Age Group Detection Using Smartphone Motion Sensors

Erhan Davarci, Betul Soysal, Imran Erguler, Sabri Orhun Aydin, Onur Dincer, Emin Anarim
Electrical and Electronics Engineering Department, Bogazici University, Bebek, Istanbul 34342, Turkey
{erhan.davarci,betul.soysal,imran.erguler,orhun.aydin,onur.dincer,anarim}@boun.edu.tr

***Abstract*—Side-channel attacks revealing the sensitive user data through the motion sensors (such as accelerometer, gyroscope, and orientation sensors) emerged as a new trend in the smartphone security. In this respect, recent studies have examined feasibility of inferring user's tap input by utilizing the motion sensor readings and propounded that some user secrets can be deduced by adopting the different side-channel attacks. More precisely, in this kind of attacks, a malware processes outputs of these sensors to exfiltrate victims private information such as PINs, passwords or unlock patterns. In this paper, we describe a new side-channel attack on smartphones that aims to predict the age interval of the user. Unlike the previous works, our attack does not directly deal with recovering a target user's some secret, rather its sole purpose is determining whether she is a child or an adult. The main idea behind our study relies on the key observation that the characteristics of children and adults differ in hand holding and touching the smartphones. Consequently, we show that there is an apparent correlation between the motion sensor readings and these characteristics that build up our attack strategy. In order to exhibit efficiency of the proposed attack, we have developed an Android application named as *BalloonLogger* that evaluates accelerometer sensor data and perform child/adult detection with a success rate of 92.5%. To the best of our knowledge, in this work, for the first time, we point out such a security breach.**

I. Introduction

Most of the smartphones have built-in motion sensors such as the accelerometer, gyroscope, and orientation sensors to gain information about the movement and orientation of the device. These sensors greatly expedites development of creative and context-aware applications, e.g. high accuracy in game control has been provided. In terms of security, the motion sensors are not considered as exposing sensitive information, so the popular mobile operating systems (iOS, Android) allow all installed applications to access the sensor data. However, recent studies, like [1], [2], [3], [4], [5], [6], [7] have demonstrated that user security can be compromised by employing device motion sensor data as a side-channel. Specifically, the presented attacks infer user taps on smartphone touchscreens by processing motion sensor readings, i.e user taps on the soft keyboards of smartphones can be guessed. Consequently, by conducting the proposed attacks, user secret information such as creditcard numbers, PINs, passwords or screen lock patterns can be revealed [8].

The results of the aforementioned research point out two common facts: First, motion sensors may carry side-channel information and they can be exploited to capture user sensitive data by an adversary. Second, touch behaviour of a user may assist verification of users, i.e. behavioral biometrics can be derived from the smartphone sensors for user identification [9], [10], [11].

In this paper, we focus on the key observation that smartphone touching and holding attributes differ for a child and an adult, e.g. strength, angle preferences of the applied force, use of single hand or two hands. In order to disclose the discriminative patterns for age groups, we have extracted principle features by analyzing correlation between accelerometer output and the user's touch and hold behaviour. In this regard, we have developed an Android app, *BalloonLogger*, that collects accelerometer data while user playing balloon popping.

In particular, after data is collected through our application (50 touch events from each user), by using the signal processing techniques we derive 16 features and calculate their values. Finally, our proposed algorithm yields the boolean answer whether the user is a child or an adult. Notice that the goal of our work is just detecting the age interval of the user, in contrast to previous work that intend to recover sensitive information like PIN, password, credit numbers etc.

The contribution of our paper is twofold: First, we introduce the possibility that user age group can be detected by evaluating the accelerometer sensor readings as a side-channel source. We demonstrate its practicality by collecting data from users through our test application and indicating its high success rates in distinguishing child and adult users. Second, we examine the success of our model in case of the test application running in the background and show that it can still detect user age interval with a success rate of 89%. To our best knowledge, there is no previous study using the motion sensors of smartphones as side-channel to infer user age interval.

The rest of this paper is organized as follows: In Section II, we introduce the threat model and give the assumptions made. Section III describes the details of our scheme and presents the evaluated results. In Section IV, we show the success rate of our model in case of Silence Mode. Finally, we conclude this paper in Section V.

II. Assumptions and Threat Model

First of all, we consider an attacker who wishes to control the smartphone illegally. Moreover, we assume that malware is capable of determining whether the current user of the smartphone is a child or an adult. Notice that this feature facilitates

ISBN 978-0-9928626-7-1

attacker to accomplish her primary goals more easily. Once the malware detects the current user is a child, it requests permissions or authorizations from the child to realize its next actions, e.g. downloading other malicious applications, getting revenue via billed events, exfiltration of personal information and user credentials or showing inappropriate content [13], [14].

On the contrary, malware suspends its suspicious activities while an adult user is recognized, so it hides its presence in the system. In other words, in case of an adult user, it proceeds to carry out its normal activities without drawing attention to its illegal actions. By taking this facts into consideration, one can see the significance of child/adult user information in terms of security.

Actually, there are large numbers of studies about biometric age-group classification. The proposed methods try to extract age related attributes from facial images [12]. However, a malicious application usually have no access to camera because accessing a camera requires a specific permission of users. In this study, we therefore investigate success of this malware in distinguishing child and adult user, if only accelerometer sensor data is available to it. Hereof, we take two scenarios into account. In the first scenario, user interacts with the malware as a legitimate app, e.g. a game app, and malware collects accelerometer data to determine whether the user is a child or an adult, while running in the foreground. In this mode, named as *Active Mode*, the app can access to touch event information. On the other hand, in the second scenario which we call *Silent Mode*, the malware runs as a background process and cannot obtain touch events directly from OS. Instead, it internally detects both of touch events and user age by analyzing accelerometer readings. In order to test these scenarios, we have developed an Android app *BalloonLogger* that runs either in *Active* or *Silent Mode* depending on the configuration.

III. Methodology

A. Data Acquisition

In order to collect data, we have designed and implemented *BalloonLogger* by using Android Studio development environment. This application logs accelerometer data in x, y and z directions while users touch the screen. Users are not guided about how they hold or touch the device (one hand/two hand holding, pressing with index finger/thumb), while playing *BalloonLogger*. An example user interface of our test application is illustrated in Figure 1. The experiment starts after user age is entered and "`START`" button is pressed. Then, a balloon through different colors and sizes appears on a random position of the touch screen. As he/she pops the balloon by touching it on the screen, next one is appeared with a random color, size and position. A total of 50 balloons are created in this way during the experiment. While our application is running, all accelerometer data and touch event information are recorded with specified sampling rate and they are saved to the internal memory of the smartphone.

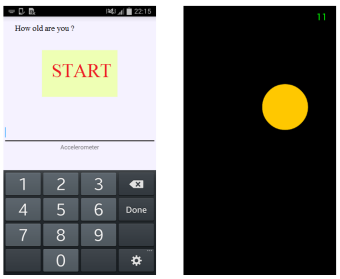


Fig. 1: The user interface of developed sensor application.

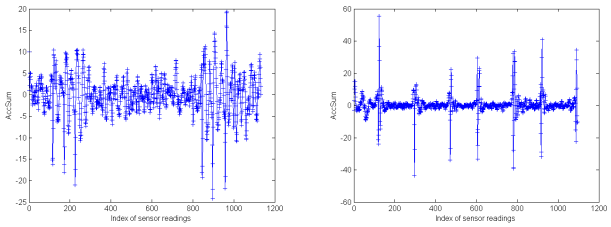
In the phase of algorithm development and training, we have collected data from 100 child users whose age vary from 3 to 11, and 100 adult users whose age are between 12 and 50. Therefore, there are more than 5000 taps data for adult and child users, separately. We use popular Android smartphones like Samsung Galaxy S3 (Android 4.3), Galaxy S4 (Andorid 4.4), Galaxy S5 (Android 5.0), LG G3 and G4 (Android 6.0) during experiment and they have all 100 Hz sampling rate. Using different smartphone models with different screen size shows that the proposed approach can be used on any Android smartphone with sufficient sampling rate.

B. Feature Extraction

While we analyze accelerometer readings, we notice that there is unique pattern in tap locations and this is the fingerprint of tap events. Actually, this change in accelerometer data is due to the external force applied on the touchscreen [3]. Therefore, we use $AccSum$ term to measure the change of external force. $AccSum$ is 2-norm of acceleration vector and represented by the following formula:

$$AccSum = |A|^2 = A_x^2 + A_y^2 + A_z^2, \tag{1}$$

where A_x, A_y, A_z are acceleration values in x, y, z dimensions, respectively. $AccSum$ is directly proportional to $|F|^2$. Examples of how $AccSum$ value changes for child and adult users are shown in Figure 2.



(a) A child (b) An adult

Fig. 2: Example $AccSum$ graph for a child and an adult user.

The fingerprint of a tap event is exhibited in Figure 3. The motions of smartphone during the tap event can be expressed as three consecutive phases: $Action\_Down$, $Action\_Hold$ and $Action\_Up$. When the user taps on the touchscreen, the smartphone will move downward and this is called as

ISBN 978-0-9928626-7-1 © EURASIP 2017

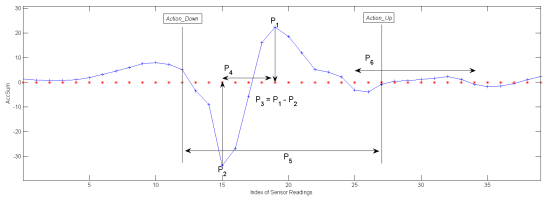


Fig. 3: The fingerprint of tap event.

TABLE I: The Effects of 6 Features on Accuracy Rate

Features	Accuracy Rate
P_1, P_2, P_3, P_4, P_5, P_6	71.8%
P_1, P_2, P_4, P_5, P_6	70.8%
P_1, P_2, P_3, P_4, P_5	70.5%
P_1, P_2, P_6	61.8%
P_1, P_2, P_3	52.5%
P_1, P_2	52.4%

$Action\_Down$ phase. When the $Action\_Down$ is over; the smartphone will stop, this step is called $Action\_Hold$ phase. Then, he/she lifts his finger and the hand holding the smartphone will cause the smartphone back to its beginning position, $Action\_Up$ phase.

After normalizing AccSum value, we firstly extract 6 features for each tap event:

1) P_1: The peak value of AccSum at the end of $Action\_Down$ phase.
2) P_2: The minimum reading of AccSum in $Action\_Down$ phase.
3) P_3: Difference between P_1 and P_2 values.
4) P_4: Difference between P_1 and P_2 in terms of sample index.
5) P_5: Time difference between $Action\_Down$ and $Action\_Up$ events.
6) P_6: Variance of AccSum after $Action\_Up$ phase to settle the original position.

Actually, P_1 measures the magnitude of tapping finger, P_2 measures the magnitude of force from the hand holding the smartphone, P_3 and P_4 measure the change rate, P_5 determines tap time and P_6 measures the fluctuation. According to our observations, we see that press magnitude of tapping finger of children is weaker compared to adult users. Similarly, adult users hold smartphone more powerfully, so reactions due to their holding hands are more strong compared to children. Therefore, we expect lower P_2 values and higher P_1 values for adult users. Additionally, our observations show that children touch on the screen longer than adult users, so we expect higher P_4 and P_5 values for children. Furthermore, we have noted that following the $Action\_Up$ state, smartphone needs more time to return its original position in case of a child user. The reason is obvious: Children's holding hands are weak and settle time is higher compared to adults. Relying on this fact, we also introduce P_6 parameter as the sixth feature.

In order to classify extracted features, we perform multi-class logistic regression by using *LIBLINEAR* implementation [16]. By using the logistic regression with 5-fold cross validation, we obtained 71.8% accuracy rate. In other words, we can determine a given tap whether belongs to child or adult user with 71.8% accuracy. We also tested effects of 6 features on the accuracy rate, the result are show in Table I.

In addition to these 6 features, we examine the distribution of each tap event. All tap events do not have the same number of samples even for different taps of the same user. Therefore, we firstly transform data of each tap to frequency domain and then retranform it to the time domain to reconstruct the original signal. In reconstruction, we determine the length of transformation as 10 samples. This operation preserves general shape of the signal but normalize the signal to 10 samples in time domain and also eliminates noisy high frequency components. For this operation, we first apply discrete Fourier transform (DFT) and then inverse discrete Fourier transform (IDFT) to tap event data. By setting DFT size equal to 10, we fix all tap events to 10 samples so we can use each of these 10 values as a feature. Correspondingly, DFT and IDFT are equations given below.

$$F[n] = \sum_{k=0}^{N-1} f[k] e^{-j2\frac{2\pi}{N}nk} \quad n = 0 : N-1 \tag{2}$$

$$f[k] = \frac{1}{N} \sum_{n=0}^{N-1} F[n] e^{+j2\frac{2\pi}{N}nk} \quad k = 0 : N-1 \tag{3}$$

Instead of DFT, we also apply Discrete Cosine Transform (DCT) to our data. We again set the transformation sample size as 10. After calculations based on the experimental data, we observe that these two methods result in nearly the same performance. The equations of DCT and IDCT are given in (4) and (5). For example, the original sensor data for one tap event and corresponding reconstructed data via IDCT are shown in Figure 4.

$$X_c[k] = \sum_{n=0}^{N-1} x[n] cos \frac{\pi(2n+1)k}{2N} \quad k = 0 : N-1 \tag{4}$$

$$x[n] = \frac{1}{N} X[0] + \frac{2}{N} \sum_{k=1}^{N-1} X[k] cos \frac{\pi(2n+1)k}{2N} \quad n = 1 : N-1 \tag{5}$$

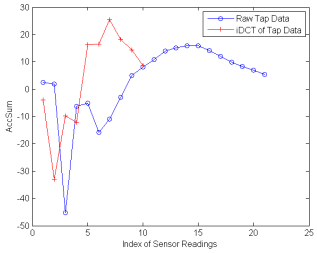


Fig. 4: Example IDCT of tap data.

ISBN 978-0-9928626-7-1 © EURASIP 2017

Then, each 10-length signal acquired from the inverse transforms is used as features. How accuracy rate changes when these parameters are adopted is given in Table II.

TABLE II: The Effects of IDFT and IDCT on Accuracy Rate

Features	# of Features	Accuracy Rate
P_1, P_2, P_3, P_4, P_5, P_6	6	71.8%
P_1, P_2, P_3, P_4, P_5, P_6, IDFT	16	77.1%
P_1, P_2, P_3, P_4, P_5, P_6, IDCT	16	77.3%
Only IDFT	10	70.5%
Only IDCT	10	71.4%

C. Classification

In order to test succession of the different classisfication methods, WEKA (Waikato Environment for Knowledge Analysis) machine learning program is used. For 16-featured data, 3 different classification methods are experimented. These are logistic regression, k-nearest neighbor and random forest. Additionally, in order to decrease the number of features, PCA (Principal Component Analysis) is also implemented. With PCA, the number of feature is reduced from 16 to 11. The highest success rate of 85.3% is obtained by using k-nearest neighbor.

TABLE III: Accuracy Results for 3 Classification Methods

Dimension Reduction	Classification Method	Accuracy Rate
none, 16 features	Logistic Regression	77.1%
none, 16 features	k-NN with k=10	84.5%
none, 16 features	k-NN with k=1	81.0%
none, 16 features	RandomForest Tree	85.0%
PCA, 11 features	Logistic Regression	77.3%
PCA, 11 features	**k-NN with k=10**	**85.3%**
PCA, 11 features	k-NN with k=1	81.6%
PCA, 11 features	RandomForest Tree	85.1%

Up to now, the classisfication results were the rate of truly finding that a tap belongs to an adult or a child. However, our main purpose is to determine whether a user is an adult or a child. For this purpose, we constiute our training set consisting of 100 adult and 100 child users, and test each user's data separately and finally classify the user as a adult or a child. Furthermore, we try to determine number of minimum tap samples from each user for a succesfull adult/child identification. In this direction, we take 10, 20 and 30 taps from collected date set of users. Finally, we have a success rate in distinguishing a user as an adult or a child. For this purpose, we tested collected data of 100 adult and 100 children, and the results are given in Table IV. Note that, if we take 30 taps from each user, we can correctly identify 92 out of 100 adult users and 93 out of 100 child users, so our average success rate in *Active Mode* is 92.5%.

IV. Tap Detection in Silence Mode

When services and applications run in background, they are not allowed to retrieve touch event information from the touchscreen of the smartphone. As it can seen from the unique

TABLE IV: Success Results for User Identification in Active Mode

Users	Number of taps	Success Rate
100 Adult and 100 Child Users	30	92.5 %
100 Adult and 100 Child Users	20	89 %
100 Adult and 100 Child Users	10	88 %

pattern of a tap event, there are some peaks in the accelerometer data when the tap occurrs. In order to detect taps from the accelerometer data, one should determine the maximum and minimum peaks. In the *Silent Mode*, we are monitoring $AccSum$ reading by using Teager's Energy Operator. Teager's energy operator is such a nonlinear operator that can be used at this stage to detect energy jump in the accelerometer data since integral of amplitude square of a signal gives the energy of the signal. This operator is beauiful since it has a small time window, making it ideal for time analysis of signal [17]. It is defined in one-dimension [18] as:

$$T[f(t)] = (\frac{df(t)}{dt})^2 - f(t)\frac{d^2f(t)}{dt^2}, \tag{6}$$

while in discrete domain, the operator becomes as

$$T[f(n)] = f^2(n) - f(n-1)f(n+1). \tag{7}$$

Thus, after the application of Teager's energy operator on the accelerometer data, index values are thresholded so that only the highly energetic jump locations are detected. Basically, the Teager's energy operator gives the notion of start and end location of tap events as shown in Figure 5. Morever, in order to improve accuracy in tap detection, these highly energetic locations are evaluated and thresholded according to their P_1, P_2, and P_4 values in the bulk of accelerometer data so that these locations are double checked and recorded as actual tap events.

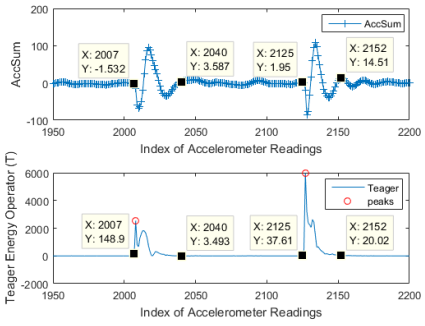


Fig. 5: The relation between Tap Events and Teager's Energy Operator.

In the evaluation step of tap detection algorithm, we used the collected accelerometer data of 200 users, 5000 taps for adult and child users, separately. This was great effort to form large data set for training phase. We investigated that there

are similar studies in which data from only 3 or 4 users are used for training set. The experimental results show that proposed tap event detection approach can be able to achieve high accuracies in identifying the tap events on the touchscreen as it can be seen from Table V. The reason of decrease in the accuracy rate of child users is due to high amplitude noises which are originated from shakes and vibrations caused by the users.

TABLE V: Experimental Results of Tap Event Detection

User Type	Precision	Recall	F-Measure
Adult	93.2%	81.2%	86.8%
Child	73.3%	64.8%	68.8%

In case of the *Active Mode*, we suppose that the exact location of the tap events is available. Nevertheless, for *Silent Mode* tap events has to be guessed by applying the proposed tap detection algorithm. When our tap detection algorithm is implemented, the success rate for identifying 100 adult and 100 child users becomes about 89% as depicted in Table VI. There are some decreases in accuracy rates when we use our proposed tap detection algorithm instead of touch event information from Android, since our tap detection algorithm sometimes misses the tap events in sensor data or gives some false tap locations.

TABLE VI: Success Results for User Identification in Silent Mode.

Users	Number of taps	Success Rate
100 Adult and 100 Child Users	30	89 %
100 Adult and 100 Child Users	20	89 %
100 Adult and 100 Child Users	10	87 %

V. Conclusion

In this study, we have examined the feasibility of detecting user age interval by analyzing accelerometer sensor data. In particular, we have designed and implemented the Android app *BalloonLogger* to collect sensor readings from both of the child and adult users while they are interacting with this application. We have demonstrated that our proposed scheme can determine whether the present user is a child or an adult with a success rate of 92.5%. Furthermore, the results of the paper show that our model can still detect the user age interval successfully up to $\%89$, even it runs in the background as *Silent Mode*. We have also addressed that child/adult information can be exploited by a malware which asks a child user to authorize its hazardous actions and hides its suspicious behavior for an adult user. Hence, we conclude that such an easily accessible information may make malwares more powerful and devious. Furthermore, our proposed method can also be developed by using additional classification schemes like precise age interval classification or gender recognition. These subjects will be considered in future works.

Acknowledgment

This work is supported by the Turkish Ministry of Development under the TAM Project Number DPT2007K120610 and by Bogazici University Research Fund, Grant Number 10129.

References

[1] A. J. Aviv, "Side Channels Enabled by Smartphone Interaction," Ph.D. dissertation, Pennsylvania State University, 2012.

[2] L. Cai and H. Chen, "Touchlogger: Inferring keystrokes on touch screen from smartphone motion." in *HotSec*, 2011.

[3] Z. Xu, K. Bai, and S. Zhu, "Taplogger: Inferring user inputs on smartphone touchscreens using on-board motion sensors," in *Proceedings of the Fifth ACM Conference on Security and Privacy in Wireless and Mobile Networks*. ACM, 2012, pp. 113–124.

[4] E. Owusu, J. Han, S. Das, A. Perrig, and J. Zhang, "ACCessory: Password Inference Using Accelerometers on Smartphones," in *Proceedings of the Twelfth Workshop on Mobile Computing Systems & Applications*, ser. HotMobile '12. ACM, 2012, pp. 9:1–9:6. [Online]. Available: http://doi.acm.org/10.1145/2162081.2162095

[5] E. Miluzzo, A. Varshavsky, S. Balakrishnan, and R. R. Choudhury, "Tapprints: Your Finger Taps Have Fingerprints," in *Proceedings of the 10th International Conference on Mobile Systems, Applications, and Services*, ser. MobiSys '12. New York, NY, USA: ACM, 2012, pp. 323–336.

[6] C. Shen, S. Pei, T. Yu, and X. Guan, "On Motion Sensors as Source for User Input Inference in Smartphones," in *Identity, Security and Behavior Analysis (ISBA), 2015 IEEE International Conference on*. IEEE, 2015, pp. 1–6.

[7] P. Bissig, P. Brandes, J. Passerini, and R. Wattenhofer, "Inferring Touch From Motion in Real World Data," in *8th International Symposium on Foundations & Practice of Security (FPS), Clermont-Ferrand, France*, October 2015.

[8] A. J. Aviv, B. Sapp, M. Blaze, and J. M. Smith, "Practicality of Accelerometer Side Channels on Smartphones," in *Proceedings of the 28th Annual Computer Security Applications Conference*. ACM, 2012, pp. 41–50.

[9] C. Bo, L. Zhang, X.-Y. Li, Q. Huang, and Y. Wang, "Silentsense: Silent User Identification via Touch and Movement Behavioral Biometrics," in *Proceedings of the 19th Annual International Conference on Mobile Computing & Networking*. ACM, 2013, pp. 187–190.

[10] A. De Luca, A. Hang, F. Brudy, C. Lindner, and H. Hussmann, "Touch me once and i know it's you!: Implicit authentication based on touch screen patterns," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2012, pp. 987–996.

[11] N. Zheng, K. Bai, H. Huang, and H. Wang, "You Are How You Touch: User Verification on Smartphones via Tapping Behaviors," in *Network Protocols (ICNP), 2014 IEEE 22nd International Conference on*. IEEE, 2014, pp. 221–232.

[12] H. Han, C. Otto, and A. K. Jain, "Age estimation from face images: Human vs. machine performance," in *Biometrics (ICB), 2013 International Conference on*. IEEE, 2013, pp. 1–8.

[13] P. Meshram and R. Thool, "A Survey Paper on Vulnerabilities in Android OS and Security of Android Devices," in *Wireless Computing and Networking (GCWCN), 2014 IEEE Global Conference on*. IEEE, 2014, pp. 174–178.

[14] G. Suarez-Tangil, J. E. Tapiador, P. Peris-Lopez, and A. Ribagorda, "Evolution, Detection and Analysis of Malware for Smart Devices," *Communications Surveys & Tutorials, IEEE*, vol. 16, no. 2, pp. 961–987, 2014.

[15] S. Read, "In-app Charges: Parents Fork out Estimated 30M after Children Use 'Freemium' Online Games," *Independent*, 2013.

[16] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, "LIBLINEAR: A Library for Large Linear Classification," *The Journal of Machine Learning Research*, vol. 9, pp. 1871–1874, 2008.

[17] E. Kvedalen, "Signal processing using the teager energy operator and other nonlinear operators, cand," *Scient Thesis, University of Oslo, Department of Informatics*, 2003.

[18] J. F. Kaiser, "On a Simple Algorithm to Calculate Theenergy'of a Signal," in *Acoustics, Speech, and Signal Processing, 1990. ICASSP-90., 1990 International Conference on*. IEEE, 1990, pp. 381–384.