https://doi.org/10.1186/s40537-019-0282-4

Journal of Big Data

RESEARCH Open Access

RETRACTED ARTICLE

Criminal tendency detection from facial images and the gender bias effect

Mahdi Hashemi[1]* and Margeret Hall[2]

*Correspondence: mhashem2@gmu.edu
[1] Department of Information Sciences and Technology, George Mason University, 4400 University Dr, Fairfax, VA 22030, USA
Full list of author information is available at the end of the article

Abstract

Explosive performance and memory space growth in computing machines, along with recent specialization of deep learning models have radically boosted the role of images in semantic pattern recognition. In the same way that a textual post on social media reveals individual characteristics of its author, facial images may manifest some personality traits. This work is the first milestone in our attempt to infer personality traits from facial images. With this ultimate goal in mind, here we explore a new level of image understanding, inferring criminal tendency from facial images via deep learning. In particular, two deep learning models, including a standard feedforward neural network (SNN) and a convolutional neural network (CNN) are applied to discriminate criminal and non-criminal facial images. Confusion matrix and training and test accuracies are reported for both models using tenfold cross-validation on a set of 10,000 facial images. The CNN was more consistent than the SNN in learning to reach its best test accuracy, which was 8% higher than the SNN's test accuracy. Next, to explore the classifier's hypothetical bias due to gender, we controlled for gender by applying only male facial images. No meaningful discrepancies in classification accuracies or learning consistencies were observed, suggesting little to no gender bias in the classifier. Finally, dissecting and visualizing convolutional layers in CNN showed that the shape of the face, eyebrows, top of the eye, pupils, nostrils, and lips are taken advantage of by CNN in order to classify the two sets of images.

Keywords: Image classification, Facial images, Convolutional neural network, Deep learning, Machine learning, Personality traits

Introduction

Face is the primary means of recognizing a person, transmitting information, communicating with others, and inferring people's feelings, among others. Our faces might disclose more than what we expect. A facial image can be informative of personal traits [1], such as race, gender, age, health, emotion, psychology, and profession.

This study is triggered by Lombroso's research [2], which showed that criminals could be identified by their facial structure and emotions. While Lombroso's study looked at this issue from a physiology and psychiatry perspective, our study investigates whether or not machine learning algorithms would be able to learn and distinguish between criminal and non-criminal facial images. More specifically, we will look for gender biases in machine predictions. This is important because criminal facial images used to train

Springer Open

© The Author(s) 2020. This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

the machine are mostly male. This is the result of the large gap between the number of mugshots for arrested males and females, available to the public and used to train the machine.

It is noteworthy that this study's scope is limited to the technical and analytical aspects of this topic, while its social implications require more scrutiny and its practical applications demand even higher levels of caution and suspicion. With that in mind, this paper explores the deep learning's capability in distinguishing between criminal and non-criminal facial images. To this effect, two deep learning models, a standard feedforward neural network (SNN) and a convolutional neural network (CNN), are trained with 10,000 neutral-emotion, mixed-gender, mixed-race facial images. A neutral or blank face expression is characterized by neutral positioning of the facial features. A neutral face expression could be caused by a lack of emotion, boredom, depression, or slight confusion. A neutral face expression is also referred to as a poker face. It is meant to conceal one's emotions when playing the card game poker [3]. While both neural network models are controlled for facial emotions by applying only neutral emotion images, no control has been imposed on race, due to our small dataset and the difficulty and occasionally subjectivity of identifying the race from low quality facial images. Both models are trained with and without controlling for gender. The results indicated that controlling for gender does not have much effect on accuracy or learning and both models reach high classification accuracies regardless. CNN achieves a tenfold cross-validation accuracy of 97%.

The strength of this study lies in its application of neural networks to investigate if a stack of non-linear functions with thousands of parameters can find useful facial features to distinguish between criminal and non-criminal face shots. Its weakness however lies in its reliance on machine to learn these features and on a limited number of images.

"Related work" section provides a review of related works. "Methodology" section elaborates on this study's methodology. "Data preparation" section describes the image dataset sources and the approach taken to prepare the dataset. "Neural network architecture" section describes the SNN's and CNN's architecture, proposed in this study, for criminal tendency recognition through facial images. "Visual criminal tendency detection results and discussion" section presents the results for both mixed gender and male only classification scenarios. "Conclusion and future directions" section concludes the paper by discussing the results and future directions.

Related work

Machine learning has shown to be more effective than humans in discovering personality traits through facial images [4]. Geng et al. [5] trained a machine to estimate the age through facial images. Reece and Danforth [6] applied an ensemble of machine learning models and image processing to detect depression and psychiatric disorder in Instagram facial images.

The goal in facial emotion detection is to train a machine to distinguish among six emotional facial expressions: happiness, surprise, sadness, disgust, anger, and fear [7]. Fuzzy inference system [8], hidden Markov model based on real-time tracking of the mouth shape [9], and Bayesian network [10] are among the approaches used for classifying facial emotions.

RETRACTED ARTICLE

RETRACTED ARTICLE

Criminal tendency is another personality trait. Lombroso [2] was the first in 1871 to point out that criminals could be identified by their facial structure and emotions. Recently, Wu and Zhang [11] revisited this theory and quantitatively demonstrated the correlation between criminality and facial features. They trained four classifiers: logistic regression, k nearest neighbors (KNN), support vector machines (SVM), and convolutional neural network (CNN) and claimed that their machine can identify a criminal face with a 90% accuracy. Their model was controlled for gender, race, and facial expression of emotions.

Neural networks have resurged and drawn much attention in the last decade [12] with the new brand of deep learning, mainly due to the significant performance gain in visual recognition tasks [13]. Deep learning has been applied to a wide range of applications, such as tree disease recognition [14]. Among the most relevant applications of deep learning to our work, we can point to the application of CNN for face recognition [15, 16]. Cristani [17] and Segalin et al. [18, 19] applied machine learning to predict the self-assessed personality traits (openness to experience, conscientiousness, extraversion, agreeableness, and neuroticism) of a person from the images he/she uploads or likes on social media, such as Flicker, and what impressions in terms of personality traits those images trigger in unacquainted people. They performed their experiments with 60,000 images from 300 Flickr users. Cristani et al. [17] and Segalin et al. [18] used a hybrid approach where generative models, used as latent representations of features (color, composition, textural properties, etc.) extracted from images, are built and then passed to a discriminative classifier to predict each user's personality traits. Simplifying the problem into five distinct binary classification problems, one for each trait, Segalin et al. [19] applied AlexNet [13], which is an eight-layer version of CNN, pre-trained on ImageNet 2012 competition dataset. The problem they pose is to detect the personality traits based on the images that one uploads or likes on social media, such as Flicker. Their results showed that the personality trait that others attribute to a person (based on the images that that individual uploads or likes on social media) can be predicted 10% more accurately than the personality traits that that individual attributes to him/her-self. Wang and Kosinski [4] trained a deep neural network to classify facial images based on sexual orientation.

Methodology

Figure 1 shows the general workflow of this study. The first step is data collection and preparation which is performed as follows.

Data preparation

A total of 8401 gray-scale mugshot images of arrested individuals are obtained from National Institute of Standards and Technology (NIST) Special Database [20]. Images are all in png format. Images are of mixed race, mixed gender, and neutral face expression and contain both front and side (profile) views. Since our focus is on frontal face shots, we need to eliminate profile views. Haar feature-based cascade classifier [21] detects images containing frontal face views and also detects the rectangular area containing the face. Images are passed to the pre-trained version of this classifier, available in the OPenCV library in Python, to keep only the images that contain frontal face views

RETRACTED ARTICLE

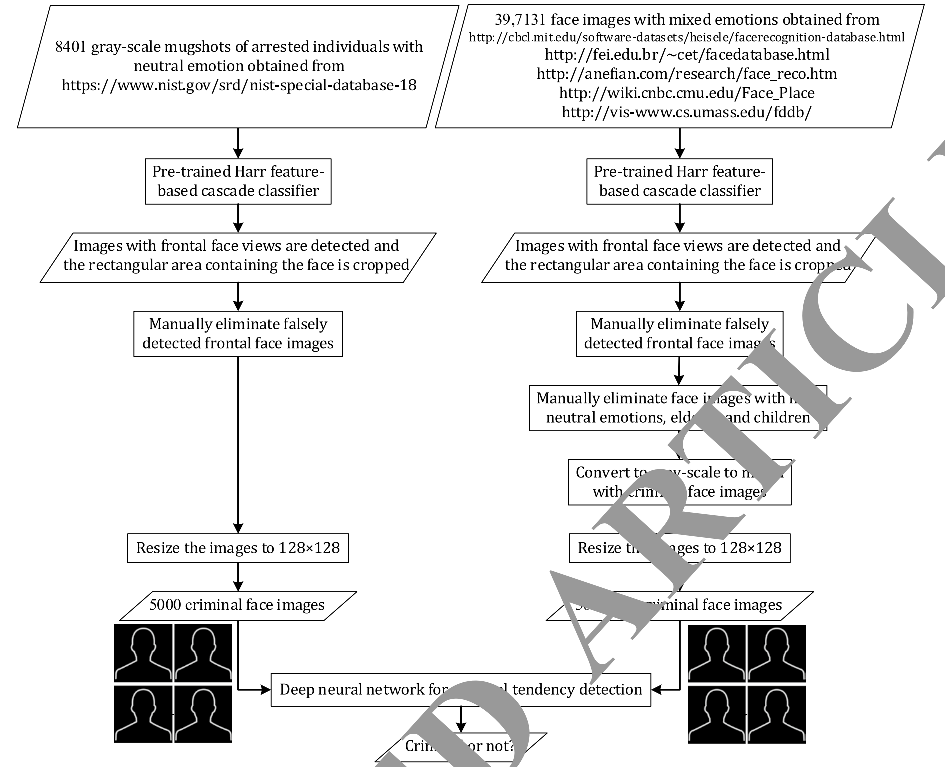
8401 gray-scale mugshots of arrested individuals with neutral emotion obtained from https://www.nist.gov/srd/nist-special-database-18
39,7131 face images with mixed emotions obtained from http://cbcl.mit.edu/software-datasets/heisele/facerecognition-database.html http://fei.edu.br/~cet/facedatabase.html http://anefian.com/research/face_reco.htm http://wiki.cnbc.cmu.edu/Face_Place http://vis-www.cs.umass.edu/fddb/
Pre-trained Harr feature-based cascade classifier
Pre-trained Harr feature-based cascade classifier
Images with frontal face views are detected and the rectangular area containing the face is cropped
Images with frontal face views are detected and the rectangular area containing the face is cropped
Manually eliminate falsely detected frontal face images
Manually eliminate falsely detected frontal face images
Manually eliminate face images with non-neutral emotions, elderly and children
Convert to gray-scale to match with criminal face images
Resize the images to 128×128
Resize the images to 128×128
5000 criminal face images
5000 non-criminal face images
Deep neural network for criminal tendency detection
Criminal or not?

Fig. 1 Constructing criminal vs. non-criminal facial image datasets

and then crop the rectangular area containing the face. Cropping the facial rectangle from the rest of the image prevents the classifier from being affected by peripheral or background effects surrounding the face. The false positive rate (none-frontal face images misclassified as frontal face images) of the Haar feature-based cascade classifier was 1.9% which were manually deleted. The result contains 5000 front view face images of 4796 male and 204 female individuals and of variable sizes, ranging from 238×238 up to 813×813 pixels. Since neural networks receive inputs of the same size, all images are resized to 128×128 using bilinear interpolation. This size is chosen considering the capacity of our platform (64-bit 3.00 GHz Xeon E3-1505 M v6 processor, 2400 MHz 16 GB DDR4 SODIMM RAM, NVIDIA Quadro M2200 4 GB GDDR5 GPU) to process the images collectively.

A total of 39,713 RGB facial images are obtained from five sources (Face Recognition Database [22], FEI Face Database [23], Georgia Tech face database [24], Face Place [25], Face Detection Data Set and Benchmark Home [26]). We consider these images as non-criminal face shots. Images are all in jpg format. Images are of mixed race, mixed gender, and mixed facial expressions. The database contains both front and side (profile) views. Since our focus is on frontal face shots, we need to eliminate profile views, using the Haar feature-based cascade classifier [21]. The false positive rate (none-frontal face images misclassified as frontal face images) of the Haar feature-based cascade classifier was 1.3% which were manually deleted. Facial images with any emotion expression but neutral are manually deleted, in order for compatibility with criminal facial images which are all neutral. Also, to keep the age, approximately, in the same range with the criminal dataset, images of elderly and children are manually deleted from this dataset. The images are then converted to gray-scale, again to be compatible with mugshots in the criminal dataset. The result contains 5000 front view face images of 3727 male and 1273 female individuals and of variable size, ranging from 87 × 87 up to 799 × 799 pixels. Images are resized to 128 × 128.

Neural network architecture

As shown in Fig. 1, the data are passed to an artificial neural network for further classification. Artificial neural networks do not rely on hand-engineered features which are hard to select and design. The neural network in our application receives, as input, 128 × 128 pixel gray-scale images. Without extra preprocessing, the image pixels are only divided by 255 so that they are in the range 0 to 1. Before describing the neural network architecture, we justify our choice of activation function, loss function, and training algorithm.

While saturated activation functions, e.g. sigmoid or tanh, could trigger the vanishing gradients problem and prevent the exploding gradients problem because of their near-zero gradient at large values, non-saturated activation functions, e.g. rectified linear unit (ReLU), could trigger the exploding gradients problem and prevent the vanishing gradients problem because of their non-zero gradient at large values. Both problems happen for synaptic weights at lower layers and will prevent the network from being properly trained. The exploding gradient problem is easier to detect because the vanishing gradients could also happen due to the training convergence. Besides, non-saturated activation functions make the training several times faster [13]. Therefore, we chose the non-saturated activation function ReLU [27, 28]. ReLU is a piecewise linear function, defined as the positive part of its argument: $ReLU(z) = max(0,z)$. By projecting negative inputs to zero, ReLU creates a sparseness in the activation of neural units, a desirable effect similar to dropout. The softmax function, $softmax(z_i) = exp(z_i)/\Sigma_j \, exp(z_j)$, used in the final layer, transforms the values (z_i) to normalized exponential probabilities whose summation is one (i.e. $\Sigma_c \, p_c = 1$). This provision ($\Sigma_c \, p_c = 1$) is a prerequisite for the application of cross-entropy loss function, which is calculated as: $-\,\Sigma_c \, y_c \, log(p_c)$, where c represents a neuron (or class) in the output layer, y_c represents the desired value (0 or 1) at that neuron, and p_c is the predicted probability at that neuron. The cross-entropy loss function simplifies to $-\,(y \, log(p) + (1 - y) \, log(1 - p))$ for the binary classification in our case. The network is trained using the Adam optimization algorithm [29], which is an extension to the stochastic gradient descent (SGD) approach, with a batch size of 100. Unlike SGD which maintains a single and fixed learning rate for all synaptic weight updates, Adam continually adjusts individual adaptive learning rates for each synaptic weight based on estimates of first and second moments of the gradients. The learning rate is initialized at 0.0001 and the exponential decay rate for the first and second moment estimates are set to 0.9 and 0.999 respectively, suggested by Kingma and Ba [29].

Two neural network architectures are applied for classifying facial images into criminal and non-criminal categories, an SNN and a CNN. The SNN composes of four fully-connected layers, in addition to the input layer which has 16,384 neurons,

RETRACTED ARTICLE

equal to the total number of pixels in an image. The first three fully-connected layers have 512 neurons and each are followed by an ReLU layer. The fourth fully-connected layer has a size equal to the number of target categories, labeled as criminal and non-criminal, and is followed by a softmax function. The overall architecture is shown in Fig. 2.

CNN has recently outperformed other neural network architectures and other machine learning and image processing approaches in image classification [1, 30–36] and object detection [37] due to its independence from hand-crafted visual features and excellent abstract and semantic abilities [34, 38]. CNN makes strong and mostly correct assumptions about the nature of images, namely, locality of pixel dependencies and stationarity of statistics. Therefore, in comparison with SNN with similarly-sized layers, CNN has much fewer connections and parameters which makes it easier to train. The applied CNN in this work composes of two convolutional layers followed by two fully-connected layers. Convolutional layers have the following settings: $f_1 = 3 \times 3 \times 1$, $s_1 = 1$, $n_1 = 8$, $f_2 = 3 \times 3 \times 8$, $s_2 = 1$, $n_2 =$ [illegible], where f_m, s_m, and n_m denote the size, stride, and number of filters of the m-th layer respectively. Every convolutional layer is followed by a max pooling and ReLU layer. Pooling summarizes the outputs of neighboring groups of neurons in the same kernel map. We use 2×2 max pooling with a stride of 2, which means the pooling regions do not overlap. Smaller pooling regions cause over-fitting (high variance) and larger regions are too generic and lose the details (high bias [39]). The first fully-connected layer has 64 neurons

RETRACTED ARTICLE

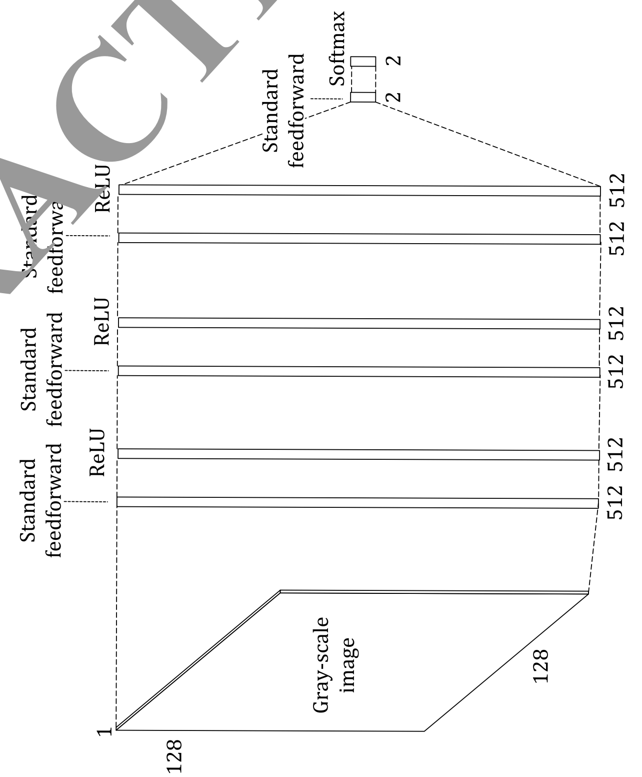

Fig. 2 The standard neural network architecture

and is followed by an ReLU layer. The second fully connected layer has a size equal to the number of target categories, labeled as criminal and non-criminal, and is followed by a softmax function. The overall architecture is shown in Fig. 3.

The convolution filter and the pooling filter (elaborated in the next section) would slip outside the input image into the void, when they attempt to center themselves at bordering pixels. There are two strategies to solve this issue: (a) stopping the filter before it slips outside the image and (b) padding the input image with zero pixels. The first approach

RETRACTED ARTICLE

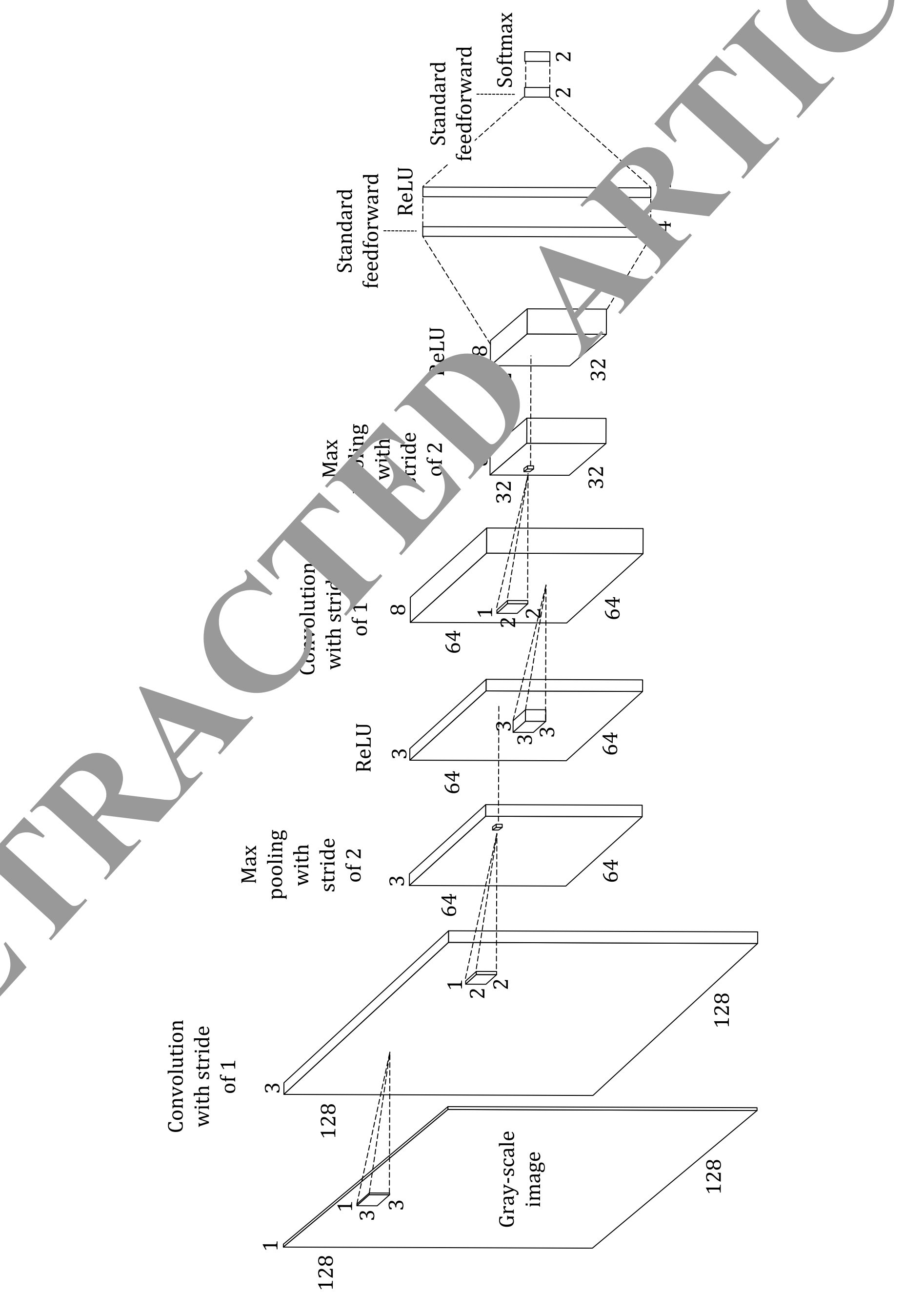

Fig. 3 The convolutional neural network architecture

comes at the cost of under-scanning the bordering pixels because the filter will not get a chance to center itself at the bordering pixels. The second approach is referred to as padding and is the one applied in our model.

Visual criminal tendency detection results and discussion

Experiments are conducted on a 64-bit 3.00 GHz Xeon E3-1505M v6 processor, 2400 MHz 16 GB DDR4 SODIMM RAM, NVIDIA Quadro M2200 4 GB GDDR5 GPU. Artificial neural network models are implemented in Python using the TensorFlow library [40].

Splitting a small dataset into training and testing sets would leave us with even a smaller training set. In cross-validation, all the samples could be used for both training and testing, while the model is evaluated on previously unseen samples. Additionally, in k-fold cross-validation, we train and test k models. This allows us to be more confident in the performance results. Consequently, we can not only report a more solid test accuracy, but also the standard deviation for this test accuracy. Finally, cross-validation allows us to tune the number of layers in our neural network, which will be further elaborated at the end of this section. With these advantages in mind, the tenfold cross-validation approach is applied here. The tenfold is preferred over its fivefold counterpart to produce a more accurate standard deviation.

The neural networks are trained up to [illegible]0 epochs, after which the change in training accuracy becomes imperceptible. The charts in Fig. 4 represent the average and standard deviation of training and test accuracies at each epoch. The tenfold cross-validation has been performed at each epoch. Thus, the training and test accuracies at each epoch, reported in Fig. 4, are the average over the ten folds. The standard deviation of accuracies is also calculated over the ten folds at each epoch and depicted using the line's thickness. The CNN achieves its highest test accuracy (97% with a standard deviation of 0.91%) at epoch 306. While the training accuracy keeps rising after this epoch, the test accuracy starts dropping. The test accuracy of 97%, achieved by CNN (Fig. 4a), exceeds our expectations and is a clear indicator of the possibility to differentiate between criminals and non-criminals using their facial images. It is noteworthy that the criminal mugshots are coming from a different source than non-criminal face shots. That means the conditions under which the criminal images are taken are different than those of non-criminal images. These different conditions refer to the camera, illumination, angle, distance, background, resolution, etc. Such disparities which are not related to facial structure, though negligible in majority of cases, might have slightly contributed in training the classifier and helping the classifier to distinguish between the two categories. Therefore, it would be too ambitious to claim that this accuracy is easily generalizable.

Interestingly but not surprisingly, the CNN (Fig. 4a) achieves a higher test accuracy than the SNN (Fig. 4b), also in a more consistent way. The CNN's best test accuracy (97%) is 8% higher than the SNN's best test accuracy (89% with a standard deviation of 1.18%). This goes back to the SNN being general purpose but the CNN being specifically designed for image classification. On the other hand, the training accuracy is only 0.37% different for CNN and SNN, pointing to their equal capacity in learning from the training data. The CNN is more consistent in learning because the variance around its

RETRACTED ARTICLE

RETRACTED ARTICLE

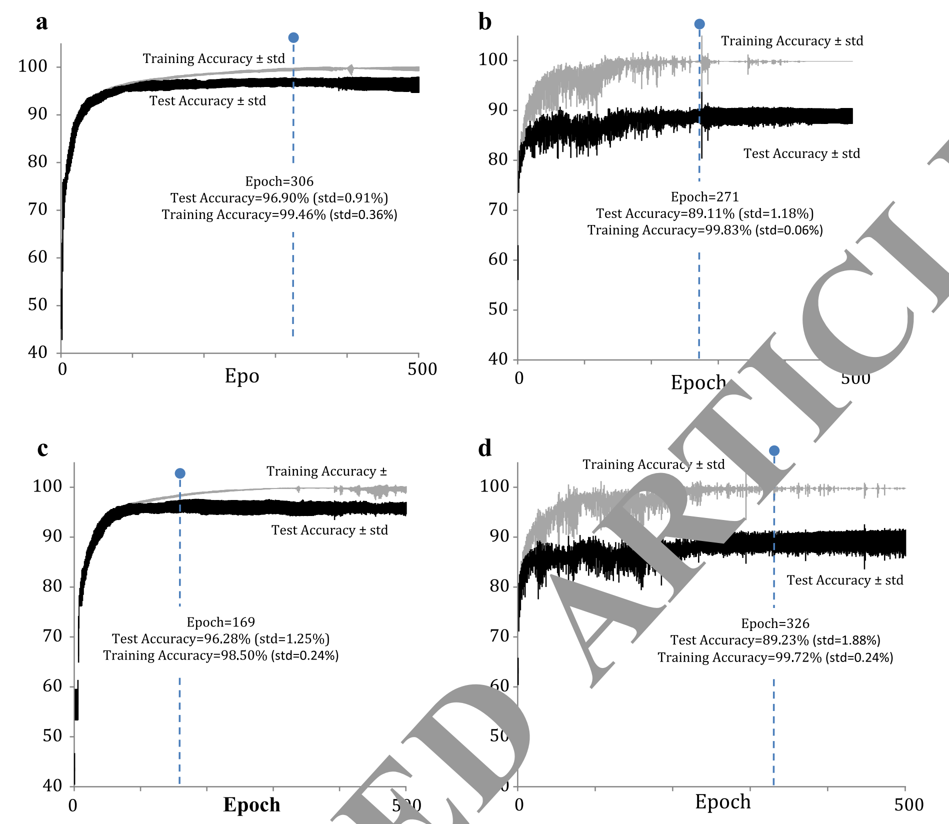

Fig. 4 Training and test accuracy with one standard deviation of uncertainty at different epochs for: **a** CNN, **b** SNN, **c** CNN when applied to only male images, and **d** SNN when applied to only male images

training and test accuracy curves (Fig. 4a) is tighter than that of the SNN (Fig. 4b). The higher consistency and accuracy of the CNN are because of its assumption of locality of pixel dependency and its fewer parameters.

The confusion matrixes for the CNN and SNN are shown in Tables 1 and 2, respectively. The difference between the false positive and false negative rates is 1% for the

Table 1 Confusion matrix for CNN

		Predicted	
		Criminal	**Non-criminal**
Truth	Criminal	4881	142
	Non-criminal	192	4785

Table 2 Confusion matrix for SNN

		Predicted	
		Criminal	**Non-criminal**
Truth	Criminal	4515	508
	Non-criminal	604	4373

RETRACTED ARTICLE

Table 3 Confusion matrix for CNN when applied to only male images

		Predicted	
		Criminal	**Non-criminal**
Truth	Criminal	4694	116
	Non-criminal	261	3452

Table 4 Confusion matrix for SNN when applied to only male images

		Predicted	
		Criminal	**Non-criminal**
Truth	Criminal	4423	387
	Non-criminal	555	3158

CNN and 2% for the SNN. In other words, the false positive and false negative rates are almost the same for both CNN and SNN, i.e. the classifier has no meaningful bias in making either type of mistake more than the other. We also observed that there are misclassified men, women, white, and colored people from both categories. Among the false negatives (criminal images which were misclassified as non-criminal) by CNN, 81% were male and 19% were female. This is proportional to the 75% male vs. 25% female images among non-criminals. Among the false positives (non-criminal images which were misclassified as criminal) by CNN, 88% were male and 12% were female. This is proportional to the 95% male vs. 5% female images among criminals. Among the false negatives by CNN, 79% were white people and 21% were colored people. This is proportional to the 69% white vs. 31% colored people among non-criminals. Among the false positives by CNN, 79% were white and 21% were colored. This is proportional to the 72% white vs. 28% colored people among criminals. This indicates that the classifier is not biased to put people of a specific gender or race in a specific category while ignoring their criminal tendency.

There are more females among non-criminal images than criminal ones. While 25% of non-criminal images are female, only 4% of criminal images are female. The machine might be unfairly taking advantage of this distinction to boost its classification accuracy. To observe and control the gender bias effect, we separate male and female images in each category. Since the number of female images is too small, we only train and cross-validate the models using male images. There are 4796 male images in the criminal and 3727 in the non-criminal category. Figure 4c, d show the average and standard deviation of training and test accuracies over different training epochs for the CNN and SNN, respectively. These charts very closely imitate their mixed gender counterparts in Fig. 4a, b, a sign that gender has no effect on biasing the classifier one way or the other. The corresponding confusion matrixes for CNN and SNN when applied to only male images, shown in Tables 3 and 4, endorse the same conclusion.

Choosing the CNN to have two convolutional layers was the result of an experimental model complexity vs. generalization accuracy analysis. Figure 5 shows how changing the number of convolutional layers affects the tenfold cross-validation accuracy and its standard deviation. According to this graph, the CNN with five convolutional

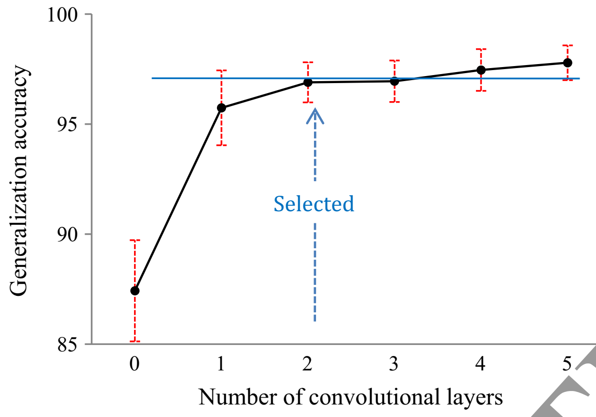

Fig. 5 Number of convolutional layers in CNN vs. tenfold cross-validation accuracy and its standard deviation

RETRACTED ARTICLE

layers achieves the highest accuracy. However, the accuracy of the CNN with four convolutional layers falls within one standard deviation margin of the accuracy of the CNN with five convolutional layers. This is true for CNNs with three and two convolutional layers as well. Thus, the CNN with two convolutional layers is considered optimum, in this case. The architecture of the CNN with two convolutional layers is explained in Sect. 3. The CNNs with less than two convolutional layers are obtained by dropping the last convolutional layers. For CNNs with more than two convolutional layers, we have: $f_3=3\times3\times16$, $s_3=1$, $n_3=32$, $f_4=3\times3\times32$, $s_4=1$, $n_4=64$ and $f_5=3\times3\times64$, $s_5=1$, $n_5=128$, where f_m, s_m, and n_m denote the size, stride, and number of filters of the m-th convolutional layer.

Facial features and criminal tendency

Convolutional layers of CNN are essentially feature generation layers. If a CNN achieves a high accuracy, it means that the generated features by convolutional layers are effective in distinguishing between classes. Therefore, to understand what facial features are used by CNN to classify the images, we need to look at the facial features that are emphasized or pinpointed by each convolutional layer. A convolutional layer usually has multiple filters. Each filter separately contributes in feature generation, though it is their cumulative knowledge that helps CNN to classify the images. Our CNN contains 2 convolutional layers; the first one has 8 filters and the second one has 16.

In Fig. 6, the output of one of the filters from the first convolutional layer and one of the filters from the second convolutional layer are visualized. They highlight the facial characteristics that are learned and used by CNN to distinguish between the two classes. Additionally, Fig. 6 compares these facial features between a criminal and non-criminal face shot. It is noteworthy that neither these facial features nor their differences are hard coded into the machine. They are learned by the machine as most helpful in classifying the two sets of images in the training dataset. Both convolutional layers detect and underscore the shape of the face, eyebrows, top of the eye, pupils, nostrils, and lips.

RETRACTED ARTICLE

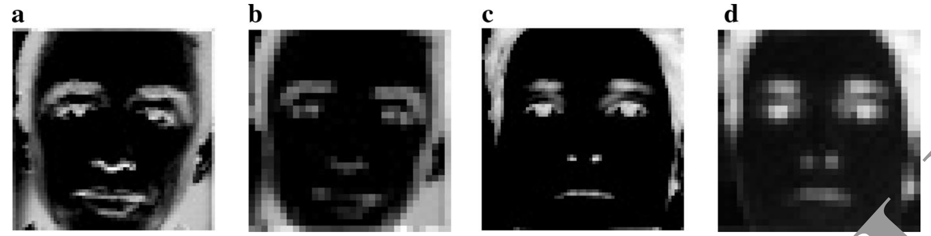

Fig. 6 Facial features detected by the first (**a**, **c**) and second (**b**, **d**) convolutional layers in CNN, for a criminal (**a**, **b**) vs. non-criminal (**c**, **d**) face shot

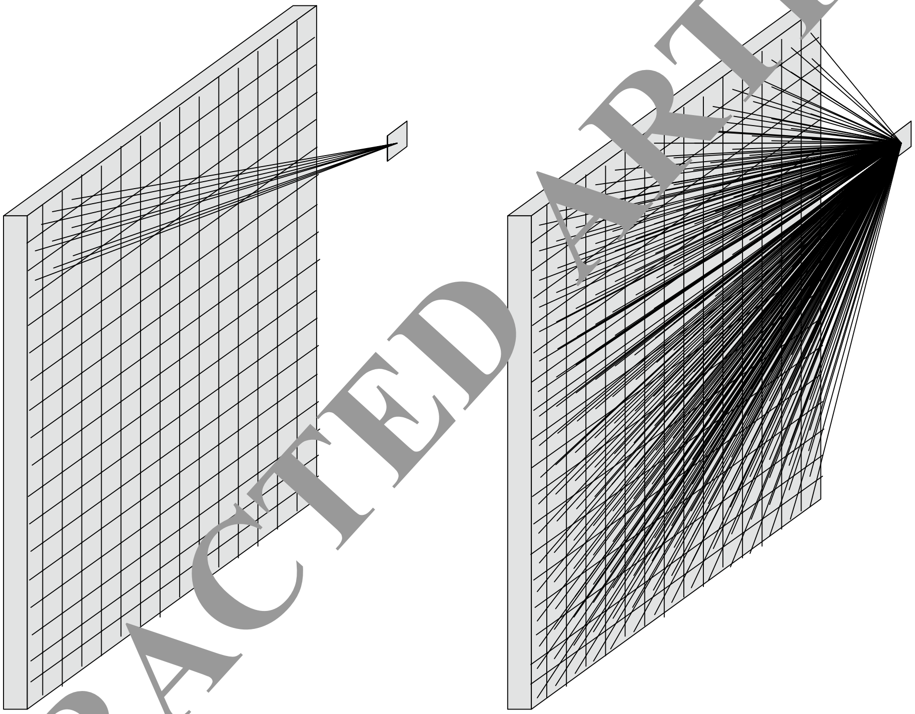

Fig. 7 A node in the first hidden layer is connected to only a small number of image pixels in CNN (left) while it is connected to all image pixels in SNN (right)

Why CNN achieves higher accuracy than SNN?

Two architectural features of CNNs making them more convincing than SNNs for image classification are as follows:

a. Partial connectivity rather than full connectivity
   A node in a CNN is connected only to a small number of nodes in the previous layer, while the same node in an SNN is connected to all nodes in the previous layer. This means that the number of synaptic weights that need to be calculated is mush fewer in CNN than SNN. Assume we use a 3×3 convolution window in the CNN, shown on the left side of Fig. 7. This means a node in the first hidden layer, for instance, is only connected to 9 pixels in the image. The same node in the SNN, shown on the

right side of Fig. 7, is connected to all the 270 pixels of the image. In other words, the number of synaptic weights is 30 times fewer in the CNN than SNN. Of course, this number depends on the size of both the image and convolution window. If the image is $n \times m$ and the convolution window is $z \times z$, the number of synaptic weights in CNN is $n \times m/z^2$ times fewer than SNN. We showed this only for the first hidden layer, but the same is true for all convolutional hidden layers. This has two advantages. First, a much fewer unknown parameters (synaptic weights) can be learned more quickly (less computational complexity) and accurately by the machine, with a significantly reduced chance of overfitting. Second, deriving the value of each node in the next layer from only a small number of neighboring pixels, rather than the entire image, is based on the assumption that the relationship between two distant pixels is probably less significant than two close neighbors. This assumption is inspired by the visual cortex system in humans and other animals.

b. Shared weights

We mentioned that $n \times m$ synaptic weights need to be learned for one node in the first hidden layer of SNN. With k nodes in the first hidden layer, a total of $n \times m \times k$ synaptic weights must be calculated, because each node in the first hidden layer has its own synaptic weights which are different than those of other nodes. In a CNN, however, the number of synaptic weights that need to be learned remains z^2, because nodes in the first hidden layer do not have different synaptic weights, but share the same weights. Therefore, regardless of how many nodes exist in the first hidden layer, the number of synaptic weights that need to be learned remains z^2. Consequently, the number of synaptic weights in CNN is $n \times m \times k/z^2$ times fewer than SNN for

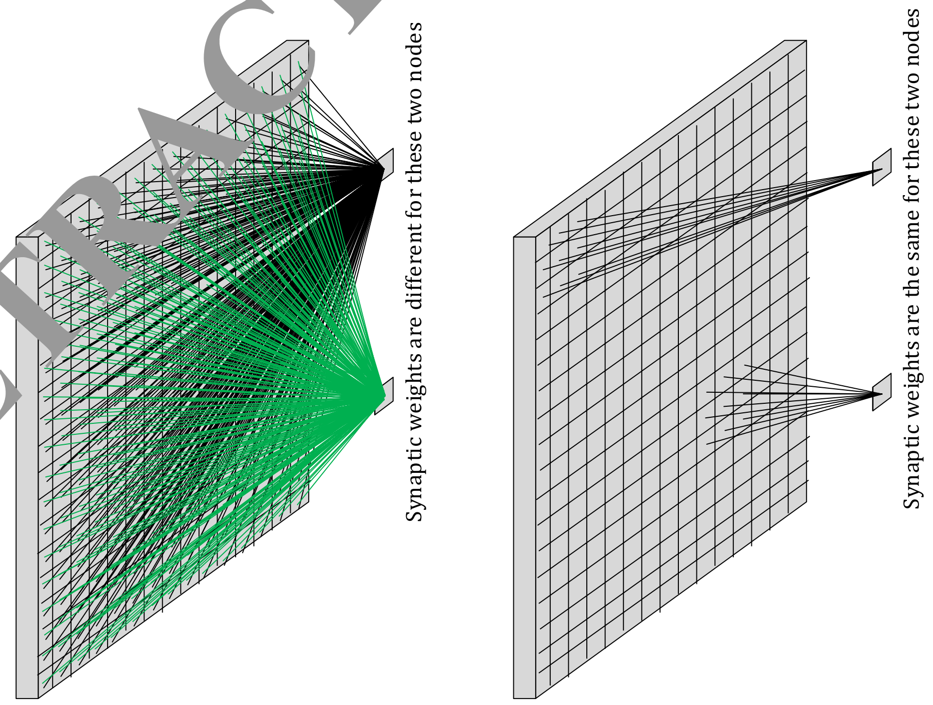
Synaptic weights are different for these two nodes
Synaptic weights are the same for these two nodes

Fig. 8 Each node in the first hidden layer has its own synaptic weights in SNN (left) while nodes share the same synaptic weights in CNN (right)

RETRACTED ARTICLE

the first hidden layer. This is referred to as weight sharing property and is depicted in Fig. 8. Despite this explanation concerned the first hidden layer, it is true for all convolutional hidden layers. This property gives CNNs two advantages over SNN. The first advantage is even less parameters for the machine to learn and the second is enabling the CNN to look for certain objects in the image, regardless of where in the image they are.

Conclusions and future directions

Classifying people in any manner requires care but predicting whether a person is a criminal demands even more caution and scrutiny and must be looked upon with suspicion. The danger of this technology lies in its imperfection, since misclassifying individuals can have grave repercussions. It would be too optimistic to claim that the 97% test accuracy, achieved by the CNN in this work, is easily generalizable to face shots from any other source. This is not only because of the small size of our dataset, but also the fact that criminal and non-criminal images come from different sources. Thus, the conditions under which the images are taken are not exactly the same, which raises the question, whether this disparity in peripheral conditions was captured by the deep classifier to unfairly distinguish between the two classes. In an ideal dataset, all face shots, criminal and non-criminal, would be taken with the same camera and under the same conditions, i.e. illumination, angle, distance, background, resolution, makeup, beard, hat, and glasses.

Facial emotions and age, major sources of bias in classifying facial images based on criminal tendency, were controlled in our work by eliminating non-neutral facial images and images of elderly and children. The bias due to background effects was mitigated by cropping the facial area out of images. The gender bias was not only eliminated by ignoring female images, but also measured and shown to be of little impact. Race, another source of bias, was not accounted for in this study because of our small dataset and the difficulty and occasionally subjectivity of identifying the race from low-quality facial images. However, both categories contain images of all races with roughly similar proportions. Enlarging our dataset, measuring the impact of racial bias, and detecting other personality traits form our future research venues.

Abbreviations
SNN: standard feedforward neural network; CNN: convolutional neural network; KNN: k nearest neighbors; SVM: support vector machines; ReLU: rectified linear unit; SGD: stochastic gradient descent.

Acknowledgements
Not applicable.

Authors' contributions
Both authors have contributed in developing the idea, documenting it, conducting the experiments, writing and revising. Both authors read and approved the final manuscript.

Funding
Not applicable.

Availability of data and materials
Not applicable.

Competing interests
The authors declare that they have no competing interests.

RETRACTED ARTICLE

Author details

[1] Department of Information Sciences and Technology, George Mason University, 4400 University Dr, Fairfax, VA 22030, USA. [2] College of Information Science and Technology, University of Nebraska at Omaha, 1110 S 67th St, Omaha, NE 68182, USA.

Received: 31 August 2019 Accepted: 30 December 2019

Published online: 07 January 2020

RETRACTED ARTICLE

References

1. Zebrowitz LA, Montepare JM. Social psychological face perception: why appearance matters. Soc Personal Psychol Compass. 2008;2(3):1497–517.
2. Lombroso C. Criminal man. 5th ed. Durham: Duke Univ. Press; 2006.
3. Hargrave J. Poker face: the art of analyzing poker tells. Dubuque: Kendall Hunt Pub Co; 2010.
4. Wang Y, Kosinski M. Deep neural networks are more accurate than humans at detecting sexual orientation from facial images. J Personal Soc Psychol. 2018;114(2):246–57.
5. Geng X, Yin C, Zhou ZH. Facial age estimation by learning from label distributions. IEEE Trans Pattern Anal Mach Intell. 2013;35(10):2401–12.
6. Reece AG, Danforth CM. Instagram photos reveal predictive markers of depression. EPJ Data Sci. 2017;6(1):15.
7. Ekman P, Friesen W. The facial action coding system (FACS): a technique for the measurement of facial action. Palo Alto: Consulting Psychologists; 1978.
8. Tsapatsoulis N, Karpouzis K, Stamou G, Piat F, Kollias S. A fuzzy system for emotion classification based on the MPEG-4 facial definition parameter set. In: 10th European signal processing conference; 2000. p. 1–4.
9. Oliver N, Pentland A, Bérard F. LAFTER: a real-time face and lips tracker with facial expression recognition. Pattern Recogn. 2000;33(8):1369–82.
10. Zhang Y, Ji Q. Active and dynamic information fusion for facial expression understanding from image sequences. IEEE Trans Pattern Anal Mach Intell. 2005;27(5):699–714.
11. Wu X, Zhang X. Automated inference on criminality using face images. 2016. arXiv preprint. arXiv:1611.04135.
12. Taneja A, Arora A. Modeling user preferences using neural networks and tensor factorization model. Int J Inf Manag. 2019;45:132–48.
13. Krizhevsky A, Sutskever I, Hinton GE. Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems; 2012. p. 1097–105.
14. Baranwal S, Khandelwal S, Arora A. Deep learning convolutional neural network for apple leaves disease detection. In: International conference on sustainability computing in science, technology & management; 2019; Jaipur, India. p. 260–7.
15. Ouyang W, Wang X, Zeng X, Qiu S, Luo P, Tian Y, Li H, Yang S, Wang Z, Loy CC, et al. Deepid-net: deformable deep convolutional neural networks for object detection. In: IEEE conference on computer vision and pattern recognition; 2015. p. 2403–12.
16. Sun Y, Liang D, Wang X, Tang X. Deepid3: face recognition with very deep neural networks. 2015. arXiv preprint. arXiv:1502.00873.
17. Cristani M, Vinciarelli A, Segalin C, Perina A. Unveiling the multimedia unconscious: Implicit cognitive processes and multimedia content analysis. In: The 21st ACM international conference on multimedia; 2013. p. 213–22.
18. Segalin C, Perina A, Cristani M, Vinciarelli A. The pictures we like are our image: continuous mapping of favorite pictures into self-assessed and attributed personality traits. IEEE Trans Affect Comput. 2017;8(2):268–85.
19. Segalin C, Cheng DS, Cristani M. Social profiling through image understanding: personality inference using convolutional neural networks. Comput Vis Image Underst. 2017;156(1):34–50.
20. NIST Special Database 18. 2010. https://www.nist.gov/srd/nist-special-database-18. Accessed 12 Apr 2019.
21. Viola P, Jones M. Rapid object detection using a boosted cascade of simple features. In: IEEE computer society conference on computer vision and pattern recognition; 2001. p. 1–9.
22. [illegible] Face Recognition Database. http://cbcl.mit.edu/software-datasets/heisele/facerecognition-database.html. Accessed 02 June 2018.
23. FEI Face Database. http://fei.edu.br/~cet/facedatabase.html. Accessed 02 June 2018.
24. Georgia Tech face database. http://www.anefian.com/research/face_reco.htm. Accessed 02 June 2018.
25. Face Place. http://wiki.cnbc.cmu.edu/Face_Place. Accessed 02 June 2018.
26. Face Detection Data Set and Benchmark Home. http://vis-www.cs.umass.edu/fddb/. Accessed 02 June 2018.
27. Hinton GE, Srivastava N, Krizhevsky A, Sutskever I, Salakhutdinov RR. Improving neural networks by preventing co-adaptation of feature detectors. 2012. arXiv preprint. arXiv:1207.0580.
28. Nair V, Hinton GE. Rectified linear units improve restricted boltzmann machines. In: The 27th international conference on machine learning; 2010. p. 807–14.
29. Kingma DP, Ba J. Adam: a method for stochastic optimization. 2014. arXiv preprint. arXiv:1412.6980.
30. Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. 2014. arXiv preprint. arXiv:1409.1556.
31. Wang M, Liu X, Wu X. Visual classification by l1-hypergraph modeling. IEEE Trans Knowl Data Eng. 2015;27(9):2564–74.
32. Yu J, Tao D, Wang M. Adaptive hypergraph learning and its application in image classification. IEEE Trans Image Process. 2012;21(7):3262–72.
33. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A. Going deeper with convolutions. In: IEEE conference on computer vision and pattern recognition; 2015. p. 1–9.

34. Zeiler MD, Fergus R. Visualizing and understanding convolutional networks. In: European conference on computer vision; 2014. p. 818–33.
35. Hashemi M, Hall M. Detecting and classifying online dark visual propaganda. Image Vis Comput. 2019;89(1):95–105.
36. Hashemi M. Enlarging smaller images before inputting into convolutional neural network: zero-padding vs. interpolation. J Big Data. 2019;6(1):98.
37. Farfade SS, Saberian MJ, Li LJ. Multi-view face detection using deep convolutional neural networks. In: 5th International conference on multimedia retrieval; 2015. p. 643–50.
38. Hashemi M. Web page classification: a survey of perspectives, gaps, and future directions. Multimedia Tools Appl. 2020. https://doi.org/10.1007/s11042-019-08373-8.
39. Zeiler MD, Fergus R. Stochastic pooling for regularization of deep convolutional neural networks. 2013. arXiv preprint. arXiv:1301.3557.
40. TensorFlow. TensorFlow Tutorials. 2019. https://www.tensorflow.org/tutorials. Accessed 01 Jan 2019.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

RETRACTED ARTICLE

Submit your manuscript to a SpringerOpen journal and benefit from:

► Convenient online submission
► Rigorous peer review
► Open access: articles freely available online
► High visibility within the field
► Retaining the copyright to your article

Submit your next manuscript at ► springeropen.com