Abstract—Facial attributes are emerging soft biometrics that have the potential to reject non-matches, for example, based on mismatching gender. To be usable in stand-alone systems, facial attributes must be extracted from images automatically and reliably. In this paper we propose a simple yet effective solution for automatic facial attribute extraction by training a deep convolutional neural network (DCNN) for each facial attribute separately, without using any pre-training or dataset augmentation, and we obtain new state-of-the-art facial attribute classification results on the CelebA benchmark. To test the stability of the networks, we generated adversarial images via a novel fast flipping attribute (FFA) technique. We show that FFA generates more adversarials than other related algorithms, and that the DCNNs for certain attributes are generally robust to adversarial inputs, while DCNNs for other attributes are not. This result is surprising because no DCNNs tested to date have exhibited robustness to adversarial images without explicit augmentation in the training procedure to account for adversarial examples. Finally, we introduce the concept of natural adversarial images, i.e., images that are misclassified but can be easily turned into correctly classified images by applying small perturbations. We demonstrate that natural adversarials commonly occur, even within the training set, and show that most of these images remain misclassified even with additional training epochs. This phenomenon is surprising because correcting the misclassification, particularly when guided by training data, should require only a small adjustment to the DCNN parameters.

I. INTRODUCTION

Facial attributes have several interesting properties from a recognition perspective: First, they are semantically meaningful to humans, which offers a level of interpretation beyond that achieved by most conventional recognition algorithms. This allows for novel applications, including descriptive searches (e.g., “Caucasian female with blond hair”) [5], [7], [12], verification systems [6], facial ordering [9], social sentiment analysis [15], and demographic profiling. Second, they provide information that is more or less independent of that distilled by conventional recognition algorithms, potentially allowing for the creation of more accurate and robust systems, narrowing down the search space, and increasing efficiency at match time. Finally, facial attributes are interesting due to their ability to convey meaningful identity information about a previously unseen face, e.g., not enrolled in a gallery or used to train a classifier.

Recent state-of-the-art approaches to facial attribute classification [8] have leveraged powerful deep learning models, which derive a generic feature space representation that is optimized for face identification or verification. They then use the truncated network to extract features upon which to train per-attribute binary classifiers. While this approach leads to a compact representation, it does not explicitly incorporate attribute information into the learnt feature space. While attributes that relate to facial identity may be implicitly captured by this representation, there is little evidence to suggest that the representation will effectively distill non-identity related attribute information, e.g., smiling. On the contrary, intuition suggests that a network trained to discriminate identities would learn to ignore non-identity related attributes. We hypothesize that by explicitly incorporating attribute information into deep learning representations, we can attain superior performance, especially for non-identity related attributes. We provide supporting evidence for this hypothesis by advancing the state-of-the-art on the CelebA benchmark – the largest publicly available attribute dataset – simply by using separate deep networks, each trained on raw attribute data. Moreover, unlike the former state-of-the-art approaches, we are able to attain such high performance without any form of data augmentation.
While our approach is state-of-the-art on the CelebA benchmark, recent research raises the question: *Does using pure end-to-end deep networks, i.e., not simply as feature extractors, but as attribute classifiers themselves, induce a risk of a non-robust attribute representations for real-world applications?* Specifically, Szegedy et al. [13] discovered that deep neural networks are susceptible to carefully chosen perturbations of even a few pixels. By adding such selected perturbations – that cannot be perceived by humans – to the original images, the resulting adversarial images become misclassified with high confidence.

Much research on adversarial images has been conducted since, and to our knowledge all these images have been easily generated independently on the dataset, network topology, training regime, hyperparameter choice, and activation type. In our experiments we attempt to generate adversarial images over a random subset of the CelebA dataset [8] using the fast gradient sign (FGS) method [3], and a new algorithm for introducing adversarial images – the fast-flipping attribute (FFA) algorithm – that efficiently leverages backpropagation without requiring groundtruth labels. We find that for both FGS and FFA attribute classifications are difficult to change, at least for some of the attributes, and that the number of adversarial images increases during training.

To date, adversarial images have been presented as inputs under the presence of slight artificial perturbations where the original input is correctly classified and the adversarial input is misclassified. In this paper, we pose the reverse question: “do there exist naturally misclassified inputs on which we can induce small artificial perturbations to correct the classification?”, or “Do adversarial images naturally occur?” We find that the answer is: yes, there are images in the training set which, even after they are used for training an attribute classifier, are still misclassified by that classifier, but can be flipped to the proper classification via an imperceptible perturbation. Further, we find that even with additional training, most of these natural adversarials are *not* learnt by the networks. This is surprising because correctly learning these natural adversarials should require only a minor tweak to the network parameters.

In Fig. 1 we show two examples of adversarial images that occurred in our experiments. Fig. 1(a) contains a natural adversarial image of a man (left) that is misclassified as a woman, where a small perturbation (center) applied to the image (right) would correct the classification. The contrary is shown in Fig. 1(b), where a correctly classified young person (left) is turned into an old person (right) by adding a small perturbation (center). Note that the real perturbations are much smaller than shown in Fig. 1, for displaying purposes we magnified the pixel changes by a factor of 50. Since perturbations can be positive and negative, gray pixels correspond to no change.

The contributions of this paper are as follows:

- We advance the state-of-the-art on the CelebA attribute classification benchmark by reducing the average classification error with more than 25% relative improvement.
- We generate adversarial images for each of the attribute networks and find that our facial attribute networks are initially robust to adversarial image generation, but they loose robustness during training.
- We introduce the notion of natural adversarial images and analyze their prevalence using our data and networks. We find that the frequency of naturally occurring adversarial images is quite large, accounting for nearly 85% of the training set images that are incorrectly classified by the network.
- We introduce the fast flipping attribute (FFA) algorithm for adversarial image generation and demonstrate that it is successful at flipping attribute classifications.

**II. Attribute Classification**

The automatic classification of facial attributes was first pioneered by Kumar et al. [6]. Their classifiers depended heavily on face alignment and each attribute used learnt combinations of features from hand-picked facial regions (e.g., cheeks, mouth, etc.). The feature spaces consisted of various simple normalizations and aggregations of color spaces and image gradients. Different features were learnt for each attribute, and one RBF-SVM per attribute was independently trained for classification.

More recent approaches leverage deep convolutional neural networks (DCNNs) to extract features. Liu et al. [8] use three DCNNs – a combination of two localization networks (LNets), and an attribute recognition network (ANet) to first localize faces and then classify facial attributes. The localization networks propose the location of the face, while the attribute network, which is first trained on external data to identify people and then fine-tuned using all attributes, is used to extract features, which are finally fed to independent linear SVMs for the final attribute classification. This approach is the current state-of-the-art on the CelebA dataset.

Contrary to [8] where the feature space is not necessarily attribute derived, our approach uses a DCNN trained for each attribute, from scratch, only on the CelebA training set. We use the network output directly as a classification result without requiring any secondary classifiers.

**A. Network Classification**

For our attribute classifiers, we adopted the 16 layer VGG network topology from [10], with two modifications. First, we altered the dimension of the RGB image input layer from $224 \times 224$ pixels to $178 \times 218$ pixels, the resolution of the aligned CelebA images. Second, we replaced the final softmax layer with a Euclidean loss function on the labels. We chose an Euclidean loss function as opposed to softmax, sigmoid, or hinge loss, because intuitively, attributes lie along a continuous range, while sigmoids tend to enforce saturation and hinge-loss enforces a large margin, neither of which is consistent with our intuition.
For an image $x$ with label $y \in \{-1, +1\}$ indicating the absence or presence of an attribute, respectively, let $f(x)$ be the DCNN classification decision. Then the loss $J$ is:

$$J(\theta, x, y) = ||f(x) - y||^2,$$

where $\theta$ are the parameters of the DCNN model.

To maintain comparability with other research reporting on the same dataset, and since the dataset only contains binary attribute labels, we decided to apply a classification function atop the network output that was trained with Euclidean loss. For input $x$, the classification result $c(x)$ and its corresponding accuracy $c(x, y)$ are obtained by thresholding $f(x)$ at 0:

$$c(x) = \begin{cases} +1 & \text{if } f(x) > 0 \\ -1 & \text{otherwise} \end{cases}, \quad c(x, y) = \begin{cases} +1 & \text{if } y \cdot c(x) > 0 \\ 0 & \text{otherwise} \end{cases}.$$

The classification error over the whole dataset of $N$ images $X_n$ including their attribute labels $Y_n$ is then given by:

$$e(X, Y) = \frac{1}{N} \sum_{n=1}^{N} (1 - c(X_n, Y_n)).$$

### B. Experiments

We conducted a comparison of our separate per-attribute neural networks with other attribute algorithms on theCelebA dataset [8], CelebA consists of more than 200K images, which show faces in a variety of different facial expressions, occlusions and illuminations, and poses from frontal to full profile. 160K images are used for training, and the remaining 40K images are equally split up into validation and test sets. Each image is annotated with binary labels of 40 facial attributes. We conducted our evaluation using the set of pre-cropped face images included in the dataset, which are aligned using hand-annotated key-points.

Due to memory limitations, we set the training batch size to 64 images per training iteration and, hence, the training requires approximately 2500 iterations to run a full epoch on the training set. In opposition to [10], we do not incorporate any dataset augmentation or mirroring, but train the network purely on the aligned images. We selected a learning rate of 0.00001. During training we update the DCNN weights using backpropagation with an RMSProp update rule and an inverse learning rate decay policy. Using the GPU implementation of Caffe [4], we trained all 40 networks until convergence on the validation set, which occurred between 2 and 10 epochs, depending on the attribute.

A comparison of our results on theCelebA test set with the original Face Tracer approach by Kumar et al. [6] as well as the LNets+ANet state-of-the-art approach of Liu et al. [8], both of which are taken from [8], is shown in Fig. 2. Due to the highly biased distributions of attribute labels in the CelebA dataset, we also included a Trivial algorithm, which simply predicts the class with the higher occurrence in the training set. For some attributes such as Attractive or Male (which are approximately balanced in the test set), the Trivial classifier obtains high errors, while for attributes like Narrow Eyes or Double Chin, the Trivial classifier even outperforms the previous state-of-the-art approach. However, our approach is at least able to beat the Trivial approach for all attributes, which is not true for any of the other algorithms.

Our approach yields a mean classification error of 9.20 %, a relative improvement of 27.5 % over the state-of-the-art (12.70 % classification error) and 51 % of improvement over the Face Tracer system (18.88 % classification error). Interestingly, we are “only” 54 % better (in terms of relative improvement) than the Trivial system that obtains 19.96 % mean classification error.

**Fig. 2: Attribute Classification Error on CelebA.** This figure shows the classification errors on the test set of the CelebA dataset for Our algorithm compared to three other algorithms, sorted based on the result of Ours. The Trivial approach simply assigns each attribute the score of the majority class. The results of Face Tracer and LNets+ANet are taken from Liu et al. [8].
classification error. For certain attributes, especially those not related to face identity (e.g., Wearing Necklace, Wearing Earrings, Blurry), our approach dramatically advances the state-of-the-art. L Nets-A Net outperforms our approach only for a few attributes, but never by more than a percentage point in classification error.

III. ADVERSARIAL IMAGES FOR ATTRIBUTES

An adversarial image is an image which looks very close to (and is generally indistinguishable from) an original image from the perspective of a human observer, but differs dramatically in classification by a machine learnt classifier. Multiple techniques have been proposed to create adversarial examples. The first reliable technique [15] uses a box-constrained optimization (L-BFGS). Starting with a randomly chosen modification, it aims to find the smallest perturbation in the input space that causes the perturbed image to be classified as a predefined target label. Baluja et al. [1] proposed generating affine perturbations, applying them to input samples, and then observing how models respond to these perturbed images. While, from an adversarial perspective, this approach has the advantage of not requiring internal network representations, it relies on “guess and check”, i.e., it creates random perturbations and determines if the results are misclassified, which can be prohibitively expensive.

Goodfellow et al. [3] introduced a more efficient algorithm to produce adversarial perturbations. Their fast gradient sign (FGS) method creates perturbations by using the sign of the gradient of loss with respect to the input. FGS is more efficient than L-BFGS because the required gradient can be effectively calculated by using a single backpropagation. Experiments demonstrate that FGS reliably causes a wide variety of learning models to misclassify their perturbed inputs, including “shallow” models, but that deep networks are especially susceptible [3]. Note that FGS is based directly upon network information: The gradient of loss defines the direction, while a quick search is used to determine the magnitude necessary to make the perturbed input adversarial with minimal deviation from the original.

Although the definitions of adversarial example vary [13], [3], [1], at their core, adversarials are modified inputs formed by imperceptible non-random perturbations that are misclassified by machine learning models. Hence, humans should not even perceive differences between adversarial examples and their originals. To formalize the definition, let $x_i$ be an input image correctly classified as $y_i$. An adversarial perturbation $\eta_i$ is given if the perturbed image $\tilde{x}_i = x_i + \eta_i$ is not classified as $y_i$:

$$f(x_i) = y_i \quad \text{and} \quad f(\tilde{x}_i) \neq y_i. \quad (4)$$

This is a necessary but not a sufficient condition as the modification needs to be imperceptible. Various measures such as $L_1$, $L_2$, and $L_\infty$ distances have been used to show how close the perturbed images are to their originals. However, these measures are not well matched to human perception [11] as they are very sensitive to even small geometric distortions, and therefore do not measure similarity in a psychophysical sense. On the other hand, the structural similarity (SSIM) index [14] measures similarity $S(\tilde{x}, x)$ based on structural and brightness differences, which we believe to be better suited to measure the degree of adversarial. Since humans are sensitive to changes in faces, we adopt the SSIM threshold $\tau = 0.95$ as a cutoff for adversarial because it is a good threshold for imperceptible differences [2].

A. Adversarial Imaging Techniques

We explore two approaches for generating the necessary perturbations $\eta_i$. Goodfellow et al. [3] introduced the fast gradient sign (FGS) method to find adversarial perturbations. Given an image $x_i$, FGS searches for perturbations causing mislabeling using the sign of the gradient of loss:

$$\eta_i^{\text{fgs}} = w_i \text{ sign}(\nabla x_i J(\theta,x_i,y_i)). \quad (5)$$

FGS takes steps in the direction that is defined by the gradient of loss in order to invert the score and cause mislabeling. This requires knowledge of the label $y_i$ of the image $x_i$. We introduce a novel approach – the fast flipping attribute (FFA) algorithm – that directly relies on the (binary) classification scores. We postulate that inverting the classification score and calculating the gradient of the inverted score with respect to the input image will eventually provide a direction where adversarial perturbations can be found. Formally:

$$\eta_i^{\text{ffa}} = -w_i \frac{\partial f(x_i)}{\partial x_i}. \quad (6)$$

can be obtained by backpropagating the inverted classification score from the decision layer, i.e., the layer that calculates $f(x_i)$.

By using FGS or FFA, we can obtain varying perturbations $\eta_i$ with respect to a given input image $x_i$ among the defined directions. To effectively search along those directions for the smallest perturbations that cause classification errors, we apply a line-search technique with increasing step-sizes to quickly reach the weight $w_i$ in Eqns. (5) and (6) that causes mislabeling. When the line-search oversteps, we employ a bisection to determine the smallest possible adversarial perturbation. Finally, please note that the generated adversarial images have rounded discrete pixel values in range $[0,255]$.

B. Natural Adversarial Images

As of today, adversarial images have been artificially generated via a computational process, but no one has yet addressed whether adversarial inputs occur in natural images: are there noticeably misclassified images for which infinitesimal changes to the inputs yield correct classifications? If so, this has tremendous ramifications on the sensitivity and robustness of decision boundaries. We seek to explore whether adversarial images naturally occur, and if so, how often. Thus, we formalize the novel concept of natural adversarial images. Let $x'_i$ be an incorrectly classified image whose correct label is $y_i$. Then $x'_i$ is a natural adversarial image if there exists
In this paper we have used a simple and effective method to train deep convolutional neural networks (DCNNs) to perform binary facial attribute classification. Experiments on the CelebA dataset show that with 9.20% average classification error our approach is able to outperform the current state-of-the-art (12.70%). This performance gain is statistically significant, resulting in $p < 10^{-27}$ in a paired t-test. Afterward, we found that the DCNNs are able to learn the adversarials that exist after two epochs, and the total number of adversarial images would decrease. Especially for the natural adversarial images, i.e., the images that were misclassified by the DCNNs after 2 epochs for which imperceptible modifications to those images would make them correct, we assumed that further training of the networks would learn these examples. However, what we found is counter-intuitive. First, for FFA, the total number of adversarial images over all attributes that we were able to create increased from 14957 to 25240, while the majority of images (14802) were in both sets (which includes images that were misclassified before and now classified correctly and vice versa). Interestingly, this trend is comparable both for the correctly and incorrectly classified images. For example, the total number of natural adversarial images for FFA was 4272 after 2 epochs and 5312 with converged networks, with an overlap of 3381 images. Hence, about 80% of the adversarial images over all attributes that we were able to create increased from 14957 to 25240, while the majority (888) were natural adversarials were not corrected with additional training epochs. At least we found that of the 891 images that were natural adversarials after 2 epochs and that are not natural adversarials were found is counter-intuitive. First, for FFA, the total number of adversarial images over all attributes that we were able to create increased from 14957 to 25240, while the majority of images (14802) were in both sets (which includes images that were misclassified before and now classified correctly and vice versa). Interestingly, this trend is comparable both for the correctly and incorrectly classified images. For example, the total number of natural adversarial images for FFA was 4272 after 2 epochs and 5312 with converged networks, with an overlap of 3381 images. Hence, about 80% of the natural adversarial images were not corrected with additional training epochs. At least we found that of the 891 images that were natural adversarials after 2 epochs and that are not natural adversarial images were classified correctly by the converged networks. Note that the numbers for FGS are similar.

**C. Experiments**

To test and compare adversarial generation with FGS and FFA, we randomly selected 1000 images of the CelebA training set and performed experiments trying to flip attributes. For each attribute and both the correctly and incorrectly classified images, we counted the number of times, in which an adversarial image could be computed, i.e., where an $\eta_i$ exists for which $S(x_i, x_i + \eta_i) < \tau$, for an $\eta_i$ generated by either of the two algorithms. The results per attribute can be obtained in Fig. 3. Interestingly, for some attributes such as Big Nose or Young, every input image can be turned into an adversarial, while for others like Double Chin or Rosy Cheeks, adversarial images were found for only few images. Even more astonishingly, incorrectly classified images can be turned into adversarials more often as correctly classified images. Also, the number of images, for which we could generate adversarial images using FFA is generally higher than for FGS, where almost all images that spawned FGS adversarials also spawned FFA adversarials.

In an attempt to test in which stage of the network training more adversarial images exist, we also tried to generate adversarial images for the same 1000 examples for DCNNs that were trained for two epochs only. Intuition would suggest that the DCNNs are able to learn the adversarials that exist after two epochs, and the total number of adversarial images would decrease. Especially for the natural adversarial images, i.e., the images that were misclassified by the DCNNs after 2 epochs for which imperceptible modifications to those images would make them correct, we assumed that further training of the networks would learn these examples. However, what we found is counter-intuitive. First, for FFA, the total number of adversarial images over all attributes that we were able to create increased from 14957 to 25240, while the majority of images (14802) were in both sets (which includes images that were misclassified before and now classified correctly and vice versa). Interestingly, this trend is comparable both for the correctly and incorrectly classified images. For example, the total number of natural adversarial images for FFA was 4272 after 2 epochs and 5312 with converged networks, with an overlap of 3381 images. Hence, about 80% of the natural adversarial images were not corrected with additional training epochs. At least we found that of the 891 images that were natural adversarials after 2 epochs and that are not natural adversarial images were classified correctly by the converged networks. Note that the numbers for FGS are similar.

**IV. Discussion and Conclusion**

In this paper we have used a simple and effective method to train deep convolutional neural networks (DCNNs) to perform binary facial attribute classification. Experiments on the CelebA dataset show that with 9.20% average classification error our approach is able to outperform the current state-of-the-art (12.70%). This performance gain is statistically significant, resulting in $p < 10^{-27}$ in a paired t-test. Afterward,
we have introduced the fast flipping attribute (FFA) algorithm, a fast and robust method to generate adversarial images by flipping the binary decision of the DCNN. We have shown that FFA can create more adversarials than the related fast gradient sign (FGS) method, but it has the limitation – by design – that it can only be applied to binary classification networks.

In Sec. III we demonstrated that a greater number of training epochs makes DCNNs more vulnerable to adversarial images. On the other hand, after 2 epochs the DCNNs obtain an average classification error of 9.78\% that is statistically significantly \( p < 10^{-10} \) (in the paired T-test) higher than the 9.20\% that we obtained with the converged DCNNs. Hence, either a trade-off must be found between vulnerability and classification success, or the network training must be made more stable against adversarials, both of which we leave for future research.

In Sec. II we chose an uncommon loss function, i.e., the Euclidean loss to train our DCNNs. Readers might ask, what happens with a more common loss function. To answer that question, we conducted some experiments, training attribute classification networks for 2 epochs using a softmax loss. Compared to Euclidean loss after 2 epochs, the DCNNs trained with a softmax loss ended up with a slightly higher classification error of 9.98\%, which is not statistically significant \(( p \approx 0.01 \) in the paired t-test). However, we also found that the DCNNs trained with softmax loss are more vulnerable to adversarial images. Using 1000 examples from the CelebA training set (cf. Sec. III), we were able to generate 10878 adversarials (over all 40 attributes) using FGS on Euclidean loss DCNNs, and 14458 using FGS on softmax loss DCNNs. Since other researchers have shown that DCNNs trained with softmax loss are generally vulnerable to adversarial images, we assume that the same trend, i.e., that more epochs make the DCNNs more vulnerable, also applies for softmax DCNNs, but we have not yet performed experiments to support that assumption.

**Acknowledgment**

This research is based upon work supported in part by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), via IARPA R&D Contract No. 2014-14071600012. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the ODNI, IARPA, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon.

**References**

[1] S. Baluja, M. Covell, and R. Sukthankar. The virtues of peer pressure: A simple method for discovering high-value mistakes. In Computer Analysis of Images and Patterns, pages 96–108. Springer, 2015.

[2] J. R. Flynn, S. Ward, J. Abich IV, and D. Poole. Image quality assessment using the ssim and the just noticeable difference paradigm. In Engineering Psychology and Cognitive Ergonomics. Understanding Human Cognition, pages 23–30. Springer, 2013.

[3] I. J. Goodfellow, J. Shlens, and C. Szegedy. Explaining and harnessing adversarial examples. In International Conference on Learning Representation (ICLR), 2015. arXiv preprint arXiv:1412.6572.

[4] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. In Proceedings of the ACM International Conference on Multimedia, pages 675–678. ACM, 2014.

[5] N. Kumar, P. Bellhumeur, and S. Nayar. Facetracer: A search engine for large collections of images with faces. In European Conference on Computer Vision, pages 340–353. Springer, 2008.

[6] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar. Attribute and simile classifiers for face verification. In International Conference on Computer Vision, pages 365–372. IEEE, 2009.

[7] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar. Describable visual attributes for face verification and image search. IEEE Transactions on Pattern Analysis and Machine Intelligence, 33(10):1962–1977, 2011.

[8] Z. Liu, P. Luo, X. Wang, and X. Tang. Deep learning face attributes in the wild. In International Conference on Computer Vision, pages 3730–3738. IEEE, 2015.

[9] D. Parikh and K. Grauman. Interactively building a discriminative vocabulary of nameable attributes. In Conference on Computer Vision and Pattern Recognition, pages 1681–1688. IEEE, 2011.

[10] O. M. Parkhi, A. Vedaldi, and A. Zisserman. Deep face recognition. British Machine Vision Conference, 1(3):6, 2015.

[11] S. Sabour, Y. Cao, F. Fah shri, and D. J. Fleet. Adversarial manipulation of deep representations. In International Conference on Learning Rep resentation (ICLR), 2016. arXiv preprint arXiv:1511.05122.

[12] W. J. Scheirer, N. Kumar, P. N. Belhumeur, and T. E. Boult. Multi-attribute spaces: Calibration for attribute fusion and similarity search. In Conference on Computer Vision and Pattern Recognition, pages 2933–2940. IEEE, 2012.

[13] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus. Intriguing properties of neural networks. In International Conference on Learning Representation (ICLR), 2014. arXiv preprint arXiv:1312.6199.

[14] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. Image Processing, IEEE Transactions on, 13(4):600–612, 2004.

[15] Z. Zhang, P. Luo, C.-C. Loy, and X. Tang. Learning social relation traits from face images. In International Conference on Computer Vision, pages 3631–3639. IEEE, 2015.