Compact and Adaptive Spatial Pyramids for Scene Recognition

Blind for review

Blind for review

Abstract

Most successful approaches on scene recognition tend to efficiently combine global image features with spatial local appearance and shape cues. On the other hand, less attention has been devoted for studying spatial texture features within scenes. Our method is based on the insight that scenes can be seen as a composition of micro-texture patterns. This paper analyzes the role of texture along with its spatial layout for scene recognition. However, one main drawback of the resulting spatial representation is its huge dimensionality. Hence, we propose a technique that addresses this problem by presenting a compact Spatial Pyramid (SP) representation. The basis of our compact representation, namely, Compact Adaptive Spatial Pyramid (CASP) consists of a two-stages compression strategy. This strategy is based on the Agglomerative Information Bottleneck (AIB) theory for (i) compressing the least informative SP features, and, (ii) automatically learning the most appropriate shape for each category. Our method exceeds the state-of-the-art results on several challenging scene recognition data sets.

Keywords: Scene Recognition, Spatial Pyramids, Texture, Dimensionality Reduction, and Agglomerative Information Theory.

1. Introduction

Scene recognition is one of the most appealing, yet challenging problems in computer vision. Fig. 1 shows such kind of challenges, namely, illumination changes, intra-class variabilities, scale variabilities, and inter-class similarities. The goal is to identify an image as belonging to one of several scene classes such as mountains, beaches, indoor-offices, etc.. Effective solutions to this problem can be useful in many other applications, such as detection [1, 2], action recognition [3], and content based image retrieval [4]. Approaches to scene recognition can be divided into two main categories. First, methods that use low-level features such as color, texture, etc. [5, 6]. Despite the good performance obtained using these approaches, they lack an intermediate image description (such as the presence of the sky, grass, or other semantic concepts) that can be extremely valuable in determining scene types [7, 8, 9]. On the other hand, other techniques make use of intermediate representations [10, 11, 12]. Towards this direction, the Bag-of-Words (BoW) approach has been used to model scenes [13, 14, 15]. However, its foremost shortcoming is the lack of spatial information. Recently, several approaches considered the immense success of Spatial Pyramid (SP) [16, 17], due to its

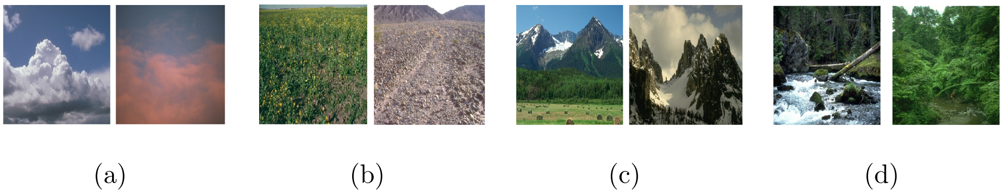


Figure 1: Scene recognition challenges are: (a) illumination changes, (b) intra-class variabilities, (c) scale variabilities, and (d) inter-class similarities (in the example, river can be easily confused with forest).

inclusion of important spatial information. For example, Bosch et. al. [17] demonstrated how spatial appearance features benefit the scene recognition task. Moreover, the work in [18, 19] showed the significance of fusing complementary spatial shape, and appearance features along with global image cues. However, much less attention has been devoted to studying the texture features within this context. To this end, we propose three contributions over the standard SP:

- The first contribution is concerned with, the exploration of the spatial texture features along with the shape, and appearance cues for scene recognition. Our novel descriptor is mainly inspired by two sources: (i) the Pyramid of Histograms of Oriented Gradients (PHOG) descriptor [18], and (ii) the Histogram of Three Patch Local Binary Patterns (TPLBP) [20, 21], which has been recently proposed to encode texture data in both static images and videos.

- The second contribution is addressing the huge dimensional histograms generated using the standard SP scheme, while going towards the finest level of representation. We address this problem by finding a more compact SP representation that maintains or even improves their original counterparts.

- The third contribution is regarding the rigid SP assumption proposed by Lazebnik et. al. [16], that suits each category. We propose a method for learning the best partition for each category. The resulting SP shapes have the advantages of being compact, while improving the original SP performance.

We refer to the resulting representation of combining the second and third contributions as *Compact Adaptive Spatial Pyramid* (CASP). This powerful representation helps in overcoming the common scene recognition challenges, shown in Fig. 1, and, consequently, improving the scene recognition performance.

Outline This paper is organized as follows: next section proposes our novel texture descriptor. Section 3 briefly explains the basic idea of *Spatial Pyramids (SP)*, and discusses how the *Agglomerative Information Bottleneck (AIB)* theory is extended for building our novel *Compact Adaptive Spatial Pyramid (CASP)*. Section 4 describes the datasets used in the experiments. Section 5 shows, and compares the experimental results with the state-of-the-art. Finally, section 6 presents the main conclusions of this paper, and shows the most important avenues of future research.

2. Pyramids of Colored Three-Patch Local Binary Patterns

Our first contribution exploits the spatial texture features for the task of scene recognition. We propose using our texture representation that retains both local image texture, and its spatial layout. This novel representation is able to capture the large illumination variabilities illustrated in Fig.1(a).

Our descriptor is an extension of the Local Binary Patterns (LBP), which has been shown to be one of the best performing texture descriptors [22, 23, 24, 21]. *LBP* has been successfully used in various applications, such as face recognition [23], background subtraction [25], object recognition [26], interest regions description [24], and action recognition [21]. It also has various properties that favor its usage such as, its tolerance against illumination changes, and its ability to discriminate a large range of rotated textures efficiently. Moreover, its computational simplicity, and efficiency makes it suitable for the scene recognition task.

The next sections describe our novel spatial texture descriptor: First, we give a brief survey on *LBP*. We further examine the incorporation of color [27, 17, 28] and spatial information to our final texture representation.

2.1. Local Binary Patterns

LBP descriptor [22], and its variants use short binary strings to encode properties of the local micro-texture around each pixel. *LBP* simplest form works as follows: For a 3×3 neighborhood, (i) the value of each pixel is compared with the central pixel's intensity value, and (ii) the result from each pixel is then concatenated to form an 8 bits binary descriptor.

Recently, significant works introduce variants of *LBP* descriptors, which are based on patch statistics, namely: Center-Symmetric LBP *(CSLBP)* [24], Three-Patch LBP *(TPLBP)* [20], and Four-Patch LBP *(FPLBP)* [20]. *CSLBP* encodes at each pixel the gradient signs at the pixel at four different angles. *TPLBP* and *FPLBP* encode the similarities between neighboring patches of pixels, thus capturing information which is complementary to pixel-based descriptors.

2.2. Colored Three-Patch Local-Binary Patterns

In this section, we propose to fuse *TPLBP* with complementary color information. We refer to it as *Colored TPLBP* (C-TPLBP). To compute *C-TPLBP*, we extract the *TPLBP* descriptor for each channel of the examined color spaces. Consequently, a dictionary is created for each color-texture channel. The generated histograms of each color-texture channel are then concatenated. This results in a ($vocabularysize \times 3$) dimensional histogram for the standard BoW representation.

In particular, we examine two standard color spaces, namely, Opponent Color *(OppC)* [28, 29] and *HSV*. *OppC* is defined as:

$$O1 = \frac{R-G}{\sqrt{2}}, O2 = \frac{R+G-2B}{\sqrt{6}}, O3 = \frac{R+G+B}{\sqrt{3}} \tag{1}$$

So the fusion of *OppC* with *TPLBP* is done as follows[1]:

$$\text{C-TPLBP(OppC)} = \text{TPLBP(O1)} + \text{TPLBP(O2)} + \text{TPLBP(O3)}. \tag{2}$$

Similarly, fusing *TPLBP* with *HSV* is done as:

$$\text{C-TPLBP(HSV)} = \text{TPLBP(H)} + \text{TPLBP(S)} + \text{TPLBP(V)}. \tag{3}$$

2.3. Pyramids of Colored TPLBP

Finally, in order to incorporate the spatial information, we follow the scheme proposed by [16]. In particular, we apply the standard spatial pyramid upon each color-texture channel, as shown in Fig. 2. This leads to a ($vocabularysize \times 21$) dimensional histogram per channel. The resulting descriptor is referred to as *Pyramids of C-TPLBP* (PC-TPLBP).

The main drawback of the resulting spatial representation, is its high dimensionality. In the following section we will introduce our *CASP* approach

[1] The + operator indicates that the histograms of each colored-texture channel are concatenated.

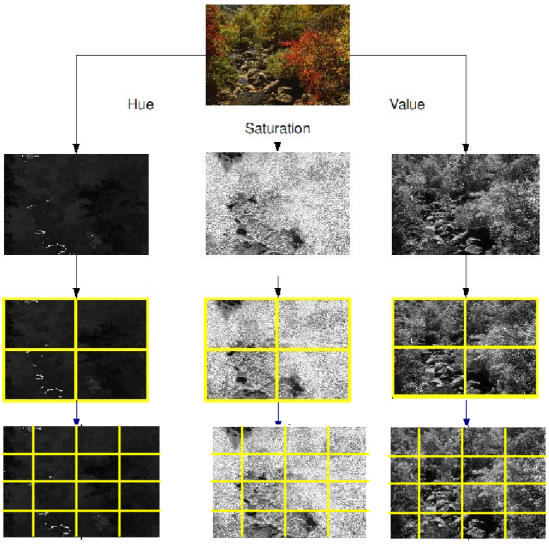


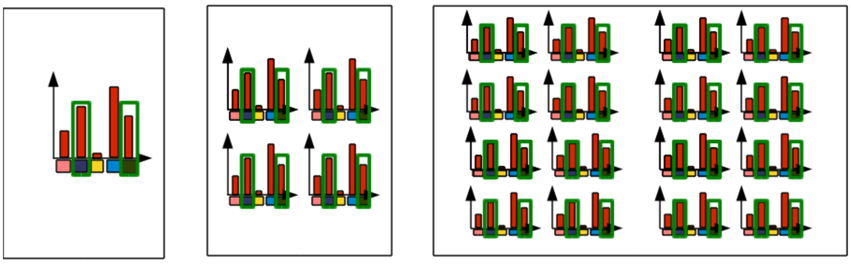
Figure 2: Given the dictionaries built for each colored-texture channel, they are concatenated using a $(1 \times 1) + (2 \times 2) + (4 \times 4) = 21$ image representation. This representation is denoted to as *PC-TPLBP* resulting in a $21 \times (vocsize)$ dimensional histogram.

for reducing the SP dimensionality of the final histogram, while preserving its original performance.

3. Compact and Adaptive Spatial Pyramids

SP proposed by [16] is a simple, and computationally efficient extension of the order-less *BoW*. This technique works by representing an image using weighted multi-resolution histograms. These histograms are obtained by repeatedly sub-dividing the image into increasingly finer sub-regions by doubling the number of divisions on each axis direction and computing histograms of features over the resulting sub-regions.

Matches within each sub-region are then determined. Matches found at finer resolutions are closer to each other in the image space, and are therefore



(a) $l0$ (b) $l1$ (c) $l2$

Figure 3: Example of SP high dimensionality problem for a three-level SP. $l0$, $l1$ and $l2$ refer to the first, second and third SP levels, respectively.

more heavily weighted. For each sub-region, a histogram of the matches is created. When histograms for all regions at all levels are created, they are concatenated to form the final image representation. Fig. 3, shows an example of a three level SP. This results in a $(21 \times vocabularysize)$ dimensional histogram. This clarifies the SP high dimensional problem, and hence, the huge memory usage during the classification stage.

In the next section, we give a brief explanation about the Agglomerative Information Bottleneck *(AIB)* algorithm. Finally, we explain our two-stages SP compression techniques, namely, *feature* and *block* compression, respectively. We refer to our two-stages SP compression approach as *Compact and Adaptive Spatial Pyramid (CASP).*

3.1. Agglomerative Information Bottleneck Theory

In our work, we aim at tailoring the high dimensional *SP* histograms to discriminate between the different categories. Towards this objective, several works address the problem of compact vocabulary construction [30, 31, 32]. In particular, *AIB* [31, 33] provides a guideline for the compression of vocabularies.

The main goal of *AIB* is reducing the dictionary of visual words X required for representing the categories Y. Using terms from information theory, this means generating a compact set of words $\hat{X}$ from the original dictionary X so that the loss of mutual information:

$$I(X;Y) = D_{KL}\left[p(x,y) \| p(x)p(y)\right], \tag{4}$$

to the categories Y is minimal [33, 31]. The functional $D_{KL}\left[p\|q\right]$ is the *Kullback-Leiber* divergence, and the joint distribution $p(x,y)$ is estimated from the training set by counting the number of occurrences of each visual word x in each category y.

The information about x captured by y can be measured by the mutual information,

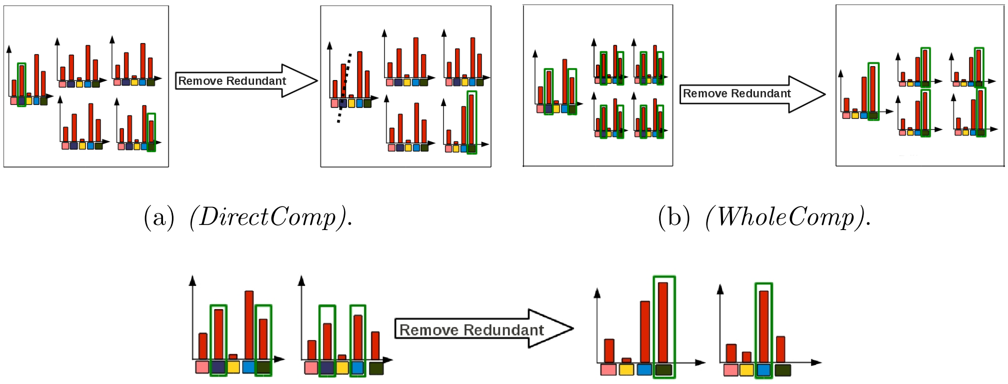
$$I(X,Y) = \sum_i \sum_t p(x_i, y_t) log \frac{p(x_i, y_t)}{p(x_i)p(y_t)}, \tag{5}$$

which measures the amount of information (discriminative power) I that one random variable carries about the other.

The merging of visual words is achieved by applying the *AIB* method [33]. In essence, *AIB* is applied by iteratively merging the two visual words x_i and x_j into $\hat{x}$ that causes the smallest decrease in the original mutual information.

3.2. Feature Compression

The usage of non-optimized dictionaries for building *SP* results in its huge dimensional histograms. Our first aim is to optimize the dictionaries in a way that maintains the original *SP* performance. For this purpose, we



(a) *(DirectComp).* (b) *(WholeComp).*

(c) *(LevelComp).*

Figure 4: Two highlighted words are the most similar ones to be merged (see text).

investigate the direct usage of the original *AIB* algorithm as proposed in [31] for the task of SP compression. We refer to this scheme as *DirectComp*.

In *DirectComp* strategy, two features in any SP region can be suggested to be fused. Since, these candidate features can be located at different SP regions, then different vocabularies for each SP region can be obtained, as shown in Fig. 4(a). Hence, discarding the important spatial property of *SP*.

Consequently, we propose two alternative spatial feature compression strategies, which we refer to them as *WholeComp* and *LevelComp*. In *WholeComp* strategy, we propose to remove the spatially least informative features within the whole *SP* levels simultaneously. Hence, the occurrence of a spatial word (x_{sp}) at index i over a three-level *SP* is measured as:

$$p(x_{sp_i}) = \sum_{j=1}^{J=21} p(x_{i+(j-1)\times v}), \tag{6}$$

where, j is the index of the current SP region, J is the total number of regions within a three-level SP and v is the vocabulary size. Fig. 4(b) shows an example of applying our *WholeComp* compression scheme on a two-level *SP*: where the original 5 features per region are reduced to 4 features for all the regions of the pyramid.

On the other hand, for *LevelComp* compression strategy, we propose to learn the most compact vocabulary $\hat{x}_{sp}$ that best suits each SP level by eliminating the occurrences of the least informative features from each *specific* SP level l, as shown in Fig. 4(c). For instance, for a three level pyramid, we first eliminate from its third level (i.e., $l = 2$) the least informative visual word from all its sixteen spatial occurrences. Subsequently, we eliminate from its second level (i.e., $l = 1$) the least informative visual word from all its four spatial occurrences, etc. To this end, the probability of a spatial visual word $p(x_{sp_i})$ at level l is then computed as follows:

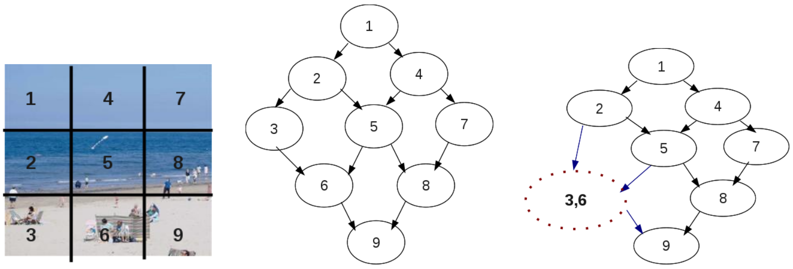
$$p(x_{sp_i}) = \begin{cases} p(x_i) & \text{if } l = 0. \\ \sum_{j=1}^{2^{2l}} p(x_{(i+j\times v)}) & \text{if } l > 0, \end{cases} \tag{7}$$

Where, i indicates the index of the spatial word x_{sp_i}, j indicates a specific region index at level l and v is the vocabulary size. Finally, we use the information content criteria to measure the discriminative power of the spatial vocabulary X_{sp} as follows:

$$I(X_{sp}, Y) = \sum_i \sum_t p(x_{sp_i}, y_t) log \frac{p(x_{sp_i}, y_t)}{p(x_{sp_i}) p(y_t)}. \tag{8}$$

3.3. Block Compression

Our last contribution, is concerned with the assumption that the hierarchical approach proposed by [16] with a set of regular grids of increasing density is inappropriate for scenes. In this section, we propose a method



(a) (b) (c)

Figure 5: Block Compression Example. (a) Given an input (3×3) block image. (b) Each block is represented by a node in a decision tree. (c) We calculate the discriminative power of each possible merging. See text for details.

for learning a proper spatial split-up that best suits each category. For this purpose, we adopt the original AIB such that we relate each block (b_k) in level (l) with the data set categories. The probability $p(b_k)$ of each block (b_k) is formulated as:

$$p(b_k) = \sum_{k=1}^{2^{2l}} \sum_{v=(k-1)s+1}^{ks} p(\hat{x}_{sp_v}), \tag{9}$$

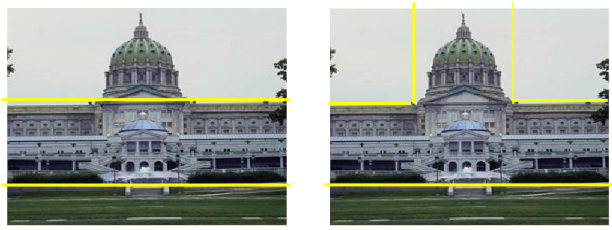
where s is the size of the compact vocabulary, k indicates the current block index within level l and v refers to the current vocabulary index within block k. In essence, $p(b_k)$ is calculated by summing up the probabilities of the compact vocabularies they contain. In order to fuse the least informative blocks, we evaluate the discriminative power of each block as follows:

$$I(X,Y) = \sum_k \sum_t p(b_k, y_t) log \frac{p(b_k, y_t)}{p(b_k)p(y_t)}. \tag{10}$$

Fig. 5 visualizes our block fusion approach. The Decision Tree *(DT)* shown in Fig. 5(b) represents our input image in Fig. 5(a). Each node in DT is equivalent to an image block, while arrows indicate neighboring blocks that are candidates for fusion. Each block can be merged with either its right or its bottom neighboring block (if any). Initially, all the adjacent blocks indicated by arrows are considered for fusion. However, the actual fusion occurred is between b_3, and b_6 as it caused the minimum loss in discriminative power, see Fig. 5(c). As a result, b_3 is updated to $b_{(3,6)}$, and the neighbors of both b_3 and b_6 are inherited (i.e. b_2, b_5, b_9). Thus, a dimensionality reduction is achieved by removing b_6 from DT. This iterative procedure results in generating adaptive shapes, and it terminates when all blocks are merged. Hence, it converges when it reaches the standard BoW representation. Fig. 8 shows a visual explanation for the whole process.

Lastly, we propose two approaches for learning the adaptive pyramid shapes, namely *Global Pyramid Shapes (GPS)* and *Class-specific Pyramid Shapes (CPS)*. In *GPS*, instead of having a fixed rigid shape for representing any task as in [16], we propose to learn the adaptive pyramid shape that best suits all the data set categories. For *CPS*, instead of learning the adaptive shape across all the data set categories, we propose learning the most suitable shape for each category separately by optimizing the classification performance for each category versus the rest.

Fig. 6 shows an example of learning the most suitable shape using the proposed approaches. Fig. 6(a), shows the globally learned shape that suits all the categories. On the other hand, Fig. 6(b) shows the learned shape specifically tailored for the *inside city* category. This shows the importance of developing specific image representations for the task at hand.



(a) (b)

Figure 6: (a) Our Global Compact, Adaptive Spatial Pyramid (CASP) learning scheme converges to the Ad-Hoc SP proposed in [34], vs. (b) Our Class-Specific (CASP).

4. Experimental Setup

In this section, we describe the data sets and the implementation details used within our experiments.

4.1. Data Sets

We use three standard scene recognition data sets are used to evaluate our approach:

- Vogel and Schiele (**VS**) data set [9] includes 7 natural scenes consisting of 6 categories: 142 coasts, 111 rivers/lakes, 103 forests, 131 plains, 179 mountains, and 34 sky/clouds. Every scene category is characterized by a high degree of diversity, and potential ambiguities.

- Oliva and Torralba (**OT**) data set [35] includes 2688 images classified as 8 categories: 360 coats, 328 forest, 374 mountain, 410 open country, 260 high way, 308 inside of cities, 356 tall buildings, 292 streets.

- Quattoni and Torralba (**QuT**) indoor scene data set [19] is a recent data set characterized by 67 indoor categories with high intra-class variations, since the classification of indoor scenes are very challenging.

4.2. Implementation Details

In this section, we discuss the implementation details. For the purpose of classification, we use a multiple-scale grid detector. In the feature extraction step, we use various state of the art features. We use GIST descriptor [35] to represent the scene semantics. We use two standard color spaces, namely, HSV, and opponent *(OppC)* [28, 29] for obtaining color information. For capturing texture aspects, we use *TPLBP* descriptor [20]. We use both Opponent SIFT [28], and PHOG [18] descriptors for capturing both appearance, and shape aspects, respectively. For the vocabulary creation, we use a standard K-means for constructing vocabularies of size $1.5k$ as in [17]. We use a three-level SP, which results in $(1.5k \times 21 = 31.5k)$ dimensional histogram. Finally, we use a non-linear Support Vector Machine (SVM) classifier with χ^2 kernel.

To evaluate the classification performance, we use the mean of the diagonal values of the confusion matrix. This score is averaged over 10 trials, where training and testing samples are replaced randomly. For **VS**, and **OT** data sets, we follow the same learning protocol proposed in [36]. Hence, the data sets are split randomly into two separate sets of images, half for training and half for testing. From the training set, we randomly select 100 images to form a validation set.

5. Experiments

In this section, we provide experimental results to validate our proposed contributions. In section 5.1, we evaluate the performance of our proposed PC-TPLBP descriptor against several baseline LBP-based texture descriptors. In section 5.2, we demonstrate the performance of the proposed pyramid compression approaches. In section 5.3, experiments using multiple complementary spatial cues along with the proposed compression method are presented. Finally, in section 5.4, we compare our results with several state-of-the-art algorithms.

5.1. Evaluation of PC-TPLBP

In this section, we investigate the effect of texture features for scene recognition. We first compare the different LBP descriptors discussed in Sec. 2.1 over OT, and VS data sets. We also report the performance score of the *Gabor* descriptor, as a baseline.

For both **OT** and **VS** data sets, the classification scores are improved by 10.6% and 9.5% respectively based on *CSLBP descriptor* relative to the baseline. Using *FPLBP* descriptor, leads to a relative improvement of 15.0% and 14.3% on both data sets. Finally, *TPLBP* leads to a major relative improvement up to 25.8%, as shown in Table 1.

Table 1, also shows the importance of fusing color information with *TPLBP*. Compared to *TPLBP*, the combination of *TPLBP* with *oppC* (denoted as *C-TPLBP in OppC* in Table 1) yields a relative improvement of 1.0% and 0.7% on both **OT** and **VS** data sets, respectively. On the other hand, the combination of *TPLBP* with *HSV* (denoted as *C-TPLBP in HSV*) yields a relative performance increase up to 2.4% and 3.0% on both data sets. This demonstrates the importance of *C-TPLBP* over *TPLBP*. However, less gain is obtained using *C-TPLBP in OppC*.

Lastly, we examine the effect of incorporating spatial information to *C-TPLBP* descriptor. For both **OT** and **VS** data sets, *PC-TPLBP* improves the performance by around 3.0% and 2.5% respectively, relative to *C-TPLBP*. We then conclude that both color, and spatial information cues play an important role for scene recognition.

Method	OT	VS
Gabor	66.0	65.2
CSLBP	73.0 (+10.9%)	71.4 (+9.5%)
FPLBP	76.0 (+15.0%)	74.5 (+14.3%)
TPLBP	83.0 (+25.8%)	82.0 (+25.8%)
C-TPLBP in OppC	83.8 (+1.0%)	82.6 (+0.7%)
C-TPLBP in HSV	85.0 (+2.4%)	84.5 (+3.0%)
PC-TPLBP	**87.0** (+**3.0**%)	**86.6** (+**2.5**%)

Table 1: Classification Score using different LBP operators. TBLBP improves by 25.8% relative to the baseline "Gabor". colored-TPLBP improves by 2.4% relative to the best performing texture descriptor "TPLBP". PC-TPLBP improves by 3.0% relative to best performing colored-TPLBP descriptor, see text.

5.2. Evaluation of CASP with PC-TPLBP

In this section, we demonstrate the benefits of our two-stages compression strategy. Based on the empirical evaluation, we determined the best compression scheme over OT data set, and continued the rest of the experiments using the same configuration.

5.2.1. Evaluation of Feature Compression with PC-TPLBP

As a baseline, we directly apply the original *AIB* on our *PC-TPLBP* (denoted as *DirectComp*). We then examine our feature compression approaches described in Sec. 3.2), namely: *WholeComp*, and *LevelComp*.

Table 2 shows a major loss in the performance by around 5.7% (relative to the original SP performance) by using the *DirectComp* scheme. We attribute this performance degradation to the fact that *AIB* is greedy in its nature. When *AIB* suggests fusing two features, it just looks greedily all over the SP features which minimizes the overall loss in its discriminative power. As explained earlier, these two candidate features can be from different SP regions, which in turn leads to obtaining different vocabularies within the SP regions. Hence, discarding the important spatial property of SPs. Subsequently, this results in dropping the final SP performance.

The quantitative results in Table 2 also show that both of our spatially enhanced feature compression schemes outperform the *DirectComp* method. A minor performance loss by around 2.1% (relative to the original SP performance) is obtained by using our *WholeComp* compression scheme. *LevelComp* is the best performing scheme; as it preserves the original SP performance of 87.0%, while reducing the dimensionality significantly.

Moreover, for *LevelComp*, we show that there is a strong relation between the vocabulary size, and the SP level of concern, see Fig. (7). In other words, the finer the *SP* level, the fewer the number of features required to represent it, while maintaining its accuracy. Hence, for a three-level SP a vocabulary of size $0.6k$ is sufficient. While, for a two-level SP a vocabulary of size $0.8k$ is needed. However, for coarser *BoW* representation, a vocabulary of size $1k$

Method	Size	Score
Original	$31.5k$	87.0
DirectComp [31]	$14k$	82.0 (-5.7%)
WholeComp	$14k$	85.2 (-2.1%)
LevelComp	$14k$	87.0 ($\pm 0\%$)
GPS	$6k$	89.5 ($+2.9\%$)
CPS	$< 6k$	**90.6** ($+4.1\%$)

Table 2: Classification Score on OT data set to compress a SP of size $31.5k$ to a $14k$ one using $PC - TPLBP$, see text for details.

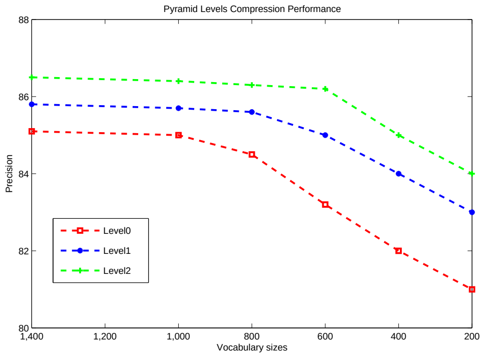


Figure 7: Learning specific vocabulary compression per SP level. See text for details.

is required.

5.2.2. Evaluation of Block Compression with PC-TPLBP

In this section, we examine the usage of our block compression scheme proposed in Sec. 3.3. The quantitative results reported in Table 2 show the successfulness of our proposed scheme, in terms of accuracy and dimensionality reduction.

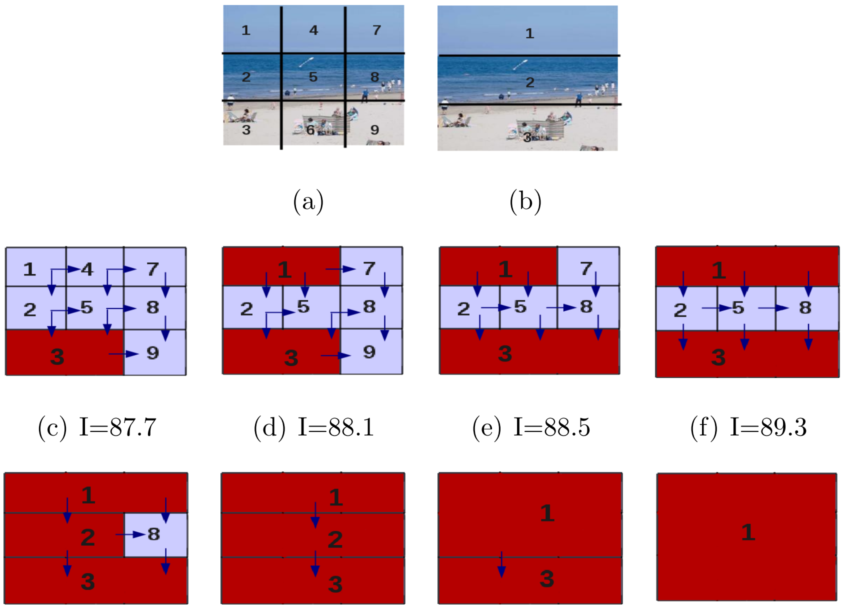
GPS reduces the SP dimensionality up to $6k$ ($1k$+$0.8k$×4+$0.6k$×3), while improving the original SP performance by around 2.9%. Fig. (8) shows an example of the adaptive shapes obtained after each iteration. Interestingly, our approach recommends the usage of the horizontal 3×1 shape for scene recognition. Thus, theoretically justifying the better performance obtained using the ad-hoc horizontal 3×1 SP proposed by Marszalek et al. [34] over the standard SP proposed by Lazebnik et al. [16]. We also demonstrate in Fig. 8(j) the convergence of our approach.

The last row in Table 2 shows that the performance of *CPS* scheme improves over *GPS*. Moreover, a significant performance improvement of 4.1% is achieved relative to the original SP performance. For dimensionality comparison, we use the notion $< 6k$ to indicate the upper bound of *CPS* dimensionality, since it varies per category, see Fig. 9. For instance, for the *coast* category, we obtain a $6k$-dimensional histogram $(1k + [0.8k \times 4] + [0.6k \times 3])$. While, for the *forest* category, a $5.4k$-dimensional histogram $(1k + [0.8k \times 4] + [0.6k \times 2])$ is obtained. Fig. (9) shows the learned shapes obtained for each category using our *CPS* scheme.

In conclusion, our feature, and block compression stages, which we refer to them as *CASP* are both necessary for obtaining compact, yet efficient SP.

5.3. Combining Multiple Cues using CASP

In this section, we investigate the importance of fusing spatial texture, shape, and appearance features besides the global image cues using our *CASP* representation. To this end, we use CASP of *PC-TPLBP* for capturing texture aspects. For appearance features, we use CASP of Opponent-SIFT features (denoted as PC-SIFT). For shape features, we extend the PHOG descriptor to incorporate color information motivated by [17, 28, 37, 38].



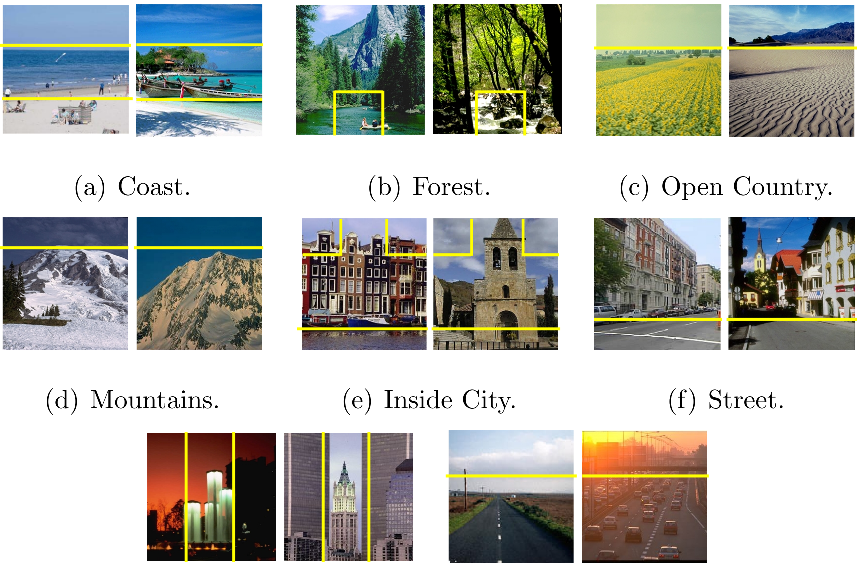
(a) (b)

(c) I=87.7 (d) I=88.1 (e) I=88.5 (f) I=89.3

(g) I=89.5 (h) **I=91.1** (i) I=89.0 (j) I=86.4

Figure 8: Top left grid in blue represents the (3 × 3) grid for an input image. Our *GPS* fuses the most similar blocks (depicted as red). The scores (I) are optimized over all the categories. Fig. 8(h) shows the successfulness of the 3 × 1 shape for scene recognition.

Table 3 demonstrates that coloring *PHOG* is beneficial for our task. Compared to PHOG, a relative performance improvement by 2.8% and 2.6% on **OT** and **VS** data sets, respectively, is obtained by fusing *PHOG with OppC*. However, a major performance improvement of 4.7% and 5.6% is achieved by fusing *PHOG with HSV* (denoted as *PC-HOG*). The results also show that our CASP compression approach (denoted as CASP-CHOG) improves the relative performance by 7.2% and 8.2% for both data sets. Similar behavior is obtained for appearance features, where a performance improvement of 2.0% and 2.1% is obtained by using our CASP compression scheme (denoted as CASP-CSIFT). Table 3 also shows the importance of fusing *CASPs* of



(a) Coast.

(b) Forest.

(c) Open Country.

(d) Mountains.

(e) Inside City.

(f) Street.

(g) Tall Building.

(h) Highway.

Figure 9: Examples of our *CPS* adaptive shapes learned over OT data set.

shape, and appearance cues. Compared with the best performing single descriptor (PC-SIFT), a significant improvement of 3.6% and 3.5% is achieved on both data sets.

In Table 4, we use the notion Local Descriptors (LD) to refer to CASP that uses pixel-based statistics features (PC-HOG, and PC-SIFT). Furthermore, we use the notion Regional Descriptors (RD) to refer to CASP that uses patch-based statistics features (PC-TPLBP). Lastly, we use the notion Global Descriptors (GD) for features that capture the global image semantics (GIST). The quantitative results in Table 4 illustrate the importance of combining (i) GD with LD as demonstrated in [19], (ii) RD with GD, (iii) RD with LD using our *CASPs* representation, and, finally (iv) GD, RD, and LD. Table 4 also shows that the *CPS* learning scheme improves the perfor-

Method	OT		VS	
	Size	Score	Size	Score
PHOG [18]	31.5k	79.5	31.5k	78.2
PHOG + OppC	31.5k	81.7 (+2.8%)	31.5k	80.2 (+2.6%)
PHOG + HSV *(PC-HOG)*	31.5k	83.2 (+4.7%)	31.5k	82.6 (+5.6%)
CASP-CHOG	6k	85.2 (+7.2%)	6k	84.6 (+8.2%)
PC-SIFT [28]	31.5k	88.4	31.5k	87.7
CASP-CSIFT	6k	90.2 (+2.0%)	6k	89.5 (+2.1%)
CASP-CHOG&CSIFT	12k	**91.6 (+3.6%)**	12k	**90.8(+3.5%)**

Table 3: Classification scores of (i) Fusing PHOG with HSV (denoted as PC-HOG) outperforms that of OppC. (ii) Combining CASPs of PC-HOG, and PC-SIFT.

mance over the *GPS* by 2%, while reducing the dimensionality to less than $< 18.4k$.

5.4. Comparison with State-of-the-Art

In this section, we evaluate the performance of our approach with state-of-the-art methods on OT, VS, and QuT data sets. Table 5, summarizes and compares these results. For **OT**, our best score using our approach which exploits the fusion of complementary CASPs of *(LD + RD + GD)* is 97.4%. The obtained result excels state-of-the-art score 92.8% on this data set [15, 16, 17, 35]. For **VS** data set, we achieve a score of 96.2%, which outperforms the best reported result 90.3% on this data set [16, 17, 9]. In **Indoor**67, our best score is 48.9%, which exceeds state-of-the-art score 45.5% for this data set [39].

Features for CASP	OT		VS	
	Size	Score	Size	Score
GD	0.4k	83.7	0.4k	82.9
RD	6k	89.5	6k	88.8
LD	12k	91.6	12k	90.8
GD + RD	6.4k	91.0	6.4k	90.2
GD + LD	12.4k	92.8	12.4k	92.0
LD + RD	18k	93.5	18k	92.5
GPS with LD+RD+GD	18.4k	95.2	18.4k	94.2
CPS with LD+RD+GD	$< 18.4k$	**97.4**	$< 18.4k$	**96.2**

Table 4: Experimental results with *CASP* demonstrate that combining shape, appearance (*LD*), texture (*RD*) with global cues (*GD*) improves the performance significantly. See text for details.

Method	OT	VS	QuT
Vogel et al.[9]	-	75.1	-
Oliva et al. [35]	83.7	-	-
Bosch et al. [17]	86.6	85.7	-
Perina et al. [15]	92.8*	90.3*	-
Quattoni et al. [19]	-	-	25.0
Nakayama at al. [39]	-	-	45.5*
CPS with LD+RD+GD	**97.4**	**96.2**	**48.9**

Table 5: Comparison with state-of-the-art. * indicates the best reported state-of-the-art results.

6. Conclusion and Future Work

In this paper, we proposed a novel and efficient texture descriptor based on patch-based texture features *TPLBP*. For this purpose, we incorporated both color information *C-TPLBP*, and spatial information *PC-TPLBP* to *TPLBP*.

Furthermore, we addressed the high dimensionality problem of the generated SP histograms. We introduced a novel SP compression approach, which works on two stages. The first compression stage is done within the SP features. We eliminated the spatially least informative features for each SP level. We also showed that there is a strong relation between the compact vocabulary size, and the SP level in concern: the finer the level, the fewer the required words for representing it. The second compression stage is done within the blocks of each level. We further introduced two alternative approaches, namely, *GPS* and *CPS* for learning the best SP block partitioning. Regarding *GPS* scheme, we justified theoretically the better performance of the ad-hoc horizontal h3 × 1 pyramid [34] over the traditional one [16] for the task of scene recognition. When the *CPS* scheme is considered, the resulting *CASP* representation maintains the performance of their original counterparts, while reducing the dimensionality significantly.

Finally, we showed the importance of combining the complementary patch-based texture features *(regional)* with the pixel-based shape and appearance ones *(local)*. In addition, we investigated the effect of fusing *global* image cues along with *regional*, and *local* ones, which resulted in improving the overall performance. Consequently, we conclude that *CASP-based* complementary descriptors, together with class-specific learning are all important for obtaining good performance. We evaluated the proposed framework on scene recognition task, and obtained state-of-the-art results on several scene recognition benchmark data sets.

For future work we are interested in applying *CASPs* to the task of *BoW-based* object detection [40, 41]. The application of *BoW-based* detection has been advanced due to the efficient sub-window search (ESS) algorithm proposed by Lampert et al. [40]. The usage of compact discriminative *SPs* to this application could help in obtaining faster detection methods without a significant loss in accuracy. Another line of future research includes investigating the application of our approach to video scenes, complementary to motion features which show excellent results recently [42, 43]. Therefore, we expect that combining the strengths of both methods will lead to further improvements.

Acknowledgments

The authors acknowledge the support of the Spanish Research Programs Consolider-Ingenio 2010: MIPRCV (CSD200700018); Avanza I+D ViCoMo (TSI-020400-2009-133) and DiCoMa (TSI-020400-2011-55); along with the Spanish projects TIN2009-14501-C02-01 and TIN2009-14501-C02-02.

References

[1] J. Alvarez, T. Gevers, A. Lopez, 3d scene priors for road detection, in: CVPR, 2010, pp. 57 –64.

[2] A. Torralba, Contextual priming for object detection, Inetnational Journal of Computer Vision. 53 (2) (2003) 169–191.

[3] N. Ikizler-Cinbis, S. Sclaroff, Object, scene and actions: Combining multiple features for human action recognition, in: ECCV (1), 2010, pp. 494–507.

[4] S. Oh, A. Hoogs, M. W. Turek, R. Collins, Content-based retrieval of functional objects in video using scene context, in: ECCV (1), 2010, pp. 549–562.

[5] A. Vailaya, M. Figueiredo, A. Jain, H. Zhang, Image classification for content-based indexing, IEEE Transactions on Image Processing 10 (1) (2001) 117–130.

[6] M. Szummer, R. W. Picard, Indoor-outdoor image classification, in: IEEE International Workshop on Content-Based Access of Image and Video Databases, 1998, pp. 42–51.

[7] J. Zhang, M. Marszalek, S. Lazebnik, C. Schmid, Local features and kernels for classification of texture and object categories: a comprehensive study, IJCV 73 (2) (June 2007) 213–238.

[8] K. Mikolajczyk, C. Schmid, A performance evaluation of local descriptors, PAMI 27 (10) (2005) 1615–1630.

[9] J. Vogel, , B.Schiele, Semantic modeling of natural scenes for content-based image retrieval., International Journal of Computer Vision 72 (2) (2007) 133–157.

[10] L. Fei-fei, P. Perona, A bayesian hierarchical model for learning natural scene categories, in: CVPR, 2005.

[11] L.-J. Li, R. Socher, L. Fei-Fei, Towards total scene understanding: Classification, annotation and segmentation in an automatic framework, in: CVPR, 2009.

[12] M. Marszalek, I. Laptev, C. Schmid, Actions in context, in: CVPR, 2009.

[13] Y.-L. Boureau, F. Bach, Y. LeCun, J. Ponce, Learning mid-level features for recognition, in: CVPR, 2010.

[14] N. Xie, H. Ling, W. Hu, X. Zhang., Use bin-ratio information for category and scene classification, in: CVPR, 2010.

[15] A. Perina, M. Cristani, U. Castellani, V. Murino, N. Jojic, A hybrid generative/discriminative classification framework based on free-energy terms, in: ICCV, 2009.

[16] S. Lazebnik, C. Schmid, J. Ponce., Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories., in: CVPR, 2006.

[17] A.Bosch, A.Zisserman, X.Munoz, Scene classification using a hybrid generative/discriminative approach, PAMI 30 (4) (2008) 712–727.

[18] A. Bosch, A. Zisserman, X. Munoz, Representing shape with a spatial pyramid kernel, in: CIVR, 2007.

[19] A. Quattoni, A. Torralba, Recognizing indoor scenes, in: CVPR, 2009.

[20] L. Wolf, T. Hassner, Y. Taigman, Descriptor based methods in the wild, in: Faces in Real-Life Images Workshop in ECCV., October 2008.

[21] L. Yeffet, L. Wolf, Local trinary patterns for human action recognition, in: ICCV, 2009.

[22] T. Ojala, M. Pietikinen, T. Menp, Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, PAMI 24 (7) (2002) 971–987.

[23] T. Ahonen, A. Hadid, M. Pietikinen, Face description with local binary patterns: Application to face recognition., PAMI 28 (2006) 2037–2041.

[24] M. Heikkil, M. Pietikinen, C. Schmid, Description of interest regions with local binary patterns, Pattern Recogn. 42 (3) (2009) 425–436.

[25] M. Heikkil, M. Pietikinen, A texture-based method for modeling the background and detecting moving objects, IEEE Transactions on Pattern Analysis and Machine Intelligence 28 (4) (2006) 657–662.

[26] P. V. Gehler, S. Nowozin, On feature combination for multiclass object classification, in: ICCV, 2009.

[27] T. Menp, M. Pietikainen, Classification with color and texture: jointly or separately?, Pattern Recognition 37 (8) (2004) 1629–1640.

[28] K. E. A. van de Sande, T. Gevers, C. G. M. Snoek, Evaluating color descriptors for object and scene recognition, PAMI 32 (9) (2010) 1582–1596.

[29] J. van de Weijer, T. Gevers, Robust optical flow from photometric invariants., in: ICIP, 2004, pp. 1835– 1838.

[30] J. Winn, A. Criminisi, T. Minka, Object categorization by learned universal visual dictionary, in: ICCV, 2005.

[31] B. Fulkerson, A. Vedaldi, S. Soatto, Localizing objects with smart dictionaries, in: ECCV, 2008.

[32] S. Lazebnik, M. Raginsky, Supervised learning of quantizer codebooks by information loss minimization., PAMI 31 (7) (2009) 1294–1309.

[33] N. Slonim, N. Tishby, Agglomerative information bottleneck, in: NIPS, 1999.

[34] M. Marszalek, C. Schmid, H. Harzallah, J. van de Weijer, Learning object representation for visual object class recognition 2007, in: Visual recognition Challenge Workshop in conjuncture with ICCV, 2007.

[35] A. Oliva, A. Torralba, Modeling the shape of the scene: A holistic representation of the spatial envelope, IJCV 42 (2001) 145–175.

[36] A. Bosch, A. Zisserman, X.Munoz, Scene classification via plsa, in: ECCV, 2006.

[37] W. R. Schwartz, A. Kembhavi, D. Harwood, L. S. Davis, Human detection using partial least squares analysis, in: ICCV, 2009.

[38] S. Ito, S. Kubota, Object classification using heterogeneous co-occurrence features, in: ECCV, 2010.

[39] H. Nakayama, T. Harada, Y. Kuniyoshi, Global gaussian approach for scene categorization using information geometry, in: CVPR, 2010.

[40] C. Lampert, M. Blaschko, T. Hofmann, Beyond sliding windows: Object localization by efficient subwindow search, in: CVPR, 2008.

[41] H. Harzallah, F. Jurie, C. Schmid, Combining efficient object localization and image classification, in: ICCV, 2009.

[42] N. Shroff, P. Turaga, R. Chellappa, Moving vistas: Exploiting motion for describing scenes, in: CVPR, 2010, pp. 1911 –1918.

[43] K. Kim, M. Grundmann, A. Shamir, I. Matthews, J. Hodgins, I. Essa, Motion fields to predict play evolution in dynamic sport scenes, in: CVPR, 2010, pp. 840 –847.