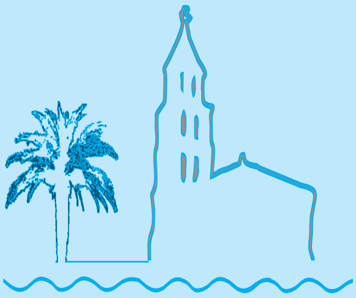
SoftCOM 2019 events

SoftCOM 2019 PhD Forum

Book of Abstracts

Split, Croatia
September 19 - 21



SoftCOM 2019 PhD Forum

Book of Abstracts

27th International Conference on Software, Telecommunications and Computer Networks

SoftCOM 2019

Split, Croatia
September 19 – 21, 2019

SoftCOM 2019 PhD Forum

SoftCOM 2019
Split, Croatia
September 19 – 21, 2019

Steering committee:

Dinko Begušić
University of Split, Faculty of Electrical Engineering, Mechanical Engineering and Naval Architecture

Tihana Galinac Grbac
University of Rijeka, Faculty of Engineering

Darko Huljenić
Ericsson Nikola Tesla

Maja Matijašević (chair)
University of Zagreb, Faculty of Electrical Engineering and Computing

Drago Žagar
Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology

Program & Organizing Committee:

Maja Škiljo (chair)
University of Split, Faculty of Electrical Engineering, Mechanical Engineering and Naval Architecture

Andrej Grgurić
Ericsson Nikola Tesla

Goran Mauša
University of Rijeka, Faculty of Engineering

Višnja Križanović
Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology

Mirko Sužnjević
University of Zagreb, Faculty of Electrical Engineering and Computing

***SoftCOM** library*

2019 27th International Conference on Software, Telecommunications and Computer Networks (SoftCOM 2019)
Technically cosponsored by IEEE Communications Society

Edited by: Dinko Begušić, Maja Škiljo, Maja Matijašević, Mirko Sužnjević

ISBN: 978-953-290-096-5

FESB, University of Split
R. Boškovića 32
21000 Split
Croatia
Tel. +385 21 305 795
Fax: +385 21 305 655
Email: softcom@fesb.hr
www.fesb.hr/SoftCOM

Editorial assistant: Katarina Radoš
Printed by Galešnik, Split

CONTENTS

SoftCOM 2019 PhD Forum

Foreword

The PhD Forum, hosted by the 27th International Conference on Software, Telecommunications and Computer Networks – *SoftCOM 2019*, in Split, Croatia, is an event dedicated to PhD students. On behalf of the steering committee, it is my pleasure and honor to write this foreword to the PhD Forum's Book of Abstracts.

Continuing the practice of co-locating the PhD Forum with IEEE technically co-sponsored conferences, this year's event takes place in the beautiful city of Split, marking the fourth anniversary of its inaugural event at *SoftCOM 2016*. In the preceeding two years, the PhD Forum took place at the 2nd International Conference on Smart Systems and Technologies (SST) in 2018 in Osijek, and at the 14th International Conference on Telecommunications – *ConTEL 2017*, in Zagreb, Croatia.

To be included in the SoftCOM 2019 PhD Forum programme, doctoral students were invited to submit a two-page (extended) abstract for review. The submissions were reviewed by the PhD Forum Program & Organizing Committee members, based on relevance to the conference, innovativeness, and quality of (written) presentation. A total of 9 submissions were finally accepted, and the final versions are included in this book.

The PhD Forum programme was organized as a poster session, preceded by a set of fast-paced introductory "pitch talks", offering a preview of the posters. The purpose of a pitch talk was to provide a brief outline of one's doctoral research work, with the goal to "set the stage" for further discussion over the upcoming poster session. Each student was given a strictly-timed 2-minutes' slot to present. Photographs at the end of this book capture some notable moments from the pitch talk session, as well as the discussions regarding the posters. The winner of the best presentation award was determined by the members of the audience in a secret ballot vote. The winner was Sara Vlahović, a doctoral student at the University of Zagreb.

I would like to thank the SoftCOM 2019 General Chair, Dinko Begušić, and all the members of the Steering Committee, as well as the members of the Program & Organizing Committee, for great support and the job well done.

Maja Matijašević, University of Zagreb
Steering Committee Chair

The Wheel Spinning Behavior in Intelligent Tutoring Systems: An Overview of the Approaches

Ines Šarić-Grgić
Faculty of Science, University of Split
Split, Croatia
ines.saric@pmfst.hr

Ani Grubišić
Faculty of Science, University of Split
Split, Croatia
ani.grubisic@pmfst.hr

***Abstract*—The wheel spinning phenomenon in intelligent tutoring systems occurs when a student spends a considerable amount of time practicing a skill but shows little or no progress towards mastery. Since its introduction in 2013, researchers have proposed several wheel spinning approaches that differ in term of the examined systems, datasets, features, techniques, as well as the model performance metrics. This paper presents an overview of existing approaches according to the previous characteristics.**

Keywords—intelligent tutoring systems, student modeling, wheel spinning.

I. Introduction

Intelligent tutoring systems (ITS) help students to learn in an individualized manner. These systems teach students specific skills while adapting their teaching and testing processes to the knowledge of a student. In that sense, student models in ITS estimate student knowledge at the given time and potentially predict whether a student will correctly perform a specific task in the future. Widely used student modeling approaches range from the simplest overlay model to the probability-based Bayesian Knowledge Tracing (BKT) and latent factors models, such as the Additive Factors Model (AFM) and Performance Factors Analysis (PFA).

Due to its well-established application in the tutoring process, most of ITS follow a mastery learning instructional strategy in which students must achieve a level of mastery in prerequisite knowledge before moving forward to learn subsequent knowledge [1]. Since these students continue to practice a skill until they achieve mastery, the strategy of mastery learning is based on the premise that students will eventually learn everything that is taught if given enough time. Besides students who will eventually master a skill with enough practice, research has shown there are students who do not learn in an assumed way. Struggling students can spend a considerable amount of time stuck in the mastery learning loop without significant learning taking place. This important drawback of mastery learning was first reported by Beck and Gong [2], who introduced the Wheel Spinning (WS) phenomenon in ITS.

Since its introduction, several WS approaches for ITS have been introduced. In this paper, we give an overview of the existing models according to the selected characteristics. In the next section, we describe the methodology, while in the results section we compare the WS models. The conclusions and future research directions are presented in the last section.

II. Methodology

Referring to a moment when a student has spent a considerable amount of time practicing a skill but shows little or no progress towards mastery, the WS phenomenon has proven to be a common problem in different ITS [2]. In order to give an overview of existing WS models, the relevant scientific databases (Web of Science and Scopus) and other resources were searched. The WS research includes keywords such as intelligent tutoring systems, student modeling (student modelling) and wheel spinning (wheel-spinning). The existing models examined different systems and domains, features, techniques, and model performance metrics.

III. Results

Based on the background research, there are several research studies that deal with high unproductive student persistence defined as a WS in ITS. In the following text, there is a brief description of each model specificity, while the selected characteristics are presented in Tables 1 and 2.

While introducing WS phenomenon, Beck and Gong [2] suggested a mastery policy as a three correct consecutive student attempts within the first 10 or 15 practice opportunities ('three in a row' policy). The main focus of Beck and Gong's approach [3] was to build a generic model with a set of features that are computable based on data commonly stored by most ITS. This model contained the total precision metrics above 71%, while the recall metrics were slightly above 52%, indicating that the model overlooked nearly half of the WS cases.

In the work by Matsuda et al. [4], the idea was to improve the simplicity and scalability of the model. Instead of various learning features, Matsuda et al. used only binary response sequences with correct or incorrect answers as input for the WS model. In this study, teachers were used as human coders for WS detection. The precision metric of the approach was about 25%, with the recall metric above 70% at the 10^{th} opportunity.

In addition to the heuristically defined 'three in a row' mastery policy, Käser et al. [5] proposed the approach based on the Predictive Stability (PS) policy [6] that stops when the probability of a student getting the next task correct stabilizes (e.g. BKT). Käser et al. introduced the Predictive Stability (PS++) policy, which provides further analysis about the mastery after the PS policy would have stopped. The synthetic data experiments using the PS++ policy and BKT showed that the policy consistently identified students unable to achieve mastery of a skill (WS students). A limitation of the PS++ policy was that it did not work for the AFM and PFA.

The main contribution of the WS models presented in the study by Sharada et al. [7] was the environment of the enhanced mastery cycle that involves periodic student retesting of the mastered skills, which helps students relearn forgotten skills and thereby enhance long-term retention. In general, the Deep Learning and Logistic Regression models were outperformed by the ensemble Random Forests that achieved AUC metric of 87% (RMSE of 26%) for detecting WS in any of the four retention tests.

With an aim of contributing to the Generalized Intelligent Framework for Tutoring (GIFT), Park and Matsuda [8] investigated the WS in the context of adaptive online courseware, where many ITS are used for the formative assessment. The precision metric of the model resulted in 87%, while the recall metric achieved 75% across the 3rd through 9th opportunity.

While the previous studies distinguished WS students from successfully persistent students, Kai et al. [9] distinguished WS students from the students who failed to master the skill (presumably for other reasons). The researchers revealed 2 types of WS students and described the model as reasonably effective in predicting whether a student will engage in WS, achieving an AUC metric of 68%.

In the recent research study, Botelho et al. [10] observed the low persistence, defined as student "stopout" and the unproductive high persistence, defined as the WS. In term of a prediction whether a student is going to wheel spin in the current assignment, the Deep Learning model performed the best with an AUC metric of 88% (RMSE of 31%).

The very last research by Zhang et al. [11] is the only study that compared two mastery learning instructional policies for WS, 'three in a row' and probability-based policy (PS++). The results showed that two WS approaches diverged substantially and that a Random Forest model had the most consistent performance in the early WS detection. At the fourth step, the previous model achieved 77% and 63% of precision and recall for the 'three in a row' policy, and 90% and 81% of precision and recall for the PS++ policy.

TABLE I. SYSTEM(S) AND DOMAIN(S)

Research study	*System(s)*	*Domain(s)*
Gong, Beck (2015)	Cognitive Algebra Tutor (CAT), ASSISTments Tutor	Algebra skills, Math skills
Matsuda et al (2016)	Geometry Cognitive Tutor	Geometry skills
Kaser et al (2016)	Calcularis, Andes2	Basic numerical skills, Subtraction skills, Physics skills
Sharada et al (2017)	ASSISTments Tutor (PASS module)	Math skills
Park and Matsuda (2018)	Geometry Cognitive Tutor	Geometry skills
Kai et al (2018)	ASSISTments Tutor (Skill builder)	Math skills
Botelho et al (2019)	ASSISTments Tutor	Math skills
Zhang et al (2019)	MATHia, Geometry Tutor	Algebra skills, Geometry skills

TABLE II. NUMBER OF FEATURES (#F) AND TECHNIQUE(S)

Research study	*#F*	*Technique(s)*
Gong, Beck (2015)	15	Logistic Regression; 'three in a row' policy
Matsuda et al (2016)	1	BKT; Deep Learning; human coders as WS detectors
Kaser et al (2016)	1	BKT (AFM, PFA), PS++ policy
Sharada et al (2017)	11	Deep Learning, Random Forest, Logistic Regression; 'three in a row' policy
Park and Matsuda (2018)	4	Logistic Regression, Decision Tree; 'three in a row' policy
Kai et al (2018)	25	Decision Tree; 'three in a row' policy
Botelho et al (2019)	15	Deep Learning, Decision Tree, Logistic Regression; 'three in a row' policy
Zhang et al (2019)	28	BKT; Logistic Regression, Random Forest, Deep Learning; 'three in a row' and PS++ policy

IV. CONCLUSION

In this paper, we have summarized the WS approaches since the introduction of the phenomenon in 2013. The WS behavior was examined across seven ITS used in the mathematics and physics domains. The proposed models included from 1 to 28 features that were used as input for various traditional and machine learning techniques for detection and prediction of WS. Compared to the more complex models, the simplest model with a single feature of binary student response was found to perform well in predicting WS at an early stage. Besides the specific ITS environment manifested in the used features, the WS approach was closely related to the applied mastery learning instructional policy. There is only one study that compared 'three in a row' and PS++ policy and found that the WS approaches agreed to a lesser extent than the researchers expected. Therefore, it would be important to distinguish the WS behavior from the other types of behavior (e.g. gaming the system) and to further investigate the WS across different ITS and knowledge domains.

ACKNOWLEDGMENT

This paper is part of the Adaptive Courseware & Natural Language Tutor (AC&NL Tutor) project that is supported by the Office of Naval Research Grant No. N00014-15-1-2789.

REFERENCES

[1] B. S. Bloom, "Learning for Mastery. Instruction and Curriculum.," *Reg. Educ. Lab. Carol. Va. Top. Pap. Repr. Number 1*, no. (Reprinted from Evaluation Comment 1(2))., 1968.

[2] J. E. Beck and Y. Gong, "Wheel-Spinning: Students Who Fail to Master a Skill," in *Artificial Intelligence in Education. AIED 2013. Lecture Notes in Computer Science, vol 7926.*, Springer, Berlin, Heidelberg, 2013, pp. 431–440.

[3] Y. Gong and J. E. Beck, "Towards Detecting Wheel-Spinning: Future Failure in Mastery Learning," in *Proceedings of the Second ACM Conference on Learning @ Scale. L@S '15.*, New York, NY, USA, 2015, pp. 67–74.

[4] N. Matsuda, S. Chandrasekaran, and J. C. Stamper, "How quickly can wheel spinning be detected?," in *Proceedings of the 9th International Conference on Educational Data Mining - EDM 2016*, Raleigh, North Carolina, USA, 2016, pp. 607–608.

[5] T. Käser, S. Klingler, and M. Gross, "When to Stop?: Towards Universal Instructional Policies," in *Proceedings of the Sixth International Conference on Learning Analytics & Knowledge*, New York, NY, USA, 2016, pp. 289–298.

[6] J. Rollinson and E. Brunskill, "From Predictive Models to Instructional Policies," in *Proceedings of the International Conference on Educational Data Mining (EDM 2015)*, Madrid, Spain, 2015.

[7] N. Sharada, M. Shashi, and X. Xiong, "Identification of Wheel Spinning Cases while Learning and Retaining a Skill in Intelligent Tutoring Systems," *Int. J. Comput. Appl.*, vol. 179, no. 3, pp. 27–31, 2017.

[8] S. Park and N. Matsuda, "Predicting Students' Unproductive Failure onIntelligent Tutors in Adaptive Online Courseware," in *Proceedings of the Sixth Annual GIFT Users Symposium*, Orlando, Florida, USA, 2018, pp. 131–138.

[9] S. Kai, M. V. Almeda, R. S. Baker, C. Heffernan, and N. Heffernan, "Decision Tree Modeling of Wheel- Spinning and Productive Persistence in Skill Builders," *J. Educ. Data Min. JEDM*, vol. 10, no. 1, pp. 36–71, 2018.

[10] A. F. Botelho, A. Varatharaj, T. Patikorn, D. Doherty, S. A. Adjei, and J. E. Beck, "Developing Early Detectors of Student Attrition and Wheel Spinning Using Deep Learning," *IEEE Trans. Learn. Technol.*, vol. 12, no. 2, pp. 158–170, 2019.

[11] C. Zhang *et al.*, "Early Detection of Wheel Spinning: Comparison across Tutors, Models, Features, and Operationalizations," in *In Proceedings of the Twelfth International Conference on Educational Data Mining (EDM 2019).*, Montreal, QC, Canada, 2019.

Cochleagram-based Approach for Detecting Perceived Emotions in Music

Luka Kraljević
University of Split, FESB
Split, Croatia
lkraljev@fesb.hr

Mladen Russo
University of Split, FESB
Split, Croatia
mrusso@fesb.hr

***Abstract*—Based on the importance of the auditory system in emotional recognition and processing, in this work, we mimic human cochlea behavior which represents the first step in the cognitive process responsible for obtaining the appropriate emotional response. A new cochleagram-based approach for detecting the affective musical content is proposed. Evaluation was performed on publicly available 1000 songs database and experimental results showed that the proposed approach can be used in state of the art music emotion recognition systems.**

***Index Terms*—Music emotion recognition, Cochleagram, CNN.**

I. Introduction

Music have affective characteristics which are used for mood and emotion regulation. With the rapid growth of easily-accessible digital music libraries emotions become an important criterion for easy and efficient search, retrieval and musical organization. Automatic music emotion recognition (MER) is a challenging task because the emotional experience is personal, depending not only on the current emotional state but also on factual factors such as age, gender, and cultural background. Also, words used to describe emotions are often ambiguous and there isn't a universal way to quantify emotions. Most related studies are based on two popular approaches to emotion conceptualization. Categorical approach is based on finding and organizing a set of universal emotional categories which makes this approach suitable for integrating with standard text-based search system where user can retrieve song by emotional keyword. Categorical MER imposes granularity problem. Choosing a small number of categories means that only a portion of the available music emotions is covered, on the contrary using finer granularity can lead to overwhelming users a cognitive level of emotion perception thus posing ambiguity issues. In this work we adopted valence-arousal (VA) dimensional approach, where emotional space is viewed as infinitive number of emotion descriptions, thus having the potential to make more delicate distinctions. This approach allows to represent each song as a point in VA emotional plan and to investigate time-varying relationship between music and emotions [1]. Valence represents natural attractiveness or awareness of any emotion, while arousal can be viewed as the strength of emotion.

This work has been fully supported by the Croatian Science Foundation under the project number UIP- 2014-09-3875.

II. Methodology

Detecting emotions from music is a perceptive task, and nature has developed an efficient way to accomplish it. Besides hearing, perceiving emotions also depends on personal and situational factors such as listeners familiarity with music and musical preferences. These factors are processed by cognitive processes involved in the formation of subjective emotional experience from perceptual input.

A. Cochleagram

In this work, we mimic human cochlea behavior which represents the first step in the cognitive process responsible for obtaining the appropriate emotional response. Gammatone filterbank is a widely used model of auditory filters. It performs frequency analysis of the audio signal and outputs the signal into channels where each channel represents motion of the basilar membrane. To obtain gammatone based cochleagram input signal is first processed by bank of gammatone filters where the energy-per-band is calculated using 50 ms windows with 25 ms overlap. The corresponding cochleagram is obtained by firstly converting the energy values to dB and then generating a spectrogram image of energy-per-band over time. Cochleagram example is shown in Fig.1

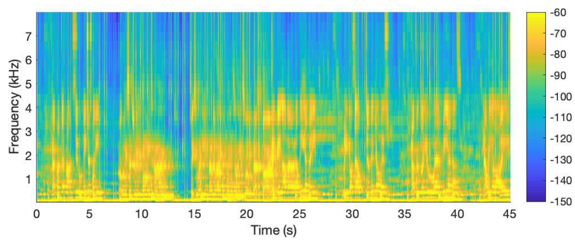


Fig. 1. Cochleagram for 2.mp3

B. Feature extraction

Feature extraction is an important step in obtaining affective information from the audio signal. Considering the benefits of using Convolution Neural Network (CNN)) in visual processing, we propose that we stand to gain from a similar process in machine hearing with neural networks. In this work two, musically motivated CNN architectures are employed for feature extraction [2]. In visual analogy dimensions of cochleagram represent fundamentally different units from those used in standard visual objects so it is natural to assume that network performance could gain of using musically motivated filters shape such as temporal filters and frequency filters.

III. Performance Study

To ensure verifiability of our approach we used official benchmark database for evaluation task: Emotions in music as part of MediaEval challenge [3]. Working set was made up of 744 music clips where each music excerpt was 45s long and sampled with the frequency rate of 44100Hz. Emotional music database contains files with averaged continuous annotation ratings collected with 2Hz sampling rate. Valence and arousal ratings were recorded separately on the scale between -1 and 1. There is also a file with arousal and valence ratings of the whole clip on a nine point scale. For performance study two experiments are conducted: static task which use annotations of the whole clip and dynamic task which use frame level annotation for continuous music emotion variation detection (MEVD). As instructed by the challenge both annotations and subsequently prediction were scaled between [-0.5, 0.5] to ensure comparability of the results. Evaluation results are given in terms of RMSE and R^2 statistic which indicates portion of underlying data variation that can be explained by the trained regression model.

A. Performance Evaluation for Dimensional MER

Dimensional approach to MER is viewed as regression task where each pair of valence-arousal values are used to describe song as a point in emotional space. Regression results in terms of R^2 were 61,1% for arousal and 36.2% for valence. Reported averaged RMSE values are 0.102 and 0.116 respectively for arousal and valence. Distributions of ground truth and prediction values for 100 songs of the total test set is shown in Fig. 2.

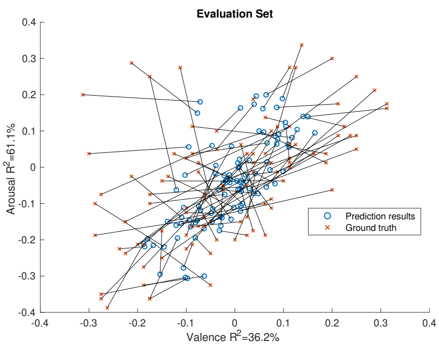


Fig. 2. Distributions of ground truth and prediction values

B. Performance Evaluation for MEVD

Music is time-varying, so it is possible that the listener's emotional response also changes with time. To investigate the time-varying relationship between music and emotion, in this work we used the provided dynamic annotation for prediction of the emotional content of the music. Evaluation metrics in terms of R^2 are 53.2% for arousal and 41.3% for valence while average RMSE score is 0.099 and 0.108 respectively for arousal and valence. To show an example of visualization how affective content of a music piece dynamically changes over time, we trained regression model to predict values of the each 500ms frame resulting in valence and arousal emotional curves as shown in Fig 3. Note that prediction curves are flattened by applying moving average to ease jumps in predicted values.

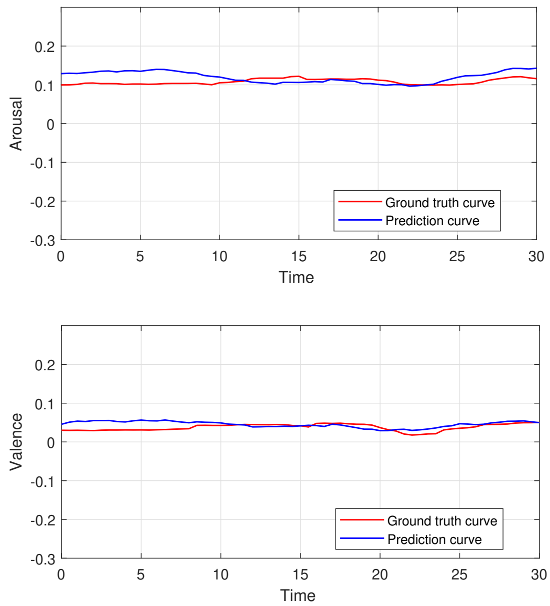


Fig. 3. Dynamic changes of the affective content of the music piece

IV. Conclusion

Modeling the natural response of the human cochlea we can simulate machines to act as the human ear, which represents the first step in cognitive processes responsible for obtaining the appropriate emotional response. This work proposes, a new approach for recognition of emotional content in music based on cochlear model of auditory perception and feature extraction using musically motivated CNN filters. Conducted performance study showed that CNN can successfully extract relevant affective features from cochleagram which can be used to achieve state of the art results in regression task. Generating cochleagram from a more complex model of auditory processing, thus simulating cochlea in more detail, could bring us further in improving music emotion recognition.

References

[1] Y. Yang, Y. Lin, Y. Su and H. H. Chen, "A Regression Approach to Music Emotion Recognition" in IEEE Transactions on Audio, Speech, and Language Processing, vol. 16, no. 2, pp. 448-457, Feb. 2008.

[2] J. Pons, T. Lidy and X. Serra, "Experimenting with musically motivated convolutional neural networks," 2016 14th International Workshop on Content-Based Multimedia Indexing (CBMI), Bucharest, 2016, pp. 1-6.

[3] M. Soleymani, M. N. Caro, E. M. Schmidt, C.-Y. Sha, and Y.-H. Yang, "1000 songs for emotional analysis of music," in Proceedings of the 2nd ACM international workshop on Crowdsourcing for multimedia. ACM, 2013, pp. 1–6.

Lane Detection problem in Automotive Applications

Denis Vajak, Mario Vranješ, Ratko Grbić
Faculty of Electrical Engineering, Computer Science and Information Technology
Osijek, Croatia
denis.vajak@ferit.hr, mario.vranjes@ferit.hr, ratko.grbic@ferit.hr

***Abstract*—Recently, Advanced Driver Assistance Systems (ADAS) are becoming a popular topic both in the automotive industry and in research. One of the commonly used systems is called Lane Detection (LD). It is used for keeping the vehicle within a lane and warning about departure from it. This paper describes the LD problem, presents some recent advancements in this area, along with providing explanations of different techniques commonly used in research. Additionally, a plan of our future work and contributions in this area are presented.**

Keywords—lane detection, lane tracking, computer vision, deep learning, automotive

I. Introduction

With the fast development of Advanced Driver Assistance Systems (ADAS), various intelligent functionalities are now implemented to vehicles on a regular basis, making way for the development of autonomous vehicles. One of the important ADAS components is the Lane Detection (LD) system. This system allows for a vehicle to recognize the lane lines on the road and offer the driver various possible functionalities: a warning in case the vehicle starts steering away from the line (departure warning), help keeping the vehicle in the lane by turning the wheel automatically, or be part of an autonomous driving system.

There is plenty of research in this area, providing various solutions to this problem. This paper provides categorization of various commonly used techniques when constructing an LD solution. Additionally, this paper points out the problems and drawbacks that these solutions seem to suffer from and provides an overview of our future work in this area.

II. Lane detection solutions – categorization and evaluation

The general design of LD includes a camera mounted on the vehicle which gathers image data and a system of algorithms which perform processing in order to detect the lane. An example of a view from the camera of a vehicle running an LD system is shown in Fig. 1, as seen in [1]. LD solutions can be categorized according to the algorithm types they use and according to the integration levels which will be discussed in this chapter. Additionally, this chapter will discuss the methodologies used while evaluating the existing LD solutions.

A. Solution categorization by algorthm types

While most of the modern LD solutions share the basic ideas about the general architecture and design, according to [2], solutions can be divided in two major categories: traditional computer vision (CV) based solutions and deep learning (DL) based solutions. These two types rely on different internal mechanisms in order to process the input image data.

Fig. 1. Example of a lane detection system

Traditional CV-based solutions mostly focus on traditional CV algorithms. Traditional CV is still highly popular in ADAS applications because it doesn't require specialized hardware for execution and because it has been efficiently used in various fields other than automotive for a long time. In traditional CV, multiple underlying algorithms are commonly combined in order to achieve the common goal of detecting and tracking the lane. The most popular algorithms used in a wide range of these solutions are: Hough transform, Gaussian mixture models (GMM), Kalman filter, defining a region of interest (ROI), vanishing point detection and random sample consensus (RANSAC). An example solution which uses the first three algorithms is presented in [3], while [4] makes use of the second three algorithms.

Unlike traditional CV-Based solutions, DL-based solutions are relatively newer and focus on using DL and artificial neural networks in order to process the incoming data. One drawback of using these methods is a requirement for more powerful hardware (usually dedicated graphical processors) in order to process the data in a timely manner. Another drawback is the need of well formatted training data sets in order to teach the deep neural networks how to properly recognize lane features. Such data sets are still rare and quite sought after. Researchers often have to manually add labels to already existing data sets in order to augment them for an application like this, like it was done in [5]. However, the positive sides of using a DL-based solution greatly outweigh the bad ones, allowing for the solution to have a much higher accuracy and be way more robust in various road conditions compared to using traditional CV algorithms. While there aren't as many DL-based solutions as there are CV-based ones, the plan is to primarily focus our research on these, as they have a better potential and prospect in the future. One such solution is presented in [6], where the authors have constructed a new type of convolutional neural network (CNN), which they dubbed Spatial CNN (SCNN). This new type of network is very suitable for long continuous objects, for instance a lane line on the road. To show that hardware choice can improve performance, the solution presented in [7] implemented the neural network on a Field-Programmable Gate Array (FPGA) board, achieving a highly efficient and fast detection rate. DL can also be used alongside with CV algorithms in order to further augment the solution. For instance, [8] used a deep neural network (DNN) as one of the steps in a longer traditional CV processing chain.

B. Solution categorization by integration levels

In order to better differentiate the solutions, [2] also proposes categorizing solutions by different levels of integration. These levels describe in which ways the algorithms are combined in order to perform their processing task. Three levels of integration exist: algorithm, system and sensor level. Algorithm level integration is the simplest and refers to the solutions which use a series of algorithms which are interconnected with each other directly. System level is more complex and covers solutions which divide their algorithms in groups, each dedicated to processing a different stage. Sensor level is the most complex and covers solutions which use multiple different sensors in order to increase the accuracy of detection. While most solutions use a simple monocular camera to acquire their data of the road, there are some solutions like [9] or [10] which may use more sensors.

C. Evaluation methodology

In order to evaluate various solutions, it is important to understand the various components which may affect the evaluation results: environmental factors, data set choice and evaluation metrics.

Environmental factors pose an important challenge for all LD solutions. As most of the solutions use simple monocular cameras, they are rarely immune to the same issues which would affect a human driver: bad driving conditions. Snow, rain, fog, night and general bad visibility all affect the algorithms negatively, so it is important to evaluate the algorithms against these kinds of situations as well. A common solution for this is to add additional sensors which are not affected by bad conditions, like it was done in [9].

Evaluation may be done online or offline. When online, a camera, along with required processing hardware, is mounted on a vehicle to perform the evaluation in real-time. When evaluating offline, a pre-recorded data set needs to be used. Choice of the data set is important, especially when using sensors which are unable to use image data, like a Light Detection and Ranging (LiDAR) system. These solutions need to be tested using a special data set which is rarely public. This may pose an issue when trying to compare these solutions which a newly made one.

Finally, when evaluating, it is important to consider various metrics. The solutions may be compared by accuracy rating, which can be calculated by comparing the detection results with the real road data. Another metric used is the confidence rating returned by the proposed solution's algorithms. Finally, it is possible to compare the results in how quickly they can process the data (frame rate), as the solutions should be able to process data as close to real-time as possible.

III. ACHIEVED RESULTS AND FUTURE WORK

In order to enter this area of research, we have analysed a number of previous solutions, mostly published after first half of 2017. We have categorized the solutions based on the previously proposed categorization techniques and identified their weaknesses and area for improvement. Additionally, we have managed to locally train and run the DL-based solution covered in [6], providing a good baseline for future research, which will focus on improving the existing techniques and constructing a new data set.

Firstly, as discussed earlier in this paper, we have identified a severe lack of well-labelled data sets which may be used to train DL-based solutions. Our plan is to create our own data set with videos recorded on the real road using an automotive camera, along with embedded ground truth data. Such a data set will allow additional training of various DL-based solutions along with our own, as well as evaluation of various other solutions.

Furthermore, we plan to construct and experiment with new DL-based solutions in order to increase the accuracy of detection or the performance of the system. This can be achieved through various changes to the already existing solutions by introducing changes to the various algorithms, neural network layouts, and so on, ultimately resulting in a new and better solution.

ACKNOWLEDGMENT

This work was supported by University of Osijek business fund through the internal competition for the research and artistic projects „UNIOS ZUP-2018“, and by Faculty of electrical engineering, computer science and information technology Osijek competition for the research projects "IZIP 2019".

REFERENCES

[1] D. Špoljar, "Detekcija voznih traka ispred vozila pomoću kamere i upozoravanje na napuštanje trake", Master Thesis, Faculty of Electrical Engineering, Computer Science and Information Technology Osijek, Josip Juraj Strossmayer University of Osijek, Croatia

[2] Y. Xing, C. Lv, L. Chen, H. Wang, H. Wang, D. Cao, E. Velenis and F.Y. Wang, "Advances in Vision-Based Lane Detection: Algorithms, Integration, Assessment, and Perspectives on ACP-Based Parallel Vision", IEEE-CAA Journal of Automatica Sinica, vol. 5, iss. 3, pp. 645-661, May 2018.

[3] Y. Xing, C. Lv, H. Wang, D. Cao and E. Velenis, "Dynamic integration and online evaluation of vision-based lane detection algorithms", IET Intelligent Transport Systems, vol. 13, iss. 1, pp. 55-62, January 2019.

[4] X. Wang, C. Kiwus, C. Wu, B. Hu, K. Huang, A. Knoll, "Implementing and Parallelizing Real-time Lane Detection on Heterogeneous Platforms", IEEE 29th International Conference on Application-specific Systems, Architectures and Processors (ASAP), pp. 25-32, July 2018.

[5] X. Liu, Z. Deng, "Segmentation of Drivable Road Using Deep Fully Convolutional Residual Network with Pyramid Pooling", Cognitive Computation, vol. 10, iss. 2, pp. 272-281, April 2018.

[6] X. Pan, J. Shi, P. Luo, X. Wang, X. Tang, "Spatial as Deep: Spatial CNN for Traffic Scene Understanding", AAAI Conference on Artificial Intelligence (AAAI), February 2018.

[7] Y. Lyu, L. Bai, X. Huang, "Real-Time Road Segmentation Using LiDAR Data Processing on an FPGA", IEEE International Symposium on Circuits and Systems (ISCAS), May 2018.

[8] W. Song, Y. Yang, M. Fu, Y. Li, M Wang, "Lane Detection and Classification for Forward Collision Warning System Based on Stereo Vision", IEEE Sensors Journal, vol. 18, iss. 12, pp. 5151-5163, June 2018.

[9] T.T. Nguyen; J. Spehr, S. Zug, R. Kruse, "Multisource Fusion for Robust Road Detection Using Online Estimated Reliabilities", IEEE Transactions on Industrial Informatics, vol. 15, iss. 11, pp. 4927-4939, November 2018.

[10] W. Li, Y. Guan, L. Chen, L. Sun, "Millimeter-Wave Radar and Machine Vision-Based Lane Recognition", International Journal of Pattern Recognition and Artificial Intelligence, vol. 32, iss. 5, May 2018.

Heart Chamber Localization and Segmentation Based on Deep Learning Methods

Student: Filip Novoselnik, Advisor: Irena Galić
Visual Computing Group
Faculty of Electrical Engineering, Computer Science and Information Technology Osijek
Osijek, Croatia
E-mail: filip.novoselnik@ferit.hr

Abstract— **Medical imaging is essential diagnosis tool in modern medicine. Large amounts of data is generated and physicians are not able to analyse and use them efficiently anymore. In the last few years, machine learning and deep learning methods are dominating the field of medical imaging. Cardiovascular imaging became crucial part of diagnosis and patient management. Localization and segmentation of the heart chambers is essential step for volumetric and functional analysis of the heart. In this paper, review of methods for heart localization and segmentation is given. Also, plan for future research is presented and discussed.**

Keywords— ***cardiac imaging, CT, MRI, deep learning;***

I. Introduction

Recent advances in medical technology resulted in huge amounts of complex multimodal data. Thus, there is a need for more advanced and sophisticated methods for their processing and analysis. Today, state-of-the art methods used in medical imaging are based on machine learning and most recently deep learning techniques. Overview of medical imaging modalities with their advantages and disadvantages is given in Tab. 1. CT (Computer Tomography) and MRI (Magnetic Resonance Imaging) are two most widely used imaging modalities. Although CT imposed itself as most promising modality, risk of radiation exposure is a big issue. On the other hand, MRI offers safer technique for analyzing heart tissue and cardiac function, but it is still very expansive.

TABLE I. CARDIOVASCULAR IMAGING MODALITIES COMPARISON

Imaging modality	Advantages	Disadvntages
CT	▪ fast data acquisition ▪ quantification of vascular calcification ▪ very high spatial resolution	▪ radiation exposure ▪ low soft tissue contrast ▪ poor temporal resolution ▪ need for breath-hold
MRI	▪ no radiation exposure ▪ both anatomical and functional information ▪ very good tissue characterization	▪ expansive ▪ time-consuming ▪ long acquisition time ▪ complications in a case of unstable heart rythm
Ultrasound	▪ simple and fast ▪ no radiation exposure ▪ portable ▪ reproducible measurements	▪ limited tissue characterization ▪ accoustic window limitations ▪ narrow field of view
PET	▪ very accurate evaluation physiological function of the heart	▪ radiation exposure ▪ expansive ▪ poor spatial resolution

According to WHO (World Health Organization) cardiovascular diseases are leading cause of death in the world [1]. Heart chambers segmentation is one of the essential steps in diagnostics. It is very time-consuming process for radiologists if it is done manually. Manual segmentation is also prone to intra- and inter-observer variability. Thus, there is a huge need for fast and accurate automatic heart chamber localization and segmentation methods. Heart CT scan with heart chambers segmented in all three planes (axial, coronal and sagittal) is given in Fig. 1. Given all three planes of the CT scan with ground truth data, it is possible to build 3D model of the heart (Fig. 2.). Images shown on Fig.1. and Fig. 2. are part of an ongoing research conducted by author.

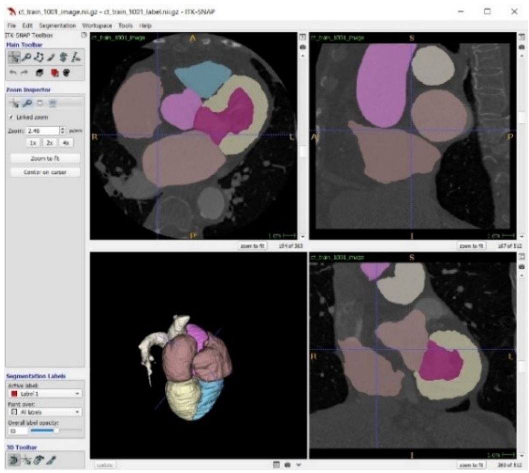


Fig. 1. CT scan with segmented heart chambers (ITK-SNAP software package [13])

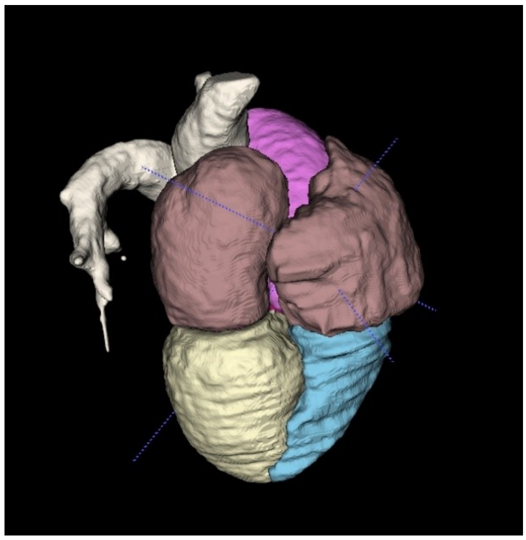


Fig.2. 3D model of human heart with segmented chambers

II. Deep Learning

With the exponential growth of available data, interpretation and analysis of medical images becomes more and more complex and time-consuming. Deep learning methods are class of machine learning algorithms that use multiple layers to extract higher level features from the raw input. One of the most important deep learning models applied to visual imagery are convolutional neural networks (CNN). Although the basic idea behind CNNs was described in 1982. due to the lack of computing power, large labeled datasets and efficient algorithms for their training they remained silent for the next decade. From the 2015 majority of papers in medical image analysis are using CNNs and similar architectures. Deep learning has begun to revolutionize medicine and cardiovascular imaging is not an exception.

III. Convolutional Neural Networks For Cardiac Imaging

In the last few years Convolutional Neural Networks (CNN) are most active area of research in medical image processing and analysis. CNNs are very suitable for image classification and recognition tasks because they are able to capture local feature relationships on image, while at the same time performing dimensionality reduction as it is shown in Fig. 3.

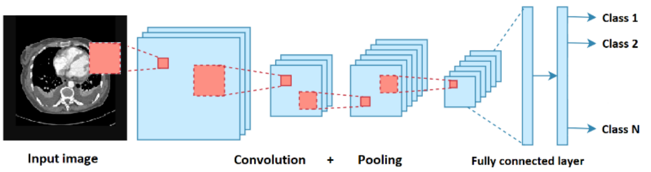


Fig.3. Convolutional Neural Network (CNN) architecture

In cardiac imaging, CNNs are mostly used for heart chambers localization and segmentation. Basic convolutional neural network architecture is shown in Fig. 3. Authors in [4] have developed the CNN model for heart localization and segmentation in chest CT or MR image. They achieve 88.9 % Dice Similarity [5] for segmentation. Authors in [6] developed hybrid method for whole heart segmentation based on statistical shape method. Research from authors [7] resulted in a CNN for localization of anatomical structures in 3D medical images based on their presence in 2D images. In [8] author use fully convolutional neural network (FCNN) for pixel-wise segmentation of right ventricle (RV) and left ventricle (LV). CNNs are mostly used for localization and segmentation of heart chambers [9], [10]. One interesting application of the CNN is to remove coronary calcification artefacts by image in-painting [11]. Many of today's methods rely on U-net neural network architecture that consists of a contracting path to capture context and a symmetric expanding path that enables precise localization [12].

IV. Conclusion

Cardiovascular imaging is undergoing major transformation and deep learning is playing the key role. As a part of the PhD research, author is developing methods for heart localization and segmentation based on deep neural networks. Future work includes improvement of existing convolutional neural network architectures for heart chamber segmentation. As a lack of large annotated datasets is often a huge problem for deep neural network models the special emphasis in research will be given to the data augmentation and synthetic data generation.

References

[1] WHO website (17 May 2017)

[2] LeCun, Y., et al.: Backpropagation applied to handwritten zip code recognition. Neural Computation 1(4), 541-551 (1989)

[3] Krizhevsky, A., et al.: Imagenet classification with deep convolutional neural networks. In: NIPS. pp. 1106-1114 (2012)

[4] Payer, C. et al.: Multi-label Whole Heart Segmentation Using CNNs and Anatomical Label Configurations (2018).

[5] Dice, Lee R. (1945). "Measures of the Amount of Ecologic Association Between Species". Ecology. 26 (3): 297–302

[6] Wang, C. et al.: Automatic Whole Heart Segmentation Using Deep Learning and Shape Context (2018)

[7] De Vos, B. et al.: ConvNet-Based Localization of Anatomical Structures in 3D Medical Images. IEEE Transactions on Medical Imaging. PP. 1-1. (2017)

[8] Vu Tran, Phi: A Fully Convolutional Neural Network for Cardiac Segmentation in Short-Axis MRI (2016).

[9] Narayan, T.: "Automated Left Ventricle Segmentation in Cardiac MRIs using Convolutional Neural Networks." (2016).

[10] Emad, O. et al.:Automatic localization of the left ventricle in cardiac MRI images using deep learning. Conference proceedings: ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Conference. 2015. 683-686.

[11] Yan, S. et al.: Calcium Removal From Cardiac CT Images Using Deep Convolutional Neural Network (2018), 2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018), 4-7 April 2018, Washington, DC, USA

[12] Ronneberger, Olaf & Fischer, Philipp & Brox, Thomas. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. arXiv:1505.04597 [cs.CV]

[13] P. A. Yushkevich et al.: User-guided 3D active contour segmentation of anatomical structures: Significantly improved efficiency and reliability. Neuroimage 2006 Jul 1;31(3):1116-28.

Location-Aware Scheduling of IoT Services in Fog Computing

Student: Petar Krivic, Advisor: Mario Kusek
Faculty of Electrical Engineering and Computing
Department of Telecommunications
Zagreb, Croatia
{petar.krivic,mario.kusek}@fer.hr

***Abstract*—Centralized processing that Cloud computing implies cannot ensure low latency, which is one of the main requirements for the upcoming concepts within the area of Internet of things (IoT) that demand autonomous reactions in real time. A possible solution for those systems could be recently emerged concept named Fog computing that implies processing of the user requests on devices placed in lower layers of network, in environments closer to the end-users. Our aim within this research is to track user locations within recognised networks and to dynamically calculate the most efficient schedule of services on available processing nodes in their environments. The premise of such an approach is that it would enable meeting the latency requirements of modern IoT case studies, lower down the network traffic and increase the level of resilience and availability for critical services.**

I. Introduction

Main entities of today's IoT infrastructure are sensors and actuators - physical devices that are the base of every service within this concept [1]. Such services generate high data volumes that require sufficient and elastic processing infrastructure offered by cloud [2], which is the reason why most of the existing IoT case studies are based on its utilization for processing and storage resources [3]. In most of the cases, such centralized architectures have delivered a sufficient quality of service by now, but the upcoming concepts of IoT that are based on autonomous and reactive processes demand lower latency and higher throughput to deliver real-time responses and reactions, and thus, it is important to bring processing services closer to the end-users when possible [4, 5].

Fog computing concept emerged as a response to such requirements, and even its name suggests that it offers the extension of Cloud computing in environments closer to the end-users, and load distribution from cloud towards entities in local environments [6]. There is no standard that completely defines this concept, but the OpenFog Reference Architecture [7] gives the most comprehensive overview of all subjects included, and their definition of Fog computing is that it is a horizontal, system-level architecture that distributes computing, storage, control and networking functions closer to the users along a cloud-to-thing continuum. Big contribution of this standard is also its description for the realization of Fog computing from different perspectives. The one concerning the software design of this concept outlines microservices and containerization as the key mechanisms to efficiently implement the required functionalities.

Following these recommendations, within this doctoral research we plan to develop algorithm that schedules IoT services towards user environments according to their current locations, and afterwards verify its efficiency.

II. Research topic presentation

Existing papers within the research area of Fog computing mostly focus on strategies that schedule user requests to available nodes in the fog layer, or in other cases, balance request processing between fog and cloud depending on various parameters such as processing capability of a fog node, task transmission time, propagation, etc. In both cases, their criteria for efficiency of proposed algorithms was based on latency optimization in comparison to the usual centralised cloud architecture [8–10].

Models that enabled the development of such algorithms were focused on the description of relevant parameters that affect the request processing efficiency. Thus, models would in most cases specify the complexity of incoming requests on one hand, and on the other, resources of available processing nodes and the quality of their mutual network connections, to calculate the probability that determines which of them is the most suitable for processing an upcoming request.

According to these facts it is important to outline that such algorithms would mainly presume that all processing services are running on each available node [8], and the algorithm that is to be proposed in this doctoral thesis focuses on scheduling required services only on nodes where exists the interest for their usage. The problem of scheduling services for mobile users along fog environment is partially analysed in paper [11] where Bittencourt et al. conclude that the most important parameters to target this problem are QoS of the specific application and the accuracy of user movement predictions.

Algorithm within this doctoral research would thus focus on determining the best strategy for user tracking around fog environment, and calculating the most efficient service schedule on top of it to lower down the latency and to increase system resilience and autonomy in local environments.

III. Research methodology

Research will be divided in several stages:

PhD FORUM - 91485- 2009 © SoftCOM 2019

A. Analysis of existing Fog computing researches and available tools for managing distributed system environment

The goal of the first phase in our research is to detect all relevant parameters that affected the delivered quality of service in existing research papers targeting Fog computing. Since our goal is to target our research towards IoT environments, we will especially focus on such case studies to identify all characteristic strategies for request and service scheduling alongside fog-to-cloud environment.

Another part of the initial phase is to explore available tools to manage distributed system services. As mentioned before microservices and containerisation are enabler technologies for presented service scheduling in Fog computing. There are multiple software tools that enable manipulation of these mechanisms around distributed environment such as: Kubernetes, Istio, Apache Mesos, Zipkin and many more. Our goal here is to investigate the most popular ones and to determine which of these could be utilised in our further work.

B. Modeling the environment to enable the determination of the optimal service schedule according to the current state of user environment

Based on the researches in previous phase, we plan to compose a model that would generically describe entities for processing in local environments, types of services that could be performed within this environment, required QoS, and user devices that utilize the requested services. On top of this description we will adopt one of the available tracking strategies to describe and embed user location in our model as well. At the end of this phase, it is expected to have a fully designed model that describes all subjects and their mutual interactions required to achieve efficient service schedule, according to the current state of the user environment. Also, we will prioritise parameters on which services will be scheduled during the execution of the optimisation algorithm.

C. Algorithm for the optimisation of distributed system by scheduling its services towards devices in user environments

In the third phase of this research an algorithm will be developed on top of the previously composed model. As depicted in Figure 1, this algorithm will dynamically schedule services towards the user environments if there are available processing nodes within them. To calculate the most efficient schedule algorithm will consider three main input information:

- available user services and their current location,
- available devices and their network locations,
- current user network locations and assigned services.

Other factors that would also affect the decision making of this algorithm, are available computing resources and current processing load of each available node, and the predicted service migration complexity. Based on all these parameters algorithm would calculate the efficient schedule and thus enable the achievement of system optimisation by utilisation of Fog computing principles.

D. Verification procedure for location-aware optimisation scheduling algorithms

Last phase will be focused on the verification of the developed algorithm within the IoT environment. However, main purpose of this stage is not just to evaluate our algorithm and to verify its efficiency, but to identify key parameters that could also guide the assessment of algorithms within similar application areas. Final result of this research stage is to clearly point out the guidelines for rating the optimisation level achieved by the usage of algorithm that is being evaluated.

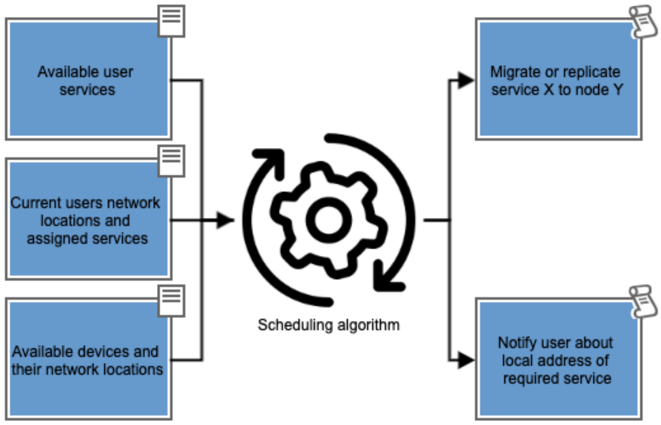


Fig. 1. High level description of location-aware service scheduling algorithm

REFERENCES

[1] S. Madakam, R. Ramaswamy, and S. Tripathi, "Internet of things (iot): A literature review," *Journal of Computer and Communications*, vol. 3, pp. 164–173, 04 2015.
[2] H. Truong and S. Dustdar, "Principles for engineering iot cloud systems," *IEEE Cloud Computing*, vol. 2, no. 2, pp. 68–76, Mar 2015.
[3] L. Hou, S. Zhao, X. Xiong, K. Zheng, P. Chatzimisios, M. S. Hossain, and W. Xiang, "Internet of things cloud: Architecture and implementation," *IEEE Communications Magazine*, vol. 54, no. 12, pp. 32–39, December 2016.
[4] R. Morabito, V. Cozzolino, A. Y. Ding, N. Beijar, and J. Ott, "Consolidate iot edge computing with lightweight virtualization," *IEEE Network*, vol. 32, no. 1, pp. 102–111, Jan 2018.
[5] S. Yi, Z. Hao, Z. Qin, and Q. Li, "Fog computing: Platform and applications," in *2015 Third IEEE Workshop on Hot Topics in Web Systems and Technologies (HotWeb)*, Nov 2015, pp. 73–78.
[6] O. Osanaiye, S. Chen, Z. Yan, R. Lu, K. R. Choo, and M. Dlodlo, "From cloud to fog computing: A review and a conceptual live vm migration framework," *IEEE Access*, vol. 5, pp. 8284–8300, 2017.
[7] "IEEE standard for adoption of OpenFog reference architecture for fog computing," *IEEE Std 1934-2018*, pp. 1–176, Aug 2018.
[8] A. Yousefpour, G. Ishigaki, and J. P. Jue, "Fog computing: Towards minimizing delay in the internet of things," in *2017 IEEE International Conference on Edge Computing (EDGE)*, June 2017, pp. 17–24.
[9] I. Filip, F. Pop, C. Serbanescu, and C. Choi, "Microservices scheduling model over heterogeneous cloud-edge environments as support for iot applications," *IEEE Internet of Things Journal*, vol. 5, no. 4, pp. 2672–2681, Aug 2018.
[10] D. Zeng, L. Gu, S. Guo, Z. Cheng, and S. Yu, "Joint optimization of task scheduling and image placement in fog computing supported software-defined embedded system," *IEEE Transactions on Computers*, vol. 65, no. 12, pp. 3702–3712, Dec 2016.
[11] L. F. Bittencourt, J. Diaz-Montes, R. Buyya, O. F. Rana, and M. Parashar, "Mobility-aware application scheduling in fog computing," *IEEE Cloud Computing*, vol. 4, no. 2, pp. 26–35, March 2017.

The Implications of End-user Service Usage Behavior Patterns on In-network Video QoE Monitoring and Management

Ivan Bartolec, Lea Skorin-Kapov
Department of Telecommunications
Faculty of Electrical Engineering and Computing
Zagreb, Croatia
Email: {ivan.bartolec, lea.skorin-kapov}@fer.hr

***Abstract*—One of the most popular types of multimedia services comprising the ever increasing global IP traffic are video streaming services. A significant amount of research has addressed the challenge of understanding, measuring and managing Quality of Experience (QoE) of online video services. One of the key aspects in understanding what impacts QoE and how to optimise systems to meet QoE requirements is end-user behavior and user's involvement in the video delivery process. User tendency to somehow interact with the video is a factor which may be exploited to properly design a system, offer a better service, and potentially detect service impairments. This paper presents research trends related to understanding end-user service usage behavior patterns and corresponding implications on in-network video QoE monitoring and management.**

***Index Terms*—Quality of Experience, QoE monitoring, QoE management, video streaming, user behavior**

I. Introduction

Video may be the most popular multimedia format, used in different Internet applications such as video streaming services, conversational services, social media, and cloud gaming. Cisco forecast that 79% of global mobile data traffic by 2022 will be video traffic[1]. As such, there is an increased interest from network operators and service providers to better understand and assess performance of applications that utilize video. Considering the growth of video traffic, availability of different formats, resolutions, and encryption as of late, management of video delivery over the Internet has become challenging. Additionally, users tend to engage with the video service, furthermore complicating performance estimation and management. Nonetheless, knowledge about certain user behavior may provide beneficial insights on performance and overall end-user satisfaction.

II. Quality of Experience Cycle

For years, the notion of quality in telecommunications was primarily linked to Quality of Service (QoS), which may not accurately reflect viewer's watching experience. The key question is how network performance impacts application level performance, which is perceived by end-users (e.g. video stalling). The definition of Quality of Experience (QoE) proposed by "QUALINET", the European Network of Excellence on QoE in Multimedia Systems and Services, states how QoE is "the degree of delight or annoyance of the user of an application or a service" [1]. Due to many factors influencing QoE (e.g., human, system, and context), measuring QoE is challenging. Overall QoE is commonly measured using subjective assessment methods, and reported as a Mean Opinion Score (MOS). In the context of adaptive video streaming, several studies report how application-level quality events such as initial delay, rebufferings and bitrate change ratios impact viewer's satisfaction and QoE [2].

Our focus in this paper is on the network provider's perspective of collecting QoE inputs needed for QoE assessment and management. QoE models dictate which parameters will be tracked and where mapping of monitored parameters to QoE is defined. An example of such a model is ITU-T Recommendation P.1203 which consists of an introductory document and documents describing three objective parametric quality assessment modules for adaptive video streaming [3]. QoE-related inputs are usually collected either via network probes, user-agents that explicitly collect data, or end-user devices. Many research activities on QoE management follow a cycle of reactive design, with focus on QoE monitoring and resource allocation decisions which occur only when QoE degradations are detected.

Firstly, service performance is monitored while important QoE inputs are collected and processed. Afterwards, given results are compared to thresholds defined by the network operators. If results are bellow a certain threshold then corrective actions have to be performed. However, most QoE management approaches lack consideration of user behavior. Figure 1 depicts a proposed generic QoE-driven network and service management process where user behavior is taken into consideration, since users may react in several ways, thus impacting the final resource allocation and actual provided quality [4].

[1]https://www.cisco.com/c/m/en_us/solutions/service-provider/forecast-highlights-mobile

TABLE I
OVERVIEW OF USER BEHAVIOR IMPLICATIONS ON QOE MONITORING AND MANAGEMENT.

Aspects of service usage behavior	Implications for QoE monitoring	Implications for QoE management
Interactions during service usage: pause, seek forward/backwards, quality switches	Need for QoE models that incorporate user interactions; investigate correlations between certain actions and QoE	If certain actions indicate dissatisfaction due to quality degradations, then perform root cause analysis, and fix the problem
Frequency of service usage and duration of service usage	Impact on monitoring costs, monitoring scalability, investigate correlation with QoE	Insights for resource allocation planning
User mobility in cellular networks	QoE model implementation location (i.e., on which network node); input duration for QoE estimation	Dynamic change of demands on network topologies; input for resource allocation algorithms

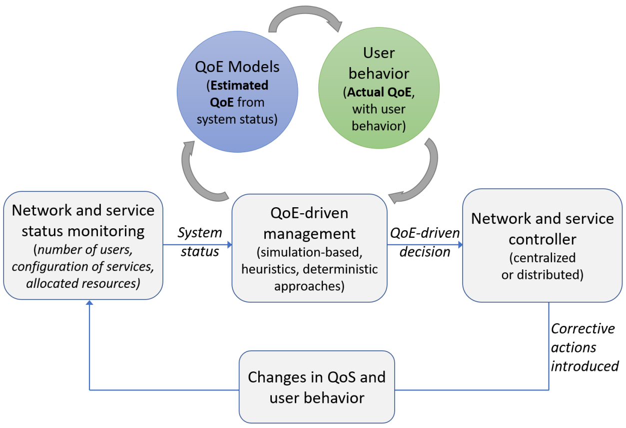


Fig. 1. Generic framework of QoE-driven management (taken from [4]).

III. USER BEHAVIOR

User behavior related to online video services can be manifested in multiple ways and considered from different standpoints. While certain behavior has a direct impact on video delivery performance, other behavior may impact background processes such as video recommendation. We can divide user behavior according to the following:

- **In-player user interactions:** behavior related to user initiated actions related to the video player (e.g., video pause, seeking forward/backwards, etc.);
- **User engagement:** Measurement of user's participation in a service (e.g., how often a user is using the service, and for how long);
- **User mobility:** Important aspect of mobile network usage due to issues that involve handovers, user location updating, registration, dimensioning of signaling network, etc.;
- **Predictions related to customer churn:** From a business perspective, tracking and preventing customer churn is crucial, along with attracting new customers;
- **Viewing context:** How users typically use the service on different access networks (e.g., WiFi, mobile data, device type, etc.).

IV. CASE STUDY: USER BEHAVIOR IMPLICATIONS ON QOE MONITORING AND MANAGEMENT FOR VIDEO-ON-DEMAND SERVICES

Understanding user behavior of online video services is important to design or improve video delivery systems to consume less network resources and at the same time satisfy QoS requirements. Table I provides an overview of end-user behavior implications on QoE monitoring and modeling of Video on Demand (VOD) services (e.g., YouTube or Netflix).

V. CHALLENGES

This section summarizes the key challenges related to implications of user behavior on in-network QoE monitoring and management, as discussed in this work. The following issues remain open research topics:

- **Investigation of usage patterns for target services:** Understanding user behavior and incorporating this knowledge in the QoE management cycle can in the long-term stop user churn, and in the short-term prevent users abandoning sessions;
- **Impact of user behavior on QoE models:** In the wild, users engage with the service in multiple ways for various reasons. Enhancing the models and monitoring of such engagements may bring a clearer picture when assessing QoE or KPIs;
- **Implications of user behavior on ML-based in-network QoE/KPI estimation models:** QoE models that were trained on datasets without any user interactions are not fully applicable for real world scenarios (i.e., estimation will not be as accurate for videos containing interactions) [5];
- **Impact of user behavior on QoE-driven resource allocation (incorporation of user behavior in the QoE management cycle):** Hence monitoring models do not consider user behavior, and user interactions may impact resource requirements, actual QoE may not be provided and appropriate actions may not be performed.

REFERENCES

[1] P. Le Callet, et al., "Qualinet White Paper on Definitions of Quality of Experience," QUALINET (COST Action IC 1003), 2012.
[2] M. Seufert, et al., "A Survey on Quality of Experience of HTTP Adaptive Streaming," in *IEEE Communications Surveys*, 2014.
[3] ITU-T, "Parametric Bitstream-based Quality Assessment of Progressive Download and Adaptive Audiovisual Streaming Services over Reliable Transport (P.1203)," 2017.
[4] T. Hoßfeld, et al., "The Interplay between QoE, User Behavior and System Blocking in QoE Management," in *ICIN*, 2019.
[5] I. Bartolec, et al., "In-Network YouTube Performance Estimation in Light of End User Playback-Related Interactions," in *QoMEX*, 2019.

QoE Assessment for Interactive Immersive AR/VR Applications

Sara Vlahovic, Lea Skorin-Kapov
University of Zagreb, Faculty of Electrical Engineering and Computing, Zagreb, Croatia
{sara.vlahovic, lea.skorin-kapov}@fer.hr

***Abstract*—Interactive Augmented Reality/Virtual Reality (AR/VR) applications and systems are designed with the goal of providing an immersive experience which enables users to interact with virtual elements in a real-world setting or creates an illusion of inhabiting a completely virtual world. However, at the time of this writing, AR/VR technology is still faced with serious obstacles, such as cybersickness, which highlights the need for further research aimed towards improving end user Quality of Experience (QoE). In this paper, we discuss key QQoE assessment methodology and research challenges we aim to address in our ongoing and future work.**

***Index Terms*—Quality of Experience, QoE modeling, Augmented Reality, Virtual Reality**

I. Introduction

The term **Augmented reality** (AR) describes different display technologies which enable various elements (e.g., text, symbols, two-dimensional, and three-dimensional graphic images) to be overlaid on top of the user's real-world view [1]. Azuma [2] defines the main characteristics of augmented reality systems: the combination of real and virtual elements, real-time interactivity and three-dimensional registration. Aukstakalnis [1] defines **virtual reality** (VR) as different display technologies (e.g., head-mounted displays, computer-assisted virtual environments) capable of generating sensations of immersion and presence inside a three-dimensional model or simulation, therefore creating a visual replacement of the real world. Our work focuses on interactive applications, i.e., those which enable users to navigate and/or manipulate the virtual environment, instead of passively observing.

Since both AR and VR are interactive multimedia services, promising unprecedented levels of immersion and completely new ways of presenting virtual worlds, it is vitally important for them to provide a comfortable and positive experience for their users in order to achieve wide-spread adoption. Therefore, as both of these technologies are yet to reach mainstream acceptance, significant effort has to be invested towards enhancing their overall Quality of Experience (QoE). Based on the definition given in the Qualinet White Paper on Definitions of Quality of Experience [5], the term Quality of Experience (QoE) is defined as [6]: *"the degree of delight or annoyance of a person whose experiencing involves an application, service, or system. It results from the person's evaluation of the fulfillment of his or her expectations and needs with respect to the utility and/or enjoyment in the light of the person's context, personality and current state"*. In case of AR/VR, improved QoE is achieved through increasing perceived immersion, presence and usability, while simultaneously finding ways to mitigate cybersickness and discomfort. Therefore, it is important to examine different QoE influence factors and assessment methods in order to understand their individual contributions to the overall QoE score.

II. Methodology and Research Challenges

We identified several key challenges related to QoE assessment for interactive immersive AR/VR applications, presented below.

A. Identification of key influence factors to be used for QoE modeling

User QoE is affected by multiple influence factors, ranging from human influence factors (e.g., experience, immersive tendency, history of illness) to system (e.g., input and output modalities, networking factors, content-related characteristics) and context (e.g., physical environment, social context, novelty) influence factors. While some of these factors are universal to virtual environments, or even multimedia services in general, some tend to be more AR/VR specific. Identifying these key factors enables defining independent variables which affect the end user QoE. These independent variables are used in designing test methodologies for user studies. Collected user ratings serve as input for QoE modeling.

B. Choice of participants/pre-screening

Choosing adequate participants for AR/VR is a challenging aspect of test methodology design because, in addition to greatly influencing QoE scores, certain human factors (e.g., vision, cybersickness succeptibility, history of illness, gender, age and experience) should be considered in advance as a precautionary measure for ensuring health and safety of all test subjects. Therefore, tests and questionnaires (e.g. visual acuity tests, cybersickness succeptibility questionnaires) should be defined and a method should be proposed for determining which participants should be excluded from the study based on collected data.

C. Chosen methods

In QoE studies, subjective metrics are commonly assessed through the use of questionnaires. Before testing, subjects are often asked to fill out a questionnaire addressing demographic information, such as age, gender and prior experience. During or after testing, they are asked to fill out questionnaires directly related to tested scenarios. In addition to subjective metrics, objective metrics (behavioral and physiological) are often used to assess user experience in a less biased way. Behavioral metrics refer to methods that are based on observing and tracking user behaviors, such as physical movement, social interaction, different choices etc. Physiological methods are based on measuring different physiological signals such as electrocardiography, electroencephalography and galvanic skin response. As discussed in [8], the use of psychophysiological measurements in assessing user experience improves existing QoE models, especially in terms of user-related factors, and mitigates issues stemming from the use of self-reported assessments [4, 3]. Both objective and subjective methods have been used in previous studies to gain a better understanding of different features of user experience. Using them in conjunction appears to be the most dependable way of obtaining results, as certain terms (e.g. presence, immersion) could be interpreted in different ways by different participants, which makes subjective methods less reliable when used alone. Relationships between subjective and objective methods with regards to a certain QoE feature should be further explored in future studies.

D. Questionnaires

So far, different types of questionnaires (both study-specific and more generalized) have been used to assess different QoE features. However, even most commonly used questionnaires (e.g. SSQ) are not necessarily adapted to AR/VR use cases, and are therefore in need of revision. When designing, or re-designing, questionnaires, effort should be made towards keeping them clear and concise, as well as incorporating them into the virtual environment.

E. Content

As observed by Schatz et al. [7], there is a lack of standardized content and test tasks specifically designed for immersive applications/systems/services. Addressing this issue would enable comparison between studies and reproduction of results.

F. Duration

Studies have shown that cybersickness accumulates with longer exposure [12], which subsequently affects user ratings, as well as physiological and behavioral metrics. AR/VR applications that require active movement may lead to exhaustion. Therefore, considering AR/VR technology tends to be more physically taxing compared to more conventional platforms, there is a need for guidelines addressing test duration.

G. Further research

Our recent work addressed the impact of locomotion methods [10] and network latency [11, 9] on user QoE in VR. Throughout our research we used study-specific questionnaires to assess features such as cybersickness and weapon precision. In our future work, we aim to examine other influence factors while simultaneously diversifying our chosen methods by incorporating physiological measurements in addition to questionnaires. Our goals also include achieving a more balanced distribution of participants in terms of static human factors.

REFERENCES

[1] Steve Aukstakalnis. *Practical Augmented Reality: A Guide to the Technologies, Applications, and Human Factors for AR and VR*. Addison-Wesley Professional, 2016.

[2] Ronald T Azuma. "A Survey of Augmented Reality". In: *Presence: Teleoperators & Virtual Environments* 6.4 (1997), pp. 355–385.

[3] Lee Anna Clark and David Watson. "Constructing Validity: Basic issues in Objective Scale Development". In: *Psychological Assessment* 7.3 (1995), p. 309.

[4] Bruno Gardlo, Sebastian Egger, and Tobias Hossfeld. "Do Scale-Design and Training Matter for Video QoE Assessments Through Crowdsourcing?" In: *Proceedings of the Fourth International Workshop on Crowdsourcing for Multimedia*. ACM. 2015, pp. 15–20.

[5] Patrick Le Callet, Sebastian Möller, Andrew Perkis, et al. "Qualinet White Paper on Definitions of Quality of Experience". In: *European network on quality of experience in multimedia systems and services (COST Action IC 1003)* 3.2012 (2012).

[6] Sebastian Möller and Alexander Raake. *Quality of Experience: Advanced Concepts, Applications and Methods*. Springer, 2014.

[7] Raimund Schatz et al. "Assessing the QoE Impact of 3D Rendering Style in the Context of VR-Based Training". In: *2018 Tenth International Conference on Quality of Multimedia Experience (QoMEX)*. IEEE. 2018, pp. 1–6.

[8] Lea Skorin-Kapov et al. "A Survey of Emerging Concepts and Challenges for QoE Management of Multimedia Services". In: *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)* 14.2s (2018), p. 29.

[9] Sara Vlahovic, Mirko Suznjevic, and Lea Skorin-Kapov. "Challenges in Assessing Network Latency Impact on QoE and In-Game Performance in VR First Person Shooter Games". In: *2019 15th International Conference on Telecommunications (ConTEL)*. IEEE. 2019, pp. 1–3.

[10] Sara Vlahović, Mirko Suznjevic, and Lea Skorin-Kapov. "Subjective assessment of different locomotion techniques in virtual reality environments". In: *2018 Tenth International Conference on Quality of Multimedia Experience (QoMEX)*. IEEE. 2018, pp. 1–3.

[11] Sara Vlahovic, Mirko Suznjevic, and Lea Skorin-Kapov. "The Impact of Network Latency on Gaming QoE for an FPS VR Game". In: *2019 Eleventh International Conference on Quality of Multimedia Experience (QoMEX)*. IEEE. 2019, pp. 1–3.

[12] Guan Wang and Ayoung Suh. "User Adaptation to Cybersickness in Virtual Reality: A Qualitative Study". In: (2019).

Overview of Big Data Optimizations in Internet of Things using Data Analytics

Jelena Čulić Gambiroža
Ericsson Nikola Tesla
Split, Croatia
jelena.culic.gambiroza@ericsson.com

Toni Mastelić
Ericsson Nikola Tesla
Split, Croatia
toni.mastelic@ericsson.com

Mario Čagalj
University of Split
Split, Croatia
mario.cagalj@fesb.hr

***Abstract*—Internet of Things (IoT) concept is growing in last few years and number of IoT devices is increasing rapidly. The amount of collected data is increasing as well, leading to large amount of data and Big data challenges such as high energy consumption. While individual IoT sensors consume relatively small amount of energy, they are mostly battery powered and numerous, which limits their lifetime and creates a great load on the backend systems. The system includes a complete data path from sensors, through network of gateways, to the backend cloud and its users. Our focus is to find and research key points of IoT system where Big data optimization can be performed.**

Keywords—Internet of Things; Big data; Energy efficiency.

I. Introduction

In the last few years Internet of Things (IoT) concept has grown and its topics are highly researched. It is predicted that there will be up to 75 billion devices connected in IoT by 2025 [1]. Basic IoT system consists of sensors collecting data, gateway as midpoint device and cloud where data is stored, streamed, analyzed, and presented.

Most of IoT systems have numerous sensors distributed within some area, each of which has its own battery with limited lifetime, which represents one of the main bottlenecks of this technology. Consequently, IoT sensor tasks have to be carefully planned as they can not handle large energy consumption requirements. Furthermore, numerous sensors collect data in short time intervals, which will result with up to 847 ZB of data per year by 2021[2]. Finally, all the collected data is sent towards cloud where it must be processed and stored in large data centers.

According to Cisco [2], data stored in data centers will nearly quintuple by 2021 and reach 1.3 ZB. However, more data is not always better than less data as collecting redundant data simply takes storage capacity without providing new information. As the volume of generated data increases, storing and processing it becomes significantly challenging. This is already recognized as the Big Data concept, commonly described with five Vs [3]. *Volume* represents the amount of data, while *velocity* is the speed at which new data is generated and the speed at which data moves around. *Variety* stands for different types of data. *Veracity* is the measure of data uncertainty, while *value* is the information obtained from the collected data.

Today, IoT requirements increase data volume and velocity, as well as variety. Finding the optimal balance between the volume reduction and the information loss (value) requires the utilization of data variety and thus reducing data velocity. The expected growth in the number of IoT data sources gives rise to network-edge computing [4]. Edge-mining stands for data processing on battery-powered devices placed at the edges of an IoT network. Such a solution would achieve reduction of data volume at the network edge and thus reduce energy consumption, bandwidth, as well as storage capacity and processing power at the cloud backend systems. The focus of our research is data velocity and volume reduction, while taking into a consideration data variety and preserving its value in all parts of an IoT system.

II. System Design

IoT systems consist of three main components, *sensors* that collect data from the environment and send it through *gateway* towards *cloud*, where data is analyzed and used for decision making. Each component receives the *input data*, performs internal data processing, and forwards *output data* (Fig.1). *Input* and *output data* can be received/forwarded using either *push* or *pull* mechanisms, or as their combination. Initiator for *push* mechanism is the southbound component, e.g., a sensor pushes the data towards a gateway. For *pull* mechanism initiator is the northbound component, e.g., a gateway requests data from sensor.

1) Sensors

Sensors are endpoint devices that read measurements from the environment. As endpoint devices, sensors are commonly battery powered and have low data processing power, while being numerous due to growth of IoT. Due to drastically reduced production costs, sensors are usually collecting more than one measurement type, so data collected by a single sensor can contain heterogenous values.

- *input data* – a measurement collected by the sensor
- *output data* – data processed on sensor and transmitted towards gateway

2) Gateways

Gateways are midpoint devices placed between sensors and a data analytics backend in the cloud. They allow sensors to communicate over shorter distances on its southbound, while on its northbound it forwards the data towards the cloud. Before forwarding data, it can perform additional processing for optimization or on-site analytics and decision making.

- *input data* – data coming from sensors
- *output data* – data received from sensors that is processed by the gateway and forwarded to the cloud

3) Clouds

Data collected by sensors and processed by gateways ends up in the cloud where they are analyzed and archived. Due to distance from sensors, real-time systems may suffer of high latency, but latency decreases with introduction of 5G and high speed Internet. Utilization of machine learning and prediction algorithms enables filling the gap between real-time systems and distant clouds. Finally, distribution of cloud towards Fog and Edge reduces this lag even more.

- *input data* – data that is received by the cloud services
- *output data* – once received by the cloud services the data is then processed, visualized and stored.

III. Research Challenges

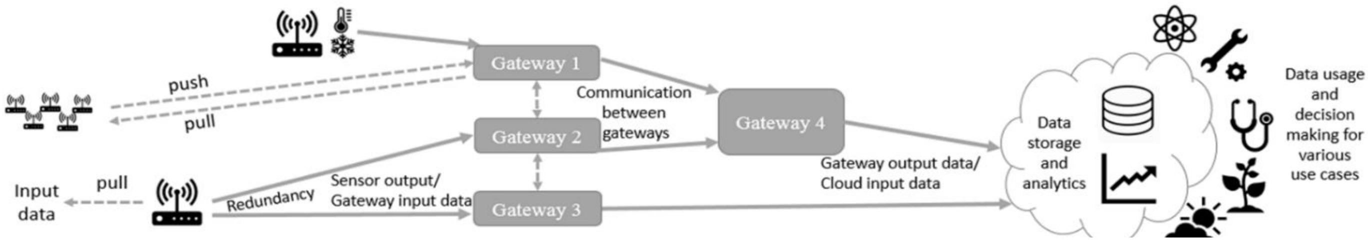


Fig. 1. IoT system design

In order to build an energy efficient IoT system that is able to tackle Big Data challenges it is important to understand where reduction in data volume and velocity can be made. This Section provides overview of the approaches with their applications on specific components of an IoT system and highlights their trade-offs. More details are available in [5].

1) Dynamic monitoring frequency

When reading data in short intervals, change in value is not necessarily detected, so data should be collected when some event occurs. While this is possible only when system itself creates event, it is possible to emulate such behavior by dynamically changing the monitoring frequency. The sensor should use lightweight algorithm to collect data only when the significant delta is expected and sleep in the meanwhile. If the current delta is lower than significant delta, monitoring interval increases and vice versa. If data is sampled after it is collected, sensor still has to spend energy for data collection and analysis. Despite changing frequency on sensor, gateway still receives large amount of data from multiple sensors in small intervals, so it can also orchestrate data collection on sensor by sending pull requests. Since the gateway is a bigger and usually plugged in, orchestration can lower sensor battery consumption. However, such approach results in centralized architecture with two-way communication, which may increase sensor battery consumption due to constantly being online. When changing monitoring frequency deltas that occur between readings are lost, making this technique hard to evaluate, as there is no data to evaluate its performance.

2) Data compression

Lossless compression on sensors can be achieved in several ways i.e., changes of values can be sent instead of full values. If sensors are collecting data in predefined period, they could exclude timestamp and it can be calculated later from a known period. While all these compressions would save only bytes of transmission data, in radio communication each byte counts. Similar compressions can be performed on inter-gateway communication and clouds as well, where due to huge amount of data few bytes become gigabytes of data.

3) Data filtering

Sensors may filter data and transmit only relevant, while ignoring the rest. Sensors do save on transmission, but they still have to consume energy for data collection. Complex filtering algorithm require processing power and more memory, so it is commonly performed on gateways. Either way, filtering requires high level knowledge of data. Filtering in the cloud results with significant reduction of volume while storing data and velocity when sending data to analysis.

4) Data consolidation

Within a certain area there is commonly a group of sensors, that may or may not communicate with each other. They can agree on transferring bulk of data from only one sensor, or a single sensor that collects different data types can bulk transfer its readings. Problem is that all types must be read at the same interval, excluding dynamic frequency. For long intervals, sensors may not have enough memory or transmission might take too long. Consolidation on gateways can be more feasible due to larger memory and better connection with the cloud. Gateways can also orchestrate connection sharing or share computation load among sensors. In the cloud, data consolidation can take place when storing and partitioning.

5) Data aggregation

Data aggregation presents data in a summarized form. Using this approach, sensor device can aggregate data in some predefined interval and send only aggregated values. It may also compare collected value with aggregated one and send both values if they vary significantly. Same procedure can be implemented on gateways – they can aggregate values from multiple sensors at the same moment in time. Cloud may also aggregate data to reduce required storage capacity.

6) Data correlation

Data correlation is a process of combining data; either with some other data and thus applying the soft(ware) sensors, or combining same type of data between different sources. If sensor collects multiple values, it can corelate them and read or transmit only one, while others can be inferred on a gateway or a server. Gateway receives large volume and variety of data, so it can correlate data form different sensors which are close to each other, or correlate different sensor types. Despite correlation in the cloud requires sending all the data trough whole network, such approach is useful because raw data is stored and available to be analyzed later on.

7) On-site data analytics

On-site data analytics performed on a gateway represents Edge – a concept for collecting and analyzing data on the spot. In such scenarios, data is only forwarded to the cloud for storing or post-analysis, while all the decision making is moved to gateways and sensors. On-site data analytics applied in the cloud would refer to the streaming analytics.

IV. Conclusion

Reducing volume or velocity comes with trade-offs - loss of data, additional processing at the network edge or in the cloud for recreating missing data so data reductions should be balanced not to create even more energy consumption or reduce data value and usage. Creating generic data reduction algorithm is challenging task due to high data variety. The first step towards solution is creating datasets used for evaluating algorithm so methodical research can be applied.

V. References

[1] Internet of Things (IoT) Connected Devices Installed Base Worldwide From 2015 to 2025 (in Billions). Accessed: July 26, 2019. Available: https://www.statista.com/statistics/471264/iotnumber-of-connected-devices-worldwide/

[2] "Cisco Global Cloud Index: Forecast and Methodology", Cisco

[3] P.Derbeko, S.Dolev, E.Gudes, J.D.Ullman,"Concise Essence-Preserving Big Data Representation", IEEE International Conference on Big Data, pp.3662-3665, 2016.

[4] V. Mushunuri, A. Kattepur, H.K. Rath and A. Simha, "Resource Optimization in Fog Enabled IoT Deployments", Second International Conference on Fog and Mobile Edge Computing, pp. 6-13, 2017.

[5] J. Č. Gambiroža, T. Mastelić, "Big Data Challenges and Trade-offs in Energy Efficient Internet of Things systems", SoftCOM, 2018.

Tangible User Interfaces and Programming for Young Children

Lea Dujić Rodić
Department of Mathematics and Physics, FESB,
University of Split,
Split, Croatia
dujic@fesb.hr

Andrina Granić (advisor)
Department of Computer Science, Faculty of Science
University of Split,
Split, Croatia
andrina.granic@pmfst.hr

***Abstract*—The aim of this paper is to present the conceptual design of a Tangible User Interface (TUI) model for young children's acquisition of programming knowledge. The model is based on the knowledge acquired during a conducted systematic review of existing scientific literature.**

Keywords—Tangible User Interfaces, TUI, programming, conceptual model, young children

I. Introduction

Rapid development of technologies in the 21 century has had an enormous impact on children's models, methods, forms and frames of learning. Contemporary children are considered to be digital natives since they were born and raised in the technology driven world. For this reason it is important to engage young children in engineering and technology education at an early age so that they may be successful in STEM (Science, Technology, Engineering and Mathematics) areas. Research has shown that children as young as four can understand the basic concepts of programming [1]. Robotics and computer programming initiatives are growing in popularity amongst researchers to teach young children technology and engineering in a developmentally appropriate way [2].

In this paper we will consider young children to be children aging from 0 to 11 years in accordance to Piaget's stages of cognitive development [3]. Piaget's work has affected researchers and scientists whose work aims to explore and explain ways in which children interact with technologies, particularly in programming. Papert extended Piaget's work resulting in Logo programming language and Lego Mindstorms, a toy construction kit which allowed children to build and program their own robots [4]. Papert's work has been continued in the *Lifelong Kindergarten* research program led by Resnick at MIT aiming to fundamentally rethink what children can and should learn by using computation media [5]. The research resulted in some notable programming languages that are in use today such as *Scratch*, which feature simple syntax with graphically nested loops and conditional statements [6]. The graphical approach allows children to program by dragging and connecting icons on the computer screen requiring the ability to map the on-screen symbolic representation to the actions they produce [7]. Still, children are limited by the use of desktop and laptop computer when it comes to learning how to program, since these environments are not always suitable for them.

Interaction with technology should enhance development of abstract concepts. One form of interaction can be manipulation of objects. Manipulative materials such as wooden blocks or jigsaw puzzles enable children to explore scientific concepts such as number, shape and size [5]. On that point, *Tangible User Interfaces* (*TUIs*) may prove to be bridge between the physical form and digital information since they can remove the clear line between the physical and digital world [8]. Through such interaction children can learn how to program by *building* programs physically thus exceeding the need to learn text-based programming language syntax by making it more direct and less abstract.

II. Objectives and Research

Our main objective is design, development and evaluation of TUI for young children which should engage them in engineering and technology, providing them fundamental skills in computer science such as programming. This work presents a conceptual model of such TUI. The proposed approach will lead to its development and evaluation in formal learning environment with children as design partners.

The conceptual model is based on the results of the conducted systematic literature review focused on a specific topic of tangible interactions and interfaces that support young children learning. Three leading databases were searched: *Web of Science*, *ERIC* and *Scopus*. Set of search terms used to identify related primary studies was specified resulting in the following searching string:

("child*" AND [("tangibl*" OR "touch") AND ("interaction" OR "interface*")])

Only peer-reviewed publications were selected to ensure quality of the studies reviewed. The publications were selected by title and then further analysed through a detailed process of reading abstracts and full texts. Among the selected primary studies, 21 publications that have addressed the use of TUIs in programming for young children were identified.

III. Results

Among the selected publications, 86% considered design, implementation and evaluation of TUIs thus giving us deeper insight into children's preferences as final users. Further analyses, enabled us to outline two aspects that influence the design of considered TUI model: (i) *form of tangible object* as well as (ii) *form of output device*.

Form of tangible objects refers to different forms of input devices through which an interaction with TUIs occurs. In general, there are three forms of tangible objects: *manipulatives*, *tablet* and *tabletop* [9]. However, when considering programming, only manipulatives and tablets have been used with rather dominate role of manipulatives. Precisely, 92% of identified primary studies use manipulatives as form of input device (i.e. 19 out of 21 studies).

When it comes to an output that TUIs provide with regards to programming, we were able to differentiate four forms of output devices: *robots*, *toys*, *graphical device* and one where the input and the output device are the same *manipulative blocks*. Fig. 1 shows ratio of use of above mentioned forms in the selected publications, with robots as the dominant form of output device.

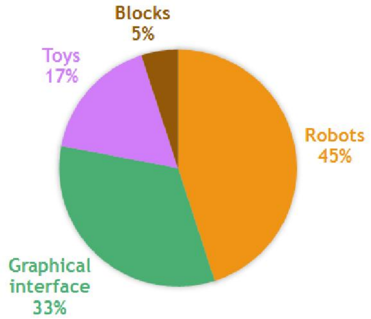


Fig. 1. Frequency of use of outpt devices

A cross-analyses of frequencies of use of the form of the tangible object with respect to the form of the program output is presented in Fig. 2.

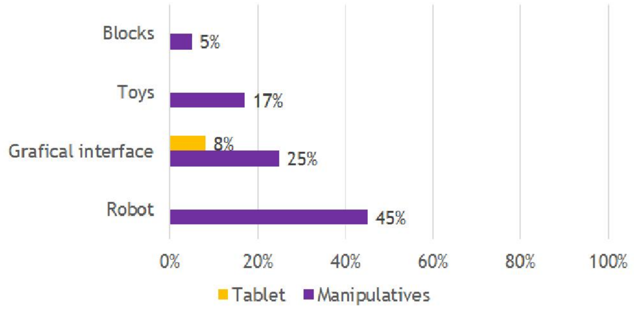


Fig. 2. Frequency of use of output device with respect to input device

IV. CONCEPTUAL DESIGN AND FUTURE WORK

The cross-analyses of results and detailed analyses of selected publications enabled us to propose the design of a conceptual model of a Tangible User Interface model for young children's acquisition of programming knowledge (see Fig. 3). In the identified publications the dominant form of tangible object were manipulatives. The design of tangible object must adequately represent the abstraction. An abstract concept that needs to be comprehended is connected with the form of tangible object trough mapping.

Since manipulation lightness the cognitive load by simplifying abstract concepts and making them more accessible to young children, we propose manipulatives as the appropriate form of input device. In the light of above reasoning, mapping between the abstract concept and the manipulative should be one-on-one, that is one programming concept should be represented with the congruent manipulative. For example, one manipulative block should represent one line of code or a command allowing the child user to link the abstract concept of language syntax with its physical representation. This will enable seamless interaction with the system.

With regards to the form of output device, we suggest a graphical interface. Although from the literature review robots emerged as the most frequent output device, they lack the ability to represent output of more complex programs. Furthermore, they lack the affordance of the graphical interface. Namely, the graphical user interface, whether it is related to laptop screen, tablet screen or a large wall projection, supports successful interactions with the world of physical things and virtual objects.

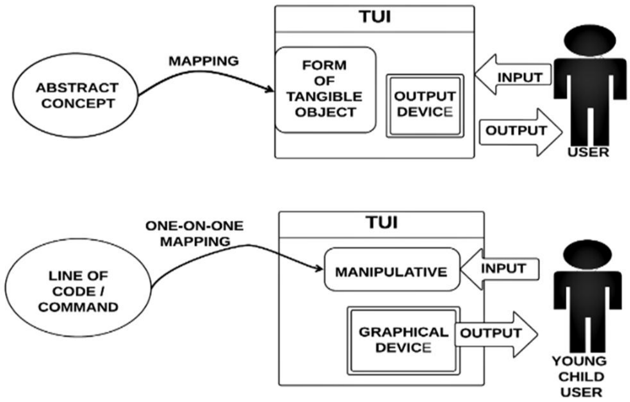


Fig. 3. Conceptual model of TUI for acquisition of programming knowledge for young childern

Our future work will comprise an implementation of the presented conceptual design as well as its evaluation in a formal school environment. We hope to address some of the several open problems that still remain unsolved: (i) a need for a concrete design framework for young children's acquisition of programming knowledge, a framework which will provide designers and researchers with the design guidelines from cognitive theories and pedagogical practices along with (ii) a long term exploration of benefits that TUIs may have for enhancement of young children's programming skills.

REFERENCES

[1] M. U. Bers and M. S. Horn. "Tangible programming in early childhood: Revisiting developmental assumptions through new technologies". High-tech tots: Childhood in a digital world. Information Age Publishing, 2009.

[2] A. Sullivan, M. Elkin, and M. U. Bers. "KIBO robot demo: engaging young children in programming and engineering". In Proceedings of the 14th International Conference on Interaction Design and Children 2015 (IDC '15). ACM, New York, NY, USA, 418-421.

[3] J. Piaget. "Cognitive Development in Children Development and Learning." *Journal of Research in Science Teaching*, 2, 1964, 176-186.

[4] T. S. McNerney. "From turtles to Tangible Programming Bricks: explorations in physical language design". *Personal Ubiquitous Comput*. 8, 5 (September 2004), 326-337.

[5] M. Resnick. "Technologies for lifelong kindergarten". *Educational Technology Research and Development*, December 1998, Volume 46, Issue 4, pp 43–55.

[6] M. Resnick et al. "Scratch: programming for all". *Commun. ACM* 52, 11 (November 2009), 60-67.

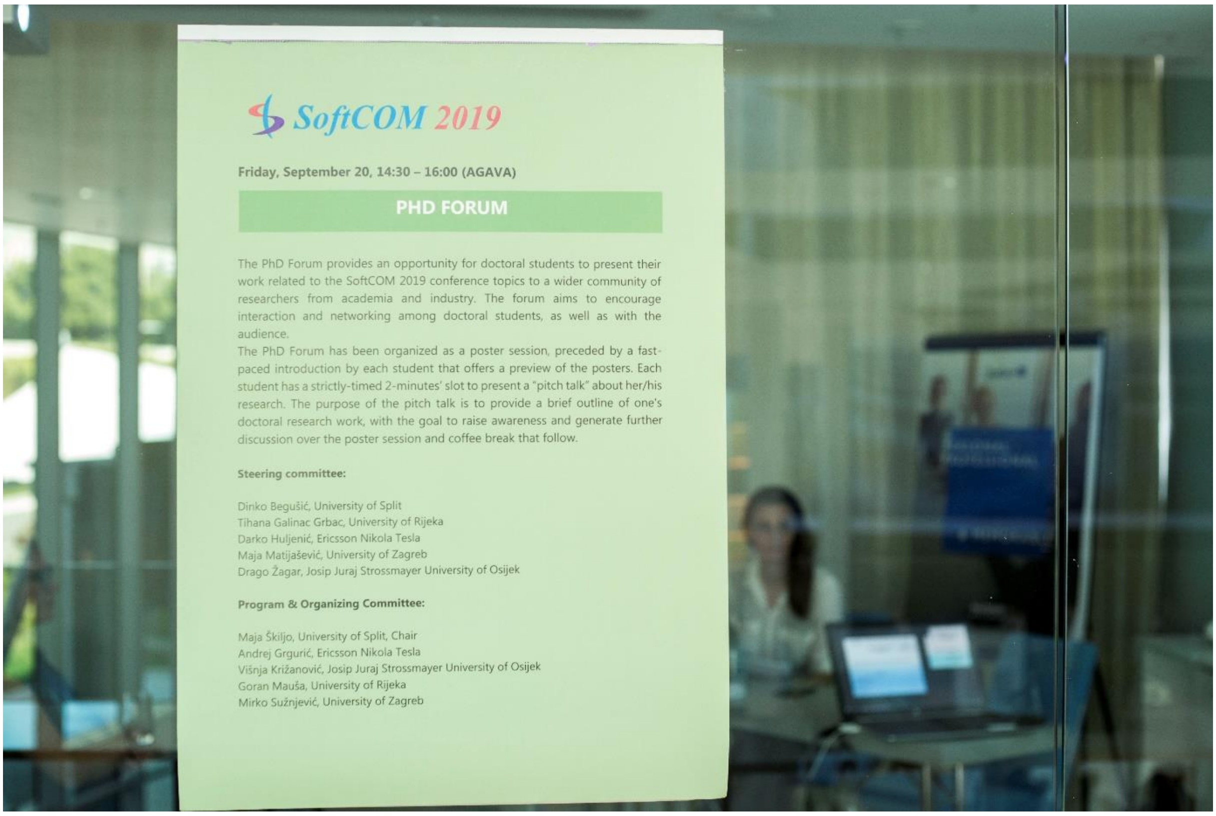
[7] T. Sapounidis and S. Demetriadis. "Tangible versus graphical user interfaces for robot programming: exploring cross-age children's preferences". *Personal Ubiquitous Comput*. 17, 8 (December 2013), 1775-1786.

[8] B. A. Ullmer. "Tangible Interfaces for Manipulating Aggregates of Digital Information". Ph.D. Dissertation. Massachusetts Institute of Technology (2002), Cambridge, MA, USA.

[9] L. Dujić Rodić and A. Granić. "Tangible User Interfaces for Enhancement of Young Children's Mathematical Problem Solving and Reasoning: A Preliminary Review of Relevant Literature". Proceedings of the 29th Central European Conference on Information and Intelligent Systems. V. Strahonja; V. Kirinić (ed.). Varaždin: Faculty of Organization and Informatics, University of Zagreb, 2018. pp. 77-84.

Author index

Photos from the PhD Forum



Welcome and opening address

Welcome and opening address

Pitch talk presentations

Pitch talk presentations

Pitch talk presentations

Pitch talk presentations

Pitch talk presentations

Pitch talk presentations

Pitch talk presentations

Poster session

Poster session

Poster session

Poster session

Poster session

Poster session

Awards ceremony

Awards ceremony