HDF: Hybrid Deep Features for Scene Image Representation

Chiranjibi Sitaula
School of IT
Deakin University
Geelong, Australia
csitaul@deakin.edu.au

Yong Xiang
School of IT
Deakin University
Geelong, Australia
yong.xiang@deakin.edu.au

Anish Basnet
Department of IT
Ambition College
Kathmandu, Nepal
anishbasnetworld@gmail.com

Sunil Aryal
School of IT
Deakin University
Geelong, Australia
sunil.aryal@deakin.edu.au

Xuequan Lu
School of IT
Deakin University
Geelong, Australia
xuequan.lu@deakin.edu.au

***Abstract*—Nowadays it is prevalent to take features extracted from pre-trained deep learning models as image representations which have achieved promising classification performance. Existing methods usually consider either object-based features or scene-based features only. However, both types of features are important for complex images like scene images, as they can complement each other. In this paper, we propose a novel type of features – hybrid deep features, for scene images. Specifically, we exploit both object-based and scene-based features at two levels: part image level (i.e., parts of an image) and whole image level (i.e., a whole image), which produces a total number of four types of deep features. Regarding the part image level, we also propose two new slicing techniques to extract part based features. Finally, we aggregate these four types of deep features via the concatenation operator. We demonstrate the effectiveness of our hybrid deep features on three commonly used scene datasets (MIT-67, Scene-15, and Event-8), in terms of the scene image classification task. Extensive comparisons show that our introduced features can produce state-of-the-art classification accuracies which are more consistent and stable than the results of existing features across all datasets.**

***Index Terms*—Deep learning, Feature extraction, Hybrid deep features, Image classification, Image representation, Machine learning.**

I. Introduction

With the fast development of camera technologies, image classification has been a fundamental problem in image processing. Solving it can benefit a variety of areas relying on images and videos, such as robotics, surveillance, forecasting, and so on. Image features are the mathematical representation of images. In general, there are three types of scene images features based on the sources of feature extraction. They are conventional computer vision based features [1]–[7], tag-based features [8]–[10], and deep learning based features [8], [11]–[18]. Conventional computer vision based methods [1]–[7] extract features based on the basic components of images such as texture, color, intensity, gradient, etc. They mainly focus on low-level features and lack details about the context in the images (e.g., objects and their spatial relationships). They are not suitable for complex images such as scene images that have intra-class dissimilarities and inter-class similarities. They work well with texture images.

Recent works [8]–[10] have used annotations of similar images available on the internet to extract tag-based features. Given an image, they first search similar images in the web and extract tag-based features from the descriptions of those similar images. These features are based on the contextual information of images. They did not use the content of images directly.

More recently, deep features [8], [11]–[18] extracted using pre-trained deep learning models have been widely used. They have been shown to work well in various image processing tasks including scene image classification, as they capture high-level semantic information of images. They used deep learning models such as VGG [19] pre-trained on datasets such as ImageNet [20] or Places [16] to extract features of objects or their background scenes. These techniques employed either object-based (foreground) features or scene-based (background) features. Still, these methods suffer from two problems on scene images. Firstly, existing approaches used either object-based or scene-based features only. For scene images, both object-based and scene-based features should be equally important. Secondly, most of these models are pre-trained on images having single objects such as ImageNet [20]. But many scene images contain multiples objects and discriminating regions. They may not be able to identify some interesting semantic regions in scene images. Some researchers adopted slicing techniques to partition images into smaller parts, and thus part image level features are extracted from image slices [21]. Whole image level features are also necessary for those scene images containing single objects or other discriminating information like background. Thus, both levels of images are important in extracting the scene features.

In this paper, we assume that features from whole image

978-1-7281-6926-2/20/$31.00 ©2020 IEEE

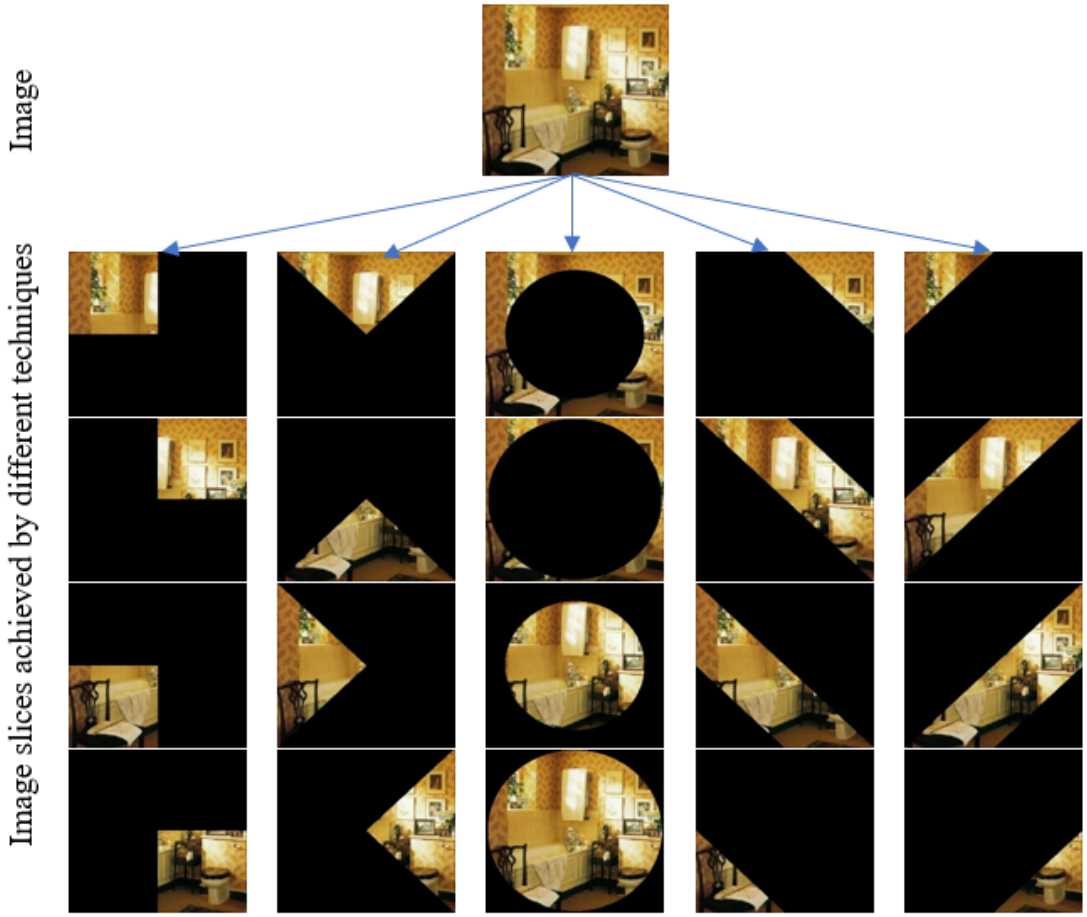

Fig. 1. Slices (or sub-images) achieved by using rectangular, triangular, circular, left diagonal cropping and right diagonal cropping techniques, respectively. (columns from left to right order.) We aggregate features of those slices to extract part image level features. Note that the two diagonal slicing techniques are introduced in this work.

level and part image level are both useful to identify different interesting regions in scene images, and propose to fuse four types of features – object-based and scene-based features from both whole images and part images, to construct our hybrid deep features (abbreviated as HDF). To get sub-images from an image, triangular, circular and rectangular slicing techniques have been presented in the literature [21]. To capture more interesting features from part images, we introduce two additional slicing techniques - left and right diagonal slicing (20 sub-images in total). We visually observe and speculate that such 20 sub-images for each scene image can provide decent semantic information. For example, in Fig. 1, the image slice at the bottom right corner can identify the object more easily compared to slices in the first three slices of the same row. The aggregation of deep features extracted from sub-images produced by five slicing techniques will construct deep features at part image level, which will complement features from the whole images level. We present some common aggregation operators and empirically select the concatenation operator due to its outstanding aggregation ability.

Extensive experiments in scene image classification on three commonly used benchmark datasets (MIT-67 [22], Scene-15 [23] and Event-8 [24]) validate the proposed hybrid deep features (HDF), and reveal that our HDF generate state-of-the-art classification accuracies which are more consistent and stable than the results of existing features across different datasets.

The remainder of this paper is organized as follows. Section II reviews related works in scene image representation using content and context features. Section III explains our proposed method to extract hybrid deep features (HDF). Experimental results and their analysis are discussed in Section IV, which follows Section V for the conclusion of our method with future works.

II. Related works

In this section, we review the state-of-the-art image feature extraction methods. Depending on the source where the features are extracted from, we generally categorize them into three groups: conventional computer vision methods [1]–[7],

[22], [25]–[31], tag-based methods [8]–[10], and deep learning based methods [8], [11]–[18].

A. Conventional computer vision based methods

Conventional vision based methods basically rely on the hand-crafted feature extraction techniques such as Generalized Search Trees (GIST) [2], GIST-color [3], Scale-invariant Feature Transform (SIFT) [1], Histogram of Gradient (HOG) [4], Spatial Pyramid Matching (SPM) [32], CENsus TRansform hISTogram (CENTRIST) [5], Oriented Texture Curves (OTC) [7], multi-channel CENTRIST (mCENTRIST) [6], RoI (regions of interest) with GIST [22], MM (Max-Margin)-Scene [25], Object bank [26], Reconfigurable Bag of Words (RBoW) [27], Bag of Parts (BoP) [28], Important Spatial Pooling Region (ISPR) [29], Laplacian Sparse coding SPM (LscSPM) [30], Improved Fisher Vector (IFV) [31] , and so on. All of these features are extracted using the fundamental information of the images such as intensity, colors, orientations, etc. Furthermore, these features are basically relied on the local details, and therefore suitable for certain images such as texture images. They are usually poor for complex images such as scenes. Also, the size of these types of features are often higher than other high-level semantic features.

B. Tag-based methods

These features are extracted based on the contextual information of images. They represent scene images by tags extracted from annotations/descriptions of similar images available on the web [8]–[10]. Zhang et al. [8] used descriptions of similar images to design bag-of-words (BoW) features directly. In this approach, there is not only the chance of having outlier tags but also high-dimensional features. To overcome this limitation, Wang et al. [9] proposed the concept of filter bank using pre-defined categories obtained from ImageNet [20] and Places [16] to filter out the outliers to some extent. Because the filter bank is solely dependent on pre-defined category names, it is more likely to miss other important tags related to images. Recently, Sitaula et al. [10] designed a novel filter bank to extract the tag-based features by exploiting the semantic similarity of tags with image category labels. It provides rich tag-based features and produces better classification accuracies compared to other tag-based features.

C. Deep learning based methods

In most cases, features extracted from deep learning models [8], [11]–[18], [33], [34] are found to have more promising classification accuracies for scene images than other methods. The popular deep learning based feature extraction methods for scene images are: CNN-MOP [11], CNN-sNBNL [15], VGG [16], ResNet152 [17] EISR [8], G-MS2F [14], SBoSP-fusion [12], BoSP-Pre_gp [13], CNN-LSTM [18], and so on.

Gong et al. [11] and Kuzborskij et al. [15] used the Caffe model [35] to extract the multi-scale deep features. Zhou et al. [16] launched a new scene related dataset and trained deep learning architectures such as VGG model [19]. The features extracted by their method produced promising classification accuracies on scene images. He et al. [17] proposed a novel deep learning architecture based on residual concepts and outperforms the previous state-of-the-art deep architectures such as the VGG model [19], GoogleNet model [36], etc. Zhang et al. [8] extracted deep features of an image using its multiple sub-images through random slicing. They concatenated deep features of each slice as a set of deep features of the image. Finally, they combined the deep features with tag-based features to produce a final set of features for the classification purpose. Tang et al. [14] employed a score-fusion approach to extract deep features. They chose the GoogleNet model [36] and extracted score features from three classification layers for the fusion. Guo et al. [12], [13] utilized the VGG16 model [19] to extract unsupervised features by introducing the concept of the bag of surrogate parts (BoSP). It not only reduced the size of features but also improved the classification accuracies. Furthermore, while comparing different pooling layers of the VGG16 model [19], they unveiled that the 5^{th} pooling is the best among others in terms of classification accuracy, owing to its better representation capability of objects in the image. Recently, Bai et al. [18] designed a new deep model by combining Convolutional Neural Networks (CNNs) with Long Short Term Memory networks (LSTMs). They cast the issue of ordered sliced images as a sequence problem and designed a network to extract scene image features.

To sum up, the limitations of the existing deep learning based methods are twofold. Firstly, the existing methods extract features at one level only, and ignore a hybrid of features in part image level and whole image level. Aggregating features extracted from both part image level and the whole image can be useful to identify interesting semantic regions in the image. Secondly, the existing methods rely on either scene-based features or object-based features only. Both types of features are equally important for scene images representation. The objects may not be the sole discriminators of the scene images since the contextual information in the image background can change their semantic meanings.

III. The Proposed Method

We propose to extract hybrid deep features (HDF) by fusing scene-based and object-based deep features at both the whole image and part image levels. Our method consists of five steps: object-based features extraction at the part image level, object-based features extraction at the whole image level, scene-based features extraction at the part image level, scene-based features extraction at the whole image level, and aggregation/fusion of the four types of features.

We employ the VGG16 models [19] and exploit the 5^{th} pooling layer, as suggested by [12], [13]. To extract scene-based and object-based deep features, we use the VGG16 models [19] pre-trained on ImageNet [20] and Places [16] datasets. The VGG16 model [19] pre-trained on ImageNet [20] provides features related to objects (foreground) in an image, whereas the VGG16 model [19] pre-trained on Places [16] provides features related to the scene (background) in the image. We resize all the images into 224×224 before

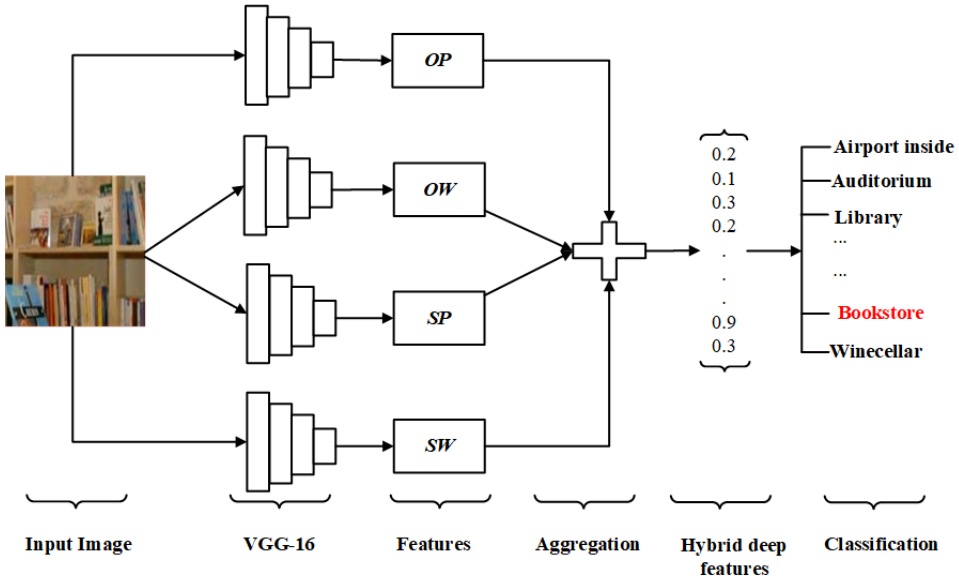

Fig. 2. Overview of our approach. The notations OP, OW, SP, and SW represent object-based features at part image level, object-based features at whole image level, scene-based features at part image level, and scene-based features at whole image level, respectively.

inputting to the VGG16 model. To extract part level features of an image, we aggregate features extracted from multiple sub-images of the image produced by five different slicing techniques (three existing techniques and two introduced in this work). We perform the Global Average Pooling (GAP) operation on the deep features extracted by the deep learning model to extract the 512-D features. The GAP operation captures both lower and higher activation values in each feature map of the deep learning model, which is suitable for scene images to grab the discriminant features. Similarly, motivated by the ability of GAP operation, we apply mean pooling to leverage both higher and lower activation values on the feature vectors extracted by GAP operation. The five steps in the proposed method is discussed successively in the next five subsections.

A. Object-based features extraction at the part image level

To extract object-based parts level (OP) features, we slice an image (I) into 20 slices (each of the five slicing techniques yields four image parts, shown in Fig. 1) $\{I_1, I_2, I_3, \cdots, I_{20}\}$. For each image slice I_i, deep features are extracted from the VGG16 model pre-trained on ImageNet using the GAP operation. To obtain OP features of I, we then aggregate deep features of the 20 slices by performing the mean pooling operation as:

$$OP(I) = Mean\{OF(I_1), OF(I_2), \cdots, OF(I_{20})\}, \quad (1)$$

where $OF(I_i) = GAP\{VGG16_{ImageNet}(I_i)\}$ indicates the GAP operation based deep features of image slice I_i from the 5^{th} pooling layer of VGG16 model pre-trained on ImageNet.

B. Object-based features extraction at the whole image level

To extract the object-based whole image level (OW) features, we again adopt the VGG16 model [19] pre-trained on ImageNet ($VGG16_{ImageNet}$). We extract deep features of the whole image I via the GAP operation from the 5^{th} pooling layer of such VGG16 model.

$$OW(I) = GAP\{VGG16_{ImageNet}(I)\} \quad (2)$$

C. Scene-based features extraction at the part image level

To extract scene-based parts level (SP) features of the image I, the deep features of the 20 slices $\{I_1, I_2, I_3, \cdots, I_{20}\}$ extracted from the the 5^{th} pooling layer of the VGG16 model pre-trained on Places ($VGG16_{Places}$) are combined through the mean pooling.

$$SP(I) = Mean\{SF(I_1), SF(I_2), \cdots, SF(I_{20})\}, \quad (3)$$

where $SF(I_i) = GAP\{VGG16_{Places}(I_i)\}$ denotes the GAP based deep features of the image slice I_i from the 5^{th} pooling layer of the VGG16 model pre-trained on Places.

D. Scene-based features extraction at the whole image level

Similarly, $VGG16_{Places}$ is used to extract scene-based whole image level (SW) features of image I as:

$$SW(I) = GAP\{VGG16_{Places}(I)\} \quad (4)$$

E. Features aggregation

After computing the above four types of deep features, we need to aggregate them to form the hybrid deep features (HDF) for the representation of the scene image I. Such features is achieved by using a pooling operator:

$$HDF(I) = Pool\{OP(I), OW(I), SP(I), SW(I)\} \quad (5)$$

Note that the size of each type of deep features is 512. The size of the final hybrid features depends on the pooling operation. It remains 512 if Min, Max or Mean pooling is

Fig. 3. Images sampled from MIT-67 [22].

Fig. 4. Images sampled from Scene-15 [23].

used, but it increases to 2048 for the concatenation operation. We empirically found that the concatenation produces better results than others. Therefore, all experiments conducted in this work are based on the concatenation operation which leads to 2048-D features. We present the comparison results of all four pooing operations in Section IV-E.

Finally, we utilize such hybrid features (HDF) obtained from Eq. (5) for the task of scene image classification. The overview of the proposed method is shown in Fig. 2.

IV. Experiments and Analysis

In this section, we will explain the used datasets, implementation and experimental results (comparisons and ablation studies) in scene classification.

A. Datasets

We employ three commonly used scene image datasets: MIT-67 [22] (Fig. 3), Scene-15 [23] (Fig. 4), and Event-8 [24] (Fig. 5).

MIT-67 [22] is the largest indoor scene dataset employed in the experiment, and has been used by previous studies [7]–[14], [17], [18], [22], [25]–[29], [31], [37]. It contains $15,620$ images belonging to 67 indoor categories. We use the same train/test split as suggested by [22]. For train/test split, 80 images per category are used for training and the remaining 20 images are used for testing.

Scene-15 [23] dataset comprises images of 15 categories. There are $4,485$ images, where each category contains 200 to 400 images. We create 10 train/test sets and present the average accuracy, as done by previous studies [3], [5], [7]–[10], [14], [17], [29], [31], [32], [37]. For each train/test set, 100 images per category are selected for the training set and the rest for the test set.

Event-8 [24] dataset includes images of 8 different sports categories. There are $1,579$ images in this dataset, where each category comprises 137 to 250 images. We also use 10 different train/test sets, as in previous studies [8]–[10], [15]–[17], [26], [29]–[31], [37], and present the average accuracy. For each train/test set, 70 images per category are involved in the train set and 60 images are involved in the test set.

B. Implementation

To implement our approach, we use the Keras python package [38] for the deep learning models pre-trained on the Places [16], [39] and ImageNet [20] datasets. The proposed hybrid features (HDF) are encoded and normalized as suggested by Guo et al. [12], [13]. We use the L_2-Regularized Logistic Regression classifier (LR) implemented using LibLinear [40] as the classifier to classify scene images. We use this classifier

Fig. 5. Images sampled from Event-8 [24].

in our experiments for two reasons: (1) it is fast owing to its simple operations; (2) it produces better classification performance on deep features as suggested by [12], [13]. In each experiment, the cost parameter (C) is automatically tuned in the range $\{1, 2, 3, \cdots, 100\}$ using a grid search technique, and the default settings of other parameters are used. All experiments are conducted on a laptop with an NVIDIA GeForce GTX 1050 GPU.

C. Comparison with the state-of-the-art methods

We evaluate the classification performance of the proposed hybrid deep features (HDF) against the features by 27 state-of-the-art image features extraction methods (12 conventional computer vision based features, 6 tag-based features and 9 deep learning based features). The classification accuracies of the proposed features and other contenders are provided in Table I. Notice that results of the contenders are taken from the corresponding published papers.

The results presented in the first column of Table I show that the proposed hybrid deep features (HDF) produce the best result with an accuracy of 82.0% on the MIT-67 dataset. CNN-LSTM [18] (80.5%) ranks the second best, followed by G-MS2F [14] (79.6%) in the third place. Results of tag-based features are at least 5.5% worse than HDF, and conventional computer vision based methods generate worse results by at least 21.2% than our HDF.

On Scene-15 (results in the second column of Table I), the proposed hybrid deep features produce the second best accuracy (93.9%) which is behind 94.5% of EISR [8]. G-MS2F [14] ranks the third place with an accuracy of 92.9%. Similar to MIT-67, conventional computer vision based features and tag-based features induce worse results than deep learning based features on Scene-15.

The results in the third column of Table I show that the proposed hybrid deep features generate the second best accuracy of 96.2% which is slightly behind 96.9% of ResNet152 [17]. VGG [16] achieved the third best result with an accuracy of 95.6%.

Classification results in three datasets show that the proposed hybrid deep features (HDF) can produce the best or second best results in all cases. It demonstrates the consistent and stable performance of our hybrid deep features across different datasets. Results of other features vary significantly across different datasets (MIT-67, Scene-15, and Event-8) . EISR [8] has the best result in Scene-15 but ranks the eighth and fifth in the MIT-67 and Event-8 datasets, respectively. Similarly, ResNet152 [17] produces the best result in Event-8, but it ranks the sixth in MIT-67 and the fourth in Scene-15, respectively. Taking a closer look at EISR [8], its features size is extremely higher than ours. They also use random cropping of the image regions which ranges from 50 to 400 and concatenate the deep features of all the regions sequentially, and then they concatenate the features with tag-based features. As a result, they have far more than $50 \times 2,048$-D features, with considering the concatenation with the tag-based features. This is probably why it generates the best accuracy on Scene-15. Similarly, ResNet152 [17] provides $2,048$-D features, the size of which is equal to ours. However, it induces the highest accuracy in only one dataset and much worse accuracies in the other two datasets, while our features are more stable and perform the best or the second best across all three datasets. We suspect that the consistent and stable performance of the proposed hybrid deep features across three datasets is mainly due to the fusion of object-based and scene-based features at both whole image and parts levels. The four types of features enable the capture of different and complementary information on one image.

D. Ablative study of individual features

We also evaluate the performances of each of the four types of features. The classification accuracies using object-based features on part images (OP), object-based features on the whole image (OW), scene-based features on part images (SP), and scene-based features on the whole image (SW) are provided in Table II.

By observing results in Table II, we notice that the SW features resulted in the best accuracy on the MIT-67 and Scene-15 datasets (79.7% and 92.8%, respectively), whereas the OW features induce the best result on the Event-8 dataset (95.7%). The images on Event-8 often contain single objects

TABLE I
CLASSIFICATION ACCURACY (%) OF THE STATE-OF-THE-ART METHODS AND OUR PROPOSED METHOD ON THE TEST SET OF THREE DATASETS. BEST ACCURACY IS IN BOLD AND THE SECOND BEST ACCURACY IS UNDERLINED. THE ASTERISK (*) SYMBOL REPRESENTS NO PUBLISHED RESULTS ON THE CORRESPONDING DATASET.

Method	MIT-67	Scene-15	Event-8
Conventional computer vision based methods			
GIST-color [3]	*	69.5	*
ROI with GIST [22]	26.1	*	*
SPM [32]	*	81.4	*
MM-Scene [25]	28.3	*	*
CENTRIST [5]	*	83.9	*
Object Bank [26]	37.6	*	76.3
RBoW [27]	37.9	*	*
BOP [28]	46.1	*	*
OTC [7]	47.3	84.4	*
ISPR [29]	50.1	85.1	74.9
LscSPM [30]	*	*	85.3
IFV [31]	60.8	89.2	90.3
Tag-based methods			
BoW [9]	52.5	70.1	83.5
CNN [37]	52.0	72.2	85.9
s-CNN(max) [9]	54.6	76.2	90.9
s-CNN(avg) [9]	55.1	76.7	91.2
s-CNNC(max) [9]	55.9	77.2	91.5
TSF [10]	76.5	81.3	94.4
Deep learning-based methods			
EISR [8]	66.2	**94.5**	92.7
CNN-MOP [11]	68.0	*	*
SBoSP-fusion [12]	77.9	*	*
BoSP-Pre_gp [13]	78.2	*	*
G-MS2F [14]	79.6	92.9	*
CNN-sNBNL [15]	*	*	95.3
VGG [16]	*	*	95.6
ResNet152 [17]	77.4	92.4	**96.9**
CNN-LSTM [18]	<u>80.5</u>	*	*
Ours *HDF*	**82.0**	<u>93.9</u>	<u>96.2</u>

TABLE II
CLASSIFICATION ACCURACY (%) OF EACH INDIVIDUAL TYPE OF FEATURES (OP, OW, SP, AND SW) ON TEST SET OF THREE DATASETS.

Dataset	OP	OW	SP	SW
MIT-67	68.2	70.7	76.5	**79.7**
Scene-15	88.8	90.3	92.1	**92.8**
Event-8	93.7	**95.7**	93.0	94.9

which can be easily captured by the object-based features at the whole image level (OW). However, images on MIT-67 and Scene-15 are more dependent on scenes, and thus scene-based features SW bring about the best accuracy. These three best accuracies are obviously lower than the accuracies led by the hybrid features of all four types of features, which further demonstrates the necessity of the aggregation of the four types of features.

TABLE III
CLASSIFICATION ACCURACY (%) OF OUR HYBRID DEEP FEATURES (HDF) ACHIEVED BY FOUR DIFFERENT AGGREGATION METHODS (MAX, MEAN, MIN, AND CONCATENATE) ON THE TEST SET OF THREE DATASETS.

Dataset	Max	Mean	Min	Concat
MIT-67	79.9	80.3	67.9	**82.0**
Scene-15	93.3	93.4	87.4	**93.9**
Event-8	96.0	95.8	90.9	**96.2**

TABLE IV
COMPUTATIONAL TIME (SECONDS) TAKEN BY THE FEATURES EXTRACTION, TRAINING AND TESTING FOR THE PROPOSED METHOD ON THREE DATASETS.

Dataset	Feat. extraction step	Training step	Testing step
MIT-67	42813.4	8.2	1.0
Scene-15	5659.9	0.7	2.1
Event-8	1348.8	0.1	0.4

E. Ablative study of aggregation methods

To study the efficacy of the four aggregation methods (Max pooling, Mean pooling, Min pooling, and Concat pooling in Eq. (5), we perform experiments on all three datasets, and the results are summarized in Table III. The results manifests that our proposed features (HDF) by the Concat aggregation yield higher accuracies than other methods on all three datasets. This is because all these features are different types of features capturing different information about images.

F. Computational time

We study the computational time (seconds) taken by our proposed method for three different steps including features extraction step, training step and testing step, and list the results in Table IV. For the Scene-15 and Event-8 datasets, we provide the average computational time of 10 runs used in the experiments. We observe that the the average features extraction time per image including training and testing images on the MIT-67 (6,700 images) is 6.3 seconds, whereas it is 1.2 seconds for both Scene-15 (4,485 images) and Event-8 (1,040 images) datasets. It unveils that the average features extraction time per image on MIT-67 is higher than other datasets because it contains images with multiple objects and regions due to which pre-trained models yield multiple activation values, and results in higher computation burden during pooling and aggregation operations in our method. Similarly, we observe that the classification time per image of testing images on MIT-67 (1,340 images), Scene-15 (2,985 images), and Event-8 (480 images) is 0.0007 seconds, 0.0007 seconds, and 0.0008 seconds, respectively. This reveals a similar classification time of a testing image on all of the datasets.

V. CONCLUSION

In this paper, we have introduced hybrid deep features to represent images by aggregating four types of features

(scene-based and object-based features at the whole image and part image levels). Since the four types of deep features capture different types of information about images, fusing them together can provide richer discriminant information of images. Experimental results in scene image classification on three widely used scene image datasets unveil that the proposed hybrid deep features are capable of producing more consistent and stable results (the best or second best) than the state-of-the-art techniques. We also notice that the proposed features are more prominent and suitable for indoor scene images because such images contain both objects and scenes as the discriminator information.

Compared to indoor scene images, the proposed features may be less powerful in representing other types of images like outdoor images. Furthermore, we only used the 5^{th} pooling layer in our method, which may not be sufficient to extract features. In the future, we would like to analyze the characteristics of different types of images and exploit other layers of the pre-trained models for the classification task.

REFERENCES

[1] O. Zeglazi, A. Amine, and M. Rziza, "Sift descriptors modeling and application in texture image classification," in *Proc. 13th Int. Conf. Comput. Graphics, Imaging and Visualization (CGiV)*, Mar. 2016, pp. 265–268.

[2] A. Oliva, "Gist of the scene," in *Neurobiology of Attention*. Elsevier, 2005, pp. 251–256.

[3] A. Oliva and A. Torralba, "Modeling the shape of the scene: a holistic representation of the spatial envelope," *Int. J. Comput. Vis.*, vol. 42, no. 3, pp. 145–175, May. 2001.

[4] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2005, pp. 886–893.

[5] J. Wu and J. M. Rehg, "CENTRIST: A visual descriptor for scene categorization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 8, pp. 1489–1501, Aug. 2011.

[6] Y. Xiao, J. Wu, and J. Yuan, "mCENTRIST: a multi-channel feature generation mechanism for scene categorization," *IEEE Trans. Image Process.*, vol. 23, no. 2, pp. 823–836, Feb. 2014.

[7] R. Margolin, L. Zelnik-Manor, and A. Tal, "Otc: A novel local descriptor for scene classification," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2014, pp. 377–391.

[8] C. Zhang, G. Zhu, Q. Huang, and Q. Tian, "Image classification by search with explicitly and implicitly semantic representations," *Information Sciences*, vol. 376, pp. 125–135, 2017.

[9] D. Wang and K. Mao, "Task-generic semantic convolutional neural network for web text-aided image classification," *Neurocomputing*, vol. 329, pp. 103–115, 2019.

[10] C. Sitaula, Y. Xiang, A. Basnet, S. Aryal, and X. Lu, "Tag-based semantic features for scene image classification," in *Proc. Int. Conf. on Neural Inf. Process. (ICONIP)*, 2019, pp. 90–102.

[11] Y. Gong, L. Wang, R. Guo, and S. Lazebnik, "Multi-scale orderless pooling of deep convolutional activation features," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2014, pp. 392–407.

[12] Y. Guo and M. S. Lew, "Bag of surrogate parts: one inherent feature of deep cnns." in *Proc. BMVC*, 2016.

[13] Y. Guo, Y. Liu, S. Lao, E. M. Bakker, L. Bai, and M. S. Lew, "Bag of surrogate parts feature for visual recognition," *IEEE Trans. Multimedia*, vol. 20, no. 6, pp. 1525–1536, Jun. 2018.

[14] P. Tang, H. Wang, and S. Kwong, "G-MS2F: GoogLeNet based multi-stage feature fusion of deep CNN for scene recognition," *Neurocomputing*, vol. 225, pp. 188 – 197, Feb. 2017.

[15] I. Kuzborskij, F. Maria Carlucci, and B. Caputo, "When naive bayes nearest neighbors meet convolutional neural networks," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 2100–2109.

[16] B. Zhou, A. Khosla, A. Lapedriza, A. Torralba, and A. Oliva, "Places: An image database for deep scene understanding," *arXiv preprint arXiv:1610.02055*, 2016.

[17] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[18] S. Bai, H. Tang, and S. An, "Coordinate cnns and lstms to categorize scene images with multi-views and multi-levels of abstraction," *Expert Systems with Applications*, vol. 120, pp. 298–309, 2019.

[19] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014.

[20] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: a large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2009.

[21] N. Ali, B. Zafar, F. Riaz, S. H. Dar, N. I. Ratyal, K. B. B., M. K. Iqbal, and M. Sajid, "A hybrid geometric spatial image representation for scene classification," *PloS one*, vol. 13, no. 9, p. e0203339, 2018.

[22] A. Quattoni and A. Torralba, "Recognizing indoor scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2009, pp. 413–420.

[23] L. Fei-Fei and P. Perona, "A bayesian hierarchical model for learning natural scene categories," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. and Pattern Recognit. (CVPR)*, vol. 2, Jun. 2005, pp. 524–531.

[24] L.-J. Li and F.-F. Li, "What, where and who? classifying events by scene and object recognition." in *ICCV*, vol. 2, no. 5, 2007, p. 6.

[25] J. Zhu, L.-j. Li, L. Fei-Fei, and E. P. Xing, "Large margin learning of upstream scene understanding models," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2010, pp. 2586–2594.

[26] L.-J. Li, H. Su, L. Fei-Fei, and E. P. Xing, "Object bank: A high-level image representation for scene classification & semantic feature sparsification," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2010, pp. 1378–1386.

[27] N. Parizi, J. G. Oberlin, and P. F. Felzenszwalb, "Reconfigurable models for scene recognition," in *Proc. Comput. Vis. Pattern Recognit.(CVPR)*, Jun. 2012, pp. 2775–2782.

[28] M. Juneja, A. Vedaldi, C. Jawahar, and A. Zisserman, "Blocks that shout: Distinctive parts for scene classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2013, pp. 923–930.

[29] D. Lin, C. Lu, R. Liao, and J. Jia, "Learning important spatial pooling regions for scene classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2014, pp. 3726–3733.

[30] I.-H. ShenghuaGao and P. Liang-TienChia, "Local features are not lonely–laplacian sparse coding for image classification," pp. 3555–3561, 2010.

[31] F. Perronnin, J. Sánchez, and T. Mensink, "Improving the fisher kernel for large-scale image classification," in *Proc. European Conference on Computer vision (ECCV)*, 2010, pp. 143–156.

[32] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2006, pp. 2169–2178.

[33] C. Sitaula, Y. Xiang, Y. Zhang, X. Lu, and S. Aryal, "Indoor image representation by high-level semantic features," *IEEE Access*, vol. 7, pp. 84 967–84 979, 2019.

[34] C. Sitaula, Y. Xiang, S. Aryal, and X. Lu, "Unsupervised deep features for privacy image classification," in *Proc. Pacific-Rim Symposium on Image and Video Technology (PSIVT)*, 2019, pp. 404–415.

[35] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *Proc. 22nd ACM Int. Conf. on Multimedia*, 2014, pp. 675–678.

[36] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," 2014.

[37] Y. Kim, "Convolutional neural networks for sentence classification," *arXiv preprint arXiv:1408.5882*, 2014.

[38] F. Chollet, "Keras," https://github.com/fchollet/keras, 2015.

[39] G. Kalliatakis, "Keras-vgg16-places365," https://github.com/GKalliatakis/Keras-VGG16-places365, 2017.

[40] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, "Liblinear: A library for large linear classification," *Journal of Machine Learning Research*, vol. 9, no. Aug, pp. 1871–1874, 2008.