AUGMENTED DATA AND IMPROVED NOISE RESIDUAL-BASED CNN FOR PRINTER SOURCE IDENTIFICATION

Sharad Joshi[⋆] *Mohit Lamba*[⋆] *Vivek Goyal*[†] *Nitin Khanna*[⋆]

[⋆] Indian Institute of Technology Gandhinagar (IITGN), Gujarat, India
[†] Microsoft Global Service Center (India) Pvt. Ltd.

ABSTRACT

With the classification revolution driven by convolutional neural networks (CNN), plenty of suitable CNN architectures are readily available. However, a limiting aspect of CNN is their hunger for data. Training data can be scarce in many forensics applications, particularly for printer forensics where hard copy pages need to be printed and then digitized using scanners. This paper aims to tackle this problem by using: 1) complementary representations of data and 2) augmented data variations. We derive a simple yet effective noise residual representation of a character image that complements the information contained in the original image. Further, rotated character variations and their monotonic grayscale transformations are used as augmented data. Since such variations are directly linked to printer mechanism, they help improve the overall classification accuracy. Finally, spatial pyramid pooling is used to accommodate characters of all sizes without compromising on a character's spatial information. The proposed method outperforms state-of-the-art CNN based method on a publicly available dataset, and its generic nature allows it to be combined with any other CNN architecture as well. Experiments indicate that for the case of very limited training data availability, the proposed method can achieve 2.5% increase in printer classification accuracy.

Index Terms— Source Printer Identification, Sensor Forensics, Intrinsic Signatures, Data Augmentation, CNN.

1. INTRODUCTION

Traditionally, a printer's *intrinsic signature* [1] is estimated using hand crafted features extracted from texture patterns in the scanned image of a printed document [2–10]. This signature results from imperfections in the mechanical parts of a printer like the optical photoconductor (OPC) drum and other gear mechanisms. The variation in texture patterns due to such imperfections could be very complex to model [11]. But, they have been shown to be characteristic of a particular brand (manufacturer) and model of a printer [12]. The signature of a printed document can be used to find its source printer which can give useful clues in a variety of forensic investigations. Further, source identification can also detect potentially forged documents in several scenarios.

Off late, CNN based data driven features for classification problems have shown tremendous potential [13]. But most architectures require a large amount of data to learn good quality features. One of the ways of increasing the training data is to use complementary data representation. Authors in [14] proposed using median and average residuals of character images to train multiple CNN in parallel. The other approach is to modify real world examples (data augmentation) by applying practically observed transformations. Data augmentation has to be performed in either data-space [15] or feature-space [16]. However, it is shown to work better in the data-space given that the label information is strictly preserved by the transformations [17].

In this work, we try to address the limitation on amount of data by using a noise residual and augmented data variations for each character image. In this paper, we compute the noise residual of an input character image directly from the original data. For data augmentation, the main intuition is to feed additional data that may be encountered in practical situations. The augmented data is derived by introducing intuitive intensity and rotation variations which are directly dependent on printer characteristics. The main advantages of the proposed method over state-of-the-art techniques are outlined by various experiments performed on a publicly available dataset [9]. In particular: 1) our algorithm uses a novel three level noise residual technique which can be combined with the existing CNN approach; 2) we confirm experimentally that the combination of original image and our noise residual consistently performs better than the state-of-the-art using the same train and test folds provided by them; 3) to the best of our knowledge, a first of its kind, augmented data based technique is introduced for printer forensics; and 4) the proposed method

This material is based upon work partially supported by the Board of Research in Nuclear Sciences (BRNS), Department of Atomic Energy (DAE), Government of India under the project DAE-BRNS-ATC-34/14/45/2014-BRNS, Indian Institute of Technology Gandhinagar internal research grant IP/IITGN/EE/NK/201516-06, and Visvesvaraya PhD Scheme, Ministry of the Electronics & Information Technology (MeitY), Government of India. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the funding agencies. Address all correspondence to Nitin Khanna at nitinkhanna@iitgn.ac.in.

achieves an increment of 2.5% in average classification accuracy for the same amount of original data.

2. PROPOSED METHOD

The pipeline of the proposed method has been outlined in Figure 1. While there exist many CNN architectures, their performance is, in general, limited by the amount and type of input data. The proposed method addresses two significant issues; 1) increasing training data and 2) utilizing complete character image. For this work, we utilize the CNN based method used in [14], but it can be easily used with any other architecture. The algorithm is as follows:

2.1. Pre-Processing

The pre-processing steps remain the same as in [14]. At first, printed documents obtained from all printers are scanned using a reference scanner. Then, characters 'e' and 'a' are extracted from it. We utilize the extracted 'e' characters provided by [9]. However, the above method does not seem to perform accurately with 'a'. As a result, there are many other characters present in the folder provided for 'a' [9]. So, we use MATLAB's inbuilt optical character recognition (OCR) function to extract 'a' and use them with our approach as well as existing method [14]. Note that 'e' were not extracted from OCR so as to keep most of the experimental conditions similar to those in [14].

2.2. Noise Residual Estimation

The proposed method introduces a new noise residual technique. This technique performs better than median or average filter residual as it contains information which is directly related to the printer's signature. On the other hand, CNN trained using filter residuals could not be guaranteed to carry a printer's signature as the filtering process might change the ground truth (printer artifact) itself. Computation of noise residual is a two-step procedure as explained below. To get an approximation of a character image printed without printer noise, we first divide the input character image I into three regions. For this, we utilize a strategy used in [10]. Here, an intensity histogram (65532 bins for 16-bit image) is generated for I. As it might contain rugged peaks, it is smoothened by a weighted mean filter of length 5. Then local maxima are used to find peaks in the smoothened intensity histogram. The first and last peaks (P_1 and P_2) are chosen, and their mean (μ) is computed which divides the character image into two levels based on their intensity. Further, to accommodate for intensities occurring at the boundary of characters, $\alpha\mu$ and $\beta\mu$ are chosen as two thresholds. Here, α and β are empirically selected constants. The group of pixels falling within the character is termed as part of flat region I_F, boundary pixels land

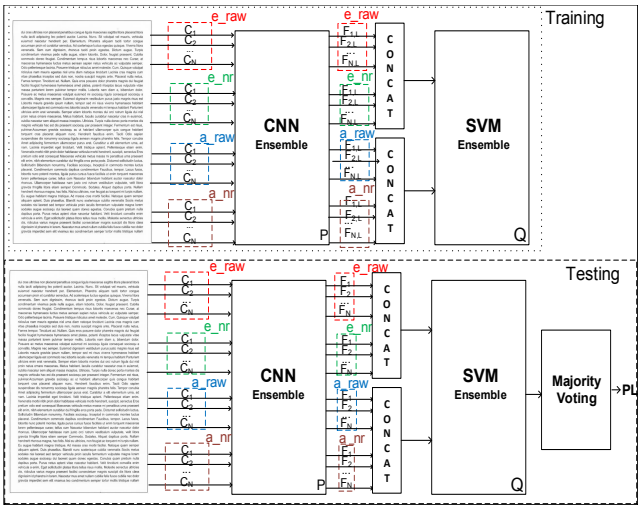

Fig. 1: Schematic representation of the proposed algorithm. Original character images (e_{raw}, a_{raw}) and their noise residuals (e_{nr}, a_{nr}) are used to train four separate CNN. The feature vectors extracted from the same type of characters ('e' and 'a') are concatenated and used to train SVM. The evaluation of test characters follow the same pipeline but use trained CNN and SVM models (P & Q respectively). Printer label (PL) for the whole test document is obtained by a majority vote taken on the predicted character labels. Noe that for experiments with data augmentation, 14 augmented versions, of each character representation, are also given as input to CNN.

in the edge region I_E and rest of the region outside a character comprises of background I_B [10]. Thus the value at any pixel p in the three regions is defined as,

$$F_I = \{I(p) \mid I(p) \in [0, \alpha\mu] \ \ and \ \ \forall p \in I\} \tag{1}$$

$$E_I = \{I(p) \mid I(p) \in (\alpha\mu, \beta\mu] \ \ and \ \ \forall p \in I\} \tag{2}$$

$$B_I = \{I(p) \mid I(p) \in (\beta\mu, max(I)] \ \ and \ \ \forall p \in I\} \tag{3}$$

Here, $max(I)$ denote the maximum intensity value in I. Next, I is converted into a tertiary (3-level) image and the intensity at any pixel p is given by,

$$I_{Ideal}(p) = \begin{cases} median(F_I), & I(p) \in (0, \alpha\mu] \\ median(E_I), & I(p) \in (\alpha\mu, \beta\mu] \\ median(B_I), & I(p) \in (\beta\mu, max(I)] \end{cases} \tag{4}$$

Here, $median(J)$ denotes the median of all intensity values of image J. At last, the noise residual (Figure 2) is obtained as,

$$I_{NR}(p) = I(p) - I_{Ideal}(p), \ \ \forall p \in I \tag{5}$$

2.3. Augmented Data Generation

In addition to noise residual, two types of data augmentations are applied on I to generate additional data. The first variation comprises of rotated character images. Characters in

a printed document have been observed to be skewed by up to $\pm 3\%$ when subjected to printing, photocopying, and then scanning [18]. Based on this observation, we use four angles of rotation, i.e., $\pm 0.5^{\circ}$ and $\pm 1^{\circ}$ to take into account rotational variations due to printing and scanning. Each character image is padded on all sides by a two pixel wide boundary, rotated, and then its boundary is cropped. The width of padding and cropping is empirically selected such that there is no loss of information. Thus, we obtain four rotated versions for each character image.

The second variation is generated for each character image and its rotated versions by uniformly translating intensities at each pixel by some amount c. The hypothesis for including intensity variations stands on the fact that the distribution of pixel intensities for a specific type of character printed by the same printer is not uniform across various occurences of that character. The exact nature of this variation can be highly complex but for simplicity, here we assume that it can be approximated by monotone variations. The value of c is estimated from a scanned document image (of size $X \times Y$ pixels) as follows:

$$\mathbf{H}_I(i,l) = \sum_{x=1}^{X}\sum_{y=1}^{Y} \delta\left[I_i(x,y) - l\right], \; \forall\, 0 \leq l < L, \; 1 \leq i \leq N \quad (6)$$

$$\mathbf{h}_{sd}(l) = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}\left(\mathbf{H}_I(i,l) - \overline{\mathbf{h}}_I(l)\right)^2}, \; \forall\, 0 \leq l < L \quad (7)$$

$$c = \frac{\mathbf{h}_{sd}(0) + \mathbf{h}_{sd}(1) + ... + \mathbf{h}_{sd}(L-1)}{L} \quad (8)$$

Here, $\delta[.]$ is the impulse function and $I_i(x,y)$ is the intensity value of i^{th} character image at location (x,y). L (=255) is the number of grayscale levels in the scanned document with N extracted characters. The l^{th} element in L-dim vector $\overline{\mathbf{h}}_I$ corresponds to average of all elements in l^{th} column of $\mathbf{H}_I$. Then, we obtain two monotonic grayscale intensity translated versions for each input image by adding and subtracting c from each pixel intensity. Thus, we have 14 variations (4 rotated and ten intensity translated) of each original character image and another 14 variations of noise residual obtained from each original character image.

2.4. Feature Extraction using CNN & SVM Training

In this step, we train CNN using input character images and spatial pyramid pooling (SPP) [19]. We use SPP layer after the convolutional layers to pool the features and generate outputs of fixed-length. In particular, the feature map obtained from last convolutional layer (second convolutional layer in CNN architecture of [14]) is pooled using local spatial bins. The number of spatial bins is kept fixed for all images, but their sizes are varied. This ensures that a fixed-length output is obtained without compromising on spatial information. A max pooling layer with three pyramid levels is chosen and implemented using SPP layer in Caffe [20] with pyramid height

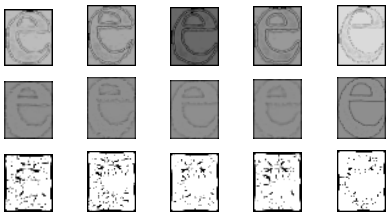

Fig. 2: Inverted pixel representations from 5 printers: proposed noise residual (1^{st} row), average filter residual (2^{nd} row) and median filter residual (3^{rd} row, gamma corrected for visualization).

fixed at 3. Rest of the layers and parameters are kept same as in [14]. For each character image (and its variations), a feature vector of length 500 elements is extracted from the trained CNN model by stopping the forward propagation at the *relu1* layer (in the CNN architecture described in Section IV C of [14]). Further, these features are used as input to linear binary SVM in a one-vs-one fashion which is implemented using the MATLAB code provided by [14]. Finally, a majority vote on predicted character labels predicts the printer label of a test document.

3. EXPERIMENTAL RESULTS

Performance of the proposed method is evaluated on a publicly available dataset [9] consisting of 1184 pages printed from 10 different printers (approx. 120 pages per printer). The first letter of printer label denotes the brand of that printer (Table 3 and 4). For example, C1150 and C4370 belong to Canon while H1518 and H225A belong to HP. The rest of the numbers denote the model. Also, there are two instances of the same brand and model, i.e., H225A and H225B. We consider all ten printers in all the experiments as reported in the state-of-the-art (baseline) method [14]. A set of experiments performed to show the efficacy of the proposed method compare: 1) the proposed noise residual against state-of-the-art character representations and 2) complete proposed method (noise residual, SPP and data augmentation scheme) against the state-of-the-art scheme.

3.1. Effectiveness of Noise Residual

We evaluate the performance of the proposed noise residual (e_{nr}) against existing data representations on train and test folds provided by [14]. Here, we choose median and average residuals (e_{med} and e_{avg} respectively) as baseline representations since they have been used successfully in [14]. In the results presented in Table 1, the second to fourth columns correspond to results for individual character representations. The classification accuracy for the proposed noise residual e_{nr}, averaged over the ten folds, is higher than the baseline representations (e_{med} and e_{avg}). Further, the features obtained from individual character representations and original

Table 1: Average classification accuracies of different character representations with 5×2 cross-validation using the folds of [14]. $e_{raw,nr}$ is the proposed combination of original image and noise residual. Train ≈ 592 pages; Test ≈ 592 pages.

Fold	e_{avg}	e_{med}	e_{nr}	$e_{raw,avg,med}$	$e_{raw,nr}$
1	96.47	94.08	96.1	97.0	**97.5**
2	95.45	93.42	93.9	96.3	**96.6**
3	94.94	93.09	95.1	97.6	**98.3**
4	92.39	84.60	94.1	94.9	**95.9**
5	95.43	94.08	94.2	95.9	**97.0**
6	93.42	93.25	94.9	95.6	**96.1**
7	94.75	93.40	97.0	97.1	**97.6**
8	94.94	94.10	96.0	96.3	**97.0**
9	95.94	95.94	95.9	**97.5**	**97.5**
10	95.45	94.94	95.3	97.1	**97.3**
Average	94.9	93.1	95.3	96.5	**97.1**
Std (σ)	1.2	3.1	1.0	0.9	**0.7**

character image (e_{raw}) are concatenated before classification using SVM. Fifth column in Table 1 lists the results obtained after concatenating features extracted from e_{raw}, e_{avg} and e_{med} [14] while sixth column corresponds to results with concatenation of features from e_{raw} and e_{nr}. This experiment suggests that the proposed combination of original and noise residual image ($e_{raw,nr}$) outperforms the state-of-the-art method ($e_{raw,avg,med}$). Also, the proposed combination requires only 4 CNNs as compared to 6 CNNs in the existing approach. Further, it can be inferred that the original image and its proposed noise residual contain some amount of complementary information. This set of experiments were performed on Matlab using Matconvnet [21].

3.2. Effectiveness of Data Augmentation

We compare the performance of the proposed method against the existing CNN based approach [14]. To use the SPP layer, we conducted this set of experiments using Caffe [20] as it is not available in Matconvnet [21] (used by [14] to implement CNN). The performance is evaluated using only about two percent of the training data used in the previous set of experiments. In particular, we randomly choose two pages per printer (one each for training and validation) from the Fold 1 of training data. Next, we pick one percent (of original training data) character samples from these two pages. The training data is supplemented with augmented data before training the CNN. In some preliminary experiments, we had observed that using only original character images for test data gives better results. So, we use only original character images for testing. We repeat this experiment for five different pairs of training pages. The results in Table 2 confirm that the proposed method outperforms existing method for this limited amount of data. The confusion matrices of the proposed method (Table 4) and the existing method (Table 3) indicate that on 7 out of 8 printers having a single model instance (i.e. except H225A & H225B), the proposed method outperforms existing method, even reaching 100% average classification accuracy for printer SC315.

Table 2: Comparison of proposed method with the state-of-the-art. Accuracy is averaged over 5 folds; Test ≈ 592 pages.

Method	# Original Train Characters per Printer (e + a)	Augmented Data	Average Accuracy	σ
State-of-the-art [14]	386	No	85.60 %	2.76
Proposed	386	Yes	88.26 %	3.33

Table 3: Confusion matrix of the state-of-the-art CNN based method [14] showing (in %) average accuracies over 5 splits.

True \ Predicted	B4070	C1150	C3240	C4370	H1518	H225A	H225B	LE260	OC330	SC315
B4070	**90.00**	**9.06**		**0.63**		**0.31**				
C1150	**5.66**	**93.21**	**1.13**							
C3240	**0.90**	**1.49**	**94.93**	**2.09**		**0.60**				
C4370	**0.66**	**0.66**	**17.38**	**80.98**		**0.33**				
H1518	**1.02**	**1.02**		**6.78**	**91.19**					
H225A					**34.39**	**52.63**	**12.98**			
H225B						**30.80**	**69.20**			
LE260		**0.31**				**1.56**	**7.81**	**90.31**		
OC330								**4.83**	**95.17**	
SC315									**4.14**	**95.86**

4. CONCLUSIONS

This work presents an approach for enhancing the capability of any CNN based method for printer attribution problems under the constraints of very limited training data per printer. We do not propose a new architecture for CNN but simply introduce new ways of enhancing the capability of a given CNN architecture. Comparison against state-of-the-art method shows promising results. We derive a noise residual from the original character image which is based on printer characteristics. On the other hand, existing character representations are derived from generic filtering operations whose connection to the printing process is difficult to trace. Moreover, experiments confirm that the original character image and its noise residual carry complementary information. Further, we evaluate performance using augmented data (rotation and intensity variations) where the intensity variations are derived from scanned document. The experiments show that the proposed technique achieves an improvement of about 2.5% in classification accuracy using only 4 CNNs as compared to 6 CNNs in existing method. Future work will include analysis of performance by varying the amount of training data and generation of more realistic models for intensity variations.

Table 4: Confusion matrix of the proposed method showing (in %) average accuracies over 5 splits.

True \ Predicted	B4070	C1150	C3240	C4370	H1518	H225A	H225B	LE260	OC330	SC315
B4070	**95.31**		**0.94**	**0.63**				3.13		
C1150	**1.89**	**96.98**	**0.75**					0.38		
C3240			**99.10**	**0.60**				0.30		
C4370			**13.77**	**86.23**						
H1518	**1.02**	**1.02**		**9.15**	**88.81**					
H225A					**34.04**	**35.44**	**30.53**			
H225B						**16.00**	**84.00**			
LE260							**4.69**	**95.31**		
OC330								**2.41**	**97.59**	
SC315										**100.00**

5. REFERENCES

[1] Gazi N Ali, Pei-Ju Chiang, Aravind K Mikkilineni, Jan P Allebach, George T.-C. Chiu, and Edward J Delp, "Intrinsic and Extrinsic Signatures for Information Hiding and Secure Printing with Electrophotographic Devices," in *Proc. IS&T's NIP19: Int. Conf. Digital Printing Technologies*, 2003, vol. 19, pp. 511–515.

[2] Eric Kee and Hany Farid, "Printer Profiling for Forensics and Ballistics," in *Proc. 10th ACM Workshop Multimedia and Security*, 2008, pp. 3–10.

[3] Jung-Ho Choi, Dong-Hyuck Im, Hae-Yeoun Lee, Jun-Taek Oh, Jin-Ho Ryu, and Heung-Kyu Lee, "Color Laser Printer Identification by Analyzing Statistical Features on Discrete Wavelet Transform," in *Proc. 16th IEEE International Conference on Image Processing (ICIP)*, 2009, pp. 1505–1508.

[4] Orhan Bulan, Junwen Mao, and Gaurav Sharma, "Geometric Distortion Signatures for Printer Identification," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing*, 2009, pp. 1401–1404.

[5] Sara Elkasrawi and Faisal Shafait, "Printer Identification Using Supervised Learning for Document Forgery Detection," in *Proc. 11th IAPR Int. Workshop on Document Analysis Systems, DAS 2014*, 2014, pp. 146–150.

[6] Min Jen Tsai, Jin Shen Yin, Imam Yuadi, and Jung Liu, "Digital Forensics of Printed Source Identification for Chinese Characters," *Multimedia Tools and Applications*, vol. 73, no. 3, pp. 2129–2155, 2014.

[7] Jianyuan Hao, Xiangwei Kong, and Shize Shang, "Printer Identification Using Page Geometric Distortion on Text Lines," in *Proc. IEEE China Summit and Int. Conf. Signal and Info. Process.*, 2015, pp. 856–860.

[8] Qianjin Zhou, Yuchen Yan, Tianhong Fang, Xiao Luo, and Qinghu Chen, "Text-Independent Printer Identification Based on Texture Synthesis," *Multimedia Tools and Applications*, pp. 1–24, 2015.

[9] Anselmo Ferreira, Luiz C Navarro, Giuliano Pinheiro, Jefersson A dos Santos, and Anderson Rocha, "Laser Printer Attribution: Exploring New Features and Beyond," *Forensic Science International*, vol. 247, pp. 105–125, 2015.

[10] Sharad Joshi and Nitin Khanna, "Single Classifier-Based Passive System for Source Printer Classification using Local Texture Features," *arXiv preprint arXiv:1706.07422*, 2017.

[11] Henry S Baird, "The State of The Art of Document Image Degradation Modelling," *Digital Document Processing*, pp. 261–279, 2007.

[12] Pei-Ju Chiang, Nitin Khanna, Aravind K Mikkilineni, Maria V Ortiz Segovia, Sungjoo Suh, Jan P Allebach, George T-C Chiu, and Edward J Delp, "Printer and Scanner Forensics," *IEEE Signal Process. Mag.*, vol. 26, no. 2, pp. 72–83, 2009.

[13] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, "Imagenet Classification with Deep Convolutional Neural Networks," in *Proc. Advances in Neural Information Processing Systems*, 2012, pp. 1097–1105.

[14] Anselmo Ferreira, Luca Bondi, Luca Baroffio, Paolo Bestagini, Jiwu Huang, Jefersson dos Santos, Stefano Tubaro, and Anderson Rocha, "Data-driven feature characterization techniques for laser printer attribution," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 8, pp. 1860–1873, 2017.

[15] Patrice Y Simard, David Steinkraus, John C Platt, et al., "Best Practices for Convolutional Neural Networks Applied to Visual Document Analysis.," in *Proc. IAPR International Conference on Document Analysis and Recognition (ICDAR)*. IEEE, 2003, vol. 3, pp. 958–962.

[16] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer, "SMOTE: Synthetic Minority Over-Sampling Technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, 2002.

[17] Sebastien C Wong, Adam Gatt, Victor Stamatescu, and Mark D McDonnell, "Understanding Data Augmentation for Classification: When to Warp?," in *Proc. Int. Conf. Digital Image Computing: Techniques and Applications (DICTA)*. IEEE, 2016, pp. 1–6.

[18] Steven H Low, Nicholas F Maxemchuk, and Aleta M Lapone, "Document Identification for Copyright Protection Using Centroid Detection," *IEEE Transactions on Communications*, vol. 46, no. 3, pp. 372–383, 1998.

[19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 9, pp. 1904–1916, 2015.

[20] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell, "Caffe: Convolutional Architecture for Fast Feature Embedding," *arXiv preprint arXiv:1408.5093*, 2014.

[21] A. Vedaldi and K. Lenc, "MatConvNet – Convolutional Neural Networks for MATLAB," in *Proceeding of the ACM Int. Conf. on Multimedia*, 2015.