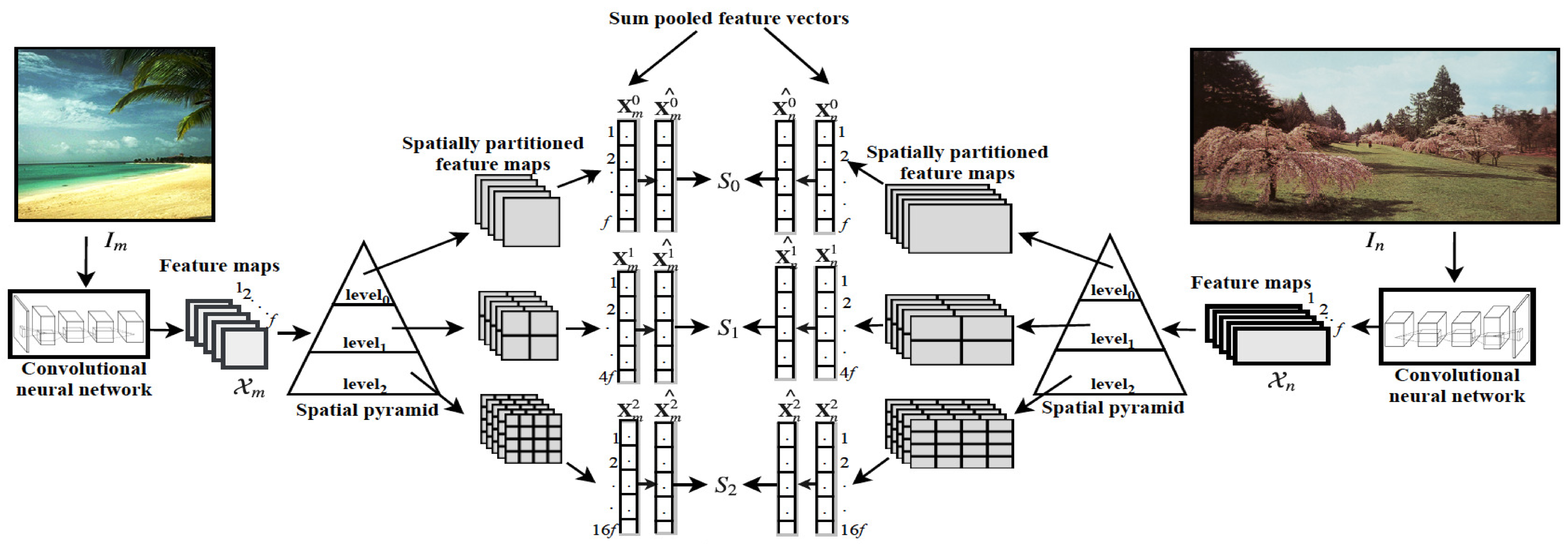
Scene Image Classification using Reduced Virtual Feature Representation in Sparse Framework

KRISHAN SHARMA   SHIKHA GUPTA   DILEEP A.D   RENU RAMESHAN
SCHOOL OF COMPUTING & ELECTRICAL ENGG., IIT MANDI, INDIA
{krishan_sharma, shikha_g}@students.iitmandi.ac.in, {addileep, renumr}@iitmandi.ac.in

INTRODUCTION 1

- Scene image datasets consist of thousands of different size images with size of the order of 10^6 pixels. Resizing these images to a fix size leads to loss of scene information.
- Images are fed to pre-trained CNN in their true size by considering the architecture only upto last convolutional pooling layer.
- A deep spatial pyramid matching kernel compute similarity score between two different size images using feature maps from last convolutional pooling layer of a pre-trained CNN.
- Reduced virtual features (RVFs) are extracted from the obtained kernel matrix and classification is performed in block sparse representation (**BSR**) framework.

DEEP SPATIAL PYRAMID MATCHING KERNEL(DSPMK) 2



$$K_{\mathrm{DSPMK}}(\mathcal{X}_m, \mathcal{X}_n) = \frac{1}{2^2}(S_0 - S_1) + \frac{1}{2}(S_1 - S_2) + S_2$$

ALGORITHM 1- $K_{DSPMK}(\mathcal{X}_m, \mathcal{X}_n)$ 3

Inputs:

(i) Feature maps set $\mathcal{X}_m$ and $\mathcal{X}_n$, where
$\mathcal{X}_m = \{\mathbf{x}_{m1}, ..., \mathbf{x}_{mi}, ..., \mathbf{x}_{mf}\}$; where $\mathbf{x}_{mi} \in \mathbb{R}^{m_p \times m_q}$
$\mathcal{X}_n = \{\mathbf{x}_{m1}, ..., \mathbf{x}_{ni}, ..., \mathbf{x}_{nf}\}$; where $\mathbf{x}_{ni} \in \mathbb{R}^{n_p \times n_q}$

(ii) L: number of pyramid levels.

1: **Procedure:**
2: **for** l=0 **to** $L-1$ **do**
3: Divide each feature map of $\mathcal{X}_m$ into 2^{2l} blocks.
$\mathcal{X}_m^l = \{\mathbf{x}_{m1(1)}^l ... \mathbf{x}_{m1(2^{2l})}^l, ..., \mathbf{x}_{mi(1)}^l ... \mathbf{x}_{mi(2^{2l})}^l, ..., \mathbf{x}_{mf(1)}^l ... \mathbf{x}_{mf(2^{2l})}^l\}$.
4: Apply sum pooling over each block such that
$x_{mi(j)}^l = \sum_u \sum_v \mathbf{x}_{mi(j)}^l(u, v)$
$\mathbf{X}_m^l = [x_{m1(1)}^l ... x_{m1(2^{2l})}^l, ..., x_{mi(1)}^l ... x_{mi(2^{2l})}^l, ..., x_{mf(1)}^l ... x_{mf(2^{2l})}^l] \in \mathbb{R}^{(2^{2l} \times f) \times 1}$.
5: ℓ_1- normalize the generated feature vector $\mathbf{X}_m^l$
$\hat{\mathbf{X}}_m^l = [\hat{x}_{m1(1)}^l ... \hat{x}_{m1(2^{2l})}^l, ..., \hat{x}_{mi(1)}^l ... \hat{x}_{mi(2^{2l})}^l, ..., \hat{x}_{mf(1)}^l ... \hat{x}_{mf(2^{2l})}^l] \in \mathbb{R}^{(2^{2l} \times f) \times 1}$.
6: Compute intermediate matching score using histogram intersection function
$$S_l = \sum_{j=1}^{f} \sum_{k=1}^{2^{2l}} min(\hat{x}_{mj(k)}^l, \hat{x}_{nj(k)}^l).$$
7: **end for**
8: Compute final matching score between $\mathcal{X}_m$ and $\mathcal{X}_n$
$$K_{DPSMK} = \sum_{l=0}^{L-2} \frac{1}{2^{(L-l-1)}}(S_l - S_{l+1}) + S_{L-1}.$$

Outputs:

(i) $K_{DSPMK}(\mathcal{X}_m, \mathcal{X}_n)$.

REDUCED VIRTUAL FEATURES 4

- Compute $\mathbf{K}_{train}$ and $\mathbf{k}(.,\mathcal{X}_{test})$ using kernel function $K_{DSPMK}(.,.)$ from ALGORITHM 1.
- Apply SVD decomposition over $\mathbf{K}_{train}$, $\mathbf{K}_{train} = \mathcal{U}\Sigma_N\mathcal{U}^\top$.
- Generate the reduced virtual features of dimension d ($d << N$)

$$\hat{\boldsymbol{\psi}}_{train}^d = {\Sigma_d}^{-\frac{1}{2}}\mathcal{U}^\top\mathbf{K}_{train},$$
$$\hat{\mathbf{y}}_{test}^d = {\Sigma_d}^{-\frac{1}{2}}\mathcal{U}^\top\mathbf{k}(.,\mathcal{X}_{test}),$$

where, $\Sigma_d = \Sigma_N(1:d, 1:N)$.

BSR BASED CLASSIFICATION 5

- We consider the dictionary formed from RVFs training data ($\hat{\boldsymbol{\psi}}_{train}^d \in \mathbb{R}^{d \times N}$) of all c scene classes. Block sparse representation for the test feature $\hat{\mathbf{y}}_{test}^d$ is obtained by solving:

$$\hat{\alpha} = \underset{\alpha}{\operatorname{argmin}} \quad \lambda \sum_{j=1}^{m} \|\alpha[j]\|_q + \|\hat{\mathbf{y}}_{test}^d - \hat{\boldsymbol{\psi}}_{train}^d\alpha\|_2^2$$

- Label for test signal $\hat{\mathbf{y}}_{test}^d$ is given by

$$label(\hat{\mathbf{y}}_{test}^d) = \underset{i=1,2,...,c}{\operatorname{argmin}} \|\hat{\mathbf{y}}_{test}^d - \hat{\boldsymbol{\psi}}_{train}^d\xi_i\|_2^2.$$

where,
$$\xi_i = \begin{cases} \hat{\alpha}[j], & \forall j \in i^{th}\text{class} \\ 0, & \text{otherwise} \end{cases}$$

DATASETS 6

(i) **Vogel Schiele dataset** consists of 6 semantic classes, namely, 'coast', 'river', 'forest', 'mountain', 'open-country' and 'sky-cloud' with total of 700 images.

(ii) **MIT-8 scene dataset** comprises of 8 scene classes, namely, 'tall building', 'street', 'inside-city', 'highway', 'coast', 'mountain', 'forest' and 'open-country' with total of 2688 images.

(iii) **MIT-67 dataset** is an indoor scene dataset with total of 15620 scene images having 67 classes. This is quite challenging dataset as interclass variation is very less.

(iv) **SUN-397 dataset** is a very huge dataset for scene classification with 397 classes including nature, indoor and urban categories.

CLASSIFICATION RESULTS 7

VGGNet-16 architecture pre-trained using	Vogel-Schiele		MIT-8 scene		MIT-67		SUN-397	
	$d=300, N=559$		$d=300, N=800$		$d=1000, N=5360$		$d=2000, N=19850$	
	$q=1$	$q=2$	$q=1$	$q=2$	$q=1$	$q=2$	$q=1$	$q=2$
ImageNet dataset	84.22	84.16	94.06	94.39	73.16	74.82	51.15	52.67
Places-205 dataset	84.23	**84.64**	94.82	95.00	78.81	**80.01**	58.92	59.73
Places-365 dataset	83.56	83.65	94.90	**95.11**	77.41	78.92	59.81	**60.63**

Classification accuracies using our proposed approach (DSPMK + RVFs + BSRC) on different datasets. Base features for the proposed method are extracted using VGGNet-16 which is pre-trained network on ImageNet, Places-205 and Places-365 datasets. d : RVF dimension, N: total training examples. Results are shown for BSRC with ℓ_q norm ($q = 1, 2$).

CONCLUSION 8

- A novel dynamic kernel known as deep spatial pyramid matching kernel (DSPMK) is proposed to generate kernel matrix.
- Reduced virtual features (RVFs) representation is obtained by diagonalizing the kernel matrix. Dictionary is built using the RVFs obtained from training images as atoms.
- Classification of test image is performed in sparse framework by imposing block sparsity constraint. The results obtained are better despite reduced size with the added advantage that no training is required.

REFERENCES 9

1. Ehsan Elhamifar and René Vidal, "Block-sparse recovery via convex optimization,"IEEE Transactions on Signal Processing, vol. 60, no. 8, pp. 4094-4107, 2012.
2. Kaiming He, Xiangyu Zhang, Shaoqing Ren and Jian Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," IEEE transactions on pattern analysis and machine intelligence, vol. 37, no. 9, pp. 1904-1916, 2015.
3. Bin-Bin Gao, Xiu-Shen Wei, Jianxin Wu, and Weiyao Lin, "Deep spatial pyramid: The devil is once again in the details," arXiv preprint arXiv:1504.05277, 2015.
4. Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in Computer vision and pattern recognition, 2006 IEEE computer society conference on. IEEE, 2006, vol. 2, pp. 2169-2178.
5. Kai Zhang, Liang Lan, Zhuang Wang, and Fabian Moerchen, "Scaling up kernel svm on limited resources: A low-rank linearization approach," in Artificial Intelligence and Statistics, 2012, pp. 1425-1434.
6. Bernhard Schölkopf and Alexander J Smola, Learning with kernels: support vector machines, regularization, optimization, and beyond, MIT press, 2002.
7. Mayank Juneja, Andrea Vedaldi, CV Jawahar, and Andrew Zisserman, "Blocks that shout: Distinctive parts for scene classification," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 923-930.