Journal of Technology and Exploitation in Mechanical Engineering
Vol. 3, no. 1, pp. 34–42, 2017
https://doi.org/10.35784/jteme.537

ISSN 2451-148X
Available online at: http://jteme.pl

Research article

TILT-BASED PREDICTIVE TEXT INPUT CONCEPT FOR MOBILE DEVICES

Marcin Badurowicz[1], *Stanisław Skulimowski*[2], *Anna Prończuk-Omiotek*[3]

[1] Lublin University of Technology, Institute of Computer Science, 38A Nadbystrzycka Str. 20-618 Lublin, Poland, orcid.org/0000-0003-2249-4219, email: m.badurowicz@pollub.pl
[2] Lublin University of Technology, Institute of Computer Science, 38A Nadbystrzycka Str. 20-618 Lublin, Poland, email: s.skulimowski@pollub.pl
[3] Maria Curie-Sklodowska University in Lublin, Faculty of Economics, Maria Curie-Sklodowska Square 5, 20-031 Lublin, Poland, email: an.pronczuk@gmail.com

Submitted: 2017-06-09 / Accepted: 2017-06-22 / Published: 2017-06-30

ABSTRACT
In the paper authors are introducing the concept of usage of physical orientation of a mobile device, calculated using built-in environmental sensors like accelerometer, gyroscope and magnetometer for detection of tilting gesture. This gesture is used as an acceptance factor for the two next probable word solutions suggested to the user during text input. By performing the device tilt, the first or second word is being automatically put into the desired text field and new prediction is performed. The text predictions are calculated and stored directly on the device to maintain privacy protection.
The founding concept of the software is being presented, as well as initial considerations and further plans. This solution is recommended especially to smartphone manufacturers like Microsoft, Samsung and Apple to deploy in their latest models.
KEYWORDS: mobile devices, human-computer interaction, accelerometer, text input, technology

KONCEPCJA WPROWADZANIA TEKSTU PRZEWIDYWANEGO POPRZEZ WYCHYLANIE URZĄDZENIA MOBILNEGO

STRESZCZENIE
W artykule zaprezentowano koncepcję wykorzystania fizycznej orientacji urządzenia mobilnego, wyliczanej z wykorzystaniem wbudowanych czujników takich jak akcelerometr, żyroskop oraz kompas cyfrowy w celu wykrywania gestów wychylenia urządzenia. Taki gest wykorzystywany jest jako metoda akceptacji jednego z dwóch następnych prawdopodobnych słów podczas wprowadzania tekstu. Poprzez wykonanie gestu proponowane słowo jest automatycznie akceptowane i użytkownikowi prezentowana jest kolejna predykcja. Prawdopodobieństwa są wyliczane i przechowywane bezpośrednio na urządzeniu, w celu zapewnienia prywatności użytkownika.
Podstawowa koncepcja oprogramowania jest zaprezentowana w artykule, jak również wstępne założenia i plany na przyszłość. Rozwiązanie jest szczególnie rekomendowane dla producentów smartfonów, jak firmy Microsoft, Samsung czy Apple, do wdrożenia w najnowszych modelach smartfonów.
Słowa kluczowe: urządzenia mobilne, interakcja człowiek-komputer, akcelerometry, wprowadzanie tekstu, nowe technologie

1. Introduction

The mobile devices, currently on the peak of their popularity, need various text entry mechanisms, different to the desktop machines, modified to be run on the go and in other changing environments. Some of the modern devices are still equipped with physical keyboards (e.g. Blackberry phones) and physical keypads (e.g. Nokia "dumb" phones), while all commonly used handheld devices with a touch screen allow to enter text via the onscreen/touch keyboard easily (e.g. iPhones, iPads, ordinary Android phones, Windows Phone based mobile devices).

Both buttons appearance (starting from physical keypad and keyboard as well as its modifications, later the T9 dictionary, and ending with touch QWERTY-style keyboards) and input techniques (from simple physical pressing, onscreen finger typing, to advance gesture typing, word and phrase prediction or contextual keyboard layouts) were improved to achieve better text input rate (Words Per Minute -- WPM and Characters Per Minute -- CPM) and the error rate reduction (misspelling, miswriting).

The need for increased WPM and CPM rates results from growing number of active, communicating users. Over 75% of people in Poland have at least one smartphone [1], however 77% of Poles state willingness to buy new device before the end of the year. The main reasons to buy new model of the smartphone are its improved functionality (45%) and new features (51%) [2].

The desire to conduct more conversations translates into reducing the time available for a single response. One of the indirect results of this phenomenon was creation of modern language abbreviations and acronyms (e.g. ASAP -- as soon as possible, LOL – laughing out loud, OMG – oh my god, CTN – cannot talk now, NC -- no comment) which allow us to pass complex content in a simple and fast way.

The main objective of this paper is to introduce an innovative method for enhancing the text input rate on mobile devices by usage of physical device orientation as a selection factor. While there are several similar concepts already discussed in the literature, the proposed concept is simpler and more intuitive for the users already composing texts on mobile devices. The proposed solution is also purely software based, to be implemented in already existing mobile devices as an only small update, without a need to design new or modify existing hardware.

The authors aim to use tilting the device as a method for selection of predictive text input results, to reduce the number of finger's (thumb) movement to the upper keyboard bar, where predictions for the next possible words are used. Tilt, while several software keyboards were developed already, is not used in any currently available commercial product.

2. Similar research and ideas

Tilting the device, and more generally speaking – orientation of the device – has already been discussed in the human-computer interaction literature, for more than 10 years. Basic implementations provided application of tilt for scrolling or navigation. The text input was discussed in several other research systems: Unigesture [3], TiltType [4], TiltText [5] and more [6] [7] [8]. They all have some flaws: Unigesture, requires specifically developed hardware, the TiltType was designed for very small devices (like smartwatches) and the TiltText was meant to be better than traditional multi-tap keyboards, which are not that popular anymore. On the other hand, YAUIM was designed not only for mobile devices, but for a wider range of different input handlers.

There is also more detailed analysis of usage only of tilt and environmental data, for example, WalkType [9] to improve typing on a standard QWERTY layout, by compensating movement due to walking. GesText [6] for example, represents a text entry design solely built on tilt input, with a prototype built using the Nintendo Wii hardware, and KeyTilt [7], where the user efficiency was however lower than using stylus and AZERTY keyboard. There are also techniques discarding traditional keyboard layouts in favour of usage of gestures [10] [11]. The highly-popular keyboard SwiftKey for Android even recreated its titular “Swipe” method by usage of tilting, but only for means of April Fool’s Day. The smartphone applications using tilt for different actions are also widely popular, especially in mobile gaming industry [12].

3. Our proposal

The authors propose usage of tilt only as an enhancement to the regular input provided. Presented in previous sections various innovations to the well-known input methods have great value, however the users are already familiar with a set of techniques; therefore, an implementation of something significantly new should overcome users' stubbornness. Enhancement of already existing techniques is a crucial way of implementing new features without users’ objections.

The predictive text methods, based on different statistical techniques, have a great value for users, but the key issue is that user must change the thumb position drastically, to achieve a tap gesture on a proposed word. While the average smartphone screen size is rising in size constantly, the available so-called "thumb zone" [13] without repositioning is going to be smaller and smaller.

To cope with this, the authors suggest displaying only two most probable next words after the current word and usage of tilting the device as a method of selecting and accepting them.

3.1. Tilt detection and threshold values

The tilt of the device may be calculated using accelerometer, which is now one of the standard environmental sensors available in the mobile devices, however, the authors recommend the usage of gyroscope as well as “sensor fusion” techniques for the better discovery of tilting gesture. By combining the results from multiple sensors, it is possible to create an artificial inclinometer device, allowing to calculate roll, pitch and yaw values, relative to the device screen. As presented in the fig. 1 below, blue arrow is a roll value, green defines pitch vector and red yaw axis.

Fig. 1. Roll, pitch and yaw on the device

The most useful value for detecting the tilt of the device is the roll angle. However, its usage is constrained by the device orientation. It is possible to use the roll value only in vertical (portrait) orientation of the device, neither the horizontal one, nor the flipped portrait, as it is always calculated with an upper part of the device in mind. To calculate the horizontal (landscape) orientation and its flipped versions, accelerometer is used to detect device orientation and then only simple change between roll, pitch and yaw values are being used, as well as negation of the reported values.

When two words are shown as a next probable option after the current word, which was entered using any currently popular method, including QWERTY tap and type or the swipe, tilting left will add the first one, and tilt right the second one. The choice of this gesture as an acceptance method is a key concept of the presented solution.

Since the accepting gesture should be intuitive and fast, the authors conducted basic experiments to evaluate the possibility of using it. Using the developed software, acquiring roll, pitch and yaw values every 0.015 of a second, by performing similar data acquisition and postprocessing procedure as already established in another research [14], the accepting gesture was experimentally validated, using the software prototype. In the fig. 2 the prototype software (version 1) during the acquisition phase of the inclinometer values is presented.

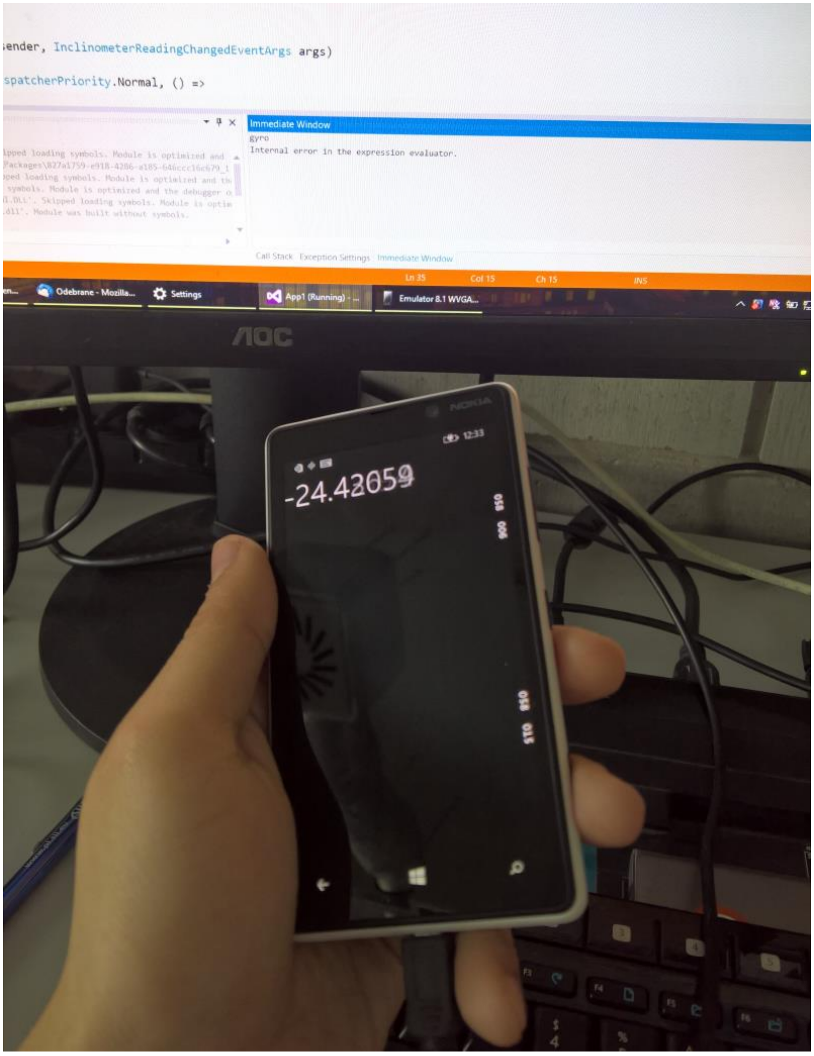

Fig. 2. The prototype acquisition software at work

In the fig. 3 below, there is a representation of values of roll angles acquired from series of tilting left and right the device in the manner which should be considered as an acceptance gesture in the portrait device orientation.

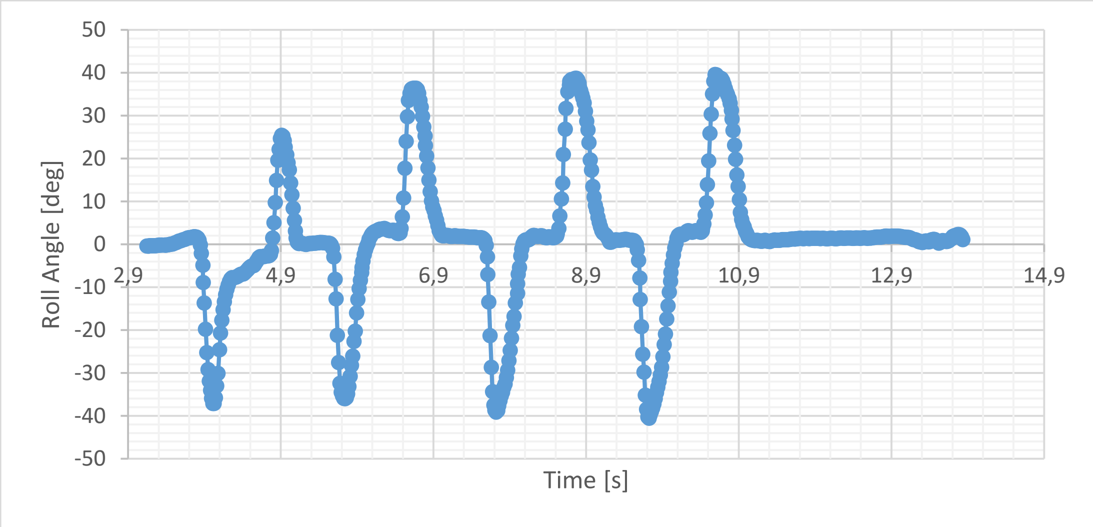

Fig. 3. Roll angles in time while tilting the device

In the fig. 4, there is a series of another tilting left and right gestures, but based on landscape left (horizontal) orientation, thus using roll angles.

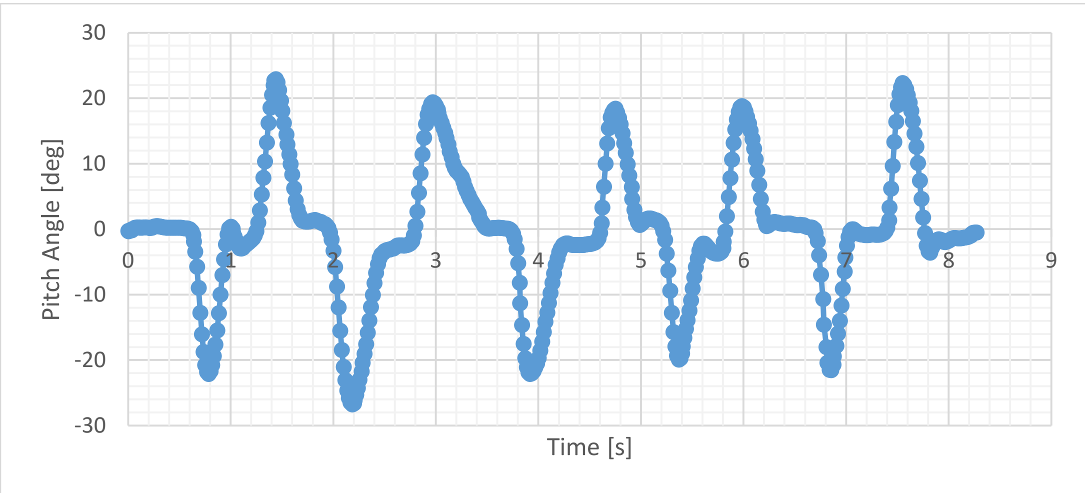

Fig. 4. Pitch angles in time while tilting the device in horizontal orientation

Based on the results obtained, the authors concluded that the acceptance gesture should be defined as a tilt resulting in:

- absolute values of a roll angle changing from the more than 5 degrees up to more than 25 degrees in the time of less than 500 milliseconds; for the portrait device orientation;
- absolute values of a pitch angle changing from the more than 5 degrees up to more than 17 degrees in the time less than 500 milliseconds, for the landscape device orientation;

Examples of acceptance gesture are presented in the fig. 4 and 5 below:

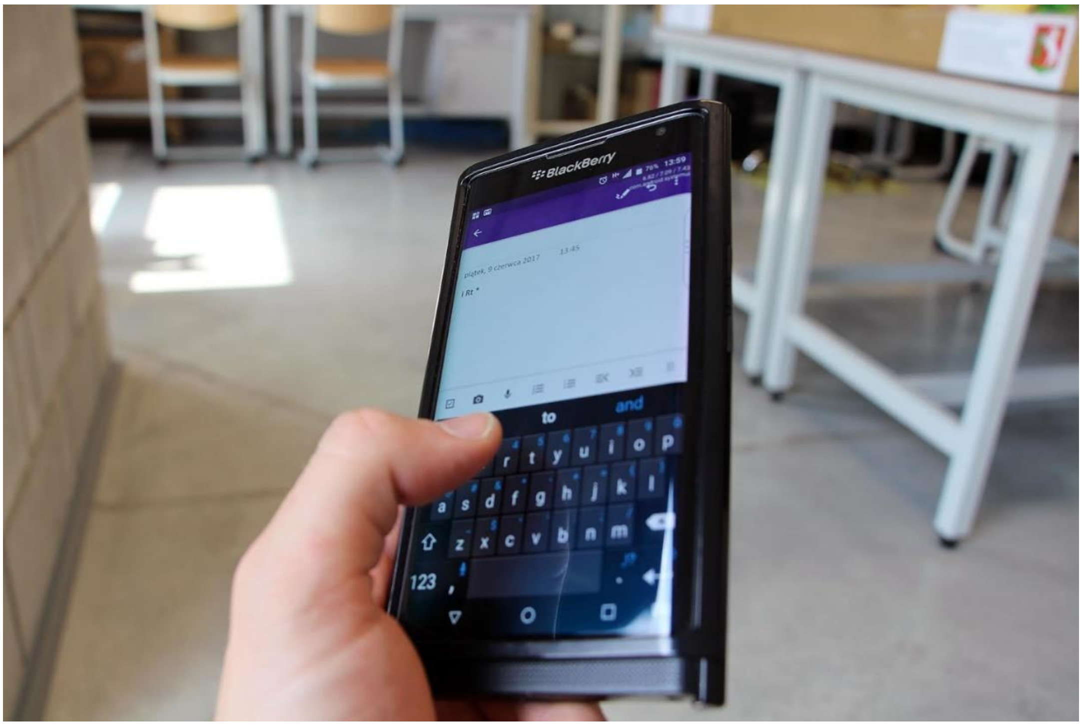

Fig. 4. Tilting the device left as an acceptance gesture in portrait orientation

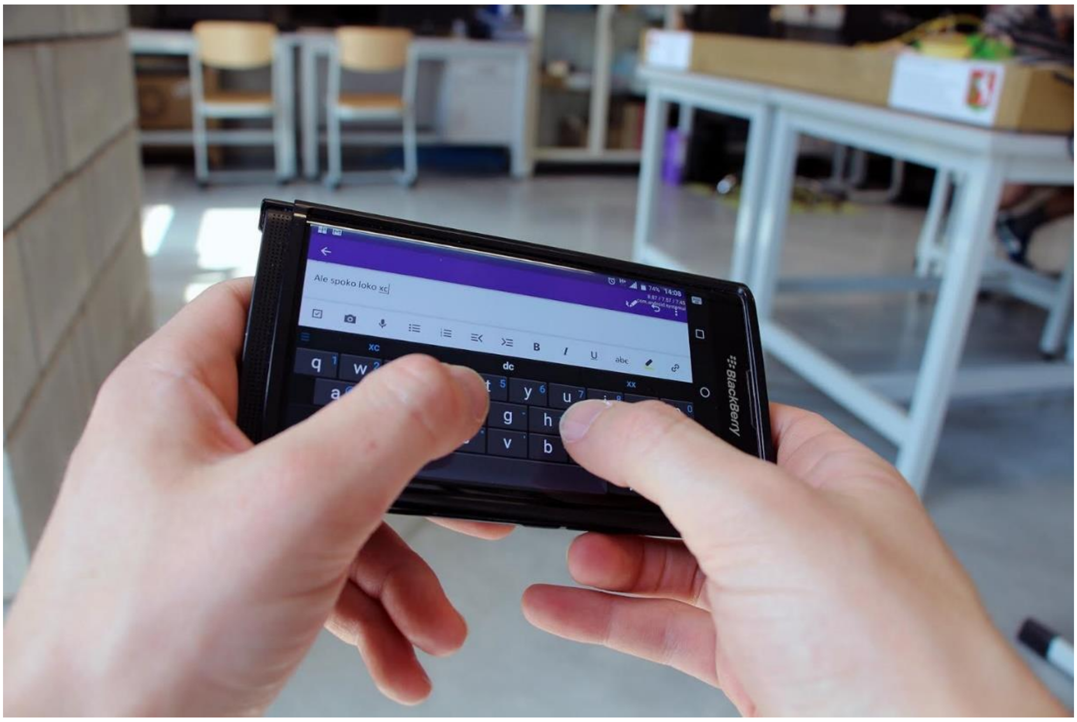

Fig. 5. Tilting the device right – two-handed landscape mode

These preliminary boundaries will be modified after user experience testing planned soon, but they looked reasonable for the first user experience prototype phase the authors was performing. There may be also a need in the future to change accepting gestures based on the user's dominating hand, which was not taken into consideration at this moment.

The algorithm for the gesture detection is a simple thresholding, looking for the values over the defined threshold and calculating the time between those thresholds, which gives enough results, but the authors are also willing to test another gesture detection algorithms in the future. The element which is not included now, but should also be examined is a power requirement of the gyroscope and accelerometer – when these devices are being used multiple times per second, there is a need to check if it impacts negatively overall battery life of the device. Both the algorithm and the sensor sampling frequency need to be carefully tuned to achieve best results while minimising energy consumption.

3.2. Text prediction algorithm

On the other hand, there is an issue of what the next most probable word would be predicted to discuss as a second major problem to solve. The statistical text methods, based on the literature, are not applicable in the case of text messaging, abounding in acronyms and simplifications. The authors suggest the parsing of user's own text messages for initial keyboard configuration, during the installation phase of software, for the best prediction values for the current user and his or her writing style. The algorithm is defined as presented in the fig. 6, starting with dividing the text message sentence by sentence:

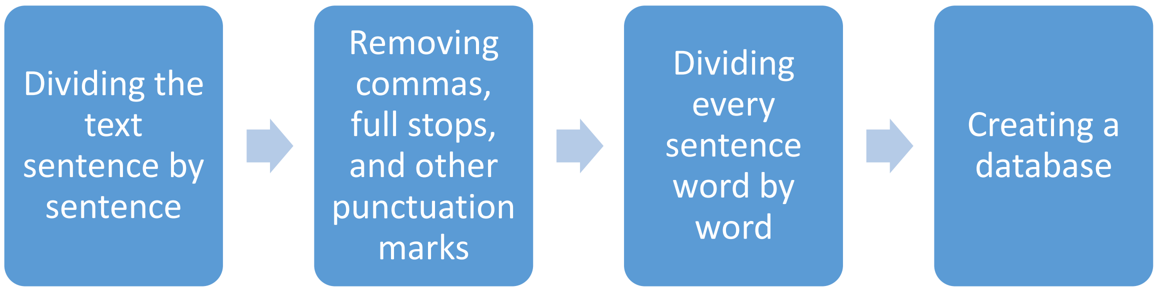

Fig. 6. Text prediction database building process

The algorithm performs the following steps after sentence by sentence division:

1) Removes commas, full stops and other punctuation marks from the input texts,
2) Divides the sentences word by word,
3) Takes the first word and saves it as a key of multi value dictionary,
4) Takes the second word and saves it as the first value of this dictionary in a form of tuple: the word, and number of usages equals to 1,
5) Moves by one word.

If the word already exists in the dictionary as the key, the following word is added as a next value, again the form of tuple. If the pair already exists in the dictionary, the number of usages is updated. The example result of the algorithm is presented in the table 1, based on the content of this paper:

Table 1. Results of algorithm run on the text of this paper

Word	Usage	Next word	Usage	Next popular words			
the	219	device	22	authors	first	user	text
of	89	the	23	titling	a	usage	physical
and	63	the	5	yaw	more	new	its
to	57	be	9	the	achieve	buy	use
in	51	the	29	a	any	time	lublin
a	46	need	4	key	simple	method	wider
is	42	a	10	also	used	being	the
for	40	the	14	example	a	mobile	users
as	30	a	11	an	well	presented	acceptance
device	27	orientation	27	in	as	right	calculated

The words “the”, “of” and another in the first column were the most used, the second column presents the most common word following the word in the first column with the number of usages, while the columns 5-8 present another popular following word, ordered by the number of occurrence, descending.

After preparing the initial database, the predictive text algorithm will provide the user with the selection of two most popular (with the highest number of usages) words after selecting the first word, where the first will be accepted by tilting left and the second by tilting the device right. Basing on the table 1, it is possible to observe that “the device” is the most common two-word phrase in this article, so the algorithm will try to propose “device” after “the”, along with “authors” as a second alternative, the second in popularity. Of course, in the case of scanning the users’ whole history of conversations, the predictions will be more accurate, as the results presented in the table 1 are very specific to this text and used only for algorithm presentation.

In the alternative approach, the Markov chains are also taken into consideration, as a tool for generation of predictive text. This is one of the future solutions which will be tested in the next phase of developing the prototype.

Scanning and analysing text on the device are serious issues in the current world. Because of the proposed text-scanning initial configuration multiple private data may be exposed. Existing keyboard software is sending statistical information to the cloud, which may be not desirable for the user. To help users’ protecting their privacy, the authors do not provide telemetry mechanisms in the proposed solution. The cloud synchronisation or backup option may be provided, but should not be forced in any way. When the user accepts the proposed solution, or puts any new following word by tapping it in, the database and its statistics are to be updated.

4. Conclusion and future works

The authors are now preparing the prototype version of the software to be used on the Android and Windows platforms, for performing the more thorough tests, including calculating the WPM and TER (Total Error Rate) statistical values to compare the presented concept with another solution already available.

There is also a need of performing tests in the real-world situations – during walking, or in the bus, again for comparison between the text input methods available currently. Some changes in the initial concept may also be tested, for example using yaw angle instead of roll to perform acceptance gesture, but this case needs a wider test group. Tests could also be conducted to analyse the impact on typing accessibility for users with motor disabilities or eye defects.

As all new inventions, this solution has disadvantages and could have negative influence, for example on health. There are known cases of eye diseases caused by staring at electronic devices screen for too long, as well as some instance of carpal tunnel syndrome. The solution presented in this article could cause similar wrist disease or arthritis of the joints. The second disadvantage for this moment relates to the risk of damages of the device. Making tilt with the wrong grip could cause drop and breakage. Such as every new skill, users should learn how to operate this properly.

The authors believe the usage of physical device orientation as a deciding factor for the software is one of the innovations which should be included in the next generation of mobile devices to achieve faster writing times – which is the key factor in the move to mobile first, cloud first world.

5. References

[1] "Report Dynamic Digital Consumers by Accenture, 2016," [Online]. Available: https://www.accenture.com/pl-pl/insight-dynamic-consumers. [Accessed 10 June 2017].

[2] "Report Dynamic Digital Consumers by Accenture, 2017," [Online]. Available: https://www.accenture.com/pl-pl/insight-dynamic-consumers. [Accessed 10 June 2017].

[3] V. Sazawal, R. Want and G. Borriello, "The Unigesture Approach," in *Proceedings of the 4th International Symposium on Mobile Human-Computer Interaction*, London, 2002.

[4] K. Partridge, S. Chatterjee, V. Sazawal, G. Borriello and R. Want, TiltType: Accelerometer-Supported Text Entry for Very Small Devices. In Proc. UIST 2002 (pp. 201-204), New, York: ACM, 2002.

[5] D. Wigdor and R. Balakrishnan, "TiltText: Using Tilt for Text Input to Mobile Phones," in *Proceedings of the 16th Annual ACM Symposium on User Interface Software and Technology*, New York, NY, USA, 2003.

[6] E. Jones, J. Alexander, A. Andreou, P. Irani and S. Subramanian, GesText: Accelerometer-Based Gestural Text-Entry Systems. In Proc. CHI 2010 (pp. 2173-2182), New, York: ACM, 2010.

[7] G. Homberg, A. Sangott and B. Martin, "KeyTilt: An inclined software keyboard [KeyTilt : Un clavier logiciel par inclinaison]," 2010.

[8] C. Gao, R. Pastel and J. Tan, "Yet another user input method: Accelerometer assisted single key input," in *2010 8th World Congress on Intelligent Control and Automation*, 2010.

[9] M. Goel, L. Findlater and J. Wobbrock, "WalkType: Using Accelerometer Data to Accomodate Situational Impairments in Mobile Touch Screen Text Entry," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, New York, NY, USA, 2012.

[10] S. Ljubic, M. Kukec and V. Glavinic, "Tilt-Based Support for Multimodal Text Entry on Touchscreen Smartphones: Using Pitch and Roll," in *Universal Access in Human-Computer Interaction. Applications and Services for Quality of Life: 7th International Conference, UAHCI 2013, Held as Part of HCI International 2013, Las Vegas, NV, USA, July 21-26, 2013, Proceedings, Part III*, C. Stephanidis and M. Antona, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 651-660.

[11] S. Ljubic, V. Glavinic and M. Kukec, "Predicting upper-bound text entry speeds for discrete-tilt-based input on smartphones," *Journal of Interaction Science,* vol. 2, p. 3, 2014.

[12] Y. Du, H. Ren, G. Pan and S. Li, "Tilt & Touch: Mobile Phone for 3D Interaction," in *Proceedings of the 13th International Conference on Ubiquitous Computing*, New York, NY, USA, 2011.

[13] "Designing for Thumbs – The Thumb Zone," [Online]. Available: http://blog.usabilla.com/designing-thumbs-thumb-zone/. [Accessed 10 June 2017].

[14] M. Badurowicz, T. Cieplak and J. Montusiewicz, "The Cloud Computing Stream Analysis System for Road Artefacts Detection," in *Computer Networks: 23rd International Conference, CN 2016, Brunów, Poland, June 14-17, 2016, Proceedings*, P. Gaj, A. Kwiecień and P. Stera, Eds., Cham, : Springer International Publishing, 2016, pp. 360-369.

[15] L. Findlater and J. O. Wobbrock, "From plastic to pixels: in search of touch-typing touchscreen keyboards," *ACM Interactions,* vol. 19, 2012.