Deep Spatial Pyramid Ensemble for Cultural Event Recognition

Xiu-Shen Wei Bin-Bin Gao Jianxin Wu*
National Key Laboratory for Novel Software Technology, Nanjing University, Nanjing, China
{weixs,gaobb}@lamda.nju.edu.cn, wujx2001@nju.edu.cn

Abstract

Semantic event recognition based only on image-based cues is a challenging problem in computer vision. In order to capture rich information and exploit important cues like human poses, human garments and scene categories, we propose the Deep Spatial Pyramid Ensemble framework, which is mainly based on our previous work, i.e.*, Deep Spatial Pyramid (DSP). DSP could build universal and powerful image representations from CNN models. Specifically, we employ five deep networks trained on different data sources to extract five corresponding DSP representations for event recognition images. For combining the complementary information from different DSP representations, we ensemble these features by both "early fusion" and "late fusion". Finally, based on the proposed framework, we come up with a solution for the track of the Cultural Event Recognition competition at the ChaLearn Looking at People (LAP) challenge in association with ICCV 2015. Our framework achieved one of the best cultural event recognition performance in this challenge.*

1. Introduction

Event recognition is one of the key tasks in computer vision. There have been many researches about video-based event recognition and action recognition [13, 15, 17, 18]. However, event recognition from still images has received little attention in the past, which is also a more challenging problem than the video-based event recognition task. Because videos could provide richer and more useful information (*e.g.*, motions and trajectories) for understanding events, while images of events just merely contain static appearance information.

Moreover, cultural event recognition is an important problem of event understanding. The goal of cultural event recognition is not only to find images with similar content, but further to find images that are semantically related to a particular type of event. Specifically, as shown in Fig. 1, images with very different visual appearances are possible to indicate the same cultural event, while images containing the same object might come from different cultural events. In addition, it is crucial for cultural event recognition to exploit several important cues like garments, human poses, objects and background at the same time.

In this paper, we propose the Deep Spatial Pyramid Ensemble framework for cultural event recognition, which is mainly based on our previous work, *i.e.*, the Deep Spatial Pyramid (DSP) method [5]. This method builds universal image representations from CNN models, while adapting this universal image representation to different image domains in different applications. In DSP, it firstly extract the deep convolutional activations of an input image with arbitrary resolution by a pre-trained CNN. These deep activations are then encoded into a new high dimensional feature representation by overlaying a spatial pyramid partition. Additionally, in order to capture the important cues (*e.g.*, human poses, objects and background) of cultural event recognition images, we employ two types deep networks, *i.e.*, VGG Nets [14] trained on ImageNet [12] and Place-CNN [23] trained on the Places database [23]. Meanwhile, we also fine-tune VGG Nets on cultural event images [3]. After that, we utilize these deep networks trained on different data sources to extract different DSP representations for cultural event images. Finally, we ensemble the information from multiple deep networks via "early fusion" and "late fusion" to boost the recognition performance.

In consequence, based on the proposed framework, we come up with a solution of five DSP deep convolutional networks ensemble for the track of Cultural Event Recognition at the ChaLearn Looking at People (LAP) challenge in association with ICCV 2015. Our proposed framework achieved one of the best cultural event recognition performance in the Final Evaluation phase.

The rest of this paper is organized as follows. In Sec. 2, we present the proposed framework, and mainly introduce the key method DSP. Implementation details and experimental results of the cultural event recognition competition are described in Sec. 3. Finally, we conclude our method and present the future works in Sec. 4.

*This work was supported by the National Natural Science Foundation of China under Grant 61422203 and the Collaborative Innovation Center of Novel Software Technology and Industrialization.

Figure 1. Images randomly sampled from 99 categories of the cultural event recognition images [3]. The cultural event recognition dataset contains 99 important cultural events from all around the globe, which includes: *Carnival of Venice* (Italy), *Gion matsuri* (Japan), *Harbin Ice and Snow Festival* (China), *Oktoberfest* (Germany), *Mardi Gras* (USA), *Tapati rapa Nui* (Chile) and so on.

2. The proposed framework

In this section, we will introduce the proposed Deep Spatial Pyramid Ensemble framework, especially the main approach used in this paper, *i.e.*, Deep Spatial Pyramid (DSP) [5].

Recently, thanks to the rich semantic information extracted by the convolutional layers of CNN, convolutional layer deep descriptors have exemplified their value and been successful in [10, 2, 20]. Moreover, these deep descriptors contain more spatial information compared to the activation of the fully connected layers, *e.g.*, the top-left cell's d-dim deep descriptor is generated using only the top-left part of the input image, ignoring other pixels. In addition, fully connected layers have large computational cost, because it contains roughly 90% of all the parameters of the whole CNN model. Thus, here we use fully convolutional networks by removing the fully connected layers as feature extractors.

In the proposed framework, we feed an input image with arbitrary resolution into a pre-trained CNN model to extract deep activations in the first step. Then, a visual dictionary with K dictionary items is trained on the deep descriptors from training images. The third step overlay a spatial pyramid partition to the deep activations of an image into m blocks in N pyramid levels. One spatial block is represented as a vector by using the improved Fisher Vector. Thus, m blocks correspond to m FVs. In the fourth and fifth step, we concatenate the m FVs to form a $2mdK$-dimensional feature vector as the final image-level representation. These steps are shown as the key parts of our framework in Fig. 2. In addition, since cultural event recognition is highly related with two high-level computer vision problems, *i.e.*, object recognition and scene recognition, we employ multiple pre-trained CNNs (*e.g.*, VGGNets [14] and Place-CNN [23]) to extract the DSP representations for each image in this competition, and then ensemble the complementary information from multiple CNNs.

In the following, we will firstly present some detailed factors in DSP, and secondly introduce the Deep Spatial Pyramid method, and finally describe the ensemble strategy used in our framework for the cultural event recognition competition.

2.1. The ℓ_2 matrix normalization in DSP

Let $X = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_t, \ldots, \boldsymbol{x}_T]^T$ ($X \in R^{T \times d}$) be the matrix of d-dimensional deep descriptors extracted from an image I via a pre-trained CNN model. X was usually processed by dimensionality reduction methods such as PCA, before they are pooled into a single vector using VLAD or FV [6, 21]. PCA is usually applied to the SIFT features or fully connected layer activations, since it is empirically shown to improve the overall recognition performance. However, as studied in [5], it shows that PCA significantly hurts recognition when applied to the fully convolutional activations. Thus, it is not applied to fully convolutional deep descriptors in this paper.

In addition, multiple types of deep descriptors normalization have been evaluated, and the ℓ_2 matrix normalization before using FV is found to be important for better performance, cf. Table 2 in [5]. Therefore, we employ the ℓ_2 matrix normalization for the cultural event recognition competition as follows:

$$\boldsymbol{x}_t \leftarrow \boldsymbol{x}_t / \|X\|_2 \,, \tag{1}$$

where $\|X\|_2$ is the matrix spectral norm, *i.e.*, largest singular value of X. This normalization has a benefit that it normalizes $\boldsymbol{x}_t$ using the information from the entire image X, which makes it more robust to changes such as illumination and scale.

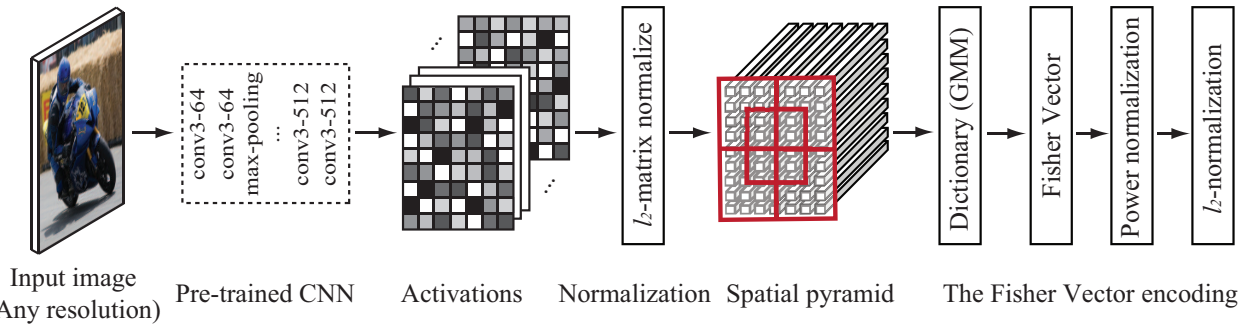


Figure 2. The image classification framework. DSP feeds an arbitrary resolution input image into a pre-trained CNN model to extract deep activations. A GMM visual dictionary is trained based on the deep descriptors from training images. Then, a spatial pyramid partitions the deep activations of an image into m blocks in N pyramid levels. In this way, each block activations are represented as a single vector by the improved Fisher Vector. Finally, we concatenate the m single vectors to form a $2mdK$-dimensional feature vector as the final image representation.

2.2. Encoding deep descriptors by FV

The size of pool_5 is a parameter in CNN because input images have arbitrary sizes. However, the classifiers (*e.g.*, SVM or soft-max) require fixed length vectors. Thus, all the deep descriptors of an image must be pooled to form a single vector. Here, similarly to DSP, we also use the Fisher Vector (FV) to encode the deep descriptors.

We denote the parameters of the GMM with K components by $\lambda = \{\omega_k, \boldsymbol{\mu}_k, \boldsymbol{\sigma}_k; k = 1, \ldots, K\}$, where ω_k, $\boldsymbol{\mu}_k$ and $\boldsymbol{\sigma}_k$ are the mixture weight, mean vector and covariance matrix of the k-th Gaussian component, respectively. The covariance matrices are diagonal and $\boldsymbol{\sigma}_k^2$ are the variance vectors. Let $\gamma_t(k)$ be the soft-assignment weight of $\boldsymbol{x}_t$ with respect to the k-th Gaussian, the FV representation corresponding to $\boldsymbol{\mu}_k$ and $\boldsymbol{\sigma}_k$ are presented as follows [11]:

$$\boldsymbol{f}_{\boldsymbol{\mu}_k}(X) = \frac{1}{\sqrt{\omega_k}} \sum_{t=1}^{T} \gamma_t(k) \left(\frac{\boldsymbol{x}_t - \boldsymbol{\mu}_k}{\boldsymbol{\sigma}_k} \right), \tag{2}$$

$$\boldsymbol{f}_{\boldsymbol{\sigma}_k}(X) = \frac{1}{\sqrt{2\omega_k}} \sum_{t=1}^{T} \gamma_t(k) \left[\frac{(\boldsymbol{x}_t - \boldsymbol{\mu}_k)^2}{\boldsymbol{\sigma}_k^2} - 1 \right]. \tag{3}$$

Note that, $\boldsymbol{f}_{\boldsymbol{\mu}_k}(X)$ and $\boldsymbol{f}_{\boldsymbol{\sigma}_k}(X)$ are both d-dimensional vectors. The final Fisher Vector $\boldsymbol{f}_\lambda(X)$ is the concatenation of the gradients $\boldsymbol{f}_{\boldsymbol{\mu}_k}(X)$ and $\boldsymbol{f}_{\boldsymbol{\sigma}_k}(X)$ for all K Gaussian components. Thus, FV can represent the set of deep descriptors X with a $2dK$-dimensional vector. In addition, the Fisher Vector $\boldsymbol{f}_\lambda(X)$ is improved by the power-normalization with the factor of 0.5, followed by the ℓ_2 vector normalization [11].

Moreover, as discussed in [5], a very small K (*e.g.*, 2, 3 or 4) in Fisher Vector surprisingly achieves higher accuracy than normally used large K values. In our experiments of cultural event recognition, we fix the K value as 2.

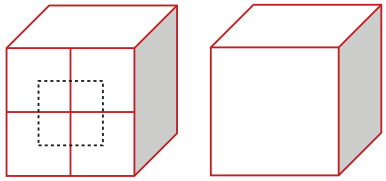


Figure 3. Illustration of the level 1 and 0 deep spatial pyramid.

2.3. Deep spatial pyramid

The key part of DSP is adding spatial pyramid information much more naturally and simply. Also, adding spatial information through a spatial pyramid [9] has been shown to significantly improve image recognition performance not only when using dense SIFT features but when using fully convolutional activations [7].

In SPP-net [7], it adds a spatial pyramid pooling layer to deep nets, which has improved recognition performance. However, in DSP, a more intuitive and natural way exists.

As previously discussed, one single cell (deep descriptor) in the last convolutional layer corresponds to one local image patch in the input image, and the set of all convolutional layer cells form a regular grid of image patches in the input image. This is a direct analogy to the dense SIFT feature extraction framework. Instead of a regular grid of SIFT vectors extracted from 16×16 local image patches, a grid of deep descriptors are extracted from larger image patches by a CNN.

Thus, DSP can easily form a natural deep spatial pyramid by partitioning an image into sub-regions and computing local features inside each sub-region. In practice, we just need to spatially partition the cells of activations in the last convolutional layer, and then pool deep descriptors in each region separately using FV. The operation of DSP is illustrated in Fig. 3.

The level 0 simply aggregates all cells using FV. The

level 1, however, splits the cells into 5 regions according to their spatial locations: the 4 quadrants and 1 centerpiece. Then, 5 FVs are generated from activations inside each spatial region. Note that the level 1 spatial pyramid used in DSP is different from the classic one in [9]. It follows Wu and Rehg [19] to use an additional spatial region in the center of the image. A DSP using two levels will then concatenate all 6 FVs from level 0 and level 1 to form the final image representation.

This DSP method is summarized in Algorithm 1.

Algorithm 1 The DSP pipeline

Input:
  An input image I
  A pre-trained CNN model
Procedure:
  Extract deep descriptors X from I using the pre-defined model, $X = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_t, \ldots, \boldsymbol{x}_T]^T$
  For each activation vector $\boldsymbol{x}_t$, perform ℓ_2 matrix normalization $\boldsymbol{x}_t \leftarrow \boldsymbol{x}_t / \|X\|_2$
  Estimate a GMM $\lambda = \{\omega_k, \boldsymbol{\mu}_k, \boldsymbol{\sigma}_k\}$ using the training set
  Generate a spatial pyramid $\{X_1, \ldots, X_m\}$ for X
  for all $1 \leq i \leq m$
    $\boldsymbol{f}_\lambda(X_i) \leftarrow [\boldsymbol{f}_{\boldsymbol{\mu}_1}(X_i), \boldsymbol{f}_{\boldsymbol{\sigma}_1}(X_i), \ldots, \boldsymbol{f}_{\boldsymbol{\mu}_K}(X_i), \boldsymbol{f}_{\boldsymbol{\sigma}_K}(X_i)]$
    $\boldsymbol{f}_\lambda(X_i) \leftarrow \mathrm{sign}(\boldsymbol{f}_\lambda(X_i))\sqrt{\boldsymbol{f}_\lambda(X_i)}$
    $\boldsymbol{f}_\lambda(X_i) \leftarrow \boldsymbol{f}_\lambda(X_i) / \|\boldsymbol{f}_\lambda(X_i)\|_2$
  end for
  Concatenate $\boldsymbol{f}_\lambda(X_i)$, $1 \leq i \leq m$, to form the final spatial pyramid representation $\boldsymbol{f}(X)$
  $\boldsymbol{f}(X) \leftarrow \boldsymbol{f}(X) / \|\boldsymbol{f}(X)\|_2$
Output: $\boldsymbol{f}(X)$.

2.4. Multi-scale DSP

In order to capture variations of the activations caused by variations of objects in an image, we generate a multiple scale pyramid, extracted from S different rescaled versions of the original input image. We feed images of all different scales into a pre-trained CNN model and extract deep activations. In each scale, the corresponding rescaled image is encoded into a $2mdK$-dimensional vector by DSP. Therefore, we have S vectors of $2mdK$-dimensions and they are merged into a single vector by average pooling, as

$$\boldsymbol{f}_m = \frac{1}{S} \sum_{s=1}^{S} \boldsymbol{f}_s \,, \tag{4}$$

where $\boldsymbol{f}_s$ is the DSP representation extracted from the scale level s. Finally, ℓ_2 normalization is applied to $\boldsymbol{f}_m$. Note that each vector $\boldsymbol{f}_s$ is already ℓ_2 normalized, as shown in Algorithm 1.

The multi-scale DSP is related to MPP proposed by Yoo *et al.* [21]. A key different between our method and MPP is that $\boldsymbol{f}_s$ encodes spatial information while MPP does not. During the competition of cultural event recognition, we find that a large scale will achieve a better performance. Thus, we employ four scales, *i.e.*, 1.4, 1.2, 1.0 and 0.8, and the experimental results are shown in Sec. 3.3.

2.5. Ensemble of multiple DSPs

In the past several years, many successful deep CNN architectures have been shown to further improve CNN performance, characterized by deeper and wider architectures and smaller convolutional filters when compared to traditional CNN such as [8, 22]. Examples of deeper nets include GoogLeNet [16], VGG Net-D and VGG Net-E [14].

Specifically, in order to exploit different types information from cultural event images, we choose the VGG Net-D and VGG Net-E for object recognition, and utilize the Place-CNN net [23] as pre-trained deep network for scene recognition. VGG Net-D and VGG Net-E consist of the similar architectures and parameters of convolutional and pooling filters. More details of these two deep networks can be found in [14]. In addition, to boost recognition performance, we also fine-tune VGG Net-D and VGG Net-E on the training and validation images/crops of the competition. Therefore, for one image/crop, we can get five DSP representations extracted from the aforementioned five CNN models. Because these CNN models are trained on different types of images (*i.e.*, object-centric images, scene-centric images and event-centric images), we ensemble the complementary information of multiple CNN models by treating these DSP representations as multi-view data.

We denote the multi-scale DSP representation extracted from the i-th CNN model by $\boldsymbol{f}_m^i$. After extracting these DSP representations, we concatenate all the features and apply ℓ_2 normalization as follows:

$$\boldsymbol{f}_{final} \leftarrow \left[\boldsymbol{f}_m^1, \boldsymbol{f}_m^2, \boldsymbol{f}_m^3, \boldsymbol{f}_m^4, \boldsymbol{f}_m^5\right] , \tag{5}$$

$$\boldsymbol{f}_{final} \leftarrow \boldsymbol{f}_{final} / \|\boldsymbol{f}_{final}\|_2 \,, \tag{6}$$

which is called as "early fusion" in this paper. Note that, the dimensionality of deep descriptors in the last convolutional layer is 512 and 256 for VGG Nets and Place-CNN, respectively. Thus, followed the aforementioned experimental settings, the DSP representations of VGG Nets and Place-CNN are of 12,288- and 6,144-dimension, and the final DSP representation of each image is a 55,296-dimensional vector.

3. Experiments

In this section, we first describe the dataset of cultural event recognition at the ICCV ChaLearn LAP 2015 competition [3]. Then we give a detailed description about the implementation details of the proposed framework. Finally,

we present and analyze the experimental results of the proposed framework on the competition dataset.

3.1. Datasets and evaluation criteria

The cultural event recognition at the ICCV ChaLearn LAP 2015 competition [3] is the second round for this track. Compared with the previous one, the task of ICCV ChaLearn LAP 2015 has significantly increased the number of images and classes, adding a new "no-event" class. As a result, for this track more than 28,000 images are labeled to perform automatic cultural event recognition from 100 categories in still images. These images are collected from two image search engines (*Google* and *Bing*), which belong to 99 different cultural events and one non-class. This is the first dataset on cultural events from all around the globe. From these images, we see that several cues like garments, human poses, objects and background could be exploited for recognizing the cultural events.

The dataset is divided into three parts: the training set (14,332 images), the validation set (5,704 images) and the evaluation set (8,669 images). During the development phase, we train our model on the training set and verify its performance on the validation set. For final evaluation, we merge the training and validation set into a single data set and re-train our model. The principal quantitative measure used is the average precision (AP), which is calculated by numerical integration.

3.2. Implementation details

Before extracting the DSP representations, we get the original distributions of the numbers of training images in both Development and Final Evaluation, which are shown in Fig. 4(a) and Fig. 4(c), respectively. From these figures, we can see the "non-event" class is of large quantity and the original dataset is apparently class-imbalanced. To fix this problem, for each image of the other 99 cultural event classes, we extract three 384×384 crops which are illustrated in Fig. 5. Moreover, in order to keep the original semantic meaning of each image, we fix the location of each corresponding crop. In addition, we also get the horizontal reflection of the 99 cultural event images. Therefore, the number of cultural event images/crops will become 5 times as the original one, which on one hand can supply diverse data sources, and on the other hand can solve the class-imbalanced problem. During the testing phase, because we do not know the classes of testing images, all the testing images will be augmented by the aforementioned process.

After data augmentation, as aforementioned, we employ three popular deep CNNs as pre-trained models, including *VGG Net-D*, *VGG Net-E* [14] and *Place-CNN* [23]. In addition, we also fine-tune VGG Net-D and VGG Net-E on the images/crops of the competition. In consequence, we obtain five deep networks (*i.e.*, VGG Net-D, VGG Net-E, fine-tuned VGG Net-D, fine-tuned VGG Net-E and Place-CNN) and use them to extract the corresponding DSP representations for each image/crop. Thus, each image of both training (except for the "no-event" class) and testing will be represented by five DSP features/instances. As described in Sec. 2.5, we concatenate these DSP features and apply ℓ_2 normalization to get the final representation for each image/crop. Finally, we feed these feature vectors into logistic regression [4] to build a classifier and use the softmax as

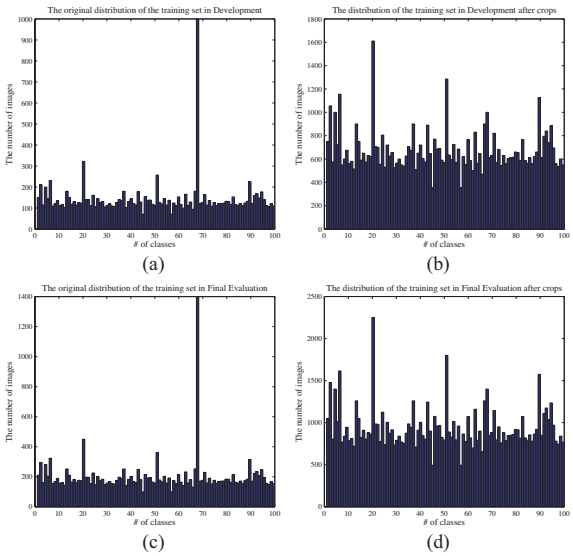


Figure 4. Distributions of the number of training images in Development and Final Evaluation. (a) and (c) are the original distributions of training images in both Development and Final Evaluation, respectively. (b) and (d) are the distributions of training images after crops.

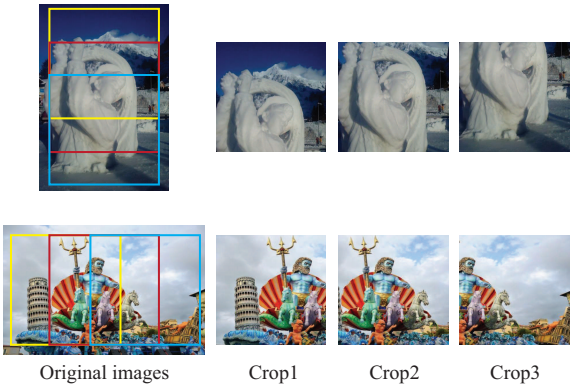


Figure 5. Crops of the original images. Different from the random crops used in other deep networks (*e.g.*, [8]), we fix the locations of these crops, which can keep the original semantic meaning of cultural event images. If we get the random crops, for example the second original image of *Carnevale di Viareggio*, it might get one crop only contains sky, which will hurt the cultural event recognition performance. These figures are best viewed in color.

the prediction scores of images/crops. And then, the final scores of testing images can be obtained by averaging the scores across their corresponding crops and horizontal reflections, which is called "late fusion" corresponding to the former one mentioned in Sec. 2.5.

3.3. Experimental results

In this section, we first present the experimental results of the Development phase and analyze our proposed framework. Finally, we show the Final Evaluation results of this cultural event recognition competition.

3.3.1 Development

In Table 1, we present the main results in the Development phase. As discussed in Sec. 2.4, the multiple scales (MS) strategy could capture the variation information, which boosts the performance by about 1% mAP on VGG Net-D ($0.761 \rightarrow 0.770$) and VGG Net-E ($0.762 \rightarrow 0.773$). In addition, the late fusion approach is also effective. From this table, it improves more than 1% mAP on the pre-trained VGG nets, and improves performance by 2% when deep networks are fine-tuned on cultural event images/crops of the competition. Because these deep networks are trained on different image sources, *i.e.*, ImageNet [12], Places [23] and Cultural Event Recognition [3], they can supply complementary information for each image of this competition. Thus, we do "early fusion" by concatenating these DSP representations extracted from the five deep networks, and then get the final prediction score of each testing image in Development via "late fusion". The ensemble performance (0.841) can significantly outperform the previous ones.

In order to further investigate this complementarity, we visualize the feature maps of these five deep networks in Fig. 6. As shown in those figures, the strongest responses in the corresponding feature maps of these deep networks are quite different from each other, especially the one of Place-CNN, *i.e.*, Fig. 6 (f). Apparently, different pre-trained deep networks trained on different data sources could extract complementary information for each image in cultural event recognition.

3.3.2 Final evaluation

As aforementioned, in the Final Evaluation phase, we merge the training and validation set into a single data set and do the similar processes, *i.e.*, data augmentation, fine-tuning, "early fusion" and "late fusion", etc. The final challenge results are shown in Table 2. Our final result (0.851) is slightly lower (0.3%) than the team ranked 1^{st}. For further improving recognition performance of the proposed framework, a very simple and straightforward way is to apply the "bagging" approach [1] on the concatenated DSP representations of each image/crop, and then get the corresponding prediction scores for the testing images/crops. After several times bagging processes, the final prediction scores can be obtained by averaging the results of multiple baggings. Moreover, advanced ensemble methods can be also simply applied into our framework to achieve better performance.

Table 2. Comparison performances of our proposed framework with that of the top five teams in the Final Evaluation phase.

Rank	Team	Score
1	VIPL-ICT-CAS	0.854
2	**FV (Ours)**	0.851
3	MMLAB	0.847
4	NU&C	0.824
5	CVL_ETHZ	0.798

4. Conclusion

Event recognition from still images is one of the challenging problems in computer vision. In order to exploit and capture important cues like human poses, human garments and other context, this paper has proposed the Deep Spatial Pyramid Ensemble framework. In consequence, based on the proposed framework, we employ five deep CNN networks trained on different data sources and ensemble their complementary information. Finally, we utilize the proposed framework for the track of cultural event recognition [3] at the ChaLearn LAP challenge in association with ICCV 2015, and achieve one of the best recognition performance in the Final Evaluation phase. In the future, we will introduce more advanced ensemble methods into our framework and incorporating more visual cues for event understanding.

References

[1] L. Breiman. Bagging predictors. *MLJ*, 24:123–140, 1996.

[2] M. Cimpoi, S. Maji, and A. Vedaldi. Deep convolutional filter banks for texture recognition and segmentation. In *CVPR*, pages 3828–3836, 2015.

[3] S. Escalera, J. Fabian, P. Pardo, X. Baró, J. Gonzàlez, H. J. Escalante, and I. Guyon. ChaLearn 2015 apparent age and cultural event recognition: Datasets and results. In *ICCV ChaLearn Looking at People workshop*, 2015.

[4] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. LIBLINEAR: A library for large linear classification. *JMLR*, 9:1871–1874, 2008.

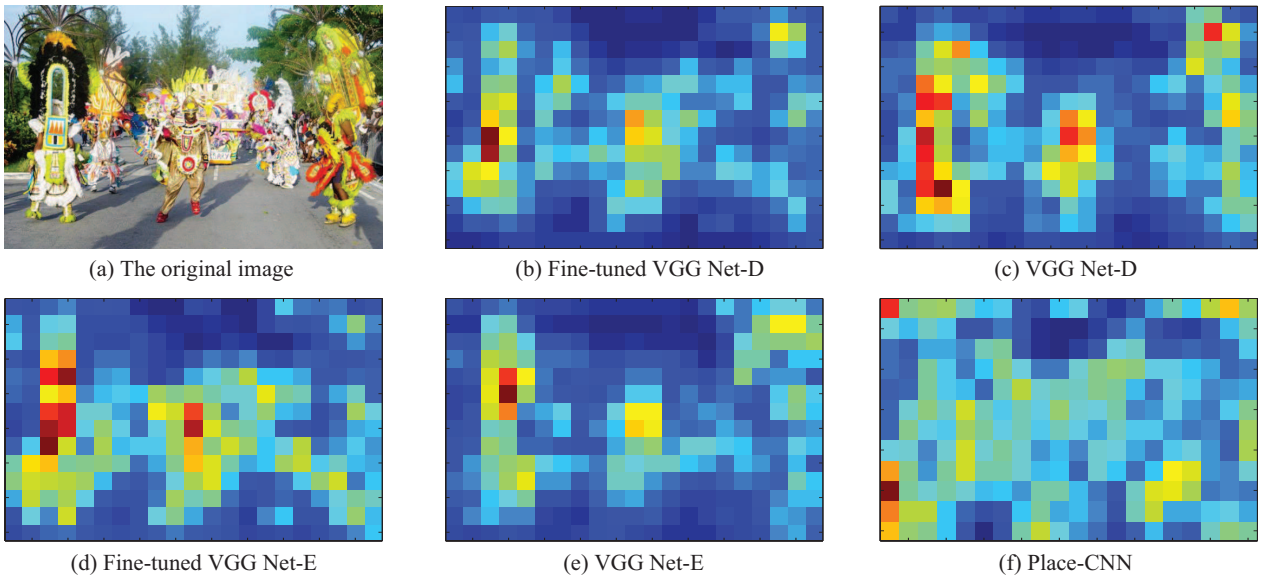
[5] B.-B. Gao, X.-S. Wei, J. Wu, and W. Lin. Deep spatial pyramid: The devil is once again in the details. *arXiv:1504.05277*, 2015.

[6] Y. Gong, L. Wang, R. Guo, and S. Lazebnik. Multi-scale orderless pooling of deep convolutional activation features. In *ECCV*, pages 392–407, 2014.

[7] K. He, X. Zhang, S. Ren, and J. Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. In *ECCV*, pages 346–361, 2014.

Table 1. Recognition mAP comparisons of the Development phase. Note that, "FT" stands for the fine-tuned deep networks; "SS" is for single scale, and "MS" is for multiple scales.

	VGG Net-D	VGG Net-E	FT VGG Net-D	FT VGG Net-E	Place-CNN
SS	0.761	0.762	–	–	–
MS	0.770	0.773	0.779	0.769	0.640
Late fusion	0.782	0.784	0.802	0.791	0.649
Ensemble	0.841				



(a) The original image (b) Fine-tuned VGG Net-D (c) VGG Net-D

(d) Fine-tuned VGG Net-E (e) VGG Net-E (f) Place-CNN

Figure 6. Feature maps of an image of *Junkanoo*. (a) is the original image. For each feature map, we summarize the responses values of all the depths in the final pooling layer for each deep network. (b) and (d) are the feature maps of the fine-tuned VGG Net-D and fine-tuned VGG Net-E, respectively. (c) and (e) are the ones of the pre-trained VGG nets. (f) is the feature map of Place-CNN. These figures are best viewed in color.

[8] A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet classification with deep convolutional neural networks. In *NIPS*, pages 1097–1105, 2012.

[9] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *CVPR*, pages 2169–2178, 2006.

[10] L. Liu, C. Shen, and A. van den Hengel. The treasure beneath convolutional layers: Cross-convolutional-layer pooling for image classification. In *CVPR*, pages 4749–4757, 2015.

[11] F. Perronnin, J. Sánchez, and T. Mensink. Improving the fisher kernel for large-scale image classification. In *ECCV*, pages 143–156, 2010.

[12] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet large scale visual recognition challenge. *IJCV*, 2015.

[13] K. Simonyan and A. Zisserman. Two-stream convolutional networks for action recognition in videos. In *NIPS*, pages 568–576, 2014.

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

[15] C. Sun and R. Nevatia. ACTIVE: activity concept transitions in video event classification. In *ICCV*, pages 913–920, 2013.

[16] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. *arXiv:1409.4842*, 2014.

[17] H. Wang and C. Schmid. Action recognition with improved trajectories. In *ICCV*, pages 3551–3558, 2013.

[18] L. Wang, Y. Qiao, and X. Tang. Action recognition with trajectory-pooled deep-convolutional descriptors. In *CVPR*, pages 1–10, 2015.

[19] J. Wu and J. M. Rehg. CENTRIST: A visual descriptor for scene categorization. *IEEE TPAMI*, 33(8):1489–1501, 2011.

[20] Z. Xu, Y. Yang, and A. G. Hauptmann. A discriminative CNN video representation for event detection. In *CVPR*, pages 1798–1807, 2015.

[21] D. Yoo, S. Park, J.-Y. Lee, and I. S. Kweon. Fisher kernel for deep neural activations. *arXiv:1412.1628*, 2014.

[22] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In *ECCV*, pages 818–833, 2014.

[23] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva. Learning deep features for scene recognition using places database. In *NIPS*, pages 487–495, 2014.