Indoor Motion Classification Using Passive RF Sensing Incorporating Deep Learning

Saad Iqbal, Usman Iqbal, Syed Ali Hassan and Sajid Saleem
School of Electrical Engineering and Computer Science (SEECS)
National University of Sciences and Technology (NUST), Islamabad, Pakistan
{14beeuiqbal, 14beesiqbal, ali.hassan, sajid.saleem}@seecs.edu.pk

***Abstract*—This paper proposes a method of recognizing and classifying the basic activities such as forward and backward motions by applying a deep learning framework on passive radio frequency (RF) signals. The echoes from the moving body possess unique pattern which can be used to recognize and classify the activity. A passive RF sensing test-bed is set up with two channels where the first one is the reference channel providing the un-altered echoes of the transmitter signals and the other one is the surveillance channel providing the echoes of the transmitter signals reflecting from the moving body in the area of interest. The echoes of the transmitter signals are eliminated from the surveillance signals by performing adaptive filtering. The resultant time series signal is classified into different motions as predicted by proposed novel method of convolutional neural network (CNN). Extensive amount of training data has been collected to train the model, which serves as a reference benchmark for the later studies in this field.**

***Index Terms*—Adaptive filtering, convolutional neural network, deep learning, passive RF sensing, motion classification**

I. Introduction

Most of the infrastructure these days possesses Wi-Fi systems that are used to connect users with the internet. However, apart from regular internet connectivity, such networks can be used to develop smart systems such as passive RF sensing where we can monitor the premises of a building using its motion detection feature. Such systems may provide freedom to have smart gesture communication to control different devices. Moreover, as the passive RF sensing has no dedicated transmitters, they are power efficient and stealth in nature. Such passive sensing systems can be used on different RF signals, e.g., global system for mobile communication (GSM), frequency modulation (FM) and satellite signals, etc. for various applications in outdoor scenarios.

Passive radars are different than the active radars in the sense that they don't require dedicated transmitters. They use non-cooperative sources of illumination such as the commercial signals to detect the presence of any object as shown in Fig. 1 [1] [2]. WiFi signals, for instance, can be used in a variety of indoor localization and tracking activities. [3] addresses an indoor geometry consisting of various WiFi access points and selects an optimal pair of WiFi transmitters for motion detection such that the detection uncertainty is least when compared to a Cramer-Rao bound. Similarly, [4] used received signal strength and channel state information to perform indoor localization. Many other works use 3G signals to perform outdoor target tracking such as [5] and [6].

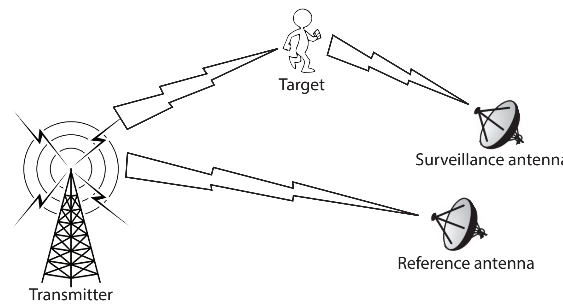

Fig. 1: Bi-static passive RF sensing model

A variety of research has been done in passive sensing with applications ranging from miniscule motion such as human respiration to very high-speed scenarios such as airplane motion. [7] presented a WiFi-based system for medical applications that include fall detection of a human body and tremor detection. [8] proposed and implemented a deep learning framework using passive WiFi sensing to classify and estimate human respiration activity. In [9], an effective signal processing scheme was presented to track moving vehicles and to obtain their cross-range profiles with a passive bistatic radar (PBR) based on the signals of opportunity emitted by a WiFi router. The work focused on targets moving with low radial velocity component which might have reasonable cross-range velocity component enabling to develop a high-resolution cross-range profile using inverse synthetic aperture radar (ISAR) techniques. [10] presented the effects of channel multipath on detection probability using a WiFi illuminator. [11] showed that a WiFi can be used to see moving objects through walls and behind closed doors. It used multiple-input and multiple-output (MIMO) interference nulling to eliminate reflections off static objects and track a human by treating the human body as an antenna array and tracking the resulting RF beam.

Passive sensing has also been utilized in product-based research such as [12]. The authors proposed and implemented a WiFi-based position and orientation agnostic gesture recognition system named WiAG. It required samples of gestures for training in only one configuration and the model generated the samples for other configurations itself accurately. [13] proposed and implemented a channel state information (CSI)-

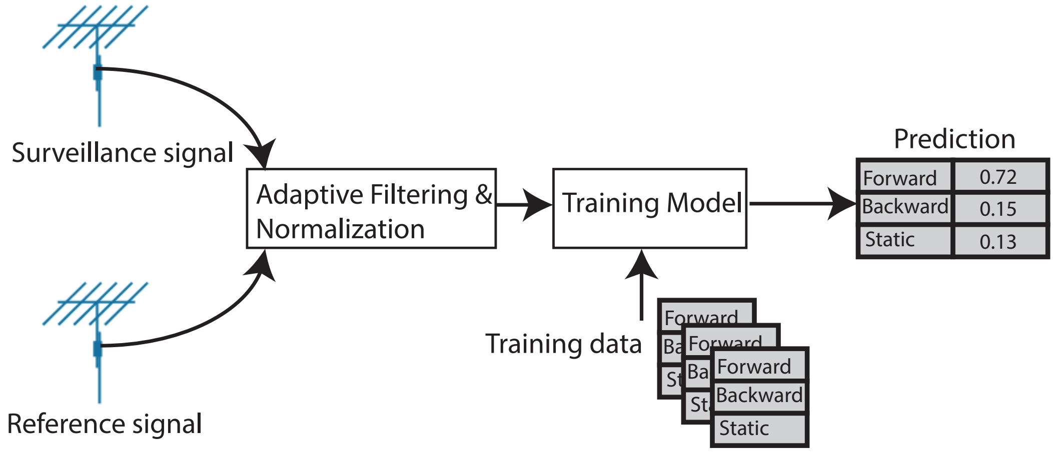

Fig. 2: Block diagram of the proposed RF passive sensing system

based human activity recognition and monitoring (CARM) system which is based on two theoretical models. It first proposed a CSI-speed model that quantifies the relation between CSI dynamics and human movement speeds. Then the CSI-activity model is proposed which quantifies the relation between human movement speeds and human activities. Similarly, [14] presented a system for activity recognition from passive radio-frequency identification (RFID) data using a deep convolutional neural network.

Deep learning network which comprises of cascaded nonlinear processing layers has been an area of active research over the past few years with a large spectrum of applications. Convolutional neural network (CNN) is one of the deep learning models which uses the spatial information of training data to learn and extract the features [15]. Deep learning has been used recently in passive sensing systems such as [8]. However, most studies either focus on detection or classification of an activity or activities. In this paper, we propose and develop a prototype which uses passive RF sensing to detect and classify the type of motion happening in the area of interest (AOI). We use the CNN model and train it beforehand and then apply that to predict the motions in real-time. Promising results have been achieved with the offline data, i.e., data taken in a controlled environment just like the training data, whereas the accuracy slightly drops in case of online data due to factors explained later in the paper.

The rest of the paper is organized as follows. System model will be described in Section II, whereas the procedure of implementations will be explained in Section III. The results and findings as given in Section IV whereas we will conclude our paper in Section V.

II. System Model

This section presents working model of the proposed system.

A. Passive Sensing Test Bed

As shown in Fig. 2, the system consists of a reference antenna which receives the unaltered signals of the illuminator of opportunity whereas the surveillance antenna receives the reflected signals which are altered due to the motion or movements in the environment. Mathematically, the received signal can be represented as

$$s[n] = \sum_{m=0}^{M} A_m x[n+\tau_m] e^{i2\pi f_m n}, \tag{1}$$

where $x[n]$ is the transmitted signal, A is the signal amplitude, τ is the path delay from the signal source to the surveillance antenna, and $f_0, f_1, f_2, ..., f_M$ are the doppler shifts in surveillance signal. For simplicity, we consider single reflection from the moving body, therefore, (1) can be written as

$$s[n] = A_m x[n+\tau_m] e^{i2\pi f_m n}. \tag{2}$$

As $x[n]$ is unpredictable and constantly changing, therefore, we require $s[n]$ to be insensitive of $x[n]$. For this purpose, we use the filtering technique explained next, which provides the only information we require, i.e., the Doppler spread and the received signal strength indicator (RSSI).

B. Adaptive Filtering

As mentioned above, the aim is to remove direct echoes of $x[n]$ from the surveillance signal $s[n]$. The reference signal

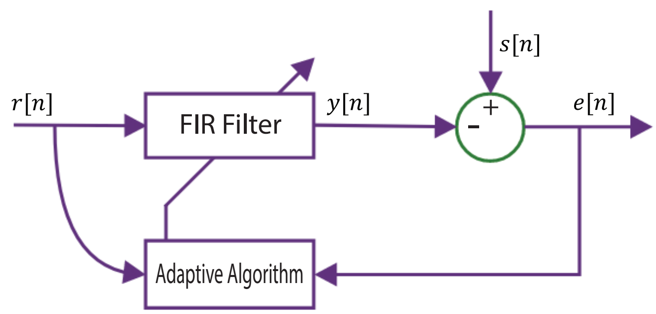

Fig. 3: LMS algorithm applied on the received signal

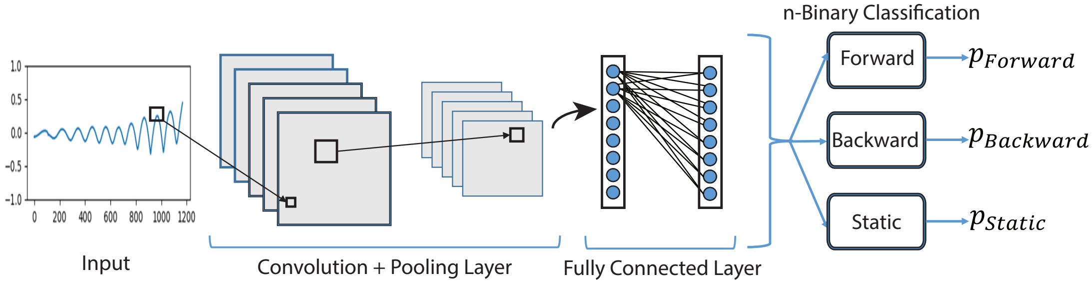

Fig. 4: CNN-based classification model for the proposed system

$r[n]$ can be represented as

$$r[n] = \sum_{m=0}^{M} A_m x[n + \tau_m]. \tag{3}$$

Least mean square (LMS) adaptive filtering technique is used to remove the $r[n]$ component from the surveillance channel $s[n]$. The filtered output $y[n]$ depicted in Fig. 3 is calculated as

$$y[n] = r[n].\omega^T[n], \tag{4}$$

while $\omega^T[n]$ is the transpose of the filter coefficients vector and [.] denotes the dot product operator. The error signal $e[n]$ is the difference of $s[n]$ and $y[n]$. At each iteration, the filter coefficients are updated as

$$\omega^*[n] = \omega[n] + (\mu r[n]e[n]), \tag{5}$$

where ω^* is the updated vector of filter coefficients and μ is the step size parameter which controls the rate of convergence of LMS filter. Through this process, $r[n]$ adjusts the weights of the filter such that $y[n]$ starts to approach $s[n]$. The error signal $e[n]$ is the output of the adaptive filter which contains the required movement features.

C. Classification Model

The classification model is based on CNN. The input to the model is one-dimensional time series signal fragmented into fixed size obtained after the filtering process. As shown in Fig. 4, data passes from input layer to convolution layers and then to fully connected layers for final classification. In convolution layers, the features obtained from previous layer are convolved by a series of convolution filters determined by the depth, D, of the convolution layer, which acts as a matched filter [8] and then, a bias value is added. To enhance the classification ability of the model, more than one convolution layers is used. Hence, the linear unit activation function (ReLu) is applied to the output of the layer to form output feature which maps on next layer [16]. For the down-sampling of information between the convolutional layers, the popular scheme of pooling layer, i.e., max-pooling layer is applied [17]. After the features pass from convolutional layers to fully connected layers (FCL), the FCL transform the mapped feature into the output vector using Softmax function [16].

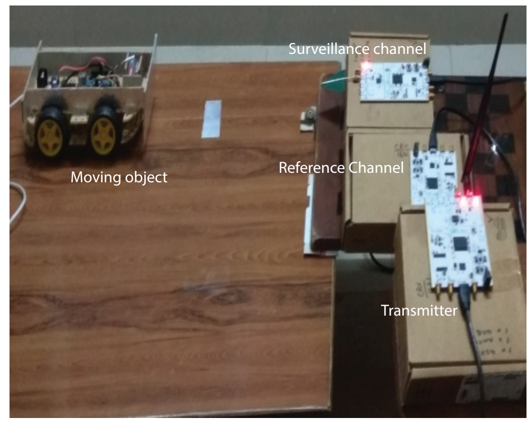

Fig. 5: $1 \times 1m^2$ passive sensing test-bed showing USRP radios

III. Implementation

The system implementation includes hardware and software interfacing, training data collection, CNN model design and construction of the prototype used for real-time classification of motion.

A. Hardware

In a $1 \times 1m^2$ passive sensing test-bed, the universal software radio peripheral (USRP) B200 (software defined radio) is used for transmitting synthetic sinusoid of 1 kHz centered at 3 GHz using omni directional antenna. The transmit power of the USRP is kept at 60-70 dBm. Similarly, two USRP B200s, each with log periodic directional antenna, are deployed as shown in Fig. 5, which act as surveillance channel and reference channel, respectively. These antennas have 6 dBi gain and 60^0 beam-width. The on-board Spartan 6 FPGA in USRP B200 digitizes the received information form the receiving channels with pre-decided sampling rate of 64000 samples/sec.

A custom programmable car, which can perform basic motion in precise manner, is manufactured for emulating the required motion profiles and provides the training data. The car can perform the forward and backward motion with constant average velocity which are the two main categories and are explained further in the subsection of dataset.

B. Software

The GNU Radio application is utilized to configure all the USRP B200s and then the Python script is written to

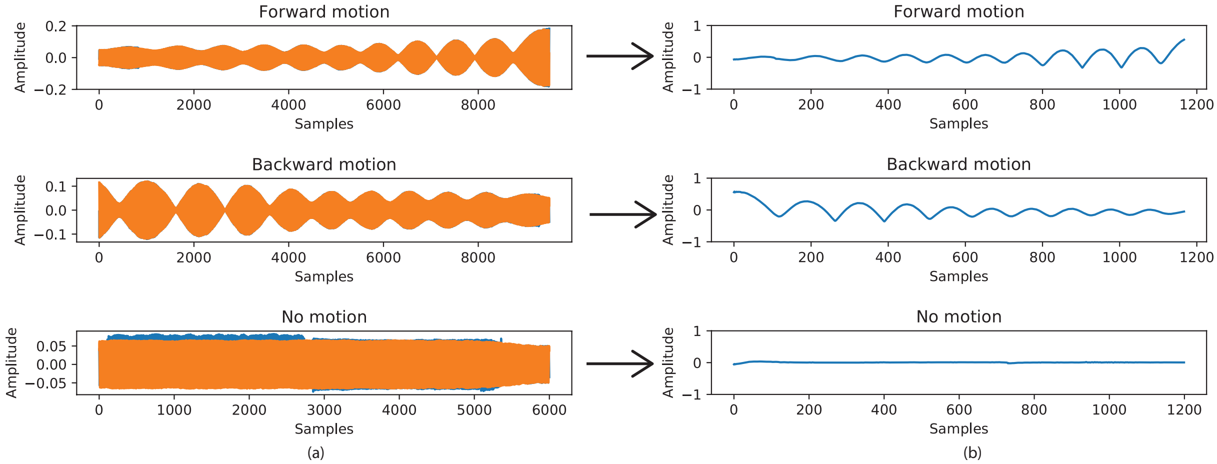

Fig. 6: (a) Received signals at surveillance channel for all three categories. (b) Ready to be used for training after filtering process.

perform the successive operations of adaptive filtering and down-sampling. The time series signal is resampled at 1000 samples/sec, and then the signal stream is sliced into 500 millisecond pieces using sliding window technique implemented in the same Python script. This output, as shown in Fig. 6(b), can now be used as training data or test data for the model.

The CNN model training is accomplished on the Keras running on the top of TensorFlow which is open-source neural network library written in Python and provides modular approach to design the CNN architecture [18]. After the training of model, test data is used for the verification. The verification results are discussed in subsection of results and discussion. The 80% of the gathered data is used for training whereas 20% is subjected as test data for verification of classifier.

C. Data Set

In this paper, we focus on the indoor motions, i.e., hand gestures, walking, nodding, etc. Therefore, the car is programmed to emulate the motion at the walking speed of a human which is found to be 1.4 m/s. The training data was gathered to train and test the CNN model for classification of the basic movements, i.e., forward, backward and static. For this purpose, we collected a 10 minutes sample data for each category in different surroundings to avoid the overfitting of the model to the specific environment. These includes noisy backgrounds and the variations in the altitudes and locations of the transmitter and receivers. The sample data is then segmented into short pieces of 500 milliseconds. Hence, 1200 sample files of each category are generated which are used as the training data and verification of the model.

D. CNN Model

The proposed model is developed in Keras which runs on the top of TensorFlow. The model has a total of eight layers out of which four are convolutional layers and four are fully connected layers. At the end of each layer, dropout is used to avoid the model from specializing on a single set of rules. Also, pooling layer with filter size 1×2 is used to gradually reduce the amount of parameter in model to ease on the computational resources and avoid overfitting [17].

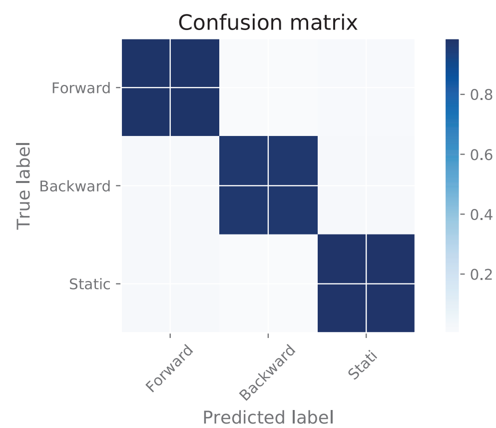

Fig. 7: CNN model verification results

IV. Results and Discussion

This section provides the performance of the proposed system. Fig. 6(a) shows the raw data waveforms attained from the antennas in three different scenarios, whereas, Fig. 6(b) shows the waveforms after they have been passed through the adaptive filtering process to get the waveforms with only the required information. After the training of CNN model with 80% of the total data obtained from training, remaining 20% data was given to the model as the input for the offline testing process. Fig. 7 shows the confusion matrix for the testing which depicts that offline testing provides 99% accuracy which proves the model's reliability for classification of motions.

After this, we conducted the experiments at different distances from the surveillance receiver. Results obtained from

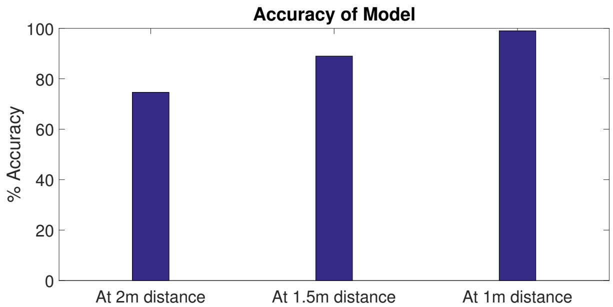

Fig. 8: Accuracy of model for different distances between motion and passive sensing system.

these experiments, providing the accuracy of model for three different distances, are shown in Fig. 8. At the distance of around 1.5m, the model shows 89% accuracy and going further away, to the range of 2m, decreases this accuracy to 74%. Hence, the model accuracy drastically decreased as location of movements is moved away from the passive sensing system. This is due to the decrease in the RSSI which makes it impossible for the model to predict correctly.

For the real-time testing of our model, the setup receives signals for two seconds and predicts the motion profile in the real-time. After excessive evaluation, accuracy of the model in real-time is around 70%. Truncated part from this evaluation process, as shown in Fig. 9, depicts that the model fails to correctly predict the transitions (deceleration) of the motions which results in the drop of accuracy of the model. The signals received at the transitions are closely resembled to the motion profile of opposite movement which causes the model to predict the forward motion as backward motion and vice versa at transitions. In order to mitigate this issue, the previous predictions from the model should be considered along with the current prediction to evaluate the motion profile.

V. Conclusion

In this paper, we have analyzed the effectiveness of CNN model in the classification of basic motions in a passive sensing system. The results prove that this approach is reliable and accurate. Few challenges such as prediction of transition motion open up new research for optimizing the model. The accuracy decreases drastically as the distance increases between movement and the sensing system because of low RSSI. Hence, better and high power opportunistic transmitters can provide high accuracy at larger distances such as apartments or rooms. It is concluded that CNN network can replace the classical systems for movement tracking in passive sensing domain as it is fast and predicts the movements accurately.

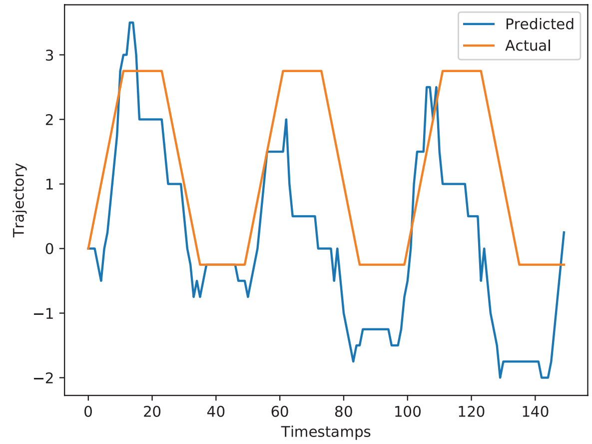

Fig. 9: Results from online evaluation of data.

References

[1] Q. Chen et. al, “Indoor target tracking using high doppler resolution passive Wi-Fi radar”, *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Apr. 2015.

[2] Q. Chen et. al, “Signs of life detection using wireless passive radar”, *IEEE Radar Conference (RadarConf)*, May 2016.

[3] J. Abdullah, S. A. Hassan, “Geometry optimization for WiFi-based indoor passive multistatic radars”, *International Wireless Communications and Mobile Computing Conference (IWCMC)*, Jun. 2017.

[4] M. Kotaru et. al, “SpotFi: Decimeter level localization using WiFi”, *ACM Special Interest Group on Data Communication (SIGCOMM)*, Aug. 2015.

[5] M. N. Javed et. al, “3D MCRLB evaluation of a UMTS-based passive multistatic radar operating in a line-of-sight environment”, *IEEE Transactions on Signal Processing*, vol. 64, no. 19, pp. 5131-5144, May 2016.

[6] M. N. Javed, S. A. Hassan, S. Ali, “Characterizing the impacts of rice dading on a WiFi-based passive multistatic radar using Cramer-Rao bound”, *Proceedings of the IEEE Radar Conference*, May 2016.

[7] U. M. Khan, Z. Kabir, S. A. Hassan, “Wireless health monitoring using passive WiFi sensing”, *International Wireless Communications and Mobile Computing Conference (IWCMC)*, Jun. 2017.

[8] U. M. Khan, Z. Kabir, S. A. Hassan, “A deep learning framework using passive WiFi sensing for respiration monitoring”, *IEEE Global Communication Conference (Globecom)*, Dec. 2017.

[9] F. Colone et. al, “WiFi-based passive ISAR for high-resolution cross-range profiling of moving targets”, *European Conference on Synthetic Aperture Radar (EUSAR)*, Apr. 2012.

[10] H. Mazhar, S. A. Hassan, “Analysis of target multipaths in WiFi-based passive radars”, *IET Radar, Sonar & Navigation*, vol. 10, no. 1, pp. 140-145, Jan. 2016.

[11] F. Adib, D. Katabi, “See through walls with Wi-Fi!”, *ACM Special Interest Group on Data Communication (SIGCOMM)*, Aug. 2013.

[12] A. Virmani, M. Shahzad, “Position and orientation agnostic gesture recognition using WiFi”, *Proceedings of the Annual International Conference on Mobile Systems, Applications, and Services (MobiSys)*, Jun. 2017.

[13] W. Wang et. al, “Device-free human activity recognition using commercial WiFi devices”, *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 5, pp. 1118-1131, May 2017.

[14] X. Li et. al, “Deep learning for RFID-based activity recognition”, *ACM Conference on Embedded Networked Sensor Systems (SenSys)*, Nov. 2016.

[15] X. Wang et. al, “CSI-based fingerprinting for indoor localization: a deep learning approach”, *IEEE Transactions on Vehicular Technology*, vol. 66, no. 1, pp. 763-776, Jan. 2017.

[16] C. Szegedy et. al, “Going deeper with convolutions”, *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2015.

[17] H. Wu, X. Gu, “Max-pooling dropout for regularization of convolutional neural netwroks”, *Neural Networks*, Nov. 2015.

[18] F. Chollet et. al, “Keras”, https://github.com/keras-team/keras.