Saccade Landing Point Prediction: A Novel Approach based on Recurrent Neural Networks

Aythami Morales[1,2], Francisco M. Costela[2,3], Ruben Tolosana[1], and Russell L. Woods[2,3]

[1]BiDA-Lab, Department of Electrical Engineering, Universidad Autonoma de Madrid, Madrid, Spain
[2]Schepens Eye Research Institute, Mass Eye and Ear, Boston, MA, USA
[3]Department of Ophthalmology, Harvard Medical School, Boston, MA, USA
{aythami.morales, ruben.tolosana}@uam.es, {russell_woods, francisco_costela}@meei.harvard.edu

ABSTRACT

A saccade is a fast eye movement that allows the change of visual fixation from one object of interest to another. These movements are characterized by very high angular velocity peaks that can reach up to 1,000º/s, making them as one of the fastest neuromotor activities in the human body. Modeling such a complex movement remains a challenge. Saccadic eye movements can be defined by initial and landing points, duration, amplitude, and velocity profile. The landing point is important as it defines the new fixation region and, therefore, the region of interest of the viewer. Its prediction may reduce problems caused by display-update latency in gaze-contingent systems that make real-time changes in the display based on eye tracking. The main contribution of this work is to propose the use of state-of-the-art machine learning techniques (i.e., Recurrent Neural Networks) for saccade landing point prediction in real-world scenarios. Our method was evaluated using 220,000 saccades from 75 subjects acquired during viewing video from "Hollywood" movies. The results obtained using our proposed methods outperform existing approaches with improvements of up to 40% error reduction. Our results show that dynamic temporal relationships exploited by Recurrent Neural Networks can improve the performance of traditional Feed Forward Neural Networks.

CCS Concepts

• Human-centered computing → Human computer interaction (HCI) → Interactive systems and tools

• Computer systems organization → Architectures → Neural Networks

Keywords

Deep Learning, saccade, eye movement, gaze-contingent, Recurrent Neural Networks, LSTM.

ICMLT'18, May 19–21, 2018, Jinan, China.

DOI: http://dx.doi.org/XXXXXXXXXXXXXX

1. INTRODUCTION

A saccade is a rapid eye movement used to align the fovea with regions of interest (between fixations). The alignment of the fovea provides highest resolution at the region of interest. People make about three saccades per second. The peak angular velocity of a saccade can reach up to 1,000º/s with a duration of about 15 to 200ms. These properties mean that saccades are one of the fastest muscle movements of the human body. Saccades are movements traditionally modeled according to ballistic trajectories defined by parameters such as initial (fixation) point, landing point, maximum velocity, velocity profile, duration and, amplitude, among others.

Modelling eye-movement dynamics is a challenging task that has attracted the interest of the research community. Bahill, Clarck, and Start [1] described the linear relationship between saccade amplitude and maximum velocity (the saccadic main sequence). Regarding velocity profiles, Lognormals functions can be used for modelling saccades that are longer in time, whereas for the short ones either Gaussian functions [2] or compressed exponential functions can be used [3]. Other models have been proposed in the literature for different applications related to person recognition [4][5][6], visual search [7], animation [8] and gaze-contingent displays [3][9].

The landing point determines the new fixation area and the fovea alignment. There has been one report on predicting the saccade landing point [10] based on polynomial fitting, suggesting that it is possible to predict the landing point of saccades. This work proposes a new approach for landing point prediction during saccadic movements. Its prediction may reduce problems caused by display-update latency in gaze-contingent display that change the image on the display according to where the viewer looks. Gaze-contingent displays have been used on video streaming [11], human-computer interfaces for new virtual reality environments [12] or driving simulators [13], among others. These systems allow researchers to investigate a variety of visual phenomena, including eye movement guidance in reading, stability of vision, visual search strategies, and scene perception. Due to data transmission, image processing, and data display preparation, the time delay between the eye tracker and the monitor update may lead to a misalignment between the eye position and the image manipulation during eye movements [14].

Figure 1 shows an example of a real saccade trajectory. The figure shows the saccade trajectory when the region of interest changed from stimulus A (the bird) to stimulus B (the tree). Figure 1 illustrates some of the main challenges associated with saccade modeling: i) nonlinear trajectory; ii) noise between trajectory

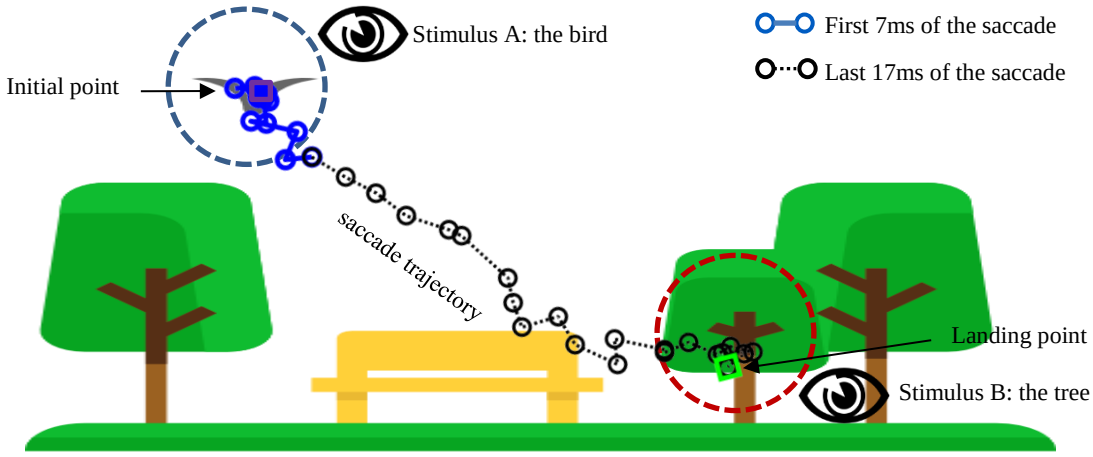

Figure 1. Example of a real saccade movement acquired with an EyeLink 1000 eye tracker (1,000Hz). Circles represent the Cartesian coordinates of each of the samples obtained from the eye tracker.

samples; and iii) nonlinear velocity profiles (e.g. the distance between data samples during the first 7ms and the last 17ms varies).

The main contributions of this work are: 1) A new approach based on Recurrent Neural Networks (RNNs) for improving the performance of saccade landing prediction; 2) An empirical study with about 220,000 saccades from 75 subjects acquired during viewing video; and 3) Comparison with other state-of-the-art approaches. The results obtained using our proposed method outperformed existing approaches with improvements ranging from 20% to 50% error reduction.

The rest of this paper is organized as follows: Section 2 describes the saccade landing point prediction approach based on Recurrent Neural Networks and the database used for the experimental protocol. Section 3 reports the experiments and results. Finally, Section 4 summarizes the conclusions.

2. METHODS

2.1 Saccade detection

The eye data were collected using an EyeLink 1000 eye tracker (SR Research Ltd., Mississauga, Ontario, Canada) at a 1,000 Hz sampling rate while subjects viewed a 27" display (60 × 34 cm) from 1 m for a 33 × 19° potential viewing area. A total of 219,335 saccades were detected off-line using the method described in [9], but the data used to test the algorithms were the raw data of the saccades so identified. Blinks were identified and removed using EyeLink's online data parser. Periods preceding and following the missing data were removed if they exceeded a speed threshold of 30°/s. Then, we interpolated over the removed blink data by applying cubic splines. For saccade detection, the raw data was smoothed by applying a 3rd-order Savinsky-Golay filter with a window size of 14. Without this smoothing, saccade detection was much less reliable. Speed was calculated as the first derivative of the eye position with respect to time. The beginning of a saccade was signaled when speed exceeded 30°/s for at least 10 ms. The end of a saccade was signaled when speed went below 30°/s. The saccades were restricted to saccades (1) smaller than 40° as this was approximately the maximum diagonal dimension of the display; and (2) larger than 1° and 15 ms in duration to exclude microsaccades. We imposed additional restrictions regarding the initial (< 0.075°/ms) and terminal velocity (<0.3°/ms), as well as the removal of saccades with a velocity at first quartile of duration lower than 0.15 peak velocity. This threshold removed those eye movements with uniform but unrealistic low velocity profiles during their initial phase, and which may have been pursuit eye movement. The smoothed data of the saccades identified using the above procedure was then replaced with the raw data, with the rationale being that a real-time algorithm will have raw data available and therefore our input is realistic.

2.2 Saccade Landing Point Prediction: An Approach Based on RNNs

Deep Learning (DL) has become a thriving topic in the recent years [16], allowing computers to learn from experience and understand the world in terms of a hierarchy of simpler units. DL has enabled significant advances in complex domains such as natural language processing [17] and computer vision [18]. The main reasons for the frequent deployment of DL are the increasing amount of available data and the deeper size of the models possible due to increased computer resources.

New trends based on the use of RNNs which is a specific DL architecture, are becoming important nowadays for modelling sequential data [19]. The range of applications of RNNs varies enormously, from handwritten-signature recognition [20] to biomedical problems [21]. RNNs are defined as a connectionist model containing self-connected hidden layers that allows to deal with dynamic temporal relationships. One benefit of the recurrent connection is that memory of previous inputs remains in the network internal state, allowing it to make use of past context. However, the range of contextual information that standard RNNs can access is extremely limited due to the well-known vanishing gradient problem [22][23].

Among the different RNN architectures proposed during the last decade, Long Short-Term Memory (LSTM) [24] has become popular because of its capacity to solve traditional shortcomings of standard RNNs. LSTMs [24] have been successfully applied to many challenges involving short utterances such as speech recognition [25] or epileptic-seizure detection [21].

The aim of the proposed system is to estimate the landing point according to the samples obtained by the eye tracker during the beginning of the saccade movement (short utterance). The prediction is made according to: i) available data provided by the eye tracker (beginning of the saccade) and; ii) prediction model (LSTM network). The length of the available sequence varies depending on the moment in which the prediction is made. In the best-case scenario, the landing point should be predicted early in the saccade (e.g. first 10ms). However, trajectories of saccades may not be monotonic, and such non-monotonic movement should be modelled. To adapt the prediction to non-monotonic trajectories, the prediction is updated every 5ms to include new samples available during the saccade movement.

Our proposed system is based on two LSTM hidden layers with 32 memory blocks each, and finally a fully connected feed-forward neural network layer with a linear activation and two units (see Figure 2) to predict the final x and y landing-point positions. Because saccades were acquired at 1,000Hz we obtained one sample every 1ms. Throughout the paper we will use time units to refer to the available samples. We trained six models that varied in the time available to make the prediction. The first model was trained to predict the landing point from given sequences with only the first 10ms, 15ms, 20ms, 2ms, 30ms or 35ms. The proposed architecture is trained using the Cartesian coordinates of the development saccades, according to the following steps:

a. Each of the saccade development sequences $(\mathbf{x}^i, \mathbf{y}^i)$ with length M^i are truncated with length $N = 10{,}15{,}20{,}25{,}30{,}35)$ depending on which of the six models was trained. Zero padding was applied when $M^i < N$.
b. The truncated training sequences $(\bar{\mathbf{x}}^i, \bar{\mathbf{y}}^i)$ and the target output $\mathbf{T} = \left(x_M^i, y_M^i\right)$ were used to train each of the six models. All LSTMs were trained using Backpropagation algorithm with a Root Mean Square Propagation optimization, learning rate = 0.0002, and 50 epochs.

During the evaluation phase, given a saccade sequence, the first prediction was provided from 10ms and updated every 5ms using its corresponding model and the samples available to make the prediction. The $(\mathbf{x}^i, \mathbf{y}^i)$ coordinates of the saccade were used as input of the trained model (truncated to the nearest N) and the output of the network was provided as the predicted landing point.

2.3 Natural Viewing Database

The database comprised 75 normally-sighted human subjects (median age: 49.3 (22-85) years; 37 female) who participated in two related studies that were approved by the Institutional Review Board of the Schepens Eye Research Institute in accordance with the Code of Ethics of the World Medical Association (Declaration of Helsinki). Preliminary screening of the participants included self-report of ocular health, measures of visual acuity and contrast sensitivity for a 2.5° high letter target and evaluation of fixation and central retinal health using retinal photography (Nidek MP-1, Nidek Technologies, Vigonza, Italy or Optos OCT/SLO, Marlborough, MA). All the participants had visual acuity of 6/7.5 or better, letter contrast sensitivity of 1.675 log units or better, and steady central fixation with no evidence of retinal defects.

In the first study, participants watched 40 to 46 of 206 thirty-second "Hollywood" video clips, which were chosen to represent a range of genres and types of depicted activities. The genres included nature documentaries (e.g., *BBC's Deep Blue, The march of the penguins*), cartoons (e.g., *Shrek, Mulan*), and dramas (e.g., *Shakespeare in love, Pay it Forward*). This group of participants (62 of the 75) contributed a total of 108,640 saccades to the dataset. Participants viewing the 30-second clips were instructed to watch the stimulus "normally, as you would watch television or a movie program at home." At the end of each clip, the participant was asked to describe the contents of the clip [15][26]. In the second study, 14 participants (one was also in the first study) watched at least two of five different 30-minute movie clips (*Bambi, Inside Job, Juno, Kpax,* and *Flash of genius*), contributing 110,695 saccades.

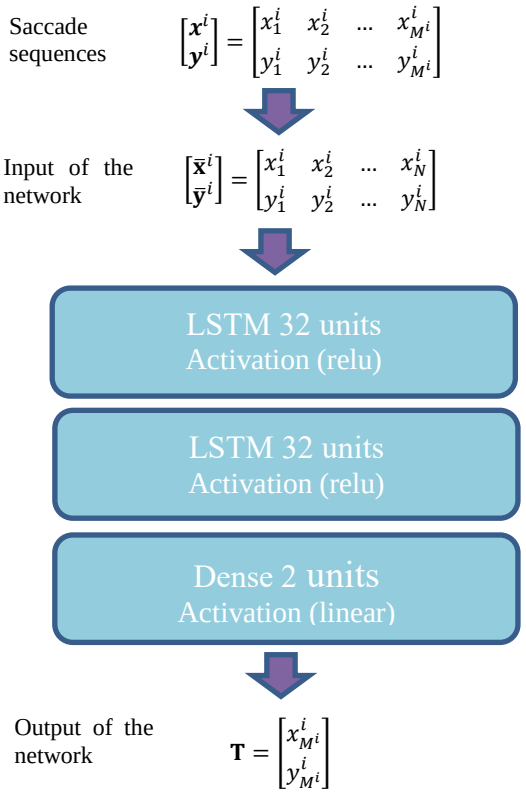

Figure 2. LSTM architecture proposed to predict the landing point of a saccade i with N training samples $(\mathbf{x}^i, \mathbf{y}^i)$.

3. EXPERIMENTS AND RESULTS

The database was divided into development (80% of the users) and evaluation set (20% of the users). The users employed during the training of the models were not used during the evaluation. The process was repeated ten times according to a cross-validation protocol and the results from all ten experiments were averaged in order to present the final results.

In addition to the LSTM approach proposed in Section 2.2, we included two baseline approaches:

- Polynomial fitting [10]: To the best of our knowledge, this method achieves the best performance in saccade landing point prediction related literature. The method defines the prediction of the landing point as a two-fold prediction problem: direction and amplitude. The direction is predicted as the direction of the last observed gaze sample while the amplitude is predicted using a polynomial fitting that minimizes the error between the prediction and the real landing point. In the present work, we have used our

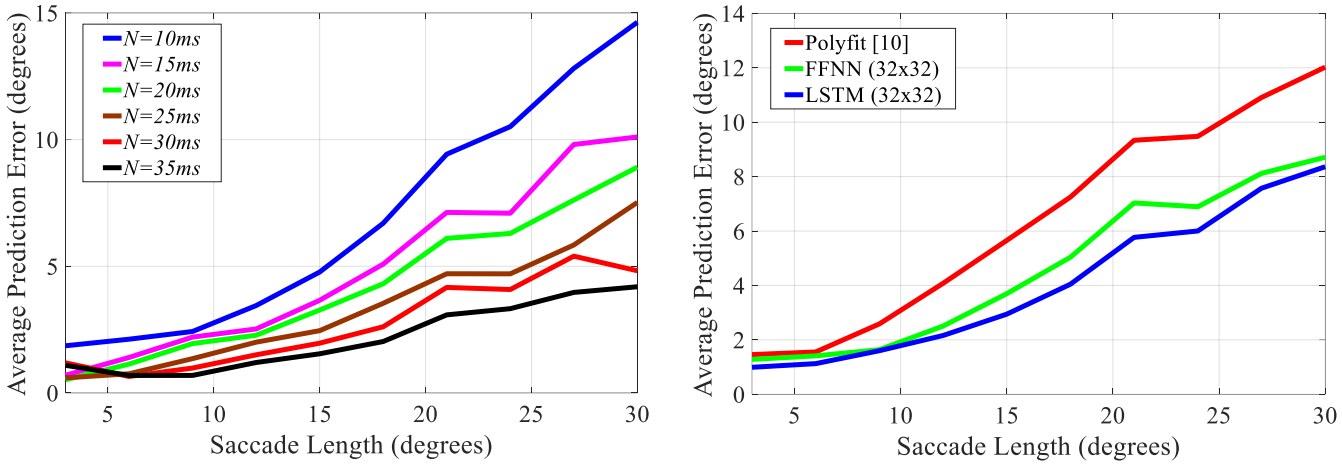

Figure 3. Average Prediction Error vs. Saccade Length of the proposed LSTM approach according to the number of samples (*N*) used for the prediction (a) and comparison with other approaches (b). The performance of the approaches showed as an unique curve on the Right was calculated as the average of the six curves (*N*=10, 15,.., 35).

implementation of the algorithm proposed in [10] performing a polynomial fitting with the same development data set employed to model our proposed LSTM network. We trained six models according to the same protocol employed for the LSTM model.

- Feed Forward Neural Networks (FFNN): During the last three decades, FFNNs have been used to model nonlinear functions [27]. This is the most basic Neural Network architecture. The layers are composed by units (neurons) and adjacent layers are fully connected (every unit from one layer to every unit of the next layer). The data flow from the input layer to the output layer (without memory units as in LSTM networks). The $(\bar{\mathbf{x}}^i, \bar{\mathbf{y}}^i)$ coordinates are concatenated as the input of the network. We present results using two hidden layers with the same number of units each (32 units) and a fully connected output layer with two units. Once again, we trained six models according to the same protocol employed for the LSTM model.

All three approaches were evaluated using the same evaluation set. Results are presented in form of average prediction error (in degrees) depending of the length of the saccade. The error was calculated as the $l^2 - norm$ between the predicted and the real landing points. Figure 3 (a) shows the performance of the proposed approach depending of the number of samples used to make the prediction(N). The results show how the prediction error decreased when more information was available for the prediction. As we showed in Section 2.2, the landing point prediction can be updated once new samples are available. Thus, the error decreases as the saccade progresses. The update is necessary to deal with non-monotonic trajectory of long saccades. The error varies from less than 1 degree for the shortest saccades to 15 degrees for predictions made with only 10ms and the largest saccades (30 degrees).

Figure 3 (b) shows a comparison between the proposed approach and baselines approaches. The error curves for all methods are calculated averaging the six curves (N=10, 15, …, 35) obtained for each of the approaches. The results show the superior performance of our proposed LSTMS network with average improvement of more than two degrees in comparison with state-of-the-art method [10]. The errors vary depending on the saccade length, with errors under three degrees for saccades smaller than 15 degrees, and errors around eight degrees for the largest ones.

Table 1. Number of parameters (assuming an input sequence of 10ms) of the different approaches evaluated.

Approach	# Hidden Layers	# Param.
Polyfit [10]	-	18
FFNN	2 (32×32×2)	1,986
LSTM	2 (32×32×2)	12,866

Table 1 summarizes the number of parameters of the approaches evaluated in this work. While the polynomial method proposed in [10] is defined by 18 parameters, the neural network architectures are comprised of many more parameters that can be successfully trained for landing point prediction due to the large number of saccades available in the datasets that are becoming available.

4. CONCLUSIONS

This work proposes a novel approach for saccade landing point prediction based on LSTM networks. Saccades are rapid eye movements that define the region of interest of a viewer. The prediction of the landing point allows a partial solution to the problems related to the update latency of gaze-contingent displays. The proposed approach was evaluated over a database with 220,000 saccades from 75 users. Data were recorded during natural viewing of videos. The results show the high prediction capabilities of LSTM networks with error rates lower than other state-of-the-art approaches.

5. ACKNOWLEDGMENTS

This work was funded by Jose Castillejo Program (CAS17/00117) from MINECO and partially funded by Neurometrics (CEAL-AL/2017-13) from UAM - Banco Santander, CogniMetrics (TEC2015-70627-R) from Spanish Government agencies

(MINECO)/European Comission grant (FEDER). Ruben Tolosana is supported by a FPU Fellowship from Spanish MECD. The study was supported by National Eye Institute (USA) grants R21EY023724 and Core grant P30EY003790.

6. REFERENCES

[1] Bahill, A. T., Clark, M. R., and Stark, L. 1975. The main sequence, a tool for studying human eye movements. *Mathematical Biosciences.* 24, 3-4 (1975), 191–204.

[2] Opstal, A. V., and Gisbergen. J. V. 1987. Skewness of saccadic velocity pro_les: A unifying parameter for normal and slow saccades. *Vision Research.* 27, 5 (1987), 731–745.

[3] Han, P., Saunders, D. R., Woods, R. L., and Luo, G. 2013. Trajectory prediction of saccadic eye movements using a compressed exponential model. *Journal of Vision*. 13, 8 (2013), 27–27.

[4] Komogortsev. O. V., and Khan, J. I. 2009. Eye movement prediction by oculomotor plant Kalman fillter with brainstem control. *Journal of Control Theory and Applications*. 7, 1 (2009), 14–22.

[5] Komogortsev, O. V., Ryu, Y. S., Koh, D. H., and Gowda, S. M. 2009. Instantaneous saccade driven eye gaze interaction. *In Proc. of the Int.l Conf. on Advances in Computer Enterntainment Technology*. ACM (Athens, Greece, 2009), 140–147.

[6] Komogortsev, O. V., Ryu, Y. S., and Koh, D. H. 2009. Quick models for saccade amplitude prediction. *Journal of Eye Movement Research*. 3, 1 (2009).

[7] Paeye, C., Schütz, A. C., and Gegenfurtner, K. R. 2016. Visual reinforcement shapes eye movements in visual search. *Journal of Vision*. 16, 10 (2016), 15–15.

[8] Yeo, S. H., Lesmana, M., Neog, D. R., and Pai, D. K. 2012. Eyecatch: Simulating visuomotor coordination for object interception. *ACM Transactions on Graphics*. 31, 4 (2012), 42.

[9] Wang, S. Woods, R., L., Costela, F. M., Luo, G. 2017. Dynamic gaze-position prediction of saccadic eye movements using a Taylor series. *Journal of vision*. 17, 14 (2017), 3-3.

[10] Arabadzhiyska, E., Tursun, O. T., Myszkowski, K., Seidel, H. P., and Didyk, P. 2017. Saccade landing position prediction for gaze-contingent rendering. *ACM Transactions on Graphics*. 36, 4 (July 2017), 1–12.

[11] Duchowski, A. T., Cournia, N., and Murphy, H. (2004). Gaze-contingent displays: A review. *CyberPsychology & Behavior*. 7, 6 (2004), 621-634.

[12] Wade, J., Zhang, L., Bian, D., Fan, J., Swanson, A., Weitlauf, A., et al. 2016. A gaze-contingent adaptive virtual reality driving environment for intervention in individuals with autism spectrum disorders. *ACM Transactions on Interactive Intelligent Systems.* 6, 1 (2016), 3.

[13] Reingold, E. M., Loschky, L. C., McConkie, G. W., and Stampe, D. M. 2003. Gaze-contingent multiresolutional displays: An integrative review. *Human Factors.* 45, 2 (2003), 307-328.

[14] Saunders DR, Woods RL. 2014. Direct measurement of the system latency of gaze-contingent displays. *Behavioral Research Methods.* 46 (2014), 439–447.

[15] Saunders, D. R., Bex, P. J., and Woods, R. L. 2013. Crowdsourcing a normative natural language dataset: a comparison of Amazon Mechanical Turk and in-lab data collection. *Journal of Medical Internet Research.* 15, 5 (2013), e100.

[16] Goodfellow, I., Bengio, Y., and Courville, A. 2016. *Deep Learning*. MIT Press, 2016, http://www.deeplearningbook.org.

[17] Sutskever, I., Vinyals, O., and Le, Q. V. 2014. Sequence to Sequence Learning with Neural Networks. *In Proc. Advances in Neural Information Processing Systems*. (Montreal, Canada, 2014).

[18] Zhou, B., Khosla, A., Lapedriza, A., Oliva A., and Torralba, A. 2016. Learning Deep Features for Discriminative Localization. *In Proc. 29th IEEE Conf. on Computer Vision and Pattern Recognition*. (Las Vegas, USA, 2016).

[19] Schmidhuber, J. 2015. Deep learning in Neural Networks: An Overview. *Neural Networks*. 61, (2015), 85–117.

[20] Graves, A., Mohamed, A. R., and Hinton, G. 2014. Towards End-To-End Speech Recognition with Recurrent Neural Networks. *In Proc. International Conference on Machine Learning*. 14, (2014), 1764–1772.

[21] Petrosian, A., Prokhorov, D., Homan, R., Dasheiff, R., and Wunsch, D. 2000. Recurrent Neural Network Based Prediction of Epileptic Seizures in Intra- and Extracranial EEG. *Neurocomputing*. 30, (2000), 201–218.

[22] Graves, A., Liwicki, M., Fernandez, S., Bertolami, R., Bunke H., and Schmidhuber, J. 2009. A Novel Connectionist System for Unconstrained Handwriting Recognition. *IEEE Trans. on Pattern Analysis and Machine Intelligence*. 31, 5 (2009), 855–868.

[23] Hochreiter, S., Bengio, Y., Frasconi. P., and Schmidhuber, J. 2001. Gradient Flow in Recurrent Nets: The Difficulty of Learning Long-Term Dependencies. S.C. Kremer and J.F. Kolen (Eds.), *A Field Guide to Dynamical Recurrent Neural Networks*. 2001.

[24] Hochreiter, S.,and Schmidhuber, J. 1997. Long Short-Term Memory. *Neural Computation*, 9, 8 (1997), 1735–1780.

[25] Tolosana, R., Vera-Rodriguez, R., Fierrez J., and Ortega-Garcia, J. 2018. Exploring Recurrent Neural Networks for On-Line Handwritten Signature Biometric. *IEEE Access*, (2018).

[26] Costela, F. M., Kajtezovic, S., and Woods, R. L. 2017. The preferred retinal locus used to watch videos. *Investigative ophthalmology & visual science*. 58-14, (2017), 6073–6081.

[27] Hornik, K., Stinchcombe, M., White, H. 1989. Multilayer feedforward networks are universal approximators. *Neural Networks.* 2, (1989), 359-366.