Faking it, Making it: Fooling and Improving Brain-Based Authentication with Generative Adversarial Networks

Tanya Piplani
School of Information
UC Berkeley
tanyapiplani@berkeley.edu

Nick Merill
School of Information
UC Berkeley
ffff@berkeley.edu

John Chuang
School of Information
UC Berkeley
john.chuang@berkeley.edu

Abstract

In this paper, we empirically demonstrate the vulnerability of a passthought authentication system to fake signals generated by Generative Adversarial Networks (GANs), and use these same signals to make authenticators more robust. We first train a classifier that is able to authenticate a subject based on their EEG signals. The classifier performs a binary classification task: either the user is who they claim to be, or not. To test the robustness of the authenticator against attacks we train a GAN to generate signals that mimic the EEG signals of the "positive" subject. We find that a well-trained GAN is able to generate signals that the classifier consistently accepts. To alleviate this vulnerability, we re-train the classifier with this GAN-generated data. We find that the classifier re-trained against synthetic data is both more robust against this attack, and more accurate in accepting real data than the initial classifier. We conclude with recommendations for the design of passthought authentication systems.

1. Introduction

Passthoughts is an authentication scheme that uses neural signals to log users into devices and services [18, 17, 12]. Users think a secret passthought to authenticate themselves and, since these passthoughts are expressed differently from person to person, the scheme combines both inherence and knowledge factors in a single step [9]. With the extensive use and ubiquity of wearable and mobile computing devices, special hardware such as a custom-made earpiece can act as a possession factor, and extend the scheme[17].

While prior work has established that passthoughts are robust against spoofing attacks (e.g., in the case of a compromised passthought), little work has examined how susceptible passthoughts may be to spoofing attacks generated from a corpus of neural data. In this threat model, an attacker with knowledge of the algorithm's training set may be able to generate realistic-looking data without using a replay attack (i.e., without using training set examples to authenticate) as using same set of signals for authenticating multiple times might not be allowed by such systems. Let us describe the following attack scenario.

1. We have a passthought authentication system with N enrolled users.
2. The authentication system works by training a classifier using EEG data from these *N* users to identify the correct user as "positive" and all others as "negative".
3. An attacker steals some EEG data from one of the *N* enrolled users.
4. The attacker uses the stolen EEG data to generate some synthetic EEG signals.
5. The attacker submits the synthesized signal to attempt to log in as the target.

In this study, we use a generative adversarial network (GAN) [6] to generate convincing neural data, and test these generated examples on a trained passthought authentication classifier. We find that our classifier incorrectly accepts our GAN-generated examples, indicating that prior work on passthoughts may have been susceptible to such an attack. However, as a follow-up study, we train a passthoughts classifier using GAN-generated data as negative examples, and find that this process improves the classifier's accuracy overall. We discuss recommendations for designing passthought authentication systems that are both accurate and robust against GAN attacks.

2. Background

User authentication systems can be described by three factors: knowledge, something only known to the user; possession, a unique physical object and inherence, a unique

[0] 978-1-5386-1737-3/18/$31.00 ©2018 IEEE

identifier of a person. We need a multi-factor authentication method which utilizes these factors, but minimizes the frustration occurred by multiple steps (e.g., entering a password, then entering a separate code from one's phone). To assist with the usability issues surrounding multi-factor authentication, passthoughts aims to provide two factors of authentication in a single step. A single mental task, or passthought, provides both a knowledge factor (a chosen secret thought) with an inherence factor (the way that thought is expressed for an individual) [3, 9]. Using a custom sensing device, passthoughts could provide an additional possession factor, all in the same step.

GANs [6] have been used in the past for various biometric tasks like face image generation also from polarimetric thermal images [20], or generation of images preconditioned on brain signals [14]. Hence, we try to use such a GAN approach to intrinsically model the underlying probability distribution of a given dataset, and use it to improve existing authentication algorithms to make them more robust to "fake" signals.

For security, classifiers trained for user authentication need to be highly accurate. While it is important to have a high number of true acceptances, and correspondingly a low number of false rejections, it is imperative to minimize false acceptances, as false acceptances form the most disastrous error in terms of security of a user authentication system. Thus we wanted to make sure that the classifiers that are being used for user authentication give as few false acceptances as possible. To ensure this, we wanted to give the classifier the most difficult negative examples for training and allow it to learn features very specific to the "positive" label. This will enable a classifier that only labels an example as "positive" when it is highly certain about its decision at the same time not increasing the number of false negatives at a commensurate rate.

There are multiple challenges with training classifiers with EEG data. First, the measurement of neuronal electrical activity, most typically non-invasively via electrodes arranged on the scalp, have noise which makes them difficult to process. By generating synthetic data from a GAN that has learned from the underlying data distribution we provide an effective means of bypassing this noise in the data collection process [16, 11]. Second, due to the unavailability of large datasets, training of state-of-the-art machine learning algorithms that have a lot of parameters tend to overfit, further adding to the difficulty of training robust detection methods.

To help alleviate those problems, we attempt to use GANs to generate synthetic signals, which can augment the training set. We believe these signals could help alleviate the risk of over-fitting, and make the authenticator more resilient to attacks. The paper makes the following two contributions: (a) a demonstration that a passthought-based authentication system is vulnerable to fake signals generated by a GAN; (b) present an effective solution to eliminate this vulnerability by re-training the authenticator using the GAN generated signals.

3. Methods

In this section, we describe the two algorithms we used in our study. First, we describe the design of the classifier we used to authenticate users, which we trained on data collected from a recorded corpus of samples. We then describe the generative adversarial network (GAN) we used to generate *synthetic* signals: fake, but realistic, EEG signals based on the observations from our corpus.

3.1. Training the classifier to authenticate users

For the system to be able to authenticate the user, the classifier performs a binary classification task: either the user is who they claim to be, or not.

We use two datasets in our study. The first dataset S_1 or the "negative" samples consists of EEG signals collected from 30 subjects while they were presented with a 5-minute-long audio-video stimulus instructing them to perform a series of different mental tasks [8]. The EEG signals were collected using the Neurosky Mindwave Mobile [13], a consumer-grade single-channel EEG device with a sampling rate of 512Hz. We collected a second dataset S_2 or the "positive" samples that consists of EEG signals collected from one subject using the same Neurosky device and the same stimulus, once a day over a period of 58 consecutive days. For this second dataset, the EEG recordings continued beyond the end of the 5-minute stimulus for an additional 5 minutes, where the subject was free to perform any task on the computer, such as reading text or watching video on the screen.

The original signals were recorded at 512Hz. For training a simple GAN model as shown in Figure 1, we first converted the input raw EEG signals into a power spectrum. This was done by first taking the available data, passing it through a Fast Fourier Transform [5], taking the power amplitude of the signal and sub sampling it by a factor of 2. This allowed us to have a signal that for each data point consisted of 256 samples. The power amplitude signal consisted of values that were between 0 and A, where A is the max amplitude of the power signal. This signal was then normalized between 0 to 1 or -1 to 1 based on the non linearities (sigmoid, hyperbolic-tangent or other non-linear activation functions) used in the various tasks as described in the following sections. We drew 30,000 samples from the "negative" dataset S_1 and 40,000 samples from the "positive" dataset S_2. We randomly split the entire dataset such that 30% of the data is the test set and the remaining 70% is the train set.

We tried different binary classification approaches including Support Vector Machines [4], Neural Networks [15], XGBoost [2] with logistic regression learning task to establish a good baseline for our classifier. Looking at the empirical results, we found that the XGBoost classifier gave the best accuracy. Below are some of the most salient features of these classifiers that we trained.

3.1.1 Neural Network

For this model we trained a feed forward multilayer perceptron. The input layer of the neural network has 256 units, corresponding to the the samples of the datasets S_1 and S_2. The was a hidden layer and an output function which had sigmoid activations that predicted the probability of the sample belonging to the positive label. The probability of the label belonging to the negative class was defined such that the sum of probabilities of the positive and the negative classes is 1. Thus,

$$P(y = 1|X) = \sigma(WX + b)$$
$$p(y = 0|X) = 1 - P(y = 1|X) = 1 - \sigma(WX + b)$$

The loss function applied to the neural network was sigmoid cross entropy such that

$$L(X, y) = ylog\hat{y} + (1 - y)log(1 - \hat{y})$$

The training is done using stochastic gradient descent where parameters are iteratively updated given the gradient with respect to the loss function as follows:-

$$W = W - \eta \frac{\partial L(X, y)}{\partial dW}$$

Specifics about the training are provided in the section describing the results.

3.1.2 XGBoost with Logistic Regression learning task

We trained a XGBoost [2] tree boosting classifier with a logistic regression learning task. The model is very close to the structure of the neural network described above, but is void of any hidden layer representation. It is an extreme boosting algorithm and therefore provides a very strong ensemble learner as compared to a single tree for supervised learning problems. This allows us to see if the data follows a linear distribution such that a linear hyperplane in a d dimensional space can describe the dataset S_1 and S_2 well. The loss function to train the XGBoost model is also the Sigmoid Cross Entropy function described above in section 3.1.1.

3.2. Adversarial Training for Fake signal generation

This section explains how we trained a GAN for generating fake user signals, that we then use to make our classifier more robust. The following sections discuss the training of a Generative Adversarial Network for generation of “fake” signals for the positive user.

3.2.1 Generative Adversarial Networks

A method for unsupervised learning, GANs [6] have shown tremendous potential for learning about complicated distributions [6, 10]. A GAN consists of two neural networks, *D* and *G*. The network *G* is called a generator, and the network *D* is called a discriminator. In the simplest case, our data consists of a set $Z \in S_2$ of unlabeled data points (passthoughts EEG signals for a subject). The goal of the generator is to take random noise as an input and produce an output that “looks real”, as if it came from *Z*. The goal of the discriminator is to take an input and decide whether it came from a generator network or a real data set. We train these networks together, so that each network will enable the other to improve, with the end result that the generator learns to generate highly realistic outputs that consistently “fool” the discriminator.

3.2.2 Simple GAN

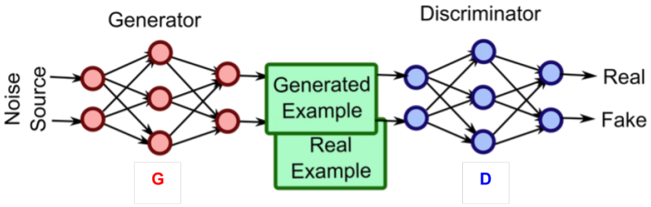

Figure 1. The structure of a Generative Adversarial Network. The red graph structure represents the generator neural network and the blue graph represents the discriminator neural network. A noise source is fed as input to the generator that transforms it into the output space representing the data distribution. It is then fed into the discriminator neural network along with a sample of the real data, which then tries to differentiate between the generated signals and the real data. The optimization is configured so that each of the generator and the discriminator want to make the task harder for the other. Hence over time, the generator gets better at generating fake signals.

The generator network is a small neural network with three layers. The input is a 100-dimensional noise vector z where $z \sim \mathbf{N}(0, 1)$. The final output of the network is a 256-dimensional vector that is in the same feature space as samples from the training data.

The activation function for the generator were chosen to be leaky rectified linear activation (PReLU) [19]. This ac-

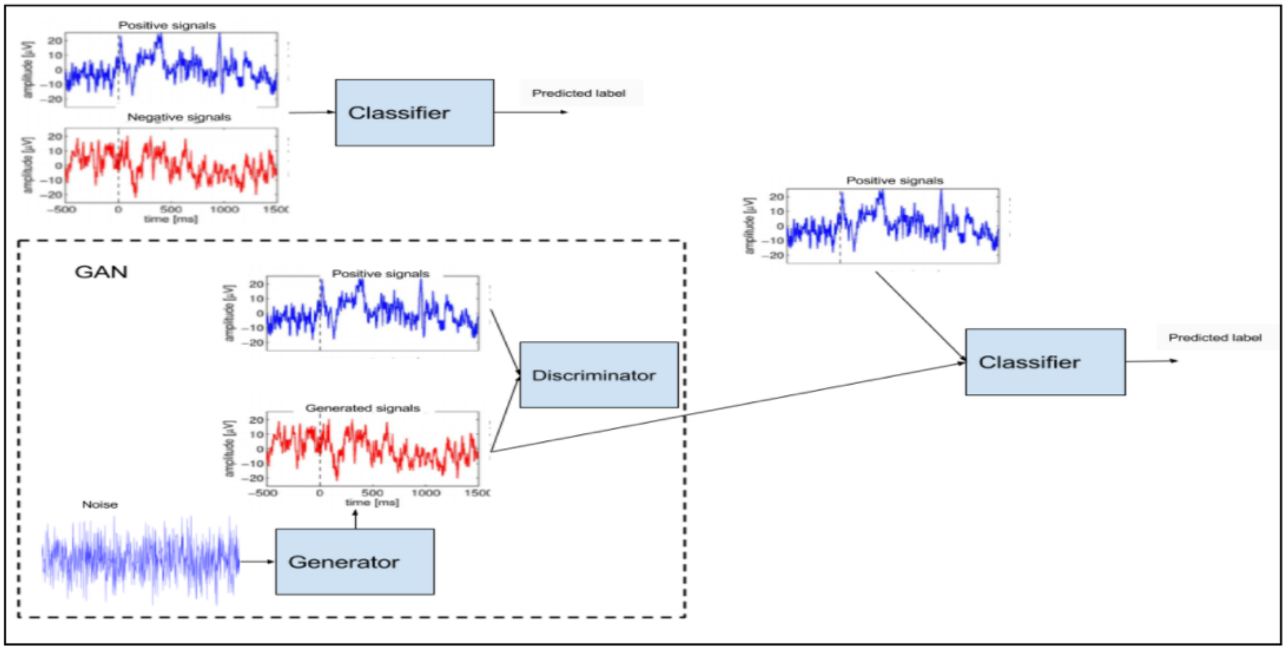

Figure 2. The complete pipeline for training a robust authentication model. A EEG based classifier is trained to authenticate the "positive" user in S_2 from all other "negative" examples in S_1. A Generative Adversarial Network (GAN) is trained on positive examples to generate fake signals. The generated signals are then used as "negative" examples to augment the training data S_1 and retrain the classifier to make it more robust.

tivation function allows for gradient to be back-propogated without any saturation points in either the positive or negative directions.

$$PReLU(x) = max(x, \alpha * x)$$

Other non-linearities like the hyperbolic-tangent and sigmoid suffered from the vanishing gradient problem [7] and hence made the generator more unstable. The slope alpha of the leaky ReLU is a hyper parameter which was chosen using grid search [1]. The structure of the neural network parts (both the generator and the discriminator) were chosen to be 3 layers feed forward networks that are popular in literature [6]. The size of the hidden layer was chosen to be 1200 neurons. The parameter initialization plays a significant role in the optimization of generative adversarial network. Each parameter w is initialized uniformly randomly with "large" bounds such that $w \sim U(-1.0, 1.0)$. The hyper-parameters chosen for optimization are equally important. The learning rate is set to be $1e-3$ with no regularization penalty.

The loss function for the GAN is then defined as follows:

$$L_G = -log(D_{fake}) L_D = log(D_{real}) + log(1 - D_{fake})$$

Here the generator is essentially trying to increase the log likelihood or decrease the negative log likelihood of the "fake" data while the discriminator is trying to decrease the log likelihood of the "fake" data and simultaneously increase the likelihood of the "real" data.

3.2.3 Feature Similarity Learning

The training of the simple GAN described above is susceptible to exploding gradients because of the non-convergent nature of the joint objective function defined above. While trying to minimize the overall cost function the generator often collapses. This means that the generator fails to minimize its cost function and so the discriminator minimizes its loss function by simply classifying anything that the GAN generates as a negative example. Thus the generator never learns the data distribution and the generated samples are close to random noise from which the prior z is sampled. Therefore, the objective function of the generator is modified and redefined to be more convex. The cost function of the generator was modified so that it becomes:

$$L_G = L'_G + |r_d - g_d|^2$$

Where r_d is the pre-activation for the discriminator when input is real data and r_g is the the pre-activation for the discriminator when generated data is passed through the discriminator. Now the generator not only increases the log likelihood of the generated data by decreasing its negative log likelihood, but also tries to decrease the L2 distance between the activation that the discriminator produces when real and generated are provided to it for classification.

3.2.4 How much data is required to train the GAN?

In addition to re-training the GAN using the entire positive dataset, we also conducted a preliminary experiment where we re-trained the GAN using just 50% and 25% of the dataset. For the experiment with 50%, the data for training the GAN is obtained by choosing the first 29 days of the training data in S_2. We generated 10,000 signals which were statistically significant to augment the "negative" dataset S_1 without overwhelming it. For the experiment that was using only 25% of the available data in S_2 using only the first 14 days, the GAN collapses, which implies that the generator cannot generate data very different from the noise it was input. This could be because the GAN starts to over fit on the small amount of variation that this subset of S_2 has.

3.2.5 Noise Analysis

In order to make sure that the GAN trained above is actually able to generate fake signals that have the same distribution as the real signals, and not just a noisy version of the training samples provided to the GAN, we added noise of different magnitudes to the positive sample data, and observed how the baseline classifier behaves. On addition of noise of any reasonable magnitude, the classifier was able to detect the signals as synthetic. This provides some empirical evidence that the classifier is robust against unstructured noise.

3.3. Retraining of classifier with GAN generated data

As shown in Figure 2 the classifier was re-trained with the signals generated by the GAN. Out of the generated 10,000 signals, 8,000 were used in the training of the classifier, labeled as negative examples. These were added to the negative examples defined above. The remaining 2,000 signals were used as the test set. The trained classifier learned to distinguish between the positive dataset and the fake generated signals classifying them as negative. In order to be able to separate the real and the fake generated signals, the classifier spent some energy to learn very specific features that could allow it to make this discrimination. This made the overall classifier more robust in the process (as can be seen from the accuracy numbers shown in later sections).

4. Results

In this study we try to alleviate the problems of traditional EEG based user authentication systems by training a robust classifier using a generative adversarial network (GAN). We first analyze the performance of a standalone classifier, trained on pre-recorded data. We show that this classifier is easily tricked into accepting the synthetic data generated by our GAN. Next, we show that we can protect against this attack by training the original classifier against GAN-synthesized data.

4.1. Training the classifier to detect a user

Empirical evidence shows that the XGBoost classifier worked best for the task of binary classification. This classifier achieves a baseline classification accuracy of 90.8% to distinguish between the positive subject and the negative samples.

4.2. Re-training classifier for added robustness

To test the robustness of this classifier we tested it with the signals generated by the GAN and it accepted the fake signals 100% of the time with a False Acceptance Rate (FAR) of 1, suggesting that the GAN was able to learn the underlying data distribution very well and able to generate signals similar to the real signals for the user of interest. This also shows that the baseline classifier is not trained enough and not resilient to such attacks. In order to make sure that our classifier is thus robust to such attacks and also in general, we re-train our classifiers as described in section 3.4. Thus, the new classifier trained on the generated dataset as well was able to distinguish the fake signals from the real signals. As an additional benefit, its accuracy on the original task of recognizing a subject went up from 90.8% to 91.9% as it was forced to learn intrinsic signals characteristics for this user. The results are summarized in Table 1. To explore how much data was required to train the GAN to produce fake signals, we also trained the GAN using only half of the positive samples available as discussed in section 3.2.4 to generate around 10,000 fake signals. These signals were tested on the baseline classifier and the classifier treated them as "positive" with a FAR of 1. These signals were also used to re-train the baseline classifier as described in section 3.4. Now the re-trained classifier was able to distinguish the fake signals from the real signals, similarly as discussed above. The accuracy of this classifier of recognizing the "positive" subject increased to 95.0% with a False Acceptance Rate (FAR) of 0.03 as shown in Table 1. Therefore the classifier now was able to learn the distribution of the positive samples even better. This improvement in accuracy with half of the available data indicates that the GAN might have been underfitting earlier with all of the data in S_2 as it has too much variability for the number of parameters that the GAN has.

5. Discussion

In this paper, we explored adversarial training for passthoughts. One of the big challenges of training machine learning models with EEG data is the low availability

Model	Accuracy	FAR	FRR
Initially trained classifier	90.8%	0.119	0.0683
Re-trained classifier with GAN data (generated from complete sample)	91.9%	0.112	0.0681
Re-trained classifier with GAN data (generated from half of the sample)	95.0%	0.030	0.0680

Table 1. Comparison of models in terms of classification accuracy, FAR (False Acceptance Rate), FRR (False Rejection Rate)

of such experimental data. The low volume of data is a limiting factor which prevents training of large machine learning models with a high number of parameters. With low amounts of data it is likely to over fit any model, thereby preventing any generalization of the solution. In the absence of appropriate number of parameters, the relatively small models that are trained are not able to learn the features required for an accurate authentication system. This over fitting is true for our classifier that is only trained on the real EEG data available. It follows from the experiments stated above where such a classifier, classifies all of the synthetic signals generated by a GAN as authentic. Thus it could lead to potential hacking and attack strategies for our proposed authentication methods.

Using generative adversarial models (GANs) to generate new data points, we can increase the classifiers specificity considerably. The GAN can theoretically act like a source of large amounts of data, thus eliminating the issues described above with over fitting. Besides acting as a data source, the data generated by the GAN being from the same data distribution as the positive samples of the real data, but still being marked as negative examples, acts as an adversary for the classifier that allows it to learn more relevant features to differentiate between the real positive data and the synthetic data generated by the GAN. As our results show above, the accuracy of the classifier increases after training on the GAN generated data, suggesting that the GAN actually captured intrinsic data distribution that also forced the classifier to learn some additional features for the classification task reducing the FAR value, whereas the FRR values are not adversely affected as shown in Table 1.

These experiments highlight the importance of considering such attacks while training a classifier for user authentication. We recommend designing a passthoughts based user authentication system by augmenting it with signals generated from a Generative Adversarial Network. The retraining of a passthoughts classifier with “fake” GAN generated data not only makes it more robust to GAN attacks but also improves the overall accuracy of the classifier.

6. Future Work

The future work could explore the dependency of the dataset that is provided to the GAN to learn the real signals and the quality of the fake signals thus generated by the GAN. Currently we feed in the data for a particular subject for a block of continuous days to the GAN. We can explore if the temporal continuity is required for the GAN to learn the underlying distribution of the dataset and thus to generate the fake signals. We can use the data samples from some other consecutive block of days for a subject, or for non-consecutive days, and study the performance of the GAN.

In the same vein, we can explore the lower limit on the amount of data needed for the GAN to learn the data distribution. As we saw earlier in section 3.2.4 the GAN was not able to generate meaningful data with 25 percent of the “positive” samples. Here we can study the performance of the GAN along two orthogonal axes. First, if GANs provide a hacking threat, we would like to know how many absolute number of samples are critical for an attacker to break an authentication system using GAN generated data. Therefore, we would like to starve the GAN by providing little amount of data. Second, instead of using the entire sample or half of the sample for the positive subject to generate fake signals, we want to explore if it is possible to generate signals for the positive subject by having very limited access to the positive samples, for example a very small leak of signals, together with the EEG signals of another subject, say the attacker. In this study, we would allow the GAN to learn from EEG signals of a larger population as a whole, but limit the availability of the EEG data of the subject of interest, i.e. the subject for which we want to generate the fake signals. This will allow us to see if EEG signals in general contribute to the learning of the GAN or it is something specific about a subject’s data for the GAN to discover and produce subject’s data distribution.

7. Conclusion

In this study we found that passthoughts are not without their problems. The machine learning models that are trained to act as authenticators are susceptible to attack from a well trained Generative Adversarial Network (GAN). However, GANs are both effective for attacking authenticators and useful for making them robust against the same attacks. We can leverage GANs to learn the underlying distribution of data and act as a large source of data for such authentication systems. This enables us to not only train larger machine learning models for authentication but also the training data is augmented with examples which are difficult to classify correctly. This allows the authenticator to have higher accuracy and less false acceptance, thus making the system more robust.

8. Acknowledgements

This work was supported in part by the Berkeley Center for Long-Term Cybersecurity and the William and Flora Hewlett Foundation. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the Hewlett Foundation or Center for Long-Term Cybersecurity.

References

[1] J. Bergstra and Y. Bengio. Random search for hyper-parameter optimization. *J. Mach. Learn. Res.*, 13(1):281–305, Feb. 2012.

[2] T. Chen and C. Guestrin. Xgboost: A scalable tree boosting system. *CoRR*, abs/1603.02754, 2016.

[3] J. Chuang, H. Nguyen, C. Wang, and B. Johnson. I think, therefore i am: Usability and security of authentication using brainwaves. In A. A. Adams, M. Brenner, and M. Smith, editors, *Financial Cryptography and Data Security*, pages 1–16, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

[4] C. Cortes and V. Vapnik. Support-vector networks. *Mach. Learn.*, 20(3):273–297, Sept. 1995.

[5] W. M. Gentleman and G. Sande. Fast fourier transforms: For fun and profit. In *Proceedings of the November 7-10, 1966, Fall Joint Computer Conference*, AFIPS '66 (Fall), pages 563–578, New York, NY, USA, 1966. ACM.

[6] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative Adversarial Networks. *ArXiv e-prints*, June 2014.

[7] S. Hochreiter. The vanishing gradient problem during learning recurrent neural nets and problem solutions. *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.*, 6(2):107–116, Apr. 1998.

[8] J. Chuang, N. Merrill, T. Maillart and S. of the UC Berkeley Spring 2015 MIDS Immersion Class. Synchronized Brainwave Recordings from a Group Presented with a Common Audio-Visual Stimulus, 2015.

[9] B. Johnson, T. Maillart, and J. Chuang. My thoughts are not your thoughts. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication*, UbiComp '14 Adjunct, pages 1329–1338, New York, NY, USA, 2014. ACM.

[10] Z. Li and Y. Luo. Generate identity-preserving faces by generative adversarial networks. *CoRR*, abs/1706.03227, 2017.

[11] R. Matovu and A. Serwadda. Your substance abuse disorder is an open secret! gleaning sensitive personal information from templates in an eeg-based authentication system. *2016 IEEE 8th International Conference on Biometrics Theory, Applications and Systems (BTAS)*, pages 1–7, 2016.

[12] N. Merrill, M. T. Curran, and J. Chuang. Is the future of authenticity all in our heads?: Moving passthoughts from the lab to the world. In *Proceedings of the 2017 New Security Paradigms Workshop*, NSPW 2017, pages 70–79, New York, NY, USA, 2017. ACM.

[13] Neurosky. Mindwave Mobile brainwave sensing headset.

[14] S. Palazzo, C. Spampinato, I. Kavasidis, D. Giordano, and M. Shah. Generative adversarial networks conditioned by brain signals. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 3430–3438, 2017.

[15] D. E. Rumelhart, B. Widrow, and M. A. Lehr. The basic ideas in neural networks. *Commun. ACM*, 37(3):87–92, Mar. 1994.

[16] A. B. Schwartz, X. T. Cui, D. Weber, and D. W. Moran. Brain-controlled interfaces: Movement restoration with neural prosthetics. *Neuron*, 52(1):205 – 220, 2006.

[17] M. T. Curran, J.-k. Yang, N. Merrill, and J. Chuang. Passthoughts authentication with low cost eareeg. In *Conference proceedings: ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Conference*, volume 2016, pages 1979–1982, 08 2016.

[18] J. Thorpe, P. C. van Oorschot, and A. Somayaji. Passthoughts: Authenticating with our minds. In *Proceedings of the 2005 Workshop on New Security Paradigms*, NSPW '05, pages 45–56, New York, NY, USA, 2005. ACM.

[19] B. Xu, N. Wang, T. Chen, and M. Li. Empirical evaluation of rectified activations in convolutional network. *CoRR*, abs/1505.00853, 2015.

[20] H. Zhang, V. M. Patel, B. S. Riggan, and S. Hu. Generative adversarial network-based synthesis of visible faces from polarimetric thermal faces. *CoRR*, abs/1708.02681, 2017.