Offline Handwritten Signature Authentication with Conditional Deep Convolutional Generative Adversarial Networks

David C. Yonekura, Elloá B. Guedes

[1]Grupo de Pesquisa em Sistemas Inteligentes
Universidade do Estado do Amazonas (UEA)
Av. Darcy Vargas, 1200 – Manaus – Amazonas
{dcy.eng17,ebgcosta}@uea.edu.br

Abstract. *Handwritten signature authentication systems are important in many real world scenarios to avoid frauds. Thanks to Deep Learning, state-of-art solutions have been proposed to this problem by making use of Convolutional Neural Networks, but other models in this Machine Learning subarea are still to be further explored. In this perspective, the present article introduces a Conditional Deep Convolutional Generative Adversarial Networks (cDCGAN) approach whose experimental results in a realistic dataset with skilled forgeries have Equal Error Rate (EER) of* $18.53\,\%$ *and balanced accuracy of* $87.91\,\%$*. These results validate a writer-dependent cDCGAN-based solution to the signature authentication problem in a real world scenario where no forgeries are available nor required in training time.*

1. Introduction

In spite of recent technological developments, handwritten signatures remain an important biometric marker in contemporary society due to its universality, uniqueness, permanence, collectability, acceptability and circumvention [Wayman et al. 2005]. According to Araújo [2019], the development of automatic handwritten signature authentication strategies is important to minimize frauds, to help verifying artistical or historical documents, among many other applications. The crucial challenge for reliable signature authentication methods is to deal with high intra-user variability and also with high-quality forgeries [Heinen 2002], tipically produced with the over-the-shoulder method [Blankers et al. 2009]. Considering the developments so far, there are many findings in literature suggesting different automatic approaches to address this problem, such as using Support Vector Machines (SVMs), Hidden Markov Models, Artificial Neural Networks, etc. [Impedovo and Pirlo 2008, Hafemann et al. 2017b, Sanmorino and Yazid 2012, Hameed et al. 2021].

Given the advent of Deep Learning, a subarea of Machine Learning in which models are trained to recognize complex patterns in high-dimensional data due to a hierarchical feature representation [Goodfellow et al. 2016], many state-of-art solutions for Computer Vision problems have emerged [Khan et al. 2018]. Such deep models, specially Convolutional Neural Networks (CNNs), have already been used in the handwritten signature authentication problem, specially in the offline case – in which no dynamics features during the signing process are captured, just the resulting signature image. In the work of Araújo [2019], for example, two signatures are taken as input – the first is an authentic signature as reference, and the second is the one under test – to a MobileNet CNN architecture that ouputs the binary classification result, whether the second one is authentic or forged. Results showed an accuracy of

$98.65\,\%$ for this task in a writer-independent (WI) scenario, a promising performance. Hafemann et al. [2017a] also considered a WI approach with CNNs applied to the feature learning process, having achieved state-of-art results in a scenario where skilled forgeries are available for training.

An advantage of using CNNs in the problem under consideration is that this model can learn features automatically with little or even no pre-processing tasks. Hence it provides an efficient and robust solution by combining automated feature extraction and prediction or classification [Hameed et al. 2021]. However, Deep Learning comprises models other than CNNs, such as the Deep Convolutional Generative Adversarial Networks (DCGANs), proposed in 2016, to synthesize verisimilar data [Goodfellow et al. 2014, Radford et al. 2016]. So a research question that naturally emerges is: *Can DCGANs be used to address the handwritten signature authentication problem?* In this work we aim at exploring this question first by investigating the findings in literature and also by providing some perspectives with the proposal of a DCGAN-based handwritten signature authentication solution.

As a result, this paper shows a preliminary assessment of a writer-dependent (WD) offline handwritten signature authentication solution based on Conditional DCGANs. The solution proposed was evaluated on a realistic scenario with the CEDAR dataset [Kalera et al. 2004], having Equal Error Rate (EER) of $18.53\,\%$ and balanced accuracy of $87.91\,\%$. Two remarkable features of the proposed solution is that it does not require forged examples in training time and that it can be built under the Semi-Supervised Learning Paradigm. In particular, both characteristics mentioned are well-suited for requirements in real-world scenarios.

The present paper is organized as follows: some background concepts on DCGANs are introduced in Sec. 2. The proposed solution architecture, experimental data and performance metrics are presented in Sec. 3. Results obtained are depicted in Sec. 4. Lastly, final remarks and further steps are shown in Sec. 5.

2. Generative Adversarial Networks

Generative Adversarial Networks (GANs) are a Machine Learning (ML) framework to synthesize verisimilar data. In this approach, two models, tipically artificial neural networks, are trained simultaneously in a zero-sum game dynamic: a generator G produces an artifical output from random noise; a discriminator D must distinguish if a given input comes from the original distribution or if it is artificially produced; G wins when D misclassifies an artificial example as true, and D wins when it correctly classifies an input. If trained jointly until Nash equilibrium, G becomes an expert in synthesizing realistic examples and D becomes an expert in distinguishing real inputs from high-quality forgeries [Goodfellow et al. 2014].

GANs training dynamics is depicted in Fig. 1. The training is said to converge when G outputs artificial examples there are indistinguishable from those in training set and when D strategy is no better than a random guess, but such conditions are hard to verify and remain an important open problem [Langr and Bok 2019]. As a result of training, G describes how the dataset is generated in terms of a probabilistic model, where the random noise z is the sthocastic component that helps producing different output when sampling from the model, accessing different positions in latent space [Foster 2019]. GANs have peculiar characteristics, such as an intuitive non-supervised training, robustness against overfitting and good capabilities to

capture data distribution [Ganguly 2017]. Some notable GANs applications can be found in artificial image colorization, music composition and even in text creation [Foster 2019].

Algorithm 1: GANs training pseudocode

Data: Training set T
foreach *training iteration* **do**
  Train D:
    Get a sample x from T;
    Get a random noise vector z;
    Use z in G to synthesize a fake example x^*;
    Use D to classify x and x^*;
    Compute classification errors;
    Backpropagate total errors in D to adjust trainable parameters aiming at *minimizing* classification errors;
  Train G:
    Get a random noise vector z;
    Use z in G to synthesize a fake example x^*;
    Use D to classify x^*;
    Compute classification errors;
    Backpropagate total errors in G to adjust trainable parameters aiming at *maximizing* D error;
end

Figure 1: Overview of GANs training algorithm.

In the present work we will use Deep Convolutional GANs (DCGANs), introduced in 2016 altogether with optimization training techniques [Radford et al. 2016], and also additional information to conditionate generator and discriminator to label matching, resulting in the so called Conditional Deep Convolutional GANs (cDCGANs). Deep Convolutional Neural Networks (CNNs) are the state-of-art solution for classification, localization and segmentation tasks in Computer Vision problems [Khan et al. 2018]. When adopted into GAN framework as in DCGANs, it results in hyperrealistic image generation, recuperation, restoration, among other tasks [Ganguly 2017, Langr and Bok 2019, Foster 2019].

Upon investigating the literature on the applications of GANs to handwriting signature authentication, there are three remarkable related work. We will discuss them considering the chronological order. In the first work, a DCGAN is used in an unsupervised feature extraction phase followed by a hybrid WI-WD classification using a Gentle Adaboost classifier [Zhang et al. 2016]. Wang and Jia [2019] designed an architecture called SIGAN based on dual learning aiming at authenticating signatures in Chinese handwriting with different pen types (neutral, black, blue, pencil and ballpoint). The authors evaluated their solution on a dataset with 640 images where half of them were positive examples from a single author. According to the experimental evaluation, the discriminator is able to identify the validity of a signature with accuracy of $91.2\,\%$. In the work of Yapıcı *et al.* [2020] a Cycle-GAN is used to artificially augment signature data, since few samples may be available in realistic scenarios both for training and testing. The signature verification step is performed with a CapsNet CNN. Authors compared their DCGAN data augmentation method with other existing solution and obtained best validation accuracy.

In the next section we will introduce our contribution to the handwriting signature verification problem to which we considered the use of cDCGANs framework with a different strategy from prior work found in the literature.

3. Proposed Solution

The proposed solution for the problem under consideration aims at authenticating offline handwritten signatures in a WD cDCGAN-based approach. Each user is associated with a label and there are multiple authentic signatures examples for each user, aiming at capturing the intra-variability of the features found in this biometric data [Hafemann et al. 2017b]. In our solution, user labels will be assigned to their respective signatures in the training set, adding an extra-layer of control for both generator and discriminator [Mirza and Osindero 2014].

In our solution, G and D are CNNs and there is a training set T with authentic signatures of each user. The training dynamics of the cDCGAN proposed is given as follows: a noise vector z and a user label i are independently randomly drawn; G synthesizes a fake signature of the i-th user, i.e., $G(z|i) = \langle x_i^*, i \rangle$; an authentic signature of user i is randomly chosen, resulting in the pair $\langle x_i, i \rangle$; D is called to classify $\langle x_i^*, i \rangle$ and $\langle x_i, i \rangle$, what can be denoted as $D(\hat{x}_i|i)$ where $\hat{x}$ is a given signature (real or fake) and i is the user label; D wins when $D(x_i|i) = 1$, $D(x_i^*|i) = 0$, and when $D(x_j|i) = 0, i \neq j$, and loses otherwise; at the end of each iteration, classification errors are used to adjust trainable parameters in D and G according to their respective goals. This dynamic is illustrated in the architecture drawn at Fig. 2. Upon convergence, D can then be used to discriminate authentic signatures.

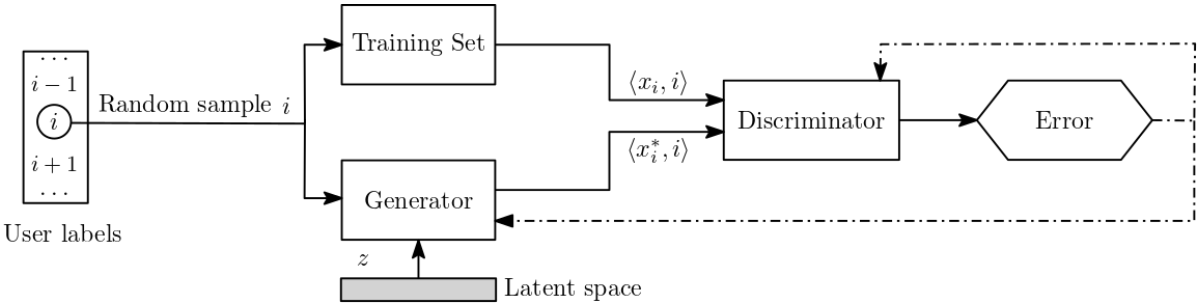


Figure 2: Architecture diagram of the proposed solution.

After training, D will be validated as solution to the signature authentication problem. To do so, it will be tested both against non-previously seen authentic signatures and skilled forgeries. Automatic discrimination of such kind of forgery is considered a challenging task, what enables a realistic evaluation of the proposed solution. The following subsections will depict the solution in terms of its experimental data, cDCGAN architecture, parameters and hyperparameters, performance metrics and fine-tuning strategies.

3.1. Experimental Data

The experimental data used in this work is the CEDAR dataset [Kalera et al. 2004]. It contains offline handwritten signatures from 55 users, with 24 authentic signatures per user and 24 skilled forgeries for each author produced by different forgers. Offline signatures are images resulting of the signing process, without any dynamic information like pen position or inclination at a given time, etc.

The examples in dataset were used in this work in the same way as reported in many findings in literature [Hafemann et al. 2017b]: for each authentic author, ten of his/her signatures were used for training, and the rest for testing the discriminator; authentic signatures per user were randomized before assignment to partitions; all forged signatures were only used for testing purposes. Test data was then organized into binary classes: authentic (positive class) and forged (negative class). It's important to notice that classes in test partition are imbalanced and that is a typical occurrence in authentication scenarios because few forgeries examples are available. To overcome such problem, performance metrics that took such imbalance into account were preferred. Data partition is depicted in Fig. 3.

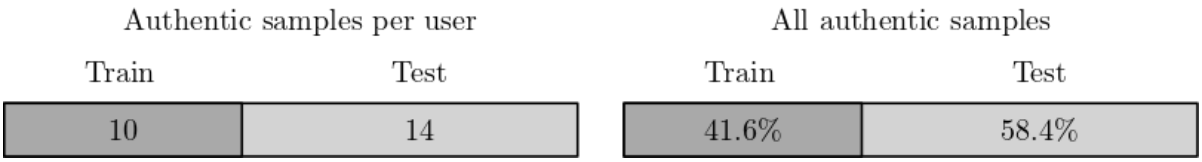


Figure 3: Data partition per user and general data partition overview.

3.2. cDCGAN: Architecture, Parameters and Hyperparameters

When considering the cDCGAN architecture, both G and D were built as sequential deep neural networks with convolutional, fully connected, regularization, and pooling layers. Although some authors suggest to use the same architecture for both generator and discriminator [Berthelot et al. 2017], we opted out to built each component respecting their own specificities. The generator adopted in our solution is depicted in Fig. 4. We used batch normalization after bidimensional transpose convolutional layers in order to normalize the input to the activation function, preventing single point collapsing, but it was not applied to the output layer to avoid sample oscillation. Pooling layers were not used to allow the network to learn its own spatial downsampling [Radford et al. 2016]. Hyperbolic tangent activation function in output layer was used because it tends to produce crisper images [Langr and Bok 2019]. The discriminator, on its turn, is detailed in Fig. 5 and uses Leaky ReLU as activation functions in internal layers, no pooling – based on the same argument considered for the generator, and fully connected layers in output to classify the features previously extracted, producing a probability p with sigmoid activation function. The layers choice and organization for discriminator also considered best practices for CNNs on Computer Vision applications [Khan et al. 2018].

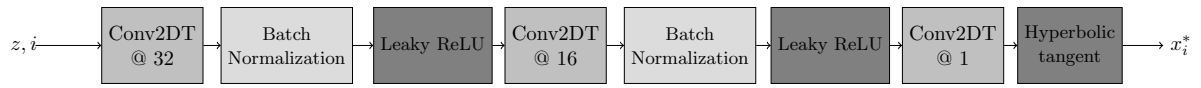


Figure 4: Generator architecture diagram.

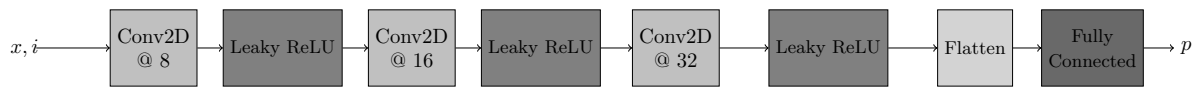


Figure 5: Discriminator architecture diagram.

Some architecture details were ommited in Figs. 4 and 5 for the sake of simplicity, but they are mostly related with the conditional behaviour: the use of label embedding with the

noise vector z in the generator, and reshape of label embeddings followed by concatenation with the input image for creating a joint representation prior classification by the discriminator.

Parameters choice, such as convolutional units, were empirically obtained after inspecting training stability on a few epochs. The dimension of latent space was defined as being equal to the number of authors multiplied by the number of authentic signatures for each author in the training set, resulting in $\dim(z) = 550$. Even with simple GANs, there are a large number of hyperparameters to tune in order to avoid model collapse and unstable training. GANs are highly sensitive to very slight changes and finding a set of hyperparameters that works is often a case of educated trial and error, rather than following an established set of guidelines [Foster 2019]. Therefore, we proposed three experimental setups aiming at optimizing results:

1. **Regular Training Setup**. In this setup, we only used authentic signatures to train the proposed cDCGAN. Discriminator uses binary cross entropy as loss function. The cDCGAN will be trained for $2{,}000$ iterations with batches containing 64 examples;
2. **Extended Training Setup**. After training the cDCGAN in the previous setup, discriminator D will be detached and it will undergo a supervised training with all authentic signatures and their respective correct labels, but also with authentic signatures associated with wrong labels, as in $\langle x_i, j\rangle$ where $i \neq j$. There will be $1{,}000$ training iterations according to this strategy. It aims at enhancing the discriminator on learning representative features of a certain signature meanwhile associating it with the matching label. It can be noticed that in the previous setup only forgeries produced by the generator where considered, what would leave a gap for random signature attacks.

It's important to notice that both setups have same space cost, but the second is more expensive in terms of processing because of the additional discriminator training iterations epochs.

3.3. Performance Evaluation

We assessed False Acceptance (FAR) and False Rejection Rates (FRR). FAR occurs when we accept a user whom we should actually have rejected, also reffered to as a false positive. FRR is the problem of rejecting a legitimate user when we should have accepted him. The Equal Error Rate (EER) is the point where FAR and FRR lines intersect, denoted here as ϑ, sometimes used as a measure of the global accuracy of biometric systems [Andress 2014]. The Detection Error Tradeoff (DET) curve will also be obtained.

Performance evaluation was also analysed under a binary classification perspective with non-previously seen examples. Given a signature x and an author i, $D(x|i)$ returns the probability p of x be an authentic signature of user i. With output p from discriminator, we took the threshold ϑ for classification, where $p \geq \vartheta$ stands for authentic signature, and $p < \vartheta$ indicates a forgery. We have four possible results for this classification task: TP (true positive), TN (true negative), FP (false positive) and FN (false negative), where the positive class corresponds to the authentic case. The balanced accuracy score in Eq. (1) was considered to evaluate the proposed solution, respecting the imbalance between classes.

$$\text{Balanced Accuracy} = \frac{1}{2}\left(\frac{\text{TP}}{\text{TP} + \text{FN}} + \frac{\text{TN}}{\text{FP} + \text{TN}}\right), \tag{1}$$

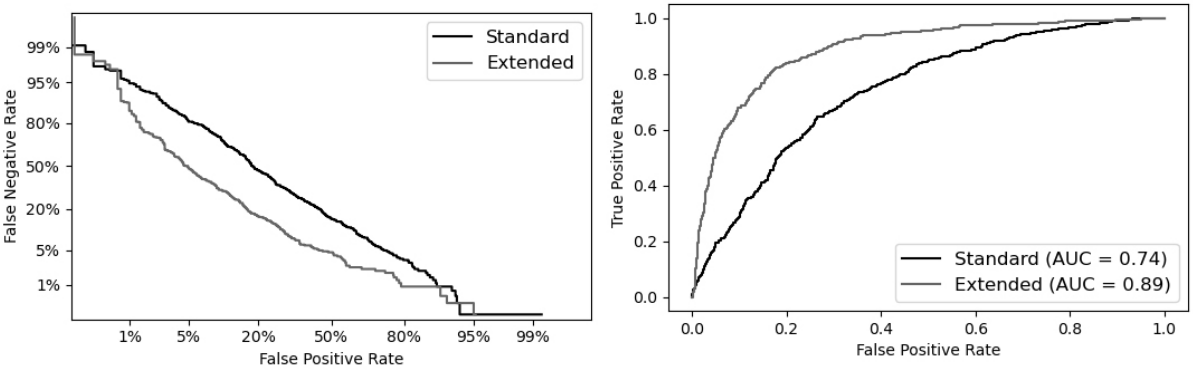
Besides the previously mentioned metrics, we also evaluated the Area Under the ROC curve (AUC) of discriminator D. AUC is an averaged minimum loss measure, where the misclassification loss is averaged over a cost ratio distribution which depends on the score distribution of the classifier in question [Hand 2009].

4. Results and Discussion

Upon implementing the proposed solution using the Python programing language using Keras and Tensorflow frameworks, the resulting scripts were executed in a dedicated computational server with Intel Core i7 processor with $3.7\,\mathrm{GHz}$, $32\,\mathrm{GB}$ of primary memory, $960\,\mathrm{GB}$ SSD secondary memory and 2 NVIDIA GTX 1080 Ti graphic cards with $11\,\mathrm{GB}$ each. Each training setup was repeated 10 times because of stochastic fluctuation in weight initialization. Results obtained are synthesized in Table 1 and Fig. 6, where metrics are depicted in terms of mean and standard deviation of values observed in the 10 runs.

Table 1: Experimental Results.

Experimental Setup	EER	Balanced Accuracy	AUC ROC
Regular Training Setup	0.3068 ± 0.0294	0.6883 ± 0.0294	0.7534 ± 0.0378
Extended Training Setup	0.1853 ± 0.0153	0.8099 ± 0.0152	0.8791 ± 0.0148



(a) DET Curve **(b) ROC Curve**

Figure 6: ROC and DET curves obtained from a randomly chosen run.

By setting the level of confidence to $\alpha = 0.05$, the $95\,\%$ confidence interval to the EER in the regular training setup is 0.3068 ± 0.0182 and in the extended training setup is 0.1853 ± 0.00948. Since confidence intervals do not overlap, it can be concluded that the training setups proposed are statistically distinct besides using the very same training data. Since the EER is lower in the extended training setup, it can be concluded that this approach delivers better results for the experimental scenario considered. This conclusion can be reinforced by analyzing the curves of Fig. 6 where the extended training setup assymptotically minimizes the error rates on DET curve and maximizes the AUC ROC when compared to the regular training setup.

A further examination to explain the gains of the extended training setup performance is necessary. The second setup takes advantage of all patterns captured in the first setup, but exposes the discriminator to more training iterations under the Supervised Learning paradigm, somehow using the CNN learning strategy. Besides that, by also showing examples with wrong labels, as in $\langle x_i, j \rangle, i \neq j$, it favours discriminator to learn more examples from the negative class than only those sampled from the latent space. These two arguments are strong hypothesis that justify such performance increase.

Upon quantitatively comparing the solution proposed to the findings in literature as reported in the survey of Hameed *et al.* [2021, vide Table 26], no DCGAN-based solutions were reported for the CEDAR dataset. However, the current state-of-art performance for this dataset comprises a model that combines both CNN and SVM, with EER of $2.33\,\%$ and using 10 authentic signatures per user. So, our results fall short with a $87.42\,\%$ EER performance decrease. Comparing with prior work on the use of DCGANs to the handwritten signature authentication problem, our contribution is architecturally simpler than SIGAN [Wang and Jia 2019], because the later has two pairs of generator and discriminator and further requires a classifier meanwhile ours is restricted to the cDCGAN itself. We refrain from comparing our solution with the proposition of Zhang *et al.* [2016] because they are fundamentally different on the writer dependence, while their contribution aims at balancing WI and WD methods, ours is strictly WD. Lastly, regarding the work which uses Cycle-GAN for data augmentation, from a structural perspective it can be noticed that a DCGAN is not at the core of the authentication problem as in our solution, since their classification is performed by a secondary CNN [Yapıcı et al. 2020]. However, in terms of performance, a fair comparison would only be possible with further experimentation using similar datasets. On the use of cDCGANs for the problem under consideration, as far as our best searchings efforts have gone, no similar propositions were found in literature.

Although not surpassing the state-of-art performance, this work contributes with the literature by proposing and validating the use of cDCGANs, a Deep Learning framework, to the handwritten signature authentication problem. We also contrasted our contribution with recent findings in literature, explicitly indicating novel experimental scenarios for better evaluating our proposition. Further exploration needs to be carried out in order to compare and contrast the performance of our contribution with existing datasets.

5. Final Remarks

The present work introduces a cDCGAN approach to the handwritten signature authentication problem. The solution proposed only makes use of authentic signatures in training time and comprises a Non-Supervised Learning procedure followed by a CNN-fashion Supervised Learning procedure to enhance the discriminator proposed. Experimental results on the CEDAR dataset show promising performance on a realistic scenario with skilled forgeries.

Aiming at providing further enhancements in the solution proposed to make it competitive with state-of-art algorithms that make use of combined Machine Learning techniques, in future work we will explore parameter and hyperparameter fine tuning, different architectures for both Generator and Discriminator, and also validating the solution with other datasets with more examples, such as the GPDSsyntheticSignature [Ferrer et al. 2013] and the MCYT75

datasets [Ortega-Garcia et al. 2003].

Acknowledgements

Authors achnowledge the financial support provided by FAPEAM and CNPq under the Grant PPP 04/2017 and Program PAIC/FAPEAM/UEA 2019-2020 & 2020-2021.

References

Andress, J. (2014). *The Basics of Information Security – Understanding the Fundamentals of InfoSec in Theory and Practice*. Elsevier, Oxford, UK.

Araújo, M. W. V. (2019). Verificação da autenticidade de assinaturas manuscritas utilizando redes neurais convolucionais. Trabalho de Conclusão de Curso de Bacharelado em Engenharia de Computação na Universidade do Estado do Amazonas.

Berthelot, D., Schumm, T., and Metz, L. (2017). Began: Boundary equilibrium generative adversarial networks. arXiv preprint arXiv:1703.10717.

Blankers, V. L., van den Heuvel, C. E., Franke, K. Y., and Vuurpijl, L. G. (2009). The icdar 2009 signature verification competition. In *10th International Conference on Document Analysis and Recognition*, pages 1403–1407, Barcelona, Catalonia, Spain. IEEE.

Ferrer, M. A., Diaz-Cabrera, M., and Morales, A. (2013). Synthetic off-line signature image generation. In *2013 International Conference on Biometrics (ICB)*, pages 1–7, Espanha.

Foster, D. (2019). *Generative Deep Learning: Teaching Machines to Paint, Write, Compose, and Play*. O'Reilly Media, United Kingdom.

Ganguly, K. (2017). *Learning Generative Adversarial Networks*. Packt Publishing, United Kingdom.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*, volume 1. The MIT Press, Cambridge.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N., and Weinberger, K. Q., editors, *Advances in Neural Information Processing Systems*, volume 27, pages 2672–2680, Barcelona. Curran Associates, Inc.

Hafemann, L. G., Sabourin, R., and Oliveira, L. S. (2017a). Learning features for offline handwritten signature verification using deep convolutional neural networks. *Pattern Recognition*, 70:163–176.

Hafemann, L. G., Sabourin, R., and Oliveira, L. S. (2017b). Offline handwritten signature verification - literature review. In *2017 Seventh International Conference on Image Processing Theory, Tools and Applications (IPTA)*, pages 1–8, Canada. IEEE.

Hameed, M. M., Ahmad, R., Kiah, M. L. M., and Murtaza, G. (2021). Machine learning-based offline signature verification systems: A systematic review. *Signal Processing: Image Communication*, 93:116139.

Hand, D. J. (2009). Measuring classifier performance: a coherent alternative to the area under the ROC curve. *Machine Learning*, 77(1):103–123.

Heinen, M. R. (2002). Autenticação on-line de assinaturas utilizando redes neurais.

Impedovo, D. and Pirlo, G. (2008). Automatic signature verification: The state of the art. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 38:609–635.

Kalera, M. K., Srihari, S., and Xu, A. (2004). Offline signature verification and identification using distance statistics. *International Journal of Pattern Recognition and Artificial Intelligence*, 18(07):1339–1360.

Khan, S., Rahmani, H., Shah, S. A. A., and Bennamoun, M. (2018). *A Guide to Convolutional Neural Networks for Computer Vision*. Morgan and Claypool.

Langr, J. and Bok, V. (2019). *Generative Adversarial Networks in Action – Deep Learning with Generative Adversarial Networks*. Manning Publications, Shelter Island.

Mirza, M. and Osindero, S. (2014). Conditional generative adversarial nets. arXiv preprint arXiv:1411.1784.

Ortega-Garcia, J., Fierrez-Aguilar, J., Simon, D., Gonzalez, J., Faundez-Zanuy, M., Espinosa, V., Satue, A., Hernaez, I., Igarza, J.-J., Vivaracho, C., Escudero, D., and Moro, Q.-I. (2003). MCYT baseline corpus: a bimodal biometric database. *IEE Proceedings - Vision, Image, and Signal Processing*, 150(6):395.

Radford, A., Metz, L., and Soumith, C. (2016). Unsupervised representation learning with deep convolutional generative adversarial networks. In *6th International Conference on Learning Representations*, page 16, Puerto Rico.

Sanmorino, A. and Yazid, S. (2012). A survey for handwritten signature verification. In *2012 2nd International Conference on Uncertainty Reasoning and Knowledge Engineering*, pages 54–57, Jalarta. IEEE.

Wang, S. and Jia, S. (2019). Signature handwriting identification based on generative adversarial networks. *Journal of Physics: Conference Series*, 1187(4):042047.

Wayman, J., Jain, A., Maltoni, D., and Maio, D. (2005). An introduction to biometric authentication systems. In *Biometric Systems*, pages 1–20. Springer-Verlag.

Yapıcı, M. M., Tekerek, A., and Topaloğlu, N. (2020). Deep learning-based data augmentation method and signature verification system for offline handwritten signature. *Pattern Analysis and Applications*, 24(1):165–179.

Zhang, Z., Liu, X., and Cui, Y. (2016). Multi-phase offline signature verification system using deep convolutional generative adversarial networks. In *2016 9th International Symposium on Computational Intelligence and Design (ISCID)*. IEEE.