A

Deep Convolutional Neural Networks for Fire Detection in Images

Jivitesh Sharma, Ole-Christoffer Granmo, Morten Goodwin and Jahn Thomas Fidje

University of Agder (UiA), Norway

Abstract. Detecting fire in images using image processing and computer vision techniques has gained a lot of attention from researchers during the past few years. Indeed, with sufficient accuracy, such systems may outperform traditional fire detection equipment. One of the most promising techniques used in this area is Convolutional Neural Networks (CNNs). However, the previous research on fire detection with CNNs has only been evaluated on balanced datasets, which may give misleading information on real-world performance, where fire is a rare event. Actually, as demonstrated in this paper, it turns out that a traditional CNN performs relatively poorly when evaluated on the more realistically balanced benchmark dataset provided in this paper. We therefore propose to use even deeper Convolutional Neural Networks for fire detection in images, and enhancing these with fine tuning based on a fully connected layer. We use two pretrained state-of-the-art Deep CNNs, VGG16 and Resnet50, to develop our fire detection system. The Deep CNNs are tested on our imbalanced dataset, which we have assembled to replicate real world scenarios. It includes images that are particularly difficult to classify and that are deliberately unbalanced by including significantly more non-fire images than fire images. The dataset has been made available online. Our results show that adding fully connected layers for fine tuning indeed does increase accuracy, however, this also increases training time. Overall, we found that our deeper CNNs give good performance on a more challenging dataset, with Resnet50 slightly outperforming VGG16. These results may thus lead to more successful fire detection systems in practice.

Keywords: Fire Detection, Deep Convolutional Neural Networks, VGG16, Resnet50.

1 Introduction

Emergency situations like floods, earthquakes and fires pose a big threat to public health and safety, property and environment. Fire related disasters are the most common type of Emergency situation which requires thorough analysis of the situation required for a quick and precise response. The first step involved in this process is to detect fire in the environment as quickly and accurately as possible.

Fire Detection in most places employs equipment like temperature detectors, smoke detectors, thermal cameras etc. which is expensive and not available to all [14]. But, after the advent of advanced image processing and computer vision techniques, detection of fire may not require any equipment other than cameras. Due to this expeditious development in vision-based fire detection models, there is a particular inclination towards replacing the traditional fire detection tools with vision-based models. These models have many advantages over their hardware based counterparts like accuracy, more detailed view of the situation, less prone to errors, robustness towards the environment, considerably lower cost and the ability to work on existing camera surveillance systems.

There have been many innovative techniques proposed in the past to build an accurate fire detection system which are broadly based on image processing and computer vision techniques. The state-of-the-art vision-based techniques for fire and smoke detection have been comprehensively evaluated and compared in [21]. The colour analysis technique has been widely used in the literature to detect and analyse fire in images and videos [2, 13, 16, 20]. On top of colour analysis, many novel methods have been used to extract high level features from fire images like texture analysis [2], dynamic temporal analysis with pixel-level filtering and spatial analysis with envelope decomposition and object labelling [22], fire flicker and irregular fire shape detection with wavelet transform [20], etc. These techniques give adequate performance but are outperformed by Machine Learning techniques. A comparative analysis between colour-based models for extraction of rules and a Machine Learning algorithm is done for the fire detection problem in [19]. The machine learning technique used in [19] is Logistic Regression which is one of the simplest techniques in Machine Learning and still outperforms the colour-based algorithms in almost all scenarios. These scenarios consist of images containing different fire pixel colours of different intensities, with and without smoke.

Instead of using many different algorithms on top of each other to extract relevant features, we can use a network that learns relevant features on its own. Neural networks have been successfully used in many different areas such as Natural Language Processing, Speech Recognition, Text Analysis and especially Image Classification. Extracting relevant features from images is the key to accurate classification and analysis which is why the problem of fire detection is ideally suited for Deep Learning. Deep Neural Networks are used to automatically 'learn' hierarchy of pertinent features from data without human intervention and the type of neural network ideally suited for image classification is the Convolutional Neural Networks (CNN).

Therefore, our approach is to employ state-of-the-art CNNs to distinguish between images that containing fire and images that do not and build an accurate fire detection system. To make these models more robust, we use a custom-made image dataset containing images with numerous scenarios.

The rest of paper is organised in the following manner: Section 2 briefly describes the previous research that uses CNNs for detecting fire. In Section 3 give a description of our proposed work. Section 4 gives the experimental results along with an illustration of our dataset, which is available online for the research community. Finally, Section 5 concludes our paper.

2 Related Work

There have been many significant contributions from various researchers in developing a system that can accurately detect fire in the surrounding environment. But, the most notable research in this field involves Deep Convolutional Neural Networks (DCNN). DCNN models are currently among the most successful image classification models which makes them ideal for a task such as Fire detection in images. This has been demonstrated by previous research published in this area.

In [5], the authors use CNN for detection of fire and smoke in videos. A simple sequential CNN architecture, similar to LeNet-5 [11], is used for classification. The authors quote a testing accuracy of 97.9% with a satisfactory false positive rate.

Whereas in [23], a very innovative cascaded CNN technique is used to detect fire in an image, followed by fine-grained localisation of patches in the image that contain the fire pixels. The cascaded CNN consists of AlexNet CNN architecture [10] with pre-trained ImageNet weights [15] and another small network after the final pooling layer which extracts patch features and labels the patches which contain fire. Different patch classifiers are compared.

The AlexNet architecture is also used in [18] which is used to detect smoke in images. It is trained on a fairly large dataset containing smoke and non-smoke images for a considerably long time. The quoted accuracies for large and small datasets are 96.88% and 99.4% respectively with relatively low false positive rates.

Another paper that uses the AlexNet architecture is [12]. This paper builds its own fire image and video dataset by simulating fire in images and videos using Blender. It adds fire to frames by adding fire properties like shadow, fore-ground fire, mask etc. separately. The animated fire and video frames are composited using OpenCV [1]. The model is tested on real world images. The results show reasonable accuracy with high false positive rate.

As opposed to CNNs which extract features directly from raw images, in some methods image/video features are extracted using image processing techniques and then given as input to a neural network. Such an approach has been used in [4]. The fire regions from video frames are obtained by threshold values in the HSV colour space. The general characteristics of fire are computed using these values from five continuous frames and their mean and standard deviation is given as input to a neural network which is trained using back propagation to identify forest fire regions. This method performs segmentation of images very accurately and the results show high accuracy and low false positive rates.

In [8], a neural network is used to extract fire features based on the HSI colour model which gives the fire area in the image as output. The next step is fire area

segmentation where the fire areas are roughly segmented and spurious fire areas like fire shadows and fire-like objects are removed by image difference. After this the change in shape of fire is estimated by taking contour image difference and white pixel ratio to estimate the burning degree of fire, i.e. no-fire, small, medium and large. The experimental results show that the method is able to detect different fire scenarios with relatively good accuracy.
All the research work done in this area has been exemplary. But, there are some issues associated with each of them that we try to alleviate in this paper. We use a dataset that consists of images that we have handpicked from the internet. The dataset contains images that are extremely hard to classify which results in poor generalization. The dataset also contains many different scenarios and is highly unbalanced to replicate real world behaviour. In this paper, we propose to use state-of-the-art pre-trained DCNN models. The reason behind using such complex models is explained in the next section. We also modify these models to improve accuracy at the cost of training time.

3 The Fire Detector

In this paper, we propose to employ Deep Convolutional Neural Networks instead of simple and shallow CNN models. The AlexNet has been used by researchers in the past for fire detection which has produced satisfactory results. We propose to use two Deep CNN architectures that have outperformed the AlexNet on the ImageNet dataset, namely VGG16 [17] and Resnet50 [7]. We use these models with pre-trained ImageNet weights. This helps greatly when there is lack of training data. So, we just have to fine-tune the fully-connected layers on our dataset.

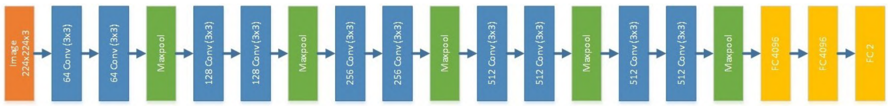
3.1 Deep ConvNet Models

The Convolutional Neural Network was first introduced in 1980 by Kunihiko Fukushima [6]. The CNN is designed to take advantage of two dimensional structures like 2D Images and capture local spatial patterns. This is achieved with local connections and tied weights. It consists of one or more convolution layers with pooling layers between them, followed by one or more fully connected layers, as in a standard multilayer perceptron. CNNs are easier to train compared to Deep Neural Networks because they have fewer parameters and local receptive fields.
In CNNs, kernels/filters are used to see where particular features are present in an image by convolution with the image. The size of the filters gives rise to locally connected structure which are each convolved with the image to produce feature maps. The feature maps are usually subsampled using mean or max pooling. The reduction is parameters is due to the fact that convolution layers share weights. The reason behind parameter sharing is that we make an assumption, that the statistics of a patch of a natural image are the same as any other patch of the image, which suggests that features learned at a location can also be learned for

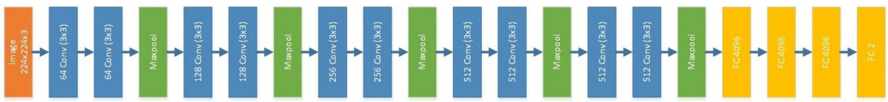
other locations. So, we can apply this learned feature detector anywhere in the image. This makes CNNs ideal feature extractors for images.
The CNNs with many layers have been used for various applications especially image classification. In this paper, we use two state-of-the-art Deep CNNs that have achieved one of the lowest errors in image classification tasks.

VGG16: The VGG16 architecture was proposed by the Visual Geometry Group at the University of Oxford [17]. The main purpose of the paper was to investigate the effect of depth in CNN models. They developed a number of models with different depths ranging from 11 layers to 19 layers and tested them on different tasks. The results on these tasks show that increasing depth also increases performance and accuracy. The 19 layer architecture, VGG19 won the ImageNet challenge in 2014, but the 16 layer architecture, VGG16 achieved an accuracy which was very close to VGG19. Both the models are simple and sequential. The 3x3 convolution filters are used in the VGG models which is the smallest size and thus captures local features. The 1x1 convolutions can be viewed as linear transformations and can also be used for dimensionality reduction. We choose the VGG16 over the VGG19 because it takes less time to train and the classification task in hand is not as complex as ImageNet challenge. Both the models have the same number of fully connected layers, i.e. 3, but differ in the number of 3x3 filters.

VGG16 (modified): In this work, we also test a modified version of VGG16 which consists of 4 fully connected layers, fine-tuned on the training data, which was able to increase the accuracy of classification. We also tested with more fully connected layers but the increase in accuracy was overshadowed by the increase in training time. The figures 1(a) and 1(b) show the original and modified VGG16 architectures respectively.



(a) VGG16 Architecture

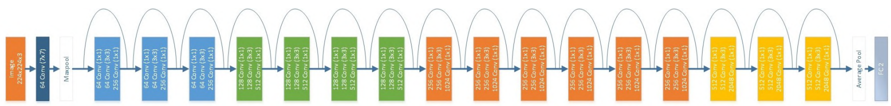


(b) Modified VGG16 Architecture

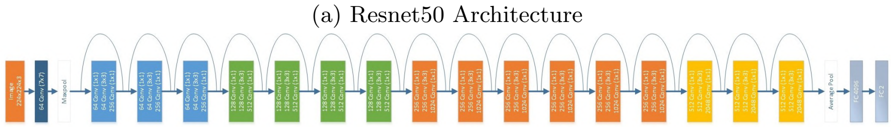
Fig. 1

Resnet50: After the success of the VGG architectures, it was established that deeper models outperform shallower networks. But, the problem with making models deeper was the difficulty in training them because model complexity increases as the number of layers increase. This issue was addressed by Microsoft Research, who proposed extremely deep architectures but with lower complexity [7]. They introduced a new framework of learning to ease training of such deep networks. This is called Residual learning and hence the models that employed this framework are called Residual Networks. Residual Learning involves learning residual functions. If a few stacked layers can approximate a complex function, $F(x)$ where, x is the input to the first layer, then they can also approximate the residual function $F(x) - x$. So, instead the stacked layers approximate the residual function$G(x) = F(x) - x$, where the original function becomes $G(x) + x$. Even though both can capable of approximating the desired function, the ease of training with residual functions is better. These residual functions are forwarded across layers in the network using identity mapping shortcut connections. The ImageNet 2015 results show that Resnet achieves the lowest error rates in image classification. The Resnet architectures consist of networks of various depths: 18-layers, 34-layers, 50-layers, 101-layers and 152-layers. We choose the architecture with intermediate depth, i.e. 50 layers. The Resnet consists of 3x3 and 1x1 filters, pooling layers and residual connections and a single softmax layer at the end.

Resnet50 (modified): We also test a modified Resnet model by adding a fully connected layer fine-tuned on the training data, which increase accuracy further. We did not add any more fully connected layers since the model is already quite deep and takes a long time to train. The figures 2(a) and 2(b) show the original and modified Resnet50 architectures respectively.



(a) Resnet50 Architecture



(b) Modified Resnet50 Architecture

Fig. 2

4 Experiments

We conducted our experiments to compare training and testing accuracies and execution times of the VGG16 and Resnet50 models including modifications.

We also trained a simple CNN which is used in [5] and compare with much deeper models to show why deeper and more complex models are necessary for fire detection on our dataset. We also train the modified VGG16 and Resnet50 models and compare the performance. We used pre-trained Keras [3] models and fine-tuned the fully-connected layers on our dataset. The training of the models was done on the following hardware specifications: Intel i5 2.5GHz, 8GB RAM and Nvidia Geforce GTX 820 2GB GPU. Each model was trained on the dataset for 10 training epochs with the ADAM optimizer [9] with default parameters $\alpha = 0.001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\epsilon = 10^{-8}$. The details of the dataset are given in the next subsection.

4.1 The Dataset

Since there is no benchmark dataset for fire detection in images, we created our own dataset by handpicking images from the internet. [1]This dataset consists of 651 images which is quite small in size but it enables us to test the generalization capabilities and the effectiveness and efficiency of models to extract relevant features from images when training data is scarce. The dataset is divided into training and testing sets. The training set consists of 549 images: 59 fire images and 490 non-fire images. The imbalance is delibrate to replicate real world situations, as the probability of occurrence of fire hazards is quite small. The datasets used in previous papers have been balanced which does not imitate the real world environment. The testing set contains 102 images: 51 images each of fire and non-fire classes. As the training set is highly unbalanced and the testing set is exactly balanced, it makes a good test to see whether the models are able to generalize well or not. For a model with good accuracy, it must be able to extract the distinguishing features from the small amount of fire images. To extract such features from small amount of data the model must be deep enough. A poor model would just label all images as non-fire, which is the case shown in the results.

Apart from being unbalanced, there are a few images that are very hard to classify. The dataset contains images from all scenarios like fire in a house, room, office, forest fire, with different illumination intensity and different shades of red, yellow and orange, small and big fires, fire at night, fire in the morning; non-fire images contain a few images that are hard to distinguish from fire images like a bright red room with high illumination, sunset, red coloured houses and vehicles, bright lights with different shades of yellow and red etc.

The figures 3(a) to 3(f) show the fire images with different environments: indoor, outdoor, daytime, nighttime, forest fire, big and small fire. And the figures 4(a) to 4(f) show the non-fire images that are difficult to classify. Considering these characteristics of our dataset, detecting fire can be a difficult task. We have made the dataset available online so that it can be used for future research in this area.

[1] The dataset is available here: `https://github.com/UIA-CAIR/Fire-Detection-Image-Dataset`

(a)

(b)

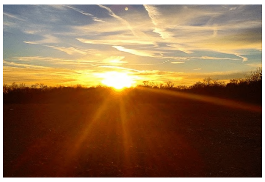
(c)

(d)

(e)

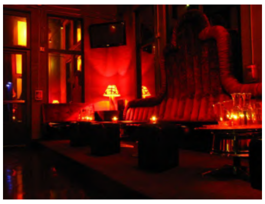
(f)

Fig. 3: Examples of Fire Images



(a)

(b)



(c)

(d)

(e)

(f)

Fig. 4: Examples of Non-Fire Images that are difficult to classify

4.2 Results

Table 1. shows the results of our experiments. The simple CNN model labels all images as non-fire which means that it is unable to extract relevant features from the dataset and cannot handle unbalanced datasets, which we can see from the training accuracy which is exactly equal to the percentage of non-fire images in the training set. So, the simple CNN overfits on the majority class of the unbalanced training dataset. Since, the training and fine-tuning methods for all models used here are the same, at the end it comes down to the architecture of the model. This justifies the use of deeper models like VGG16 and Resnet50. The simple CNN tested on our dataset is similar to the one used in [5]. The deep models achieve testing accuracy greater than 90%. And, the modified VGG16 and Resnet50 models outperform the base models by a small margin with slightly higher training time. It seems obvious that adding fully-connected layers to a network would increase accuracy. But on such a small dataset, the trade-off between accuracy and training time is quite poor, so we stop after adding just one fully connected layer. We also tested for more fully-connected layers(which is feasible since the model is pre-trained) but the increase in accuracy compared to increase in training time was too small.

Table 1: Comparison between CNN models

Model	Training accuracy	Training time (in sec)	Testing accuracy	Testing time (in sec)
VGG16	100	7149	90.19	121
VGG16 (modified)	100	7320	91.18	122
Resnet50	100	15995	91.18	105
Resnet50 (modified)	100	16098	92.15	107
Simple CNN [5]	89.25	112	50.00	2

Overall, the deep models perform well on this dataset. This shows that these models generalize well even when there is lack of training data. This means that if we want to slightly alter what the model does, we do not require large amount of data for retraining.

5 Conclusion

In this paper, we have proposed to use two state-of-the-art Deep Convolutional Neural Networks for fire detection in images, VGG16 and Resnet50. We test these models on our dataset which is made specifically to replicate real world environment. The dataset includes images that are difficult to classify and is highly unbalanced by including less fire images and more non-fire images since fire is a rare occurrence in the real world. We rationalize the use of such deep and complex models by showing that a simple CNN performs poorly on our dataset.

To further increase accuracy, we added fully connected layers to both VGG16 and Resnet50. Results show that adding fully connected layers does improve the accuracy of the detector but also increases its training time. In practice, increasing the number of fully connected layers by more than one results in minute increase in accuracy compared to the large increase in training time, even if the models are pre-trained. To conclude, we found that deep CNNs provide good performance on a diverse and highly imbalanced dataset of small size, with Resnet50 slightly outperforming VGG16 and adding fully connected layers slightly improves accuracy but takes longer to train.

References

1. G. Bradski. Opencv. *Dr. Dobb's Journal of Software Tools*, 2000.
2. Daniel Yoshinobu Takada Chino, Letricia P. S. Avalhais, José Fernando Rodrigues Jr., and Agma J. M. Traina. Bowfire: Detection of fire in still images by integrating pixel color and texture analysis. *CoRR*, abs/1506.03495, 2015.
3. Francois Chollet. Keras, 2015.
4. J. Zhao Z. Zhang C. Qu Y. Ke D. Zhang, S. Han and X. Chen. Image based forest fire detection using dynamic characteristics with artificial neural networks. In *2009 International Joint Conference on Artificial Intelligence*, pages 290–293, April 2009.
5. S. Frizzi, R. Kaabi, M. Bouchouicha, J. M. Ginoux, E. Moreau, and F. Fnaiech. Convolutional neural network for video fire and smoke detection. In *IECON 2016 - 42nd Annual Conference of the IEEE Industrial Electronics Society*, pages 877–882, Oct 2016.
6. Kunihiko Fukushima. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, 36(4):193–202, 1980.
7. Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.
8. Wen-Bing Horng and Jian-Wen Peng. Image-based fire detection using neural networks. In *JCIS*, 2006.
9. Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014.
10. Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012.
11. Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, Nov 1998.
12. Bc. Tomas Polednik. Detection of fire in images and video using cnn. *Excel@FIT*, 2015.
13. K. Poobalan and S.C. Liew. Fire detection algorithm using image processing techniques. In *3rd International Conference on Artificial Intelligence and Computer Science (AICS2015)*, Ocotober 2015.
14. Richard Bright Richard Custer. Fire detection: The state of the art. *NBS Technical Note, US Department of Commerce*, 1974.

15. Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015.
16. Jing Shao, Guanxiang Wang, and Wei Guo. An image-based fire detection method using color analysis. In *2012 International Conference on Computer Science and Information Processing (CSIP)*, pages 1008–1011, Aug 2012.
17. Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014.
18. C.Tao, J.Zhang, and P.Wang. Smoke detection based on deep convolutional neural networks. In *2016 International Conference on Industrial Informatics - Computing Technology, Intelligent Technology, Industrial Information Integration (ICIICII)*, pages 150–153, Dec 2016.
19. Tom Toulouse, Lucile Rossi, Turgay Celik, and Moulay Akhloufi. Automatic fire pixel detection using image processing: a comparative analysis of rule-based and machine learning-based methods. *Signal, Image and Video Processing*, 10(4):647–654, 2016.
20. B.Ugur Toreyin, Yigithan Dedeoglu, Ugur Gudukbay, and A.Enis Cetin. Computer vision based method for real-time fire and flame detection. *Pattern Recognition Letters*, 27(1):49 – 58, 2006.
21. Steven Verstockt, Peter Lambert, Rik Van de Walle, Bart Merci, and Bart Sette. State of the art in vision-based fire and smoke dectection. In Heinz Luck and Ingolf Willms, editors, *International Conference on Automatic Fire Detection, 14th, Proceedings*, volume 2, pages 285–292. University of Duisburg-Essen. Department of Communication Systems, 2009.
22. Jerome Vicente and Philippe Guillemant. An image processing technique for automatically detecting forest fire. *International Journal of Thermal Sciences*, 41(12):1113 – 1120, 2002.
23. Qingjie Zhang, Jiaolong Xu, Liang Xu, and Haifeng Guo. Deep convolutional neural networks for forest fire detection. February 2016.