https://doi.org/10.2352/ISSN.2470-1173.2019.5.MWSF-530

Deep Learning Methods for Event Verification and Image Repurposing Detection

M.Goebel*; University of California, Santa Barbara; Santa Barbara, CA
A. Flenner*; NAVAIR; China Lake, CA
L. Nataraj, B.S. Manjunath; Mayachitra Inc.; Santa Barbara, CA
'*'-equal contribution

Abstract

The authenticity of images posted on social media is an issue of growing concern. Many algorithms have been developed to detect manipulated images, but few have investigated the ability of deep neural network based approaches to verify the authenticity of image labels, such as event names. In this paper, we propose several novel methods to predict if an image was captured at one of several noteworthy events. We use a set of images from several recorded events such as storms, marathons, protests, and other large public gatherings. Two strategies of applying pre-trained Imagenet network for event verification are presented, with two modifications for each strategy. The first method uses the features from the last convolutional layer of a pre-trained network as input to a classifier. We also consider the effects of tuning the convolutional weights of the pre-trained network to improve classification. The second method combines many features extracted from smaller scales and uses the output of a pre-trained network as the input to a second classifier. For both methods, we investigated several different classifiers and tested many different pre-trained networks. Our experiments demonstrate both these approaches are effective for event verification and image re-purposing detection. The classification at the global scale tends to marginally outperform our tested local methods and fine tuning the network further improves the results.

Introduction

Today social media websites are emerging as a dominant news source, but verifying the validity of the news stories is a difficult problem. In the few months before the 2016 US Presidential elections, the average American saw at least one fake news story, and of those who saw one, half of them believed it to be true [1]. In practice, countering these sources of fake news is a complex problem. There is little entry cost to distributing false information on Facebook, with large potential for ad revenue if the story gains popularity. While in the past people relied on a few well known sources, the website publishing false stories is often removed before being identified as illegitimate [1]. For these reasons, an automated algorithm is needed to identify these fake stories before they reach a large number of users.

A common approach to distribute false information is to select an authentic image from some previously recorded event which convincingly supports their message, and re-brand it as a current story. For example, during the times of storms and hurricanes, images from previous hurricanes are usually re-purposed and uploaded in social media to create a scare. If we have a database of previously recorded events, then we will be able to verify those images that have been re-purposed from older events. In this paper we investigate several methods to automatically identify images for event verification and image repurposing detection. We explore a transfer learning approach [2] to event verification by using a network pre-trained on the ImageNet dataset [3], but instead of using the network for image classification we use the network for event verification . After observing many images from different events, certain features may stand out to distinguish between two similar classes. Similar locations, architectures, or identifying symbols may be associated with each. For example, marathon race bibs are generally consistent for all participants in a single event, and different from other races. In this paper, we outline two approaches of applying pre-trained ImageNet network for event verification, one at the global image level and other at the local image level. At the global level, we compare the effect of fine tuning a pre-trained network to a particular dataset to a method that is not tuned to any dataset. At the local level, we explore the effects of spatial context at smaller scales using one method that does averaging and another method without averaging. Figure. 1 provides a summary of the methods. Our experiments on several datasets show that both approaches are promising for event verification and image re-purposing detection.

Related Work

While there are several works on image classification, retrieval, event classification, and more, very few works address the problem of event verification. At the time of writing, we are aware of two recent works that are related to this problem [4, 5]. The first considers pairs of images and captions[4], and then trains a network to decide how consistent the caption is with the associated image. The second work [5] altered GPS coordinates, captions, and the actual image pixels for image re-purposing detection. In contrast to these methods, our approach operates only at the image pixel level and does not need any metadata, which may not be always available at hand.

We approach the event verification problem as a transfer learning classification problem and distinguish between a finite set of events by using pre-trained networks. A recent paper [2] systematically tested how well features from different architectures transfer to other classification tasks. Their tests showed that when comparing two different architectures, ImageNet performance was only weakly correlated with performance on another task if the network is fixed. However, this effect diminished once the network was fine-tuned [2].

Our methods also do not address the possibility of modifica-

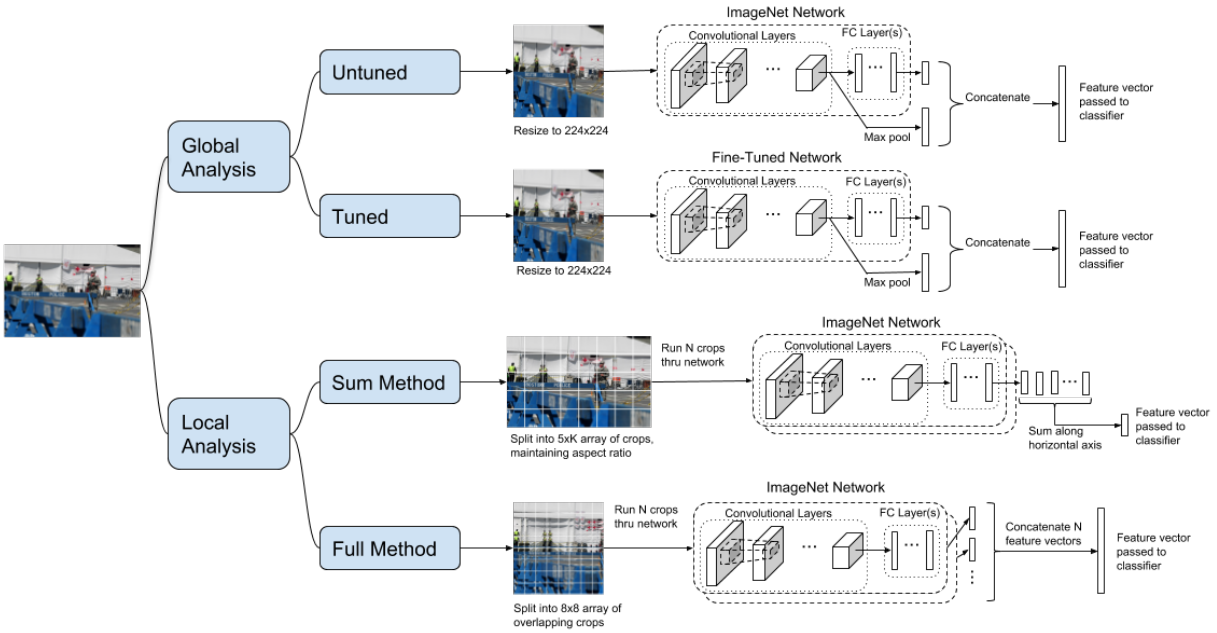


Figure 1: Block schematic of proposed approach: the two broad methods are based on analyzing an image at the global and local level. Within each of these categories, we further analyzed two approaches, one that considers fine-tuning to a dataset, and the other that analyzes the effect of spatial context.

tion to image pixels. This is a well studied problem, with many viable solutions [6, 7, 8, 9] Our models are created under the assumption that they will be used with one of these image manipulation detectors, and that all images given to our detector are unmodified.

Event Verification and Image Repurposing Detection

In this paper, our goal is to demonstrate an approach to verify if an image was taken at the claimed event. We explored two different approaches where one approach used global image features and another used local image features. More specifically, we treated event verification as a transfer learning problem. A deep learning classifier can be represented as a composition of the two function $f(\cdot) \in R^{N\times K}$ and $g(\cdot) \in R^{K\times M}$ where $f(\cdot)$ consists of the convolutional, non-linear, and pooling layers while $g(\cdot)$ is the fully connected classification layer with M possible classes (see Figure 1). During the training phase, the parameters of $f(\cdot)$ and $g(\cdot)$ were learned using a large image database such as ImageNet [3].

We investigated two transfer learning strategies that consisted of using the features from the pre-trained network and learning a new classifier using these features. The first strategy, or global method, extracted features by mapping the entire image using either $f(\cdot)$, $g(f(\cdot))$ or both. The second local method extracted features by mapping image patches using $g(f(\cdot))$, and the output from the patches were then averaged or concatenated into one large feature vector. A classifier was then trained on the extracted features to identify the respective events. Finally, for the global method we also tested allowing the parameters of $f(\cdot)$ to be fine tuned using the event data training set. In total, there were four different strategies as illustrated in Figure 1.

Global Method

The global method consists of a two part structure. The first section uses deep convolutional neural networks (CNNs) to extract features from each image. The second is a separate classifier which takes these features as input.

Untuned model: The untuned network uses standard models such as ResNet-50 that are trained on ImageNet, and not tuned to the dataset under test. The images are first re-sized to the native resolution used by the CNN (either 224x224 or 299x299). The final feature vector is derived from the outputs from the last convolutional layer and the final output layer. This process replaces a large image with a much smaller feature vector, while hopefully retaining as much relevant information as possible. Using these features, we investigated several classifiers such as extra trees, random forests, nearest neighbors, support vector machines, and convolutional nets. The three variables tested were: the CNN used to extract features, which layers were used as features, and the final classifier algorithm.

Tuned model: We also investigated the effect of fine-tuning the CNN. A recent work suggested using at least 800 training images that contained at least 32 images per class [2]. In our case we trained on 800 images with 200 per class, which is close to their minimum for total number of images. The fine tuning was done in Keras, by removing the last fully connected layer of dimension 1000 used for ImageNet and replacing it with a fully con-

(a) (b) (c) (d)

Figure 2: Examples of images included in Ver_1 subset of the Event Verification dataset comprising the following events: (a) Austin Marathon, (b) Boston Marathon, (c) Occupy Baltimore, (d) Occupy Portland

nected layer with 4 outputs. An Adam optimizer with learning rate of 0.0001 was run and the training data was iterated through 10 times. Rates slower than 0.0001 did not show any improvement in validation accuracy while taking longer to converge. At our chosen rate, performance generally plateaued after 10 epochs.

Local Method

The local method differs from the global method in two key areas. First, the local method uses a sliding window to extract features from local image patches. Second, the local method derives its feature vector from the output of the fully connected classification layer. The goal with the local method is to capture more image details through the aggregation of information from local image patches.

Sum Features: In order to deal with different image sizes, we rescaled each image to have 1120 rows while preserving the aspect ratio of the original image. We then divided the image into overlapping 224x224 patches. Each block was processed through the entire ResNet-50 classifier [10]. The output vectors for each patch were summed and the resulting sum was normalized to one, which produced a 1000 dimensional feature vector. This procedure extracted a feature vector of constant length from each image. A classifier was then used to identify the event. Several classifiers were considered including support vector machines, extra trees, random forests, and xgboost [11].

The sum feature vectors are similar to the bag-of-words image classification model [12]. In particular, the output of each patch can be interpreted as a distribution over classes, which is a 1000 dimensional vector of words or phrases. The words for each patch are then summed, generating a bag of words. The original bag-of-words model would identify one word per patch, however our approach outputs a probability distribution of words for each patch and we sum the different probability distribution to generate our feature vector.

Full Features: The sum features average the information gained from all the patches and removes any spatial context. As a simple test to investigate the importance of the spatial knowledge, we removed the sum over the patches. In order to obtain feature vectors that had consistent dimensions, we rescaled each image to have 1120 rows and 1120 columns. ResNet-50 and a sliding window was used to produce a feature vector for each 224x224 block with an overlap of 100 pixels. The final output was the concatenation of the output of all the feature vectors. The same classifiers were tested as in the sum features.

Experiments and Results

Here we will detail the datasets used in our experiments, the results on the global and local methods and a comparison of both the methods. For each method, a receiver operating curve (ROC) was obtained, and the area under this curve (AUC) is used as the primary metric for comparison.

Event Verification (EV) Dataset

The classification procedures were tested on the Event Verification dataset generated by NIST as part of the DARPA Media Forensics (MediFor) project. This dataset contained three different subsets, each already divided into training and testing. We refer to them as Ver_1, Ver_2 and Ver_{eval}. The first two versions each had 4 events, with 200 training images for each event, and 100 testing images per event. Ver_1 subset contained images associated with the Boston Marathon, Austin Marathon, Occupy Portland, and Occupy Baltimore (see Figure 2). Ver_2 had images from Hurricane Sandy, Hurricane Matthew, the Oshkosh Air Show, and Berlin Air Show. The third set was held out for evaluation (Ver_{eval}). This subset had 12 events, with 200 images per event, and 600 total test images with held out labels. Testing on Ver_{eval} subset was done by submission of results to a NIST server, which in turn returned the ROC curve. Ver_1, Ver_2 subsets were used to test across a wide number of models, while Ver_{eval} subset was reserved to see how well these models generalized.

Results on Global Method

For the global method, many combinations of ImageNet CNNs, feature extraction locations, and standard machine learning classifiers were tested. In general, using only intermediate layer features performed better than only considering output layer features. Using both gave a slight boost for some classifiers though the difference was insignificant for the best classifiers. The results presented for the rest of this section are from the intermediate layer. For most cases, ResNet-50 had the best performance, which is consistent with the results from the recent work on transfer learning [2]. For a network strictly trained on ImageNet, the classifier at the end had a strong impact on performance. Random Forest and Extra Trees classifiers performed the best in most cases. On Ver_1, the fine-tuned ResNet universally outperformed the top untuned method. For Ver_2 the results were less clear as AUC scores were close. At low false alarm rates the tuned method performs better, while the untuned method has better detection at high false alarm rates. AUC is slightly higher for the untuned case overall and per class ROC curves are shown in

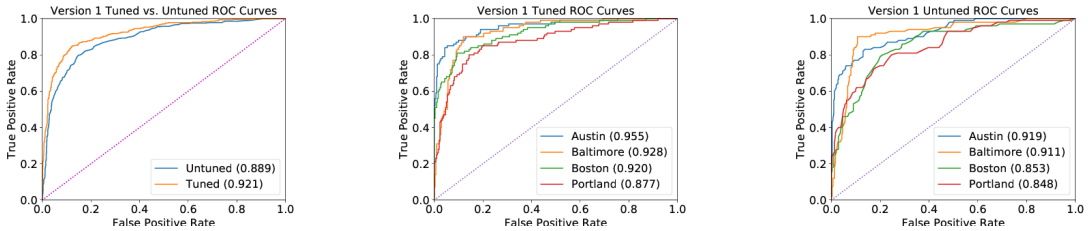


Figure 3: ROC curves using the global features and tested on the Ver_1 subset.

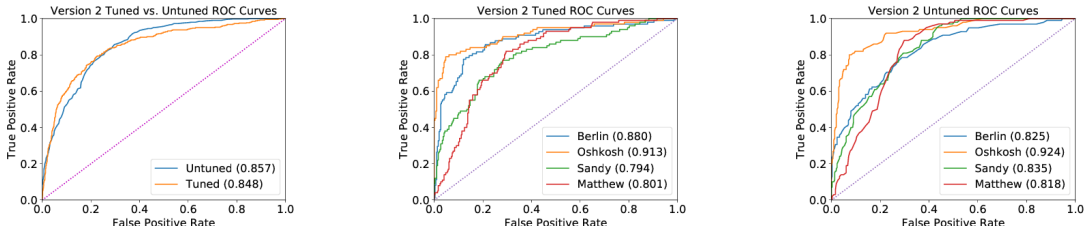


Figure 4: ROC curves using the global features and tested on the Ver_2 subset.

Figure 3 for Ver_1 subset and Figure 4 for Ver_2 subset.

After fine tuning, the classifier at the end seemed to have less of an impact. The best performing of the fine-tuned classifiers were simply the dense output layer generated during the tuning process. Using a different classifier in the end may not have much benefit. However, distance based classifiers do have the ability to provide training images which it found to be similar to the query image, and can be helpful in some applications where justification is needed. More detailed experiments on the effect of different classifiers on various pre-trained networks are presented later in Table. 3.

Results on Local Method

Based on the results of the global method, only the ResNet-50 network trained using ImageNet was investigated. The classifiers tested were extra trees, random forests, support vector machines, one nearest neighbor, and xgboost which are summarized for the Ver_1 subset in Table 1. The ensemble methods of extra trees and random forest performed the best for the sum features and were statistically equivalent. For the full features, the extra trees, random forest, and support vector machines had equivalent results. The extra trees and random forest ensemble classifiers performed better on the sum features than the full features, while the support vector machine results was equivalent for the sum and full features. As with the global method, due to random initialization, the results using the same classifier would fluctuate by 1%. The local method was tested on the Ver_1, Ver_2 and Ver_{eval} subsets of the Event Verification dataset. For all the classifiers except xgboost, the python package scikit-learn version 0.19.1 was used and the default settings obtained the best performance. The python package xgboost was used for the xgboost implementation with a max depth of two and binary logistic objective. The ROC curves for this method are shown in Figures 5 and 6 for the Ver_1 and Ver_2 subsets.

Table 1: Local method AUC results for different classifiers and different features on the Ver_1 subset. From left to right the classifiers are extra trees, random forest, 1 nearest neighbor with Euclidean distance, support vector machines, and xgboost.

Features	ET	RF	1-NN	SVM	XGB
Local Sum	**0.885**	.882	0.74	0.855	.872
Local Full	**0.857**	0.851	0.612	0.852	.841

Comparison of Global and Local Method

The comparison of the global method and the local method on Ver_1, Ver_2 and Ver_{eval} subsets of the Event Verification dataset are summarized in Table 2 and Figure 7. On average, we can see that the global methods perform better than the local methods. Among the global methods, the tuned model obtained the highest AUC for Ver_1 and Ver_{eval} subsets while the untuned model performed slightly better than the tuned model for the Ver_2 subset. In general, we see that the tuned model would be the most preferable though it would come at the extra cost of tuning the model for every dataset. Next we analyzed the performance of individual events for the four methods. The results are summarized for both the Ver_1 and Ver_2 subsets in Figure. 7. The global tuned model performed the best for most events (Austin Marathon, Occupy Baltimore, Boston Marathon, Occupy Portland and Berlin Airshow), while the global untuned model performed the best for two events (Hurricane Sandy, Hurricane Matthew). While the global method tends to outperform the local method in most events, there are cases in which one may be preferred. In the one event where the local method outperformed the global one, Oshkosh Airshow, we hypothesize that there was more detail at the smaller scale. Also, for events such as Occupy Baltimore and Hurricane Sandy, there isn't any significant difference in the performance of the four methods. Another consideration is that all of the methods besides local sum resize the image to be square. If preserving the aspect ratio is important to a particular task, this may perform relatively better. Testing on several different datasets also showed that performance is highly data dependant. Given our results, we expect that the best out-of-the-box approach is either the global tuned or

Table 2: Comparison of AUC between Global and Local Method

Dataset	Global Untuned	Global Tuned	Local Sum	Local Full
Ver_1	0.889	**0.921**	0.885	0.857
Ver_2	**0.857**	0.848	0.831	0.798
Ver_{eval}	0.88	**0.89**	0.85	0.82

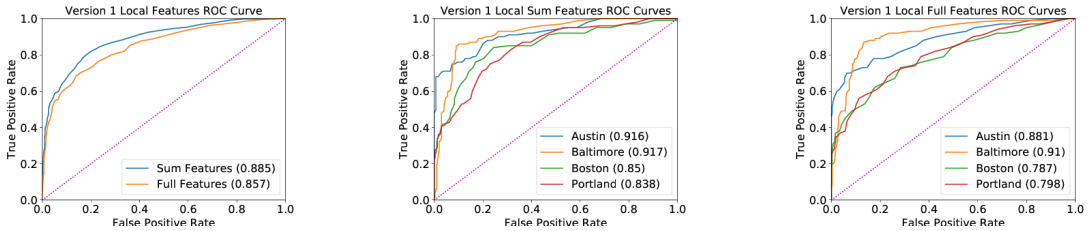


Figure 5: ROC curves using the local features and tested on Ver_1 subset.

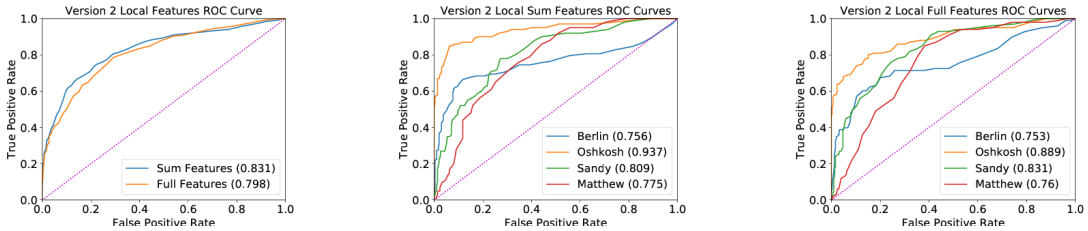


Figure 6: ROC curves using the local features and tested on the Ver_2 subset.

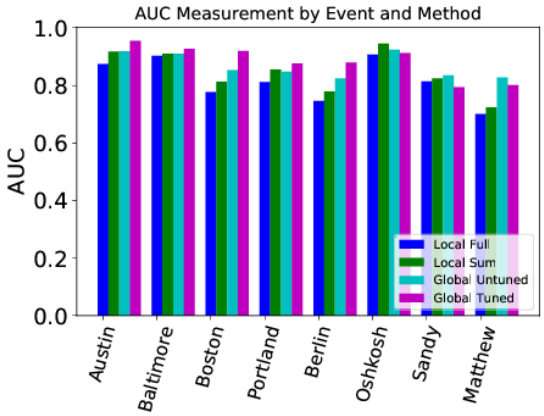


Figure 7: Comparison of AUCs on individual events on Ver_1 and Ver_2 subsets of the Event Verification Dataset.

global untuned.

Detailed experiments with Pre-trained models

Given the success of the global method over the local method, we performed a more exhaustive set of tests on the global method, summarized for Ver_1 subset in Table 3. Twelve different ImageNet classifiers were tested along with the fine-tuned ResNet50 network. For all networks, the output arrays from the last convolutional layer and the class output layer were saved. The classifiers given only the output layers have been omitted as they did not perform well compared to those given strictly the intermediate layer, or the intermediate and output layers concatenated. Then each combination of the previous two conditions was given to a classifier available in scikit-learn. With this common interface, over a dozen different classifiers with varying parameters were tested. Only a few are shown in Table 3. Based on these results, ResNet seemed to perform the best, which was consistent with the results in [2]. Based on these results, and those from Ver_2 subset (not shown due to lack of space), the fine-tuned ResNet50 network had the best performance overall.

Discussion and Future Work

There are many additional methods which can be used as an add-on to the end-to-end trainable network. For example, face matching was one method we tested with mixed results. On Ver_1 subset it gave a 1% boost in AUC, while it had little effect on Ver_2 or $\text{Ver}_e val$ subset. Given that Ver_1 subset contained marathons and protests, while Ver_2 subset contained storms and airshows, this result was generally expected. Text extraction may be another area of interest as a supplementary decision. In visually inspecting the data, many images contained text which was unique to the event. This would include street signs, advertisements, and in the case of marathon events, race bibs. In future work we will attempt to accommodate images in their native resolutions. The dataset tested here consisted of images from 0.077 MP to 30MP. Resizing all images into the same height and/or width eliminates much of the additional information that large images contain. A brute force solution may be to sweep the feature extractor across multiple scales of an image pyramid. This will then give a pyramid of features extracted at different scales, leaving the open question of how to combine these. The two approaches represented here are a subset of this, where only the largest scales are used, and all others effectively ignored. Finally, a collection of a larger dataset will facilitate in understanding how our methods will generalize in a real world application that may consist of hundreds of events and several thousand images.

Conclusions

This paper demonstrated the viability of several possible methods for applying pre-trained ImageNet classifiers to the problem of event verification. In particular, we explored analyzing an image at the global level and the local level, and studied the impact of fine-tuning a model for a specific dataset. While the global classification methods out-performed the local methods in our experiments, more data will be needed to confirm this hypothesis. The images included in our tested dataset contained less than a dozen events, and a broader range of data would clarify if there are certain types of events where one method may be superior. More research still needs to be done in this area, as it remains a pressing concern without a clear implementation at full scale.

Table 3: AUC results for various combinations of CNN, feature extraction locations, and end classifiers. The classifiers on the top row are: Extra Trees, Random Forest, 1 Nearest Neighbor (NN) with L1 distance, 1 NN with L2 distance, 2 NN with L2, 4 NN with L2.

Features only from intermediate layer

	ET	RF	L1	L2	2NN	4NN
Xception	0.766	0.753	0.641	0.638	0.675	0.695
VGG16	0.880	0.874	0.751	0.775	0.818	0.845
VGG19	0.883	0.874	0.756	0.768	0.820	0.838
ResNet50	0.889	0.886	0.79	0.781	0.839	0.877
InceptionV3	0.765	0.772	0.648	0.655	0.702	0.714
InceptionResNet	0.682	0.695	0.575	0.573	0.586	0.614
MobileNet	0.842	0.847	0.739	0.735	0.786	0.825
MobileNet2	0.887	0.876	0.766	0.753	0.805	0.843
DenseNet121	0.877	0.871	0.745	0.73	0.785	0.819
DenseNet169	0.869	0.870	0.748	0.745	0.787	0.810
DenseNet201	0.735	0.728	0.636	0.636	0.655	0.690
NASNetMobile	0.764	0.763	0.648	0.646	0.680	0.701
Tuned ResNet	0.909	0.905	0.845	0.831	0.867	0.876

Features from intermediate and output layers

	ET	RF	L1	L2	2NN	4NN
Xception	0.760	0.756	0.641	0.638	0.675	0.695
VGG16	0.879	0.876	0.751	0.775	0.818	0.845
VGG19	0.884	0.884	0.756	0.768	0.820	0.838
ResNet50	0.894	0.895	0.79	0.781	0.839	0.877
InceptionV3	0.762	0.769	0.648	0.655	0.702	0.714
InceptionResNet	0.681	0.690	0.575	0.573	0.586	0.614
MobileNet	0.843	0.837	0.739	0.735	0.786	0.824
MobileNet2	0.875	0.884	0.766	0.753	0.805	0.843
DenseNet121	0.876	0.872	0.745	0.73	0.785	0.819
DenseNet169	0.877	0.864	0.746	0.745	0.787	0.810
DenseNet201	0.725	0.719	0.636	0.636	0.655	0.690
NASNetMobile	0.770	0.770	0.648	0.646	0.680	0.701
Tuned ResNet	0.912	0.906	0.845	0.835	0.865	0.878

Acknowledgments

This research was developed with funding from the Defense Advanced Research Projects Agency (DARPA). The views, opinions and/or findings expressed are those of the author and should not be interpreted as representing the official views or policies of the Department of Defense or the U.S. Government. The paper is approved for public release, distribution unlimited.

References

[1] H. Allcott and M. Gentzkow, "Social media and fake news in the 2016 election," *Journal of Economic Perspectives* **31**(2), 211–36 (2017).

[2] S. Kornblith, J. Shlens, and Q. V. Le, "Do better imagenet models transfer better?," *arXiv preprint arXiv:1805.08974* (2018).

[3] J. Deng, W. Dong, R. Socher, *et al.*, "ImageNet: A Large-Scale Hierarchical Image Database," in *CVPR09*, (2009).

[4] A. Jaiswal, E. Sabir, W. AbdAlmageed, *et al.*, "Multimedia semantic integrity assessment using joint embedding of images and text," in *Proceedings of the 2017 ACM on Multimedia Conference*, 1465–1471, ACM (2017).

[5] E. Sabir, W. AbdAlmageed, Y. Wu, *et al.*, "Deep multimodal image-repurposing detection," in *2018 ACM Multimedia Conference on Multimedia Conference*, 1337–1345, ACM (2018).

[6] H. Farid, "Image forgery detection," *IEEE Signal processing magazine* **26**(2), 16–25 (2009).

[7] B. Mahdian and S. Saic, "A bibliography on blind methods for identifying image forgery," *Signal Processing: Image Communication* **25**(6), 389–399 (2010).

[8] X. Lin, J.-H. Li, S.-L. Wang, *et al.*, "Recent advances in passive digital image security forensics: A brief review," *Engineering* (2018).

[9] S. Walia and K. Kumar, "Digital image forgery detection: a systematic scrutiny," *Australian Journal of Forensic Sciences* , 1–39 (2018).

[10] K. He, X. Zhang, S. Ren, *et al.*, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778 (2016).

[11] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 785–794, ACM (2016).

[12] J. Yang, Y.-G. Jiang, A. G. Hauptmann, *et al.*, "Evaluating bag-of-visual-words representations in scene classification," in *Proceedings of the international workshop on Workshop on multimedia information retrieval*, 197–206, ACM (2007).

Author Biography

Michael Goebel *received his B.S. and M.S. degrees in Electrical Engineering from Binghamton University in 2016 and 2017. He is currently a PhD student in Electrical Engineering at University of California Santa Barbara.*

Arjuna Flenner *received his Ph.D. in Physics at the University of Missouri-Columbia located in Columbia MO in the year 2004. His major emphasis was mathematical Physics. After obtaining his Ph.D., Arjuna Flenner obtained a job as a research physicist for NAVAIR at China Lake CA. He won the 2013 Dr. Delores M. Etter Navy Scientist and Engineer award for his work on Machine Learning.*

Lakshmanan Nataraj *received his B.E degree in Electronics and Communications Engineering from Anna university in 2007, and the Ph.D. degree in the Electrical and Computer Engineering from the University of California, Santa Barbara in 2015. He is currently a Senior Research Staff Member at Mayachitra Inc., Santa Barbara, CA. His research interests include malware analysis and image forensics.*

B. S. Manjunath *(F05) received the Ph.D. degree in Electrical Engineering from the University of Southern California in 1991. He is currently a Distinguished Professor at the ECE Department at the University of California at Santa Barbara. He has co-authored about 300 peer-reviewed articles. His current research interests include image processing, computer vision and biomedical image analysis.*

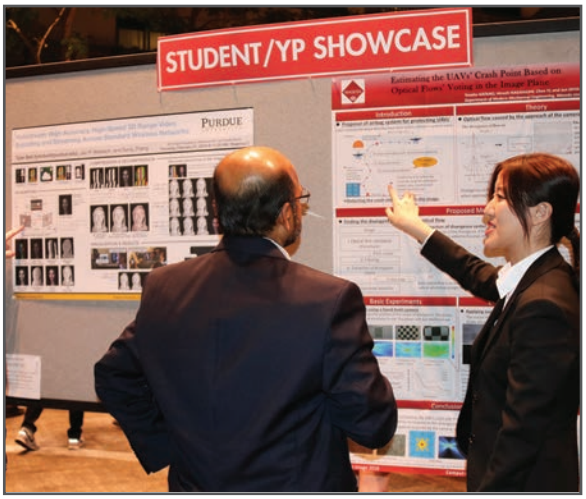
JOIN US AT THE NEXT EI!

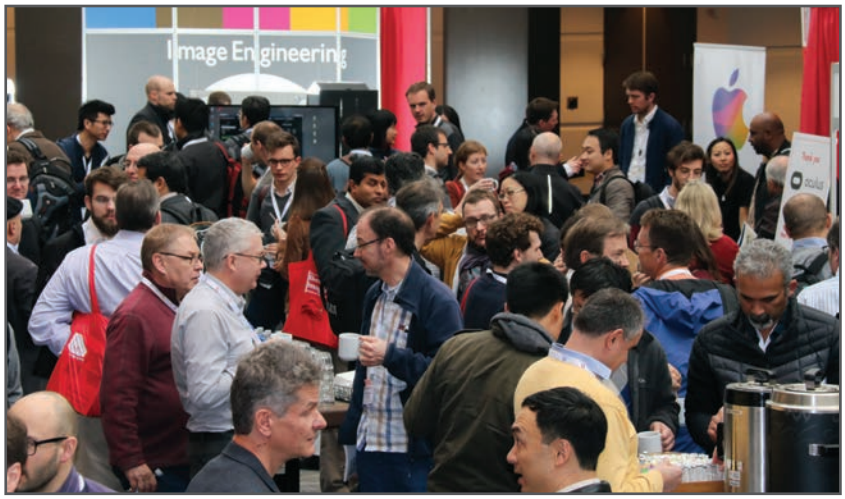
IS&T International Symposium on

Electronic Imaging

SCIENCE AND TECHNOLOGY

Imaging across applications . . . Where industry and academia meet!





• SHORT COURSES • EXHIBITS • DEMONSTRATION SESSION • PLENARY TALKS •
• INTERACTIVE PAPER SESSION • SPECIAL EVENTS • TECHNICAL SESSIONS •

www.electronicimaging.org

IS&T
imaging.org