Energy Disaggregation with NILM on a Raspberry Pi with Smart-Metering Extension

Johannes Winkler
Institute of Networked and Embedded Systems
University of Klagenfurt
Klagenfurt, Austria
johannes.winkler@aau.at

Hafsa Bousbiat
Institute of Networked and Embedded Systems
University of Klagenfurt
Klagenfurt, Austria
hafsabo@edu.aau.at

Stefan Jost
Institute of Networked and Embedded Systems
University of Klagenfurt
Klagenfurt, Austria
stjost@edu.aau.at

Wilfried Elmenreich
Institute of Networked and Embedded Systems
University of Klagenfurt
Klagenfurt, Austria
Wilfried.Elmenreich@aau.at

***Abstract*—Non-intrusive load monitoring (NILM) is emerging as a crucial technique for providing detailed and effective energy feedback, thereby facilitating the development of low-cost and low-energy management systems in the residential sector. However, achieving acceptable disaggregation results requires sampling frequencies that exceed the 15-minute intervals of commercial smart meters. State-of-the-art approaches require a measurement frequency of at least 1Hz. While technically possible, measurement devices that can provide measurements at such intervals increase the cost of the overall system. In addition, consumers may raise privacy concerns regarding these systems. To address these issues, we propose a low-cost single-device smart meter that provides direct feedback based on a local processing of the user's data. The proposed system leverages the Raspberry Pi and the YoMoPie Monitor to provide an efficient, compact, and accurate system. The system's performance was tested in a laboratory setting under two different scenarios, and promising results were obtained considering the disaggregation performance and computational complexity.**

***Index Terms*—NILM, Smart Metering, Energy Awareness**

I. Introduction

The increasing energy demand in the residential sector has been an appealing problem in the last decade. Designing more robust and privacy-preserving technologies that support consumers in improving their energy efficiency is thus becoming more and more important. Energy is often wasted unwittingly by forgetting to switch off devices or through devices in standby mode. These wastes can be avoided through real-time feedback about the energy consumption. Several works, including [4], [7], and [19] show the connection between energy awareness and energy saving. The given works clearly demonstrate that there is a correlation between a detailed and graphically prepared energy measurement and the potential for savings (see figure 1).

Several initiatives have already been initiated worldwide to raise energy awareness and help consumers adopt an environmentally-friendly consumption. An example of these initiatives is the extensive deployment campaign of smart meters around the European Union countries, including the Energy Services Directive (2006/32/EC) and the Electricity Directive (2009/72/EC), requiring the implementation of intelligent metering systems to enable better management of energy networks and more efficient consumption. However, as suggested in related literature, the previous goals can only be achieved through energy feedback on the appliance level.

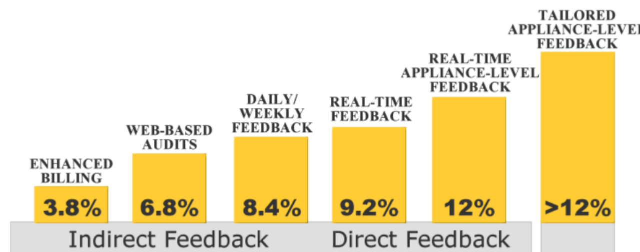

Fig. 1. Effects of feedback on energy savings, see [19].

NILM algorithms are a viable solution to infer the usage of different appliances using a single metering point (i.e., a smart electrical meter). These algorithms generally rely on Machine Learning (ML) approaches to detect and classify ON/OFF events of different appliances. However, since these algorithms require a considerable amount of computational resources, it becomes mandatory to instead rely on cloud services to offer feedback about appliance-level information, which may raise privacy concerns from the consumer's side. Moreover, the currently installed commercial smart meters have constraints on the sampling frequency, with intervals between reported measurements typically 15 minutes, 60 minutes, or 1 day. Nonetheless, NILM algorithms require a sampling frequency of at least 1 Hz as suggested in the list of requirements published by Zeifman [24]. Thus, despite the wide availability of installed smart meters, it still is challenging to benefit from

NILM algorithms to provide energy feedback in the residential sector.

To address these issues, the current manuscript suggests an embedded system with a low-cost measurement unit based on a Raspberry Pi and a customized metering shield, the so-called YoMoPie framework [16]. YoMoPie is meant to be installed as an additional meter inside the household, providing measurements at a level of detail that enables energy feedback at appliance level. The Raspberry Pi stores and processes the measurements and provides information about the disaggregated data via a webserver running locally. Since no data is transferred through the communication network, the suggested system is thus preserving privacy. Since privacy concerns [22] constitute a significant hindrance to the acceptance of smart meters, our system, when extended with a user-friendly interface, would also contribute to the acceptance of smart meter-based services. Furthermore, by offering adequate sampling frequency, it enables the effective application of NILM algorithms . The suggested system is mainly based on a Commercial-off-the-Shelf Raspberry Pi and thus comes at an affordable cost.

The remainder of this manuscript is organised as follows: the section II describes related systems previously suggested, their advantages and drawbacks, the section III describes the suggested system, IV-A presents a testing scenario of the suggested system with real devices, finally section V summarises the main contributions and critically discusses limits of the proposed system.

II. Related Work

A. Measurement devices

The recent surge of research in topics related to energy and sustainability goals shed light on energy measurement hardware. Several surveys [8], [21] have already discussed related contributions. Commercial solutions tend to have fixed frequency measurements [16], which could be an obstacle when using NILM algorithms to provide feedback about appliance-level power consumption. The YoMo [15] and the YoMoPie [16] framework, illustrated in the figure2, are viable alternatives to overcome these issues, where the authors provided YoMo as an Arduino-based smart metering board. This board was further adjusted to be compatible with the Raspberry Pi. Similar systems exist in the literature. However, they suffer from some limitations. For example, [2] is constrained to a sampling frequency of 1 Hz and only has a serial communication interface. A system with a Raspberry Pi was suggested in [14]. Yet, it requires the use of two different devices.

B. NILM algorithms

Non-intrusive Load Monitoring (NILM) algorithms, also called load disaggregation algorithms, are techniques that allow to infer the individual power consumption of different appliances considering only aggregate measurements. Even though George W. Hart has introduced these techniques in the 80's [10], [11], the approach only gained considerable attention from the research community in the past decade due to the upgrade of the electrical grid that brought with it the installation of an advanced metering infrastructure in merely every household.

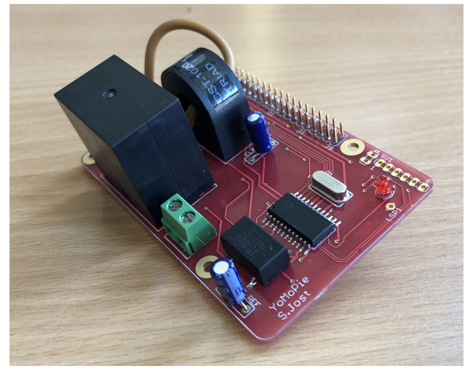

Fig. 2. The YoMoPie Raspberry Pi smart metering extension [16].

Factorial Hidden Markov Models (FHMMs) were among the earliest techniques used to identify appliances' operational states [17], [20], [24] where the observed variable represents a function of the aggregate power and the hidden variables represent the states of appliances. Egarter et al. demonstrate an FHMM approach in combination with Particle Filtering (PF). [6] However, these probabilistic models fail to model all types of appliances. Adopting deep models for load disaggregation [13] was a turning point in the NILM research. The success achieved by these models in estimating the power consumption of different appliances motivated a tremendous number of contributions [9], [12], [18], [23]. These new models have been proven to outperform probabilistic models on different datasets [1]. Nonetheless, this gain in performance can only be achieved with energy measurements of sufficient measurement frequencies $> 1Hz$. However, such measurements contain information about a user's behavior which could lead to privacy problems when the NILM algorithm is implemented in the cloud. A viable solution to the previous issues is a decentralised implementation of the NILM algorithm enabling its execution on edge devices (i.e., the smart meter). However, this decentralised implementation would still require special measurement hardware, providing more frequent measurements, which is often a significant cost factor.

III. Proposed system

The main goal of this work is to demonstrate and evaluate the feasibility of installing and operating a NILM-capable system algorithm on an embedded device. Besides the economic feasibility of NILM at home, the proposed device could enable many research opportunities in future work, such as adopting federated learning in embedded system setups. To achieve this goal, we leverage the YoMo metering device and the YoMoPie framework, which are both previous contributions of one of the authors, to evaluate the local applicability of a supervised NILM algorithm. More precisely, we aim to implement NILM

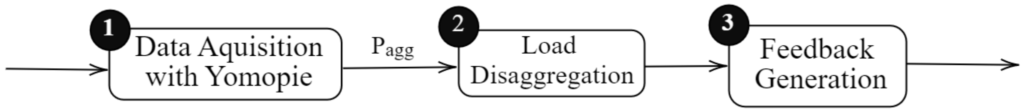

Fig. 3. The main operational steps of the proposed system.

on a Raspberry Pi. In order to measure the current and voltage of a 230V power line, we use the YoMoPie smart metering extension, which is introduced in [16]. YoMoPie measures active as well as apparent power and comes with an easy-to-use Python library[1], which is already included in PyPI and can be installed over the Linux shell. In our experiments, this library had been extended with some features to run our experiments which will be discussed in section IV.

The NILM algorithm is supposed to run on a Raspberry Pi 4 Model B (4 GB RAM version) equipped with the YoMoPie smart metering extension. Following our premise, the measured data will be processed on the metering device, with no additional devices required. The previous feature protects the user's privacy and mitigates the required communication overhead. The proposed system has the following characteristics:

1) **Accuracy**: the YoMoPie smart metering extension's accuracy is measured in the original contribution [16], sufficient for an energy consumption feedback system and even comparable with some commercial smart meters. It yields errors mostly below 3.5%. As our tests will show, the accuracy is sufficient for the considered use cases of NILM.
2) **Efficiency**: the proposed system consumes a reduced amount of energy since it leverages on a Raspberry Pi. In our case, the Raspberry Pi always consumes under 8 watt at any load of the CPU. As our use case does not require a full CPU load, the system's actual wattage is around 3.57 W. The load of the CPU depends on the NILM algorithm used. Furthermore, since the metered data is processed on the smart meter device, it avoids the need for transmitting measurement data, which avoids security problems due to eavesdropping and reduces communication overhead.
3) **Privacy**: the proposed system protects the user's privacy since the data is locally processed and stored. This would improve the acceptance of smart metering applications, which have been challenged with privacy concerns [3], [5].

A. System Implementation

The system consists of three components:

1) data acquisition with a YoMoPie
2) NILM algorithm for data disaggregation
3) feedback on energy consumption

[1] https://github.com/klemenjak/YoMoPie

Figure 3 illustrates the main processing steps of the suggested system implemented on a Raspberry Pi. They consist of three main blocks:

1) **YoMoPie data acquisition**: For acquiring the data, we use the YoMoPie shield on a Raspberry Pi 4 Model B. Figure 2 shows the YoMoPie-board. This board was inspired by YoMo [15] and uses an ADE7754 meter chip from Analog Digitals. The meter chip is ideal for high-accuracy metering with an error of less than 0.1% and low power consumption. The chip communicates with the Raspberry Pi over the SPI. The sampling rate for power measurement is 10 Hz and the data disaggregation is triggered every full second. The acquired data is stored on the Raspberry Pi's local SD card. We use single-phase measurement currently, but the system could easily be extended to three phases as the ADE7754 chip supports three-phase energy measurement. For measuring the power consumption, we used a slightly modified version of YoMoPie Python library. Furthermore, the NILM algorithm operates only on apparent energy.
2) **NILM data disaggregation**: A four-layered feedforward neural network was implemented using Tensorflow/Keras in Python to disaggregate the measured apparent power. The model takes as inputs the changes in power consumption in a moving observation window of 50 samples (5 seconds) and performs classification of the employed devices based on these data. The model is capable of reporting switching on/off events and uses a threshold of 0.8 to minimize false detections. The 'adam' optimizer was employed for model training, which is a commonly used optimizer for deep learning models. This supervised learning approach requires relearning for new devices. The selected model serves as a proof of concept for running learning and disaggregation directly on the Raspberry Pi.
3) **Feedback on energy consumption for the user**: One of the core features of NILM is the feedback for the user. On the one hand, it raises energy awareness and, therefore, it could help to save energy. On the other hand, it saves costs by reducing energy consumption. In our implementation, we provide a detailed log file, which contains timestamps, event details such as switching on/off, and energy consumption over time.

IV. Evaluation of the proposed system

A. Experimental setup

All measurements and the code including all configuration parameters are available in a public repository.[2] The repository also includes the dataset with the measurements for this experiment and the modified code for the YoMoPie library. The repository also includes the code for analyzing the power data. The suggested system was tested under two scenarios. During both scenarios, five different appliances were used: a pedestal fan, vacuum cleaner, water kettle, light bulb, and a hairdryer. The choice of this set of appliances is justified by their presence in approximately every household. We evaluate the system performance for a single appliance recognition during the first scenario. The second scenario evaluates the system for the detection of multiple appliances with possible simultaneous usage.

The implemented NILM algorithm, implemented as a our-layered feedforward neural network relies on two steps. First, the algorithm records one activation for each of the considered appliances in a small database. Second, for every new acquired power value, it will compare the recorded set of activations and attribute the event to most approximate activation minimising the energy error.

During the first scenario, all devices were switched on and off in a random order, such that only one device was running at a specified time. A total set of 38 events were collected during this first set of experiments. On the other hand, the second scenario consisted of switching the devices on and off in a random order, such that multiple devices could be switched on simultaneously.

B. Results

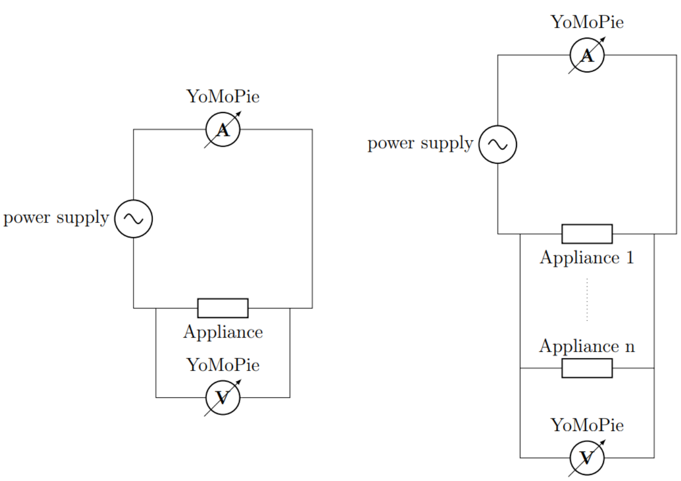

Fig. 4. Test setup.

We measured 5 on and off cycles for each device in the experimental setup and compiled them into a set of training data. These data were trained using 5 different classifiers from the sklearn package in Python. Evaluation was done based on an unseen measured dataset using real devices. The MLPClassifier was found to be unsuitable, as it only achieved a score of 26.67% on the test data. The other 4 classifiers, namely AdaBoost, KNeighbors, RandomForest, and LogisticRegression, achieved highest scores in that order. Table I shows the results of the score, which was achieved after the training.

TABLE I
Comparing the scores of all devices for each classifier. Runtimes are measured on a AMD Ryzen 7 PRO 5850U. Runtimes on as Raspberry Pi are expected to be under 10 seconds.

		Score	Evaluation/Training Time
MLPClassifier	all devices	0.26666	$\leq$ 1 sec
	fan	0.73333	
	hairdryer	1.00000	
KNeighborsClassifier	lamp	0.76666	$\leq$ 1 sec
	vacuum	0.86666	
	kettle	0.86666	
	fan	0.8	
	hairdryer	1.0	
AdaBoostClassifier	lamp	0.9	$\leq$ 1 sec
	vacuum	0.9	
	kettle	0.86666	
	fan	0.73333	
	hairdryer	1.0	
RandomForestClassifier	lamp	0.76666	$\leq$ 1 sec
	vacuum	0.86666	
	kettle	0.86666	
	fan	0.5	
	hairdryer	1.0	
LogisticRegression	lamp	0.76666	$\leq$ 1 sec
	vacuum	0.63333	
	kettle	0.73333	

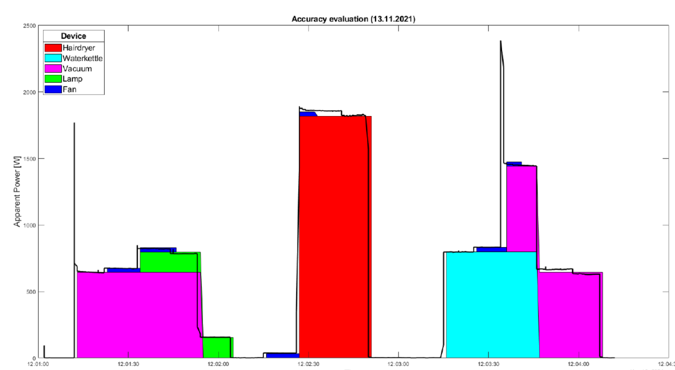

Fig. 5. Results of the second test.

The obtained results show a reasonable classification rate. However, there are various NILM algorithms, some requiring more computation power than a Raspberry Pi 4 can provide. In this case, a possible outlook would be combining multiple Raspberry Pis for parallel computation. The algorithm and the computation power depend on the use case. Costs may vary depending on how many computation units are needed. For our use case, the CPU load increases by approximately 5% when running NILM. The consumption is 700 mA at

[2]https://github.com/smartgrids-aau/NILM_Raspberrypi

5.1 V. Therefore, the power consumption is 3.57 W, which is 31.3 kWh per year.

The system's cost consists of initial costs for the purchase and installment and the system's running costs. The current price of available Raspberry Pis on the market is fluctuating. At the time of the writing, one can expect a price of around 150 Euro for a Raspberry Pi 4 with 4GB RAM as well as the required hardware, consisting of an SD card for the operating system, power supply for the power supply, and mouse or keyboard for the input. A Raspberry Pi can be controlled remotely, so no costs are calculated for a screen. The YoMoPie board cannot be ordered prefabricated, which is why the production of the PCB and the soldering of the components must be carried out in addition to the material cost. The pure material costs, including the costs for the PCB can be estimated at around 50 Euro. In total, the user will incur costs of 200 Euro for a system with a single Raspberry Pi, which would be amortized after 2 years with an average savings potential.

V. Conclusion and Outlook

The present study aims to evaluate the feasibility of implementing a non-intrusive load monitoring (NILM) algorithm on an embedded system with limited computational resources. The obtained results demonstrate promising outcomes with satisfactory disaggregation quality for a set of five appliances. The employed resources on computation, working memory, and storage suggest that implementing measurement device, data storage, NILM processing, and visualization in a single embedded device installed locally is feasible.

Both the software and hardware used in our approach have been published as open-source, allowing researchers to reproduce this paper's results and enabling further research, development, and practical application of the approach by the research community and other third parties.

In future work, we plan to implement a more sophisticated NILM algorithm using machine learning algorithms, which is becoming more attainable with new lightweight ML libraries such as TinyML and Tensorflow lite. Additionally, we aim to evaluate the feasibility of using federated learning on a Raspberry Pi, which would offer a decentralized privacy-preserving NILM algorithm with even better disaggregation accuracy.

References

[1] Nipun Batra, Rithwik Kukunuri, Ayush Pandey, Raktim Malakar, Rajat Kumar, Odysseas Krystalakos, Mingjun Zhong, Paulo Meira, and Oliver Parson. Towards reproducible state-of-the-art energy disaggregation. In *Proceedings of the 6th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation*, pages 193–202, 2019.

[2] Somchai Biansoongnern and Boonyang Plangklang. Nonintrusive load monitoring (nilm) using an artificial neural network in embedded system with low sampling rate. In *2016 13th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*, pages 1–4, 2016.

[3] Dylan Bugden and Richard Stedman. A synthetic view of acceptance and engagement with smart meters in the united states. *Energy Research & Social Science*, 47:137–145, 2019.

[4] K. Carrie Armel, Abhay Gupta, Gireesh Shrimali, and Adrian Albert. Is disaggregation the holy grail of energy efficiency? the case of electricity. *Energy Policy*, 52:213–234, 2013. Special Section: Transition Pathways to a Low Carbon Economy.

[5] Yash Chawla and Anna Kowalska-Pyzalska. Public awareness and consumer acceptance of smart meters among polish social media users. *Energies*, 12(14):2759, 2019.

[6] Dominik Egarter, Venkata Pathuri Bhuvana, and Wilfried Elmenreich. Paldi: Online load disaggregation via particle filtering. *IEEE Transactions on Instrumentation and Measurement*, 64(2):467–477, 2014.

[7] Karen Ehrhardt-Martinez, Kat Donnelly, and John Laitner. Advanced metering initiatives and residential feedback programs: a meta-review for household electricity-saving opportunities. *American Council for an Energy-Efficient Economy*, 01 2010.

[8] Anwar Ul Haq and Hans-Arno Jacobsen. Prospects of appliance-level load monitoring in off-the-shelf energy monitors: A technical review. *Energies*, 11(1):189, 2018.

[9] Alon Harell, Stephen Makonin, and Ivan V Bajić. Wavenilm: A causal neural network for power disaggregation from the complex power signal. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8335–8339. IEEE, 2019.

[10] George W. Hart. Nonintrusive appliance load monitoring. *Proceedings of the IEEE*, 80(12):1870–1891, 1992.

[11] G.W. Hart. Residential energy monitoring and computerized surveillance via utility power flows. *IEEE Technology and Society Magazine*, 8(2):12–16, 1989.

[12] Patrick Huber, Alberto Calatroni, Andreas Rumsch, and Andrew Paice. Review on deep neural networks applied to low-frequency nilm. *Energies*, 14(9):2390, 2021.

[13] Jack Kelly and William Knottenbelt. Neural nilm: Deep neural networks applied to energy disaggregation. In *Proceedings of the 2nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments*, pages 55–64. ACM, 2015.

[14] Sheharyar Khan, Ahmad Farhan Latif, and Sarmad Sohaib. Low-cost real-time non-intrusive appliance identification and controlling through machine learning algorithm. In *2018 International Symposium on Consumer Technologies (ISCT)*, pages 32–36, 2018.

[15] Christoph Klemenjak, Dominik Egarter, and Wilfried Elmenreich. Yomo: the arduino-based smart metering board. *Computer Science - Research and Development*, 31, 07 2015.

[16] Christoph Klemenjak, Stefan Jost, and Wilfried Elmenreich. YoMoPie: A user-oriented energy monitor to enhance energy efficiency in households. In *2018 IEEE Conference on Technologies for Sustainability (SusTech)*, Nov 2018.

[17] J Zico Kolter and Tommi Jaakkola. Approximate inference in additive factorial hmms with application to energy disaggregation. In *Artificial Intelligence and Statistics*, pages 1472–1482, 2012.

[18] Henning Lange, Mario Bergés, and Zico Kolter. Neural variational identification and filtering for stochastic non-linear dynamical systems with application to non-intrusive load monitoring. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8340–8344. IEEE, 2019.

[19] Andrea Monacchi, Wilfried Elmenreich, Salvatore D'Alessandro, and Andrea M. Tonello. Strategies for domestic energy conservation in Carinthia and Friuli-Venezia Giulia. In *39th Annual Conf. of the IEEE Industrial Electronics Society (IECON'13)*, pages 4791–4796, 2013.

[20] Sundeep Pattem. Unsupervised disaggregation for non-intrusive load monitoring. In *2012 11th International Conference on Machine Learning and Applications*, volume 2, pages 515–520. IEEE, 2012.

[21] James Pierce and Eric Paulos. Beyond energy monitors: interaction, energy, and emerging energy systems. In *Proceedings of the SIGCHI Conf. on Human Factors in Computing Systems*, pages 665–674, 2012.

[22] Pascal Schirmer, Iosif Mporas, and Akbar Sheikh Akbari. Identification of tv channel watching from smart meter data using energy disaggregation channel watching from smart meter data using energy disaggregation. *Energies*, 14:2485, 04 2021.

[23] Changho Shin, Sunghwan Joo, Jaeryun Yim, Hyoseop Lee, Taesup Moon, and Wonjong Rhee. Subtask gated networks for non-intrusive load monitoring. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 1150–1157, 2019.

[24] Michael Zeifman and Kurt Roth. Disaggregation of home energy display data using probabilistic approach. In *2012 IEEE International Conference on Consumer Electronics (ICCE)*, pages 630–631. IEEE, 2012.