Comparison of Two Algorithms for ECG Signal Denoising: A Recurrent Neural Network and A Support Vector Regression

Sahar Keshavarzi

Department of Electrical and Computer Engineering
Isfahan University of Technology, Isfahan, Iran.
E-mail: saharkeshavarzi20@gmail.com

Abstract - **The Electrocardiogram (ECG) signal is usually degraded by noise. The use of a denoising technique, therefore, is required before applying any analyses (for example, for diagnoses purpose) to this signal. This paper investigated two algorithms for enhancing the ECG signal in the presence of noise: a Support Vector Regression (SVR) and a deep Recurrent Neural Network (RNN). We compared the performance of the two algorithms using three case studies that differed in terms of the size of data used to train the algorithms. The performance of both algorithms was evaluated at different SNRs (-5 dB to 15 dB, with the step size of 2 dB) using two objective metrics including Pearson correlation and mean squared error. The simulation results demonstrated that the RNN outperformed the SVR algorithm for ECG enhancement in noise for all the three case studies. In particular, the mean correlation scores at low SNRs (-5 dB to 1dB) averaged across the three studies were 0.7 and 0.85 for SVR and RNN, respectively. We also found that the performance of both algorithms was degraded by decreasing in SNR. In addition, improvement in the performance of both algorithms was observed when the size of training data increased.**

Keywords - ECG signal; denoising; recurrent neural network; support vector regression

I. INTRODUCTION

Electrocardiogram (ECG) signals, as the recording of the electrical activity of the heart, are widely used for screening and diagnosing heart diseases. The ECG signal can be degraded by different types of noise. The noise can be generated by various sources such as baseline wander, power line interference, motion artefacts, muscle artifacts, poor contact of electrodes. Hence, it would be very important to denoise the ECG signals to have an accurate diagnosis. Many algorithms have been presented to pre-process and to denoise the ECG signals. These algorithms can be generally classified into two main groups: classical algorithms and machine learning-based algorithms.

The classical methods mostly are based on techniques such as adaptive filters, Empirical Mode Decomposition (EMD), independent component analysis (ICA), wavelet techniques, Savitzky Golay filter, Kalman filters. On the other hand, support vector regressions (SVRs), convolutional neural networks (CNNs), recurrent neural networks (RNNs) were also investigated as machine learning-based algorithms such as to enhance the noisy ECG signal.

In this paper, two different algorithms were investigated for enhancing the ECG signal in the presence of noise at SNRs of -5 dB to 15 dB. The first algorithm was based on the SVR, and the second one was an RNN. We then compared the performance of the SVR with that of RNN using three case studies that differed in terms of the size of data used to train the algorithms. Two objective measures including MSE and Pearson correlation were used to quantify the performance. The simulation results suggested that the RNN outperforms the SVR for enhancing ECG corrupted by noise. We also found that the performance of both algorithms is improved as the size of the training data increases across the case studies.

The rest of paper is structured as follows. In section II, we review the related literature. The SVR and RNN algorithms are introduced in section III. In section IV, the experimental methodology, software, and hardware are discussed. The simulation results are described in V. Finally, conclusions are presented in section VI.

II. LITERATURE REVIEW AND LIMITATIONS

Thakor and Zhu [1] proposed a method based on the adaptive filter for the ECG noise cancellation and arrhythmia detection. The adaptive filter was developed to minimize the mean squared error (MSE) between the noisy ECG signal and a reference input which was either noise or a signal that was correlated only with ECG in the primary input. In [2], a two-steps method based on the EMD was presented. The first step utilized the first-order intrinsic mode function (F-IMF) from the EMD, and the second step used three statistical measures on the F-IMF time series to find characteristics of randomness and variability: the Shannon entropy, mean, and variance. Kuzilek et al. [3] employed an approach based on ICA that combines JADE source separation and binary decision tree for identification and then for removing ECG noise.

In [4], a threshold-based approach using wavelet transform coefficients was proposed to denoise the ECG signal. The proposed approach decomposed the input signal into five levels of wavelet transform by using Daubechies and determined a threshold value under which the error between

DOI 10.5013/IJSSST.a.23.01.02 ISSN: 1473-804x online, 1473-8031 print

the detailed coefficients of the noisy signal and those of original signal is minimized. Chakrabortya and Das [5] used the Savitzky-Golay filter to denoise the ECG signal recorded from a normal participant. They compared the performance of the Savitzky-Golay filter with the band pass filter of Pan-Tompkins algorithm and reported a better performance for the former. In [6] an extended Kalman smoother with the differential evolution technique was presented for noise cancellation of the ECG signal. The proposed method used an automatic parameter selection for selection of ten optimized components of the ECG signal to create the ECG signal based on the real ECG signal.

Arsene et al. [7] applied two deep learning-based algorithms including a CNN and an RNN to noisy ECG signals. They further investigated a wavelet technique based on an empirical Bayesian method with a Cauchy prior as the baseline. The results showed that the CNN model was superior to both RNN and the wavelet technique. In [8], a model based on a deep RNN was proposed for denoising the ECG signals. The proposed model was a specific hybrid of deep RNN and a denoising autoencoder, and achieved the SNR of 7.71 dB for the input ECG signal with the SNR of -8.82 dB.

There are several limitations with the existing ECG denoising techniques. First, the most of these studies did not take into account high levels of noise. Second, since a standard set of ECG data and noise were not used in the previous studies, it is difficult to compare the results with those obtained using other algorithms. Third, none of the previous machine learning-based algorithms evaluated the effects of size of data set on the performance of the ECG denoising algorithm.

III. PROPOSED METHODS

As mentioned above, two different algorithms were separately used to denoise ECG signal corrupted by noise. In this section, both algorithms are described in detail.

A. SVR

SVR, inspired by support vector machine, is a supervised machine learning algorithm that was developed to solve the regression problems [9]. The SVR maps the input data into a feature space F, constructs a linear regression model and then solves it in the feature space using a nonlinear (or linear) mapping. Given input data x and output data y, the linear regression function can be written as:

$$f(\mathrm{x}) = w\phi(\mathrm{x}) + b \quad , \quad \phi: R^n \longrightarrow F \;, \; w \in F \tag{1}$$

where w and b are weight and constant coefficients respectively, and ϕ denotes the nonlinear mapping. The SVR objective is to find a regression function f (.) that precisely estimates the output data corresponding to each input data point. To this end, the SVR performs linear regression in the feature space by consideration of ε-insensitive loss and by optimizing the following convex problem [9]:

$$\begin{aligned} &minimize \quad \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{l}(\xi_i + \xi_i^*) \\ &subject\ to \quad \begin{cases} \langle w, \phi(x)\rangle + b - y_i \le \varepsilon + \xi_i \\ y_i - \langle w, \phi(x)\rangle - b \le \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* \ge 0 \end{cases} \end{aligned} \tag{2}$$

where C denotes the penalty factor, and variables ξ_i and ξ_i^* determine the amount of difference between the estimated value and the target value.

B. RNN

Over the past few years, artificial neural networks (ANNs) have been widely used to achieve significant improvements in different areas such as the visual, auditory, speech, and language domains. The ANNs are generally divided into three categories [10]: feed-forward deep neural networks (DNNs), CNNs and RNNs. Among these three categories, RNNs have been found to achieve superior outcome for sequential data and for tasks such as biomedical signals denoising, speech enhancement, and time series prediction, etc. In particular, the “long short-term memory” (LSTM, [11]), as one of RNN variants, has achieved significant advances in reducing background noise [12].

In this paper, an RNN-based architecture was used as the second algorithm to enhance the noisy ECG signal. The RNN consisted of an input layer, two LSTM layers with 100 units followed by a fully connected layer with 100 units, and an output layer. The RNN took frames (with the length of 100 time-samples) of noisy ECG signals and predicted the enhanced ECG signal at its output. The schematic diagram of the RNN algorithm is shown in Fig. 1.

IV. EXPERIMENTAL METHODOLOGY, SOFTWARE AND HARDWARE

In this paper, both algorithms (SVR and RNN) were implemented using Python (version 3.8.5) installed on a Dell 9310 machine running Windows 10 with an Intel Core i7-1165G7 Processor. The SVR was implemented using a Python library named Scikit-Learn [13]. The SVR kernel was linear, and the Squared Error Epsilon Insensitive Loss was used as the loss function. The maximum number of iterations and tolerance were 20000 and 0.05, respectively. The python libraries TensorFlow [14] and Keras [15] were also used to build, train, and test the RNN. We utilized the “Adam” optimizer [16] with learning rate = 0.001, $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\varepsilon = 10^{-7}$ as the optimizer method during training to minimize the mean squared error as the loss function. The batch size and number of epochs were 15 and 30, respectively.

The ECG data and the noise were generated using Matlab. The noisy ECG signals were also created by adding white Gaussian noise to clean ECG signals, using Matlab function *awgn*(), at SNRs of -5 dB to 15 dB with the step size of 2 dB.

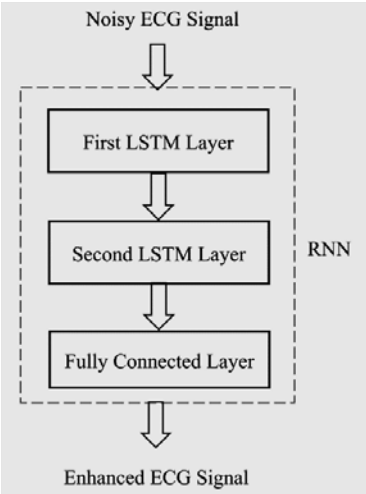

Figure 1. The schematic diagram of RNN algorithm used in this paper.

V. SIMULATION RESULTS

To assess the performance of the two algorithms (SVR and RNN) in ECG denoising, we simulated clean and noisy ECG signals. The simulated signals were then considered as data samples for training and testing the models.

To evaluate the effects of size of training data on the algorithm performance, three case studies were considered in this paper. The testing data set were same across all case studies. Also, same SVR algorithms and same RNN architectures were used in the three case studies. Both algorithms processed frames of ECG signal with the length of 100 time-samples as their inputs and predicted the same length of signal at their outputs. Two objective criteria, including MSE and Pearson correlation, were employed to quantify the performance. The MSE between the original ECG signal, $y(t)$, and the predicted ECG signal, $\hat{y}(t)$, is calculated as:

$$MSE = \frac{1}{m}\sum_{i=1}^{m}[y(i) - \hat{y}(i)]^2 \qquad (3)$$

where m is the number data points. The Pearson correlation, r, between the original and predicted ECG signals is also determined as:

$$r = \frac{\sum_{i=1}^{m}(y(i)-\bar{y})(\hat{y}(i)-\bar{\hat{y}})}{[\sum_{i=1}^{m}(y(i)-\bar{y})^2 \sum_{i=1}^{m}(\hat{y}(i)-\bar{\hat{y}})^2]^{0.5}} \qquad (4)$$

where $\bar{y}$ and $\hat{y}$ are means of original and predicted ECG signals, respectively.

A. First Case Study

In the first case, the size of data used for testing and for training (and evaluating) models were 2200 and 4400 frames of ECG, respectively. Both data sets included SNRs of -5 dB to 15 dB with the step size of 2 dB (200 frames per SNR for testing and 400 frames per SNR for training).

Fig. 2 shows the performance of both SVR and RNN in terms of MSE and Pearson correlation versus SNR (dB). As it can be seen from Fig. 2 A, the performance of both algorithms in terms of MSE would increase by increasing the SNR. Same trend is observed for the second objective measures, namely correlation, as demonstrated in Fig. 2 B. Comparing the MSE values between the algorithms, the RNN outperforms the SVR for almost all SNRs used in this study (except for -5 dB). In particular, the average MSE values at low SNRs (-5 dB to 1dB) were 0.080 and 0.067 for SVR and RNN, respectively. Similarly, the correlation score for RNN is greater than that of SVR for all SNRs. The average correlation scores at low SNRs (-5 dB to 1dB) were 0.69 and 0.84 for SVR and RNN, respectively.

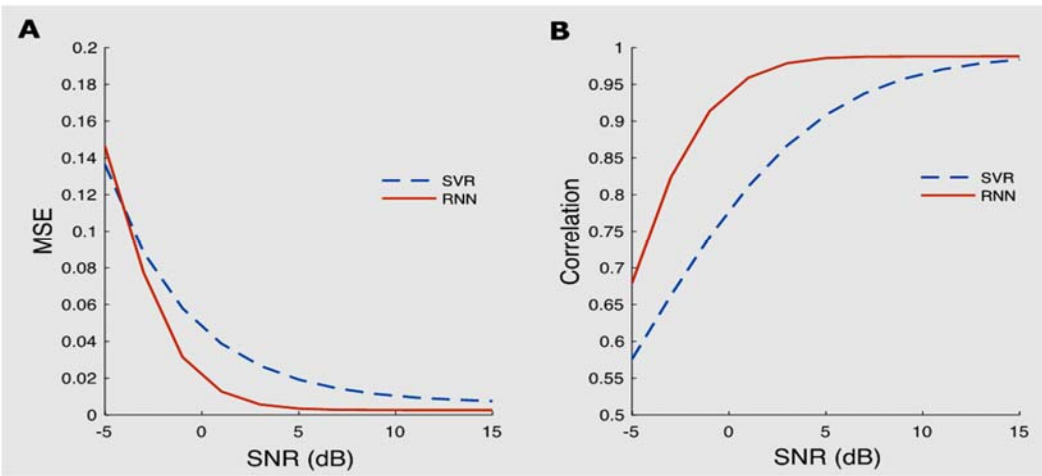

Figure 2. The performance of SVR and RNN based on (A) MSE and (B) Pearson correlation for the first case study.

Fig. 3 below shows an example of ECG signal (2 frames), noisy signal at SNR = 1dB, predicted signal by SVR, and predicted Signal using RNN.

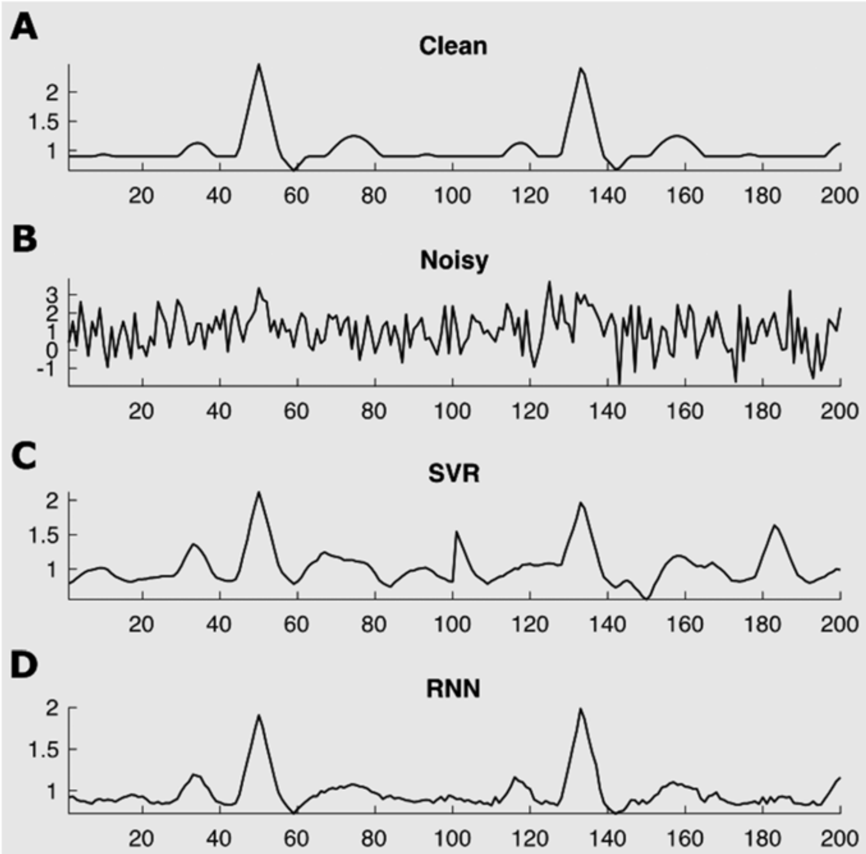

Figure 3. An example of (A) clean ECG signal (200-time samples or two frames), (B) noisy signal with SNR = 1 dB, (C) noisy signal processed by SVR, and (D) noisy signal processed by RNN.

B. Second Case Study

In the second case study, the size of data used for testing and for training algorithms were 2200 and 6600 frames of ECG (200 frames per SNR for testing and 600 frames per SNR for training), respectively. The performance of both SVR and RNN is depicted in terms of MSE and correlation score against SNR in Fig. 4.

According to Fig. 4, the performance of both algorithms, based on both MSE and correlation, would increase by increasing the SNR. In addition, both objective measures predict better performance for RNN compared to SVR for all SNRs studied here. In particular, the average MSE values at low SNRs (-5 dB to 1dB) were 0.074 and 0.037 for SVR and RNN, respectively. The average correlation scores at low SNRs were 0.70 and 0.85 for SVR and RNN, respectively.

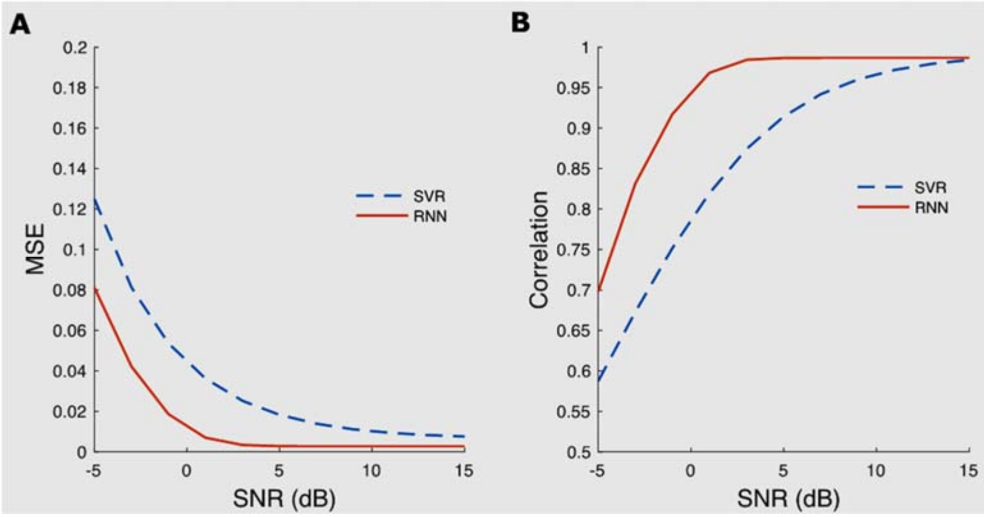

Figure 4. The performance of SVR and RNN in terms of (A) MSE and (B) Pearson correlation for the second case study.

C. Third Case Study

In the third study, the same models used in the two previous case studies were trained using 8800 frames of ECG and tested on 2200 frames (200 frames per SNR for testing and 800 frames per SNR for training). We then quantified the performance of the two algorithms using MSE and Correlation, as seen in Fig. 5. As demonstrated in Fig. 5, the performance of both SVR and RNN, based on MSE and correlation values, increases when the SNR increases. Also, both objective measures predict better performance for RNN compared to SVR across all SNRs. Particularly, the average MSE values at low SNRs (-5 dB to 1dB) were 0.067 and .023 for SVR and RNN, respectively. The average correlation scores at low SNRs were 0.72 and 0.87 for SVR and RNN, respectively.

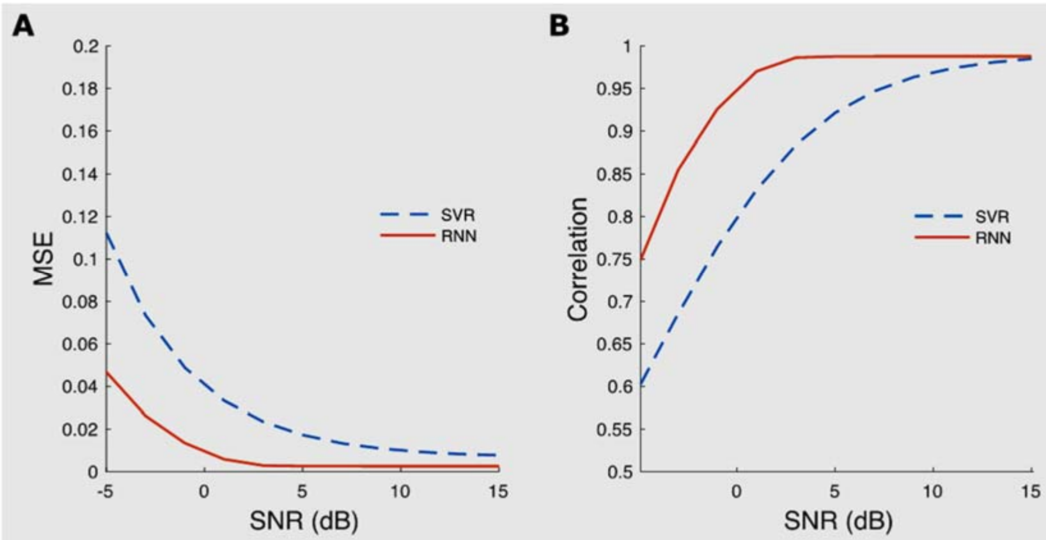

Figure 5. The performance of SVR and RNN in based on (A) MSE and (B) Pearson correlation for the third case study.

Comparing the results obtained for the two algorithms across the three case studies, we found that the performance of both algorithms would improve by increasing the size of training data. This improvement, particularly for correlation scores, was greater for RNN compared to the SVR, and more remarkable for the low SNRs.

VI. CONCLUSIONS

In this paper, two algorithms including a SVR and an RNN were investigated to denoise ECG signals corrupted by noise at different SNRs (-5 dB to 15 dB). Three different case studies (each with different training data set) were simulated. two objective measures, MSE and Pearson correlation, were used to quantify and compare the performance of the two algorithms for the considered SNRs. Both metrics showed that the RNN was superior to SVR for enhancing noisy ECG signals. Additionally, the performance of both algorithms would improve by increasing the size of training data, although this improvement was higher for RNN. We further found that the performance of the two algorithms is degraded by decreasing SNR.

REFERENCES

[1] N. V. Thakor and Y. S. Zhu, "Applications of adaptive filtering to ECG analysis: noise cancellation and arrhythmia detection," IEEE transactions on biomedical engineering, 38(8), pp. 785–794, 1991.

[2] J. Lee, D. D. McManus, S. Merchant, and K. H. Chon, "Automatic motion and noise artifact detection in Holter ECG data using empirical mode decomposition and statistical approaches," IEEE Transactions on Biomedical Engineering, 59(6), pp. 1499–1506, 2011.

[3] K. Jakub, V. Kremen, F. Soucek, and L. Lhotska. "Independent component analysis and decision trees for ECG holter recording de-noising," PLoS One, 9(6): e98450, 2014.

[4] M. Alfaouri K. Daqrouq, "ECG signal denoising by wavelet transform thresholding," American Journal of applied sciences, 5(3), pp. 276–281, 2008.

[5] M. Chakraborty and S. Das, "Determination of signal to noise ratio of electrocardiograms filtered by band pass and Savitzky-Golay filters," Procedia Technology, 4, pp. 830–833, 2012.

[6] D. Panigrahy and P. K. Sahu, "Extended Kalman smoother with differential evolution technique for denoising of ECG signal," Australasian physical & engineering sciences in medicine, 39(3), pp. 783-795, 2016.

[7] C. T. Arsene, R. Hankins, and H. Yin, "Deep learning models for denoising ECG signals," European Signal Processing Conference, pp. 1–5, 2019.

[8] K. Antczak, "Deep recurrent neural networks for ECG signal denoising," arXiv preprint arXiv:1807.11551, 2018.

[9] V. Vapnik, "The Nature of Statistical Learning Theory," Springer, New York, 1995.

[10] M. Keshavarzi, T. Goehring, J. Zakis, R. E. Turner, and Brian C. J. Moore, "Use of a deep recurrent neural network to reduce wind noise: effects on judged speech intelligibility and sound quality," Trends in hearing, 22, 2331216518770964, 2018.

[11] S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural Computation, 9, pp. 1735–1780, 1997.

[12] M. Keshavarzi, T. Reichenbach, and B. C. J. Moore, "Transient Noise Reduction Using a Deep Recurrent Neural Network: Effects on Subjective Speech Intelligibility and Listening Comfort," Trends in Hearing, 25, 23312165211041475, 2021.

[13] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, et al. "Scikit-learn: Machine learning in Python," The Journal of machine Learning research, 12, pp. 2825–2830, 2011.

[14] M. Abadi, A. Agarwal, P. Barham, E. Brevado, Z. Chen, C. Citro, et al., "TensorFlow: Large-scale machine learning on heterogeneous distributed systems," arXiv 1603.004467, 2016.

[15] F. Chollet. https://Github.Com/Fchollet/Keras. Last accessed March 2021.

[16] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," Preprint arXiv:1312.6199, 2014.

.