1 Team details

- **Team name**
  SSTK

- **Team leader name**
  Vaibhav Malpani

- **Team leader address, phone number and email**
  350 Fifth Avenue, 21st Floor, New York, NY 10118
  (410) 209-9611
  vaibhav.malpani@gmail.com

- **Rest of the team members**
  David Chester, Nathan Hurst, Mike Ranzinger

- **Affiliation**
  Shutterstock Inc

2 Contribution details

- **Title of the contribution**
  Transfer Learning for Cultural Event Recognition in Images using Convolutional Neural Networks

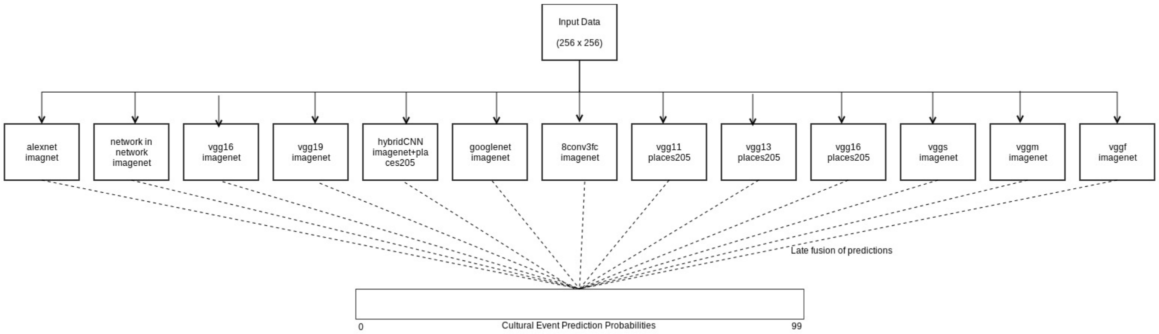
- **Final score**
  Validation Phase: 74%
  Test phase score on our 93/7 train/test split: 77.68%

- **General method description**
  It is known that transferability of features decreases as the distance between the base task and target task increases, but transferring features even from distant tasks can be better than using random features. So we follow the usual transfer learning approach to train a base network and then copy its first n layers to the first n layers of a target network. The remaining layers of the target network are then randomly initialized and trained towards the target task.
  We know that cultural event images involve scene understanding and some object recognition [11]. So we create an ensemble of CNNs originally trained either on Places-205 dataset [7] or ILSVRC-2012 dataset [2].
  We also find that the type of features learnt from the networks are complementary to each other to some extent. Finally, we perform late fusion to further boost the recognition performance.

- **References**

1. Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. B. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. CoRR, abs/1408.5093, 2014.
2. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., and Fei-Fei, L. ImageNet large scale visual recognition challenge. CoRR, abs/1409.0575, 2014
3. K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman. Return of the devil in the details: Delving deep into convolutional nets. In BMVC, pages 1–12, 2014.
4. A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet classification with deep convolutional neural networks. In NIPS, pages 1106–1114, 2012.
5. K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. CoRR, abs/1409.1556, 2014.
6. C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. CoRR, abs/1409.4842, 2014.
7. 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.
8. Network In Network M. Lin, Q. Chen, S. Yan International Conference on Learning Representations, 2014
9. Training Deeper Convolutional Networks with Deep Supervision L.Wang, C.Lee, Z.Tu, S. Lazebnik, 2015
10. ParseNet: Looking Wider to See Better, Wei Liu, Andrew Rabinovich, Alexander C. Berg
11. Object-Scene Convolutional Neural Networks for Event Recognition in Images, Limin Wang, Zhe Wang, Wenbin Du, Yu Qiao
12. Cultural Event Recognition by Subregion Classification with Convolutional Neural Network, Sungheon Park and Nojun Kwak

- **Representative image / diagram of the method**



3 Data Preprocessing

- **Describe features used or data representation model**
  We use convolutional neural networks to automatically generate image features. No hand-crafted feature engineering was done. We see that the type of features learnt from each of the networks are complementary to each other either due to a different network topology or due to a different dataset being used for pretraining these deep networks.

- **Other techniques/strategy used not included in previous items for data preprocessing (if any)**
  Input images are normalized by mean subtraction. Also we feed in multiple crops of the input image while training and oversample an image at prediction time.

4 Classification details

- **Classifier or method used to train/ validate your results (if any)**
  - Stochastic gradient descent with momentum is used for training and several models are averaged to improve the generalization.
  - As we are finetuning from a pretrained model, we start with a low global learning rate of 0.0005 which is decayed by a factor of 0.5 every 5000 iterations. The learning rate for the final fully connected layer is increased relative to other layers.
  - Softmax loss is used for error backpropagation.

- **Compositional model used (scene context representation), i.e. pictorial structure (if any)**
  Our architecture is inspired by [11]. It comprises of networks trained on object categories (ILSVRC2012) [2] as well as networks trained on

scene categories (Places-205) [7]. They extract useful information for event understanding from the perspective of objects and scene context.

5 Global Method Description

- **Which pre-trained or external methods have been used (for any stage, if any)**
  We used pretrained models publically available on Caffe Model Zoo page.
  - alexnet pretrained on imagenet [4, 7]
  - network in network pretrained on imagenet [7, 8]
  - vgg-16 pretrained on imagenet [5, 7]
  - vgg-19 pretrained on imagenet [5, 7]
  - hybridCNN pretrained on imagenet + places-205 [3,7]
  - googlenet pretrained on imagenet [6, 7]
  - 8conv3fc pretrained on imagenet [7, 9]
  - vgg11 pretrained on places205 [5,7]
  - vgg13 pretrained on places205 [5,7]
  - vgg16 pretrained on places205 [5,7]
  - vgg-s pretrained on imagenet [2,3]
  - vgg-m pretrained on imagenet [2,3]
  - vgg-f pretrained on imagenet [2,3]

- **Qualitative advantages of the proposed solution**
  As stated above, we find that the type of features learnt by each of the networks are complementary to each other.
  Using multiple models enabled us to get a better generalization in case some model overfits.
  We observed that the top-5 accuracies were around 90% while the top-1 accuracy were around 70% on our train/test split. This motivated us to think about using multiple models as an technique to boost up the right event category amongst the top-5 predicted by any single model.

- **Results of the comparison to other approaches (if any)**
  On submitting results during the first phase of the competition we saw that any individual model wasn't able to secure more than 58% mean average precision. But as soon as we started fusing models capturing complementary features, our performance went on improving linearly with the number of models. Our final submission during the first phase of the competition consisted of an ensemble of 15 model predictions getting a mean average precision score of 74% [12].

- **Novelty degree of the solution and if is has been previously published**
  Early and late fusion are well established techniques in the field of computer vision [12]. We know that different model architectures trained on different datasets capture varied features [11]. We simply built on top of these ideas to secure a position in the top 5 teams during the first phase of the competition.

6 Other details

- **Language and implementation details (including platform, memory, parallelization requirements)**
  - Python was the primary programming language
  - We used publicly available C++ Caffe [1] toolbox for training models
  - Models were trained on NVIDIA Titan X GPU with 12GB memory
- **Human effort required for implementation, training and validation?**
  1 developer month
- **Training/testing expended time?**
  Very deep networks like googlenet and vgg16/vgg19 took around a day to train. Deep networks like alexnet and network in network trained in about 6-8 hours.
- **General comments and impressions of the challenge?**
  It was a great experience overall. Trying out a number of techniques to improve our performance, helped us sharpen our deep learning skills and toolset. The dataset was quite challenging with twice the number of classes as the previous iteration of the competition. At the same time, it became more competitive with around 6 teams having mean average precision over 70%.
  We would like to thank the organizers for their time and effort in being very responsive to our queries.