

2.3 Image Description Methods on JPEG Compressed Domain

The aim of present work is to propose and study the effectiveness of local features derived from DCT domain that. JPEG related features were widely studied in field such as computer vision, patter recognition and other, while don't exist research or proposal in RS topics. One of the first attempt of CBIR that exploits compressed-domain feature were conduct by Shneier and Abdel-Mottaleb in 1996 [16] in which they divide DCT image in some sub-images, or windows, and construct a kind of keys by averaging DCT coefficients of each block across the window. For each pair of window present in the image, they perform the difference and if the magnitude is under a certain threshold value 0 is assigned, otherwise 1. This results in a feature vector that is compared with other present in the database by means of hamming distance. Lay and Guan in [17] exploit the low frequency AC coefficients to build an histogram of the image and then retrieving performance are measuring by means of L1 norm comparison. Shaefer in [18] employ the so-called DC image to extract color histogram and texture feature with local binary pattern(LBP) operator, both color and texture histogram are compared with L1 norm. Method in [19] exploit color information by extracting four mean colors, or DC values, of the 4x4 sub-block derived from each 8x8 JPEG block, hence a color histogram is built among extracted DC components. Feng and Jiang in [20] take advantage of a statistical analysis of block DCT coefficients to roughtly estimate textural behavior, so two histograms are built using subtle pixels mean and variance calculated in compressed domain. Eom and Choe in [21] use an edge histogram detector in order to describe image content. The edge orientation is determine by using the magnitude of low frequency coefficients, remember that low frequency coefficients reflect directional texture behavior. In Lu, Li and Burkhardt [22] color, texture and basic edge information is extract by combining various subset of DCT coefficients. Then some efforts in [23] were done by reorganizing DCT coefficients of all blocks in multiresolution fashion, as happens in wavelet rearrangement. Is worth of mentioning also methods that exploit JPEG header information as in [24,25]. They based recognition process by exploiting the so called optimize huffman and quantization tables, that are made in function of image content. Another feature that is possible to extract from DCT domain is the so-called Markov features [26], in which a transition probability matrix is constructed from the residual of the subtraction between coefficients. So basically this features model the statistical coefficient behaviour of investigated area. Has shown in literature, most of proposed features are derived by selecting some coefficient from each block, so is natural to follow a similar approach to derive low-level features.

2.4 Remote Sensing Content-Based Image Retrieval Systems

Due to the rapid growth of data volume, CBIR become an essential tool for retrieve informations from massive EO archive, for this reason interest of RS community for CBIR systems has been increasing in the last years. Several low-level features have been investigated in CBIR applications, as intensity features [27–29], or shape features as in [30, 31] and texture features [32, 33]. Different researchers proposed CBIR architectures in which local features, as SIFT [34], are exploited [35]. In this last work SIFT is used in conjunction with bag of visual word representation. In [36] a comparison between various local features and global features has been conducted. In [37] a retrieval system is implemented by means of morphological texture descriptors, used in the form of bag-of-morphological-words representation. In [38] a local binary pattern(LBP) like feature is proposed. Other methods are based on the user interaction, for example Schröder *et al.* in [39] proposed a probabilistic framework for retrieval system based on interacting learning. Other feedback-based CBIR architectures are [40, 41] or again [42]. The main problem of CBIR system is the so-called semantic gap, then in order to mitigate its effects region based model have been proposed, as in [43–46]. In these works both region attributes and spatial relationship among regions are modelled. In [43] each image is partitioned in regions, which properties are nmodelled as graph nodes and spatial relationships among regions are represented as graph edges. In [44] first regions are formed by a pixel-based classifiers jointly with a split-and-merge technique, then attributed relational graphs are used to represent both the region information and their spatial relations. Semantic representation extracted from low level features is exploited in [45]. In [46] regions of interest are finded with maximally stable extremal regions(MSER) method, which selects highly stable regions from all possible, then a graph is used to describe spatial arrangements.

In this work a CBIR system based on compressed features is proposed. Has shown in JPEG compressed domain features literature, most of proposed features are derived by selecting some coefficient from each block, so is natural to follow a similar approach to derive low-level features. It is clear that due to embedded characteristics of the JPEG compression standard, i.e. blocks image subdivision of limited size, it is straight to extract local information.A widely used approach to exploit local features is to use the so-called bag-og-visual-word(BOVW) representation. In RS there are many studies on this subject, for example in the context of recognition related problems, following some works about it are listed. In [47] is applied to detect landslide, in [48] is used jointly with some global features to perform scene classification on high spatial resolution imagery. In [49] a multi BOVW representation derived from different features is used and is extended with a feature significance score. In [50] a hierarchical coding approach is used and final representation is made by means of Fisher coding step. In the context of scene classification in [51] correlatons are used in order to integraete pixel homogeneity. In [52] an approximate earth movers distance is proposed in the framework of BOW based on SIFT features. For integrating spatial and word distribution in [53] a spatial pyramid image subdivision join to a co-occurence word measure is investigated. In [54] BOW is used with a combination of various texture and spectral features for perform land use land cover(LULC) mapping purposes. In order to estimate the best method for selecting interesting patches, from witch to extract features, in [55] BOW is used. Always in the framework of scene classification BOW is used [56] feature automatically learned directly from the image with an unsupervised method. Another attempt for integrating spatial information in BOW model is considered in [57], in which circular image portions are considered in aggregation step. In [58] a BOW representation based in the so-called pyramid of spatial relatons is proposed, this method is proposed for integrating both absolute and relative spatial relationships between local features. In [59] BOVW is compared with the so-called bag of topics, that is a description method that take into account semantic informations. As is clear from this short review, BOW became an important description method for performing discriminative analysis on RS imagery. The mathematical formalization of BOW, or Bag-Of-Visual-Words(BOVW), or more generally Bag-Of-Features(BOF), representation. As BOW name suggests, this technique was initially developed in the context of text and documents retrieval and further it was adapted to visual information. First visual adaptation of BOW was proposed in [60]. BOW pipeline is composed by the following steps:

1. First local low-level feature are extracted from image. Local features are a visual representiation of a limited portion of the image. Famous features largerly used also in RS are SIFT [34] and the respective dense sampling version [61] , HOG [62] and others …

2. The second step regards the individuation of a subset of prototypes learnt in the feature space. Commonly unsupervised clustering methods as k -means [63, 64] or c -means [65] are used. Collection of meaninful prototypes is called vocabulary, while prototypes-self are called visual words, anchors, centers or atoms.

3. After definition of dictionary, low-level features are transformed in the so-called mid-level representation. This process is called featurs coding and is introduced in order to express each low-level descriptor with a combination of visual words.

4. Finally mid-level features are combined in order to obtain final image signature.

Then this final representation is the feature that feed a classifier of a retrieval module.

Every step of the BOW pipeline strongly affects final quality of image representation. Therefore a particular attention must be give to the combination of techniques used in each processing step for obtain positive results. In the following section a mathematical formalization of features coding and pooling is given.

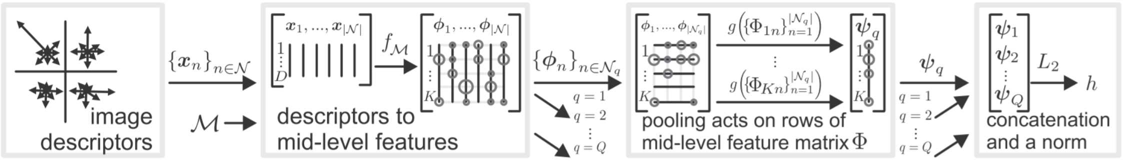


image descriptors
$\{x_n\}_{n\in\mathcal{N}}$
$\mathcal{M}$
$x_1, ..., x_{|\mathcal{N}|}$
1
D
$f_{\mathcal{M}}$
$\phi_1, ..., \phi_{|\mathcal{N}|}$
1
K
descriptors to mid-level features
$\{\phi_n\}_{n\in\mathcal{N}_q}$
$q=1$
$q=2$
$q=Q$
$\phi_1, ..., \phi_{|\mathcal{N}_q|}$
$g\left(\{\Phi_{1n}\}_{n=1}^{|\mathcal{N}_q|}\right)$
$g\left(\{\Phi_{Kn}\}_{n=1}^{|\mathcal{N}_q|}\right)$
ψ_q
pooling acts on rows of mid-level feature matrix Φ
ψ_q
$q=1$
$q=2$
$q=Q$
ψ_1
ψ_2
ψ_Q
L_2
h
concatenation and a norm

Figure 2.4: Bag of visual words pipeline. Adapted from ”Comparison of mid-level feature coding approaches and pooling strategies in visual concept detection”, by Piotr Koniusz, Fei Yan, Krystian Mikolajczyk, 2013,*Computer Vision and Image Understanding*, Volume 117, Issue 5,479-492

Once visual dictionary is composed, is necessary to enclose low-level features in the so-called visual vocabulary space. Let assume a feature vector $\boldsymbol{x}_n \in \mathbb{R}^D$ such that $n = 1, ..., N$ and N is the total number of features extracted from the image I. At that point consider to have a visual vocabulary composed by K anchors, so avery visual atom is indicated as $\boldsymbol{m}_k \in \mathbb{R}^D$. Therefore visual dictionary is described as a matrix $\mathcal{M} = \{\boldsymbol{m}_k\}_{k=1}^K$ such that $\mathcal{M} \in \mathbb{R}^{D\times K}$. Then formalism for coding and pooling step is shown below:

$$\begin{aligned} \phi_n &= \Big[\Phi_{1n}, ..., \Phi_{Kn}\Big]^T \\ &= f(\boldsymbol{x}_n, \mathcal{M}) \end{aligned} \qquad \forall n \in N \tag{2.4.1}$$

$$\Psi_k = g(\Phi_{kn}) \tag{2.4.2}$$

$$\boldsymbol{h} = \boldsymbol{\Psi}/\|\boldsymbol{\Psi}\|_2 \tag{2.4.3}$$

Equation 2.4.1 represents the mapping from features space to the visual dictionary space and f is the mapping function such that $f : \mathbb{R}^D \mapsto \mathbb{R}^K$. Shortly it describes the image contents in terms of visual words. Noticed that dictionary learning step is not included in the mathematical analysis. Equation 2.4.2 is the mathematical formalization of the pooling or aggregation step, that is the process in which mid-level features are quatified. Last equation 2.4.3 represents the normalization of the image signature, that is useful to preserve only the relative statistics of a specific visual words.

Aggregation step, namely equation 2.4.2 and 2.4.3, could be reformulate in order to taking into account image spatial information. Precisely consider an image split in Q partitions, then aggregation step could be generalized, by applying independently pooling operation to each partition. Then aggregation equations are extended as follow:

$$\psi_q = \left[\Psi_{1q}, ..., \Psi_{Kq}\right]^T \ , \ \Psi_{kq} = g(\{\Phi_{kn}\}_{n \in N_q}) \ , \ \forall q = 1, ..., Q \tag{2.4.4}$$

$$\boldsymbol{h} = \widehat{\boldsymbol{h}}/\|\widehat{\boldsymbol{h}}\|_2 \ , \qquad \widehat{\boldsymbol{h}} = \left[\psi_1^T, ..., \psi_Q^T\right]^T \tag{2.4.5}$$

In which N_q is the total number of subimage, while q is the partition index. A famous approach that exploit spatial subdivision of the image is Spatial Pyramid Matching(SPM) [66]

For a in-depth review of feature coding and pooling methods, please refer to [67, 68]

2.5 Overview Multiple Kernel Learning Methods

Since the introduction of Support Vector Machine(SVM), the biggest problem is to choose the kernel function that best fits the problem. Many successful kernel functions have been proposed in the literature over the years, some times also enginnered for specific applications. Then the optimal selection of parameters in the kernel function is one of the decisive elements for the proper functioning of the kernel-method used. A possible solution to avoid the hard task of kernel selection is to combine multiple kernel functions together, even if not necessarily optimized for the specific task. At this point the problem is to understand how to combine the various functions in order to increased as most as possible final results. In recent years, a technique called multiple kernel learning(MKL) has been proposed to solve the problem of the optimal combination of kernel functions. Aim of MKL is to learn the weights to associated to a specific kernel depending on its importance in relation to the final result. A straightforward approach to combine kernel functions is to weighting and linearly combining them. Following formalism used in [69] let $T\{\boldsymbol{x}_i, y_i\}_{i=1}^n$ be the training set amde up of n labeled featuressamples, where $\boldsymbol{x}_i \in \mathbb{R}^D$ is the i-th sample associated with the binary class label $y_i \in \{+1, -1\}$. From this labeled samples, M basis kernels $\{\boldsymbol{K}_1, \boldsymbol{K}_2, \ldots, \boldsymbol{K}_M\}$ are constructed, where $\boldsymbol{K}_m(\cdot, \cdot) = \langle \phi_m(\cdot), \phi_m(\cdot) \rangle$ is the m-th basis kernel associated to the so-called reproducing kernel Hilbert space(RKHHS) and is specified with $\mathcal{H}_m$. Then the linear combinantion of M kernel function is given by:

$$\boldsymbol{K}_c = \sum_{m=1}^{M} d_m \boldsymbol{K}_m \tag{2.5.1}$$

In which $\boldsymbol{K}_m$ is the m-th kernel function and d_m is its associated weight. In MKL framework exist various method to resoplve the problem of linear combination, for example the easiest is to consider a fixel-rule MKL algorithm, for example by averaging to the number of involved kernel function. The most effective method of using a MKL approach is to use an optimization algorithm, at this point there are two types of algorithms: classifier-dependet or classifier-indipendent. In the first type of algorithms, kernel function weights and classifier parameters are optimized simultaneously. Then the estimated weights are used to construct the composite kernel and the final classifier is trained [70]. In turn, classifier-dependent algorithms are divided into two other categories: *direct* algorithms or *wrapper* algorithms. In *direct* algorithms, the kernel weights and the SVM parameters are directly estimatedthrough the optimaztion of the so-called primal (or dual) problem of the MKL. Depending on used direct MKL algorithm, primal or dual problem is reformulated in function of the optimization strategy used. Machine learning literature is rich of this type of reformulations, for example in [71] a primal problem is optimized by means of semidefinite programming. The second type of classifier-dependent MKL algorithm is the so-called *wrapper* algorithms. In this typology of algorithms a two-step optimization procedure, to obtain the kernel weights and the SVM parameters, is used. The first step consist in assign some initial values to the weights associated to the basis kernels and then SVM parameters are solved. Then in the second step SVM parameters are fixed to the values obtained in the initial step and then the SVM objective function is optimized with respect to kernel weights. A powerful wrapper algorithm, also used in this thesis, is the so called generalized MKL(GMKL) [72].

The other kind of algorithms are the so-called classifier-independent MKL algorithms, which aims to model the target function independently from the used classifier. Then in this typology of algorithms a simpler optimization method is used. Due to this semplification, this kind of MKL algorithm can easily handle several kernels without difficulties. Classifier-independent MKL algorithm are subdivided in: similarity-based algorithms, generalization-error-based algorithms, subspace-based algorithms and heuristic MKL algorithm.

The aim of similarity-based algorithms is to optimize the composite kernel function with respect to a measure of similarity to a so-called target kernel. A famous similarity measure for kernel function is the so-called kernel alignment(KA) investigate in [73], which estimates the cosine angle between two kernels $\boldsymbol{K}_p$ and $\boldsymbol{K}_q$ by using:

$$KA(\boldsymbol{K}_p, \boldsymbol{K}_q) = \frac{\langle \boldsymbol{K}_p, \boldsymbol{K}_q \rangle_F}{\sqrt{\langle \boldsymbol{K}_p, \boldsymbol{K}_p \rangle_F \langle \boldsymbol{K}_q, \boldsymbol{K}_q \rangle_F}} \tag{2.5.2}$$

KA works exactly as the standard cosine similarity, the when is equal to 0, means that the operands are completely dissimilare, while if it is equal to 1 means that the two involved kernel are completely similar.

One of the most recent development in MKL is the so-called Generalization error-based MKL algorithm, in which weights learning is based on the minimization of the upper bound of the leave-one-out error which is considered as the estimation of the expected generalization error of the classifier [74].

Then there is the so-called subspace-based MKL algorithms in which the basis kernel are initially reshape as vectors and then stacked in a matrix. Finally using a sbuspace learning method a 1D subspace is learned. Then the kernel weights are represent by the projection of each vectorialized-kernel on to the learned 1D subspace. The first proposed algorithm of this category of MKL methods was the so-called Representative MKL, proposed in [75].

Finally last classifier-independent MKL algorithm type is the so-called heuristic MKL, in which some heuristic rules are used to learn kernel weight. For example an easy heuristic method exploit a cross-validation procedure, in which the weights are selected from a predefiniteset of candidate values.

For the purposes of this thesis it is not necessary to go further, therefore please refer to [76] for a complete theoretical view of MKL and for MKL RS applications to [69].

Chapter 3

Considered Unsupervised Remote Sensing Image Description Algorithm in JPEG Compressed Domain

In this chapter an explanation about the unsupervised proposed setting is shown. Following a general explanation is given, then proposed features are explained.

3.1 General Architecture

As already shown, JPEG-compressed features are considered as such when are extracted from the DCT-coefficients image. Obviously in real application that means to perform a partial decoding of JPEG image, in fact first step consists of applying Huffman decoding to compressed image. Then a features extraction step takes place, in next paragraphs proposed features are presented. Consider that each color band is compressed independently, therefore feature extraction scheme is performed in every image band. Finally BOVW representation is used to form final image signature. Concerning dictionary, visual atoms are extracted from features database by means of standars k-means algorithm [63, 64]. Then coding step is applied and in this work method proposed in [60] is followed, therefore so-called hard assignment is applied.

As is shown in figure 3.1 hard assignment associates features, in this case represents by green triangle, to the nearest visual word. Is clear that hard quantization is the simplest approach to apply, but has also the biggest loss of information. Below mathematical formalization of hard assignment is given:

$$\begin{gathered}\boldsymbol{\phi} = \underset{\bar{\boldsymbol{\phi}}}{\operatorname{argmin}} \|\boldsymbol{x}_n - \mathcal{M}\bar{\boldsymbol{\phi}}\|_2^2 \\ s.t. \|\bar{\boldsymbol{\phi}}\|_1 = 1, \bar{\boldsymbol{\phi}} \in \{0,1\}^K\end{gathered} \tag{3.1.1}$$

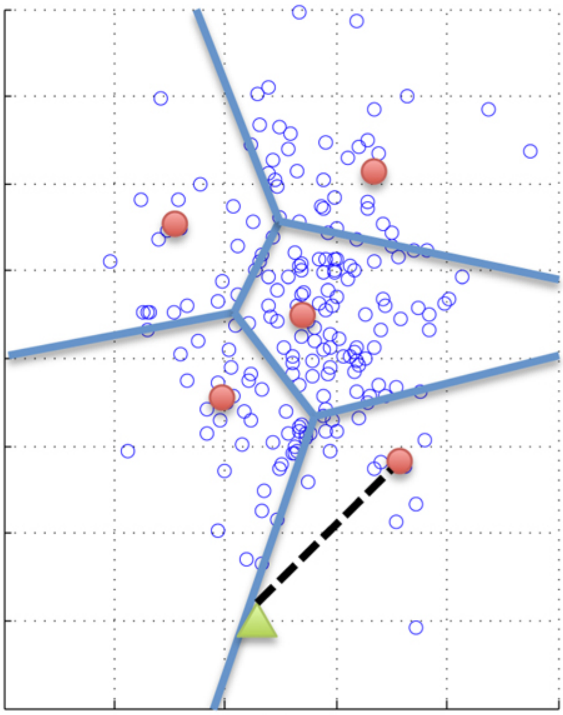


Figure 3.1: Visualization of Hard assignment process

Equation 3.1.1 means that every descriptor $\boldsymbol{x}_n \in \mathcal{X}$, which $\mathcal{X}$ is the set of all image descriptors, is assigned to its nearest cluster with activation function equal to 1. Always following seminal work [60], after each image features is descripted in the visual dictionary space an average pooling step is applied. Average pooling, or simply sum pooling, counts all occurences of some visual word $\boldsymbol{m}_k$ in the image and normalizes such counts by the total number of coded descriptors in the image. Formally:

$$\begin{aligned} \widehat{\boldsymbol{h}}_k &= avg(\{\phi_{kn}\}_{n \in \mathcal{N}}) \\ &= \frac{1}{|\mathcal{N}|} \sum_{n \in \mathcal{N}} \phi_{kn} \end{aligned} \tag{3.1.2}$$

In equation 3.1.2 set $\mathcal{N}$ refers to the total number of features present in the image and k is the k-th word of dictionary $\mathcal{M}$.

3.2 Dominant Features

As mentioned in section 2.1.1 DCT transform concentrates most of energy in few low frequency coefficients. For this reason in [77] have been proposed features that take into account only DCT low frequency coefficients. Precisely five coefficients of each direction and DC value were taken as block local features. Thus inspiring by [77] from every color band these features are taken and then concatenated. In the image 3.2 below, extracted features are shown:

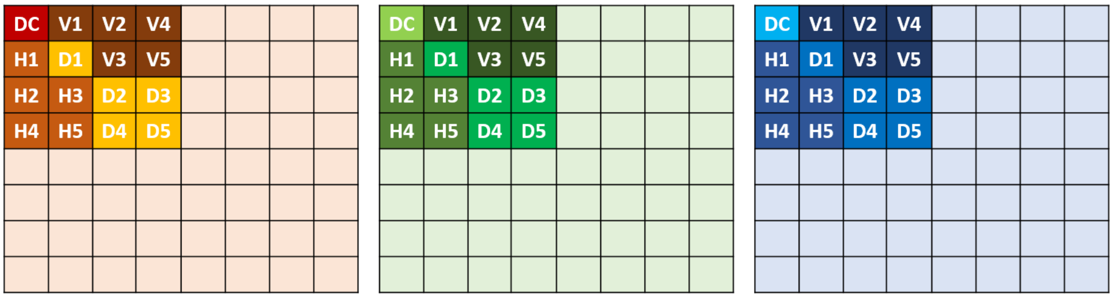


Figure 3.2: Extracted features from DCT blocks of each color band

In image 3.2, Hi coefficients are the horizontal edge/texture information, Vi coefficients are the vertical edge/texture information, Di coefficients are the diagonal edge/texture information and DC is the mean color of 8-by-8 image patch. After coefficients extraction, features vector with same type of information but different color band are concatenated, to forming final representation of a particolar block.

$$\begin{aligned} H_R &= [H_{1r}, \dots, H_{5r}]\ H_G = [H_{1g}, \dots, H_{5g}]\ H_B = [H_{1b}, \dots, H_{5b}] \\ V_R &= [V_{1r}, \dots, V_{5r}]\ V_G = [V_{1g}, \dots, V_{5g}]\ V_B = [V_{1b}, \dots, V_{5b}] \\ D_R &= [D_{1r}, \dots, D_{5r}]\ D_G = [D_{1g}, \dots, D_{5g}]\ D_B = [D_{1b}, \dots, D_{5b}] \end{aligned} \tag{3.2.1}$$

$$\begin{aligned} DC &= [DC_R, DC_G, DC_B] \\ H &= [H_R, H_G, H_B] \\ V &= [V_R, V_G, V_B] \\ D &= [D_R, D_G, D_B] \end{aligned} \tag{3.2.2}$$

Equations 3.2.2 are the final local feature vectors extracted from every DCT block. In the proposed architecture features are merge by performing fusion at representation level, that means that BOVW chain is applied independently to each kind of features and finally aggregation histograms are simply concatenated together.

Formally final image signature is obtained by:

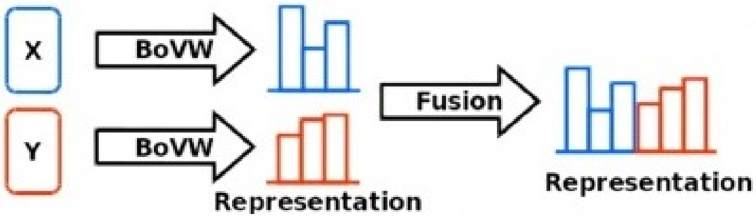


Figure 3.3: Representation level fusion illustration. Adapted from ”Bag of visual words and fusion methods for action recognition: Comprehensive study and good practice”, by Xiaojiang Peng, Limin Wang, Xingxing Wang, Yu Qiao, 2016, *Computer Vision and Image Understanding, Volume 150, 109-125*

$$\Phi_{DC} = f(DC, \mathcal{M}_{DC})\ \Psi_{DC} = g(\Phi_{DC}) \tag{3.2.3}$$

$$\Phi_{H} = f(H, \mathcal{M}_{H})\ \Psi_{H} = g(\Phi_{H}) \tag{3.2.4}$$

$$\Phi_{V} = f(V, \mathcal{M}_{V})\ \Psi_{V} = g(\Phi_{V}) \tag{3.2.5}$$

$$\Phi_{D} = f(D, \mathcal{M}_{D})\ \Psi_{D} = g(\Phi_{D}) \tag{3.2.6}$$

$$\widehat{\boldsymbol{h}} = \Big[\Psi_{DC}, \Psi_{H}, \Psi_{V}, \Psi_{D}\Big] \tag{3.2.7}$$

At the end of BOVW process final image signature is given by $\boldsymbol{h}$, that is $\mathcal{L}_1$ normalized version of $\widehat{\boldsymbol{h}}$.

3.3 Statistics from Dominant Features

An other considered local features extracted from blocks could be some statistics of coefficient, so in present work inspiring by [78], second central moment and fourth order moment are taken from features vectors proposed in the previous section described by the equations 3.2.1. Only these central moments are taken because during the experiments result to be the most informative statistics to extract from the features, while other central order moments result to affect negatively on the final performances of the system. Then features are the statistics extract from each band, followed by a concatenation:

$$
\begin{aligned}
H_{varR} &= [var(H_{1r},\ldots,H_{5r})]\ H_{kurtR} = [kurt(H_{1r},\ldots,H_{5r})]\\
H_{varG} &= [var(H_{1g},\ldots,H_{5g})]\ H_{kurtG} = [kurt(H_{1g},\ldots,H_{5g})]\\
H_{varB} &= [var(H_{1b},\ldots,H_{5b})]\ H_{kurtB} = [kurt(H_{1b},\ldots,H_{5b})]\\
V_{varR} &= [var(V_{1r},\ldots,V_{5r})]\ V_{kurtR} = [kurt(V_{1r},\ldots,V_{5r})]\\
V_{varG} &= [var(V_{1g},\ldots,V_{5g})]\ V_{kurtG} = [kurt(V_{1g},\ldots,V_{5g})]\\
V_{varB} &= [var(V_{1b},\ldots,V_{5b})]\ V_{kurtB} = [kurt(V_{1b},\ldots,V_{5b})]\\
D_{varR} &= [var(D_{1r},\ldots,D_{5r})]\ D_{kurtR} = [kurt(D_{1r},\ldots,D_{5r})]\\
D_{varG} &= [var(D_{1g},\ldots,D_{5g})]\ D_{kurtG} = [kurt(D_{1g},\ldots,D_{5g})]\\
D_{varB} &= [var(D_{1b},\ldots,D_{5b})]\ D_{kurtB} = [kurt(D_{1b},\ldots,D_{5b})]
\end{aligned}
\tag{3.3.1}
$$

$$
\begin{aligned}
DC &= [DC_R, DC_G, DC_B]\\
H &= [H_{varR}, H_{kurtR}, H_{varG}, H_{kurtG}, H_{varB}, H_{kurtB}]\\
V &= [V_{varR}, V_{kurtR}, V_{varG}, V_{kurtG}, V_{varB}, V_{kurtB}]\\
D &= [D_{varR}, D_{kurtR}, D_{varG}, D_{kurtG}, D_{varB}, D_{kurtB}]
\end{aligned}
\tag{3.3.2}
$$

Then as in the case of the so-called dominant features, multiple BOVW streams are implemented for forming final image representation. As before in feature coding step hard assignment 3.1.1 is used and average pooling 3.1.2 is used in aggregation step.

$$\Phi_{DC} = f(DC, \mathcal{M}_{DC})\ \Psi_{DC} = g(\Phi_{DC}) \tag{3.3.3}$$

$$\Phi_H = f(H, \mathcal{M}_H)\ \Psi_H = g(\Phi_H) \tag{3.3.4}$$

$$\Phi_V = f(V, \mathcal{M}_V)\ \Psi_V = g(\Phi_V) \tag{3.3.5}$$

$$\Phi_D = f(D, \mathcal{M}_D)\ \Psi_D = g(\Phi_D) \tag{3.3.6}$$

$$\widehat{\boldsymbol{h}} = \Big[\Psi_{DC}, \Psi_H, \Psi_V, \Psi_D\Big] \tag{3.3.7}$$

3.4 Spatial Pyramid Matching for Local JPEG-based Features

Main problem of BOVW representation is that the spatial distribution of visual atoms is not considered. This is could be a problem when different images with similar words, but with different distribution, are compared. In computer vision and pattern recognition literature many approaches were proposed and one of the first effective attempt that were proposed was Spatial Pyramid Matching(SPM) [66]. Then instead of integrating spatial information in the low-level image descriptors, by considering for example geometric clues around low-level features, is considered at comparison level, by dividing the image in spatial bins. Foundamental idea behind SPM is to partition the image into a sequence of increasingly coarser grids and then compute a weighted sum over the distances of aggregated representations.

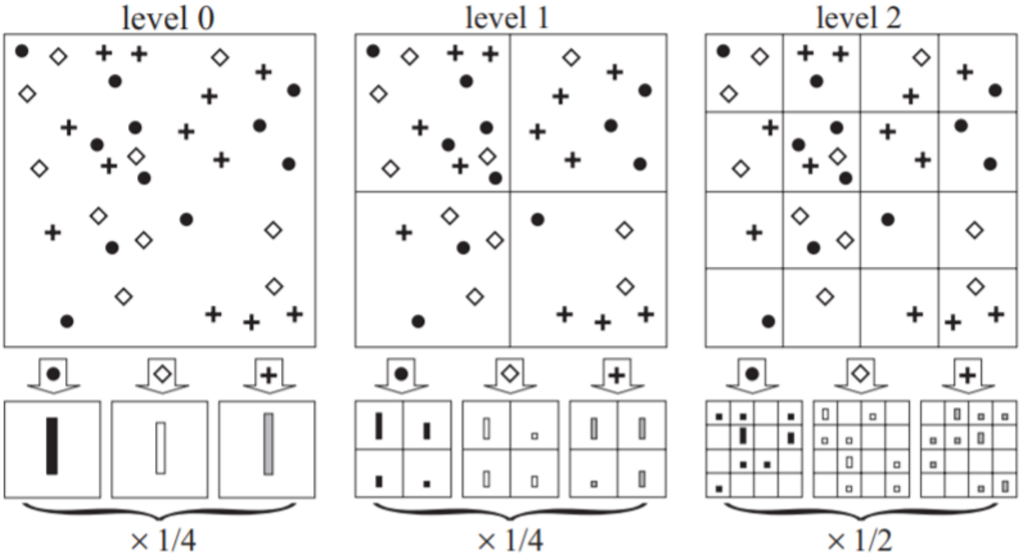


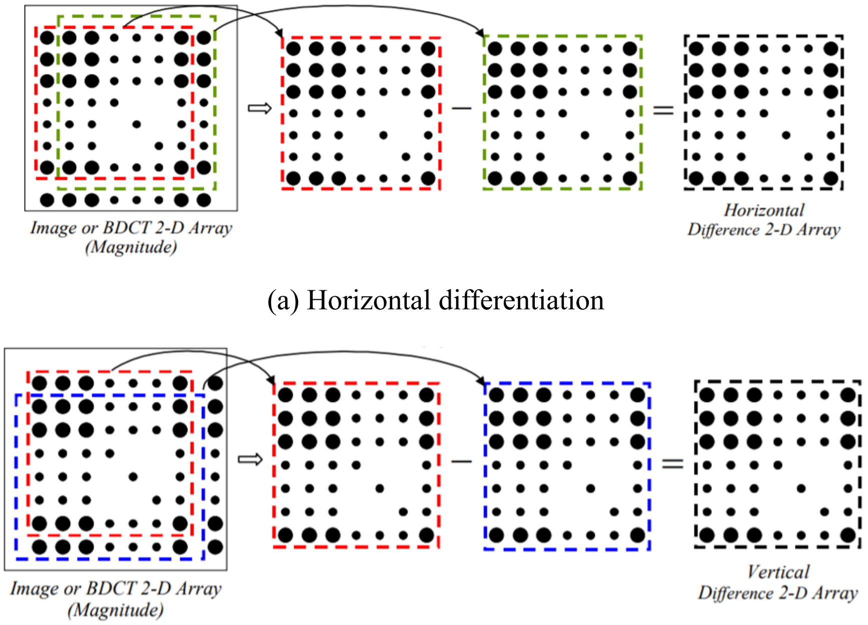
Figure 3.4: Example of spatial binning with relative weight. Adapted from ”Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories”, by S. Lazebnik, C. Schmid and J. Ponce,2006, *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, pp. 2169-2178

Spatial pyramid matching is consider an absolute spatial arrangement of the visual words, because two visual words of different images match only if they fall in the same spatial bin. Specifically suppose to partition image into a sequence of spatial grids at resolutions $0, \ldots, L$ such that the grid at level l is comprehensive of 4^l spatial bins. Then from each spatial bin an aggregation step is performed and compared with the aggragated features form of the same spatial bin of the other image. Finally this comparison is weighted in function of the resolution level l, precisely is equal to $\frac{1}{2^{L-l}}$, which is inversely proportional to the cell size and thus penalizes matches found in larger cells. The main problem of SPM is the features dimensionality, in fact all spatial bins of a resolution grids are concatenated together, to be precise for a dictionary of length M and L resolution level, final histogram representation has a length equal to $M \sum_{l=0}^{M} 4^l$.

3.5 Markov features

Always following the idea to include spatial information, in the present work has been studied the possibility to extract a statistical model from image regions and encode statistical description in terms of BOW representation. An approach to extract statistics behaviour from DCT coefficients image is the so-called Markov features [26, 79]. Aim of

Markov features is to capture the correlation among close DCT coefficients belonging to the same block and also relationships between coefficients wit hthe same frequency but belonging to adjacent JPEG blocks. Basically coefficients relationships are modelled through a Markovian process, that is defined by the so-called empirical transition probability matrix. A transition matrix element enclose the information about the probability that certain coefficient magnitude is surrounded to a coefficient with some magnitude. Obviously if DCT coefficients are taken as they are value range is huge, then in order to cover each transition case a $N \times N$ matrix should be use, in which N is equal to the number of possible values that can assume each coefficient. Therefore before to starting to extract transition matrix is necessary to derive a residual DCT coefficients image. To derive residual image is necessary to exploit locality principle, that states that adjacent element in images are correlated in some measure, so differentiation between close coefficients in some direction has to take in place.



(a) Horizontal differentiation

(b) Vertical differentiation

Figure 3.5: Derivation of residual DCT coefficients image. Adapted from ”Digital image splicing detection based on Markov features in DCT and DWT domain”, by Zhongwei He, Wei Lu, Wei Sun, Jiwu Huang, 2012,*Pattern Recognition*, Volume 45, Issue 12, 4292-4299

In present work residual images have been derived by taking differentiation in horizontal and vertical direction, either for capture intra-block and inter-block correlation. Mathematically speaking intra-block and inter-block difference array are derived as follow:

$$\begin{aligned} R_{intra_h}(u,v) &= F(u,v) - F(u,v+1) \quad & R_{inter_h}(u,v) &= F(u,v) - F(u,v+8) \\ R_{intra_v}(u,v) &= F(u,v) - F(u+1,v) \quad & R_{inter_v}(u,v) &= F(u,v) - F(u+8,v) \end{aligned} \tag{3.5.1}$$

Then the transition matrix calculation takes place, but before starting with this step is necessary to limit the residual values, because it was proven that most of remaining magnitudes remains inside a limited collection of values. Therefore a threshold T is introduced:

$$R_* = \begin{cases} T & \text{if } R_* > T \\ -T & \text{if } R_* < -T \end{cases} \tag{3.5.2}$$

Consequently transition matrix has cardinality equal to $|\mathcal{T}| = (2T+1) \times (2T+1)$. From each residual image R_* two transition probability matrix are extracted, one for horizontal direction transition and one for vertical direction transition. Following is illustrated the derivation of probability transitions:

$$\mathcal{T}_{intra_{hh}}(i,j) = \frac{\sum_{u=1}^{S_u} \sum_{v=1}^{S_v-2} \delta(R_{intra_h}(u,v) = i, R_{intra_h}(u,v+1) = j)}{\delta(R_{intra_h}(u,v) = i)} \tag{3.5.3}$$

$$\mathcal{T}_{intra_{hv}}(i,j) = \frac{\sum_{u=1}^{S_u-1} \sum_{v=1}^{S_v-1} \delta(R_{intra_h}(u,v) = i, R_{intra_h}(u+1,v) = j)}{\delta(R_{intra_h}(u,v) = i)} \tag{3.5.4}$$

$$\mathcal{T}_{intra_{vh}}(i,j) = \frac{\sum_{u=1}^{S_u-1} \sum_{v=1}^{S_v-1} \delta(R_{intra_v}(u,v) = i, R_{intra_v}(u,v+1) = j)}{\delta(R_{intra_v}(u,v) = i)} \tag{3.5.5}$$

$$\mathcal{T}_{intra_{vv}}(i,j) = \frac{\sum_{u=1}^{S_u-2} \sum_{v=1}^{S_v} \delta(R_{intra_v}(u,v) = i, R_{intra_v}(u+1,v) = j)}{\delta(R_{intra_v}(u,v) = i)} \tag{3.5.6}$$

In which S_u and S_v are respectively number of row and number of column of the original DCT coefficients image and $\delta(\bullet)$ is dirac delta operation, then $\delta(\bullet) = 1$ if and only if arguments are satisfied, otherwise $\delta(\bullet) = 0$.

$$\mathcal{T}_{inter_{hh}}(i,j) = \frac{\sum_{u=1}^{S_u} \sum_{v=1}^{S_v-16} \delta(R_{inter_h}(u,v) = i, R_{inter_h}(u,v+8) = j)}{\delta(R_{inter_h}(u,v) = i)} \tag{3.5.7}$$

$$\mathcal{T}_{inter_{hv}}(i,j) = \frac{\sum_{u=1}^{S_u-8} \sum_{v=1}^{S_v-8} \delta(R_{inter_h}(u,v) = i, R_{inter_h}(u+8,v) = j)}{\delta(R_{inter_h}(u,v) = i)} \tag{3.5.8}$$

$$\mathcal{T}_{inter_{vh}}(i,j) = \frac{\sum_{u=1}^{S_u-8} \sum_{v=1}^{S_v-8} \delta(R_{inter_v}(u,v) = i, R_{inter_v}(u,v+8) = j)}{\delta(R_{inter_v}(u,v) = i)} \tag{3.5.9}$$

$$\mathcal{T}_{inter_{vv}}(i,j) = \frac{\sum_{u=1}^{S_u-16} \sum_{v=1}^{S_v} \delta(R_{inter_v}(u,v) = i, R_{inter_v}(u+8,v) = j)}{\delta(R_{inter_v}(u,v) = i)} \tag{3.5.10}$$

As before S_u, S_v and $\delta(\bullet)$ have the same meaning.Then from each investigated area four transition matrix $\mathcal{T}$ are extracted for describing intra-block correlation and other four transition matrix $\mathcal{T}$ are extracted for describing inter-block relationships. Final feature vector are the vectorialized transition matrix. Usually this kind of features are extracted for representing the whole images, but here are extracted from partial overlapped image patches in order to characterize a group of DCT blocks. Therefore at the end an image will be represented by a number of intra-block and inter-block correlation vectors coming from each image patches of every color band, in that case RGB. Once all features are extracted two dictionary are constructed, one for intra-block transition probability vectors and the other for inter-block features. While the aggregation step is done independetly for each color band. Below mathematical formalization is given:

$$\begin{aligned} \Phi_{R_{intra*}} &= f(R_{intra*}, \mathcal{M}_{R_{intra*}}) \\ \Phi_{R_{inter*}} &= f(R_{inter*}, \mathcal{M}_{R_{inter*}}) \end{aligned} \tag{3.5.11}$$

$$\Psi_{R_{intraR}} = g(\Phi_{R_{intra*R}})\ \Psi_{R_{interR}} = g(\Phi_{R_{inter*R}}) \tag{3.5.12}$$

$$\Psi_{R_{intraG}} = g(\Phi_{R_{intra*G}})\ \Psi_{R_{interG}} = g(\Phi_{R_{inter*G}}) \tag{3.5.13}$$

$$\Psi_{R_{intraB}} = g(\Phi_{R_{intra*B}})\ \Psi_{R_{interB}} = g(\Phi_{R_{inter*B}}) \tag{3.5.14}$$

$$\widehat{\boldsymbol{h}} = \left[\Psi_{R_{intraR}}, \Psi_{R_{interR}}, \Psi_{R_{intraG}}, \Psi_{R_{interG}}, \Psi_{R_{intraB}}, \Psi_{R_{interB}}\right] \tag{3.5.15}$$

Actually more complex relationships coming from different direction at different distance or even correlations between coefficients of different band could be used but the feature extraction time would grow too much, without adding noticeable benefits to the retrieving results.

Chapter 4

Proposed Query Sensitive Feature Weighting Algorithm in JPEG Compressed Domain

4.1 General Architecture

So far, the features have been fused at mid-level representation, i.e. at the level of the BOVW representation, and then comparing without being weighed, therefore without valorizing the information provided by each features. It is therefore natural to emphasize the features in function of the visual clues of the query image and then fuse this at the level of this new features representation. To do this an MKL approach is applied, precisely for each features is associated to a basis kernel and then a weight is used to valorizing the visual importance of each features in the final image representation. Then a substantial change in the retrieving architecture is applied. First big difference respect proposed unsupervised retrieval system 3 is the shift to a supervised method, in this case the natural choice for supervised kernel method to be used falls on support vector machine(SVM). Second big difference respect before is that retrieval image problem becomes a binary problem, in which the retrieved images by the system are those more distant from the hyperplane. It is modelled as a binary problem in the sense that images that are on the same side of the hyperplane of query image are considered retrieved image, while the other are discarded. A rough viaul representation of the system is given in figure 4.1

4.2 Proposed Feature Weighting Method

Consider to have a set of histogram-based fearures f so that $f_i \in F$, the aim of this new architecture is to valorize the importance of each feature respect the content of image, so assuming to have a step in which a MKL algorithm is involved to learn the weight to assign to each image signature, then feature weighting problem become the follow:

$$\boldsymbol{K}_c = \sum_{m=1}^{M} d_m \boldsymbol{K}_m \tag{4.2.1}$$

In which m is the m-th selected histogram based features associate to a specific kind of features and $\boldsymbol{K}$ is the used kernel basis. To make sure that the retrieving algorithm prioritizes the most relevant visual clues of the query image during the relevant image recovery process, the query image must be inserted into the training set used to find the optimal set of weights. Therefore when the query image I_q is selected, a training set including the I_q is defined. In this thesis a classifier-based MKL algorithm is used, because they generally approach best results respect other algorithms, bbut the proposed architecture is not constrained by the chosen MKL algorithm. In present work the so-called generalized MKL(GMKL) proposed in [72] is used. GMKL is also used because results to be more flexible respect to other classifier-based algorithm, e.g. SimpleMKL [80]. Given that histogram-based image representations are used the natural choice for the basis kernel is the χ^2 kernel, that is shown below.

$$k_{\chi^2}(x, y) = 2 \sum_{i=1}^{N} \frac{x_i y_i}{x_i + y_i} \tag{4.2.2}$$

GMKL algorithm is a classifier-dependent MKL, that is the family of MKL algorithms that jointly learn the classifier paramenters and the kernel weights by minimizing the SVM classifier error. Let $T\{\boldsymbol{x}_i, y_i\}_{i=1}^{n}$ be the training set amde up of n labeled featuressamples, where $\boldsymbol{x}_i \in \mathbb{R}^D$ is the i-th sample associated with the binary class label $y_i \in \{+1, -1\}$. From this labeled samples, M basis kernels $\{\boldsymbol{K}_1, \boldsymbol{K}_2, \ldots, \boldsymbol{K}_M\}$ are constructed. Each basis kernel is associated to the so-called reproducing kernel Hilbert space(RKHHS) and is specified with $\mathcal{H}_m$. Then in to minimizing SVM classification error, classifier-dependent MKL adopts objective function of the SVM trained usign the composite kernel on the RKHS of $\mathcal{H}_c$ as the target function. Then the primal problem of the MKL is written as follows:

$$\underset{\boldsymbol{w}, b, \xi_i, \boldsymbol{d}}{\operatorname{argmin}} \left(\frac{1}{2} \|\boldsymbol{w}\|^2_{\mathcal{H}_c} + C \sum_{i=1}^{n} \xi_i \right) \tag{4.2.3}$$

$$s.t. : y_i \left(\sum_{m=1}^{M} \sqrt{d_m} \boldsymbol{w}_m^T \phi_m(\boldsymbol{x}_i) + b \right) \geq 1 - \xi_i \; \forall i = 1, \ldots, n \tag{4.2.4}$$

$$\xi_i \geq 0 \forall i = 1, \ldots, n \tag{4.2.5}$$

$$\boldsymbol{d} \in \Delta \tag{4.2.6}$$

Where Δ is the norm regularization of kernel weights and can be $\Delta 1$, $\Delta 2$ or Δp. C is a positive regularization parameter that controls the tradeoff between generalization of the classifier and the training error ξ_i and b is the bias term. This problem can be resolved by substituting $\boldsymbol{w}_m$ with $\boldsymbol{w}_m/\sqrt{d_m}$, then by following [80] a min-max problem is taken as dual problem. In GMKL algorithm, which is part of the wrapper algorithm family, an added regularization term $\Omega(\boldsymbol{d})$ makes the dual problem become as follows:

$$\underset{\boldsymbol{d}}{\text{argmin}}_{\boldsymbol{\alpha}} \left(1^T \boldsymbol{\alpha} - \frac{1}{2} (\boldsymbol{\alpha} \circ \boldsymbol{y})^T \boldsymbol{K}_c(\boldsymbol{d}) (\boldsymbol{\alpha} \circ \boldsymbol{y}) + \Omega(\boldsymbol{d}) \right) \tag{4.2.7}$$

$$s.t. : y_i \sum_{i=1}^{n} \alpha_i y_i = 0 \tag{4.2.8}$$

$$0 \leq \alpha_i \leq C \forall i = 1, \ldots, n \tag{4.2.9}$$

$$d_i \geq 0 \forall i = 1, \ldots, M \tag{4.2.10}$$

Assuming that the combination function and the regularization term are differentiable functions of the kernel weights, gradient desccendant method can be used to obtain the oprimal kernel weights. After that features are weighted and combined a final representation of the query image is embedded in the final composite kernel. At that point learned weighted composite kernel $\boldsymbol{K}_c$ is used in conjunction with an SVM, that through its discrimant function the system is able to divides archive image set in retrieved images, that are the images that have similar signature to the query, and discarted images, that are the images not similar to the query. At the end k most distant images from the SVM hyperplane are selected as retrieved image.

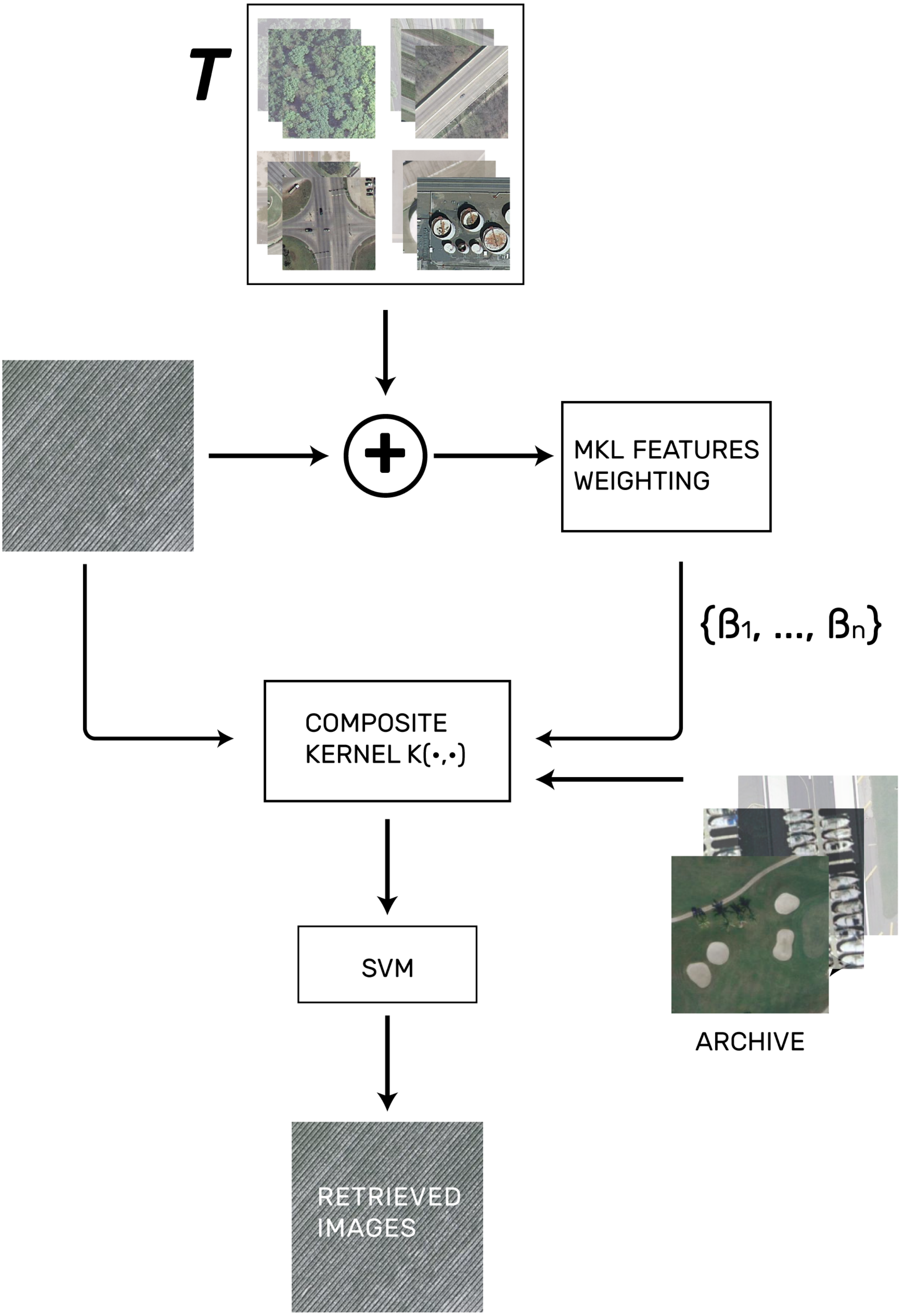


Figure 4.1: Proposed Query Sensitive Feature Weighting architecture

Chapter 5

Experimental Results

5.1 Dataset Description and Design of Experiments

To evaluate retrieving capabilities of proposed features, experiments were performed on the UC MERCED LULC dataset, proposed in [81]. Archive contains 2100 images manually extracted from the USGS National Map Urban Area Imagery collection, with size of 256x256 pixels and a spatial resolution of 1 feet, that is approximately 30.48 cm. Original archive were subdivided in 21 categories, that are: agricultural, airplane, baseball diamond, beach,buildings, chaparral, dense residential, forest, freeway, golf course, harbor, intersection,medium residential, mobile home park, overpass, parking lot, river, runway, sparse residential, storage tanks and tennis court. It is clear that this archive has two limitations to take into account in case of testing a CBIR system, first one is about archive's size, in fact in real applications more images are used. Thus, the impact of the results presented should be properly rescaled when larger archives are considered. Second limitation concerns the use of single-label annotation. Therefore in this work, to resolve this second limitation of the archive, were decided to follow the multi-label extention of [82]. Multi-label extention is mandatory to make reliable performance measurements of a CBIR system, in fact aim of retrieval system is to recover images that share visual clues with the query images, therefore this approach suite better the CBIR paradigm. The total number of considered multi-labels is 17 and they are airplane, bare soil, buildings, cars, chaparral, court, dock, field, grass, mobile home, pavement, sand, sea, ship, tanks, trees and water. Always following [82], the labels associated with each image varies between 1 and 7. In table 5.1 are shown what primitive class is possible to find inside images of single-label annotated subset. Finally in figures 5.1 some sample from each single label category is shown. Before to start the experiments, all images in the archive are converted from the original *.tif* format to *.jpeg*, through standard $textMATLAB^{\circledR}$ *imwrite* function with a quality factor $Q = 100$. In order to apply the proposed architectures, each color band is compressed independently. As already mentioned all the experiments have been done in $textMATLAB^{\circledR}$ 2017b environment, conducted on a machine equipped with Windows

10 Pro 64-bit with regard the OS, while as regards hardware, testing machine is equipped with a processor $textIntel^{®}$Xeon$^{®}$ CPU E5-1650v2 with clock speed of 3.50GHz and a 16.0GB of memory. To extract DCT coefficients directly from the *.jpeg* compressed file JPEG Matlab toolbox created by [83] has been used. Finally the other external library used during the experiments of supervised proposed method is the famous LIBSVM created by [84].

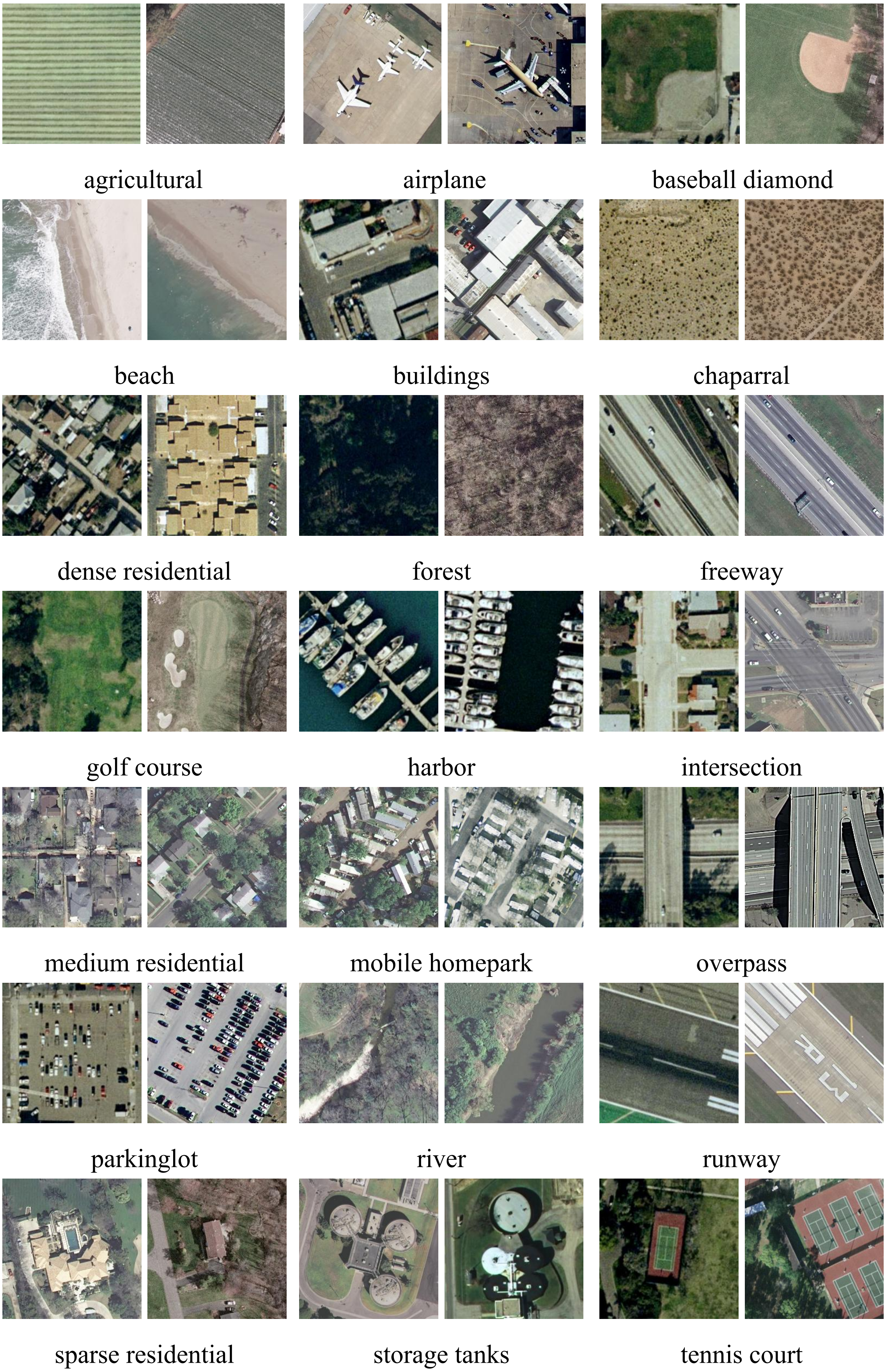
Table 5.1: Multi-labels distribution respect original annotation

Category labels	**Associated multi-labels**
agricultural	field, trees
airplane	airplane, pavement, grass, buildings, cars
baseball diamond	bare soil, pavement, grass, trees, buildings
beach	sea, sand, trees
buildings	buildings, pavement, trees, cars
chaparral	sand, chaparral
dense residential	buildings, pavement, trees, cars
forest	trees, bare soil
freeway	pavement, cars, grass, trees, bare soil
golf course	grass, trees, bare soil
harbor	ship, dock, water
intersection	pavement, bare soil, cars, buildings, grass
medium residential	buildings, cars, trees, grass, pavement, bare soil
mobile home park	mobile home, pavement, cars, trees, bare soil
overpass	pavement, cars, bare soil, grass, trees
parking lot	cars, pavement, bare soil, grass
river	water, trees, bare soil
runway	pavement, grass, bare soil
sparse residential	buildings, grass, bare soil, trees, sand, chaparral
storage tanks	tanks, bare soil, grass, pavement, buildings
tennis court	court, grass, trees, pavement, buildings

In table below 5.2 shows distribution, in terms of number of images, of each primitive class.

Table 5.2: Number of images associated to each promitive label

Category labels	**Number of image**
airplane	100
bare soil	627
buildings	695
cars	884
chaparral	118
court	105
dock	100
field	106
grass	978
mobile home	102
pavement	1302
sand	389
sea	100
ship	102
tanks	100
trees	1015
water	203



agricultural airplane baseball diamond

beach buildings chaparral

dense residential forest freeway

golf course harbor intersection

medium residential mobile homepark overpass

parkinglot river runway

sparse residential storage tanks tennis court

Figure 5.1: Some examples with original annotation

In the first series of experiments proposed features were tested on a CBIR system equipped with a k-NN module [85] for retrieving the most similar images for the system perspective.

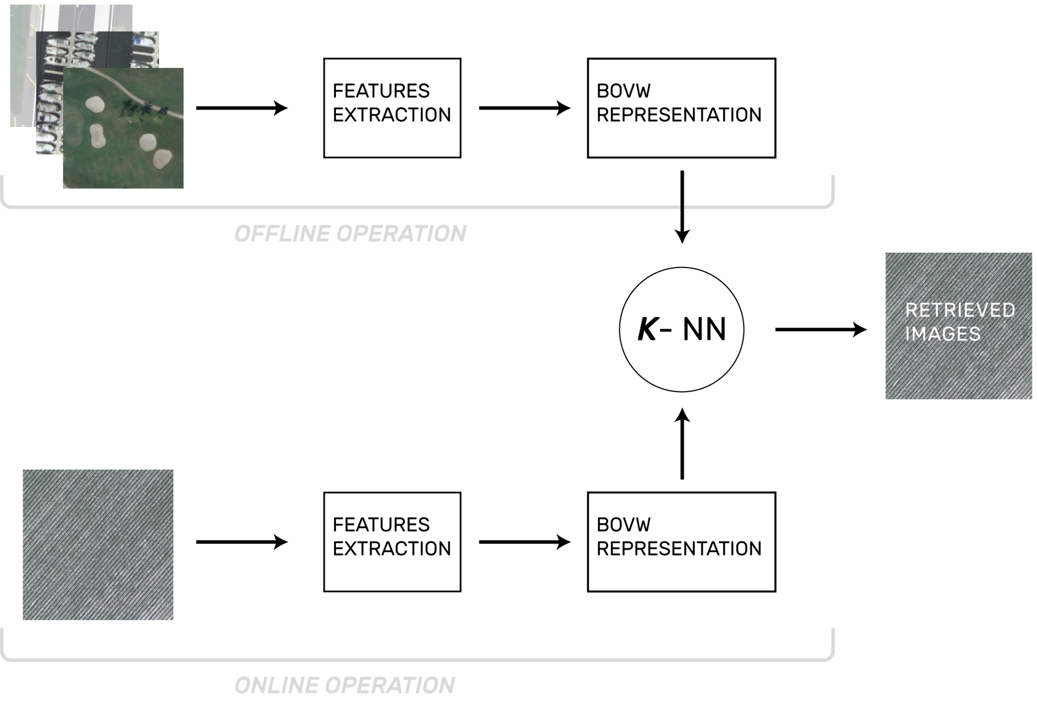


Figure 5.2: Illustration of used CBIR system

As explained before, in this thesis a multi-label approach has been adopted, then the metrics used to evaluate the performances of proposed method is the average recall. In order to define the recall, let $\boldsymbol{L}_r \subset \boldsymbol{L}$ be the set of class labels present in the retrieved image $\boldsymbol{X}_r \in \boldsymbol{X}$ retrieved, where $\boldsymbol{X}$ retrieved is the subset of retrieved images. Similarly, let $\boldsymbol{L}_q \subset \boldsymbol{L}$ be the set of class labels present in $\boldsymbol{X}_q$. Recall is defined as follows:

$$Recall = \frac{1}{|\boldsymbol{X}^{retrieved}|} \sum_{r=1}^{|\boldsymbol{X}^{retrieved}|} \frac{|\boldsymbol{L}_q \cap \boldsymbol{L}_r|}{|\boldsymbol{L}_q|} \tag{5.1.1}$$

In which operator $|\bullet|$ stands for set cardinality. Finally overall average recall is given by averaging mean recall of all query images. For retrieving similar images from the archive is necessary to define a similarity or distance measure, in this thesis χ^2-distance is used, because is one of standard distance measure used to do histogram comparisons.

$$\chi^2(x, y) = \sum_{i=1}^{N} \frac{(x_i - y_i)^2}{x_i + y_i} \tag{5.1.2}$$

In which x and y are the image signature of query image and retrieving image. It is worth of noticing that χ^2-distance is *bin-to-bin* distance measure, then correlations among bins have not been take in consideration. In this series of experiments it was studied the sensitivity of retrieving performance to the dictionary-size. Has already explained in 3 final image signature is given by a multi-dictionary BOVW representation, then following dictionary-lengths refer to the size of a single dictionary and they are 50,100,200,300,400,500,600,1000. In all experiments 20 images are retrieved from the archive.

While in the second part of the experiments the proposed supervised retrieving method is used 4. Then in this series of tests there is a substantial change of methodology for image retrieval. First big difference respect other experiments is the used of a supervised method, as already explained an SVM approach is used. Second big differece respect first part of experiments is the shift from multi-label to single-label paradigm, then from here onwards original label set is considered. In these experiments following version of χ^2 kernel is used:

$$k_{\chi^2}(x, y) = 2\sum_{i=1}^{N} \frac{x_i y_i}{x_i + y_i} \tag{5.1.3}$$

Then for each features a χ^2 kernel is associated, since the tests were carried out on the dominant features 3.2 means that only four basis kernel are used.

$$\boldsymbol{K}_c = \beta_{DC} \cdot \boldsymbol{K}_{DC} + \beta_H \cdot \boldsymbol{K}_H + \beta_V \cdot \boldsymbol{K}_V + \beta_D \cdot \boldsymbol{K}_D \tag{5.1.4}$$

In order to implement the system for each single label class a training set T is formed offline, that means that 21 training sets are fixed from the beginning. Then query image is added to the training set belonging on the same class and the process of MKL training begins. At the end the 20 images more distant from the hyperplane are considered right retrieved images. In order to measure the effectiveness of the proposed architecture, recall measure has been used and to understand the behaviour of MKL various size of training set were tested. Then as before recall measure is defined as:

$$Recall = \frac{1}{|\boldsymbol{X}^{retrieved}|} \sum_{r=1}^{|\boldsymbol{X}^{retrieved}|} \frac{|\boldsymbol{L}_q \cap \boldsymbol{L}_r|}{|\boldsymbol{L}_q|} \tag{5.1.5}$$

But that time is refers to a single-label measure.

All test were conduct only on dominant features with a dictionary length of 100 and 20 images are retrieved in order to measure the system performances.

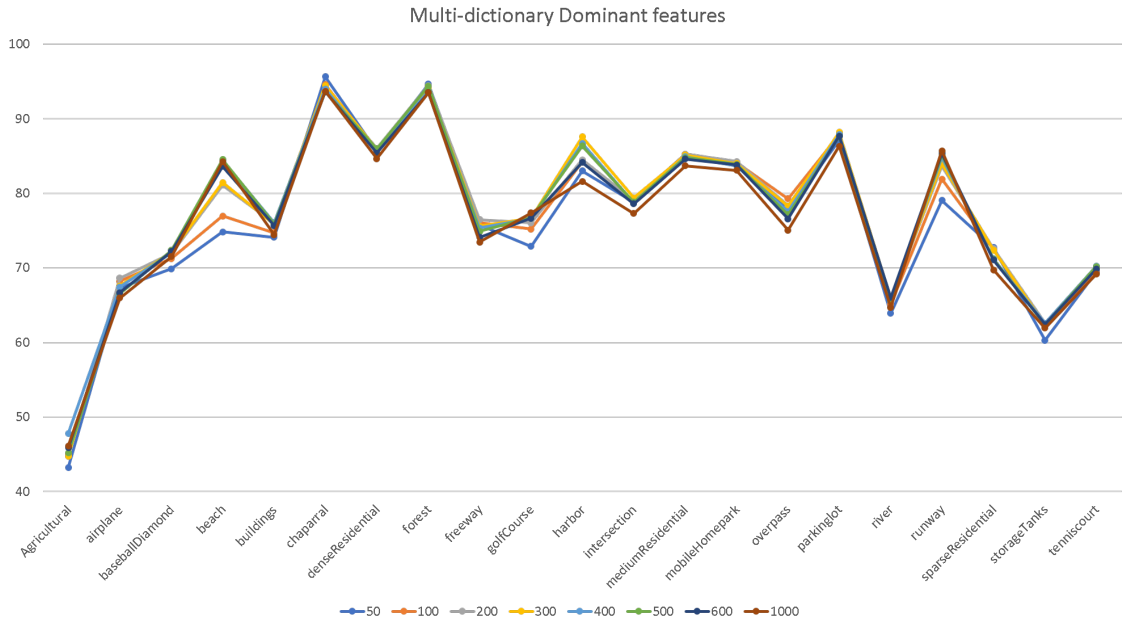


Figure 5.3: Multi-dictionary intra single label dominant features average recall

5.2 Experimental Results of the Considered Unsupervised Descriptors

5.2.1 Results of Dominant Features

In this kind of features four dictionary are used, that means that the total length of image signature is four times dictionar-size. In graph below 5.6 is depicted the intra-single-class average recall in function of dictionary length.

That which is coming to light is that images under the single label agricultural have the worst results, while images such as chaparall and forest reach best retrieving percentage. This is related to the strong directional component of agricultural images, in fact the proposed features do not take into account the visual importance of one direction rather than another and therefore the retrieving method confuses with images with a high texture activity such as forest. Most of the time the confusion concerns only natural images because the color features, aka DC value, directs the retrieving to the images containing natural scenes. Below 5.5 some examples of single-label class buildings images retrieving is given and as before the images are confused with others with similar texture activity and similar range of colors. In figure 5.6 average multi-class recall is given in function to the dictionary size, as is shown multi-labels performance remains stable during the course of all experiments with this setting.

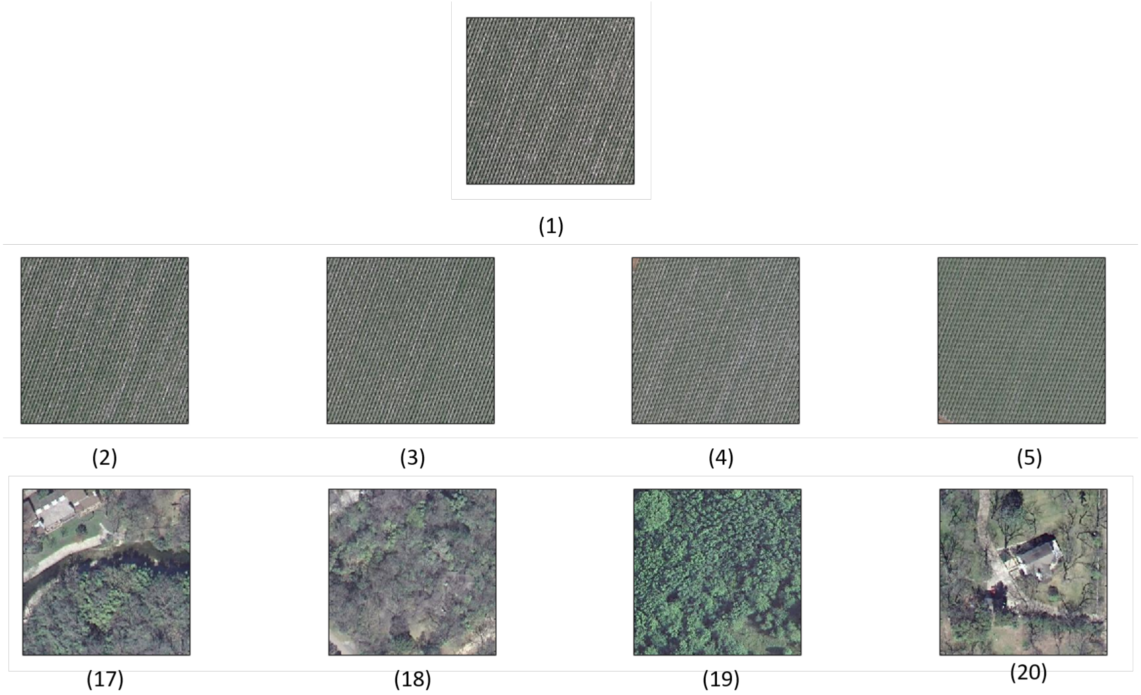


Figure 5.4: Example of retrieved agricultural images

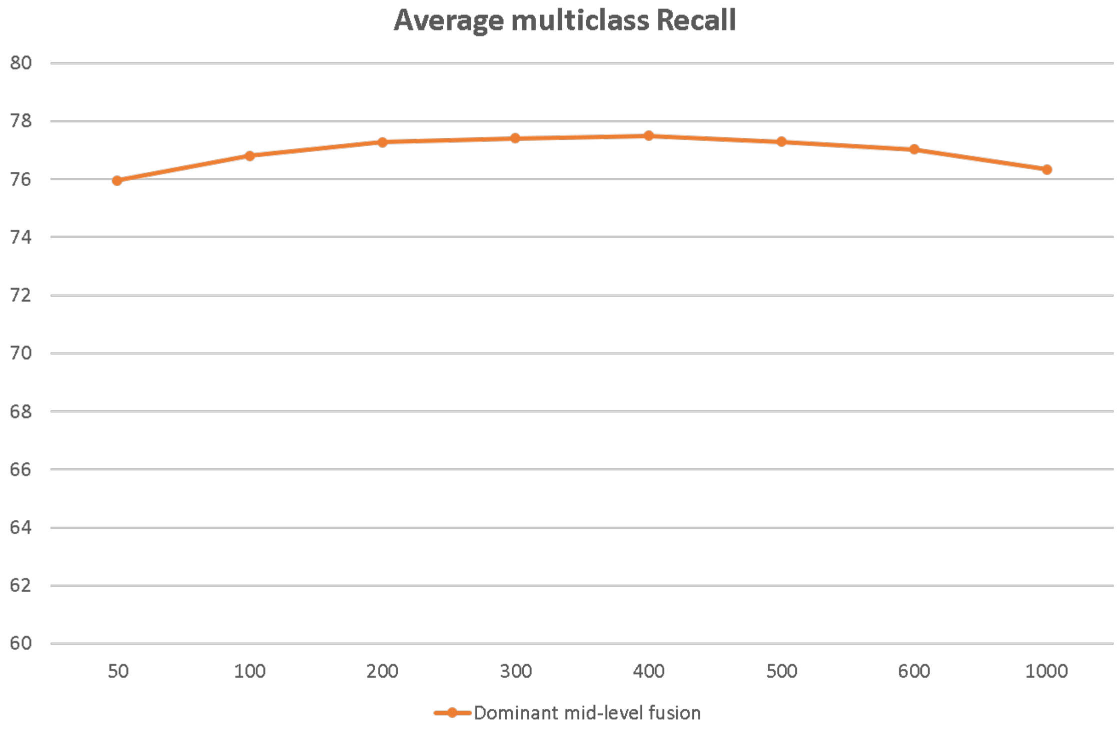


Figure 5.6: Dominant features multi-dictionary representation average recall

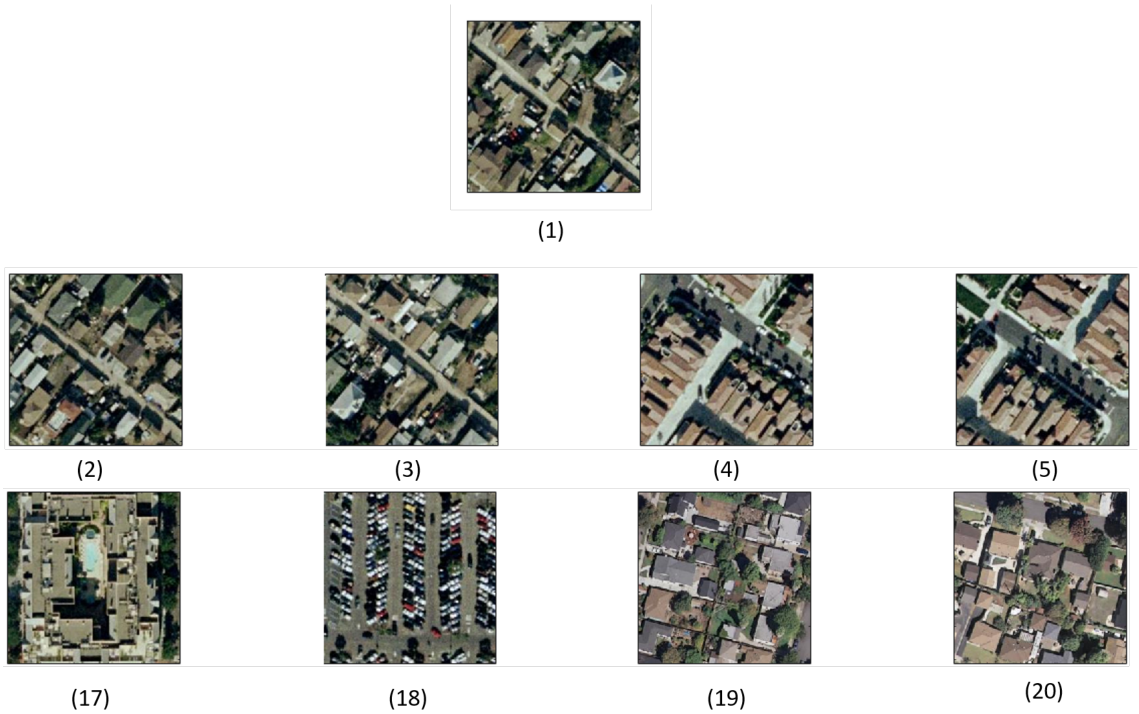


Figure 5.5: Example of retrieved buildings images

5.2.2 Results of Statistical Features

As in the previous experiments with dominant features four dictionaries are involved. In figure 5.7 is possible to see that agricultural single-label class still has the worst results, but that time beach single-label class average recall dramatically decrease. This has to do with the fact that the coefficients statistics suppress the frequency location information carried by individual coefficients, in fact it is impossible to determine a preferred single-label class of confusion, that means that confusion happens with images coming from very different classes. another reason is that the sea primitive class is only present in this kind of single-label class and the is not modelled properly during the clustering operation perform to create dictionaries.

But in general, as is shown in 5.8, in other cases retriving works quite well. In figure 5.10 the average measure of multi-class recall is presented, as it is straightforward to see, linear behaviour along the whole experiment can also be affirmed in this case, but differently from before the average is less than a few percentage points. The advantage of this feature compared to the previous one is that at low-level has a lower dimensionality, due to the fact is composed only by two statistics for each colo band and therefore the dictionaries can be built faster.

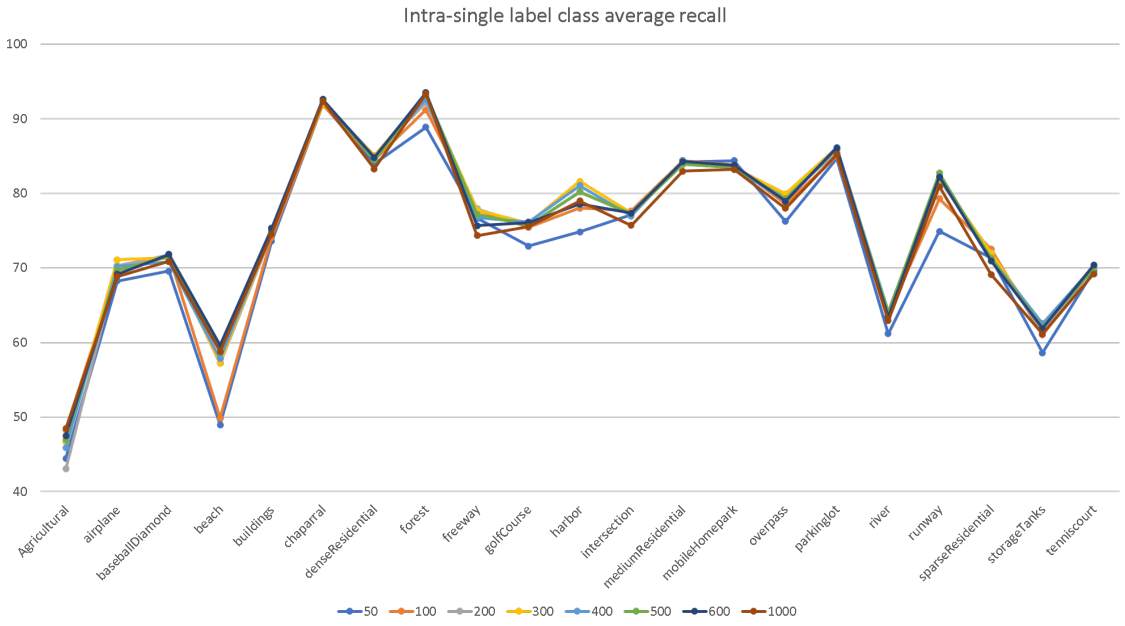


Figure 5.7: Multi-dictionary intra single label statistical features average recall

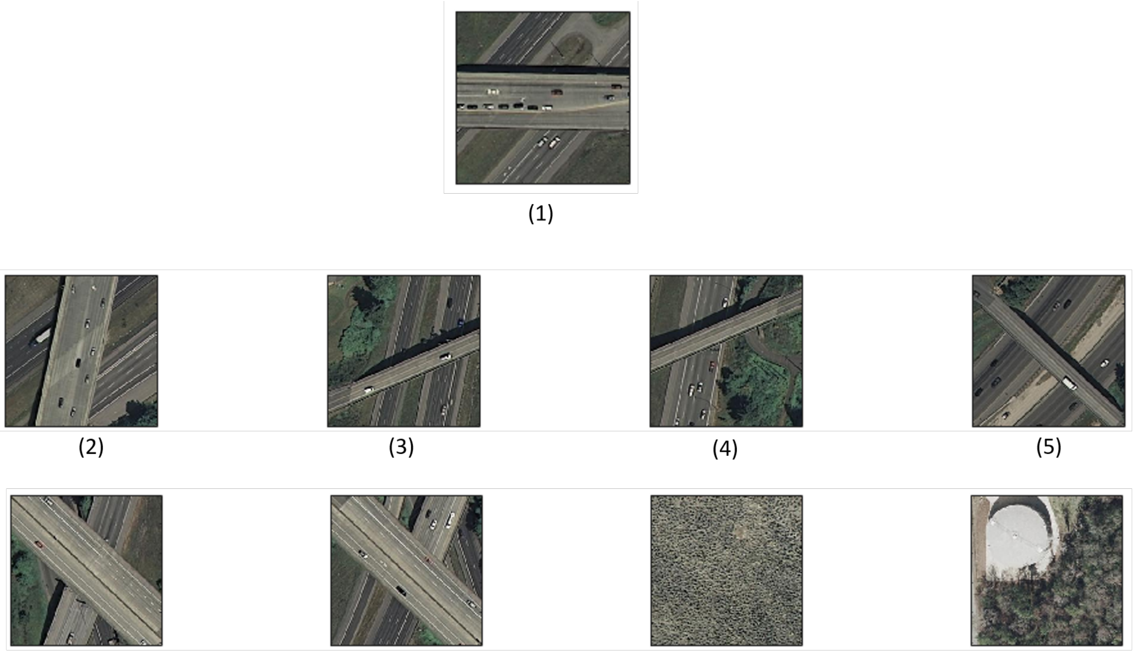


Figure 5.8: Example of retrieved overpass images

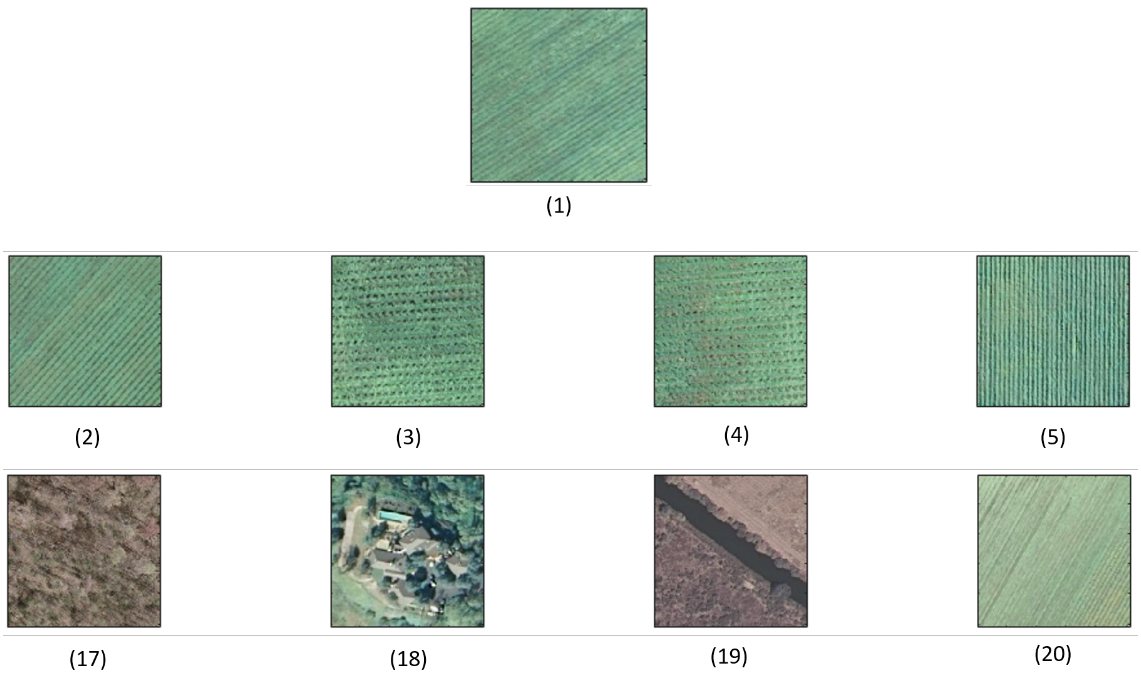


Figure 5.9: Example of retrieved agricultural images

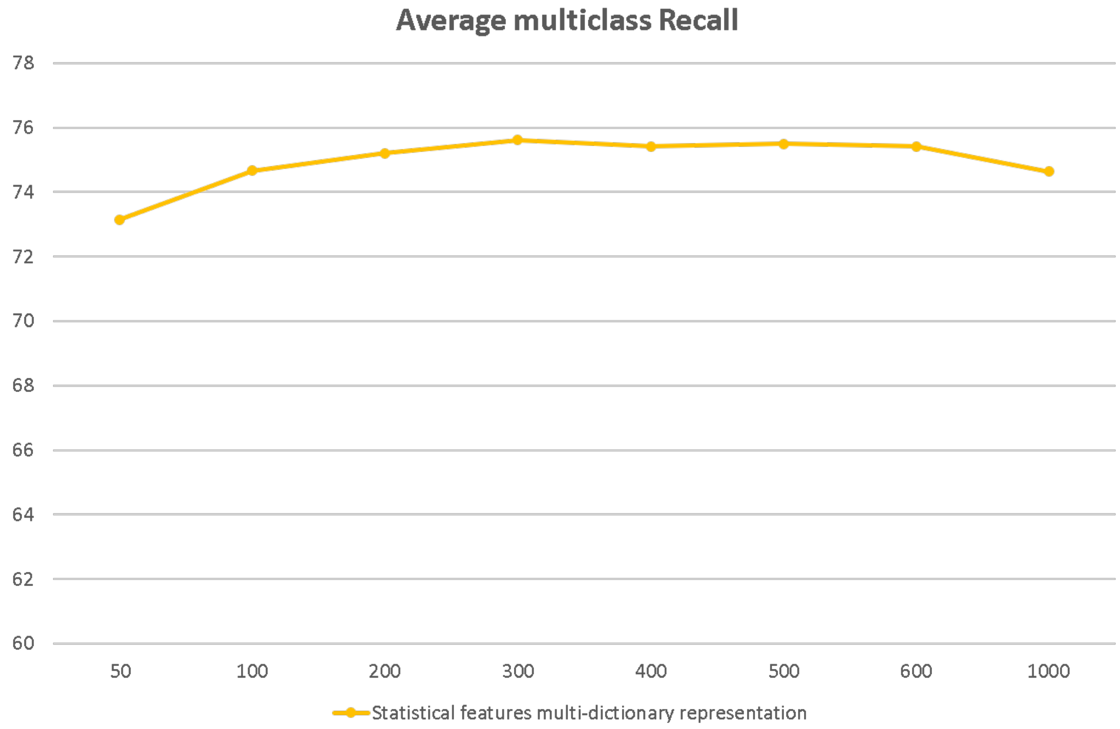


Figure 5.10: Statistical features multi-dictionary representation average recall

5.2.3 Results of Dominant Features with SPM Comparison

As already mentioned in 3.4, SPM is neither a features nor an alternative to BOVW representation, but is only a method to make a comparison between BOVW histogram-based image signatures. One of the main problems with SPM used in combination with *k*-NN is the so called curse of dimensionality, caused by the length of final representation obtained by comparing efatures with this algorithm. In fact with this testing architecture, results are evidently lower compared to the previous outcomes. It is worth of noticing that in 5.11 for some single-label classes there is an high sensibility to the dictionary size such as beach or buildings.Spatial pyramid matching is used only with dominant features, because they have proved to be more performing.

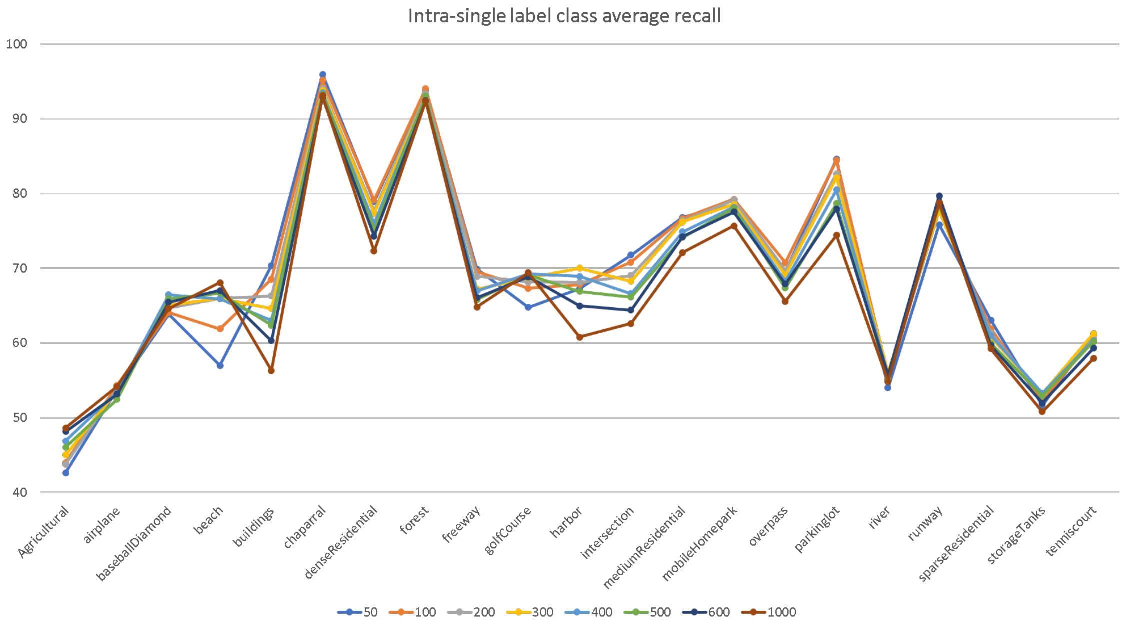
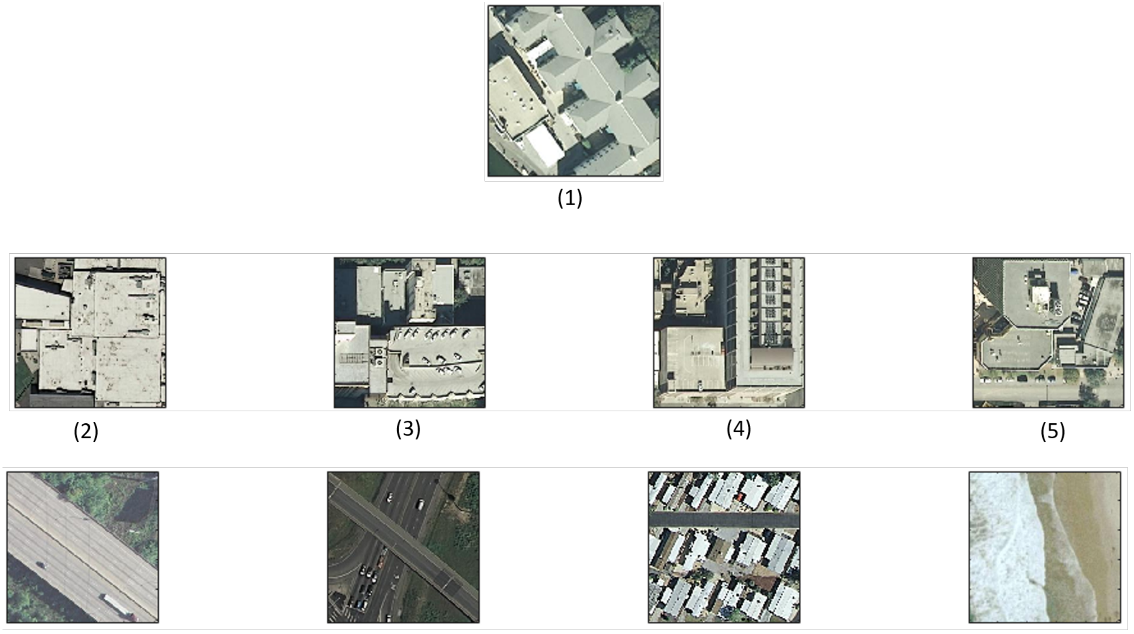


Figure 5.11: Multi-dictionary intra single label statistical features average recall

As shown in figures 5.13 retrieving of high textural natural images remains unchanged, while for other classes sometimes the confusion made in the first twenty retrieved images could be high as in 5.12. In figure 5.14 it is noticeable that as the dimensionality of the BOVW representations increases, the result decreases continuously and this is a clear effect of curse of dimensionality with a *k*-NN retrieval method.

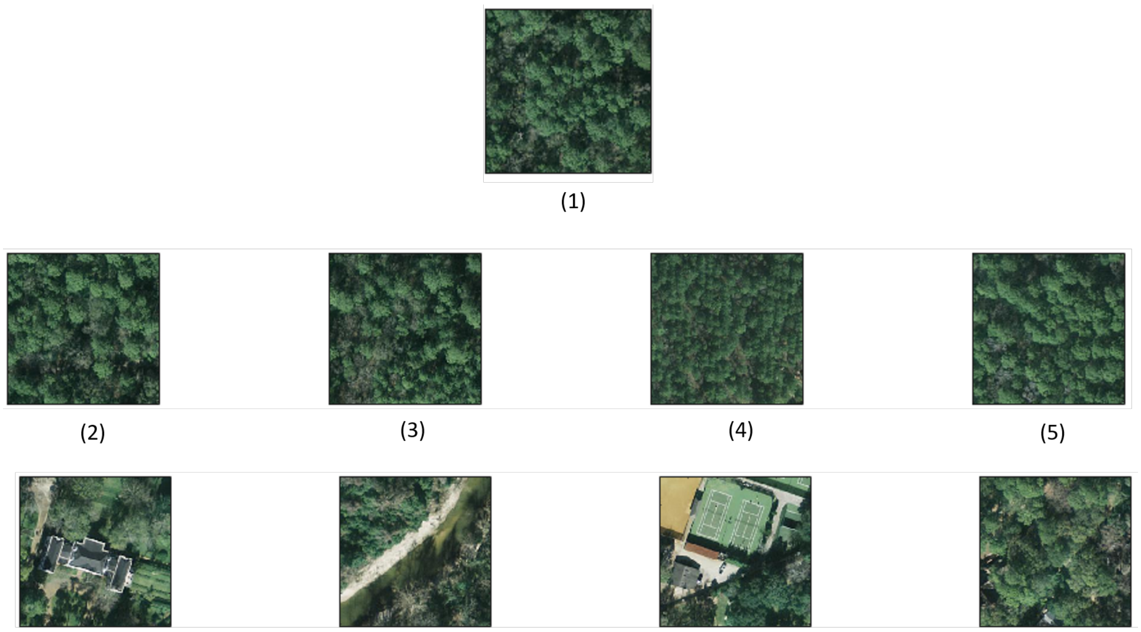


(1)

(2) (3) (4) (5)

(17) (18) (19) (20)

Figure 5.12: Example of retrieved buildings images



(1)

(2) (3) (4) (5)

(17) (18) (19) (20)

Figure 5.13: Example of retrieved forest images

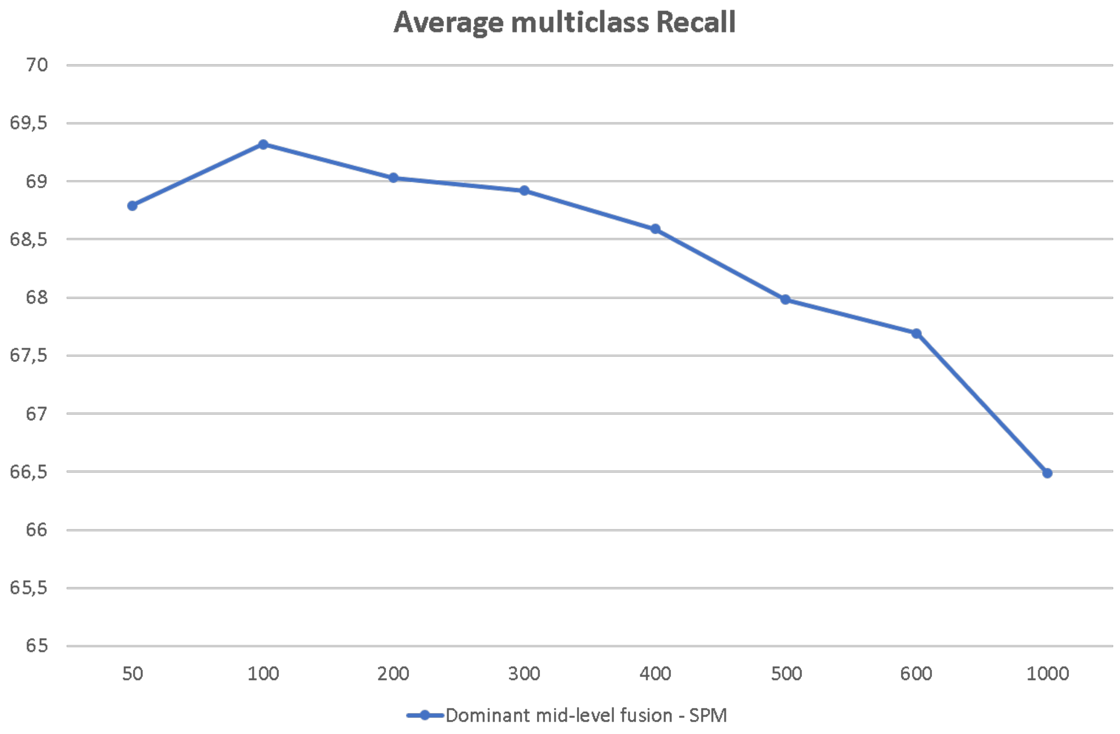


Figure 5.14: Average recall in function of dictionary length

5.2.4 Results of Markov Features

Also in that case a multi-dictionary approach is used, then final image representation is made by concatenating aggregated mid-level features coming from different band and different features. Then at the end 6 histogram-base representation are joined together. From figure 5.15 is possible that these features are not too sensitive to the different lengths of the dictionary, except for single-label class chaparall, which is very variable. In this case the worst results are held by singkle-label class river.

As is possible to see in figures 5.17 and 5.16 in average this features work quite well, but only if the dimensionality of the BOVW-representations remains limited. In fact in 5.18 is possible to see that after dictionary length 600 average recall start to decrease.

5.2.5 Comparison of Considered Features

This final section show all result depicted on the same graph at the same scale. As previously said dominat feature and statistical feature remain stable along all the experiments in function to different dictionary sizes, while the worst results are held by SPM and markov features. As closing note is possible to se that after a dictionary length of 600 SPM start tto decrease more fastly compared to Markov-base features.

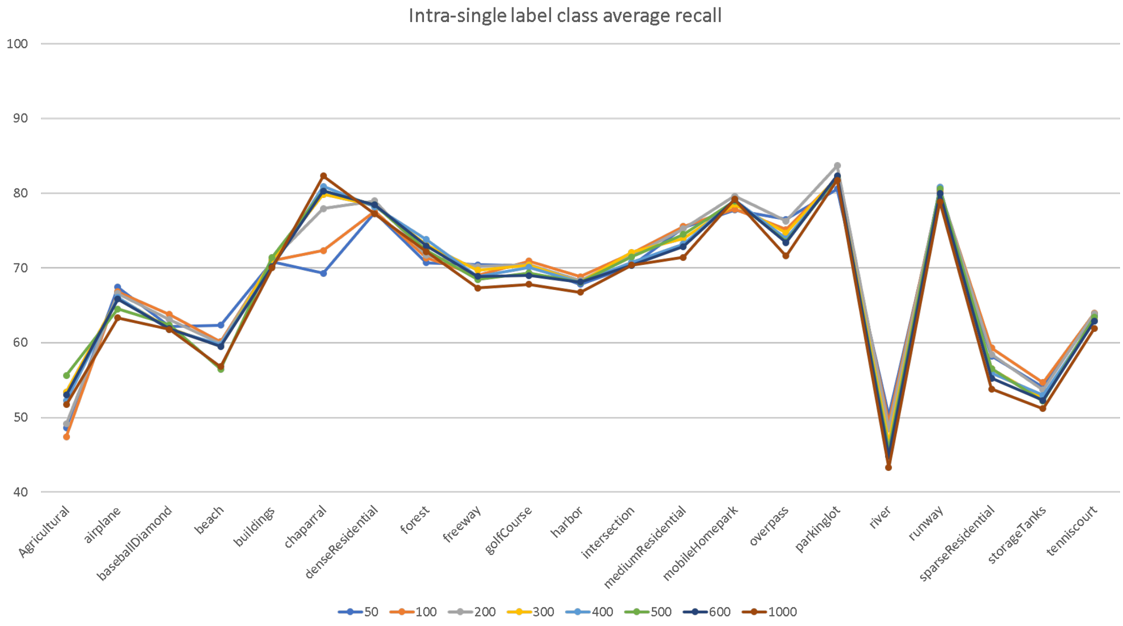
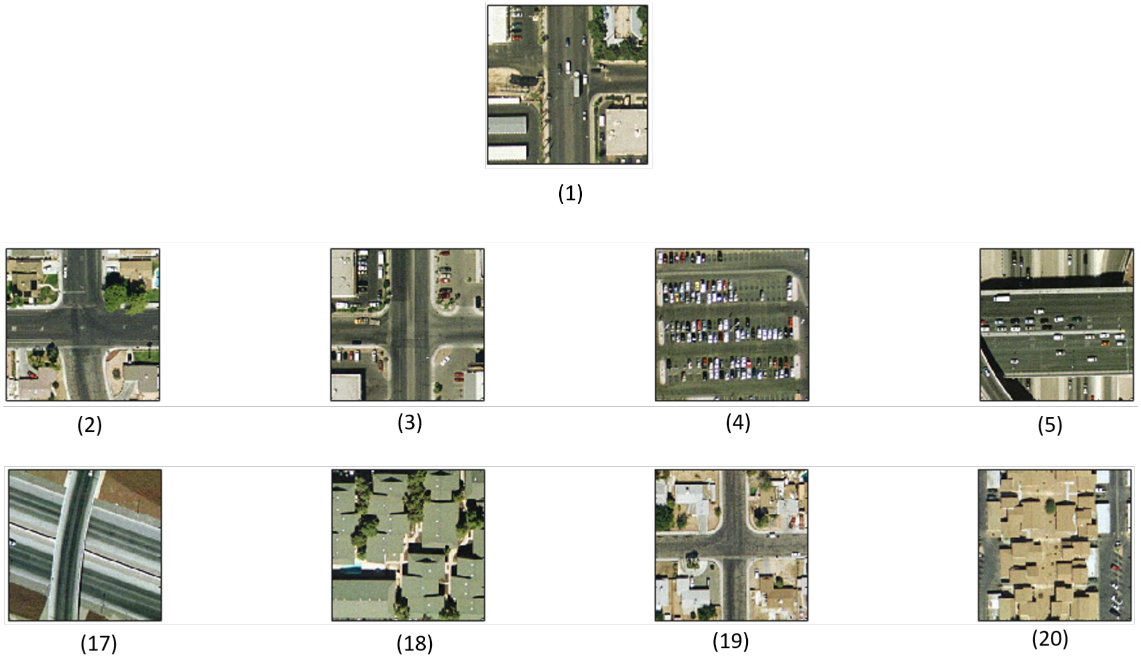


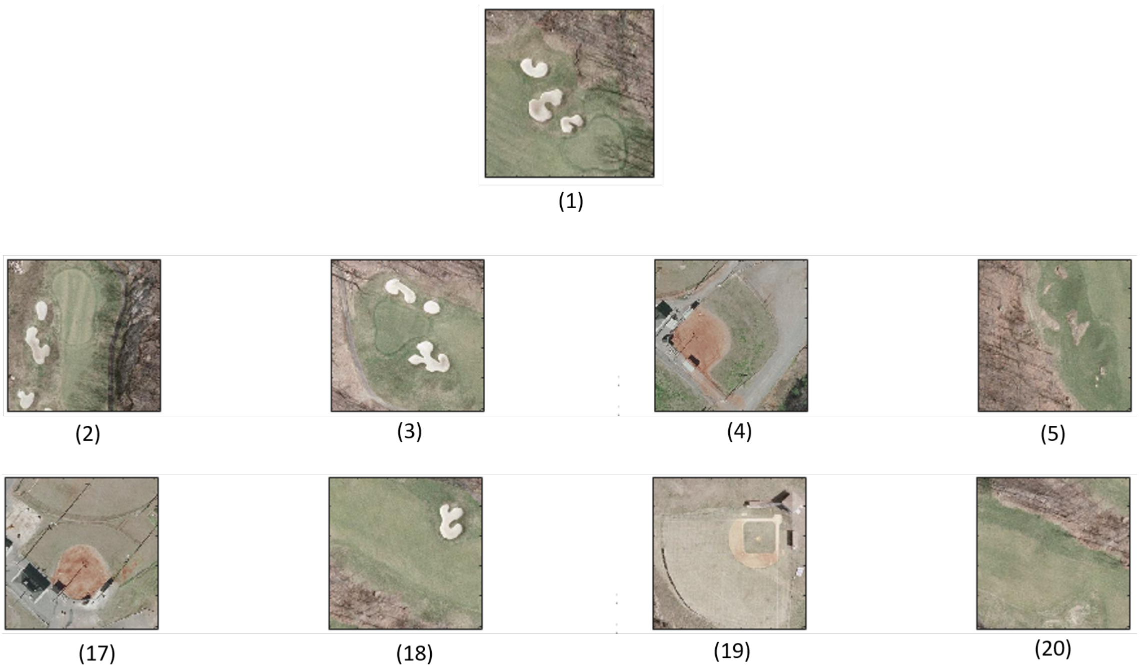
Figure 5.15: Multi-dictionary intra single label statistical features average recall


(1)

(2) (3) (4) (5)

(17) (18) (19) (20)

Figure 5.16: Example of retrieved intersection images


(1)

(2) (3) (4) (5)

(17) (18) (19) (20)

Figure 5.17: Example of retrieved golf course images

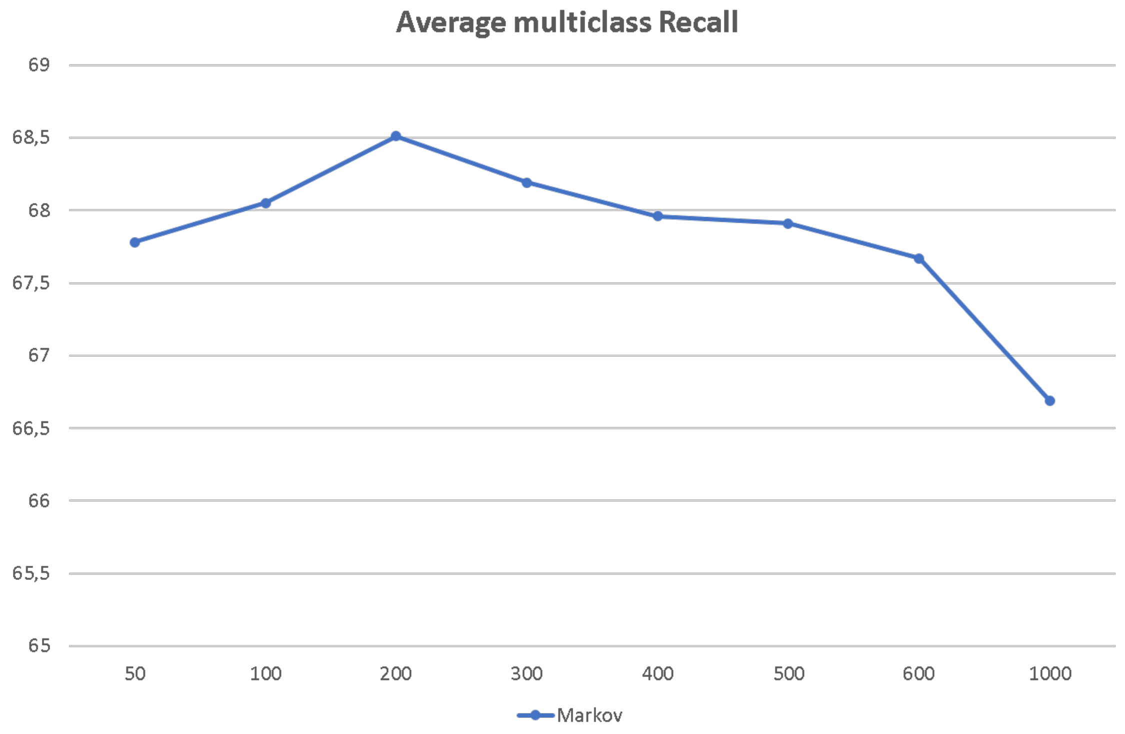


Figure 5.18: Average recall in function of dictionary length

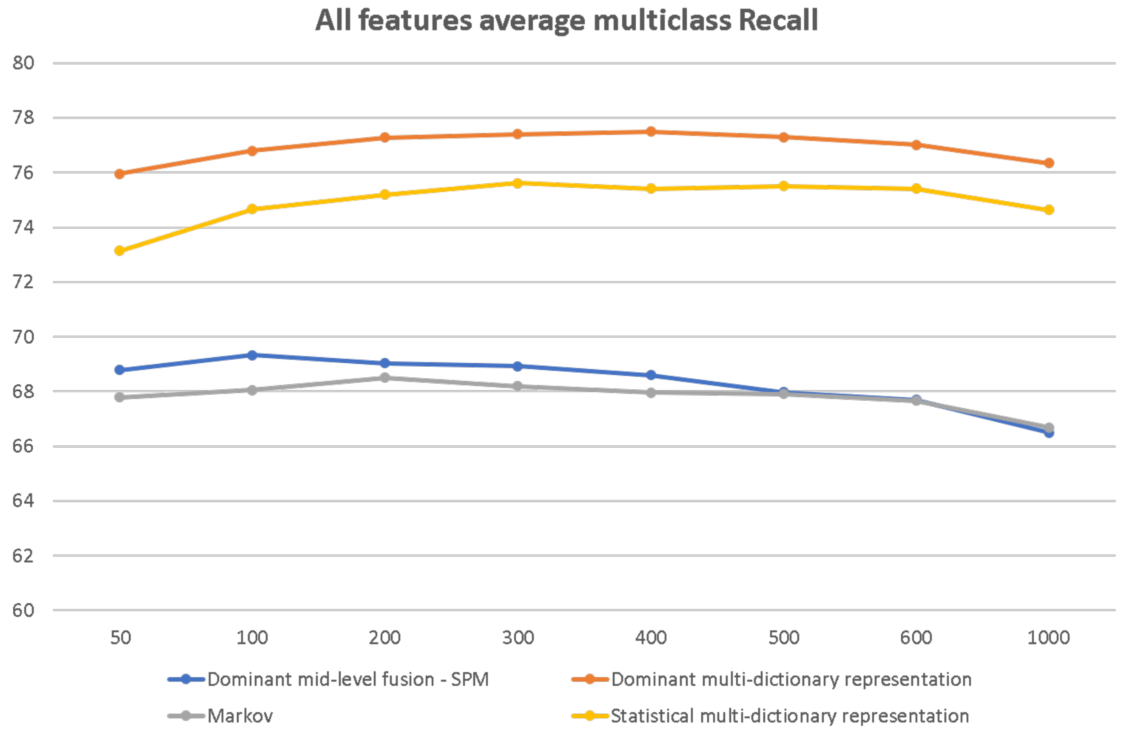


Figure 5.19: Average recall of all descriptors

5.3 Experimental Results of Proposed Query Sensitive Feature Weighting Algorithm

In table 5.3 is illustrated the average signle label recall against size of training set, in which first value $\bullet/\cdot$ indicates how many image in training set are representative of query class, while second value $\cdot/\bullet$ is the number of how many images are taken from each class to forming the against class in the training set.

Table 5.3: Average recall in function of training set

Recall	**25/2**	**25/5**	**25/10**	**25/15**
SVM	73.75	79.81	81.49	85.06
MKL + SVM	75.80	81.63	83.85	87.90

While table 5.4 represents the mean accuracy in percentage of scene recognition on the entire database, so that measures the accuracy of the system to recognize correctly query class in the whole database. Then is obvious that by incrementing number of training sample in the against class, the learned hyperplane starts to include in the query side also a large number of images belonging to other single-label classes.

Table 5.4: Average scene recognition accuracy in function of training set

Scene accuracy	**25/2**	**25/5**	**25/10**	**25/15**
SVM	82.18	74.20	70.32	64.95
MKL + SVM	83.87	75.83	71.70	66.76

Following some visual results are given, as before are shown the first 5 ranked images and the lasts 5 of the retrieved image set.

In this series of experiments benefits of MKL on final retrieval performances are clear, but on the other hand, considering how system was implement, i.e. simply include the query image in a training set depending on its class and then training, is a time costing operation. In order to weighting in real time query features is necessary to boost this part of the process.

5.4 Computational Complexity Analysis

The main motivation behind this study is concerning the advantage of extracting and manipulating images directly in DCT domain. This fact was clear from the very beginning of the JPEG images processing ”era”, for example already in 1993 in [86] was stated that manipulating images in compressed domain, yielding performance 50 to 100 times than

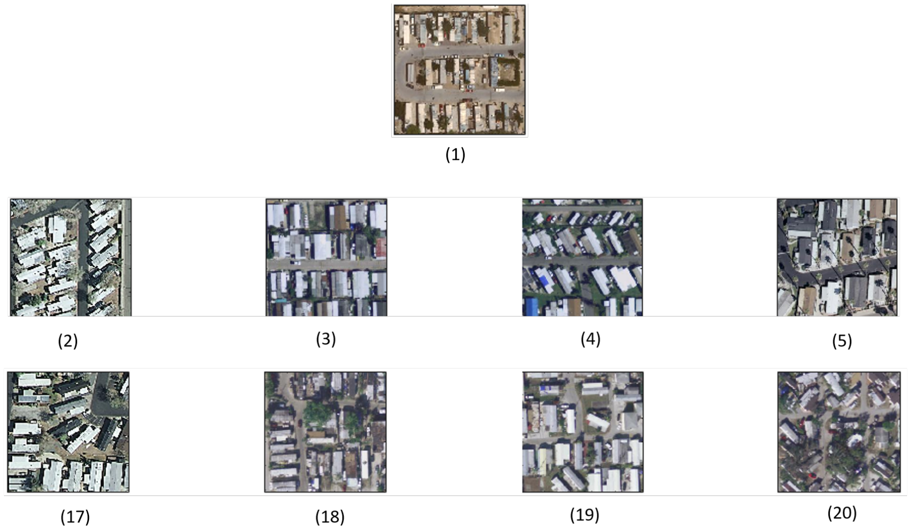


Figure 5.20: Example of retrieved mobile homepark images

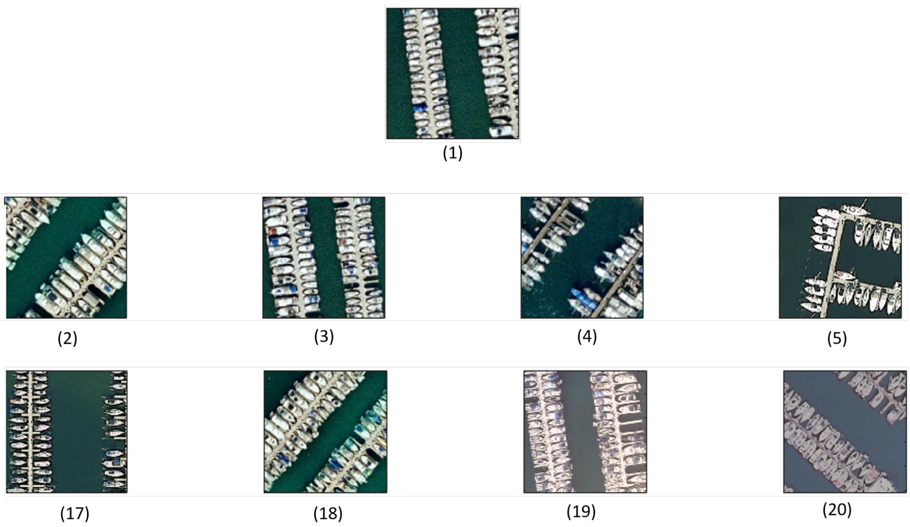


Figure 5.21: Example of retrieved harbor images

pixel based manipulaltion algorithms. Further think about the fact that fast DCT implementation studied by E. Feig and S. Winograd in [87] involve 94 multiplication and 454 addition/subtraction for each block. For example in this work each image is subdivided in 1024 blocks, therefore is evident that overhead operations, as decompression, in a massive database have a huge impact. So decompression time is a way to measure the advantages of partial decompression among fully image decompression in a real-world system. Then in this experiment average time of whole archive has been taken. In algorithm 1 all operations are shown. At beginning, through JPEG matlab toolbox function *jpeg_read*(●), DCT coefficients are derived for each color band, at this point time for partial decompression is taken. It is straightful that proposed features can be already extracted, for example dominant features 3.2 does not need overhead time to be extracted. Partial decompression is over, while for full-decompression 2D IDCT is needed, in that case standard matlab implementation is used. Dequantization step is not needed, because in the experiments images have been compressed with a Q-factor equal to 100, that means that quantization table is depicted in table 5.6.

To have a reliable time measure, experiment has been run for 100 times and finally all average values were averaged in turn. Final results are given in the following table:

Table 5.5: Decompression time results

Partial decompression	**Full decompression**
0.0016 s	0.1062 s

The results show that full decompression is 100 times slower, respect partial one. It is obvious that full decompression can be parallelized by processing multiple DCT blocks at times, but compressed-based processing, as the proposed methods, still remains the only way to effectively manage huge amount of data in less time. Then another important paramenter is time to evaluate dictionary values, but cosider that k-means algorithm has a time complexity equal to $\mathcal{O}(n \cdot k \cdot d)$ in which n is the number of features, k is the number of cluster and d is the dimension of features. For example proposed dominant features has a constructed by $15D$ vectors, while another famous local features such as SIFT is constructed with $128D$ descriptors, then is clear that also in that case JPEG-compressed features outperforms in terms of time other kind of descriptors.

Table 5.6: Used quantization table

1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1

```
Result: Calculate average decompression time
initialization;
while not at the end of image archive do
    start partial decompression timer;
    start full decompression timer;
    extract DCT coefficients;
    stop partial decompression timer;
    accumulate partial decompression time;
    2D 8x8 IDCT over DCT coefficients;
    stop full decompression timer;
    accumulate full decompression time;
end
$average$ operation on partial decompression time;
$average$ operation on full decompression time;
```

Algorithm 1: time experiment routine

Chapter 6

Conclusions and Discussions

In this thesis a series of JPEG-compressed based desriptors were proposed and tested in order to study the effectiveness of a JPEG-compressed-based image retrieval system. The proposed descriptors provide, for the first time in RS context, a rough characterization of the image content without complete decompression taking place. The first two kind of features, dominant e statistical, were extracted from the low frequency DCT coefficients of the JPEG block, in which one is formed by coefficients as they are, while the other is made up of statistics derived from coefficients. Previous local features does not take into account the spatial correlation with between neighbor coefficients, then a features derived from steganalysis, called JPEG Markov feature, is used to capture the statistical behaviour among near blocks and near coefficients of a image patch.Then these local features were represented with a multi-dictionary BOVW approach, in which a set of meaninful features, called dictionary, were used to describe the whole features archive and then a global signature of the image were derivated. In addition to all this spatial pyramid matching, that is a weighted comparison strategy between image signatures, were used. In the first part of experiments a CBIR system based on simple k-NN strategy was implemented and tested with different dictionary lengths, from the experiments it has been found that the most effective image characterisation is given by the so-called dominant features. While representations that take into account a weak spatial information, such as JPEG markov features and spatial pyramid matching, are given lowest results due to the sensitivity of k-NN to the representation length.

After this first part of unsupervised experiments a supervised features weighting scheme were proposed and tested for valorizing and fusiong optimale visual meaning of the proposed features. In this last part of features testing only dominant features were testing in a CBIR system based on SVM. For performing feature weighting multiple kernel learning algorithm were used. Then to each mid-level kind of feature representation a basis kernel was associated and finally all kernel representations were optimally combined in a supervised way. These experiments were conducted in a single-label environment and in function to different training set size. From the results is clear that a CBIR system that combines multiple kernel learning with SVM, outperforms the one based on simple SVM.

Final section of experiments demonstrates the time-efficiency of extracting features from compressed stream compared to the standard full image decompression method. From the experiments are clear the possible benefits hat can be obtained in a real massive Earth observations archive retrieval system. Then experiments shown that proposed descriptors could help in numerous way applications in which a large amount of data are involved. Obviously there is to consider that all obtained results are derivated by testing a small benchmark archive, therefore, a decrease in performance is expected when methods are applied to real massive archives. Moreover, although RGB aerial orthoimagery images were used in the experimental analysis, the proposed descriptors can also be applied on images with more than three bands, as the features are extracted independently from each band.

During the processing chain there are several points where improvements can be made. For example it can be possible to work at low-level features extraction in order to find a stronger block representation compared with those proposed. Just think to the proposed dominant features that are rotational variant. Another point of improvement could be at level of BOVW representation, in which computer vision and pattern recognition literatures are rich of methods that extend BOVW based features in powerful ways. Then instead of building a strong features at low-level is can be possible to work at higher representation level to make solid image descriptors.

Bibliography

[1] M. Chi, A. Plaza, J. A. Benediktsson, Z. Sun, J. Shen, and Y. Zhu. Big data for remote sensing: Challenges and opportunities. *Proceedings of the IEEE*, 104(11):2207–2219, Nov 2016.

[2] Yan Ma, Haiping Wu, Lizhe Wang, Bormin Huang, Rajiv Ranjan, Albert Zomaya, and Wei Jie. Remote sensing big data computing: Challenges and opportunities. *Future Generation Computer Systems*, 51:47 – 60, 2015. Special Section: A Note on New Trends in Data-Aware Scheduling and Resource Provisioning in Modern HPC Systems.

[3] Amir Gandomi and Murtaza Haider. Beyond the hype: Big data concepts, methods, and analytics. *International Journal of Information Management*, 35(2):137 – 144, 2015.

[4] Charles Toth and Grzegorz Jóźków. Remote sensing platforms and sensors: A survey. *ISPRS Journal of Photogrammetry and Remote Sensing*, 115:22 – 36, 2016. Theme issue 'State-of-the-art in photogrammetry, remote sensing and spatial information science'.

[5] ESA.

[6] Du Peijun, Chen Yunhao, Tang Hong, and Fang Tao. Study on content-based remote sensing image retrieval. In *Proceedings. 2005 IEEE International Geoscience and Remote Sensing Symposium, 2005. IGARSS '05.*, volume 2, pages 4 pp.–, July 2005.

[7] P. Kempeneers and P. Soille. Optimizing sentinel-2 image selection in a big data context. *Big Earth Data*, 1(1-2):145–158, 2017.

[8] Erik Borg, Bernd Fichtelmann, and Hartmut Asche. Cloud classification in jpeg-compressed remote sensing data (landsat 7/etm+). In Beniamino Murgante, Osvaldo Gervasi, Sanjay Misra, Nadia Nedjah, Ana Maria A. C. Rocha, David Taniar, and Bernady O. Apduhan, editors, *Computational Science and Its Applications – ICCSA 2012*, pages 347–357, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

[9] Erik Borg, Bernd Fichtelmann, Christian Fischer, and Hartmut Asche. Design and implementation of data usability processor into an automated processing chain for

optical remote sensing data. In Anna-Lena Lamprecht, editor, *Leveraging Applications of Formal Methods, Verification, and Validation*, pages 21–37, Cham, 2016. Springer International Publishing.

[10] Curtis E. Woodcock, Richard Allen, Martha Anderson, Alan Belward, Robert Bindschadler, Warren Cohen, Feng Gao, Samuel N. Goward, Dennis Helder, Eileen Helmer, Rama Nemani, Lazaros Oreopoulos, Joh Schott, Prasad S. Thenkabail, Eric F. Vermote, James Vogelmann, Michael A. Wulder, and Randolph Wynne. Free access to landsat imagery. *Science*, 320(5879):1011–1011, 2008.

[11] G. K. Wallace. The jpeg still picture compression standard. *IEEE Transactions on Consumer Electronics*, 38(1):xviii–xxxiv, Feb 1992.

[12] F. Tintrup, F. De Natale, and D. Giusto. Compression algorithms for classification of remotely sensed images. In *Acoustics, Speech and Signal Processing, 1998. Proceedings of the 1998 IEEE International Conference on*, volume 5, pages 2565–2568 vol.5, May 1998.

[13] Guoxia Yu, Tanya Vladimirova, and Martin N. Sweeting. Image compression systems on board satellites. *Acta Astronautica*, 64(9):988 – 1005, 2009.

[14] A. Zabala, X. Pons, R. Diaz-Delgado, F. Garcia, F. Auli-Llinas, and J. Serra-Sagrista. Effects of jpeg and jpeg2000 lossy compression on remote sensing image classification for mapping crops and forest areas. In *2006 IEEE International Symposium on Geoscience and Remote Sensing*, pages 790–793, July 2006.

[15] W.-L. Lau, Z.-L. Li, and K. W.-K. Lam. Effects of jpeg compression on image classification. *International Journal of Remote Sensing*, 24(7):1535–1544, 2003.

[16] M. Shneier and M. Abdel-Mottaleb. Exploiting the jpeg compression scheme for image retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(8):849–853, Aug 1996.

[17] J. A. Lay and Ling Guan. Image retrieval based on energy histograms of the low frequency dct coefficients. In *1999 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings. ICASSP99 (Cat. No.99CH36258)*, volume 6, pages 3009–3012 vol.6, Mar 1999.

[18] Gerald Schaefer. Jpeg image retrieval by simple operators, 2001.

[19] J. Jiang, A. Armstrong, and G.C. Feng. Direct content access and extraction from jpeg compressed images. *Pattern Recognition*, 35(11):2511 – 2519, 2002.

[20] Guocan Feng and Jianmin Jiang. Jpeg compressed image retrieval via statistical features. *Pattern Recognition*, 36(4):977 – 985, 2003.

[21] Minyoung Eom and Yoonsik Choe. Fast extraction of edge histogram in dct domain based on mpeg7. In *Proceedings of World Academy of Science, Engineering and Technology Volume 9 November 2005 ISSN*, pages 1307–6884.

[22] Zhe ming Lu and Hans Burkhardt. A content-based image retrieval scheme in jpeg compressed domain, 2005.

[23] K.M. Au, N.F. Law, and W.C. Siu. Unified feature analysis in jpeg and jpeg 2000-compressed domains. *Pattern Recognition*, 40(7):2049 – 2062, 2007.

[24] G. Schaefer, D. Edmundson, K. Takada, S. Tsuruta, and Y. Sakurai. Effective and efficient filtering of retrieved images based on jpeg header information. In *2012 Eighth International Conference on Signal Image Technology and Internet Based Systems*, pages 644–649, Nov 2012.

[25] D. Edmundson and G. Schaefer. Efficient and effective online image retrieval. In *2012 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 2312–2317, Oct 2012.

[26] Zhongwei He, Wei Lu, Wei Sun, and Jiwu Huang. Digital image splicing detection based on markov features in dct and dwt domain. *Pattern Recognition*, 45(12):4292 – 4299, 2012.

[27] T. Bretschneider, R. Cavet, and O. Kao. Retrieval of remotely sensed imagery using spectral information content. In *IEEE International Geoscience and Remote Sensing Symposium*, volume 4, pages 2253–2255 vol.4, 2002.

[28] T Bretschneider and O Kao. A retrieval system for remotely sensed imagery. In *International Conference on Imaging Science, Systems, and Technology*, volume 2, pages 439–445, 2002.

[29] Qian Bao and Ping Guo. Comparative studies on similarity measures for remote sensing image retrieval. In *2004 IEEE International Conference on Systems, Man and Cybernetics (IEEE Cat. No.04CH37583)*, volume 1, pages 1112–1116 vol.1, Oct 2004.

[30] G. J. Scott, M. N. Klaric, C. H. Davis, and C. R. Shyu. Entropy-balanced bitmap tree for shape-based object retrieval from large-scale satellite imagery databases. *IEEE Transactions on Geoscience and Remote Sensing*, 49(5):1603–1616, May 2011.

[31] A. Ma and I. K. Sethi. Local shape association based retrieval of infrared satellite images. In *Seventh IEEE International Symposium on Multimedia (ISM'05)*, pages 7 pp.–, Dec 2005.

[32] Yao Hongyu, Li Bicheng, and Cao Wen. Remote sensing imagery retrieval based-on gabor texture feature classification. In *Signal Processing, 2004. Proceedings. ICSP '04. 2004 7th International Conference on*, volume 1, pages 733–736 vol.1, Aug 2004.

[33] Shawn D Newsam and Chandrika Kamath. Retrieval using texture features in high-resolution multispectral satellite imagery. In *Data Mining and Knowledge Discovery: Theory, Tools, and Technology VI*, volume 5433, pages 21–33. International Society for Optics and Photonics, 2004.

[34] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, Nov 2004.

[35] Y. Yang and S. Newsam. Geographic image retrieval using local invariant features. *IEEE Transactions on Geoscience and Remote Sensing*, 51(2):818–832, Feb 2013.

[36] L. Chen, W. Yang, K. Xu, and T. Xu. Evaluation of local features for scene classification using vhr satellite images. In *2011 Joint Urban Remote Sensing Event*, pages 385–388, April 2011.

[37] E. Aptoula. Remote sensing image retrieval with global morphological texture descriptors. *IEEE Transactions on Geoscience and Remote Sensing*, 52(5):3023–3034, May 2014.

[38] M. Musci, R. Q. Feitosa, G. A. O. P. Costa, and M. L. F. Velloso. Assessment of binary coding techniques for texture characterization in remote sensing imagery. *IEEE Geoscience and Remote Sensing Letters*, 10(6):1607–1611, Nov 2013.

[39] M. Schroder, H. Rehrauer, K. Seidel, and M. Datcu. Interactive learning and probabilistic retrieval in remote sensing image archives. *IEEE Transactions on Geoscience and Remote Sensing*, 38(5):2288–2298, Sep 2000.

[40] Xiang Sean Zhou and Thomas S Huang. Relevance feedback in image retrieval: A comprehensive review. *Multimedia systems*, 8(6):536–544, 2003.

[41] Pengyu Hong, Qi Tian, and T. S. Huang. Incorporate support vector machines to content-based image retrieval with relevance feedback. In *Proceedings 2000 International Conference on Image Processing (Cat. No.00CH37101)*, volume 3, pages 750–753 vol.3, 2000.

[42] B. Demir and L. Bruzzone. A novel active learning method in relevance feedback for content-based remote sensing image retrieval. *IEEE Transactions on Geoscience and Remote Sensing*, 53(5):2323–2334, May 2015.

[43] B. Chaudhuri, B. Demir, L. Bruzzone, and S. Chaudhuri. Region-based retrieval of remote sensing images using an unsupervised graph-theoretic approach. *IEEE Geoscience and Remote Sensing Letters*, 13(7):987–991, July 2016.

[44] Selim Aksoy. Modeling of remote sensing image content using attributed relational graphs. In Dit-Yan Yeung, James T. Kwok, Ana Fred, Fabio Roli, and Dick de Ridder, editors, *Structural, Syntactic, and Statistical Pattern Recognition*, pages 475–483, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

[45] M. Wang and T. Song. Remote sensing image retrieval by scene semantic matching. *IEEE Transactions on Geoscience and Remote Sensing*, 51(5):2874–2886, May 2013.

[46] B. Ozdemir and S. Aksoy. Image classification using subgraph histogram representation. In *2010 20th International Conference on Pattern Recognition*, pages 1112–1115, Aug 2010.

[47] Gong Cheng, Lei Guo, Tianyun Zhao, Junwei Han, Huihui Li, and Jun Fang. Automatic landslide detection from remote-sensing imagery using a scene classification method based on bovw and plsa. *International Journal of Remote Sensing*, 34(1):45–59, 2013.

[48] Q. Zhu, Y. Zhong, B. Zhao, G. S. Xia, and L. Zhang. Bag-of-visual-words scene classifier with local and global features for high spatial resolution remote sensing imagery. *IEEE Geoscience and Remote Sensing Letters*, 13(6):747–751, June 2016.

[49] Lianzhi Huo Lijun Zhao, Ping Tang. Feature significance-based multibag-of-visual-words model for remote sensing image scene classification. *Journal of Applied Remote Sensing*, 10:10 – 10 – 21, 2016.

[50] Hang Wu, Baozhen Liu, Weihua Su, Wenchang Zhang, and Jinggong Sun. Hierarchical coding vectors for scene level land-use classification. *Remote Sensing*, 8(5), 2016.

[51] K. Qi, H. Wu, C. Shen, and J. Gong. Land-use scene classification in high-resolution remote sensing images using improved correlatons. *IEEE Geoscience and Remote Sensing Letters*, 12(12):2403–2407, Dec 2015.

[52] Y. Zhang, X. Sun, H. Wang, and K. Fu. High-resolution remote-sensing image classification via an approximate earth mover's distance-based bag-of-features model. *IEEE Geoscience and Remote Sensing Letters*, 10(5):1055–1059, Sept 2013.

[53] Yi Yang and S. Newsam. Spatial pyramid co-occurrence for image classification. In *2011 International Conference on Computer Vision*, pages 1465–1472, Nov 2011.

[54] S. Xu, T. Fang, D. Li, and S. Wang. Object classification of aerial images with bag-of-visual words. *IEEE Geoscience and Remote Sensing Letters*, 7(2):366–370, April 2010.

[55] Jingwen Hu, Gui-Song Xia, Fan Hu, and Liangpei Zhang. A comparative study of sampling analysis in the scene classification of optical high-spatial resolution remote sensing imagery. *Remote Sensing*, 7(11):14988–15013, 2015.

[56] F. Hu, G. S. Xia, Z. Wang, X. Huang, L. Zhang, and H. Sun. Unsupervised feature learning via spectral clustering of multidimensional patches for remotely sensed scene classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 8(5):2015–2030, May 2015.

[57] L. J. Zhao, P. Tang, and L. Z. Huo. Land-use scene classification using a concentric circle-structured multiscale bag-of-visual-words model. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 7(12):4620–4631, Dec 2014.

[58] S. Chen and Y. Tian. Pyramid of spatial relatons for scene-level land use classification. *IEEE Transactions on Geoscience and Remote Sensing*, 53(4):1947–1957, April 2015.

[59] R. Bahmanyar, S. Cui, and M. Datcu. A comparative study of bag-of-words and bag-of-topics models of eo image patches. *IEEE Geoscience and Remote Sensing Letters*, 12(6):1357–1361, June 2015.

[60] J. Sivic and A. Zisserman. Video google: a text retrieval approach to object matching in videos. In *Proceedings Ninth IEEE International Conference on Computer Vision*, pages 1470–1477 vol.2, Oct 2003.

[61] A. Vedaldi and B. Fulkerson. VLFeat: An open and portable library of computer vision algorithms. http://www.vlfeat.org/, 2008.

[62] Navneet Dalal and Bill Triggs. Histograms of Oriented Gradients for Human Detection. In Cordelia Schmid, Stefano Soatto, and Carlo Tomasi, editors, *International Conference on Computer Vision & Pattern Recognition (CVPR '05)*, volume 1, pages 886–893, San Diego, United States, June 2005. IEEE Computer Society.

[63] J. MacQueen. Some methods for classification and analysis of multivariate observations. In *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Statistics*, pages 281–297, Berkeley, Calif., 1967. University of California Press.

[64] S. Lloyd. Least squares quantization in pcm. *IEEE Transactions on Information Theory*, 28(2):129–137, March 1982.

[65] James C. Bezdek, Robert Ehrlich, and William Full. Fcm: The fuzzy c-means clustering algorithm. *Computers and Geosciences*, 10(2):191 – 203, 1984.

[66] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 2169–2178, 2006.

[67] Y. Huang, Z. Wu, L. Wang, and T. Tan. Feature coding in image classification: A comprehensive study. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(3):493–506, March 2014.

[68] Piotr Koniusz, Fei Yan, and Krystian Mikolajczyk. Comparison of mid-level feature coding approaches and pooling strategies in visual concept detection. *Computer Vision and Image Understanding*, 117(5):479 – 492, 2013.

[69] S. Niazmardi, B. Demir, L. Bruzzone, A. Safari, and S. Homayouni. Multiple kernel learning for remote sensing image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 56(3):1425–1443, March 2018.

[70] Tinghua Wang, Dongyan Zhao, and Yansong Feng. Two-stage multiple kernel learning with multiclass kernel polarization. *Knowledge-Based Systems*, 48:10–16, 2013.

[71] Gert RG Lanckriet, Nello Cristianini, Peter Bartlett, Laurent El Ghaoui, and Michael I Jordan. Learning the kernel matrix with semidefinite programming. *Journal of Machine learning research*, 5(Jan):27–72, 2004.

[72] Manik Varma and Bodla Rakesh Babu. More generality in efficient multiple kernel learning. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 1065–1072. ACM, 2009.

[73] Nello Cristianini, John Shawe-Taylor, Andre Elisseeff, and Jaz S Kandola. On kernel-target alignment. In *Advances in neural information processing systems*, pages 367–373, 2002.

[74] Yong Liu, Shizhong Liao, and Yuexian Hou. Learning kernels with upper bounds of leave-one-out error. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, pages 2205–2208. ACM, 2011.

[75] Yanfeng Gu, Chen Wang, Di You, Yuhang Zhang, Shizhe Wang, and Ye Zhang. Representative multiple kernel learning for classification in hyperspectral imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 50(7):2852–2865, 2012.

[76] Mehmet Gönen and Ethem Alpaydın. Multiple kernel learning algorithms. *Journal of machine learning research*, 12(Jul):2211–2268, 2011.

[77] T. Tsai, Y. P. Huang, and T. W. Chiang. Dominant feature extraction in block-dct domain. In *2006 IEEE International Conference on Systems, Man and Cybernetics*, volume 5, pages 3623–3628, Oct 2006.

[78] Fazal e Malik and B. Baharudin. Effective content-based image retrieval: Combination of quantized histogram texture features in the dct domain. In *2012 International Conference on Computer Information Science (ICCIS)*, volume 1, pages 425–430, June 2012.

[79] D. Fu, Y. Q. Shi, D. Zou, and G. Xuan. Jpeg steganalysis using empirical transition matrix in block dct domain. In *2006 IEEE Workshop on Multimedia Signal Processing*, pages 310–313, Oct 2006.

[80] Alain Rakotomamonjy, Francis R Bach, Stéphane Canu, and Yves Grandvalet. Simplemkl. *Journal of Machine Learning Research*, 9(Nov):2491–2521, 2008.

[81] Yi Yang and Shawn Newsam. Bag-of-visual-words and spatial extensions for land-use classification. In *Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems*, GIS '10, pages 270–279, New York, NY, USA, 2010. ACM.

[82] B. Chaudhuri, B. Demir, S. Chaudhuri, and L. Bruzzone. Multilabel remote sensing image retrieval using a semisupervised graph-theoretic method. *IEEE Transactions on Geoscience and Remote Sensing*, 56(2):1144–1158, Feb 2018.

[83] Phil Sallee. Matlab jpeg toolbox, 2003.

[84] Chih-Chung Chang and Chih-Jen Lin. Libsvm: A library for support vector machines. *ACM Trans. Intell. Syst. Technol.*, 2(3):27:1–27:27, May 2011.

[85] T. Cover and P. Hart. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1):21–27, January 1967.

[86] B. C. Smith and L. A. Rowe. Algorithms for manipulating compressed images. *IEEE Computer Graphics and Applications*, 13(5):34–42, Sept 1993.

[87] E. Feig and S. Winograd. Fast algorithms for the discrete cosine transform. *IEEE Transactions on Signal Processing*, 40(9):2174–2193, Sep 1992.