Utilizing Overlapping Windows in Spatial Pyramid Matching

Kristo, Chua Chin Seng
School of Electrical and Electronic Engineering
Nanyang Technological University (NTU) Singapore
50 Nanyang Avenue, S2.1-B4-01, Singapore 639768
kkristo1@e.ntu.edu.sg, ecschua@ntu.edu.sg

Abstract

Spatial Pyramid Matching (SPM) has been a major breakthrough in the field of object and scene recognition. Using this approach, an image is divided into $2^l \times 2^l$ disjoint sub-windows for each pyramid level l. However, the disjoint arrangement of the sub-windows can be too restrictive, especially when we consider that each window may benefit from broader context. Inspired by human habit in viewing an image, we introduced two overlapping spatial windows in this paper: rectangular overlapping windows (OWSPM) and circular overlapping windows (CWSPM). We found that the introduction of overlapping spatial window allows us to achieve better performances in Caltech 101, Caltech 256 and 15-Scene databases (up to 3.68% *compared to traditional SPM using ScSPM algorithm). Furthermore, it enables us to bypass the 0^{th} and 1^{st} layer and use the 2^{nd} pyramid layer directly for recognition, considerably cutting memory consumption while achieving better recognition rate than traditional SPM at the same time.*

1 Introduction

Spatial Pyramid Matching [1] was proposed as an extension for the traditional *bag-of-features* (BoF) approach to incorporate spatial configuration of an image. Each image is now described using not only one histogram but a concatenation of multiple histograms using the concept of spatial pyramid. The l^{th} ($l \in \{0, 1, ..., L\}$) level of the pyramid is formed by dividing the image I into $2^l \times 2^l$ disjoint sub-windows ($L = 2$ is usually used by most researchers). A histogram is extracted from each sub-window to be concatenated with histograms from other sub-windows. This image representation may grow into a very large vector as more layers are considered.

We propose an extension to this traditional SPM approach by introducing two types of overlapping windows (rectangular and circular) to replace the disjoint sub-windows. The idea behind this concept is that in human perception, it is very rare for us to examine an image or a scene in a disjoint way. More often than not, we examine an image using regions that are likely to be overlapping with each other. In addition, by utilizing overlapping sub-windows, it is more probable that they will enclose larger part of the object to be observed. This will lead to increased discriminability of the image representation. The proposed concept is tested using variety of popular databases (Caltech 101, Caltech 256, and 15-Scene) using ScSPM framework developed in [2] as the baseline.

Our experiments have shown that the introduction of overlapping spatial windows improves the recognition rate of all datasets considerably (up to 3.68%). Furthermore, as we will demonstrate in this paper, the introduction of overlapping spatial windows improves the performance of the $l = 2$ layer significantly, exceeding the performance of traditional SPM with three pyramid layers. This enables us to bypass the first two layers entirely and use only the $l = 2$ layer, cutting memory cost by 24%.

Our contributions are thus summarized in three points: (1) introducing OWSPM and finding the optimal overlap size to increase the recognition rate of traditional SPM, (2) introduction of overlapping circular window SPM (CWSPM), and (3) using CWSPM to bypass the first two layers of traditional SPM. The rest of the paper is organized as follows: Sec. 2 talks about some related works. Sec. 3 presents the framework of our proposed method. Sec. 4 discusses on the implementation of the system in our experiments, followed by experiment results in Sec. 5. Finally, Sec. 6 concludes our paper.

2 Related Works

In the past decade, BoF based approach has been a popular approach in object recognition research. Lazebnik et al. [1] proposed to extend the BoF model by including the spatial configurations of local patches. The addition of spatial pyramid representation allows spatial configurations to be captured in the image representation. Because of its success, it has been adopted in many subsequent works as a major block of recognition systems.

As research on object recognition progressed, quantization of local patch descriptors into discrete vocabulary has shifted from *hard-assignment* (where one local patch is assigned to one vocabulary) to *soft-assignment* (a local patch is now assigned as to several vocabularies by a membership indicator) [3, 4]. Setting the coefficients in the soft-assignment to have only a few non-zero components (sparse-coding) proves to be very powerful when paired with SPM. The combination of the two allows the classifier to be learned using a simple linear SVM, as opposed to the costly non-linear SVM from the traditional SPM approach. This concept is called ScSPM [2], and it has since been an integral part of several cutting edge object and scene recognition approach [5, 6].

The usage of disjoint sub-window in SPM, however, is challenged only by few researchers. Ergul and Arica [7] used half-size overlapping spatial window for scene recognition, but retained the window size for each pyramid level, leading to the increased number of sub-blocks, leading to the increase of memory cost (storage of image representations tripled as 59 sub-blocks are now used in computing the features, compared to 21 of normal SPM when $L = 2$). Yan et al. [8] used dense spatial sampling to replace SPM with variable sub-block size. While this dense spatial sampling led to the usage of overlapping spatial window, its memory complexity increased considerably as many more blocks are involved.

To the extent of our knowledge, our work is the first to propose the usage of overlapping spatial window while retaining the same storage and computational cost.

3 Overlapping Spatial Pyramid Matching

3.1 Feature Pooling by OWSPM

Let $\mathbf{X}$ be a collection of local patch descriptors in a D-dimensional feature space consisting of M descriptors, i.e. $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, ..., \mathbf{x}_M]^T \in \mathbb{R}^{M \times D}$. ScSPM learns a *codebook dictionary* $\mathbf{V} \in \mathbb{R}^{D \times K}$ and a membership indicators of each local patch descriptor $\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, ..., \mathbf{u}_M]^T$ by optimizing the following objective function:

$$\begin{aligned} \min_{\mathbf{U},\mathbf{V}} \sum_{m=1}^{M} \|\mathbf{x}_m - \mathbf{u}_m \mathbf{V}\|^2 + \lambda |\mathbf{u}_m| \\ \text{subject to} \quad \|\mathbf{v}_k\| \leq 1, \ \forall k = 1, 2, ..., K. \end{aligned} \tag{1}$$

By putting an additional $L1$-norm regularization on $\mathbf{u}_m$, we enforce $\mathbf{u}_m$ to have a small number of nonzero elements, thus forcing coefficients in $\mathbf{u}_m$ to have few non-zero elements. Unit $L2$-norm constraint on $\|\mathbf{v}_k\|$ is generally applied to avoid trivial solutions. The *codebook* $\mathbf{V}$ is normally designed to be *overcomplete* by setting $K > D$. The optimizing process for Eq. 1 can be found in [9].

Instead of pooling the local patches by averaging, ScSPM proposed the usage of *max-pooling* approach to calculate the image representation. We calculate $\mathbf{z}$, the feature representation of a particular sub-window, using the max-pooling approach by:

$$z_j = \max\{|u_{1j}|, |u_{2j}|, ..., |u_{Mj}|\} \tag{2}$$

In this equation, z_j denotes the j-th element of $\mathbf{z}$, while u_{mj} denotes the j-th element of $\mathbf{u}_m$. Using the traditional SPM approach with $L = 2$, each level of the pyramid is divided into $2^l \times 2^l$ disjoint sub-windows with $l = \{0, 1, ..., L\}$. Pooling method is then applied to each sub-window, producing $\sum_{l=1}^{L} 4^l$ vectors $\mathbf{z}$. These vectors are then concatenated into a single feature vector as the representation of image I.

We propose the use of overlapping spatial window for all layers with $l > 0$ (since we don't need to apply overlapping windows when $l = 0$ as it correspond to the full image I). The overlapping windows are implemented without changing the number of spatial window used to represent the image (in contrast to [7, 8]) but rather by changing the size of the sub-windows in each pyramid level. Let θ to be the fraction of overlap between two adjacent sub-window (we define adjacency in left/right and top/bottom fashion) compared to the area of a single window. Then, each window in pyramid level l will have a size of $(\frac{\text{image height}}{2^l(1-\theta)+\theta}) \times (\frac{\text{image width}}{2^l(1-\theta)+\theta})$. Max-pooling method is applied to each sub-window, and the resulting vectors are concatenated to produce the feature vector.

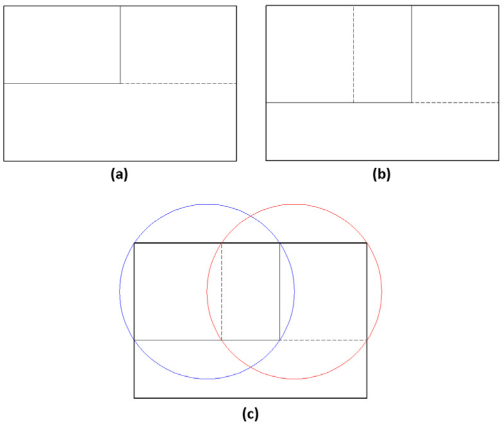

Figure 1. Sub-window division of (a) traditional SPM, (b) OWSPM with its overlapping rectangular windows, and (c) CWSPM with its overlapping circular windows. In this illustration, l and θ is set to 1 and 0.4 respectively and only the top two sub-windows are shown for clarity.

The usage of overlapping sub-windows takes its inspiration from how we as a human being gather information from what we see. We localize our focus on some interest region, but very rarely that there will be only a single interest region. Additionally, when multiple interest regions are considered, most of these regions will be overlapping with each other. However, traditional SPM method divides the spatial region disjointly, disregarding the potential benefits from this aspect. A further confusion may arise when disjoint windows are applied when we consider the case where an object is shared equally by two sub-windows. In this case, the two sub-windows may not represent the object effectively. By introducing overlapping sub-windows, we will be able to enclose more of the object which in turn increases the discriminability of image representation.

3.2 Circular Window SPM

We extend the concept further by replacing the pooling mechanism of OWSPM. The traditional and overlapping SPM divides the image into rectangular sub-windows with proportional sizes. Patches descriptors are pooled together based on their membership of sub-windows. As OWSPM aims to achieve a better coverage of context obtained by the sub-windows, it is intuitive that the rectangular shape will not be optimal for our purposes.

If we consider the rectangle's center of gravity as the focus point of a sub-window, then the farthest point that is being pooled by it is located at the rectangle's corner. If we want to include every patches within that distance from the focus point, a circular window is the obvious choice. By using a circular spatial window, the descriptive power of the image representation are improved, since: (1) the context are described completely

Table 1. Recognition rate of OWSPM under different θ values

	15 Scene	Caltech 101	Caltech 101
θ	100 train	15 train	30 train
0	80.28%	66.28%	72.46%
0.1	81.06%	66.74%	73.73%
0.2	81.19%	67.44%	**73.73%**
0.3	**81.54%**	**67.57%**	73.72%
0.4	81.09 %	67.49%	73.60%
0.5	80.76%	66.78%	73.72%

in every direction, and (2) circular windows will receive the same benefit with OWSPM as it require the sub-windows to overlap with each other.

We construct the circular windows by first defining the regular sub-windows of SPM. Circumcircles are constructed over each of this rectangular sub-window, creating circular windows with radius of $0.5 \times$ (window height^2 + window width^2) centered at the rectangle's center of gravity. Note that we can also define the rectangles to be overlapping with each other (as in OWSPM) to modify the overlap value of the circles.

4 Experiments

We implemented both OWSPM and CWSPM and use them to train the datasets using ScSPM as a benchmark. Three datasets are used in our experiments: 15 Scene, Caltech 101 and Caltech 256. Our implementations utilized only the SIFT descriptor, extracted from 16×16 pixel patches sampled regularly using a grid with 8 pixels spacing. For all database, we set codebook size K as 1024. All experiments are repeated 10 times and we report the mean of the recognition rates in this paper. These settings are similar to the settings used in [2] for direct comparison.

To allow concise writing, we introduce the notation of pyramid configurations $\mathbf{P}$ throughout this paper to denote the layers used in a particular experiment. In example, setting $\mathbf{P} = \{0, 1, 2\}$ means that we are using $l = 0$, $l = 1$, and $l = 2$ layers for our image representation (the layers with 1×1, 2×2, and 4×4 sub-windows respectively). We use the subscript o and c to indicate whether the layers is set using the overlapping rectangle sub-windows or the overlapping circle sub-windows, respectively. The absence of a subscript means that the layer follows traditional SPM definition.

5 Results

5.1 Testing of OWSPM and CWSPM

OWSPM and CWSPM are first tested using 15 Scene and Caltech 101 database to find the optimal value for θ. We tested both algorithms using $\theta = \{0, 0.1, 0.2, 0.3, 0.4, 0.5\}$ and show the results in Table 1 and Table 2. For OWSPM, θ value of 0.3 allows us to get the optimal result for all dataset under different number of training images ($\theta = 0.2$ produces the best performance for 30 training image Caltech 101, but only by 0.01% compared to when $\theta = 0.3$). The CWSPM exhibits its peak in when θ is around 0 to 0.1 (note that setting $\theta = 0$ in CWSPM will still lead to overlapping sub-windows). In our paper, we select $\theta = 0.3$ and $\theta = 0$ for OWSPM and CWSPM, respectively.

Table 2. Recognition rate of CWSPM under different θ values

	15 Scene	Caltech 101	Caltech 101
θ	100 train	15 train	30 train
0	**81.62%**	67.68%	**74.14%**
0.1	81.52%	**67.84%**	73.91%
0.2	81.06%	66.70%	73.83%
0.3	81.46%	66.92%	72.63%
0.4	80.84%	66.46%	71.85%
0.5	80.06%	64.95%	71.03%

Table 3. Recognition rate of ScSPM, OWSPM and CWSPM on 15-Scene.

Algorithms	100 training image
ScSPM [2]	80.28%
OWSPM	81.54%
CWSPM	**81.62%**

Using the obtained θ value, we compared the performance of ScSPM under three different image representation schemes: (1) traditional SPM, (2) OWSPM, and (3) CWSPM. Tables 3, 4, 5 shows the result for 15 Scene, Caltech 101, and Caltech 256 datasets, respectively.

As shown in the tables above, it is clear that the usage of OWSPM and CWSPM outperforms the traditional SPM (up to 3.68%) in its recognition rate. This is quite a significant increase considering that the effort on implementing these changes can be done with similar computational cost as the traditional SPM. In addition, CWSPM consistently outperforms OWSPM, even though only by a slight margin. These results have confirmed our claims that using disjoint sub-windows in spatial pyramid matching omits important information. It also confirm our hypothesis that including all equidistant local patches with respect to the rectangle's center of gravity captures the complete context surrounding the sub-windows. In addition, as ScSPM has been used as a basis for many *state-of-the-art* approach in image recognition, we strongly believe that the adoption of overlapping windows (be it OWSPM or CWSPM) may improve the results further.

5.2 The performance of $l = 2$ layer under OWSPM and CWSPM

A closer inspection of performance coming from each layer of ScSPM, OWSPM, and CWSPM provide us with another important discovery. In the traditional SPM with $L = 2$ ($\mathbf{P} = \{0, 1, 2\}$), we are using 21 sub-windows to represent each image. After coding of local patches and max-pooling, we will end up with $21K$ dimensional vector for each image. That is, when

Table 4. Recognition rate of ScSPM, OWSPM and CWSPM on Caltech 101.

Algorithms	15 train	30 train
ScSPM [2]	67.00%	73.20%
OWSPM	67.57%	73.72%
CWSPM	**67.68%**	**74.14%**

Table 5. Recognition rate of ScSPM, OWSPM and CWSPM on Caltech 256.

Algorithms	15 train	30 train	45 train	60 train
ScSPM [2]	27.73%	34.02%	37.46%	40.14%
OWSPM	31.31%	36.57%	39.15%	41.27%
CWSPM	**31.41%**	**36.59%**	**39.32%**	**41.50%**

Table 6. Average recognition rate of Caltech 101 database with 30 training images for various spatial pyramid configurations.

Pyramid Configurations	Recognition Rate
$\mathbf{P}_2 = \{2\}$	70.13%
$\mathbf{P}_{2o} = \{2_o\}$	73.19%
$\mathbf{P}_{2c} = \{2_c\}$	73.48%
$\mathbf{P}_{nw} = \{0, 1, 2\}$	72.46%
$\mathbf{P}_{ow} = \{0, 1_o, 2_o\}$	73.72%
$\mathbf{P}_{cw} = \{0, 1_c, 2_c\}$	74.14%

$K = 1024$ we will have 21504 dimensional vector for each image. Calculation of such data could be very costly both in terms of training complexity and memory complexity (using Caltech 256 with 60 training images will cost around 2.5 GB of memory to simply store the training data). A reduction on the dimensionality of image representation may benefit us greatly.

By inspecting each layer and their possible combinations, we discover that the usage of $l = 2$ layer provides us with a comparable result to the complete set of $\mathbf{P} = \{0, 1, 2\}$. As an example, Table 6 shows the recognition rate of Caltech 101 with 30 training images under selected pyramid configurations.

From these results we can see that while the performance of $\mathbf{P}_2$ is far from $\mathbf{P}_{nw}$ (both are using the traditional SPM) with 2.33% difference, OWSPM and CWSPM managed to shrink that difference into mere 0.53% ($\mathbf{P}_{ow} - \mathbf{P}_{2o}$) and 0.67% ($\mathbf{P}_{cw} - \mathbf{P}_{2c}$), respectively. Both displayed surprisingly good results and in fact exceed the results from the traditional SPM by 0.63% ($\mathbf{P}_{2o} - \mathbf{P}_{nw}$) and 1.02% ($\mathbf{P}_{2c} - \mathbf{P}_{nw}$). This result suggest that when memory allocation is limited, one can cut 24% of the memory cost by simply bypassing the $l = 0$ and $l = 1$ layer using OWSPM or CWSPM. By doing so, rather than using 21 sub-windows, we are only using 16 sub-windows for our image representation. These results are still consistent when tested on different datasets and training number, as shown in Table 7.

While it is clear that the complete pyramid gives the best results, our experiment shows that image representation using the $l = 2$ layer only is not falling too far behind, and may be used for a reasonable compromise where memory cost are of importance.

Table 7. Average recognition rate of Caltech 101 database, Caltech 256 database, and 15 scene database using various pyramid configurations.

Database	$\mathbf{P}_{nw}$ [2]	$\mathbf{P}_{2o}$	$\mathbf{P}_{ow}$
15-Scene (100 train)	80.28%	80.30%	81.54%
Caltech 101 (15 train)	66.28%	66.83%	67.57%
Caltech 101 (30 train)	72.46%	73.19%	73.72%
Caltech 256 (15 train)	27.73%	29.79%	31.31%
Caltech 256 (30 train)	34.02%	34.96%	36.57%
Caltech 256 (45 train)	37.46%	37.60%	39.15%
Caltech 256 (60 train)	40.14%	40.36%	41.27%

Database	$\mathbf{P}_{nw}$ [2]	$\mathbf{P}_{2c}$	$\mathbf{P}_{cw}$
15-Scene (100 train)	80.28%	80.48%	81.62%
Caltech 101 (15 train)	66.28%	67.28%	67.68%
Caltech 101 (30 train)	72.46%	73.48%	74.14%
Caltech 256 (15 train)	27.73%	30.83%	31.41%
Caltech 256 (30 train)	34.02%	35.19%	36.59%
Caltech 256 (45 train)	37.46%	38.86%	39.32%
Caltech 256 (60 train)	40.14%	40.38%	41.50%

6 Conclusion

We proposed two extensions to the traditional Spatial Pyramid Matching (SPM) throughout this paper by using the concept of overlapping spatial windows. The first proposal, called OWSPM, extends the rectangular sub-windows to be overlapping with each other without changing the number of sub-windows, while the second proposal, called CWSPM, extends OWSPM further by using the circumcircle of the rectangles. Our experiments shows that the average recognition rate for all datasets are increased with the introduction of OWSPM and CWSPM, where the circular windows SPM performs best out of all image representation method tested.

In addition, it has been shown that the usage of OWSPM and CWSPM opens up possibilities of bypassing the lower layers of traditional SPM to cut both computational and memory cost by 24%. As this concept are tested using ScSPM which has been a major building block of current *state-of-the-art* approach, we are confident that applying overlapping windows may give a significant contributions to the latest image recognition approaches.

References

[1] S. Lazebnik and A. Zimmerman: "Beyond bags of features: Spatial pyramid matching for recognition natural scebe categories" *Proc. of CVPR*, 2006.

[2] J. Yang, K. Yu, Y. Gong and T. Huang: "Linear spatial pyramid matching using sparse coding for image classification" *Proc. of CVPR*, 2009.

[3] Y. L. Boureau, F. Bach, Y. LeCun and J. Ponce: "Learning mid-level features for recognition" *Proc. of CVPR*, 2010.

[4] L. Liu, L. Wang and X. Liu: "In defense of soft-assignment coding" *Proc. of ICCV*, 2011.

[5] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang and Y. Gong: "Learning locality-constrained linear coding for image classification" *Proc. of CVPR*, 2010.

[6] S. Kong and D. Wang: "A dicitionary learning approach for classification: separating the particularity and the commonality" *Proc. of ECCV*, 2012.

[7] E. Ergul and N. Arica: "Scene classification using spatial pyramid of latent topics" *Proc. of ICPR*, pp.3603-3609, 2010.

[8] S. Yan, X. Xu, D. Xu, S. Lin, and X. Li: "Beyond spatial pyramids: a new feature extraction framework with dense spatial sampling for image classification" *Proc. of ECCV*, pp.473-487, 2012.

[9] H. Lee, A. Battle, R. Raina, and A. Y. Ng: "Efficient sparse coding algorithms" *NIPS*, 2006.