# Deep Learning for Scene Classification: A Survey

Delu Zeng, Minyu Liao, Mohammad Tavakolian, Yulan Guo
Bolei Zhou, Dewen Hu, Matti Pietikäinen, and Li Liu

**Abstract**—Scene classification, aiming at classifying a scene image to one of the predefined scene categories by comprehending the entire image, is a longstanding, fundamental and challenging problem in computer vision. The rise of large-scale datasets, which constitute a dense sampling of diverse real-world scenes, and the renaissance of deep learning techniques, which learn powerful feature representations directly from big raw data, have been bringing remarkable progress in the field of scene representation and classification. To help researchers master needed advances in this field, the goal of this paper is to provide a comprehensive survey of recent achievements in scene classification using deep learning. More than 260 major publications are included in this survey covering different aspects of scene classification, including challenges, benchmark datasets, taxonomy, and quantitative performance comparisons of the reviewed methods. In retrospect of what has been achieved so far, this paper is concluded with a list of promising research opportunities.

**Index Terms**—Scene classification, Deep learning, Convolutional neural network, Scene representation, Literature survey.



## 1 Introduction

The goal of scene classification is to classify a scene image[1] to one of the predefined scene categories (such as beach, kitchen, and bakery), based on the image's ambient content, objects, and their layout. Visual scene understanding requires reasoning about the diverse and complicated environments that we encounter in our daily life. Recognizing visual categories such as objects, actions and events is no doubt the indispensable ability of a visual system. Moreover, recognizing the scene where the objects appear is of equal importance for an intelligent system to predict the context for the recognized objects by reasoning "What is happening? What will happen next?". Humans are remarkably efficient at categorizing natural scenes [3], [4]. However, it is not an easy task for machines due to the scene's semantic ambiguity, and the large intraclass variations caused by imaging conditions like variations in illumination, viewing angle and scale, imaging distance, *etc*. As a longstanding, fundamental and challenging problem in computer vision, scene classification has been an active area of research for several decades, and has a wide range of applications, such as content based image retrieval [5], [6], robot navigation [7], [8], intelligent video surveillance [9], [10], augmented reality [11], [12], and disaster detection applications [13] (*e.g.*, earthquakes, flash floods, road accidents, *etc*.).

As the core of scene classification, **scene representation** is the process of transforming a scene image into its concise descriptors (*i.e.*, features), and still attracts tremendous and increasing attention. The recent revival of interest in Artificial Neural Networks (ANNs), particularly deep learning [17], has revolutionized computer vision and been ubiquitously used in various tasks like object classification and detection [53], [54], semantic segmentation [55], [56], [57] and scene classification [18], [28]. In 2012, object classification with the large-scale ImageNet dataset [58] achieved a significant breakthrough in performance by a Deep Neural Network (DNN) named AlexNet [17], which is arguably what reignited the field of ANNs and triggered the recent revolution in computer vision. Since then, research focused on scene classification has begun to move away from handcrafted feature engineering to deep learning, which can learn powerful representations directly from data. Recent advances in deep learning have opened the possibility of scene classification towards the datasets *of large scale* and *in the wild* [18], [25], [59], and many scene representations [21], [28], [30], [60] have been proposed. As illustrated in Fig. 1, deep learning has brought significant improvements in scene classification. Given the exceptionally

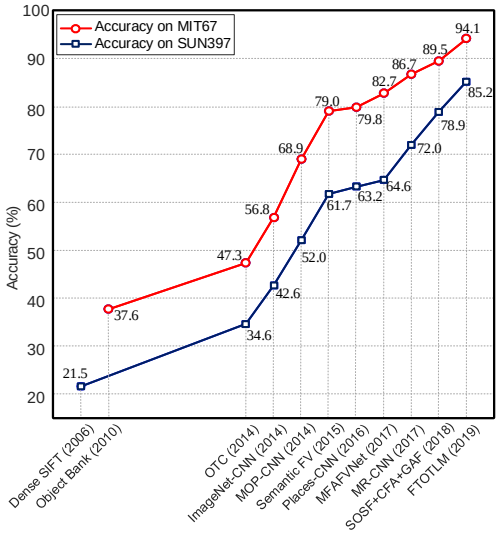



Fig. 1. A performance overview of scene classification: we can observe a significant improvement on two benchmark datasets since the reignition of deep learning. Dense-SIFT [14], Object Bank [15] and OTC [16] are handcrafted methods, while the others are deep learning based methods.

- *Delu Zeng (dlzeng@scut.edu.cn) and Minyu Liao (201820127075@scut.edu.cn) are with South China University of Technology, China. Mohammad Tavakolian (mohammad.tavakolian@oulu.fi), Matti Pietikäinen (matti.pietikainen@oulu.fi) and Li Liu (li.liu@oulu.fi) are with the Center for Machine Vision and Signal Analysis, University of Oulu, Finland. Li Liu is also with the National University of Defense Technology, China. Yulan Guo (yulan.guo@nudt.edu.cn) is with Sun Yatsen University, China, and is also with National University of Defense Technology. Bolei Zhou (bzhou@ie.cuhk.edu.hk) is with the Chinese University of Hong Kong, China. Dewen Hu (dwhu@nudt.edu.cn) is the National University of Defense Technology, China.*
- *Li Liu is the corresponding author. Delu Zeng, Minlu Liao, and Mohammad Tavakolian have equal contribution to this work and are cofirst authors.*

1. A scene is a semantically coherent view of a real-world environment that contains background and multiple objects, organized in a spatially licensed manner [1], [2].

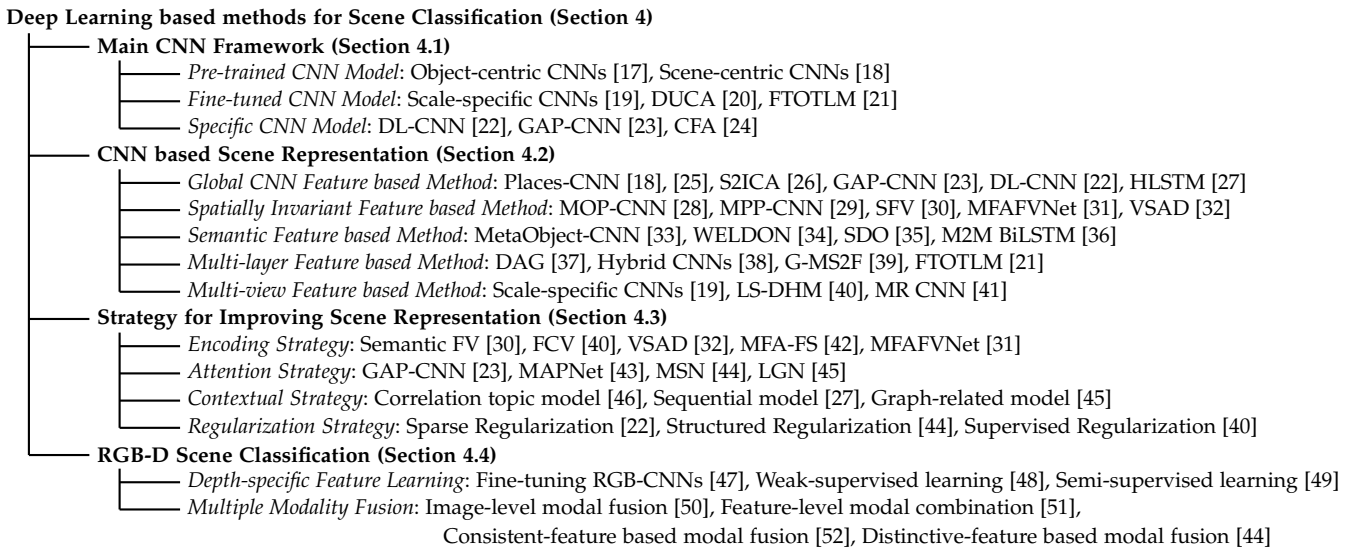



Fig. 2. A taxonomy of deep learning based methods for scene classification. With the rise of large-scale datasets, powerful feature representations are directly learned by using pre-trained CNNs, fine-tuned CNNs, or specific CNNs, having made remarkable progress. The representations mainly consist of global CNN features, spatially invariant features, semantic features, multi-layer features, and multi-view features. At the same time, the performances of many methods are improved due to effective strategies, like orderless encoding, attention learning, context modeling, and regularization. In addition, methods using RGB-D datasets, as a new issue for scene classification, mainly focus on learning depth specific features, and fusing multi-modal features.

rapid rate of progress, this article attempts to track recent advances and summarize their achievements to gain a clearer picture of the current panorama in scene classification using deep learning.

Recently, several surveys for scene classification have also been available, such as [61], [62], [63]. Cheng *et al.* [62] provided a recent comprehensive review of the recent progress for remote sensing image scene classification. Wei *et al.* [61] carried out an experimental study of 14 scene descriptors mainly in the handcrafted feature engineering way for scene classification. Xie *et al.* [63] reviewed scene recognition approaches in the past two decades, and most of discussed methods in their survey appeared in this handcrafted way. *As opposed to* these existing reviews [61], [62], [63], this work herein summarizes the striking success and dominance in indoor/outdoor scene classification using deep learning and its related methods, but not including other scene classification tasks, *e.g.*, remote sensing scene classification [62], [64], [65], acoustic scene classification [66], [67], place classification [68], [69], *etc*. The major contributions of this work can be summarized as follows:

- As far as we know, this paper is the first to specifically focus on deep learning methods for indoor/outdoor scene classification, including RGB scene classification, as well as RGB-D scene classification.
- We present a taxonomy (see Fig. 2), covering the most recent and advanced progresses of deep learning for scene representation.
- Comprehensive comparisons of existing methods on several public datasets are provided, meanwhile we also present brief summaries and insightful discussions.

The remainder of this paper is organized as follows: Challenges and the related progress in scene classification made during the last two decades are summarized in Section 2. The benchmark datasets are summarized in Section 3. In section 4, we present a taxonomy of the existing deep learning based methods. Then in section 5, we provide an overall discussion of their corresponding performance (Tables 2, 3, 5). Followed by Section 6 we conclude important future research outlook.

## 2 Background

### 2.1 The Problem and Challenge

Scene classification can be further dissected through analyzing its strong ties with related vision tasks such as object classification and texture classification. Scenes are usually composed of multiple semantic parts, some of which correspond to objects. And the texture information across scene categories is used to identify scenes [70]. As typical pattern recognition problems, these tasks consist of feature representation and classification. The strong ties between these tasks have led to the fact that the division between scene representation methods and object/texture representation methods [71], [72] has been narrowing. However, in contrast to object classification (images are object-centric) or texture classification (images include only textures), the observed scene images are more complicated, and further analysis are needed by exploring the overall semantic content of scene, *e.g.*, what the semantic parts (*e.g.*, objects, textures, background) are, in what way they are organized together, and what their semantic connections with each other are.

Human scene processing is characterized by its remarkable efficiency [3], [4]. Despite over several decades of development in scene classification, most of methods still have not been capable of performing at a level sufficient for various real-world scenes. The inherent difficulty of scene classification is due to the nature of complexity and high variance of real-world scenes. Overall, significant challenges in scene classification stem from large intraclass variations, semantic ambiguity, and computational efficiency.

**Large intraclass variation.** Intraclass variation mainly originates from intrinsic factors of the scene itself and imaging conditions. In terms of intrinsic factors, each scene can have many different example images, possibly varying with large variations among various objects, background, or human activities. Imaging conditions like changes in illumination, viewpoint, scale and heavy occlusion, clutter, shading, blur, motion, *etc*. contribute to large intraclass variations. Further challenges may be added by digitization artifacts, noise


Fig. 3. Scene exemplars of illustrating large intraclass variation and semantic ambiguity. Top: The shopping malls are quite different with each other caused by lighting conditions and overall content, which leads to large intraclass variation. Below: General layout and uniformly arranged objects are similar on archive, bookstore, and library, which leads to semantic ambiguity.

corruption, poor resolution, and filtering distortion. For instance, three shopping malls (top row of Fig. 3) are shown with different lighting conditions, viewing angle, and objects.

**Semantic ambiguity.** Since images of different classes may share similar objects, textures, background, *etc.*, they look very similar in visual appearances, which causes ambiguity among them [73], [74]. The bottom row of Fig. 3 depicts strong visual correlation between three different indoor scenes, *i.e.*, archive, bookstore, and library. With the emerging of new scene categories, the problem of semantic ambiguity would be more serious. In addition, scene category annotation is subjective, relying on the experience of the annotators, therefore a scene image may belong to multiple semantic categories [73], [75].

**Computational efficiency.** The prevalence of social media networks and mobile/wearable devices has led to increasing demands for various computer vision tasks including scene recognition. However, mobile/wearable devices have constrained computing related resources, making efficient scene recognition a pressing requirement.

### 2.2 A Road Map of Scene Classification in 20 years

Scene representation or scene feature extraction, the process of converting a scene image into feature vectors, plays the critical role in scene classification, and thus is the focus of research in this field. In the past two decades, remarkable progress has been witnessed in scene representation, which mainly consists of two important generations: handcrafted feature engineering, and deep learning (feature learning). The milestones of scene classification in the past two decades are presented in Fig. 4, in which two main stages (SIFT vs. DNN) are highlighted.

**Handcrafted feature engineering era.** From 1995 to 2012, the field was dominated by the Bag of Visual Word (BoVW) model [80], [91], [92], [93], [94] borrowed from document classification which represents a document as a vector of word occurrence counts over a global word vocabulary. In the image domain, BoVW firstly probes an image with local feature descriptors such as Scale Invariant Feature Transform (SIFT) [76], [95], and then represents an image statistically as an orderless histogram over a pre-trained visual vocabulary, in a similar form to a document. Some important variants of BoVW such as Bag of Semantics [15], [84], [96], [97] and Improved Fisher Vector (IFV) [81], have also been proposed.

Local invariant feature descriptors play an important role in BoVW because they are discriminative, yet less sensitive to image variations such as illumination, scale, rotation, viewpoint *etc*, and thus have been widely studied. Representative local descriptors for scene classification have started from SIFT [76], [95] and Global Information Systems Technology (GIST) [77], [98]. Other local descriptors, such as Local Binary Patterns (LBP) [99], Deformable Part Model (DPM) [100], [101], [102], CENsus TRansform hISTogram (CENTRIST) [79], also contribute to the development of scene classification. To improve the performance, research focus shifts to feature encoding and aggregation, mainly including Bag-of-Visual-Words (BoVW) [92], Latent Dirichlet Allocation (LDA) [103], Histogram of Gradients (HoG) [78], Spatial Pyramid Matching (SPM) [14], [104], Vector of Locally Aggregated Descriptors (VLAD) [82], Fisher kernel coding [81], [105] and Orientational Pyramid Matching (OPM) [106]. The quality of the learned codebook has a great impact on the coding procedure. The generic codebooks mainly include Fisher kernels [81], [105], sparse codebook [107], [108], Locality-constrained Linear Codes (LLC) [109], Histogram Intersection Kernels (HIK) [110], contextual visual words [111], Efficient Match Kernels (EMK) [112] and Supervised Kernel Descriptors (SKDES) [113]. Particularly, semantic codebooks generate from salient regions, like Object Bank [15], [83], [114], object-to-class [115], Latent Pyramidal Regions (LPR) [116], Bags of Parts (BoP) [84] and Pairwise Constraints based Multiview Subspace Learning (PC-MSL) [117], capturing more discriminative features for scene classification.

**Deep learning era.** In 2012, Krizhevsky *et al.* [17] introduced a DNN, commonly referred to as "AlexNet", for the object classification task, and achieved breakthrough performance surpassing the best result of hand-engineered features by a large margin, and thus triggered the recent revolution in AI. Since then, deep learning has started to dominate various tasks (like computer vision [72], [118], [119], [120], speech recognition [121], autonomous driving [122], cancer detection [123], [124], machine translation [125], playing complex games [126], [127], [128], [129], earthquake forecasting [130], medicine discovery [131], [132]), and scene classification is no exception, leading to a new generation of scene representation methods with remarkable performance improvements. Such substantial progress can be mainly attributed to advances in deep models including VGGNet [85], GoogLeNet [86], ResNet [87], *etc.*, the availability of large-scale image datasets like ImageNet [58] and Places [18], [25] and more powerful computational resources.

Deep learning networks have gradually replaced the local feature descriptors of the first generation methods and are certainly the engine for scene classification. Although the major driving force of progress in scene classification has been the incorporation of deep learning networks, the general pipelines like BoVW, feature encoding and aggregation methods like Fisher Vector, VLAD of the first generation methods have also been adapted in current deep learning based scene methods, *e.g.*, MOP-CNN [28], SCFVC [133], MPP-CNN [29], DSP [134], Semantic FV [30], LatMoG [60], MFA-FS [42] and DUCA [20]. To take fully advantage of back-propagation, scene representations are extracted from end-to-end trainable CNNs, like DAG-CNN [37], MFAFVNet [31], VSAD [32], G-MS2F [39], and DL-CNN [22]. To focus on main content of the scene, object detection is used to capture salient regions, such as MetaObject-CNN [33], WELDON [34], SDO [35], and BiLSTM [36]. Since features from multiple CNN layers or multiple views are complementary, many literatures [19], [21],

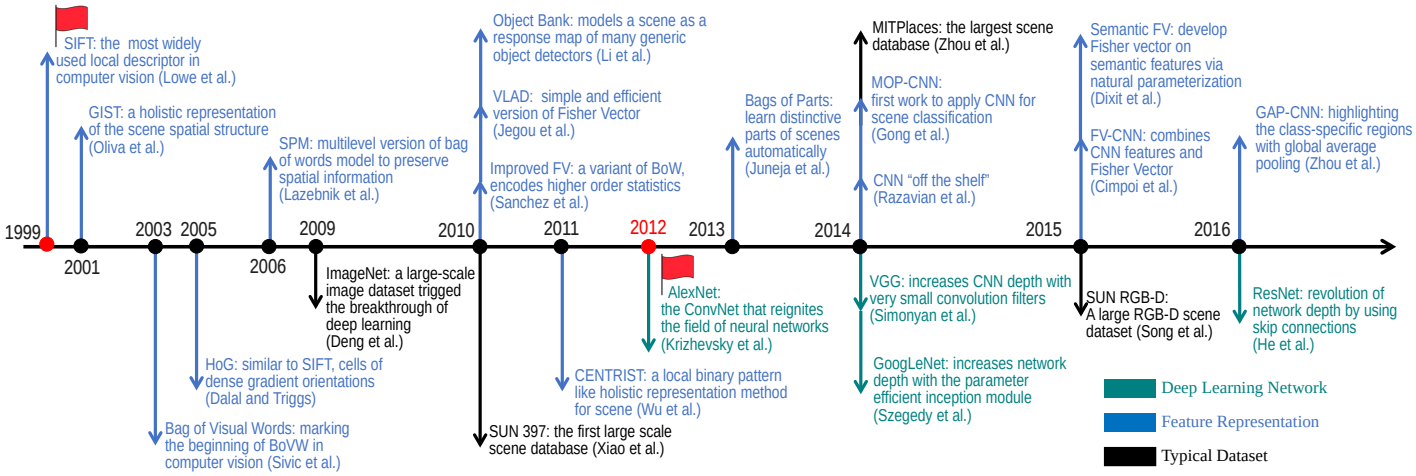



Fig. 4. Milestones of scene classification. Handcrafted features gained tremendous popularity, starting from SIFT [76] and GIST [77]. Then, HoG [78] and CENTRIST [79] were proposed by Dalal *et al.* and Wu *et al.*, respectively, further promoting the development of scene classification. In 2003, Sivic *et al.* [80] proposed BoVW model, marking the beginning of codebook learning. Along this way, more effective BoVW based methods, SPM [14], IFV [81] and VLAD [82], also emerged to deal with larger-scale tasks. In 2010, Object Bank [15], [83] represents the scene as object attributes, marking the beginning of more semantic representations. Then, Juneja *et al.* [84] proposed Bags of Part to learn distinctive parts of scenes automatically. In 2012, AlexNet [17] reignites the field of artificial neural networks. Since then, CNN-based methods, VGGNet [85], GoogLeNet [86] and ResNet [87], have begun to take over handcrafted methods. Additionally, Razavian *et al.* [88] highlights the effectiveness and generality of CNN representations for different tasks. Along this way, in 2014, Gong *et al.* [28] proposed MOP-CNN, the first deep learning methods for scene classification. Later, FV-CNN [89], Semantic FV [30] and GAP-CNN [23] are proposed one after another to learn more effective representations. For datasets, ImageNet [58] triggers the breakthrough of deep learning. Then, Xiao *et al.* [59] proposed SUN database to evaluate numerous algorithms for scene classification. Later, Places [18], [25], the largest scene database currently, emerged to satisfy the need of deep learning training. Additionally, SUN RGBD [90] has been introduced, marking the beginning of deep learning for RGB-D scene classification. See Section 2.2 for details.

[24], [40], [41] also explored their complementarity to improve performance. In addition, there exists many strategies (like attention mechanism, contextual modeling, multi-task learning with regularization terms) to enhance representation ability, such as CFA [24], BiLSTM [36], MAPNet [43], MSN [44], and LGN [45]. For datasets, because depth images from RGB-D cameras are not vulnerable to illumination changes, since 2015, researchers have started to explore RGB-D scene recognition. Some works [48], [49], [135] focus on depth-specific feature learning, while other alternatives, like DMMF [52], ACM [136], and MSN [44] focus on multi-modal feature fusion.

## 3 Datasets

This section reviews publicly available datasets for scene classification. The scene datasets (image examples are shown in Fig. 5) are broadly divided into two main categories based on the image type: RGB and RGB-D datasets. The datasets can further be divided into two categories in terms of their size. Small-size datasets (*e.g.*, Scene15 [14], MIT67 [137], SUN397 [59], NYUD2 [138], SUN RGBD [90]) are usually used for evaluation, while large-scale datasets, *e.g.*, ImageNet [58] and Places [18], [25], are essential for pre-training and developing deep learning models. Table 1 summarizes the characteristics of these datasets for scene classification.

**Scene15** dataset [14] is a small scene dataset containing 4,448 grayscale images of 15 scene categories, *i.e.*, 5 indoor scene classes (office, store, kitchen, bedroom, living room) along with 10 outdoor scene classes (suburb, forest, mountain, tall building, street, highway, coast, inside city, open country, industrial). Each class contains 210−410 scene images, and the image size is around 300×250. The dataset is divided into two splits; there are at least 100 images per class in the training set, and the rest are for testing.

**MIT Indoor 67 (MIT67)** dataset [137] covers a wide range of indoor scenes, *e.g.*, store, public space, and leisure. MIT67 comprises 15,620 scene images from 67 indoor categories, where each category has about 100 images. Moreover, all images have a minimum resolution of 200×200 pixels on the smallest axis. Because of the shared similarities among objects in this dataset, the classification of images is challenging. There are 80 and 20 images per class in the training and testing set, respectively.

**Scene UNderstanding 397 (SUN397)** dataset [59] consists of 397 scene categories, in which each category has more than 100 images. The dataset contains 108,754 images with an image size of about 500×300 pixels. SUN397 spans over 175 indoor, 220 outdoor scene classes, and two classes with mixed indoor and outdoor images, *e.g.*, a promenade deck with a ticket booth. There are several train/test split settings with 50 images per category in the testing.

**ImageNet** dataset [58], derived from the Stanford Computer Vision Lab, is one of the most famous large-scale image databases particularly used for visual tasks. It is organized in terms of the WordNet [140] hierarchy, each node of which is depicted by hundreds and thousands of images. Up to now, there are more than 14 million images and about 20 thousand notes in the ImageNet. Usually, a subset of ImageNet dataset (about 1000 categories with a total of 1.2 million images [17]) is used to pre-train the CNN for scene classification.

**Places** dataset [18], [25] is a large-scale scene dataset, which provides an exhaustive list of the classes of environments encountered in the real world. The Places dataset has inherited the same list of scene categories from SUN397 [59]. As a result, it contains 7,076,580 images from 476 scene categories. Four benchmark subsets of Places are shown as follows:

- **Places205** [18] has 2.5 million images from scene categories. The image number per class varies from 5,000 to 15,000. The training set has 2,448,873 images, with 100 images per category for the validation set and 200 images per category for the testing set.
- **Places88** [18] contains the 88 common scene categories among the ImageNet [58], the SUN397 [59], and the

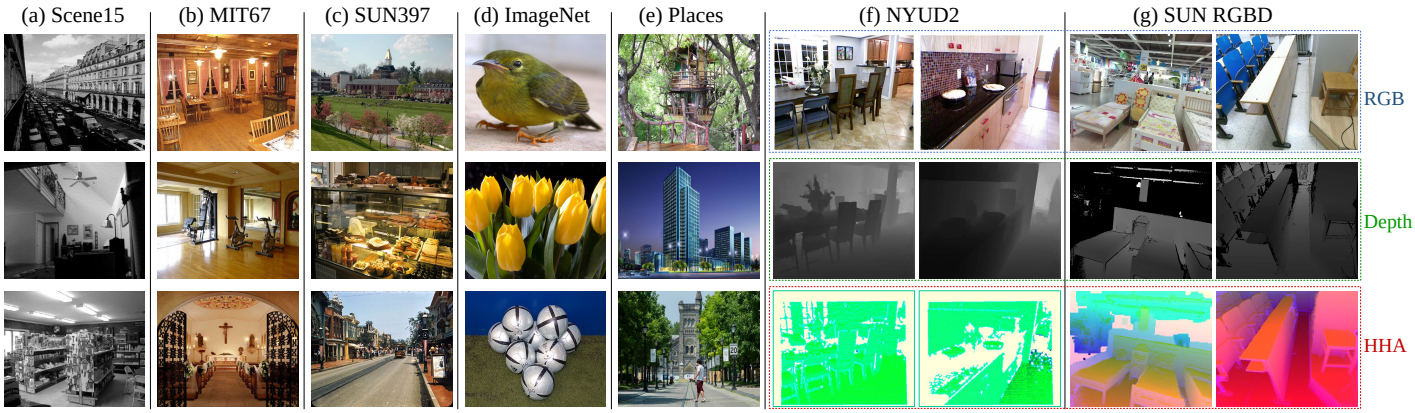



Fig. 5. Some example images for scene classification from benchmark datasets (see Table 1 for a summary of these datasets). RGB-D images consist of RGB and a depth map. Gupta *et al.* [139] proposed to convert depth map into three-channel feature map, *i.e.*, Horizontal disparity, Height above the ground, and Angle of the surface norm (HHA). Such HHA encoding is useful for the visualization of depth data.

TABLE 1
Popular datasets for scene classification. "#" denotes *the number of*.

| Type | Dataset | Start year | # Images | # Class | # Image/class | Resolution | Class label |
|---|---|---|---|---|---|---|---|
| RGB | Scene15 [14] | 2006 | 4,488 | 15 | 210–410 | $\approx 300\times250$ | Indoor and outdoor scenes |
| | MIT67 [137] | 2009 | 15,620 | 67 | $>100$ | $\geq 200\times200$ | Indoor scenes |
| | SUN397 [59] | 2010 | 108,754 | 397 | $>100$ | $\approx 500\times300$ | Indoor and outdoor scenes |
| | ImageNet [17] | 2009 | 14 millions+ | 21,841 | – | $\approx 500\times400$ | Objects |
| | Places205 [18] | 2014 | 7,076,580 | 205 | 5,000–15,000 | $\geq 200\times200$ | Indoor and outdoor scenes |
| | Places88 [18] | 2014 | – | 88 | 5,000–15,000 | $\geq 200\times200$ | Indoor and outdoor scenes |
| | Places365-Standard [25] | 2016 | 1,803,460 | 365 | 3,068–5,000 | $\geq 200\times200$ | Indoor and outdoor scenes |
| | Places365-Challenge [25] | 2016 | 1,803,460 | 365 | 5,000-30,000 | $\geq 200\times200$ | Indoor and outdoor scenes |
| RGB-D | NYUD2 [138] | 2012 | 1,449 | 10 | – | $\approx 640\times480$ | Indoor scenes |
| | SUN RGBD [90] | 2015 | 10,355 | 19 | $>80$ | RGB: $\geq640\times480$, depth: $\geq512\times424$ | Indoor scenes |

Places205 datasets. Places88 includes only the images obtained in the second round of annotation from the Places.

- **Places365-Standard** [25] has 1,803,460 training images with the image number per class varying from 3,068 to 5,000. The validation set has 50 images per class and the test set has 900 images per class.
- **Places365-Challenge** contains the same categories as the Places365-Standard, but its training set is significantly larger with a total of 8 million images. The validation and testing sets are the same as the Place365-Standard. This subset was released for the Places Challenge 2016 held in conjunction with the European Conference on Computer Vision (ECCV), as part of the ILSVRC 2016 Challenge.

**NYU-Depth V2 (NYUD2)** dataset [138] is comprised of video sequences from a variety of indoor scenes as recorded by both the RGB and depth cameras using the Microsoft Kinect. The dataset consists of 1,449 densely labeled pairs of aligned RGB and depth images from 27 indoor scene categories. It features 464 scenes taken from 3 cities and 407,024 unlabeled frames. With the publicly available split, NYUD2 for scene classification offers 795 images for training while 654 images for testing.

**SUN RGBD** dataset [90] consists of 10,335 RGB-D images with dense annotations in both 2D and 3D, for both objects and rooms. The dataset is collected by four different sensors at a similar scale as PASCAL VOC [141]. The whole dataset is densely annotated and includes 146,617 2D polygons and 58,657 3D bounding boxes with accurate object orientations, as well as a 3D room layout and category for scenes.

## 4 Deep Learning based Methods

In this section, we present a comprehensive review of deep learning methods for scene classification. The most common deep learning architecture is Convolutional Neural Network (CNN) [142]. A typical CNN is a stack of layers, mainly consisting of convolutional (CONV) layer, pooling layer, and fully-connected (FC) layer. With CNN as feature extractor, Fig. 6 shows the generic pipeline of most CNN based methods for scene classification. Almost without exception, given an input scene image, the first stage is to use CNN extractors to obtain local features. Then, the second process is to aggregate these features into an image-level representation via encoding, concatenating, or pooling. Finally, with the representation as input, the classification stage is to get a predicted category.

The taxonomy, shown in Fig. 2, covers different aspects of deep learning for scene classification. In the following investigation, we firstly study the main CNN frameworks for scene classification. Then, we review existing CNN based scene representations. Furthermore, we explore various techniques for improving the obtained representations. Finally, as a supplement, we investigate scene classification using RGB-D data.

### 4.1 Main CNN Framework

Convolutional Neural Networks (CNNs) are common deep learning models to extract high quality representation. At the beginning, limited by computing resources and labeled scene datasets, scene features are usually extracted from the pre-trained CNN model. Then, fine-tuned CNN models are used to keep last layers more data-specific. Alternatively, specific CNN models have emerged to adapt to scene attributes.

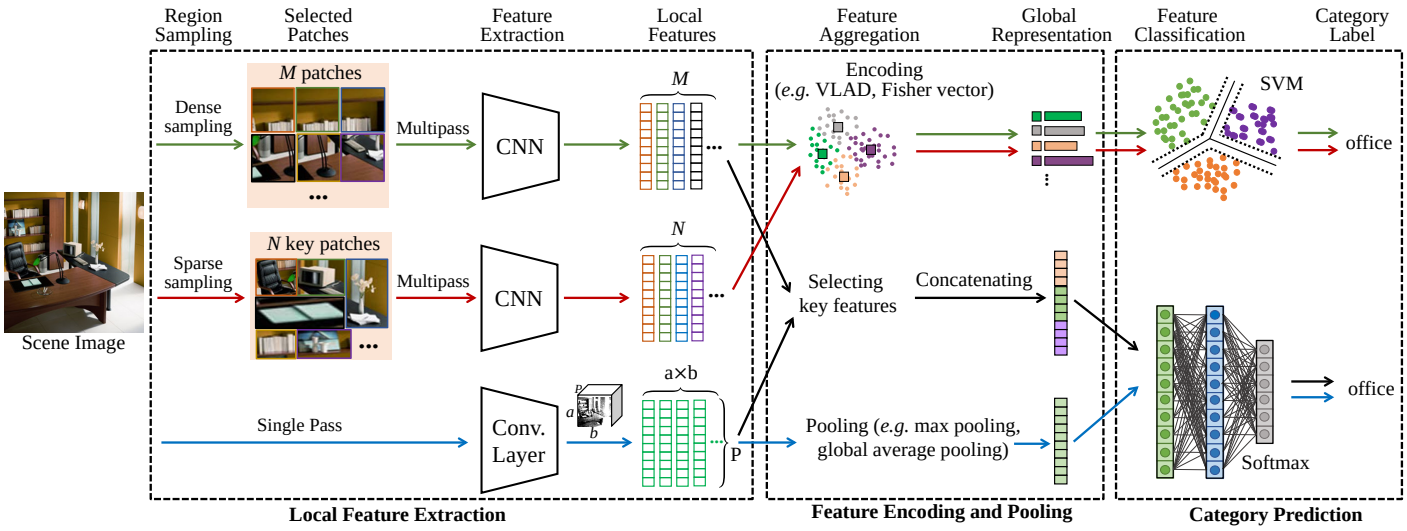



Fig. 6. Generic pipeline of deep learning for scene classification. An entire pipeline consists of a module in each of the three stages (local feature extraction, feature encoding and pooling, and category prediction). The common pipelines are shown with arrows in different colors, including global CNN feature based pipeline (blue arrows), spatially invariant feature based pipeline (green arrows), and semantic feature based pipeline (red arrows). Although the pipeline of some methods (like [31], [32]) are unified and trained in an end-to-end manner, they are virtually composed of these three stages.

#### 4.1.1 Pre-trained CNN Model

The network architecture plays a pivotal role in the performance of deep models. In the beginning, AlexNet [17] served as the mainstream CNN model for feature representation and classification purposes. Later, Simonyan *et al.* [85] developed VGGNet and showed that, for a given receptive field, using multiple stacked small kernels is better than using a large convolution kernel, because applying non-linearity on multiple feature maps yields more discriminative representations. On the other hand, the reduction of kernels receptive filed size decreases the number of parameters for bigger networks. Therefore, VGGNet has 3×3 convolution kernels instead of large convolution kernels (*i.e.*, 11×11, 7×7, and 5×5) in AlexNet. Motivated by the idea that only a handful of neurons have an effective role in feature representation, Szegedy *et al.* [86] proposed an Inception module to make a sparse approximation of CNNs. Deeper the model, the more descriptive representations. This is the advantage of hierarchical feature extraction using CNN. However, constantly increasing CNNs depth could result in vanishing the gradient through back-propagation of error from the last FC layer to the input. To address this issue, He *et al.* [87] included skip connection to the hierarchical structure of CNN and proposed Residual Networks (ResNets). ResNets are easier to optimize and can gain accuracy from considerably increased depth.

In addition to the network architecture, the performance of CNN interwinds with a sufficiently large amount of training data. However, the training data are scarce in certain applications, which results in the under-fitting of the model during the training process. To overcome this issue, pre-trained models can be employed to effectively extract feature representations of small datasets [118]. Training CNN on large-scale datasets, such as the ImageNet [58] and the Places [18], [25], makes them learn enriched visual representations. Such models can further be used as pre-trained models for other tasks. However, the effectiveness of the employment of pre-trained models largely depends on the similarity between the source and target domains. Yosinski *et al.* [143] documented that the transferability of pre-trained CNN models decreases as the similarity of the target task and original source task decreases. Nevertheless, pre-trained models still have better performance than random initialization of the models [143].

Pre-trained CNNs, as fixed feature extractors, are divided into two categories: object-centric and scene-centric CNNs. Object-centric CNNs refer to the model pre-trained on object datasets, *e.g.*, the ImageNet [58], and deployed for scene classification. Since object images do not contain the diversity provided by the scene [18], object-centric CNNs have limited performance for scene classification. Hence, scene-centric CNNs, pre-trained on scene images, like Places [18], [25], are more effective to extract scene-related features.

**Object-centric CNNs.** Cimpoi *et al.* [89] asserted that the feature representations obtained from object-centric CNNs are object descriptors since they have likely more object descriptive properties. The scene image is represented as a bag of semantics [30], and object-centric CNNs are sensitive to the overall shape of objects, so many methods [28], [30], [31], [89], [133] used object-centric CNNs to extract local features from different regions of the scene image. Another important factor in the effective deployment of object-centric CNNs is the relational size of images in the source and target datasets. Although CNNs are generally robust against size and scale, the performance of object-centric CNNs is influenced by scaling because such models are originally pre-trained on datasets to detect and/or recognize objects. Therefore, the shift to describing scenes, which have multiple objects with different scales, would drastically affect their performance [19]. For instance, if the image size of the target dataset is smaller than the source dataset to a certain degree, the accuracy of the model would be compromised.

**Scene-centric CNNs.** Zhou *et al.* [18], [25] demonstrated the classification performance of scene-centric CNNs is better than object-centric CNNs since the former use the prior knowledge of the scene. Herranz *et al.* [19] found that Places-CNNs achieve better performance at larger scales; therefore, scene-centric CNNs generally extract the representations in the whole range of scales [23]. Guo *et al.* [40] noticed that the CONV layers of scene-centric CNNs capture more detail information of a scene, such as local semantic regions and fine-scale objects,

which is crucial to discriminate the ambiguous scenes, while the feature representations obtained from the FC layers do not convey such perceptive quality. Zhou *et al.* [144] showed that scene-centric CNNs may also perform as object detectors without explicitly being trained on object datasets.

#### 4.1.2 Fine-tuned CNN Model

Pre-trained CNNs, described in Section 4.1.1, perform as deep feature extractor with prior knowledge of the image dataset, on which they are trained [6], [71]. However, using only the pre-training strategy would prevent exploiting the full capability of the deep models in describing the target scenes adaptively. Hence, fine-tuning the pre-trained CNNs using the target scene dataset improves their performance by reducing the possible domain shift between two datasets [71]. Notably, a suitable weight initialization becomes very important, because it is quite difficult to train a deep network model with many adjustable parameters and non-convex loss functions [145]. Therefore, fine-tuning the pre-trained CNN contributes to the effective training process [29], [30], [34], [146].

Fine-tuning a models parameters is a simple and effective technique for transferring knowledge from a pre-trained model. For CNNs, a common fine-tuning technique is the freeze strategy. In this method, the last FC layer of a pre-trained model is replaced with a new FC layer with the same number of neurons as the classes in the target dataset (*i.e.,* MIT67, SUN397), while the previous CONV layers parameters are frozen, *i.e.,* they are not updated during the fine-tuning process. Then, this modified CNN is fine-tuned by training on the target dataset. Herein, the back-propagation is stopped after the last FC layers, which allows these layers to learn discriminative knowledges from the learned CONV layers. Through updating few parameters, training a complex model using small datasets would be affordable. Optionally, it is also possible to gradually unfreeze some layers to further enhance the learning quality as the earlier layers would adapt new representations from the target dataset. Alternatively, different learning rates could be assigned to different layers of CNN, in which the early layers of the model have very low learning rate and the last layers have higher learning rates. In this way, the early CONV layers that have more abstract representations are less affected, while the specialized FC layers are fine-tuned with higher speed.

The size of the target dataset is an important factor for the fine-tuning process. Deep models may not benefit from fine-tuning on a small target dataset [71]. In this case, fine-tuning has negative effects since the structure of specialized FC layers has changed while inadequate training data are provided for fine-tuning. Data augmentation is one alternative to deal with the small size of the target dataset [20], [21], [49], [147]. Khan *et al.* [20] augmented the scene image dataset with flipped, cropped, and rotated versions to increase the size of the dataset and further improve the robustness of the learned representations. Liu *et al.* [21] used a sliding cropping window to generate new patches from an image and selected those patches with sufficiently enough representative information of the original image.

There exists a problem via data augmentation to fine-tune CNNs for scene classification. Herranz *et al.* [19] asserted that fine-tuning a CNN model have certain "equalizing" effect between the input patch scale and final accuracy, *i.e.,* to some extent, with too small patches as CNN inputs, the final classification accuracy is worse. This is because the small patch inputs only contain part of image information, while the final labels indicate scene categories. Moreover, the number of cropped patches is huge, so just a tiny part of these patches is used to fine-tune CNN models, rendering limited overall improvement [19]. On the other hand, Herranz *et al.* [19] also explored the effect of fine-tuning CNNs on different scales, *i.e.,* with different scale patches as inputs. From the practical results, there is a moderate accuracy gain in the range of scale patches where the original CNNs perform poorly, *e.g.,* in the cases of global scales for ImageNet-CNN and local scales for Places-CNN. However, there is marginal or no gain in ranges where CNN have already strong performance. For example, since Places-CNN has the best performance in the whole range of scale patches, in this case, fine-tuning on target dataset leads to negligible performance improvement.

#### 4.1.3 Specific CNN Model

In addition to the generic CNN models, *i.e.,* pre-trained CNN models and the fine-tuned CNN models, another group of deep models are specifically designed for scene classification. These models are specifically developed to extract effective scene representations from the input by introducing new network architectures. As is shown in Fig. 7, we only show four typical specific models [22], [23], [24], [26].

To capture discriminative information from regions of interest, Zhou *et al.* [23] replaced the FC layers in a CNN model with a Global Average Pooling (GAP) layer [148] followed by a Softmax layer, *i.e.,* GAP-CNN. As shown in Fig. 7 (a), by a simple combination of the original GAP layer and the $1 \times 1$ convolution operation to form a class activation map (CAM), GAP-CNN can focus on class-specific regions and perform scene classification well. Although the GAP layer has a lower number of parameters than the FC layer [23], [49], the GAP-CNN can obtain comparable classification accuracy.

Hypothesizing that a certain amount of sparsity improves the discriminability of the feature representations [149], [150], [151], Liu *et al.* [22] proposed a sparsity model named Dictionary Learning CNN (DL-CNN), seen in Fig. 7 (b). They replaced FC layers with new dictionary learning layers, which are composed of a finite number of recurrent units that correspond to iteration processes in the Approximate Message Passing [152]. In particular, these dictionary learning layers parameters are updated through back-propagation in an end-to-end manner.

Since the CONV layers perform local operations on small patches of the image, they are not able to explicitly describe the contextual relation between different regions of the scene image. To address this limitation, Sun *et al.* [24] proposed Contextual features in Appearance (CFA) based on LSTM [153]. As shown in Fig. 7 (c), CONV feature maps are regarded as the input of LSTM layers, which is transformed into four directed sequences in an acyclic way. Finally, LSTM layers are used to describe spatial contextual dependencies, and the output of four LSTM modules are concatenated to describe contextual relations in appearance.

Sequential operations of CONV and FC layers in standard CNNs retain the global spatial structure of the image, which shows global features are sensitive to geometrical variations [28], [43], *e.g.,* object translations and rotation directly affect the obtained deep features, which drastically limits the application of these features for scene classification. To achieve geometric invariance, as shown in Fig. 7 (d), Hayat *et al.* [26] designed a spatial unstructured layer to introduce robustness against spatial layout deformations.

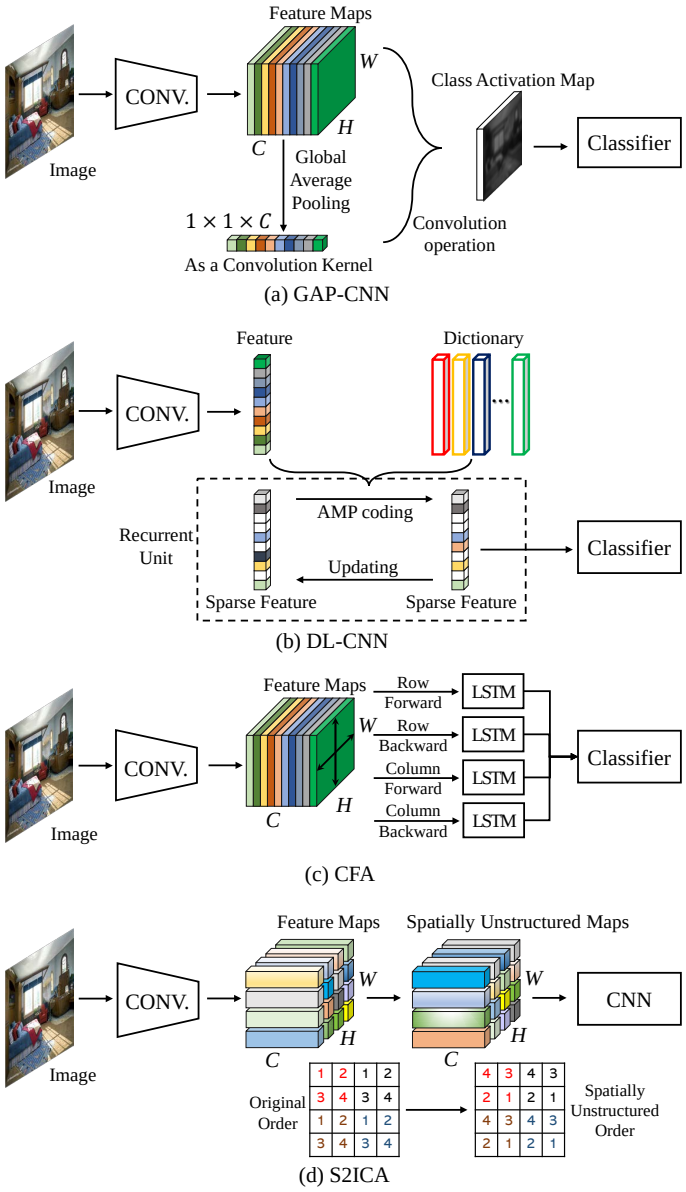



Fig. 7. Illustrations of some typical specific CNNs for scene classification. (a) In GAP-CNN [23], to reduce parameters of the standard CNN, FC layers are removed, and GAP layer is introduced to form Class Activation Mapping (CAM). (b) In DL-CNN [22], to reduce parameters of CNN model and obtain enhanced sparse features, Dictionary Learning (DL) layers are proposed to replace FC layers. (c) In CFA [24], Sun *et al.* bypassed four directional LSTM layers on the CONV maps to capture contextual information. (d) In S2ICA [26], to achieve spatial invariance, the spatial unstructured layer is placed before a CONV layer, shuffling the original position of the original feature maps by swapping adjacent diagonal image blocks.

Back-propagation algorithm is the essence of CNN training. It is the practice of fine-tuning the weights of a neural net based on the error rate (*i.e.,* loss) obtained in the previous epoch (*i.e.,* iteration). Proper tuning of the weights ensures lower error rates, making the model reliable by increasing its generalization. Therefore, many approaches [34], [37], [43], [69] developed new layers with parameters that can be updated via back-propagation. The end-to-end system is trained via back-propagation in a holistic manner, which has been proved as a powerful training manner in various domains, and scene classification is no exception. Many methods [31], [34], [37], [43], [69] are training in an end-to-end manner. According to our investigations, theoretically, these models can learn more discriminative information through end-to-end optimization; however, the optimization results may fall into bad local optima [43], [154], so methods training in a multi-stage manner may achieve better results in some case.

## 4.2 CNN based Scene Representation

Scene representation, the core of scene classification, has been the focus of this research. Hence, many methods have been put forward for effective scene representations, broadly divided into five categories: global CNN features, spatially invariant features, semantic features, multi-layer features, and combined features, *i.e.,* multi-view features.

### *4.2.1 Global CNN feature based Method*

Global CNN feature based methods directly predict the probabilities of scene categories from the whole scene image. Frequently, global CNN features are extracted from input images via using generic CNN models, pre-trained on large-scale datasets (*e.g.,* ImageNet [58] and Places [18], [25]), or then fine-tuned on target datasets (*e.g.,* SUN397 [59] and MIT67 [137]). Owing to the available large datasets and powerful computing resources (e.g., GPUs and parallel computing clusters) [155], deep networks have been developed into deeper and more complicated, and thus global representations from these networks are able to achieve a more advanced performance on many applications including scene classification.

Except for generic CNNs, some scene-specific CNNs are designed to extract global features. For instance, as shown in Fig. 8 (a), Zuo *et al.* [27] proposed Hierarchical LSTM (HLSTM) to describe the contextual relation. They treated CONV maps as an undirected graph, which is transformed into four directed acyclic graphs, and LSTM modules are used to capture spatial contextual dependencies in an acyclic way. They also explored the potential spatial dependencies among different scale CONV maps, so HLSTM features not only involve the relations within the same feature maps but also contain the contextual dependencies among different scales. In addition, Liu *et al.* [22] proposed DL-CNN model to extract sparse global features from entire scene image. Xie *et al.* [156] presented InterActive, a novel global CNN feature extraction algorithm which integrates high-level visual context with low-level neuron responses. InterActive increases the receptive field size of low-level neurons by allowing the supervision of the high-level neurons. Hayat *et al.* [26] designed a spatial unstructured layer to address the challenges of large-scale spatial layout deformations and scale variations. Along this way, Xie *et al.* [157] designed a Reversal Invariant Convolution (RI-Conv) layer so that they can obtain the identical representation for an image and its left-right reversed copy. Nevertheless, global CNN feature based methods have not fully exploited the underlying geometric and appearance variability of scene images.

The performance of global CNN features is greatly affected by the content of the input image. CNN models can extract generic global feature representations once trained on a sufficiently large and rich training dataset, as opposed to hand-crafted feature extraction methods. It is noteworthy that global representations obtained by scene-centric CNN models yield more enriched spatial information than those obtained using object-centric CNN models, arguably since global representations from scene-centric CNNs contain spatial correlations between objects and global scene properties [18], [19], [25]. In

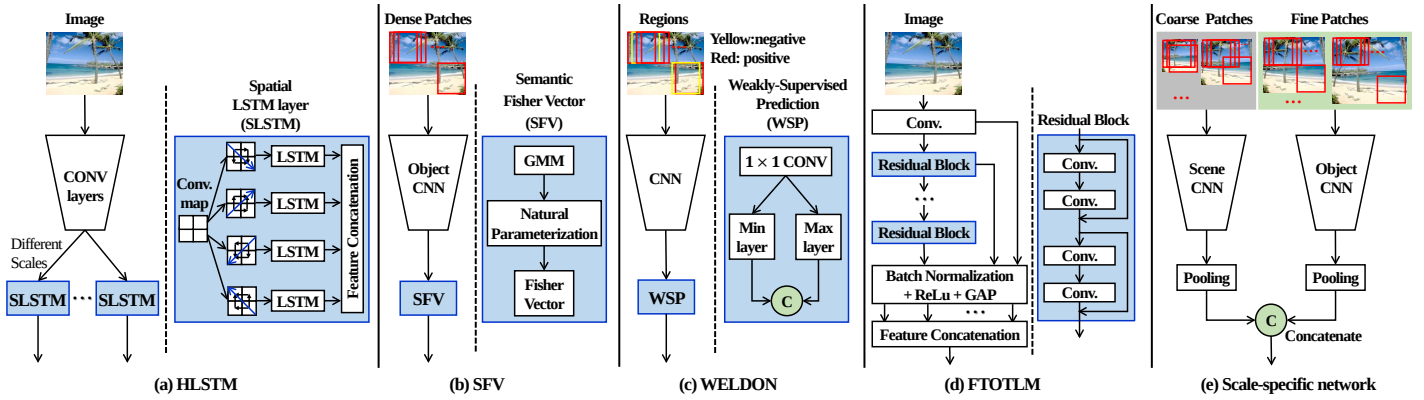



Fig. 8. Five typical architectures to extract CNN based scene representations (see Section 4.2), respectively. Hourglass architectures are backbone networks, such as AlexNet or VGGNet. (a) HLSTM [27], a global CNN feature based method, extracts deep feature from the whole image. Spatial LSTM is used to model 2D characteristics among the spatial layout of image regions. Moreover, Zuo *et al.* captured cross-scale contextual dependencies via multiple LSTM layers. (b) SFV [30], a spatially invariant feature based method, extract local features from dense patches. The highlight of SFV is to add a natural parameterization to transform the semantic space into a natural parameter space. (c) WELDON [34], a semantic feature based method, extracts deep features from top evidence (red) and negative instances (yellow). In WSP scheme, Durand *et al.* used the max layer and min layer to select positive and negative instances, respectively. (d) FTOTLM [21], a typical multi-layer feature based method, extracts deep feature from each residual block. (e) Scale-specific network [19], a multi-view feature based architecture, used scene-centric CNN extract deep features from coarse versions, while object-centric CNN is used to extract features from fine patches. Two types of deep features complement each other.

addition, Herranz *et al.* [19] showed that the performance of a scene recognition system depends on the entities in the scene image, *i.e.*, when the global features are extracted from images with chaotic background, the models performance is degraded compared to the cases that the object is isolated from the background or the image has a plain background. This suggests that the background may introduce some noise into the feature that weakens the performance. Since contour symmetry provides a perceptual advantage when human observers recognize complex real-world scenes, Rezanejad *et al.* [158] studied global CNN features from the full image and only contour information and showed that the performance of the full image as input is better, because CNN captures potential information from images. Nevertheless, they still concluded that contour is an auxiliary clue to improve recognition accuracy.

#### 4.2.2 Spatially Invariant Feature based Method

To alleviate the problems caused by sequential operations in the standard CNN, a body of alternatives [28], [40], [89] proposed spatially invariant feature based methods to maintain spatial robustness. The "spatially invariant" means that the output features are robust against the geometrical variations of the input image [43].

As shown in Fig. 8 (b), spatially invariant features are usually extracted from multiple local patches. The visualization of such a feature extraction process is shown in Fig. 6 (marked in green arrows). The entire process can be decomposed into five basic steps: 1) Local patch extraction: a given input image is divided into smaller local patches, which are used as the input to a CNN model, 2) Local feature extraction: deep features are extracted from either the CONV or FC layers of the model, 3) Codebook generation: this step is to generate a codebook with multiple codewords based on the extracted deep features from different regions of the image. The codewords usually are learned in an unsupervised way (*e.g.*, using GMM), 4) Spatially invariant feature generation: given the generated codebook, deep features are encoded into a spatially invariant representation, and 5) Class prediction: the representation input is classified into a predefined scene category.

As opposed to patch-based local feature extraction (each local feature is extracted from an original patch by independently using the CNN extractor), local features can also be extracted from the semantic CONV maps of a standard CNN [29], [44], [134], [159]. Specifically, since each cell (deep descriptor) of the feature map corresponds to one local image patch in the input image, each cell is regarded as a local feature. In this approach, the computation time is decreased, compared to independently processing of multiple spatial patches to obtain local features. For instance, Yoo *et al.* [29] replaced the FC layers with CONV layers to obtain large amount of local spatial features. They also used multi-scale CNN activations to achieve geometric robustness. Gao *et al.* [134] used a spatial pyramid to directly divide the activations into multi-level pyramids, which contain more discriminative spatial information.

The feature encoding technique, which aggregates the local features, is crucial in relating local features with the final feature representation, and it directly influences the accuracy and efficiency of the scene classification algorithms [71]. Improved Fisher Vector (IFV) [81], Vector of Locally Aggregated Descriptors (VLAD) [82], and Bag-of-Visual-Word (BoVW) [92] are among the popular and effective encoding techniques that are used in deep learning based methods. For instance, many methods, like FV-CNN [89], MFA-FS [42], and MFAFVNet [31], apply IFV encoding to obtain the image embedding as spatially invariant representations, while MOP-CNN [28], SDO [35], *etc* utilize VLAD to cluster local features. Noteworthily, the codebook selection and encoding procedures result in disjoint training of the model. To this end, some works proposed networks that are trained in an end-to-end manner, *e.g.*, NetVLAD [69], MFAFVNet [31], and VSAD [32].

Spatially invariant feature based methods are efficient to achieve geometric robustness. Nevertheless, the sliding windows based paradigm requires multi-resolution scanning with fixed aspect ratios, which is not suitable for arbitrary objects with variable sizes or aspect ratios in the scene image. Moreover, using dense patches may introduce noise into the final representation, which decreases the classification accuracy. Therefore, extracting semantic features from salient regions of the scene image can circumvent these drawbacks.

#### 4.2.3 Semantic Feature based Method

Processing all patches of the input image requires computational cost while yields redundant information. Object detection determines whether or not any instance of the salient regions is presented in an image [72]. Inspired by this, object detector based approaches allow identifying salient regions of the scene, which provide distinctive information about the context of the image.

Different methods have been put forward to effective saliency detection, such as selective search [160], unsupervised discovery [161], Multi-scale Combinatorial Grouping (MCG) [162], and object detection networks (*e.g.,* Fast RCNN [53], Faster RCNN [54], SSD [163], Yolo [164], [165]). For instance, since selective search combines the strengths of exhaustive search and segmentation, Liu *et al.* [146] used it to capture all possible semantic regions, and then used a pre-trained CNN to extract the feature maps of each region followed by a spatial pyramid to reduce map dimensions. Because the common objects or characteristics in different scenes lead to the commonality of different scenes, Cheng *et al.* [35] used a region proposal network [54] to extract the discriminative regions while discarde non-discriminative regions. These semantic feature based methods [35], [146] harvest many semantic regions, so encoding technology is adapted to aggregate key features, which pipeline is shown in Fig. 6 (red arrows).

On the other hand, some semantic feature based methods [33], [34] are based on weakly supervised learning, which directly predicts categories by several semantic features of the scene. For instance, Wu *et al.* [33] generated high-quality proposal regions by using MCG [162], and then used SVM on each scene category to prune outliers and redundant regions. Semantic features from different scale patches supply complementary cues, since the coarser scales deal with larger objects, while the finer levels provide smaller objects or object parts. In practice, they found two semantic features sufficient to represent the whole scene, comparable to multiple semantic features. Training a deep model only using a single salient region may result in a suboptimal performance due to the possible existence of outliers in the training set. Hence, multiple regions can be selected to train the model together [34]. As shown in Fig. 8 (c), Durand *et al.* [34] designed a Max layer to select the attention regions to enhance the discrimination. To provide a more robust strategy, they also designed a Min layer to capture the regions with the most negative evidence to further improve the model.

Although better performance can be obtained via using more semantic local features, semantic feature based methods deeply rely on the performance of object detection. Weak supervision settings (*i.e.,* without the patch labels of scene images) make it difficult to accurately identify the scene by the key information of an image [34]. Moreover, the error accumulation problem and extra computation cost also limit the development of semantic feature based methods [159].

#### 4.2.4 Multi-layer Feature based Method

Global feature based methods usually extract the high-layer CNN features, and feed them into a classifier to achieve classification task. Due to the compactness of such high-layer features, it is easy to miss some important slight clues [40], [166]. Features from different layers are complementary [33], [167]. Low-layer features generally capture small objects, while high-layer features capture big objects [33]. Moreover, semantic information of low-layer features is relatively less, but the object location is accurate [167]. To take full advantage of features from different layers, many methods [21], [38], [39], [48] used the high resolution features from the early layers along with the high semantic information of the features from the latest layers of hierarchical models (*e.g.,* CNNs).

As shown in Fig. 8 (d), typical multi-layer feature formation process includes: 1) Feature extraction: the outputs (feature maps) of certain layers are extracted as deep features, 2) Feature vectorization: vectorize the extracted feature maps, 3) Multi-layer feature combination: multiple features from different layers are combined into a single feature vector, and 4) Feature classification: classify the given scene image based on the obtained combined feature.

Although using all features from different layers seems to improve the final representation, it likely increases the chance of overfitting, and thus hurts performance [37]. Therefore, many methods [21], [37], [38], [39], [48] only extract features from certain layers. For instance, Xie *et al.* [38] constructed two dictionary-based representations, Convolution Fisher Vector (CFV), and Mid-level Local Discriminative Representation (MLR) to classify subsidiarily scene images. Tang *et al.* [39] divided GoogLeNet layers into three parts from bottom to top and extracted final feature maps of each part. Liu *et al.* [21] captured feature maps from each residual block from ResNet independently. Song *et al.* [48] selected discriminative combinations from different layers and different network branches via minimizing a weighted sum of the probability of error and the average correlation coefficient. Yang *et al.* [37] used greedily select to explore the best layer combinations.

Feature fusion in multi-layer feature based methods is another important direction. Feature fusion techniques are mainly divided into two groups [168], [169], [170]: 1) Early fusion: extracting multi-layer features and merging them into a comprehensive feature for scene classification, and 2) Late fusion: directly learning each multi-layer feature via a supervised learner, which enforces the features to be directly sensitive to the category label, and then merging them into a final feature. Although the performance of late fusion is better, it is more complex and time-consuming, so early fusion is more popular [21], [38], [39], [48]. In addition, addition and product rules are usually applied to combine multiple features [39]. Since the feature spaces in different layers are disparate, product rule is better than addition rule to fusing features, and empirical experiments on [39] also show this statement. Moreover, Tang *et al.* [39] proposed two strategies to fuse multi-layer features, *i.e.,* 'fusion with score' and 'fusion with features'. Fusion with score technique has obtained a better performance over fusion with feature thanks to the end-to-end training.

#### 4.2.5 Multiple-view Feature based Method

Describing a complex scene using just a single and compact feature representation is a non-trivial task. Hence, there has been a lot of effort to compute a comprehensive representation of a scene by integrating multiple features obtained from complementary CNN models [24], [32], [41], [171], [172], [173].

Feature, from the networks trained on different datasets, usually are complementary. As shown in Fig. 8 (e), Herranz *et al.* [19] found the best scale response of object-centric CNNs and scene-centric CNNs, and they combine the knowledge in a scale-adaptive way via either object-centric CNNs or scene-centric CNNs. Along this way, Wang *et al.* [173] used an object-centric CNN to carry information about object depicted in the

image, while a scene-centric CNN was used to capture global scene information. The authors in [32] designed PatchNet, a weakly supervised learning method, which uses image-level supervision information as the supervision signal for effective extraction of the patch-level features. To enhance the recognition performance, Scene-PatchNet and Object-PatchNet jointly used to extract features for each patch.

Employing the complementary CNN architectures is essential for obtaining discriminative multi-view feature representations. Wang *et al.* [41] proposed a multi-resolution CNN (MR-CNN) architecture to capture visual content in multiple scale images. In their work, normal BN-Inception [174] is used to extract coarse resolution features, while deeper BN-Inception is employed to extract fine resolution features. Jin *et al.* [175] used global features and spatially invariant features to account for both the coarse layout of the scene and the transient objects. Sun *et al.* [24] separately extracted three representations, *i.e.,* object semantics representation, contextual information, and global appearance, from discriminative views, which are complementarity to each other. Specifically, the object semantic features of the scene image are extracted by a CNN followed by spatial fisher vectors, while the deep feature of a multi-direction LSTM-based model represents contextual information, and the FC feature represents global appearance. Li *et al.* [171] used off-the-shelf ResNet18 [87] to generate discriminative attention maps, which is used as an explicit input of CNN together with the original image. Using global features extracted by ResNet-18 and attention map features extracted from the spatial feature transformer network, the attention map features are multiplied to the global features for adaptive feature refinement so that the network focuses on the most discriminative parts.

### 4.3 Strategies for Improving Scene Representation

To obtain more discriminative representations for scene classification, a range of strategies has been proposed. Four major categories (*i.e.,* encoding strategy, attention strategy, contextual strategy, and regularization strategy) will be discussed below.

#### *4.3.1 Encoding strategy*

Although the current driving force has been the incorporation of CNNs, encoding technology of the first generation methods have also been adapted in deep learning based methods. Fisher Vector (FV) coding [81], [105] is an encoding technique commonly used in scene classification. Fisher vector stores the mean and the covariance deviation vectors per component of the GMM and each element of the local features together. Thanks to the covariance deviation vectors, FV encoding leads to excellent results. Moreover, it is empirically proven that Fisher vectors are complementary to global CNN features [30], [31], [38], [40], [42], [47]. Therefore, this survey takes FV-based approaches as the major cue and discusses the adapted combination of encoding technology and deep learning.

Generally, CONV features and FC features are regarded as Bags of Features (BoF), they can be readily modeled by the Gaussian Mixture Model followed by Fisher Vector (GMM-FV) [30], [31]. To avoid the computation of the FC layers, Cimpoi *et al.* [89] utilized GMM-FV to aggregate BoF from different CONV layers, respectively. Comparing their experiment results, they asserted that the last CONV features can more effectively represent scenes. To rescue the fine-grained information of early/middle layers, Guo *et al.* [40] proposed Fisher Convolutional Vector (FCV) to encode the

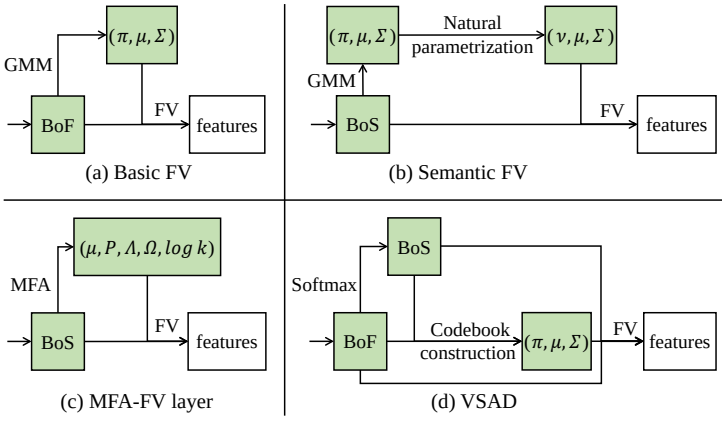



Fig. 9. Structure comparisons of (a) basic Fisher vector [89] and its variations. BoF denotes bag of features, while BoS represents bag of semantic probabilities. (b) In semantic FV [30], natural parameterization is added to map multinomial distribution (*i.e.,* $\pi$) to its natural parameter space (*i.e.,* $\nu$). (c) In MFAFVNet [31], GMM is replaced with MFA to build codebook. (d) In VSAD [32], codebook is constructed via exploiting semantics (*i.e.,* BoS) to aggregate local features (*i.e.,* BoF).

feature maps from multiple CONV layers. Wang *et al.* [47] extracted the feature maps from RGB, HHA, and surface normal images, and then directly encoded these maps by FV coding. In addition, through the performance comparisons of GMM-FV encoding on CONV features and FC features, respectively, Dixit *et al.* [30] asserted that the FC features is more effective for scene classification. However, since the CONV features and FC features do not derive from semantic probability space, it is likely to be both less discriminant and less abstract than the truly semantic features [30], [89]. The activations of Softmax layer are probability vectors, inhabiting the probability simplex, which are more abstract and semantic, but it is difficult to implement an effective invariant coding (*e.g.,* GMM-FV) [30], [96]. To this end, Dixit *et al.* [30] proposed an indirect FV implementation to aggregate these semantic probability features, *i.e.,* adding a step to convert semantic multinomials from probability space to the natural parameter space, as shown in Fig. 9 (b). Inspired by FV and VLAD, Wang *et al.* [32] proposed Vector of Semantically Aggregated Descriptors (VSAD) to encode the probability features, as shown in Fig. 9 (d). Comparing the discriminant probability learned by the weakly-supervised method (PatchNet) with the generative probability from an unsupervised method (GMM), the results show that the discriminant probability is more expressive in aggregating local features. From the above discussion, representation encoding local features on probability space outperforms that on non-probability space.

Deep features usually are high dimensional ones. Therefore, more Gaussian kernels are needed to accurately model the feature space [133]. However, this would a lot of overhead to the computations and, hence, it is not efficient. Liu *et al.* [133] empirically proved that the discriminative power of FV features increases slowly as the number of Gaussian kernels increases. Therefore, dimensionality reduction of the features is very important, as it directly affects the computational efficiency. A wide range of approaches [29], [30], [32], [40], [42], [60], [133] used poplular dimensionality reduction techniques, Principal Component Analysis (PCA), for pre-processing of the local features. Moreover, Liu *et al.* [133] drew local features from Gaussian distribution with a nearly zero mean, which ensures the

sparsity of the resulting FV. Wang *et al.* [47] enforced inter-component sparsity of GMM-FV features via component regularization to discount unnecessary components.

Due to the non-linear property of deep features and a limited ability of the covariance of GMM, a large number of diagonal GMM components are required to model deep features so that the FV has very high dimensions [31], [42]. To address this issue, Dixit *et al.* [42] proposed MFA-FS, in which GMM is replaced by Mixtures of Factor Analysis (MFA) [176], [177], *i.e.*, a set of local linear subspaces is used to capture non-linear features. MFA-FS performs well but does not support end-to-end training. However, end-to-end training is more efficient than any disjoint training process [31]. Therefore, Li *et al.* [31] proposed MFAFVnet, an improved variant of MFA-FS [42], which is conveniently embedded into the state-of-the-art network architectures. Fig. 9 (c) shows the MFA-FV layer of MFAFVNet, compared with the other two structures.

In FV coding, local features are assumed to be independent and identically distributed (iid), which violates intrinsic image attributes that these patches are not iid. To this end, Cinbis *et al.* [60] introduced a non-iid model via treating the model parameters as latent variables, rendering features related locally. Later, Wei *et al.* [46] proposed a correlated topic vector, treated as an evolution oriented from Fisher kernel framework, to explore latent semantics, and consider semantic correlation.

#### 4.3.2 Attention strategy

As opposed to semantic feature based methods, focusing on key cues generally from original images, attention policy aims to select distinguishing features from the extracted feature space [43], [44], [74], [178]. The attention maps are learned without any explicit training signal, rather task-related loss function alone provides the training signal for the attention weights. Noteworthily, GAP-CNN [23] is a simple attention method, widely used in many works [179], [180], [181].

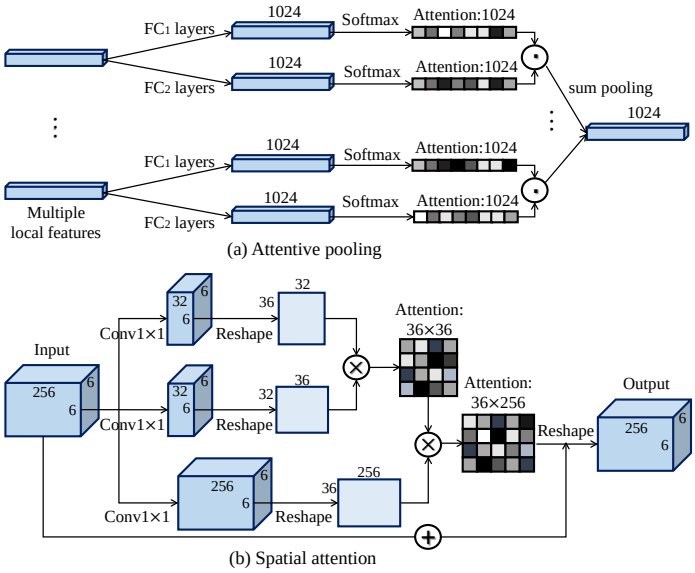



Fig. 10. Illustrations of two attention strategies. (a) Attentive pooling [43] maintains spatial invariance meanwhile discerns discriminative semantic cues. (b) Spatial attention [44] is used to enhance the local feature selection.

Since some salient regions may mislead classification, Li *et al.* [43] designed a class-agnostic attentive pooling to incorporate the attention mechanism for enhancing the discriminative capability of pooling, as shown in Fig. 10 (a). Since the same semantic cue has different roles for different types of scenes in some cases, they designed a class-aware attentive pooling strategy, including intra-modality attentive pooling and cross-aware attentive pooling, to learning class-special attention weights and the contributions of each modality, respectively. Here the attention mechanism is used to enhance discrimination in each modality, while it is also used to further fuse the learned discriminate semantic cues across RGB and depth modalities. To enhance local feature selection, Xiong *et al.* [44], [159] designed a spatial attention module, shown in Fig. 10 (b), to generate attention masks. The attention masks of RGB and depth images are encouraged to be similar, and then learn the modal-consistent features. Inspired by the idea that a specific object is associated with the particular scene, Seong *et al.* [182] proposed correlative context gating with attention maps to activate object features. The attention module is a Softmax function in essence, then the overall architecture is easy to be designed as an end-to-end paradigm.

Attention maps can also be computed based on different sources of information. Xia *et al.* [178] designed a Weakly Supervised Attention Map (WS-AM) by proposing a gradient-weighted class activation mapping technique and privileging weakly supervised information. Unlike other attention modules, WS-AM does not use image-level label information since the backbone network is just a pre-trained CNN. In the work of Lopez *et al.* [74], the attention maps are generated by extracting meaningful features and encoding contextual information from a semantic segmentation score map. The semantic-based attention map is used to gate color features so that more discriminative features are obtained.

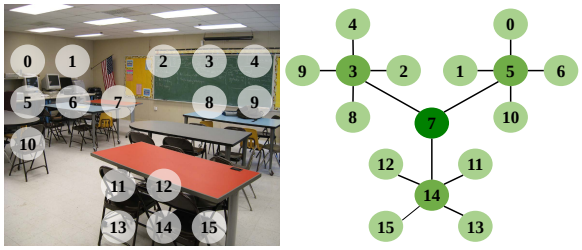
#### 4.3.3 Contextual strategy

Contextual information (the correlations among image regions, and local features), and objects/scenes may provide beneficial information in disambiguating visual words [183]. However, convolution and pooling kernels are locally performed on image regions separately, and encoding technologies usually integrate multiple local features into an unstructured feature. As a result, contextual correlations among different regions have not been taken into account [184]. To this end, contextual correlations have been further explored to focus on the global layout or local region coherence [185].

The contextual relations for scene classification can broadly be grouped into two major categories: 1) spatial contextual relation: the correlations of neighboring regions, in which capturing spatial contextual relation usually encounters the problem of incomplete regions or noise caused by predefined grid patches, and 2) semantic contextual relation: the relations of salient regions. The network to extract semantic relations is often a two-stage framework (*i.e.*, detecting objects and classifying scenes). Therefore, accuracy is also influenced by object detection. Generally, there are three types of algorithms to capture contextual relations: 1) sequential model, like RNN [186] and LSTM [153], and 2) graph-related model, such as Markov Random Field (MRF) [187], [188], Correlated Topic Model (CTM) [46], [97] and graph convolution [189], [190], [191].

**Sequential Model.** With the success of sequential models, such as RNN and LSTM, capturing the sequential information among local regions has shown promising performance for scene classification [36]. Spatial dependencies are captured from direct or indirect connections between each region and its


(a) Sequential model



(b) Graph-related model

Fig. 11. Illustrations of a sequence model and a graph model to related contextual information. (a) Local feature is extracted from each salient region via a CNN, and a bidirectional LSTM is used to model synchronously many-to-many local feature relations [36]. (b) Graph is constructed by assigning selected key features to graph nodes (including the center nodes, sub-center nodes and other nodes) [136].

surrounding neighbors. Zuo *et al.* [27] stacked multi-directional LSTM layers on the top of CONV feature maps to encode spatial contextual information in scene images. Furthermore, a hierarchical strategy was adopted to capture cross-scale contextual information. Like the work [27], Sun *et al.* [24] bypassed two sets of multi-directional LSTM layers on the CONV feature maps. In their framework, the outputs of all LSTM layers are concatenated to form a contextual representation. In addition, the works [36], [192] captured semantic contextual knowledge from variable salient regions. In [192], two types of representations, *i.e.*, COOR and SOOR, are proposed to describe object-to-object relations. Herein, COOR adapts the co-occurring frequency to represent the object-to-object relations, while SOOR is encoded with sequential model via regarding object sequences as sentences. Rooted in the work of Javed *et al.* [193], Laranjeira *et al.* [36] proposed a bidirectional LSTM to capture the contextual relations of regions of interest, as shown in Fig. 11 (a). Their model supports variable length sequences, because the number of object parts of each image are different.

**Graph-related Model.** The sequential models often simplify the contextual relations, while graph-related models can explore more complicated structural layouts. Song *et al.* [194] proposed a joint context model that uses MRFs to combine multiple scales, spatial relations, and multiple features among neighboring semantic multinomials, showing that this method can discover consistent co-occurrence patterns and filter out noisy ones. Based on CTM, Wei *et al.* [46] captured relations among latent themes as a semantic feature, *i.e.*, corrected topic vector (CTV). Later, with the development of Graph Neural Network (GNN) [190], [191], graph convolution has become increasingly popular to model contextual information for scene classification. Yuan *et al.* [136] used spectral graph convolution to mine the relations among the selected local CNN features, as shown in Fig. 11 (b). To use the complementary cues of multiple modalities, Yuan *et al.* also considered the inter-modality correlations of RGB and depth modalities through a cross-modal graph. Chen *et al.* [45] used graph convolution [195] to model the more complex spatial structural layouts via pre-defining the features of discriminative regions as graph nodes. However, the spatial relation overlooks the semantic meanings of regions. To address this issue, Chen *et al.* also defined a similarity subgraph as a complement to the spatial subgraph.

#### 4.3.4 Regularization strategy

The training classifier not only requires a classification loss function, but it may also need multi-task learning with different regularization terms to reduce generalization error. The regularization strategies for scene classification mainly include sparse regularization, structured regularization, and supervised regularization.

**Sparse Regularization**. Sparse regularization is a technique to reduce the complexity of the model to prevent overfitting and even improve generalization ability. Many works [22], [40], [146], [196], [197] include $\ell_0$, $\ell_1$, or $\ell_2$ norms to the base loss function for learning sparse features. For example, the sparse reconstruction term in [146] encourages the learned representations to be significantly informative. The loss in [22] combines the strength of the Mahalanobis and Euclidean distances to balance the accuracy and the generalization ability.

**Structured Regularization.** Minimizing the triplet loss function minimizes the distance between the anchor and positive features with the same class labels while maximizing the distance between the anchor and negative features with one different class labels. In addition, according to the maximum margin theory in learning [198], hinge distance focus on the hard training samples. Hence, many research efforts [44], [52], [159], [196], [199] have utilized structured regularization of the triplet loss with hinge distance to learn robust feature representations. The structured regularization term is $\sum_{a,p,n} max(d(x_a, x_p) - d(x_a, x_n) + \alpha, 0)$, where $x_a, x_p, x_n$ are anchor, positive, negative features, and $\alpha$ is an adjustable parameter, while the function $d(x, y)$ denotes calculating a distance of $x$ and $y$. The structured regularization term promotes exemplar selection, while it also ignores noisy training examples that might overwhelm the useful discriminative patterns [159], [196].

**Supervised Regularization.** Supervised regularization uses the label information for tuning the intermediate feature maps. The supervised regularization is generally expressed in terms of $\sum_i d(y_i, f(x_i))$, where $x_i and y_i$ denote the middle-layer activated features and real label of the image $i$, respectively, and $f(x_i)$ is a predicted label. For example, Guo *et al.* [40] utilized an auxiliary loss to directly propagate the label information to the CONV layers, and thus accurately captures the information of local objects and fine structures in the CONV layers. Similarly, these alternatives [44], [136], [159] used supervised regularization to learn modal-specific features.

**Others.** Extracting discriminative features by incorporating different regularization techniques has been always a mainstream topic in scene classification. For example, label consistent regularization [146] guarantees that inputs from different categories have discriminative responses. The shareable constraint in [196] can learn a flexible number of filters to represent common patterns across different categories. Clustering loss in [175] is utilized to further fine-tune confusing clusters to overcome the intra-class variation issues inherent. Since assigning soft labels to the samples cause a degree of ambiguity, which reaps high benefits when increasing the number of scene categories [94], [200], Wang *et al.* [41] improved generalization ability by exploiting soft labels contained in knowledge networks

as a bias term of the loss function. Noteworthily, optimizing proper loss function can pick up effective patches for image classification. Fast RCNN [53] and Faster RCNN [54] use regression loss to learn effective region proposals. Wu *et al.* [33] adopted one-class SVMs [201] as discriminative models to get meta-objects. Inspired by MANTRA [202], the main intuition in [34] is to equip each possible output with pairs of latent variables, *i.e.*, top positive and negative patches, via optimizing max+min prediction problem.

Nearly all multi-task learning approaches using regularization aim to find a trade-off between conflicting requirements such as accuracy, generalization robustness, and efficiency. Researchers apply completely different supervision information to a variety of auxiliary tasks in an effort to facilitate the convergence of the major scene classification task [40].

## 4.4 RGB-D Scene Classification

RGB modality provides the intensity of the colors and texture cues, while depth modality carries information regarding the distance of the scene surfaces from a viewpoint. The depth information is invariant to lighting and color variations, and contains geometrical and shape cues, which is useful for scene representation [203], [204], [205]. Moreover, HHA data [139], an encoding result of depth image, depth information presents a certain color modality, which somewhat is similar to the RGB image. Hence, some CNNs trained on RGB images can transfer their knowledge and be used on HHA data.

The depth information of RGB-D image can further improve the performance of CNN models compared to RGB images [135]. For the task of RGB-D scene classification, except for exploring suitable RGB features, described in Section 4.2, there exists another two main problems, *i.e.*, 1) how to extract depth-specific features and 2) how to properly fuse features of RGB and depth modalities.

### 4.4.1 Depth-specific feature learning

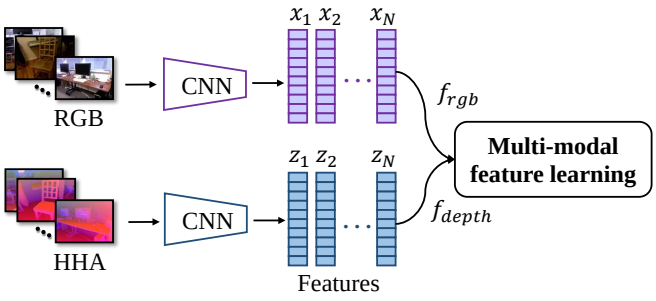
Depth information is usually scarce compared to RGB data. Therefore, it is non-trivial to train CNNs only on limited depth data to achieve depth-specific models [135], *i.e.*, depth-CNN. Hence, different training strategies are employed to train CNNs using limited amount of depth images.

**Fine-tuning RGB-CNNs for depth images.** Due to the availability of RGB data, many models [47], [51], [52] are first pre-trained on large-scale RGB datasets, such as ImageNet and Places, followed by fine-tuning on depth data. Fine-tuning only updates the last few FC layers, while the parameters of the previous layers are not adjusted. Therefore, the fine-tuned models layers do not fully leverage depth data [135]. However, abstract representations of early CONV layers play a crucial role in computing deep features using different modalities. Weakly-supervised learning and semi-supervised learning can enforce explicit adaptation in the previous layers.

**Weak-supervised learning with patches of depth images.** Song *et al.* [48], [135] proposed to learn depth features from scratch using weakly supervised learning. Song *et al.* [135] pointed out that the diversity and complexity of patterns in the depth images are significantly lower than those in the RGB images. Therefore, they designed a Depth-CNN (DCNN) with fewer layers for depth features extraction. They also trained the DCNN by three strategies of freezing, fine-tuning, and training from scratch to adequately capture depth information. Nevertheless, weakly-supervised learning is sensitive to the noise in the training data. As a result, the extracted features may not have good discriminative quality for classification.

**Semi-supervised learning with unlabeled images.** Due to the convenient collection of unlabeled RGB-D data, semi-supervised learning can also be employed in the training of CNNs with a limited number of labeled samples compared to very large size of unlabeled data [49], [205]. Cheng *et al.* [205] trained a CNN using a very limited number of labeled RGB-D images while a massive amount of unlabeled RGB-D images via a co-training algorithm to preserve diversity. Subsequently, Du *et al.* [49] developed an encoder-decoder model to construct paired complementary-modal data of the input. In particular, the encoder is used as a modality-specific network to extract specific features for the subsequent classification task.

### 4.4.2 Multiple modality fusion




(a) Feature-level modal combination

Three typical combination ways:

(1) $F_i = concat(x_i, z_i)$

(2) $F_i = \lambda_{rgb} x_i + \lambda_{depth} z_i$

(3) $F_i = W_{rgb} x_i + W_{depth} z_i$

$\lambda$: scalar, $W$: a matrix, $F_i$: combined feature

(b) Consistent- feature based modal fusion

Three typical consistent loss terms $L_c$:

(1) $L_c = \sum_{i,j}[d(f_{rgb}(x_i), f_{rgb}(x_j) - d(f_{depth}(z_i), f_{depth}(z_j)]$

(2) $L_c = d(f_{rgb}, f_{depth})$

(3) $L_c = \sum_i d(f_{rgb}(x_i), f_{depth}(z_i))$

$d(x, y)$: a distance of $x$ and $y$

(c) Distinctive-feature based modal fusion

Three typical distinction loss terms $L_d$:

(1) $L_{rgb\_d} = \sum_{a,p,n}[d(f_{rgb}(x_a), f_{rgb}(x_p) - d(f_{rgb}(x_a), f_{rgb}(x_n)]$

(2) $L_{rgb\_d} = \sum_i d(f_{rgb}(x_i), y_i))$

(3) $L_d = \sum_i cos(f_{rgb}(x_i), f_{depth}(z_i))$

(a, p, n): triplets (anchor, positive, negative),

$y_i$: the true label of Image$_i$

Fig. 12. Illustrations of multi-modal feature learning. (a) Three popular ways to achieve feature combination: directly concatenate features, combine weighted features and combine features with linear converting. (b) Three methods to achieve modal-consistent: minimize the pairwise distances between modalities [52]; encourage the attention maps of modalities similar [159]; minimize the distances between modalities [44], [199]. (c) Three strategies to achieve modal-distinctive: learn the model structure via triplet loss [52], [199]; use label information to guide modal-specific learning [136], [159]; minimize cosine similarity between modalities [44].

Various modality fusion methods [44], [199], [206] have been put forward to combine the information of different modalities to further enhance the performance of the

classification model. The fusion strategies are mainly divided into four categories, *i.e.,* image-level modal combination, feature-level modal combination, consistent feature based fusion, and distinctive feature based fusion. Fig. 12 shows illustrations of the late three categories. Despite the existence of four different fusion categories, some works [44], [47], [199] combine multiple fusion strategies to achieve better performance for scene classification.

**Image-level modal fusion.** RGB modality and depth modality are directly stacked together as input [50]. Liao *et al.* [50] used a single network to extract deep features from RGB-D images. Couprie *et al.* [207] constructed the RGB-D Laplacian pyramid with the RGB and depth modalities. Due to the pixel value difference of RGB and depth modalities, image-level fusion may fail to adequately exploit the consistency and complementary information between the two modalities [52].

**Feature-level modal combination.** Song *et al.* [48] proposed a multi-modal combination approach to select discriminative combinations of layers from different source models. They concatenated RGB and depth features for not losing the correlation between the RGB and depth data. Reducing the redundancy of features can significantly improve the performance when RGB and depth features have correlations; especially, in the case of extracting depth features merely via RGB-CNNs [135]. Because of such correlation, direct concatenation of features may result in redundancy of information. To avoid this issue, Du *et al.* [49] performed global average pooling to reduce the feature dimensions after concatenating modality-specific features. Wang *et al.* [47] used the modality regularization based on exclusive group lasso to ensure feature sparsity and co-existence, while features within a modality are encouraged to compete. Li *et al.* [43] used an attention module to discern discriminative semantic cues from intra- and cross-modalities. Moreover, Cheng *et al.* [51] proposed a gated fusion layer to adjust the RGB and depth contributions on image pixels.

**Consistent-feature based modal fusion.** Images may suffer from missing information or noise pollution so that multi-modal features are not consistent, hence it is essential to exploit the correlation between different modalities to exclude such issue. To drive feature consistency of different modalities, Zhu *et al.* [52] introduced an inter-modality correlation term to minimize pairwise distances of two modalities from the same class, while maximize pairwise distances from different classes. Zheng *et al.* [208] used multi-task metric learning to learn linear transformations of RGB and depth features, making full use of inter-modal relations. Li *et al.* [199] learned a correlative embedding module between the RGB and depth features inspired by Canonical Correlation Analysis [209], [210]. Xiong *et al.* [44], [159] proposed a learning approach to encourage two modal-specific networks to focus on features with similar spatial positions to learn more discriminative modal-consistent features.

**Distinctive-feature based modal fusion.** In addition to constructing multimodal consistent features, features can also be processed separately to increase discriminative capability. For instance, Li *et al.* [199] and Zhu *et al.* [52] adopted structured regularization in the triplet loss function, in which to encourage the model to learn the modal-specific features under the supervision of this regularization. Li *et al.* [199] designed a distinctive embedding module for each modality to learn distinctive features. Using labels for separate supervision of model-specific representation learning for each modality is also another technique of individual processing [44], [136], [159]. Moreover, by minimizing the feature correlation, Xiong *et al.* [44] learned the modal distinctive features as the RGB and depth modalities have different characteristics.

## 5 Performance Comparison

### 5.1 Performance on RGB scene data

TABLE 2
Performance comparison of pre-deep learning approaches on two benchmark scene datasets. For each dataset, the highest classification score is highlighted.

| Group | Methods | Venue | Results | |
|---|---|---|---|---|
| | | | Scene15 | MIT67 |
| Feature Engineering | CENTRIST [79] | TPAMI2010 | 83.9 | 36.9 |
| | OTC [16] | ECCV2014 | 84.4 | 47.3 |
| | GG (KL-div.) [211] | CVPR2010 | 86.1 | 45.5 |
| | LPC [212] | ECCV2010 | 83.4 | 39.6 |
| | NNbMF [213] | CVIU2011 | 82.1 | 47.0 |
| | RBoVW [214] | CVPR2012 | 78.6 | 37.9 |
| | SCP+SPM [215] | CVPR2012 | 80.4 | 33.7 |
| | BoF+HoG [216] | CVPR2013 | 85.6 | 58.9 |
| | **IFV** [84] | CVPR2013 | ***89.2*** | 60.8 |
| | **MVED** [217] | NeurIPS2013 | – | ***64.0*** |
| | MR-BoVW [218] | PR2013 | 82.9 | – |
| | VC+VQ [219] | CVPR2013 | 85.4 | 52.3 |
| | MMDL [220] | ICML2013 | 86.4 | 50.2 |
| | ISPR [221] | CVPR2014 | 85.1 | 50.1 |
| | DPM [102] | ICCV2011 | – | 30.4 |
| | LPR [116] | ECCV2012 | 85.8 | 44.8 |
| | Hybrid-Parts [222] | ECCV2012 | 84.7 | 39.8 |
| | SPMSM [96] | ECCV2012 | 82.3 | 44.0 |
| | Object Bank [15] | NeurIPS2010 | 80.9 | 37.6 |
| Classifier | BRSP [223] | ECCV2012 | 88.1 | – |
| | DPD [149] | ICCV2013 | 86.0 | 51.4 |

Table 2 summarizes the performance of the recent pre-deep learning methods from 2010 to 2014. We can observe that most of these approaches achieve good performance by handcrafted feature engineering, while a few attempts were dedicated to designing a more powerful classifier. The accuracy on Scene15 has been up to 89.2% achieved by Improved Fisher Vector (IFV) [81], while Mid-level Visual Element Discovery (MVED) [217] achieved the highest accuracy (*i.e.,* 64.0%) on MIT67. Nevertheless, experimental results demonstrate these handcrafted feature methods have not been capable of performing at a level sufficient for publicly available datasets.

In contrast, CNN-based methods have quickly demonstrated their strengths in scene classification. Table 3 compares the performance of deep models for scene classification on RGB datasets. To gain insight into the performance of the presented methods, we also provided input information, feature information, and architecture of each method. The results show that a simple deep model (*i.e.,* AlexNet), which is trained on ImageNet, achieves 84.23%, 56.79%, and 42.61% accuracy on Scene15, MIT67, and SUN397 datasets, respectively. This accuracy is comparable with the best non-deep learning methods in Table 2. Starting from the generic deep models [17], [18], [25], CNN-based methods improve steadily when more effective strategies (*e.g.,* improved encoding strategy [30], [31], attention strategy [23], [43], contextual strategy [36], [45] and regularization strategy [22], [40]) have been introduced. As a result, nearly all the approaches yielded an accuracy of 90% on the Scene15 dataset. Moreover, FTOTLM [21] combined with a novel data augmentation outperforms other state-of-the-art

TABLE 3
Performance (%) summarization of some representative methods on popular benchmark datasets. All scores are quoted directly from the original papers. For each dataset, the highest three classification scores are highlighted. Some abbreviations. Column "Scheme": Whole Image (WI), Dense Patches (DP), Regional Proposals (RP); Column "Init."(Initialization): ImageNet (IN), Places205 (PL1), Places365 (PL2).

| Group | Method | Input information | | | Feature Information | | Architecture | | Results (RGB) | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Scale | Scheme | Data Aug. | Aggregation | Dimension | Init. | Backbone | Scene15 | MIT67 | SUN397 |
| Global CNN features based methods | ImageNet-CNN [17] | Single | WI | × | pooling | 4,096 | IN | AlexNet | 84.2 | 56.8 | 42.6 |
| | PL1-CNN [18] | Single | WI | × | pooling | 4,096 | PL1 | VGGNet | 91.6 | 79.8 | 62.0 |
| | PL2-CNN [25] | Single | WI | × | pooling | 4,096 | PL2 | VGGNet | 92.0 | 76.5 | 63.2 |
| | S2ICA [26] | Multi | DP | √ | pooling | 8,192 | IN | AlexNet | 93.1 | 71.2 | – |
| | GAP-CNN [23] | Single | WI | √ | GAP | 4,096 | PL1 | GoogLeNet | 88.3 | 66.6 | 51.3 |
| | InterActive [156] | Single | WI | × | pooling | 4,096 | IN | VGG19 | – | 78.7 | 63.0 |
| | C-HLSTM [27] | Multi | WI | √ | LSTM | 4,096 | PL1 | AlexNet | – | 75.7 | 60.3 |
| | DL-CNN [22] | Single | WI | × | DL | – | PL1 | VGGNet | 96.0 | 86.4 | 70.1 |
| Spatially invariant features based methods | SCFVC [133] | Single | DP | × | FV | 200,000 | IN | AlexNet | – | 68.2 | – |
| | MOP-CNN [28] | Multi | DP | × | VLAD | 12,288 | IN | AlexNet | – | 68.9 | 52.0 |
| | DSP [134] | Multi | WI | × | FV | 12,288 | IN | VGGNet | 91.8 | 78.3 | 59.8 |
| | MPP-CNN [29] | Multi | RP | × | FV | 65,536 | IN | AlexNet | – | 80.8 | – |
| | SFV [30] | Multi | DP, WI | × | FV | 9,216 | IN | AlexNet | – | 72.8 | 54.4 |
| | FV-CNN [89] | Multi | RP, WI | × | FV | 4,096 | IN | VGGNet | – | 81.0 | – |
| | LatMoG [60] | Multi | RP | × | FV | – | IN | AlexNet | – | 69.1 | – |
| | DUCA [20] | Single | DP | √ | CSMC | 4,096 | IN | AlexNet | 94.5 | 78.8 | – |
| | D3 [224] | Single | DP | × | D3, FV | 1,048,576 | IN | VGG16 | 92.8 | 77.1 | 61.5 |
| | MFA-FS [42] | Multi | DP | × | MFA-FV | 5,000 | IN | VGGNet | – | 81.4 | 63.3 |
| | CTV [46] | Multi | WI | × | CTV | – | PL1 | AlexNet | – | 73.9 | 58.4 |
| | MFAFVNet [31] | Multi | DP | × | MFA-FV | 500,000 | IN | VGG19 | – | 82.7 | 64.6 |
| | EMFS [194] | Multi | DP | × | SM | 4,096 | IN, PL2 | VGGNet | – | 86.5 | 72.6 |
| | VSAD [32] | Multi | DP | √ | VSAD | 25,600 | IN, PL1 | VGG16 | – | 86.1 | 72.0 |
| | LLC [197] | Single | DP | × | SSE | 3,072 | IN, PL2 | VGG16 | – | 79.6 | 57.5 |
| Semantic features based methods | URDL [146] | Multi | SS | √ | pooling | 4,096+ | IN | VGG16 | 91.2 | 71.9 | – |
| | MetaObject-CNN [33] | Multi | RP | √ | LSAQ | 4,096 | PL1 | AlexNet | – | 78.9 | 58.1 |
| | SOAL [225] | Multi | RP | × | CRF | 1,024 | PL1 | VGGNet | – | 82.5 | 75.5 |
| | WELDON [34] | Single | WI | × | pooling | 4,096 | IN | VGG16 | 94.3 | 78.0 | – |
| | Adi-Red [226] | Multi | DisNet | × | GAP | 12,288 | IN, PL1-2 | ResNet | – | – | 73.6 |
| | **SDO** [35] | Multi | OMD | × | VLAD | 8,192 | PL1 | VGGNet | ***95.9*** | 86.8 | 73.4 |
| | **M2M BiLSTM** [36] | Single | SS | × | LSTM | – | IN | ResNet | ***96.3*** | 88.3 | 71.8 |
| | LGN [45] | Single | WI | × | LGN | 8,192 | PL2 | ResNet | – | 88.1 | 74.1 |
| Multilayer features based methods | Deep19-DAG [37] | Single | WI | × | pooling | 6,144 | IN | VGG19 | 92.9 | 77.5 | 56.2 |
| | Hybrid CNNs [38] | Multi | SS | × | FV | 12,288+ | IN, PL1 | VGGNet | – | 82.3 | 64.5 |
| | G-MS2F [39] | Single | WI | √ | pooling | 3,072 | IN, PL1 | GoogLeNet | 93.2 | 80.0 | 65.1 |
| | FTOTLM [21] | Single | WI | × | GAP | 3,968 | IN, PL2 | ResNet | 94.0 | 74.6 | 65.5 |
| | **FTOTLM Aug.** [21] | Single | WI | √ | GAP | 3,968 | IN, PL2 | ResNet | ***97.4****| ***94.1****| ***85.2**** |
| Multiview features based methods | [227] | Multi | WI | √ | pooling | 8,192 | IN, aratio | AlexNet | 92.1 | 70.1 | 54.7 |
| | Scale-specific CNNs [19] | Multi | Crops | × | pooling | 4,096 | IN, PL1 | VGGNet | 95.2 | 86.0 | 70.2 |
| | LS-DHM [40] | Single | WI, DP | × | FV | 40,960 | IN | VGGNet | – | 83.8 | 67.6 |
| | [228] | Multi | WI, DP | × | SC | 6,096 | IN, PL1 | VGG16 | 95.7 | 87.2 | 71.1 |
| | MR CNN [41] | Multi | WI | √ | pooling | – | Places401 | Inception 2 | – | 86.7 | 72.0 |
| | **SOSF+CFA+GAF** [24] | Single | WI, DP | √ | SFV | 12,288 | IN | VGG16 | – | ***89.5*** | ***78.9*** |
| | **FOSNet** [182] | Single | WI | √ | GAP | 4,096 | PL2 | ResNet | – | ***90.3*** | ***77.3*** |

models and achieves the best accuracy on three benchmark datasets so far.

Extracting global CNN features, which are computed using a pre-trained model, is faster than other deep feature representation techniques, but their quality is not good when there are large differences between the source and target datasets. Comparing these performances [17], [18], [25] demonstrates that the expressive power of global CNN features is improved as richer scene datasets appear. In GAP-CNN [23] and DL-CNN [22], new layers with a small number of parameters substitute for FC layers, but they can still achieve considerable results comparing with benchmark CNNs [17], [18].

Spatially invariant feature based methods are usually time-consuming, especially the computational time of sampling local patches, extracting individually local features, and building codebook. However, these methods are robust against geometrical variance, and thus improve the accuracy of benchmark CNNs, like SFV [30] vs. ImageNet-CNN [17], and MFA-FS [42] vs. PL1-CNN [18]. Encoding technologies generally include more complicated training procedure, so some architectures (*e.g.*, MFAFVNet [31] and VSAD [32]) are designed in a unified pipeline to reduce the operation complexity.

Semantic feature based methods [35], [36], [45], [226] demonstrate very competitive performance, due to the discriminative information laying on the salient regions, compared to global CNN feature based and spatially invariant feature based methods. Salient regions generally are generated by region selection algorithms, which may cause a two-stage training procedure and require more time and computations [63]. In addition, even though the contextual

TABLE 4
Performance (%) comparison of related methods with/without concatenating global CNN feature on benchmark scene datasets.

| | | DSFL [196] | SFV [30] | MFA−FS [42] | MFAFVNet [31] | VSAD [32] | SOSF+CFA [24] | SDO [35] | LGN [45] |
|---|---|---|---|---|---|---|---|---|---|
| MIT67 | Baseline | 52.2 | 72.8 | 81.4 | 82.6 | 84.9 | 84.1 | 68.1 | 85.2 |
| | +Global feature | 76.2 (↑ 24) | 79 (↑ 6.2) | 87.2 (↑ 5.8) | 87.9 (↑ 5.3) | 85.3 (↑ 0.4) | 89.5 (↑ 5.4) | 84 (↑ 15.9) | 85.4 (↑ 0.2) |
| SUN397 | Baseline | – | 54.4 | 63.3 | 64.6 | 71.7 | 66.5 | 54.8 | – |
| | +Global feature | – | 61.7 (↑ 7.3) | 71.1 (↑ 7.8) | 72 (↑ 7.4) | 72.5 (↑ 0.8) | 78.9 (↑ 12.4) | 67 (↑ 12.2) | – |

TABLE 5
Performance (%) comparison of representative methods on benchmark RGB-D scene datasets. For each dataset, the top three scores are highlighted.

| Group | Method | Architecture | | Detailed Information | | | Results | |
|---|---|---|---|---|---|---|---|---|
| | | RGB-CNN | Depth-CNN | Dimension | Modal Fusion | Classifier | NYUD2 | SUN RGBD |
| Dataset | SUN RGBD [90] | PL1-AlexNet | | 8,192 | Feature-level concatenation | SVM | – | 39 |
| Feature learning | SS-CNN [50] | PL1-ASPP | | 4,096 | Image-level concatenation | Softmax | – | 41.3 |
| | MMML [208] | IN-DeCAF | | 256 | Feature-level concatenation | SVM | 61.4 | – |
| | MSMM [48] | PL1-AlexNet | | 12,288+ | Feature-level concatenation | wSVM | 66.7 | 52.3 |
| | **MAPNet** [43] | PL1-AlexNet | | 5,120 | Local and semantic feature concatenation | Softmax | 67.7 | ***56.2*** |
| | SOOR [192] | PL1-AlexNet | PL1-DCNN | 512 | Local and global feature concatenation | SVM | 67.4 | 55.5 |
| | ACM [136] | PL2-AlexNet | | 8192+ | Feature-level concatenation | Softmax | 67.2 | 55.1 |
| | LM-CNN [229] | IN-AlexNet | | 8,192 | Local feature concatenation | Softmax | – | 48.7 |
| Depth feature learning | DCNN [135] | PL1-RCNN | PL1-DCNN | 4,608 | Feature-level concatenation | wSVM | 67.5 | 53.8 |
| | **TRecgNet** [49] | SUN RGBD & PL2 ResNet18 | | 1024 | Feature-level concatenation | Softmax | ***69.2*** | ***56.7*** |
| Multiple modal fusion | DMMF [52] | PL1-AlexNet | | 4096 | Inter- and intra- modal correlation and distinction | L-SVM | – | 41.5 |
| | MaCAFF [47] | PL1-AlexNet | | – | Local and global feature concatenation | L-SVM | 63.9 | 48.1 |
| | DF2Net [199] | PL1-AlexNet | | 512 | Modal correlation and distinction | Softmax | 65.4 | 54.6 |
| | **KFS** [159] | PL1-AlexNet | | 9,216+ | Modal correlation and distinction | Softmax | 67.8 | 55.9 |
| | **CBSC** [230] | PL2-VGG16 | | – | Feature-level concatenation | Softmax | ***69.7**** | ***57.8**** |
| | **MSN** [44] | PL1-AlexNet | | 9,216+ | Modal correlation and distinction | Softmax | ***68.1*** | ***56.2*** |

TABLE 6
Ablation study on benchmark datasets to validate the performance (%) of depth information.

| | | MaCAFF [47] | MSMM [48] | DCNN [135] | DF2Net [199] | TRecgNet [49] | ACM [136] | CBCL [230] | KFS [159] | MSN [44] |
|---|---|---|---|---|---|---|---|---|---|---|
| NYUD2 | Baseline | 53.5 | – | 53.4 | 61.1 | 53.7 | 55.4 | 66.4 | 53.5 | 53.5 |
| | +Depth | 63.9 (↑ 10.4) | – | 67.5 (↑ 14.1) | 65.4 (↑ 4.3) | 67.5 (↑ 13.8) | 67.4 (↑ 12) | 69.7 (↑ 3.3) | 67.8 (↑ 14.3) | 68.1 (↑ 14.6) |
| SUN RGBD | Baseline | 40.4 | 41.5 | 44.3 | 46.3 | 42.6 | 45.7 | 48.8 | 36.1 | – |
| | +Depth | 48.1 (↑ 7.7) | 52.3 (↑ 10.8) | 53.8 (↑ 9.5) | 54.6 (↑ 8.3) | 53.3 (↑ 10.7) | 55.1 (↑ 9.4) | 57.8 (↑ 9.0) | 41.3 (↑ 5.2) | – |

analysis demands more computational power, methods [36], [45], exploring the contextual relations among salient regions, can significantly improve the classification accuracy.

Multi-layer feature based methods employ the complementary features from different layers to improve performance. It is a simple way to use more feature cues, while it also does not require to add any other layers. However, these methods are structurally complicated and have high-dimensional features, which make training models difficult and prone to overfitting [37]. Nevertheless, owing to a novel data augmentation, FTOTLM [21] yields a gain of 19.5% and 19.7% on MIT67 and SUN397, respectively, and has achieved the best results so far.

Multi-view feature based methods take full advantage of features extracted from various CNNs to achieve high classification accuracy. For instance, Table 4 shows that combining global features with other baselines significantly improves their original classification accuracy, *e.g.*, a baseline model "SFV" [30] achieves 72.8% on MIT67, while "SFV+global feature" yields 79%. Moreover, there are two aspects to emphasize: 1) Herranz *et al.* [19] empirically proved that combining too much invalid features is marginally helpful and significantly increases calculation and introduces noise into the final feature. 2) It is essential to improve the expression ability of each view feature, and thus enhance the entire ability of multi-view features.

In summary, the scene classification performance can be boosted by adopting more sophisticated deep models [85], [87] and large-scale datasets [18], [25]. Meanwhile, deep learning based methods can obtain relatively satisfied accuracy on public datasets via combining multiple features [24], focusing on semantic regions [35], augmenting data [21], and training in a unified pipeline [182]. In addition, many methods also improve their accuracy via adopting different strategies, *i.e.*, improved encoding [31], [32], contextual modeling [36], [45], attention policy [23], [74], and multi-task learning [22], [40].

### 5.2 Performance on RGB-D scene datasets

The accuracy of different methods on RGB-D datasets is summarized in Table 5. By adding depth information with different fusion strategies, accuracy results (see Table 6) are improved over 10.8% and 8.8% on average on NYUD2 and

SUN RGBD datasets, respectively. Since depth data provide extra information to train classification model, this observation is within expectation. Noteworthily, it is more difficult to improve the effect on a large dataset (SUN RGBD) than a small dataset (NYUD2). Moreover, the best results on NYUD2 and SUN RGBD datasets achieved by CBSC [230] are as high as 69.7% and 57.84%, respectively.

RGB-D scene data for training are relatively limited, while the dimension of scene features is high. Hence, Support Vector Machines (SVMs) are commonly used in RGB-D scene classification [48], [90], [135], [208] in the early stages. Thanks to data augmentation and back-propagation, Softmax classifier becomes progressively popular, and it is an important reason to yield a comparable performance [43], [44], [49], [230].

Many methods, such as [47], [52], fine-tune RGB-CNNs to extract deep features of depth modality, where the training process is simple, and the computational cost is low. To adapt to depth data, Song *et al.* [135], [231] used weakly-supervised learning to train depth-specific models from scratch, which achieves a gain of 3.5% accuracy, compared to the fine-tuned RGB-CNN. TRecgNet [49], which is based on semi-supervised learning, requires complicated training process and high computational cost, but it obtains comparable results (69.2% on NYUD2 and 56.7% on SUN RGBD).

Feature-level fusion based methods are commonly used due to their high cost-effectiveness, *e.g.,* [43], [136], [230]. Along this way, consistent-feature based, and distinctive-feature based modal fusion use complex fusion layer with high cost, like inference speed and training complexity, but they generally yield more effective features [44], [159], [199].

We can observe that the field of RGB-D scene classification has constantly been improving. Weakly-supervised and semi-supervised learning are useful to learn depth-specific features [48], [49], [135]. Moreover, multi-modal feature fusion is a major issue to improve performance on public datasets [44], [199], [230]. In addition, effective strategies (like contextual strategy [136], [192] and attention mechanism [43], [44]) are also popular for RGB-D scene classification. Nevertheless, the accuracy achieved by current methods is far from expectation and there remains significant rooms for future improvement.

## 6 Conclusion and Outlook

As a contemporary survey for scene classification using deep learning, this paper has highlighted the recent achievements, provided some structural taxonomy for various methods according to their roles in scene representation for scene classification, analyzed their advantages and limitations, summarized existing popular scene datasets, and discussed performance for the most representative approaches. Despite great progress, there are still many unsolved problems. Thus in this section, we will point out these problems and introduce some promising trends for future research. We hope that this survey not only provides a better understanding of scene classification for researchers but also stimulates future research activities.

**Develop more advanced network frameworks.** With the development of deep CNN architectures, from generic CNNs [17], [85], [86], [87] to scene-specific CNNs [22], [23], the accuracy of scene classification is getting increasingly comparable. Nevertheless, there still exists lots of works to be explored on the theoretical research of deep learning [232]. It is a further direction to solidify the theoretical basis so as to get more advanced network frameworks. In particular, it is essential to design specific frameworks for scene classification, such as using automated Neural Architecture Search (NAS) [233], [234], or according to scene attributes.

**Release rich scene datasets.** Deep learning based models require enormous amounts of data to initialize their parameters so that they can learn the scene knowledge well [18], [25]. However, compared to scenes of real world, the publicly available datasets are not large or rich enough, so it is essential to release datasets that encompass richness and high-diversity of environmental scenes [235], especially large-scale RGB-D scene datasets. As opposed to object/texture datasets, scene appearance may be changed dramatically as time goes by, and there emerges new functional scenes as humans develop activity places. Therefore, it requires updating the original scene data and releasing new scene datasets regularly.

**Reduce the dependence of labeled scene images.** The success of deep learning heavily relies on gargantuan amounts of labeled images. However, the labeled training images are always very limited, so supervised learning is not scalable in the absence of fully labeled training data and its generalization ability to classify scenes frequently deteriorates. Therefore, it is desirable to reduce dependence on large amounts of labeled data. To alleviate this difficulty, if with large numbers of unlabel data, it is necessary to further study semi-supervised learning [236], unsupervised learning [237], or self-supervised learning [238]. Even more constrained, without any unlabel training data, the ability to learn from only few labeled images, small-sample learning [239], is also appealing.

**Few shot scene classification.** The success of generic CNNs for scene classification relies heavily on gargantuan amounts of labeled training data [72]. Due to the large intra-variation among scenes, scene datasets cannot cover various classes so that the performance of CNNs frequently deteriorates and fails to generalize well. In contrast, humans can learn a visual concept quickly from very few given examples and often generalize well [240], [241]. Inspired by this, domain adaptation approaches utilize the knowledge of labeled data in task-relevant domains to execute new tasks in target domain [242], [243]. Furthermore, domain generalization methods aim at learning generic representation from multiple task-irrelevant domains to generalize unseen scenarios [244], [245].

**Robust scene classification.** Once scene classification in the laboratory environment is deployed in the real application scenario, there will still be a variety of unacceptable phenomena, that is, the robustness in open environments is a bottleneck to restrict pattern recognition technology. The main reasons why the pattern recognition systems are not robust are basic assumptions, *e.g.,* closed world assumption, independent identical distribution and big data assumption [246], which are main differences between machine learning and human intelligence; hence, it is a fundamental challenge to improve the robustness by breaking these assumptions. It is a nature consider via utilizing adversarial training and optimization [247], [248], [249], which have been applied to pattern recognition [250], [251].

**Realtime scene classification.** Many methods for scene classification, trained in a multiple-stage manner, are computationally expensive for current mobile/wearable devices, which have limited storage and computational capability, therefore researchers have begun to develop convenient and efficient unified networks (encapsulating all computation in a one-stage network) [22], [31], [44]. Moreover, it is also a challenge to keep the model scalable and efficient well when big data from smart wearables and mobile applications is growing rapidly in size temporally or spatially [252].

**Imbalanced scene classification.** The Places365 challenge dataset [25] has more than 8M training images, and the numbers of images in different classes range from 4,000 to 30,000 per class. It shows that scene categories are imbalanced, *i.e.*, some categories are abundant while others have scarce examples. Generally, the minority class samples are poorly predicted by the learned model [253], [254]. Therefore, learning a model which respects both type of categories and equally performs well on frequent and infrequent ones remains a great challenge and needs further exploration [254], [255], [256].

**Continuous scene classification.** The ultimate goal is to develop methods capable of accurately and efficiently classifying samples in thousands or more unseen scene categories in open environments [72]. The classic deep learning paradigm learns in isolation, *i.e.*, it needs many training examples and is only suitable for well-defined and narrow tasks in closed environments [142], [257]. In contrast, "human learning" is a continuous learning and adaptation to new environments: humans accumulate the knowledge gained in the past and use this knowledge to help future learning and problem solving with possible adaptations [258]. Ideally, it should also be capable to discover unknown scenarios and learn new works in a self-supervised manner. Inspired by this, it is necessary to do lifelong machine learning via developing versatile systems that continually accumulate and refine their knowledge over time [259], [260]. Such lifelong machine learning has represented a long-standing challenge for deep learning and, consequently, artificial intelligence systems.

**Multi-label scene classification.** Many scenes are semantic multiplicity [73], [75], *i.e.*, a scene image may belong to multiple semantic classes. Such a problem poses a challenge to the classic pattern recognition paradigm and requires developing multi-label learning methods [75], [261]. Moreover, when constructing scene datasets, most researchers either avoid labeling multi-label images or use the most obvious class (single label) to annotate subjectively each image [73]. Hence, it is hard to improve the generalization ability of the model trained on single-label datasets, which also brings problems to classification task.

**Other-modal scene classification.** RGB images provide key features such as color, texture, and spectrum of objects. Nevertheless, the scenes reproduced by RGB images may have uneven lighting, target occlusion, *etc*. Therefore, the robustness of RGB scene classification is poor, and it is difficult to accurately extract key information such as target contours and spatial positions. In contrast, the rapid development of sensors has made the acquisition of other modal data easier and easier, such as RGB-D [90], video [262], 3D point clouds [263], [264]. Recently, research on recognizing and understanding various modalities has attracted an increasing attention [135], [265], [266].

## Acknowledgments

The authors would like to thank the pioneer researchers in scene classification and other related fields. This work was supported in part by grants from National Science Foundation of China (61872379, 91846301, 61571005), the Academy of Finland (331883), the fundamental research program of Guangdong, China (2020B1515310023), the Hunan Science and Technology Plan Project (2019GK2131), the China Scholarship Council (201806155037), the Science and Technology Research Program of Guangzhou, China (201804010429), the National Key Research and Development Program of China (2016YFB1200402020).

## References

[1] J. M. Henderson and A. Hollingworth, "High-level scene perception," *Annual review of psychology*, vol. 50, no. 1, pp. 243–271, 1999.
[2] R. Epstein, "The cortical basis of visual scene processing," *Visual Cognition*, vol. 12, no. 6, pp. 954–978, 2005.
[3] M. R. Greene and A. Oliva, "The briefest of glances: The time course of natural scene understanding," *Psychological Science*, vol. 20, no. 4, pp. 464–472, 2009.
[4] D. B. Walther, B. Chai, E. Caddigan, D. M. Beck, and L. Fei-Fei, "Simple line drawings suffice for functional MRI decoding of natural scene categories," *PNAS*, vol. 108, no. 23, pp. 9661–9666, 2011.
[5] J. Vogel and B. Schiele, "Semantic modeling of natural scenes for content-based image retrieval," *IJCV*, vol. 72, no. 2, pp. 133–157, 2007.
[6] L. Zheng, Y. Yang, and Q. Tian, "SIFT meets CNN: A decade survey of instance retrieval," *IEEE TPAMI*, vol. 40, no. 5, pp. 1224–1244, 2017.
[7] W. Zhang, X. Yu, and X. He, "Learning bidirectional temporal cues for video-based person re-identification," *IEEE TCSVT*, vol. 28, no. 10, pp. 2768–2776, 2017.
[8] J. Hou, H. Zeng, J. Zhu, J. Hou, J. Chen, and K.-K. Ma, "Deep quadruplet appearance learning for vehicle re-identification," *IEEE TVT*, vol. 68, no. 9, pp. 8512–8522, 2019.
[9] T. Zhang, S. Liu, C. Xu, and H. Lu, "Mining semantic context information for intelligent video surveillance of traffic scenes," *IEEE TII*, vol. 9, no. 1, pp. 149–160, 2012.
[10] G. Sreenu and M. S. Durai, "Intelligent video surveillance: A review through deep learning techniques for crowd analysis," *Journal of Big Data*, vol. 6, no. 1, p. 48, 2019.
[11] A. H. Behzadan and V. R. Kamat, "Integrated information modeling and visual simulation of engineering operations using dynamic augmented reality scene graphs," *ITcon*, vol. 16, no. 17, pp. 259–278, 2011.
[12] A. Y. Nee, S. Ong, G. Chryssolouris, and D. Mourtzis, "Augmented reality applications in design and manufacturing," *CIRP annals*, vol. 61, no. 2, pp. 657–679, 2012.
[13] K. Muhammad, J. Ahmad, and S. W. Baik, "Early fire detection using convolutional neural networks during surveillance for effective disaster management," *Neurocomputing*, vol. 288, pp. 30–42, 2018.
[14] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *CVPR*, vol. 2, 2006, pp. 2169–2178, https://figshare.com/articles/15-Scene_Image_Dataset/7007177.
[15] 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 *NeurIPS*, 2010, pp. 1378–1386.
[16] R. Margolin, L. Zelnik-Manor, and A. Tal, "OTC: A novel local descriptor for scene classification," in *ECCV*, 2014, pp. 377–391.
[17] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *NeurIPS*, 2012, pp. 1097–1105.
[18] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva, "Learning deep features for scene recognition using places database," in *NeurIPS*, 2014, pp. 487–495, http://places.csail.mit.edu/downloadData.html.
[19] L. Herranz, S. Jiang, and X. Li, "Scene recognition with CNNs: Objects, scales and dataset bias," in *CVPR*, 2016, pp. 571–579.
[20] S. H. Khan, M. Hayat, M. Bennamoun, R. Togneri, and F. A. Sohel, "A discriminative representation of convolutional features for indoor scene recognition," *IEEE TIP*, vol. 25, no. 7, pp. 3372–3383, 2016.
[21] S. Liu, G. Tian, and Y. Xu, "A novel scene classification model combining ResNet based transfer learning and data augmentation with a filter," *Neurocomputing*, vol. 338, pp. 191–206, 2019.
[22] Y. Liu, Q. Chen, W. Chen, and I. Wassell, "Dictionary learning inspired deep network for scene recognition," in *AAAI*, 2018.
[23] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *CVPR*, 2016, pp. 2921–2929.
[24] N. Sun, W. Li, J. Liu, G. Han, and C. Wu, "Fusing object semantics and deep appearance features for scene recognition," *IEEE TCSVT*, vol. 29, no. 6, pp. 1715–1728, 2018.
[25] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba, "Places: A 10 million image database for scene recognition," *IEEE TPAMI*, vol. 40, no. 6, pp. 1452–1464, 2017, http://places2.csail.mit.edu/download.html.
[26] M. Hayat, S. H. Khan, M. Bennamoun, and S. An, "A spatial layout and scale invariant feature representation for indoor scene classification," *IEEE TIP*, vol. 25, no. 10, pp. 4829–4841, 2016.
[27] Z. Zuo, B. Shuai, G. Wang, X. Liu, X. Wang, B. Wang, and Y. Chen, "Learning contextual dependence with convolutional hierarchical

recurrent neural networks," *IEEE TIP*, vol. 25, no. 7, pp. 2983–2996, 2016.
[28] Y. Gong, L. Wang, R. Guo, and S. Lazebnik, "Multi-scale orderless pooling of deep convolutional activation features," in *ECCV*, 2014, pp. 392–407.
[29] D. Yoo, S. Park, J.-Y. Lee, and I. So Kweon, "Multi-scale pyramid pooling for deep convolutional representation," in *CVPRW*, 2015, pp. 71–80.
[30] M. Dixit, S. Chen, D. Gao, N. Rasiwasia, and N. Vasconcelos, "Scene classification with semantic fisher vectors," in *CVPR*, 2015, pp. 2974–2983.
[31] Y. Li, M. Dixit, and N. Vasconcelos, "Deep scene image classification with the MFAFVNet," in *ICCV*, 2017, pp. 5746–5754.
[32] Z. Wang, L. Wang, Y. Wang, B. Zhang, and Y. Qiao, "Weakly supervised patchnets: Describing and aggregating local patches for scene recognition," *IEEE TIP*, vol. 26, no. 4, pp. 2028–2041, 2017.
[33] R. Wu, B. Wang, W. Wang, and Y. Yu, "Harvesting discriminative meta objects with deep CNN features for scene classification," in *ICCV*, 2015, pp. 1287–1295.
[34] T. Durand, N. Thome, and M. Cord, "WELDON: Weakly supervised learning of deep convolutional neural networks," in *CVPR*, 2016, pp. 4743–4752.
[35] X. Cheng, J. Lu, J. Feng, B. Yuan, and J. Zhou, "Scene recognition with objectness," *Pattern Recognition*, vol. 74, pp. 474–487, 2018.
[36] C. Laranjeira, A. Lacerda, and E. R. Nascimento, "On modeling context from objects with a long short-term memory for indoor scene recognition," in *SIBGRAPI*, 2019, pp. 249–256.
[37] S. Yang and D. Ramanan, "Multi-scale recognition with DAG-CNNs," in *ICCV*, 2015, pp. 1215–1223.
[38] G.-S. Xie, X.-Y. Zhang, S. Yan, and C.-L. Liu, "Hybrid CNN and dictionary-based models for scene recognition and domain adaptation," *IEEE TCSVT*, vol. 27, no. 6, pp. 1263–1274, 2015.
[39] 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, 2017.
[40] S. Guo, W. Huang, L. Wang, and Y. Qiao, "Locally supervised deep hybrid model for scene recognition," *IEEE TIP*, vol. 26, no. 2, pp. 808–820, 2016.
[41] L. Wang, S. Guo, W. Huang, Y. Xiong, and Y. Qiao, "Knowledge guided disambiguation for large-scale scene classification with multi-resolution CNNs," *IEEE TIP*, vol. 26, no. 4, pp. 2055–2068, 2017.
[42] M. D. Dixit and N. Vasconcelos, "Object based scene representations using fisher scores of local subspace projections," in *NeurIPS*, 2016, pp. 2811–2819.
[43] Y. Li, Z. Zhang, Y. Cheng, L. Wang, and T. Tan, "MAPNet: Multi-modal attentive pooling network for RGB-D indoor scene classification," *Pattern Recognition*, vol. 90, pp. 436–449, 2019.
[44] Z. Xiong, Y. Yuan, and Q. Wang, "MSN: Modality separation networks for RGB-D scene recognition," *Neurocomputing*, vol. 373, pp. 81–89, 2020.
[45] G. Chen, X. Song, H. Zeng, and S. Jiang, "Scene recognition with prototype-agnostic scene layout," *IEEE TIP*, vol. 29, pp. 5877–5888, 2020.
[46] P. Wei, F. Qin, F. Wan, Y. Zhu, J. Jiao, and Q. Ye, "Correlated topic vector for scene classification," *IEEE TIP*, vol. 26, no. 7, pp. 3221–3234, 2017.
[47] A. Wang, J. Cai, J. Lu, and T.-J. Cham, "Modality and component aware feature fusion for RGB-D scene classification," in *CVPR*, 2016, pp. 5995–6004.
[48] X. Song, S. Jiang, and L. Herranz, "Combining models from multiple sources for RGB-D scene recognition." in *IJCAI*, 2017, pp. 4523–4529.
[49] D. Du, L. Wang, H. Wang, K. Zhao, and G. Wu, "Translate-to-recognize networks for RGB-D scene recognition," in *CVPR*, 2019, pp. 11 836–11 845.
[50] Y. Liao, S. Kodagoda, Y. Wang, L. Shi, and Y. Liu, "Understand scene categories by objects: A semantic regularized scene classifier using convolutional neural networks," in *CRA*, 2016, pp. 2318–2325.
[51] Y. Cheng, R. Cai, Z. Li, X. Zhao, and K. Huang, "Locality-sensitive deconvolution networks with gated fusion for RGB-D indoor semantic segmentation," in *CVPR*, 2017, pp. 3029–3037.
[52] H. Zhu, J.-B. Weibel, and S. Lu, "Discriminative multi-modal feature fusion for RGB-D indoor scene recognition," in *CVPR*, 2016, pp. 2969–2976.
[53] R. Girshick, "Fast R-CNN," in *ICCV*, 2015, pp. 1440–1448.
[54] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE TPAMI*, vol. 39, no. 6, pp. 1137–1149, 2016.
[55] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE TPAMI*, vol. 39, no. 12, pp. 2481–2495, 2017.
[56] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *IEEE TPAMI*, vol. 40, no. 4, pp. 834–848, 2017.
[57] S. Cai, J. Huang, D. Zeng, X. Ding, and J. Paisley, "MEnet: A metric expression network for salient object segmentation," in *IJCAI*, 2018, pp. 598–605.
[58] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *CVPR*, 2009, pp. 248–255, http://image-net.org/download.
[59] J. Xiao, J. Hays, K. A. Ehinger, A. Oliva, and A. Torralba, "Sun database: Large-scale scene recognition from abbey to zoo," in *CVPR*, 2010, pp. 3485–3492, http://places2.csail.mit.edu/download.html.
[60] R. G. Cinbis, J. Verbeek, and C. Schmid, "Approximate fisher kernels of non-iid image models for image categorization," *IEEE TPAMI*, vol. 38, no. 6, pp. 1084–1098, 2015.
[61] X. Wei, S. L. Phung, and A. Bouzerdoum, "Visual descriptors for scene categorization: Experimental evaluation," *AI Review*, vol. 45, no. 3, pp. 333–368, 2016.
[62] G. Cheng, J. Han, and X. Lu, "Remote sensing image scene classification: Benchmark and state of the art," *Proceedings of the IEEE*, vol. 105, no. 10, pp. 1865–1883, 2017.
[63] L. Xie, F. Lee, L. Liu, K. Kotani, and Q. Chen, "Scene recognition: A comprehensive survey," *Pattern Recognition*, vol. 102, p. 107205, 2020.
[64] K. Nogueira, O. A. Penatti, and J. A. Dos Santos, "Towards better exploiting convolutional neural networks for remote sensing scene classification," *Pattern Recognition*, vol. 61, pp. 539–556, 2017.
[65] F. Hu, G.-S. Xia, J. Hu, and L. Zhang, "Transferring deep convolutional neural networks for the scene classification of high-resolution remote sensing imagery," *Remote Sensing*, vol. 7, no. 11, pp. 14 680–14 707, 2015.
[66] A. Mesaros, T. Heittola, and T. Virtanen, "Tut database for acoustic scene classification and sound event detection," in *EUSIPCO*, 2016, pp. 1128–1132.
[67] Z. Ren, K. Qian, Y. Wang, Z. Zhang, V. Pandit, A. Baird, and B. Schuller, "Deep scalogram representations for acoustic scene classification," *JAS*, vol. 5, no. 3, pp. 662–669, 2018.
[68] S. Lowry, N. Sünderhauf, P. Newman, J. J. Leonard, D. Cox, P. Corke, and M. J. Milford, "Visual place recognition: A survey," *IEEE T-RO*, vol. 32, no. 1, pp. 1–19, 2015.
[69] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "NetVLAD: CNN architecture for weakly supervised place recognition," in *CVPR*, 2016, pp. 5297–5307.
[70] L. W. Renninger and J. Malik, "When is scene identification just texture recognition?" *Vision research*, vol. 44, no. 19, pp. 2301–2311, 2004.
[71] L. Liu, J. Chen, P. Fieguth, G. Zhao, R. Chellappa, and M. Pietikäinen, "From BoW to CNN: Two decades of texture representation for texture classification," *IJCV*, vol. 127, no. 1, pp. 74–109, 2019.
[72] L. Liu, W. Ouyang, X. Wang, P. Fieguth, J. Chen, X. Liu, and M. Pietikäinen, "Deep learning for generic object detection: A survey," *IJCV*, vol. 128, no. 2, pp. 261–318, 2020.
[73] M. R. Boutell, J. Luo, X. Shen, and C. M. Brown, "Learning multi-label scene classification," *Pattern Recognition*, vol. 37, no. 9, pp. 1757–1771, 2004.
[74] A. López-Cifuentes, M. Escudero-Viñolo, J. Bescós, and Á. García-Martín, "Semantic-aware scene recognition," *Pattern Recognition*, vol. 102, p. 107256, 2020.
[75] M.-L. Zhang and Z.-H. Zhou, "Multi-label learning by instance differentiation," in *AAAI*, vol. 7, 2007, pp. 669–674.
[76] D. G. Lowe, "Object recognition from local scale-invariant features," in *IJCV*, vol. 2, 1999, pp. 1150–1157.
[77] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *IJCV*, vol. 42, no. 3, pp. 145–175, 2001.
[78] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *CVPR*, vol. 1, 2005, pp. 886–893.
[79] J. Wu and J. M. Rehg, "CENTRIST: A visual descriptor for scene categorization," *IEEE TPAMI*, vol. 33, no. 8, pp. 1489–1501, 2010.
[80] J. Sivic and A. Zisserman, "Video google: A text retrieval approach to object matching in videos," in *ICCV*, vol. 2, 2003, p. 14701477.
[81] J. Sánchez, F. Perronnin, T. Mensink, and J. Verbeek, "Image classification with the Fisher vector: Theory and practice," *IJCV*, vol. 105, no. 3, pp. 222–245, 2013.
[82] H. Jégou, M. Douze, C. Schmid, and P. Pérez, "Aggregating local descriptors into a compact image representation," in *CVPR*, 2010, pp. 3304–3311.
[83] L.-J. Li, H. Su, Y. Lim, and L. Fei-Fei, "Objects as attributes for scene classification," in *ECCV*, 2010, pp. 57–69.

[84] M. Juneja, A. Vedaldi, C. Jawahar, and A. Zisserman, "Blocks that shout: Distinctive parts for scene classification," in *CVPR*, 2013, pp. 923–930.

[85] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *ICLR*, 2015.

[86] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *CVPR*, 2015, pp. 1–9.

[87] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016, pp. 770–778.

[88] A. Sharif Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, "CNN features off-the-shelf: An astounding baseline for recognition," in *CVPRW*, 2014, pp. 806–813.

[89] M. Cimpoi, S. Maji, and A. Vedaldi, "Deep filter banks for texture recognition and segmentation," in *CVPR*, 2015, pp. 3828–3836.

[90] S. Song, S. P. Lichtenberg, and J. Xiao, "SUN RGB-D: A RGB-D scene understanding benchmark suite," in *CVPR*, 2015, pp. 567–576, https://github.com/ankurhanda/sunrgbd-meta-data.

[91] N. Vasconcelos and A. Lippman, "A probabilistic architecture for content-based image retrieval," in *CVPR*, 2000, pp. 216–221.

[92] G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual categorization with bags of keypoints," in *ECCVW*, 2004, pp. 1–2.

[93] C. Wallraven, B. Caputo, and A. Graf, "Recognition with local features: The kernel recipe," in *ICCV*, 2003, pp. 257–264.

[94] J. C. Van Gemert, J.-M. Geusebroek, C. J. Veenman, and A. W. Smeulders, "Kernel codebooks for scene categorization," in *ECCV*, 2008, pp. 696–709.

[95] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *IJCV*, vol. 60, no. 2, pp. 91–110, 2004.

[96] R. Kwitt, N. Vasconcelos, and N. Rasiwasia, "Scene recognition on the semantic manifold," in *ECCV*, 2012, pp. 359–372.

[97] N. Rasiwasia and N. Vasconcelos, "Holistic context models for visual recognition," *IEEE TPAMI*, vol. 34, no. 5, pp. 902–917, 2012.

[98] A. Oliva and A. Torralba, "Building the gist of a scene: The role of global image features in recognition," *Progress in brain research*, vol. 155, pp. 23–36, 2006.

[99] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE TPAMI*, vol. 24, no. 7, pp. 971–987, 2002.

[100] P. Felzenszwalb, D. McAllester, and D. Ramanan, "A discriminatively trained, multiscale, deformable part model," in *CVPR*, 2008, pp. 1–8.

[101] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE TPAMI*, vol. 32, no. 9, pp. 1627–1645, 2009.

[102] M. Pandey and S. Lazebnik, "Scene recognition and weakly supervised object localization with deformable part-based models," in *ICCV*, 2011, pp. 1307–1314.

[103] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *JMLR*, vol. 3, no. 1, pp. 993–1022, 2003.

[104] K. Grauman and T. Darrell, "The pyramid match kernel: Discriminative classification with sets of image features," in *ICCV*, vol. 2, 2005, pp. 1458–1465.

[105] F. Perronnin and C. Dance, "Fisher kernels on visual vocabularies for image categorization," in *CVPR*, 2007, pp. 1–8.

[106] L. Xie, J. Wang, B. Guo, B. Zhang, and Q. Tian, "Orientational pyramid matching for recognizing indoor scenes," in *CVPR*, 2014, pp. 3734–3741.

[107] J. Yang, K. Yu, Y. Gong, and T. Huang, "Linear spatial pyramid matching using sparse coding for image classification," in *CVPR*, 2009, pp. 1794–1801.

[108] S. Gao, I. W.-H. Tsang, L.-T. Chia, and P. Zhao, "Local features are not lonely–laplacian sparse coding for image classification," in *CVPR*, 2010, pp. 3555–3561.

[109] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong, "Locality-constrained linear coding for image classification," in *CVPR*, 2010, pp. 3360–3367.

[110] J. Wu and J. M. Rehg, "Beyond the euclidean distance: Creating effective visual codebooks using the histogram intersection kernel," in *ICCV*, 2009, pp. 630–637.

[111] J. Qin and N. H. Yung, "Scene categorization via contextual visual words," *Pattern Recognition*, vol. 43, no. 5, pp. 1874–1888, 2010.

[112] L. Bo and C. Sminchisescu, "Efficient match kernel between sets of features for visual recognition," in *NeurIPS*, 2009, pp. 135–143.

[113] P. Wang, J. Wang, G. Zeng, W. Xu, H. Zha, and S. Li, "Supervised kernel descriptors for visual recognition," in *CVPR*, 2013, pp. 2858–2865.

[114] L.-J. Li, H. Su, Y. Lim, and L. Fei-Fei, "Object bank: An object-level image representation for high-level visual recognition," *IJCV*, vol. 107, no. 1, pp. 20–39, 2014.

[115] L. Zhang, X. Zhen, and L. Shao, "Learning object-to-class kernels for scene classification," *IEEE TIP*, vol. 23, no. 8, pp. 3241–3253, 2014.

[116] F. Sadeghi and M. F. Tappen, "Latent pyramidal regions for recognizing scenes," in *ECCV*, 2012, pp. 228–241.

[117] J. Yu, D. Tao, Y. Rui, and J. Cheng, "Pairwise constraints based multiview features fusion for scene classification," *Pattern Recognition*, vol. 46, no. 2, pp. 483–496, 2013.

[118] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *CVPR*, 2014, pp. 580–587.

[119] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "Deepface: Closing the gap to human-level performance in face verification," in *CVPR*, 2014, pp. 1701–1708.

[120] Y. Guo, H. Wang, Q. Hu, H. Liu, L. Liu, and M. Bennamoun, "Deep learning for 3d point clouds: A survey," *IEEE TPAMI*, 2020.

[121] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed *et al.*, "Deep neural networks for acoustic modeling in speech recognition," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, 2012.

[122] C. Chen, A. Seff, A. Kornhauser, and J. Xiao, "Deepdriving: Learning affordance for direct perception in autonomous driving," in *ICCV*, 2015, pp. 2722–2730.

[123] A. Esteva, B. Kuprel, R. A. Novoa, J. Ko, S. M. Swetter, H. M. Blau, and S. Thrun, "Dermatologist level classification of skin cancer with deep neural networks," *Nature*, vol. 542, no. 7639, pp. 115–118, 2017.

[124] S. M. McKinney, M. Sieniek, V. Godbole, J. Godwin, N. Antropova *et al.*, "International evaluation of an AI system for breast cancer screening," *Nature*, vol. 577, no. 7788, pp. 89–94, 2020.

[125] Y. Wu, M. Schuster, Z. Chen, Q. V. Le, M. Norouzi *et al.*, "Google's neural machine translation system: Bridging the gap between human and machine translation," *arXiv:1609.08144*, 2016.

[126] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre *et al.*, "Mastering the game of go with deep neural networks and tree search," *Nature*, vol. 529, no. 7587, pp. 484–489, 2016.

[127] D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang *et al.*, "Mastering the game of go without human knowledge," *Nature*, vol. 550, no. 7676, pp. 354–359, 2017.

[128] D. Silver, T. Hubert, J. Schrittwieser, I. Antonoglou, M. Lai *et al.*, "A general reinforcement learning algorithm that masters chess, shogi, and go through self-play," *Science*, vol. 362, no. 6419, pp. 1140–1144, 2018.

[129] O. Vinyals, I. Babuschkin, W. M. Czarnecki, M. Mathieu, A. Dudzik *et al.*, "Grandmaster level in starcraft ii using multi-agent reinforcement learning," *Nature*, vol. 575, no. 7782, pp. 350–354, 2019.

[130] P. M. DeVries, F. Viégas, M. Wattenberg, and B. J. Meade, "Deep learning of aftershock patterns following large earthquakes," *Nature*, vol. 560, no. 7720, pp. 632–634, 2018.

[131] https://www.technologyreview.com/lists/technologies/2020/.

[132] J. M. Stokes, K. Yang, K. Swanson, W. Jin, A. Cubillos-Ruiz, N. M. Donghia, C. R. MacNair, S. French, L. A. Carfrae, Z. Bloom-Ackerman *et al.*, "A deep learning approach to antibiotic discovery," *Cell*, vol. 180, no. 4, pp. 688–702, 2020.

[133] L. Liu, C. Shen, L. Wang, A. Van Den Hengel, and C. Wang, "Encoding high dimensional local features by sparse coding based fisher vectors," in *NeurIPS*, 2014, pp. 1143–1151.

[134] B.-B. Gao, X.-S. Wei, J. Wu, and W. Lin, "Deep spatial pyramid: The devil is once again in the details," *arXiv:1504.05277*, 2015.

[135] X. Song, S. Jiang, L. Herranz, and C. Chen, "Learning effective RGB-D representations for scene recognition," *IEEE TIP*, vol. 28, no. 2, pp. 980–993, 2018.

[136] Y. Yuan, Z. Xiong, and Q. Wang, "Acm: Adaptive cross-modal graph convolutional neural networks for RGB-D scene recognition," in *AAAI*, vol. 33, 2019, pp. 9176–9184.

[137] A. Quattoni and A. Torralba, "Recognizing indoor scenes," in *CVPR*, 2009, pp. 413–420, http://web.mit.edu/torralba/www/indoor.html.

[138] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from RGB-D images," in *ECCV*, 2012, pp. 746–760, https://cs.nyu.edu/~silberman/datasets/nyu_depth_v2.html.

[139] S. Gupta, R. Girshick, P. Arbeláez, and J. Malik, "Learning rich features from RGB-D images for object detection and segmentation," in *ECCV*, 2014, pp. 345–360.

[140] G. A. Miller, "Wordnet: A lexical database for english," *Communications of the ACM*, vol. 38, no. 11, pp. 39–41, 1995.

[141] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *IJCV*, vol. 88, no. 2, pp. 303–338, 2010.

[142] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[143] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" in *NeurIPS*, 2014, pp. 3320–3328.

[144] Z. Bolei, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Object detectors emerge in deep scene CNNs," *ICLR*, 2015.

[145] I. Sutskever, J. Martens, G. Dahl, and G. Hinton, "On the importance of initialization and momentum in deep learning," in *ICML*, 2013, pp. 1139–1147.

[146] B. Liu, J. Liu, J. Wang, and H. Lu, "Learning a representative and discriminative part model with deep convolutional features for scene recognition," in *ACCV*, 2014, pp. 643–658.

[147] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, "Return of the devil in the details: Delving deep into convolutional nets," *arXiv:1405.3531*, 2014.

[148] M. Lin, Q. Chen, and S. Yan, "Network in network," *ICLR*, 2014.

[149] J. Sun and J. Ponce, "Learning discriminative part detectors for image classification and cosegmentation," in *ICCV*, 2013, pp. 3400–3407.

[150] K. J. Shih, I. Endres, and D. Hoiem, "Learning discriminative collections of part detectors for object recognition," *IEEE TPAMI*, vol. 37, no. 8, pp. 1571–1584, 2014.

[151] Y. Sun, X. Wang, and X. Tang, "Deeply learned face representations are sparse, selective, and robust," in *CVPR*, 2015, pp. 2892–2900.

[152] D. L. Donoho, A. Maleki, and A. Montanari, "Message-passing algorithms for compressed sensing," *PNAS*, vol. 106, no. 45, pp. 18 914–18 919, 2009.

[153] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[154] P. Mirowski, R. Pascanu, F. Viola, H. Soyer, A. J. Ballard, A. Banino, M. Denil, R. Goroshin, L. Sifre, K. Kavukcuoglu *et al.*, "Learning to navigate in complex environments," *ICLR*, 2016.

[155] T. J. Sejnowski, "The unreasonable effectiveness of deep learning in artificial intelligence," *PNAS*, 2020.

[156] L. Xie, L. Zheng, J. Wang, A. L. Yuille, and Q. Tian, "Interactive: Inter-layer activeness propagation," in *CVPR*, 2016, pp. 270–279.

[157] L. Xie, J. Wang, W. Lin, B. Zhang, and Q. Tian, "Towards reversal-invariant image representation," *IJCV*, vol. 123, no. 2, pp. 226–250, 2017.

[158] M. Rezanejad, G. Downs, J. Wilder, D. B. Walther, A. Jepson, S. Dickinson, and K. Siddiqi, "Scene categorization from contours: Medial axis based salience measures," in *CVPR*, 2019, pp. 4116–4124.

[159] Z. Xiong, Y. Yuan, and Q. Wang, "RGB-D scene recognition via spatial-related multi-modal feature learning," *IEEE Access*, vol. 7, pp. 106 739–106 747, 2019.

[160] J. R. Uijlings, K. E. Van De Sande, T. Gevers, and A. W. Smeulders, "Selective search for object recognition," *IJCV*, vol. 104, no. 2, pp. 154–171, 2013.

[161] S. Singh, A. Gupta, and A. A. Efros, "Unsupervised discovery of mid-level discriminative patches," in *ECCV*, 2012, pp. 73–86.

[162] P. Arbeláez, J. Pont-Tuset, J. T. Barron, F. Marques, and J. Malik, "Multiscale combinatorial grouping," in *CVPR*, 2014, pp. 328–335.

[163] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shot multibox detector," in *ECCV*, 2016, pp. 21–37.

[164] J. Redmon and A. Farhadi, "Yolo9000: Better, faster, stronger," in *CVPR*, 2017, pp. 7263–7271.

[165] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *CVPR*, 2016, pp. 779–788.

[166] C.-Y. Lee, S. Xie, P. Gallagher, Z. Zhang, and Z. Tu, "Deeply-supervised nets," in *AISTATS*, 2015, pp. 562–570.

[167] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *CVPR*, 2017, pp. 2117–2125.

[168] C. G. Snoek, M. Worring, and A. W. Smeulders, "Early versus late fusion in semantic video analysis," in *Proceedings of the 13th annual ACM international conference on Multimedia*, 2005, pp. 399–402.

[169] H. Gunes and M. Piccardi, "Affect recognition from face and body: Early fusion vs. late fusion," in *International conference on systems, man and cybernetics*, vol. 4, 2005, pp. 3437–3443.

[170] Y. Dong, S. Gao, K. Tao, J. Liu, and H. Wang, "Performance evaluation of early and late fusion methods for generic semantics indexing," *Pattern Analysis and Applications*, vol. 17, no. 1, pp. 37–50, 2014.

[171] J. Li, D. Lin, Y. Wang, G. Xu, Y. Zhang, C. Ding, and Y. Zhou, "Deep discriminative representation learning with attention map for scene classification," *Remote Sensing*, vol. 12, no. 9, p. 1366, 2020.

[172] F. Zhang, B. Du, and L. Zhang, "Scene classification via a gradient boosting random convolutional network framework," *IEEE TGRS*, vol. 54, no. 3, pp. 1793–1802, 2015.

[173] L. Wang, Z. Wang, W. Du, and Y. Qiao, "Object-scene convolutional neural networks for event recognition in images," in *CVPRW*, 2015, pp. 30–35.

[174] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *ICML*, 2015.

[175] H. Jin Kim and J.-M. Frahm, "Hierarchy of alternating specialists for scene recognition," in *ECCV*, 2018, pp. 451–467.

[176] Z. Ghahramani, G. E. Hinton *et al.*, "The em algorithm for mixtures of factor analyzers," University of Toronto, Tech. Rep., 1996.

[177] J. Verbeek, "Learning nonlinear image manifolds by global alignment of local linear models," *IEEE TPAMI*, vol. 28, no. 8, pp. 1236–1250, 2006.

[178] S. Xia, J. Zeng, L. Leng, and X. Fu, "WS-AM: Weakly supervised attention map for scene recognition," *Electronics*, vol. 8, no. 10, p. 1072, 2019.

[179] A. Kolesnikov and C. H. Lampert, "Seed, expand and constrain: Three principles for weakly-supervised image segmentation," in *ECCV*, 2016, pp. 695–711.

[180] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual explanations from deep networks via gradient-based localization," in *IEEE ICCV*, 2017, pp. 618–626.

[181] J. Zhang, S. A. Bargal, Z. Lin, J. Brandt, X. Shen, and S. Sclaroff, "Top-down neural attention by excitation backprop," *IJCV*, vol. 126, no. 10, pp. 1084–1102, 2018.

[182] H. Seong, J. Hyun, and E. Kim, "Fosnet: An end-to-end trainable deep neural network for scene recognition," *IEEE Access*, vol. 8, pp. 82 066–82 077, 2020.

[183] Z. Niu, G. Hua, X. Gao, and Q. Tian, "Context aware topic model for scene recognition," in *CVPR*, 2012, pp. 2743–2750.

[184] Z. Zuo, B. Shuai, G. Wang, X. Liu, X. Wang, B. Wang, and Y. Chen, "Convolutional recurrent neural networks: Learning spatial dependencies for image representation," in *CVPRW*, 2015, pp. 18–26.

[185] X. Wang and E. Grimson, "Spatial latent dirichlet allocation," in *NeurIPS*, 2008, pp. 1577–1584.

[186] J. L. Elman, "Finding structure in time," *Cognitive science*, vol. 14, no. 2, pp. 179–211, 1990.

[187] G. R. Cross and A. K. Jain, "Markov random field texture models," *IEEE TPAMI*, no. 1, pp. 25–39, 1983.

[188] S. Z. Li, *Markov random field modeling in image analysis*. Springer, 2009.

[189] J. Bruna, W. Zaremba, A. Szlam, and Y. LeCun, "Spectral networks and locally connected networks on graphs," *arXiv:1312.6203*, 2013.

[190] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *ICLR*, 2016.

[191] J. Zhou, G. Cui, Z. Zhang, C. Yang, Z. Liu, L. Wang, C. Li, and M. Sun, "Graph neural networks: A review of methods and applications," *arXiv:1812.08434*, 2018.

[192] X. Song, S. Jiang, B. Wang, C. Chen, and G. Chen, "Image representations with spatial object-to-object relations for RGB-D scene recognition," *IEEE TIP*, vol. 29, pp. 525–537, 2019.

[193] S. A. Javed and A. K. Nelakanti, "Object-level context modeling for scene classification with context-CNN," *arXiv:1705.04358*, 2017.

[194] X. Song, S. Jiang, and L. Herranz, "Multi-scale multi-feature context modeling for scene recognition in the semantic manifold," *IEEE TIP*, vol. 26, no. 6, pp. 2721–2735, 2017.

[195] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *ICLR*, 2017.

[196] Z. Zuo, G. Wang, B. Shuai, L. Zhao, Q. Yang, and X. Jiang, "Learning discriminative and shareable features for scene classification," in *ECCV*, 2014, pp. 552–568.

[197] S. Jiang, G. Chen, X. Song, and L. Liu, "Deep patch representations with shared codebook for scene classification," *ACM TOMM*, vol. 15, no. 1s, pp. 1–17, 2019.

[198] B. E. Boser, I. M. Guyon, and V. N. Vapnik, "A training algorithm for optimal margin classifiers," in *Annual workshop on Computational learning theory*, 1992, pp. 144–152.

[199] Y. Li, J. Zhang, Y. Cheng, K. Huang, and T. Tan, "Df2net: Discriminative feature learning and fusion network for RGB-D indoor scene classification," in *AAAI*, 2018.

[200] J. C. Van Gemert, C. J. Veenman, A. W. Smeulders, and J.-M. Geusebroek, "Visual word ambiguity," *IEEE TPAMI*, vol. 32, no. 7, pp. 1271–1283, 2009.

[201] B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson, "Estimating the support of a high-dimensional distribution," *Neural computation*, vol. 13, no. 7, pp. 1443–1471, 2001.

[202] T. Durand, N. Thome, and M. Cord, "Mantra: Minimum maximum latent structural svm for image classification and ranking," in *ICCV*, 2015, pp. 2713–2721.

[203] R. Socher, B. Huval, B. Bath, C. D. Manning, and A. Y. Ng, "Convolutional-recursive deep learning for 3d object classification," in *NeurIPS*, 2012, pp. 656–664.

[204] A. Wang, J. Cai, J. Lu, and T.-J. Cham, "MMSS: Multi-modal sharable and specific feature learning for RGB-D object recognition," in *ICCV*, 2015, pp. 1125–1133.

[205] Y. Cheng, X. Zhao, R. Cai, Z. Li, K. Huang, Y. Rui *et al.*, "Semi-supervised multimodal deep learning for RGB-D object recognition," *IJCAI*, pp. 3346–3351, 2016.

[206] Q. Wang, M. Chen, F. Nie, and X. Li, "Detecting coherent groups in crowd scenes by multiview clustering," *IEEE TPAMI*, vol. 42, no. 1, pp. 46–58, 2018.

[207] C. Couprie, C. Farabet, L. Najman, and Y. LeCun, "Indoor semantic segmentation using depth information," in *ICLR*, 2013.

[208] Y. Zheng and X. Gao, "Indoor scene recognition via multi-task metric multi-kernel learning from RGB-D images," *Multimedia Tools and Applications*, vol. 76, no. 3, pp. 4427–4443, 2017.

[209] B. Thompson, "Canonical correlation analysis," *Encyclopedia of statistics in behavioral science*, 2005.

[210] G. Andrew, R. Arora, J. Bilmes, and K. Livescu, "Deep canonical correlation analysis," in *ICML*, 2013, pp. 1247–1255.

[211] H. Nakayama, T. Harada, and Y. Kuniyoshi, "Global gaussian approach for scene categorization using information geometry," in *CVPR*, 2010, pp. 2336–2343.

[212] N. Morioka and S. Satoh, "Building compact local pairwise codebook with joint feature space clustering," in *ECCV*, 2010, pp. 692–705.

[213] F. Cakir, U. Güdükbay, and Ö. Ulusoy, "Nearest-neighbor based metric functions for indoor scene recognition," *CVIU*, vol. 115, no. 11, pp. 1483–1492, 2011.

[214] S. N. Parizi, J. G. Oberlin, and P. F. Felzenszwalb, "Reconfigurable models for scene recognition," in *CVPR*, 2012, pp. 2775–2782.

[215] L. Wang, Y. Li, J. Jia, J. Sun, D. Wipf, and J. M. Rehg, "Learning sparse covariance patterns for natural scenes," in *CVPR*, 2012, pp. 2767–2774.

[216] T. Kobayashi, "BFO meets HOG: Feature extraction based on histograms of oriented p.d.f gradients for image classification," in *CVPR*, 2013, pp. 747–754.

[217] C. Doersch, A. Gupta, and A. A. Efros, "Mid-level visual element discovery as discriminative mode seeking," in *NeurIPS*, 2013, pp. 494–502.

[218] L. Zhou, Z. Zhou, and D. Hu, "Scene classification using a multi-resolution bag-of-features model," *Pattern Recognition*, vol. 46, no. 1, pp. 424–433, 2013.

[219] Q. Li, J. Wu, and Z. Tu, "Harvesting mid-level visual concepts from large-scale internet images," in *CVPR*, 2013, pp. 851–858.

[220] X. Wang, B. Wang, X. Bai, W. Liu, and Z. Tu, "Max-margin multiple-instance dictionary learning," in *ICML*, 2013, pp. 846–854.

[221] D. Lin, C. Lu, R. Liao, and J. Jia, "Learning important spatial pooling regions for scene classification," in *CVPR*, 2014, pp. 3726–3733.

[222] Y. Zheng, Y.-G. Jiang, and X. Xue, "Learning hybrid part filters for scene recognition," in *ECCV*, 2012, pp. 172–185.

[223] Y. Jiang, J. Yuan, and G. Yu, "Randomized spatial partition for scene recognition," in *ECCV*, 2012, pp. 730–743.

[224] J. Wu, B.-B. Gao, and G. Liu, "Representing sets of instances for visual recognition," in *AAAI*, 2016, pp. 2237–2243.

[225] J. H. Bappy, S. Paul, and A. K. Roy-Chowdhury, "Online adaptation for joint scene and object classification," in *ECCV*, 2016, pp. 227–243.

[226] Z. Zhao and M. Larson, "From volcano to toyshop: Adaptive discriminative region discovery for scene recognition," in *ACM MM*, 2018, pp. 1760–1768.

[227] M. Koskela and J. Laaksonen, "Convolutional network features for scene recognition," in *ACM MM*, 2014, pp. 1169–1172.

[228] G. Nascimento, C. Laranjeira, V. Braz, A. Lacerda, and E. R. Nascimento, "A robust indoor scene recognition method based on sparse representation," in *CIARP*, 2017, pp. 408–415.

[229] Z. Cai and L. Shao, "RGB-D scene classification via multi-modal feature learning," *Cognitive Computation*, vol. 11, no. 6, pp. 825–840, 2019.

[230] A. Ayub and A. Wagner, "Cbcl: Brain-inspired model for RGB-D indoor scene classification," *arXiv:1911.00155*, 2019.

[231] X. Song, L. Herranz, and S. Jiang, "Depth CNNs for RGB-D scene recognition: Learning from scratch better than transferring from RGB-CNNs," in *AAAI*, vol. 31, no. 1, 2017.

[232] M. M. Najafabadi, F. Villanustre, T. M. Khoshgoftaar, N. Seliya *et al.*, "Deep learning applications and challenges in big data analytics," *Journal of Big Data*, vol. 2, no. 1, p. 1, 2015.

[233] B. Zoph and Q. V. Le, "Neural architecture search with reinforcement learning," *ICLR*, 2017.

[234] T. Elsken, J. H. Metzen, F. Hutter *et al.*, "Neural architecture search: A survey." *JMLR*, vol. 20, no. 55, pp. 1–21, 2019.

[235] J. Xiao, K. A. Ehinger, J. Hays, A. Torralba, and A. Oliva, "SUN database: Exploring a large collection of scene categories," *IJCV*, vol. 119, no. 1, pp. 3–22, 2016.

[236] O. Chapelle, B. Scholkopf, and A. Zien, "Semi-supervised learning," *IEEE TNN*, vol. 20, no. 3, pp. 542–542, 2009.

[237] H. B. Barlow, "Unsupervised learning," *Neural computation*, vol. 1, no. 3, pp. 295–311, 1989.

[238] A. Kolesnikov, X. Zhai, and L. Beyer, "Revisiting self-supervised visual representation learning," in *CVPR*, 2019, pp. 1920–1929.

[239] Y.-X. Wang and M. Hebert, "Learning to learn: Model regression networks for easy small sample learning," in *ECCV*, 2016, pp. 616–634.

[240] L. Fei-Fei, R. Fergus, and P. Perona, "One-shot learning of object categories," *IEEE TPAMI*, vol. 28, no. 4, pp. 594–611, 2006.

[241] B. M. Lake, R. Salakhutdinov, and J. B. Tenenbaum, "Human-level concept learning through probabilistic program induction," *Science*, vol. 350, no. 6266, pp. 1332–1338, 2015.

[242] M. Long, Y. Cao, J. Wang, and M. Jordan, "Learning transferable features with deep adaptation networks," in *ICML*, 2015, pp. 97–105.

[243] M. Wang and W. Deng, "Deep visual domain adaptation: A survey," *Neurocomputing*, vol. 312, pp. 135–153, 2018.

[244] K.-C. Peng, Z. Wu, and J. Ernst, "Zero-shot deep domain adaptation," in *ECCV*, 2018, pp. 764–781.

[245] H. Li, S. Jialin Pan, S. Wang, and A. C. Kot, "Domain generalization with adversarial feature learning," in *CVPR*, 2018, pp. 5400–5409.

[246] X.-Y. Zhang, C.-L. Liu, and C. Y. Suen, "Towards robust pattern recognition: A review," *Proceedings of the IEEE*, vol. 108, no. 6, pp. 894–922, 2020.

[247] U. Shaham, Y. Yamada, and S. Negahban, "Understanding adversarial training: Increasing local stability of supervised models through robust optimization," *Neurocomputing*, vol. 307, pp. 195–204, 2018.

[248] C. Qin, J. Martens, S. Gowal, D. Krishnan, K. Dvijotham, A. Fawzi, S. De, R. Stanforth, and P. Kohli, "Adversarial robustness through local linearization," in *NeurIPS*, 2019, pp. 13 847–13 856.

[249] A. Shafahi, M. Najibi, M. A. Ghiasi, Z. Xu, J. Dickerson, C. Studer, L. S. Davis, G. Taylor, and T. Goldstein, "Adversarial training for free!" in *NeurIPS*, 2019, pp. 3358–3369.

[250] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky, "Domain-adversarial training of neural networks," *JMLR*, vol. 17, no. 1, pp. 2096–2030, 2016.

[251] A. Athalye, L. Engstrom, A. Ilyas, and K. Kwok, "Synthesizing robust adversarial examples," in *ICML*, 2018, pp. 284–293.

[252] A. R. Dargazany, P. Stegagno, and K. Mankodiya, "WearableDL: Wearable internet-of-things and deep learning for big data analyticsconcept, literature, and future," *Mobile Information Systems*, vol. 2018, 2018.

[253] L. Shen, Z. Lin, and Q. Huang, "Relay backpropagation for effective learning of deep convolutional neural networks," in *ECCV*, 2016, pp. 467–482.

[254] F. Thabtah, S. Hammoud, F. Kamalov, and A. Gonsalves, "Data imbalance in classification: Experimental evaluation," *Information Sciences*, vol. 513, pp. 429–441, 2020.

[255] M. Buda, A. Maki, and M. A. Mazurowski, "A systematic study of the class imbalance problem in convolutional neural networks," *Neural Networks*, vol. 106, pp. 249–259, 2018.

[256] J. M. Johnson and T. M. Khoshgoftaar, "Survey on deep learning with class imbalance," *Journal of Big Data*, vol. 6, no. 1, p. 27, 2019.

[257] Y. Guo, Y. Liu, A. Oerlemans, S. Lao, S. Wu, and M. S. Lew, "Deep learning for visual understanding: A review," *Neurocomputing*, vol. 187, pp. 27–48, 2016.

[258] Z. Chen and B. Liu, "Lifelong machine learning," *Synthesis Lectures on Artificial Intelligence and Machine Learning*, vol. 12, no. 3, pp. 1–207, 2018.

[259] S. Thrun and T. M. Mitchell, "Lifelong robot learning," *Robotics and autonomous systems*, vol. 15, no. 1-2, pp. 25–46, 1995.

[260] D. Hassabis, D. Kumaran, C. Summerfield, and M. Botvinick, "Neuroscience-inspired artificial intelligence," *Neuron*, vol. 95, no. 2, pp. 245–258, 2017.

[261] M.-L. Zhang and Z.-H. Zhou, "Ml-knn: A lazy learning approach to multi-label learning," *Pattern Recognition*, vol. 40, no. 7, pp. 2038–2048, 2007.

[262] C. Feichtenhofer, A. Pinz, and R. P. Wildes, "Temporal residual networks for dynamic scene recognition," in *CVPR*, 2017, pp. 4728–4737.

[263] T. Hackel, N. Savinov, L. Ladicky, J. D. Wegner, K. Schindler, and M. Pollefeys, "Semantic3d.net: A new large-scale point cloud classification benchmark," *arXiv:1704.03847*, 2017.

[264] A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, and M. Nießner, "Scannet: Richly-annotated 3d reconstructions of indoor scenes," in *CVPR*, 2017, pp. 5828–5839.

[265] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *ICCV*, 2015, pp. 4489–4497.

[266] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3D classification and segmentation," in *CVPR*, 2018, pp. 652–660.