2021-08

DeepFake Detection with Inconsistent Head Poses: Reproducibility and Analysis

Lutz, Kevin; Bassett, Robert

ArXiv

http://hdl.handle.net/10945/69345

DeepFake Detection with Inconsistent Head Poses: Reproducibility and Analysis

Kevin Lutz
Naval Postgraduate School
1 University Circle, Monterey, CA 93943
kevin.lutz@nps.edu

Robert Bassett
Naval Postgraduate School
1 University Circle, Monterey, CA 93943
robert.bassett@nps.edu

Abstract

Applications of deep learning to synthetic media generation allow the creation of convincing forgeries, called DeepFakes, with limited technical expertise. DeepFake detection is an increasingly active research area. In this paper, we analyze an existing DeepFake detection technique based on head pose estimation, which can be applied when fake images are generated with an autoencoder-based face swap [34]. Existing literature suggests that this method is an effective DeepFake detector, and its motivating principles are attractively simple. With an eye towards using these principles to develop new DeepFake detectors, we conduct a reproducibility study of the existing method. We conclude that its merits are dramatically overstated, despite its celebrated status. By investigating this discrepancy we uncover a number of important and generalizable insights related to facial landmark detection, identity-agnostic head pose estimation, and algorithmic bias in DeepFake detectors. Our results correct the current literature's perception of state of the art performance for DeepFake detection.

1. Introduction

Detecting manipulated media is a research priority for both public and private sector institutions. This broad interest reflects the fact that visual misinformation has the potential to cause grave damage in a number of areas, including financial and political systems. In recent years, applications of deep learning to image manipulation have led to the emergence of *DeepFakes*, which lower the technological barriers required to create high-quality manipulations. The accessibility of this technology has led to a proportional response from researchers attempting to distinguish authentic from manipulated media, with the result that many promising methods for detecting DeepFakes have recently been developed. Though the term "DeepFake" is often used as a universal descriptor for any synthetic media generated wholly or partially with deep learning, in this paper we will focus on images which are created with an autoencoder-based face swap.

One particularly promising technique for detecting DeepFakes is based on finding head pose inconsistencies in modified images [34]. This method, which we henceforth refer to as the *analytic*, was demonstrated by its authors to classify fake from real images with high accuracy. Moreover, its motivating principle is attractively simple, in contrast to many other methods for detecting DeepFakes which lack interpretation. These merits have led to the analytic becoming extremely popular, garnering nearly 200 citations in the two years since its publication.

In an attempt to objectively assess the utility of this analytic for DeepFake detection, we conduct an in-depth analysis of its underlying methodology, in addition to a reproducibility study. We find a stark lack of generalizability of the analytic's performance to any data other than those originally included in the paper. Moreover, we are able to *explain* this lack of generalizability by identifying a number of incorrect assumptions from the original manuscript. The first of these is related to the utility of estimated head poses as a feature. We show that, surprisingly, pose estimates contain enough information to identity unique individuals. It follows that, when trained on faces which exhibit similar structure (for example, due to a shared ethnicity or gender), this analytic can exhibit algorithmic bias because of its tendency to classify images as authentic or manipulated based on the facial structure of its subject. Another incorrect assumption we identify relates to the head pose estimation itself. We show that, without additional modifications, iterative methods for pose estimation tools often fail to accurately estimate head poses because of their tendency to get stuck in a strong local minimum. The analytic demonstrates this behavior, resulting in head pose estimates which are frequently very poor. We propose a simple correction which avoids this local minimum. Although it does not remedy the performance issues of the analytic, we expect this contribution will have utility in other contexts which require head pose estimation as a computational primitive.

The rest of this paper is organized as follows. In the next subsection we outline related work, and in section 2 we re-

view the analytic. Section 3 introduces and then refutes four assumptions necessary to the analytic's rationale. In section 4, we conduct a set of numerical experiments on a variety of DeepFake data sets. We conclude with section 5, where we summarize our contributions and their implications.

1.1. Related Work

One of the most common tools for generating DeepFakes is the autoencoder-based face swap [1, 2, 3, 4, 5]. We will focus on this technique exclusively in this paper, though other techniques, such lip-sync or puppet master, also exist for creating DeepFakes [7, §2.1]. To perform a face swap between two individuals, faces must be detected, aligned, and segmented from a collection of images of each individual. Then, an autoencoder network is adversarially trained on the aligned faces, where a single encoder network is used for both individuals but a unique decoder is used for each. After training, one can swap the face of the first individual into an image of the second as follows. First, an image of the second individual is compressed using the encoder. To perform the swap, the encoded image of the second individual is decoded using the decoder trained on the first individual. Afterwards, postprocessing is an important final step to eliminate visual artifacts where the swapped inner region and the original outer portion of the face meet [24].

Motivated by the convincing nature of modern DeepFakes, many methods have been proposed to detect them from authentic images. These methods can be partitioned by the features they use to detect the manipulated image. Though not specific to DeepFake detection, methods which are designed to detect image splicing, where a part of a source image is placed into a target image, can be applied for this purpose. Examples include [27], where an autoencoder is trained to partition a face into spliced and unmanipulated regions, and [18], where image metadata is leveraged to detect inconsistencies and generate a splice mask for various regions of an image. In [32], the authors take a different perspective by formulating a hypothesis test for each pixel, using Z-scores constructed for various neighborhoods of the pixel as a test statistic. In [37], the authors detect spliced regions using steganalysis features unique to the processing of individual cameras by comparing these features in two different regions of an image.

Another large class of techniques trains a classifier to detect images generated by either convolutional or generative adversarial networks. Examples include [17], which uses an expectation maximization algorithm (EM) to detect patterns of correlation among pixel neighborhoods. The correlation patterns are used to classify whether an image was generated by a convolutional network. Similarly, [31] shows that images generated by a large suite convolution-based generators (including DeepFakes) can be detected by training a ResNet model on images from only a single generator, which suggests that these images exhibit common structure. Considering GANs instead of convolutional networks, in [35] the residual of an autoencoder-based image reconstruction is deemed a "GAN fingerprint" and used to classify GAN-generated from authentic images. Along similar lines, [36] uses a discrete Fourier transform to detect artifacts of the upsampling procedure used by many GAN-based image generators.

Some methods for classifying authentic from manipulated images use deep learning, but do not focusing on specific features of the DeepFake generation process. Examples include [6], which emphasizes using a lower number of layers in its architecture, and [28], which uses a capsule network as opposed to more traditional convolutional layers.

A final class of techniques uses features derived from knowledge that the image contains a human face. In [23], manipulated videos are detected by noting the lack of natural blinking patterns. More recent DeepFakes circumvent this by including images where with a blinking subject in the training data. The authors of [26] detect manipulated images using inconsistencies in the eye and teeth regions, where lack of detail is common. In [8], the authors consider the problem of determining whether a video contains a certain individual, such as a public figure. The individual's facial mannerisms are captured using a correlation matrix of facial actions, such as eyebrow or chin raising, across frames in a video, and this correlation matrix is used as features in an SVM to determine authentic from manipulated videos. In follow-up work [7], the authors use a convolutional network to capture facial mannerisms, which avoids the labor-intensive process of hand-crafting the correlation-based features. Finally, [12] estimates blood volume changes that occur due to a rhythmic heartbeat, and uses this information to distinguish authentic from synthetic videos. The analytic we consider in this paper falls into this final class of techniques that leverage the fact that the image contains a human subject, because its method for distinguishing real from manipulated images utilizes the structure of the human face.

2. Review of Analytic

In this section we will review the analytic, which detects a DeepFake image or video frame using inconsistencies between the inner (swapped) region of a face and the outer (unaltered) region. The analytic computes two 3D head pose estimates, one using only the central region of the face and the other its entirety, and using various features derived from these head poses classifies an image as either manipulated or authentic. The authors of the analytic claim that, because only the inner portion of a DeepFake image is manipulated, inconsistencies can be detected this way, despite not being detectable to a human observer [34, §1].

In this section, we review the main steps of the analytic: detecting faces, locating facial landmarks, estimating 3D head poses, and training an SVM classifier on the resulting head poses. In each of these steps, we pay special attention to the techniques used and the assumptions required for valid application.

2.1. Detecting Faces

Given an image to be classified, the first step in the analytic is to detect faces in the image. To do so, the analytic uses a Histogram of Oriented Gradients (HOG) [15] face detector as implemented in the `dlib` computer vision library [21]. If no faces are detected in the image, which frequently occurs when faces are not oriented towards the camera, the analytic exits without making a prediction.

2.2. Locating Facial Landmarks

Once a face has been detected, the analytic estimates the position of 68 facial landmarks, 51 of which are in the inner region of the face. Thirty of the interior facial landmarks are not used by the analytic, like those outlining the lips and eyes, because they are dynamic and do not indicate the position and orientation of the head relative to the camera. To locate the facial landmarks, the analytic applies a well-regarded technique for landmark estimation [20], which uses an ensemble of gradient-boosted regression trees to find the landmarks' positions. Because of the important role it plays in the analytic, we will describe this landmark detection technique in more detail.

We focus our attention on the application of a pretrained landmark estimation model, since this reflects its usage in the analytic. The landmark detection technique refines, for a fixed number of iterations T, an initial estimate $\hat{S}_0 \in \mathbb{R}^{68\times 2}$ of the (x, y) pixel coordinates for the 68 facial landmarks. At each iteration, k, the landmark positions are updated as

$$\hat{S}_{k+1} = \hat{S}_k + \gamma\, r_k(\hat{S}_k, I)$$

where

- $I \in \mathbb{R}^{m\times n}$ is a gray-scale representation of the image or region of interest, with dimensions $m \times n$.
- $\left\{r_k : \mathbb{R}^{68\times 2} \times \mathbb{R}^{m\times n} \to \mathbb{R}^{68\times 2}\right\}_{k=1}^{T}$ is a sequence of random forest regression functions.
- $\gamma \in (0, 1)$ is a *shrinkage parameter*, commonly used in boosted forests to mitigate overtraining.

Each random forest r_k acts on $\hat{S}_k$ and I using only a finite number of points in I, the location of which are fixed relative to $\hat{S}_k$'s deviation from a set of mean facial landmarks. Specifically, split nodes in the decision trees of r_k are based on the criteria

$$I\left(A(\hat{S}_k)u\right) - I\left(A(\hat{S}_k)v\right) > \tau$$

where $\tau \in \mathbb{R}$ and $u, v \in \mathbb{R}^2$ are fixed at each split node, and $A : \mathbb{R}^{68\times 2} \to \mathbb{R}^{2\times 2}$ is a function mapping the current landmark estimates $\hat{S}_k$ to a 2×2 matrix. This matrix is a similarity transform that maps from a set of mean facial landmarks to $\hat{S}_k$.

We note a couple aspects of this landmark detection technique which are relevant to its application in the analytic. First, the training data for the ensemble are exclusively unmanipulated faces and their landmarks, so that its performance on DeepFakes remains to be validated. Second, the landmark refinements rely on an ensemble of decision trees, with split nodes depending on the current landmark estimate and a pair of grey-scale pixels u and v. These points are optimally selected from a random sample during training, with the important consequence that u and v need not respect the face's local geometry. For example, u and v both on a subject's chin have the potential to affect landmark refinements around the nose. This does not respect the analytic's assumption that the estimated position of outer landmarks will be minimally affected by the manipulation of the inner facial region, a point which we revisit in section 3.

2.3. Estimating 3D Head Poses

After the 68 facial landmarks have been estimated, the analytic uses a pinhole camera model, combined with some assumptions about the camera's focal length and 3D geometry of the subject's head, to estimate the head's orientation and position relative to the camera. Two of these estimates are taken, one using all the facial landmarks and the other using those in the inner region of the face. In this section we summarize this pose estimation procedure.

In order to estimate the head pose of a subject using a set of landmarks, the analytic considers the error between the set of 2D facial landmarks detected in the image and a set of 3D facial landmarks from an "average" model of the human face [9], which are projected into the image using the pose estimate and the geometry of the pinhole camera model. The subject's head pose is estimated by minimizing the error between the projected 3D landmarks and the detected 2D ones over all possible head poses.

Let $\mathcal{L}$ index a collection of landmarks. For our purposes, $\mathcal{L}$ will either index all of the landmarks or those in the inner region of the face. Denote by $\{(x_i, y_i)\}_{i\in\mathcal{L}}$ the landmarks detected in the image and $\{(U_i, V_i, W_i)\}_{i\in\mathcal{L}}$ the corresponding landmarks from an average 3D model of the human face. The 3D facial landmarks are given in a basis with the face's center as the origin and eyes looking in the positive z direction. The analytic then estimates the head pose of the subject relative to the camera by finding the optimal rotation matrix R and translation vector t such that the $\{(U_i, V_i, W_i)\}_{i\in\mathcal{L}}$ coordinates align with the landmarks

detected in the image. That is, the analytic minimizes

$$\sum_{i \in \mathcal{L}} \left\| s \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} - \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \left(R \begin{bmatrix} U_i \\ V_i \\ W_i \end{bmatrix} + t \right) \right\|^2 \quad (1)$$

over rotation matrices $R \in \mathbb{R}^{3\times3}$, translation vectors $t \in \mathbb{R}^3$, and projective scaling $s \in \mathbb{R}$. In this problem, the upper triangular matrix is the *camera matrix* which projects a 3D object into pixel coordinates. The f_x and f_y values denote the camera's focal length multiplied by the pixel density in the x and y dimensions, respectively. Together, (c_x, c_y) give the principal point of the image in pixel units. In summary, the problem in (1) minimizes the projection error between the image landmarks and the 3D coordinates as projected into the image. For more information on the geometry of the pinhole camera model we refer the reader to [16].

The head pose estimation problem in (1) relies on a number of important assumptions. First, one might expect that detecting inconsistencies in inner facial landmarks requires estimating the position of these landmarks to a reasonably high accuracy, and using an average 3D model of the human face as a substitute for the subject's face may undermine that accuracy. Second, the camera matrix must be estimated because these physical properties of the camera cannot be derived from a single image. When the image considered is of resolution $m \times n$, the analytic approximates the principal point with the center of the image, $c_x = m/2$, $c_y = n/2$, and the focal length terms with $f_x = f_y = m$. Third, computing minima in (1) is nontrivial because it requires optimizing over the nonconvex set of 3D rotation matrices, and the analytic assumes the head poses it computes are accurate. The analytic circumvents this concern by using a well-regarded method to solve this problem, as implemented in `opencv` [10]. This optimization algorithm initializes values of R and t by computing a Direct Linear Transform, in which the rotation matrix constraint on R is relaxed. The relaxed solution is projected onto the set of rotation matrices, after which an iterative Levenberg-Marquardt algorithm is applied to further reduce the objective function [22].

2.4. Training the Classifier

The final step in the analytic uses features derived from the head pose estimates–an R matrix and t vector for both inner and all landmark pose estimates–to classify an image as DeepFake or authentic. The analytic uses a support vector machine with radial basis function kernel to perform the classification. In their original paper, the authors test various features constructed from the R matrices and t vectors of the inner and all landmark estimates, ultimately concluding that the flattened difference between the R matrices for each head pose estimate, with the difference in t vectors appended, yields superior classification results. The γ parameter in the radial basis function kernel $K(x, y) = \exp(-\gamma\|x - y\|^2)$ is fixed at $1/(\text{number of features})$, and the SVM model is trained to output class probabilities using Platt scaling [29].

3. Problematic Assumptions

In this section we introduce a number of methodological issues related to the analytic. We do so by formulating and then refuting various, often implicit, assumptions justifying its use.

3.1. Landmark Estimates

The motivating principle behind the analytic–that DeepFakes introduce inconsistency between the inner and outer regions of the face–relies on the following assumption.

Assumption 1. *Landmark estimates in the inner region of the face are affected by manipulations of that region, while landmark estimates outside of the inner region are minimally affected.*

This assumption makes the restriction to inner landmarks meaningful, because it implies that only the inner landmark estimates will change in a response to a splice in the inner facial region. Though this assumption seems intuitive, the landmark estimation technique the analytic uses is not designed for use on manipulated images. In the context of authentic images, it is *desirable* for a landmark estimation method to predict realistic landmark estimates despite inconsistencies in certain regions of the face. These inconsistencies often occur due to obstructions, such as sunglasses or hair styles. Figure 1 demonstrates this point, where the estimation method from the analytic overcomes facial obstructions to produce landmark estimates which appear to be consistent with unobstructed regions of the face. This suggests that the assumed insensitivity of outer landmark estimates to manipulations in the inner face is in conflict with the design goals of landmark estimation in general.

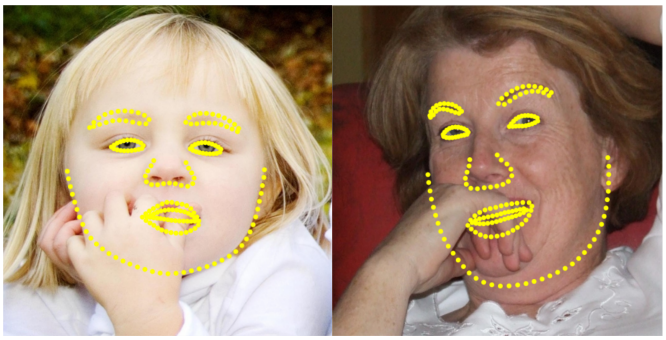

Figure 1. The analytic's landmark estimates are designed to overcome locally inconsistent regions, such as facial obstructions. Source: [20].

To test the validity of Assumption 1, we estimate the facial landmarks in an image where the inner region of the

face is obviously and dramatically inconsistent with the outer region. Figure 2 gives a manipulated image, with an authentic image for comparison. The inner region of the manipulated image is shrunk beyond reasonable proportions, while the outer region of the face is left unaltered. The outer landmark estimates do not extend to the outer region of the face as they do in the authentic image, because the predictive model produces landmark estimates similar to the authentic landmarks used to train the model. These results undermine Assumption 1, and suggest that landmark detectors designed for authentic images may not reliably detect inconsistencies in manipulated images.

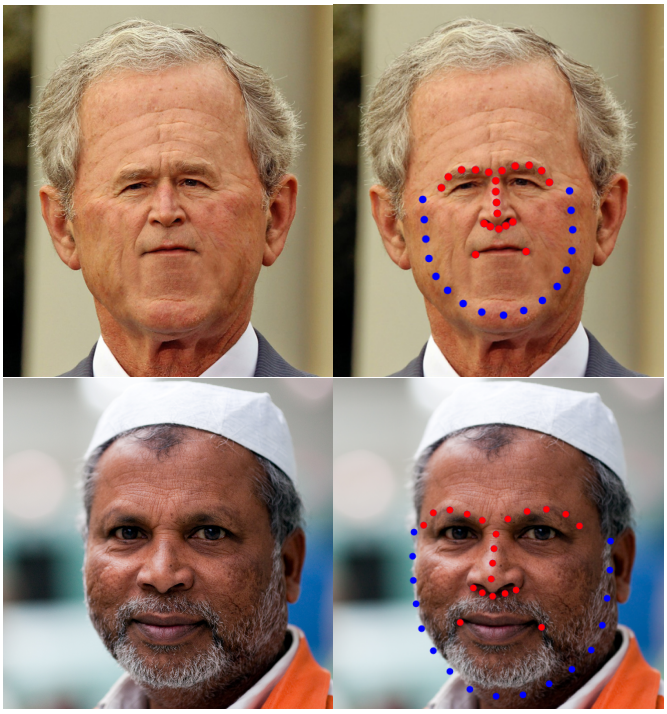

Figure 2. (Top) Manipulated inner face (red) leads to outer landmarks (blue) which are incorrectly located. (Bottom) Authentic image and landmarks. Sources: [11] and [19].

3.2. Head Pose Optimization

For a given collection of 2D landmarks detected in the image and 3D landmarks from a model of the typical human face, minimizing (1) over R, t, and s yields an estimate of the head pose. The head's orientation is given by rotation matrix R, and the head's location by the translation vector t. This is a special case of the classical *Perspective-n-Point* (PnP) problem, which estimates an object's pose using 3D points, their projections into the image, and a known camera matrix. Because of its important role in various computer vision applications, many algorithms have been introduced to solve the PnP problem. By relying on the accuracy of an existing and well-reputed solver for the PnP problem [25], the analytic assumes the following.

Assumption 2. *The rotation matrix R and translation vector t obtained from minimizing* (1) *are accurate estimates of the head's true orientation and position relative to the camera.*

We show next that general purpose PnP solvers should not be applied to human faces without special consideration for their symmetry. These observations demonstrate that the analytic's Assumption 2 does not hold.

Consider, for some choice of R and t, the transformed 3D landmarks and their projections into the image.

$$\begin{bmatrix} X_i \\ Y_i \\ Z_i \end{bmatrix} := R \begin{bmatrix} U_i \\ V_i \\ W_i \end{bmatrix} + t$$

$$\begin{bmatrix} \hat{x}_i \\ \hat{y}_i \end{bmatrix} = \begin{bmatrix} \frac{X_i}{Z_i} f_x + c_x \\ \frac{Y_i}{Z_i} f_y + c_y \end{bmatrix}$$

Note that $\hat{x}_i$ and $\hat{y}_i$ are invariant with respect to the transformation $(X_i, Y_i, Z_i) \to (-X_i, -Y_i, -Z_i)$. By increasing R's yaw rotation by π and adjusting the z-component of t, we can approximately perform this transformation up to a slight deviation that can be attributed to the 3D landmarks' deviation from planarity. In this way we can construct, for each choice of R and t, an estimate with nearly identical projection error.

This observation has important consequences for the validity of Assumption 2, because it shows that the optimization problem (1) has at least two local minima with similar squared projection error. Figure 3 shows the transformed 3D landmarks in their estimated poses, for both the inner and all landmark estimates. These two pose estimates demonstrate the two local minima of (1). Even though the pose estimate using all landmarks suggests that the landmarks are visible through the back of the subject's head, this solution is consistent with the provided landmarks and the geometry of the pinhole camera model.

Further experiments demonstrate that the analytic frequently uses head poses which suggest the image was captured through the back of the subject's head. We refer to these poses as *flipped*, and refer to a pair of head pose estimates with one correct and one flipped as *conflicting* head pose estimates. Table 1 demonstrates that, for the University of Albany Deep Fake Video (UADFV) data used by its authors to demonstrate the analytic's performance, 96% of video frames in the training set had at least one flipped pose estimate and 33% had conflicting pose estimates. The high proportion of flipped poses used in the analytic invalidates Assumption 2.

Detecting a flipped head pose estimate is straightforward by examining the sign of Z_i, the z component of the trans-

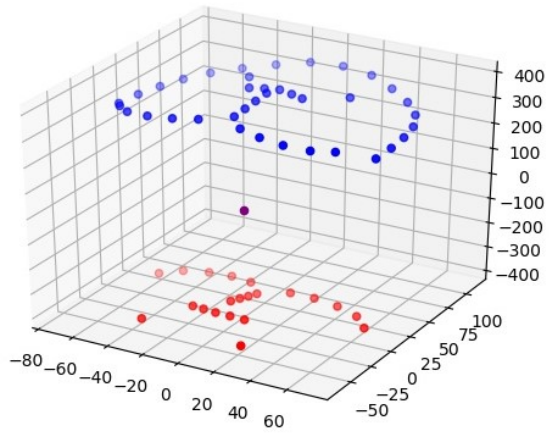

Figure 3. Oriented 3D landmarks for both inner (red) and all (blue) landmark estimates, with the pinhole camera aperture in purple at the origin. In this example, the conflicting head pose estimates illustrate the two local minima of (1).

		inner landmark	
		flipped	correct
all landmark	flipped	.63	.28
	correct	.05	.04

Table 1. The proportion of head pose estimates in the UADFV training set which exhibit a flipped pose.

formed 3D landmarks. Alternatively, because the 3D landmarks have $W_i \approx 0$, the sign of the translation vector's z component will suffice. Figure 3 illustrates this point; the correct head pose has $Z_i < 0$, whereas the flipped pose has $Z_i > 0$ for all landmarks. Furthermore, we can modify the head pose estimation algorithm from 2.3 to avoid the local minimum of a flipped head poses by checking for $Z_i > 0$ and correcting the yaw and translation vector. This step can be naturally incorporated immediately after projecting the DLT solution onto the set of rotation matrices, or after some amount of Levenberg-Marquardt iterations have been applied. We opt for the latter, and refer to this as the corrected version of the analytic in the remainder.

One implication of the analytic's use of flipped head pose estimates is that any trends which correlate with conflicting pose estimates will be easy to identify. For the UADFV data set, manipulated frames are four times more likely to have conflicting pose estimates than authentic frames. Since the features used in the SVM are the flattened differences of the rotation matrices and translation vectors, conflicting pose estimates are easy for the classifier to identify.

3.3. Overtraining and Algorithmic Bias

Though certain methods for DeepFake detection are concerned with deciding whether an image or video contains a person of interest, this analytic does not make such a restriction. As such, the analytic should generalize to individuals not used in the training set.

Assumption 3. *The identity of the image's subject has no bearing on its predicted label.*

One aspect of the analytic that invites further scrutiny is the data used to demonstrate its performance. The UADFV data set used by the analytic's authors swaps a single individual's face–that of actor Nicolas Cage–into all of the manipulated images. A simple test shows that this results in a model trained to detect images of Nicolas Cage and not DeepFakes. Figure 4 shows an ROC curve for a test set of 111 images, all of which are authentic, where 56 of the images contain Nicolas Cage and 55 contain other celebrities. Applying the authors' pretrained model to these images shows that the analytic reliably identifies authentic images of Nicolas Cage as manipulated.

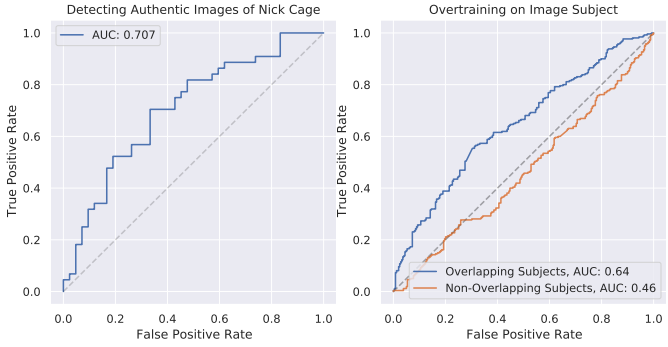

Figure 4. (Left) The authors' pretrained model classifies authentic images of Nicolas Cage as manipulated because he is depicted in all of the manipulated images of the UADFV training set. (Right) Performance of two models on images from DFDC indicate that the analytic overtrains based on the depicted individual. In the Overlapping Subjects model, subjects appear in either manipulated or authentic images across both the training and testing sets. In the Non-Overlapping Subjects model, the training and testing sets feature disjoint subjects.

A follow-up experiment confirms the analytic's tendency to overtrain based on identity of the image's subject. Figure 4 contains ROC curves for two train/test splits of video frames from the DeepFake Detection Challenge (DFDC) data set [13]. In the first split, each subject appears in either manipulated or authentic images, but not both, across the training and testing sets. The second split is like the first, except that the sets are additionally restricted so that testing and training images have no subjects in common. We perform the experiment using a modification of the analytic that includes the head pose correction detailed in Section 3.2, but note that the uncorrected results are similar and culminate in the same conclusion. The results indicate

that the model AUC increases by .18 when the context in which a subject appears (manipulated or authentic images) persists across the training and testing sets. Moreover, when the subjects in the testing set are disjoint from the training image, the model's performance is roughly equivalent to choosing the predicted label uniformly at random.

These results demonstrate that Assumption 3 is invalid, because the analytic demonstrates the tendency to overtrain based on the identity of the subjects in an image. This is especially concerning because of the analytic's potential to classify images based on shared facial structure as those in the training set, which may be due to attributes like age, race, or gender of the images' subjects.

3.4. DeepFakes Exhibit Inconsistent Head Orientation

In order for the analytic to distinguish DeepFakes from authentic images, the distribution of head poses should differ between real and manipulated images.

Assumption 4. *The distribution of head poses differs between DeepFakes and authentic images.*

In the paper introducing the analytic, the authors claim that, with R_c and R_a denoting the estimated rotation matrices for the central landmark and all landmark estimates, respectively, the cosine distances of the head orientation vector can be used to separate DeepFakes from authentic images. That is, for $w = [0, 0, 1]^\top$,

$$1 - \frac{\langle R_a^\top w, R_c^\top w \rangle}{\|R_a^\top w\| \, \|R_c^\top w\|} \tag{2}$$

separates DeepFakes and authentic images. We find this claim, and the more general Assumption 4, to be unreproducible across all the data sets considered, and regardless of whether the uncorrected or corrected analytic is used. Additionally, due to the occurrence of flipped poses, we find that the cosine distance metric is not as effective as the authors intend since the metric does not detect the difference in yaw rotation between conflicting pose estimates. Figure 5 shows the histograms of the cosine distances between head orientation vectors for various DeepFake data sets introduced in section 4.

Additional investigations yield no evidence of separation between the DeepFake and authentic image classes for any of the twelve underlying estimated parameters (roll, pitch, yaw, and a 3D translation vector for both head pose estimates) for any of the data sets in table 3, when considering either the original head pose estimation procedure or our corrected version. Moreover, table 2 shows that the correlation of flipped head poses with DeepFakes in UADFV does not hold in general. The exact cause of this correlation in UADFV is unknown, but we conjecture that certain faces

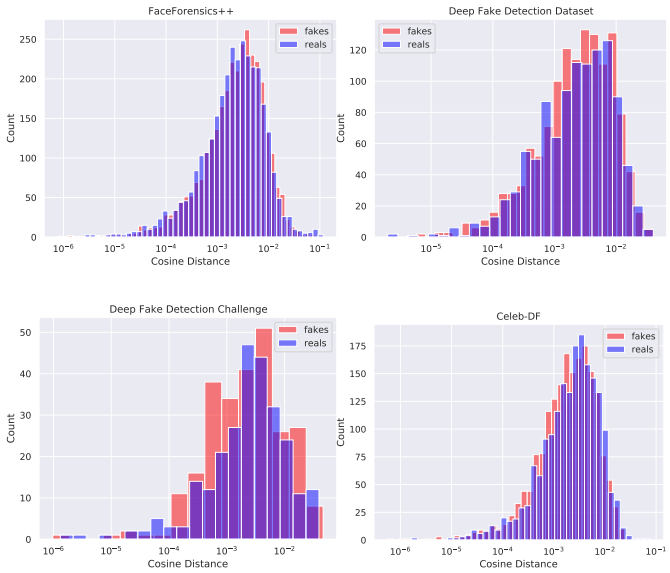

Figure 5. Histograms of cosine distances between "inner" and "all" head orientation vectors. Contrary to the analytic's Assumption 4, this feature does not separate DeepFakes from authentic images.

	$P(\text{fake}\|\text{conflicting})$	$P(\text{fake}\|\text{non-conflicting})$
UADFV	.78	.38
DARPA	.64	.39
FF++	.47	.51
DFDD	.55	.54
DFDC	.49	.50
Celeb-DF	.49	.51

Table 2. The empirical probability of an image being manipulated given that the pose estimates are conflicting. In this table, "conflicting" denotes the case that exactly one of the pose estimates is flipped, which is easily distinguishable using the translation vector feature of the SVM. The relationship between conflicting pose estimates and DeepFakes in UADFV and DARPA-GAN does not generalize.

(i.e. that of Nicolas Cage) are more susceptible to the local optimum in (1) than others. These observations suggest that Assumption 4 may not hold, and brings the utility of features related to estimated head poses into question.

4. Reproducibility Study

This section contains a large scale reproducibility study aimed at measuring the analytic's ability to generalize to new data sets. The various problematic assumptions discussed in section 3 suggest that, despite its celebrated status, many of the analytic's motivating principles are flawed. This section continues this investigation by comparing its predictive performance on different collections of DeepFakes.

First, we note that we are able to reproduce the authors'

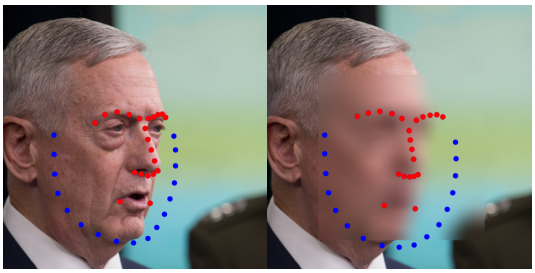

Figure 6. Blurry faces, commonly found in low-quality DeepFakes, make landmark estimation unreliable. Source: [19].

results on the UADFV data set used in the original paper, using both the authors' pretrained model and one which we trained ourselves using the authors' training split. Both models have a near-perfect AUC of approximately .98. Correcting for the flipped head poses from section 3.2 reduces the performance of these models to .88, as the model cannot use UADFV's correlation between flipped head poses and fake images observed in table 2. However, the problematic nature of the UADFV data set discussed in the previous sections prompts us to look at other data sets to test the analytic's utility as a DeepFake detector.

In addition to UADFV, the analytics' authors also consider another data set, a subset of images from the DARPA GAN challenge. However, the authors acknowledge that the faces shown in these DeepFakes are often extremely blurry, which makes landmark estimation unreliable. Figure 6 illustrates this point, with the landmark estimates degrading as the blur increases in the facial region. In addition, table 2 demonstrates that the DARPA data exhibit a similar correlation between conflicting head poses and fake images as UADFV, a trend which does not generalize. We also find that the DARPA data set has many near-duplicate images, suggestive of consecutive video frames, which in combination with its lower sample size (< 500 images) greatly undermines its quality. Though we are able to reproduce the authors' results on the DARPA GAN data set, we do not consider it further because of these limitations.

We compare the analytic's performance on four modern DeepFake data sets: FaceForensics++ (FF++) [30], Deep Fake Detection Dataset (DFDD) [14], Deep Fake Detection Challenge (DFDC) [13], and Celeb-DF [24]. Table 3 gives the number of randomly selected frames in our training and testing sets for each data set. Because of the analytic's tendency to overtrain based on the identity of the image's subject, whenever the data allows we construct a random split of the data such that subjects are disjoint across the training and testing sets. The only data set which does not permit such a split is DFDD, which features 28 actors in manipulated and authentic videos, acting in a number of scenarios. For this data set, we randomly partitioned into training and testing set by scenario, while also ensuring that each actor occurs as both an authentic and manipulated subject in at least one video from each of the training and testing sets. In our experience, the analytic is sensitive to imbalanced data, so each set contains an equal number of authentic and manipulated images.

	Training Frames	Testing Frames
FF++	6392	1608
DFDD	2632	372
DFDC	600	600
Celeb-DF	3766	944

Table 3. Details of Train/Test Splits

	Provided	Uncorrected	Corrected
FF++	.52	.60	.63
DFDD	.46	.47	.28
DFDC	.54	.45	.45
Celeb-DF	.50	.58	59

Table 4. AUC scores for various models. The "Provided" model is pretrained on UADFV and provided by the authors [33]. The "Uncorrected" model is trained on the training frames from table 3, without the head pose correction in section 3.2. The "Corrected" model is also trained on the frames in table 3, but uses the corrected head pose model described in section 3.2.

Table 4 gives our results, where we see that the analytic's performance is approximately what would be expected by assigning a predicted label to each image uniformly at random. The catastrophic performance of the corrected model on the DFDD data suggests that the analytic is again overtraining on the image's subject, despite our attempts to mitigate it. The corrected model appears to be especially sensitive to overtraining because it estimates the head pose more accurately than the uncorrected model.

5. Conclusion

In this work we show that a celebrated method for detecting DeepFakes is not reproducible. This study extends well beyond a set of numerical experiments, because we introduce a number of methodological flaws associated with using head pose estimates to detect manipulated images. We demonstrate that the features derived from this procedure contain information about the identity of the subjects contained in the training set, which raises privacy concerns for resulting classifiers. We also show that the approximate planarity of the facial landmarks considered leads to a strong local optimum in the 3D pose estimation problem, which skewed the originally presented results.

We hope that our contributions will allow the performance of new DeepFake detectors to be properly interpreted, in addition to encouraging the development of these detectors in more productive directions.

References

[1] DeepFaceLab github. https://github.com/iperov/DeepFaceLab.

[2] DFaker github. https://github.com/dfaker/df.

[3] faceswap-GAN github. https://github.com/shaoanlu/faceswap-GAN.

[4] faceswap github. https://github.com/deepfakes/faceswap.

[5] FakeApp. https://www.malavida.com/en/soft/fakeapp/.

[6] Darius Afchar, Vincent Nozick, Junichi Yamagishi, and Isao Echizen. Mesonet: a compact facial video forgery detection network. In *2018 IEEE International Workshop on Information Forensics and Security (WIFS)*, pages 1–7. IEEE, 2018.

[7] Shruti Agarwal, Tarek El-Gaaly, Hany Farid, and Ser-Nam Lim. Detecting deep-fake videos from appearance and behavior. *arXiv preprint arXiv:2004.14491*, 2020.

[8] Shruti Agarwal, Hany Farid, Yuming Gu, Mingming He, Koki Nagano, and Hao Li. Protecting world leaders against deep fakes. In *CVPR Workshops*, pages 38–45, 2019.

[9] Tadas Baltrusaitis, Amir Zadeh, Yao Chong Lim, and Louis-Philippe Morency. Openface 2.0: Facial behavior analysis toolkit. In *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*, pages 59–66. IEEE, 2018.

[10] Gary Bradski. The opencv library. *Dr Dobb's J. Software Tools*, 25:120–125, 2000.

[11] @CelebsWith. George bush with a tiny face. https://twitter.com/CelebsWith/status/425398141136154624, Jan 20, 2014. Accessed: Feb 26, 2021.

[12] Umur Aybars Ciftci, Ilke Demir, and Lijun Yin. Fakecatcher: Detection of synthetic portrait videos using biological signals. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[13] Brian Dolhansky, Joanna Bitton, Ben Pflaum, Jikuo Lu, Russ Howes, Menglin Wang, and Cristian Canton Ferrer. The deepfake detection challenge dataset. *arXiv preprint arXiv:2006.07397*, 2020.

[14] Nick Dufour and Andrew Gully. Deepfake detection dataset. https://ai.googleblog.com/2019/09/contributing-data-to-deepfake-detection.html. Accessed: 2020-10-15.

[15] Pedro F Felzenszwalb, Ross B Girshick, David McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *IEEE transactions on pattern analysis and machine intelligence*, 32(9):1627–1645, 2009.

[16] David A Forsyth and Jean Ponce. *Computer vision: a modern approach*. Pearson,, 2012.

[17] Luca Guarnera, Oliver Giudice, and Sebastiano Battiato. Deepfake detection by analyzing convolutional traces. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 666–667, 2020.

[18] Minyoung Huh, Andrew Liu, Andrew Owens, and Alexei A Efros. Fighting fake news: Image splice detection via learned self-consistency. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 101–117, 2018.

[19] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4401–4410, 2019.

[20] Vahid Kazemi and Josephine Sullivan. One millisecond face alignment with an ensemble of regression trees. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1867–1874, 2014.

[21] Davis E. King. Dlib-ml: A machine learning toolkit. *Journal of Machine Learning Research*, 10:1755–1758, 2009.

[22] Kenneth Levenberg. A method for the solution of certain non-linear problems in least squares. *Quarterly of applied mathematics*, 2(2):164–168, 1944.

[23] Yuezun Li, Ming-Ching Chang, and Siwei Lyu. In ictu oculi: Exposing ai created fake videos by detecting eye blinking. In *2018 IEEE International Workshop on Information Forensics and Security (WIFS)*, pages 1–7. IEEE, 2018.

[24] Yuezun Li, Xin Yang, Pu Sun, Honggang Qi, and Siwei Lyu. Celeb-df: A large-scale challenging dataset for deepfake forensics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3207–3216, 2020.

[25] David G Lowe et al. Fitting parameterized three-dimensional models to images. *IEEE transactions on pattern analysis and machine intelligence*, 13(5):441–450, 1991.

[26] Falko Matern, Christian Riess, and Marc Stamminger. Exploiting visual artifacts to expose deepfakes and face manipulations. In *2019 IEEE Winter Applications of Computer Vision Workshops (WACVW)*, pages 83–92. IEEE, 2019.

[27] Huy H Nguyen, Fuming Fang, Junichi Yamagishi, and Isao Echizen. Multi-task learning for detecting and segmenting manipulated facial images and videos. In *2019 IEEE 10th International Conference on Biometrics Theory, Applications and Systems (BTAS)*, pages 1–8. IEEE, 2019.

[28] Huy H Nguyen, Junichi Yamagishi, and Isao Echizen. Use of a capsule network to detect fake images and videos. *arXiv preprint arXiv:1910.12467*, 2019.

[29] John Platt et al. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. 1999.

[30] Andreas Rossler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. Faceforensics++: Learning to detect manipulated facial images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1–11, 2019.

[31] Sheng-Yu Wang, Oliver Wang, Richard Zhang, Andrew Owens, and Alexei A Efros. Cnn-generated images are surprisingly easy to spot... for now. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8695–8704, 2020.

[32] Yue Wu, Wael AbdAlmageed, and Premkumar Natarajan. Mantra-net: Manipulation tracing network for detection and localization of image forgeries with anomalous features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9543–9552, 2019.

[33] Xin Yang, Yuezun Li, and Siwei Lyu. Exposing deep fakes using inconsistent head poses (implementation). `https://bitbucket.org/ericyang3721/headpose_forensic/`. Accessed: 2020-10-15.

[34] Xin Yang, Yuezun Li, and Siwei Lyu. Exposing deep fakes using inconsistent head poses. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8261–8265. IEEE, 2019.

[35] Ning Yu, Larry S Davis, and Mario Fritz. Attributing fake images to gans: Learning and analyzing gan fingerprints. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7556–7566, 2019.

[36] Xu Zhang, Svebor Karaman, and Shih-Fu Chang. Detecting and simulating artifacts in gan fake images. In *2019 IEEE International Workshop on Information Forensics and Security (WIFS)*, pages 1–6. IEEE, 2019.

[37] Zheng Zhao, Penghui Wang, and Wei Lu. Detecting deepfake video by learning two-level features with two-stream convolutional neural network. In *Proceedings of the 2020 6th International Conference on Computing and Artificial Intelligence*, pages 291–297, 2020.