David C. Wyld
Jan Zizka (Eds)

Computer Science & Information Technology

Fifth International Conference on Information Technology Convergence and Services (ITCS 2016)
Sydney, Australia, February 06~07, 2016

AIRCC Publishing Corporation

Volume Editors

David C. Wyld,
Southeastern Louisiana University, USA
E-mail: David.Wyld@selu.edu

Jan Zizka,
Mendel University in Brno, Czech Republic
E-mail: zizka.jan@gmail.com

ISSN: 2231 - 5403
ISBN: 978-1-921987-48-9
DOI : 10.5121/csit.2016.60301 - 10.5121/csit.2016.60307

Typesetting: Camera-ready by author, data conversion by NnN Net Solutions Private Ltd., Chennai, India

Preface

The Fifth International Conference on Information Technology Convergence and Services (ITCS 2016) was held in Sydney, Australia, during February 06~07, 2016. The Fifth International Conference on Software Engineering and Applications (JSE 2016), The Fifth International Conference on Signal & Image Processing (SIP 2016), The Third International Conference on Artificial Intelligence & Applications (ARIA 2016) and The Fifth International Conference on Natural language Processing (NLP 2016) were collocated with the ITCS-2016. The conferences attracted many local and international delegates, presenting a balanced mixture of intellect from the East and from the West.

The goal of this conference series is to bring together researchers and practitioners from academia and industry to focus on understanding computer science and information technology and to establish new collaborations in these areas. Authors are invited to contribute to the conference by submitting articles that illustrate research results, projects, survey work and industrial experiences describing significant advances in all areas of computer science and information technology.

The ITCS-2016, JSE-2016, SIP-2016, ARIA-2016, NLP-2016 Committees rigorously invited submissions for many months from researchers, scientists, engineers, students and practitioners related to the relevant themes and tracks of the workshop. This effort guaranteed submissions from an unparalleled number of internationally recognized top-level researchers. All the submissions underwent a strenuous peer review process which comprised expert reviewers. These reviewers were selected from a talented pool of Technical Committee members and external reviewers on the basis of their expertise. The papers were then reviewed based on their contributions, technical content, originality and clarity. The entire process, which includes the submission, review and acceptance processes, was done electronically. All these efforts undertaken by the Organizing and Technical Committees led to an exciting, rich and a high quality technical conference program, which featured high-impact presentations for all attendees to enjoy, appreciate and expand their expertise in the latest developments in computer network and communications research.

In closing, ITCS-2016, JSE-2016, SIP-2016, ARIA-2016, NLP-2016 brought together researchers, scientists, engineers, students and practitioners to exchange and share their experiences, new ideas and research results in all aspects of the main workshop themes and tracks, and to discuss the practical challenges encountered and the solutions adopted. The book is organized as a collection of papers from the ITCS-2016, JSE-2016, SIP-2016, ARIA-2016, NLP-2016.

We would like to thank the General and Program Chairs, organization staff, the members of the Technical Program Committees and external reviewers for their excellent and tireless work. We sincerely wish that all attendees benefited scientifically from the conference and wish them every success in their research. It is the humble wish of the conference organizers that the professional dialogue among the researchers, scientists, engineers, students and educators continues beyond the event and that the friendships and collaborations forged will linger and prosper for many years to come.

David C. Wyld
Jan Zizka

Organization

General Chair

Jan Zizka	Mendel University in Brno, Czech Republic
Dhinaharan Nagamalai	Wireilla Net Solutions PTY LTD, Australia

Program Committee Members

Abd El-Aziz Ahmed	Cairo University, Egypt
Abdolreza Hatamlou	Islamic Azad University, Iran
Abe Zeid	Northeastern University, USA
Ahmed Hussein Aliwy	University of Kufa, Iraq
Ahmed Taisser	Cairo University, Egypt
Aiden B. Lee	University of La Jolla, USA
Akira Otsuki	Nihon University, Japan
Alejandro Regalado Mendez	Universidad del Mar, Mexico
Ali Abid D. Al-Zuky	Mustansiriyah University Baghdad, Iraq
Ali Elkateeb	University of Michigan-Dearborn, USA
ALI ZAART	Beirut Arab University,Lebanon
Alireza Sahab	Lahijan Branch Islamic Azad University, Iran
Alvin Lim	Auburn University, USA
Amar Faiz bin Zainal Abidin	Universiti Teknologi MARA, Malaysia
Amol D Mali	Univ. of Wisconsin-Milwaukee, USA
Anja Richert	RWTH Aachen University, Germany
Ankit Chaudhary	Truman State University, USA
Apai	Universiti Malaysia Perlis, Malaysia
Arifa Ferdousi	Varedndra University, Bangladesh
Atif Farid Mohammad	University of North Carolina,Charlotte
Ayad N.M.A	Atomic Energy authority, Egypt
Ayad Salhieh	Australian College of Kuwait, Kuwait
Azween Bin Abdullah	Universiti Teknologi Petronas, Malaysia
Bai Li	Woodside Energy Ltd, Australia
Barbaros Preveze	Cankaya University, Turkey
Bouix Emmanuel	iKlax Media, France.
Braham Barkat	The petroleum Institute, Saudi Arabia
Chandan Kumar Karmakar	University of Melbourne, Australia
Cheng fang	Zhejiang University, China
Chih-Lin Hu	National Central University, Taiwan
Chin-Chih Chang	Chung-Hua University, Taiwan
Chiranjib Sur	University of Florida, USA
Christian Esposito	National Research Council, Italy
Dac-Nhuong Le	Haiphong University, Vietnam
Dalila Guessoum	Saad Dahleb University, Algeria
Danda B.Rawat	Georgia Southern University, USA
David B. Bracewell	General Electric Global Research, USA

Technically Sponsored by

Networks & Communications Community (NCC)

Computer Science & Information Technology Community (CSITC)

Digital Signal & Image Processing Community (DSIPC)

Organized By

Academy & Industry Research Collaboration Center (AIRCC)

TABLE OF CONTENTS

The Fifth International Conference on Information Technology Convergence and Services (ITCS 2016)

The Fifth International Conference on Software Engineering and Applications (JSE 2016)

The Fifth International Conference on Signal & Image Processing (SIP 2016)

The Third International Conference on Artificial Intelligence & Applications (ARIA 2016)

The Fifth International Conference on Natural language Processing (NLP 2016)

COMPRESSED SENSING OF EXCITATORY POSTSYNAPTIC POTENTIAL BIO-SIGNALS

Hyejin An and Hyun-Chool Shin

Department of Electronic Engineering,
Soongsil University, Seoul, Republic of Korea
{ahj, shinhc}@ssu.ac.kr

ABSTRACT

To reduce the size of the biosignal data is important because a huge amount of data is made by various experiments. In the paper, we efficiently compress the excitatory postsynaptic potentials (EPSPs) which is one of the biosignal types. To the best of authors' knowledge, EPSPs compression has not been studied yet. The EPSP signal has a feature that the adjacent signals in single excitatory postsynaptic potential have similar characteristics. Using this feature, we propose a method which removes temporal redundancy and statistical redundancy of EPSPs. The compressed and reconstructed EPSPs are similar to the original signal without the loss of analytic information.

KEYWORDS

Excitatory Postsynaptic Potentials, Compressed Sensing, Long-term Potentiation, Long-term Depression, Data Compression

1. INTRODUCTION

Biosignals contain highly important information with which researchers in neuroscience and neural engineering fields attempt to disclose the neurological mechanisms of living organisms by measuring and analyzing various biosignals. Biosignals can be recorded using a variety of forms such as electroencephalogram (EEG), electrocorticography (ECoG), neural spike train, electrocardiography (ECG), electromyography (EMG), and excitatory postsynaptic potentials (EPSPs). To acquire more information about living organisms, researchers are recording biosignals by increasing the number of sensors, the measurement time, the number of experiment samples, and the measurement frequency. As a result, storage space, analysis time and data bandwidth for biosignals are being increasing as well. These problems can be solved with integrated memory chips, fast data processors and data communication technologies, but a more efficient method is to use data compression technology with which the biosignal data size is reduced while minimizing information loss.

Many data compression researches have been conducted to achieve the above-mentioned goals in recent years [1-5]. Sriraam and Eswaran [1] compressed EEG obtained from the scalp. Salman et al. [2] restored EMG using convex optimization to reduce energy consumption. Allstot et al. [3]

David C. Wyld et al. (Eds) : ITCS, JSE, SIP, ARIA, NLP - 2016
pp. 01–07, 2016. © CS & IT-CSCP 2016 DOI : 10.5121/csit.2016.60301

conducted a study similar to [2] using ECG signals. In addition, Hegde et al. [4] proposed a compression method by creating a neuronal spike train model.

There are other studies about the compression of biosignals as well, but most studies have concentrated on the development of signal compression technology for EEG, ECoG, and neural spike trains acquired from in-vivo experiments. Although in-vitro or ex-vivo experiments are important in studying the neurological mechanism and biosignals recorded from these experiments require a large storage space [6-9], no study on the compression of EPSPs has been done. In general, EPSPs create data of 16 Kbits (10,000 Hz × 0.1 sec. × 16 bits) in a single sensor with a single measurement. Assuming that 400 repetitive experiments using four to eight measurement sensors are conducted, the data recorded in a single experiment is approximately 26~51 Mbits, which is very large. In addition, experiments related to EPSPs measure biosignals through a large number of experiment samples with various parameters. As a result, actual data amounts could increase further.

This paper proposes a method that can compress EPSPs signals within a range that has no effect on the analysis of the EPSPs. The EPSPs signal occurs in synapses where two neurons are close to each other and it is created because the membrane potential of neurons behind the synapse increases due to neurotransmitters secreted from neurons in front of the synapse. Thus, changes in the EPSPs signal over time are slow and not large compared to neural spike trains or EEG. Inspired by this fact, EPSPs are compressed by eliminating information redundancy between adjacent signals.

This paper deals with the EPSPs signal which has not been tried for compression. The proposed method is suitable for the features of the EPSPs signal and includes uncomplicated arithmetic operations.

2. MATERIALS AND METHODS

2.1. Materials

EPSPs were measured via neurons of the hippocampus, which plays an important role in learning and memory. EPSPs signals were measured when hippocampus slices obtained from mice were stimulated. Normally, a single mouse has four to six slices, and a single slice can be a sensing channel. For both long-term potentiation (LTP) and long-term depression (LTD), experiments were conducted for 100 minutes, and the data for 80 minutes were used to draw a slope graph. A single EPSPs signal was measured every 15 seconds. The LTP and LTD experiments were conducted 381 times and 435 times, respectively. During the experiment, slices were immersed in an ACSF solution at approximately 28℃ while bubbling was maintained as a mixed gas of 95% oxygen and 5% carbon dioxide was introduced at a rate of 4 ml/min for pH adjustment. The LTP was induced by providing theta burst stimulation (TBS) for one minute 40 minutes after the experiment started while LTD was induced by providing single-pulse low-frequency stimulation (SP-LFS) six times per minute for 15 minutes.

2.2. Methods

To compress the signals, it is necessary to eliminate temporal redundancy and statistical redundancy. In this paper, two signal-processing filters were used to eliminate temporal redundancy. The first filter was a low-pass filter (LPF), which passes only frequencies below 300

Hz, and the second filter was a first-order difference filter that calculated a difference between adjacent EPSPs signals. Since adjacent signals in single EPSPs change slowly and have similar characteristics, EPSPs after filtering become sparse.

Once temporal redundancy is eliminated, a compressed sensing technique was applied to eliminate statistical redundancy in spare EPSPs [10]. Assuming that the signal to be compressed is $\mathbf{x} \in \mathbb{R}^{N\times 1}$ and the compressed signal is $\mathbf{y} \in \mathbb{R}^{M\times 1}$, then it can be expressed as follows:

$$\mathbf{y} = \mathbf{\Phi x} \tag{1}$$

where $\mathbf{\Phi} \in \mathbb{R}^{M\times N}$ is called a sensing matrix, which uses a binary matrix consisting of 0 and 1 in general. The most important point in compressed sensing is to express a signal to be sparse. To do this, conventionally, new sparse vector $\mathbf{z} \in \mathbb{R}^{N\times 1}$ and dictionary matrix $\mathbf{D} \in \mathbb{R}^{N\times N}$ are introduced so that **x** is expressed with **z** and **D**, thereby generating sparse **x**.. A matrix used in **D** is an inverse discrete cosine transform (DCT) or inverse discrete wavelet transform (DWT), in which **z** uses DCT or DWT coefficients. This procedure is quite complex. However, in this paper, we easily obtain sparse **x** by eliminating temporal redundancy without using conventional method. Since EPSPs show slow transition in time, the spare **x** is obtained by filtering EPSPs with a first-order difference filter.

Block sparse Bayesian learning-bound optimization (BSBL-BO) [11] was applied to restore $\hat{\mathbf{x}}$ from the compressed EPSPs signal **y**. The compression ratio (CR) was calculated using

$$\text{CR} = \frac{\text{the number of bits in } \mathbf{x}}{\text{the number of bits in } \mathbf{y}}. \tag{2}$$

A compression ratio of LTP and LTD has three values: 2, 3, and 4. The higher the compression ratio, the more often compression occurs.

To compare the similarity between restored and original signals, the normalized mean square error (NMSE) [12] was used.

$$\text{NMSE} = \frac{\|\mathbf{x}-\hat{\mathbf{x}}\|_2^2}{\|\mathbf{x}\|_2^2} \tag{3}$$

where x is an original EPSPs signal and $\hat{\mathbf{x}}$ is a restored EPSPs signal. The closer the NMSE value is to 0, the closer it is to the original signal.

3. RESULTS

Figure 1 shows the comparison of the original single EPSPs signal, **x** and the restored single EPSPs signal, $\hat{\mathbf{x}}$ for LTP and LTD. In the figure, a compression ratio was set to 2; a dotted line represents the original EPSPs signal while a solid line represents the recovered EPSPs signal. Figure 2 shows the comparison between the original EPSPs and the restored signals, depicting NMSE compared over various compression ratios. For LTP, depending on compression ratios 2, 3, and 4, the mean and standard deviation (STD) of the NMSE were 0.020 ± 0.014, 0.15 ± 0.032 and 0.35 ± 0.060, respectively. For LTD, depending on the compression ratio, the mean and STD were 0.084 ± 0.021, 0.18 ± 0.021 and 0.46 ± 0.026, respectively. As the compression ratio becomes lower, which means more compression, the NMSE increases, which means the restoration performance decreases.

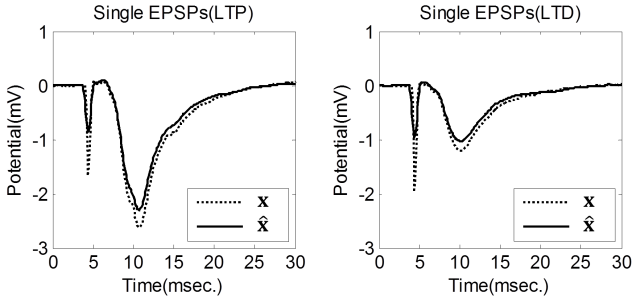


Figure 1. Comparison of original signal and recovered signal. They are the single EPSPs signals generated after the stimulation (CR = 2).

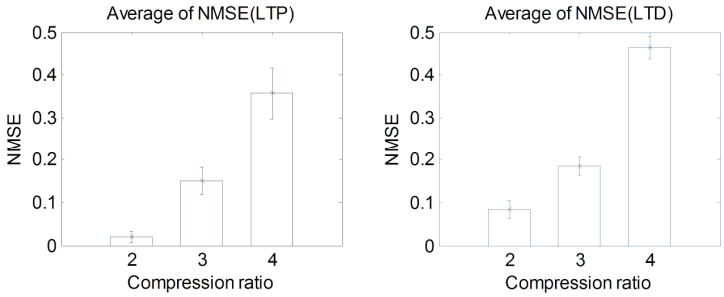


Figure 2. NMSE mean and STD comparison according to compression ratio. NMSE compares the difference between the original signal and the recovered signal.

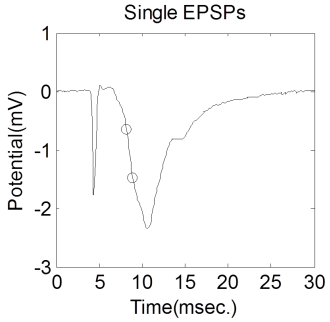
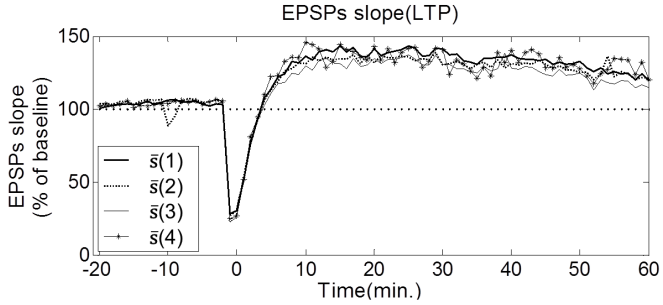


Figure 3. Single EPSPs signal in which a section slope is calculated. The two points in the middle are a 1/3~2/3 section designated for the slope calculation.



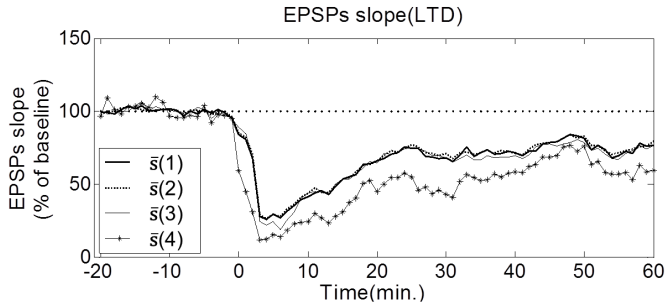


Figure 4. Comparison of changes in slopes calculated from original single EPSPs signals and recovered single EPSPs signals. On the time axis in the figure, 0 refers to a time when the stimulation was given and the slope change over time from 20 minutes prior to the stimulation and 60 minutes after the stimulation, which was a total of 80 minutes, shown as a percentage. Each point represents a slope mean of EPSPs signals calculated every one minute. The values in the brackets refer to compression ratios. $\bar{\mathbf{s}}(1)$ means the slope without compression.

To analyze EPSPs signals in general, the slope in temporal EPSPs is compared before and after the stimulation is given. For the first step in the comparison, a slope was calculated as a section of 1/3~2/3 set in the single EPSPs signal slope, as shown in Figure 3. Each single EPSPs signal has its own single calculated slope. A method of calculating a slope is to divide changes in potential in the designated section by time. As a next step, a mean is calculated every one minute using a single EPSPs signal slope in the section from 20 minutes prior to stimulation to 60 minutes after stimulation. The baseline that is a criterion for comparison before and after stimulation is a slope mean of the EPSPs signal calculated from 20 minutes prior to stimulation until the stimulation was given; this value was again converted into 100% proportion. All slope means obtained every minute are normalized using the above mentioned criterion, and the slope, $\bar{s}(t)$ calculated by

$$\bar{s}(t) = \frac{s(t)}{\frac{1}{N}\sum s_b(t)} \times 100 \tag{4}$$

where $s(t)$ is the EPSPs slope whose means are obtained every one minute. $s_b(t)$ is a part of $s(t)$ that corresponds to the baseline section which is for 20 minutes prior to stimulation. $\frac{1}{N}\sum s_b(t)$ is the mean of $s_b(t)$. This $\bar{s}(t)$ is the normalized slope value of $s(t)$.

Figure 4 shows the normalized slope, $\bar{s}(t)$ from the original EPSPs and the recovered ones. For the low compression ratio, the figure shows that the slope graph is very similar to the slope of the original signal. When the LTD is restored with a compression ratio of 4, the slope value seems to change more than the slope of the original signal, but it did not affect the overall slope trend significantly.

4. CONCLUSION

EPSPs signals have been studied by many researchers in recent years and require a large storage capacity whenever experiments are conducted. In addition, researchers set various parameters to conduct experiments. Because of this, it is very important to reduce the data size of EPSPs signals. However, unlike other biosignals, no studies have been done on EPSPs compression. In this study, an efficient compression method was proposed without any problems in analyzing EPSPs. Using the proposed method, EPSPs signals can be stored and analyzed with a smaller amount of data than that of the original signals.

ACKNOWLEDGEMENTS

This research was supported by the Original Technology Research Program for Brain Science through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (NRF-2011-0019212).

REFERENCES

[1] Sriraam, N. & Eswaran, C., (2004) "EEG signal compression using radial basis neural networks", Journal of Mechanics in Medicine and Biology, Vol. 4, No. 2, pp143-152.

[2] Salman, A., Allstot, E.G., Chen, A.Y., Dixon, A.M.R., Gangopadhyay, D. & Allstot, D.J., (2011) "Compressive sampling of EMG bio-signals", IEEE International Symposium on Circuits and Systems, Rio de Janeiro, Brazil, pp2095-2098.

[3] Allstot, E.G., Chen, A.Y., Dixon, A.M.R., Gangopadhyay, D. & Allsot, D.J., (2010) "Compressive sampling of ECG bio-signals: Quantization noise and sparsity considerations", IEEE Conference on Biomedical Circuits and Systems Conference, Paphos, Cyprus, pp41-44.

[4] Hegde, C., Duarte, M.F. & Cevher, V., (2009) "Compressive sensing recovery of spike trains using a structured sparsity model", Signal Processing with Adaptive Sparse Structured Representations, Saint Malo, France.

[5] Sharma, L.N., Dandapat, S. & Mahanta, A., (2012) "Multichannel ECG data compression based on multiscale principal component analysis", IEEE Transactions on Information Technology in Biomedicine, Vol. 16, No. 4, pp730-736.

[6] Womble, M.D., Andrew, J.A. & Crook, J.J., (2002) "17β-Estradiol reduces excitatory postsynaptic potential (EPSP) amplitude in rat basolateral amygdala neurons", Neuroscience letters, Vol. 331, No. 2, pp83-86.

[7] Choi, T.-Y., Kwon, J.E., Durrance, E.S., Jo, S.-H., Choi, S.-Y. & Kim, K.-T., (2014) "Melatonin inhibits voltage-sensitive Ca2+ channel-mediated neurotransmitter release", Brain Research, Vol. 1557, pp34-42.

[8] Seo, J., Kim, K., Jang, S., Han, S., Choi, S.-Y., & Kim, E., (2012) "Regulation of hippocampal long-term potentiation and long-term depression by diacylglycerol kinase ζ", Hippocampus, Vol. 22, No. 5, pp1018-1026.

[9] Bliss, T.V.P. & Cooke, S.F., (2011) "Long-term potentiation and long-term depression: a clinical perspective", Clinics, Vol. 66, pp3-17.

[10] Candès, E.J. & Wakin, M.B., (2008) "An introduction to compressive sampling", IEEE Transactions on Signal Processing Magazine, Vol. 25, No. 2, pp21-30.

[11] Zhang, Z. & Rao, B.D., (2013) "Extension of SBL algorithms for the recovery of block sparse signals with intra-block correlation", IEEE Transactions on Signal Processing, Vol. 61, No.8, pp2009-2015.

[12] Zhang, Z., Jung, T.-P., Makeig, S. & Rao, B.D., (2013) "Compressed sensing for energy-efficient wireless telemonitoring of noninvasive fetal ECG via block sparse Bayesian learning", IEEE Transactions on Biomedical Engineering, Vol. 60, No. 2, pp300-309.

AUTHORS

Hyejin An was born in Seoul, Korea, in 1988. She received the B.Sc. degrees in electronic engineering from Soongsil University, Korea, in 2012, where she is currently working toward the Ph.D. degree in signal processing in the Department of Electronic Engineering. Her research interests include neural and intelligent signal processing focusing on brain and bio-signals.

Hyun-Chool Shin (M'04) was born in Seoul, Korea, on June 21, 1974. He received the B.Sc., M.Sc., and Ph.D. degrees in electronic and electrical engineering from Pohang University of Science and Technology (POSTECH), Korea, in 1997, 1999, and 2004 respectively. From 2004 to 2007, he has been a postdoctoral researcher at the Department of Biomedical Engineering, Johns Hopkins University, School of Medicine. Since 2007, he is currently an associate professor of electronic engineering at the Soongsil University. His research interests include neural and intelligent signal processing focusing on brain and bio-signals and radar array signals.

INTENTIONAL BLANK

DEVELOP QUALITY CHARACTERISTICS BASED QUALITY EVALUATION PROCESS FOR READY TO USE SOFTWARE PRODUCTS

Daiju Kato[1] and Hiroshi Ishikawa[2]

[1]WingArc1st Inc., Tokyo, Japan
kato.d@wingarc.com
[2]Graduate School of System Design,
Tokyo Metropolitan University, Tokyo, Japan
ishikawa-hiroshi@tmu.ac.jp

ABSTRACT

The users who use ready to use software products had better getting the products' quality information with classified by some kind of global standard metrics or technique for their evaluation. But many of those software products' companies don't provide the quality information because of the products are developed by their own development and evaluation process. But those users want to get quality information to reduce evaluation cost. Therefore, we develop our quality evaluation process with using quality characteristics of software external quality model on ISO/IEC 9126-1 for our software products. This evaluation process has feature of unsynchronized quality evaluation process toward development process. Also, the process starts from definition of classified quality requirement based on quality sub-characteristics at test planning process to quality analysis at test completion process under software development cycle. We provide precise our products quality data for our users through this evaluation process and the process might be reduced cost of our customers' evaluation.

KEYWORDS

Evaluation process, Quality model, Quality analysis, Software development cycle, Ready to Use Software, ISO/IEC 9126, ISO/IEC 25051

1. INTRODUCTION

Many companies develop their own corporate IT systems to combine or be based on ready to use software products. They evaluate those software products to fit their systems and to determine those quality by themselves. This evaluation is very important for suitable and reliable to their systems, but however, many companies can't have enough cost and time for the evaluation. This concern becomes industry wide issue and many countries have established products certification programs based on ISO/IEC 25051[1]. For example, The Computer Software Association of Japan (CSAJ) has started PSQ Certification System [2]. According to CSAJ web site, this certification has below responsibility.

David C. Wyld et al. (Eds) : ITCS, JSE, SIP, ARIA, NLP - 2016
pp. 09–21, 2016. © CS & IT-CSCP 2016 DOI : 10.5121/csit.2016.60302

PSQ Certification System is the certification program conforming to international standards, which confirms and certifies that the documents and the functions of the software product are consistent. The system therefore evaluates descriptions including the product description (catalogue), user documentation (e.g. manuals etc.), and test along the document to ensure compliance using test report. Certification will then be issued to the authorized product and will receive the certification logo.

This kind of certification programs give positive effect to many users who plan to use certified software products. But the users need some evaluation cost to check whether the products have right quality and fit their corporate systems because the certified programs don't suggest the products' quality and what kinds of evaluation process are used. So, we believe our customers might be reduce their evaluation cost if they get quality data to fit our customers' request.

Most of quality data as metrics are related to development process. For example, bugs ratio, code review ratio, pass rate of test cases and test coverages are internal data under development projects and it is very difficult to use key value indicator unless same development process. Those data are not suitable to standard scale for quality analysis without explanation of software development cycle.

We need to solve this issue to provide useful quality information for our customers. So, we had started to classify our products' quality using by quality characteristics at ISO/IEC 9126-1 [3]. But we needed to spend much cost to analyse our quality after end of projects. Also, classified quality by quality characteristics doesn't fit basic development process, like V-model or agile model. Therefore, we develop our own evaluation process using by quality characteristics. This process needs unsynchronized toward development process and covers from test planning process to completion process within software development cycle. Also, we provide detail test reports analysed by quality characteristics.

2. BACKGROUND AND PROBLEM

Our company has developed and maintained several software products. But we don't have companies' standard development process and each development teams decide development process to fit for their development style. So, each QA teams needed to create testing process to fit each products. Unless using same development style, each QA teams has accumulated quality metrics data, as like bugs ratio or test cases density per source codes, to judge for product shipment. Those quality data depended on each development process and we couldn't compare our products' reliability correctly with those quality data. Also, we couldn't provide suitable quality information for our customers to evaluate our products. We knew quality model using quality characteristics at ISO/IEC 9126-1 was some of key solution to classify software quality. But it was very difficult to classify result of our test case in each test types to fit quality characteristics. Therefore, we have started developing evaluation process to use quality characteristics.

3. DEVELOPING QUALITY EVALUATION PROCESS

After researching our evaluation process to define by quality characteristics, we realize that we need to divide our evaluation process from development process. But complete splitting out from development process makes our test cost raised up because of rework due to bugs. We have deal

milestone with synchronized points between development process and evaluation process to solve for this rework. We decide to use quality characteristics for our key value indicator.

Our evaluation process has three process, test planning, test management process for monitoring and controlling verification and test completion process. Verification period has several test levels which we call test stages. We define test planning period as alpha stage, test verification period as beta stages and test completion period as RC stage and those stages are timely continuous, “Figure 1”.

But beta stages as verification period doesn’t need to keep turn like water hole model. Each beta stage has started from acceptance test for entrance criteria of test levels and we can decide whether we start the test level or not.

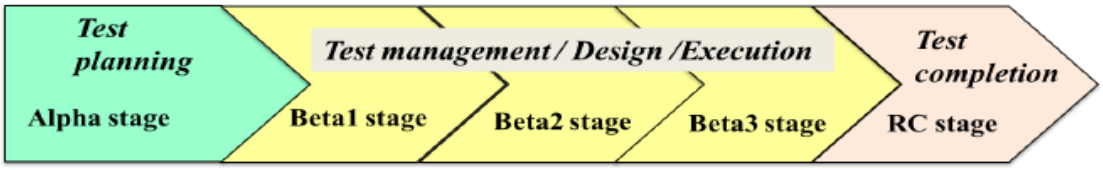


Figure 1. Brief of evaluation process

QA manager needs to control cost of testing for functionality characteristics doesn’t exceed 70% at total cost of all testing when QA manager define test types. This reason is that verification is not bias as functional test.

There is two way to fit test types to each test level. One way is that QA managers list up test types at first and looks for dependency of each test type. After defining the dependency, QA managers are mapping each test type to suitable test level referred by milestone, defined synchronized point to development process. The other way is that QA manager

3.1. Test planning process

Test planning process is developing master test plan (MTP) and level test plan (LTP). MTP and LTP are finally authorized by stakeholders. QA managers write MTP with some of test bases like product requirement plan, marketing requirement documents and product vision documents. We use test document template on IEEE 829 [4]. QA managers determine test strategy for their project and they think of test approach.

After test strategy is established, QA managers focus on definition of quality for shipment of our target software product. The quality is classified by each quality sub-characteristics, “Table.1”.

QA managers define test level to accomplish the definition of quality. Each test level also has quality definition classified by quality sub-characteristics and each test level has several test types to fit the definition and results of each test types becomes the evidence for each target quality classified by quality sub-characteristics, “Figure 2”.

QA manager needs to control cost of testing for functionality characteristics doesn’t exceed 70% at total cost of all testing when QA manager define test types. This reason is that verification is not bias as functional test.

Table 1. Example of definition of quality requirement

Characteristics	Sub-characteristics	Target quality
Functionality	*Suitability*	• The development requirement of new features have been verified. • Migration from past version are possible and compatibility is collateral.
	Accuracy	• The results of new functional requirements are correct and have been verified. • Each product definition files developed by past version are worked properly. • All functions under English and Chinese environment has same quality as Japanese environment. • The result of scenario test has no problem under the estimated users operations.
	Interoperability	• The results of combination of other WingArc products have no problem.
	Security	• No vulnerabilities is detected including used OSS components and web interface..
Reliability	*Maturity*	• Each test level has analysis action for test coverage and turn-around time to fix bugs.
		• Each test level has quality improvement action derived from quality analysis of previous test level. • All bugs are verified at RC stage.
	Fault Tolerance	• Operations has perfect continuous even though one node is stopped under clustering environment. • Migration program is keeping on running under some of definition files have some errors.
	Recoverability	• Past version of definition files can be saved even though the files have some errors.
Usability	*Understand-ability*	• There is no difference between operation steps in users' manual and actual operation. • Localized UI provide same quality as Japanese environment.
	Learnability	• Sample files and tutorial manual can be used self-study materials.
	Operability	• Scenario test has behavior and procedures which are aware of target customers' operations.
Efficiency	*Time Behavior*	• Performance is less or equal to maximum 3% than past version. • Concurrent multi-access test is performed with no error. • There is no high CPU load or I/O load condition under some functional operations.
	Resource Utilization	• There is no memory leak under usual operations. • Memory and I/O resources are effectively used.

Maintainability	*Analyzability*	• Functionality of debug logs are implemented and the logs can be used for error analyzing.
	Changeability	• External API is easily extended.
	Stability	• The result of Load test doesn't have no error.
	Testability	• Testing layers are prepared between client and server communications.
Portability	*Adaptability*	• Product is perfectly running under supported environment.
	Installability	• Product is easily installed under supported environment.
	Co-Existence	• Product can be used with antivirus software and backup software.
	replaceability	• Product is easily replaced from past version.

There is two way to fit test types to each test level. One way is that QA managers list up test types at first and looks for dependency of each test type. After defining the dependency, QA managers are mapping each test type to suitable test level referred by milestone, defined synchronized point to development process. The other way is that QA managers define quality of each test level referred by milestone and look for the test types to meet. We choose the way whether development process is clear or not. After mapping of test types to test level, QA managers define quality characteristics for each test level like "Table 2".

QA managers also write organization of test teams, test bases, verification schedule, product and project risk, training plan, policy of criteria, policy of test development and used metrics from development process, to MTP referred by IEEE 829 template.

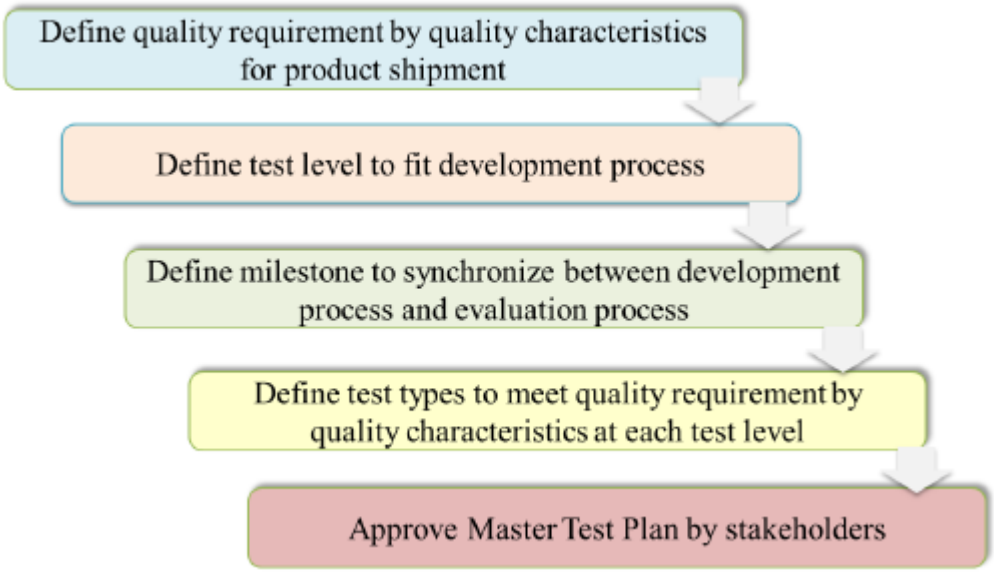


Figure 1. Outline of MTP developing process

Table 2. Mapping of quality characteristics to test levels

Stage	Quality characteristics					
	Functionality	Reliability	Usability	Efficiency	Maintain-ability	Portability
Beta1	◎	○	○	○		
Beta2	◎	○	◎	○	○	
Beta3	◎	◎	◎	◎	○	○
RC	◎	◎	◎	◎	○	○

After writing up MTP, QA managers start writing LTP. Basically LTP in our evaluation process is only acceptance test plan for each test level. Policy of criteria for acceptance test is defined on MTP. This acceptance test is entrance criteria whether we can start the test level or not.

During QA managers write MTP and LTP, QA teams start test designs and analyse software product by test types. QA managers manage test designs with test management process.

3.2. Acceptance test for test level

Acceptance test is minimum set of all test types operated in the test level, “Table.3”. QA managers define the acceptance test made from all test types at target test level. We choose less than 10% of test cases from all test types with using stratification method.

Our evaluation process defines period of acceptance test is within one week including all test execution and test analysis. This rule is we don’t spend much time if acceptance test is failed.

According to this acceptance test, QA teams can reduce much rework because QA teams realize the situation of software quality under development project. QA teams can request quality improvement to the development with static testing if result of the acceptance test doesn’t reach the criteria. QA managers can plan and evaluate several times of acceptance test unless passing criteria. QA managers can change criteria policy for the acceptance test under negotiation with development teams.

By acceptance test as this entrance criteria for test level, QA teams can execute several beta stages at same time even though QA teams finish prior beta stages. For example, our QA teams can execute beta1 stages and beta2 stages simultaneously unless both acceptance test is passed. This acceptance test as entrance criteria helps to reduce test cost.

Table 3. Example of acceptance test

User Type	Test Types	Guarantee of sub-characteristics
Beta2 Stage		
New	New requirement test	Suitability Accuracy Interoperability Security Fault Tolerance Recoverability Resource Utilization
Upgrade	Compatibility test	Suitability Accuracy Interoperability Security Maturity Fault Tolerance Recoverability Resource Utilization
New/Upgrade	Performance test	Time Behavior Resource Utilization
Beta3 Stage		
New/Upgrade	Mobile function test	Suitability Accuracy Interoperability
New/Upgrade	Secinario test	Suitability
		Accuracy Interoperability Security Maturity Fault Tolerance Recoverability Understandability Learnability Operability Stability
New/Upgrade	Quality inprovement test at beta2	Suitability Accuracy Security Maturity Stability
New/Upgrade	Performance and load test	Time Behavior Resource Utilization Stability
New/Upgrade	Platform test	Suitability Accuracy Interoperability Operability

3.3. Test management process as monitoring and controlling verification

After acceptance test is passed, QA teams execute all test types planned in the test level. Each test type has targeted quality sub-characteristics for guarantee. For example, performance test ensures Efficiency-Time Behaviour.

In test designing, QA teams develop test viewpoint for each test types. Test viewpoint is bird's-eye view representation to analyse test. We develop test viewpoint by design base or requirement base and we define quality sub-characteristics to all test viewpoint. So, all test cases derived from test viewpoint also have relationship with some quality sub-characteristics.

QA managers check progress of text execution to each test type. Also, they manage bugs founded by those test types. Those bugs can also mapping to quality sub-characteristics because test cases have definition of quality sub-characteristics, "Table 4". QA managers can analyse easily what kind of sub-characteristics is weak.

Table 4. Example of found bugs at beta2 test stage

Characteristics	Sub-Characteristics	Total test cases	Total Bugs
Functionality	Suitability	848	7
	Accuracy	9662	383
	Security	97	2
Reliability	Maturity	38	19
Usability	Understandability	1543	60
Efficiency	Time Behavior	1414	8
	Resource Utilization	1414	8
Portability	Installability	332	6
	Replaceability	1513	55

Basically, definition of quality classified by quality characteristics is qualitative analysis, but, QA managers define pass rate and test density for each test type derived from development process. QA manages can analyse quality with both qualitative analysis and quantitative analysis.

Basically our test management process is no difference as usual test management process even though quality characteristics are key value indicator. QA managers focus on progress of both test design and test execution compared with plan and real progress.

3.4. Testing for 'Quality in Use'

Quality model on ISO/IEC 9126 defines 'Quality in Use', "Figure 3". This quality characteristics needs ready to use software for operation and sustain. Lack of this quality rises total maintenance cost even though internal software quality is mature. But testing for 'Quality in Use' is difficult for ready to use software. One reason is our software customers use our products with other software and their own system, therefore, we can't estimate benefit of our product easily. Another reason is characteristics at 'Quality in Use' is not defined within development project.

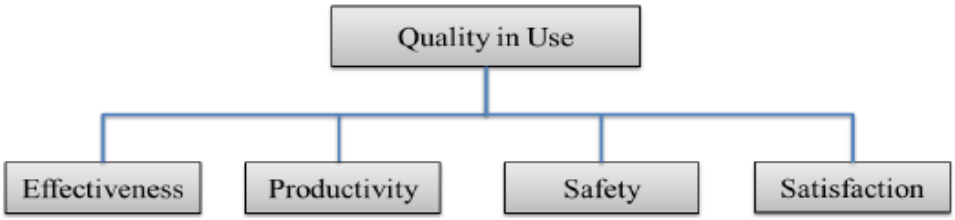


Figure 3. Definition of 'Quality in Use'

We solve this issue with defined method of 'Manual based Testing', "Figure 4". The method manual based testing is not testing for user manual or some kind of documentations. The testing method is below three approach.

• Define four actors referred by persona for developing product and use cases

• Classify all functionalities by actors

• List up view points for reading and searching manuals by actors

We defines four actors: administrators, developers, power users and general users. Also, we believe some of functionalities used by specific actors and general users can't distinguish between individual system and ready to use software. So, we verify whether each manuals are suitable for actors using by developing test cases. Using this method, we can check the quality whether concept of functionalities and user documentations are matched completely. We also define each use cases testing by actors.

We believe unclear descriptions at manuals connect to lack of the quality, satisfaction characteristics. Confusing and vague functionalities meets lack of productivity characteristics. Also, Confusing manual is difficult to operate for administrators, it is considered to be lower safety.

At finally, we can verify quality characteristics at 'Quality in Use' by results of manual based testing and use cases testing.

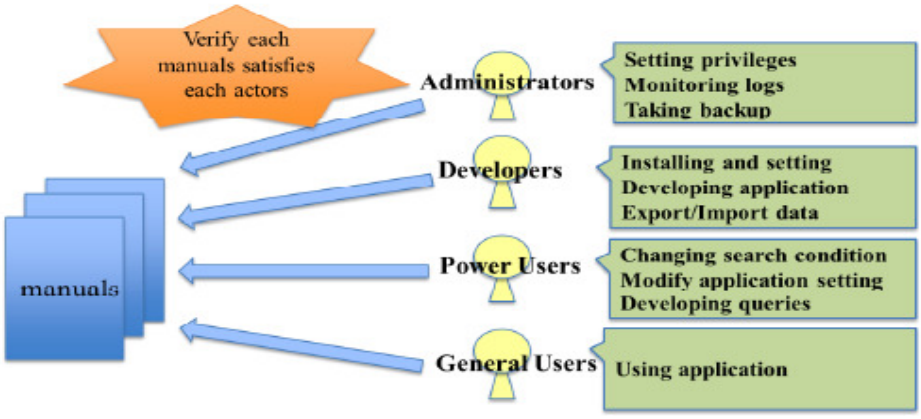


Figure 4. Concept of manual based testing method

3.5. Test completion process

A managers determine test level completion from result of each test type and verifies whether cause all bugs found at the test level are identified or not. QA managers can authorize completion of the test level when all condition of test completion defined by MTP are verified.

QA manager can add extra test for quality improvement if QA managers judge QA team need to find remaining bugs.

At RC stage, QA managers make judge whether all quality definition written in MTP is completed entirely or not. QA managers can declare end of all test.

Same as test management process, test completion process is no difference as usual process except we deal quality characteristics with analysis method.

3.6. Quality report for customers

After end of all test, QA managers write test report. Test report is summarized result of all test. QA managers write view of quality analysis classified by quality characteristics, as like bugs rate, “Figure 5” and test cases rate, ”Figure 6”. QA managers describe the reason whether the project is achieved the quality standard of our company for product releasement. The standard of our company has four rank starting from B to AAA as like automobile car safety integrity. For example, our quality standard of AAA needs that bugs rate is less than average of 3% through evaluation process, test density(test cases number per KS of source codes) than 50, and all quality characteristics are guaranteed with several test types.

QA managers need to describe product quality covered with both qualitative analysis and quantitative analysis. All test results and data, especially performance test, multi-concurrent access testing and load verification, and results of acceptance test at each test level is put at appendix section. Those test results become evidence guaranteed to each quality characteristics.

Our test report is finally authorized by stakeholders and becomes complete evidence of project. We also provide this test reports to our partners. The partners can easily judge our products’ quality and get variety of quality data at appendix section. Also, the partners can provide those quality data to customers who use or evaluate our developed products.

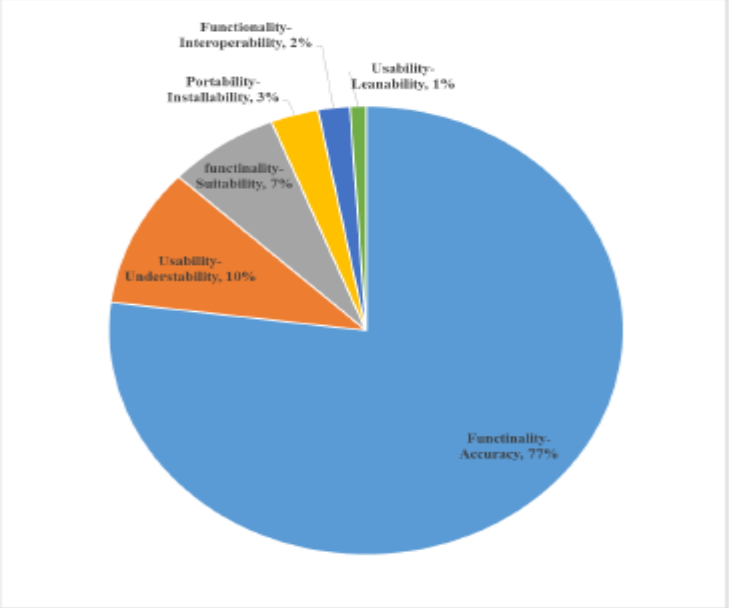


Figure 5. Bug rates classification by quality sub-characteristics

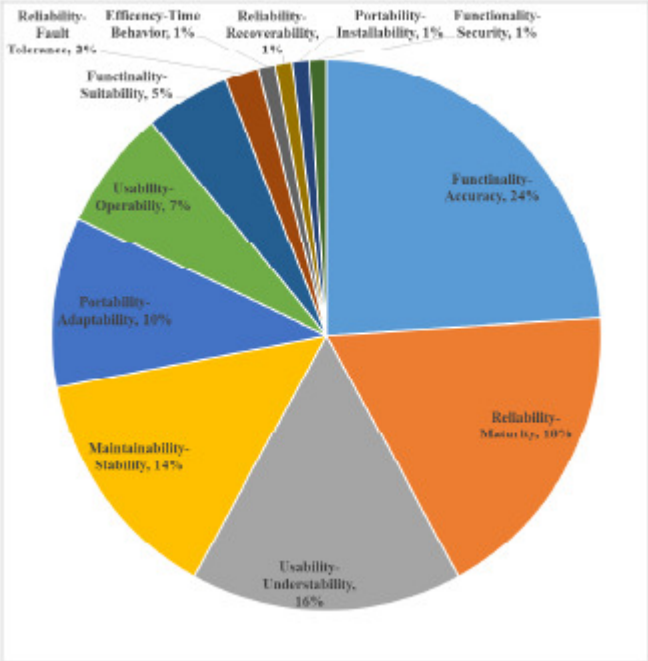


Figure 6. Test cases ratio classification by quality sub-characteristic

This approach of providing test reports with analysed quality characteristics for customers meets a part of concept of ISO/IEC 25051. According to this standard, customers can know risk, functional quality, performance, test results, existed bugs information before they buy or use ready to use software products. Our test report fits this think of the standard.

Our products have already get approval of PSQ certification with our test reports in Japan. We believe our evaluation process and our test reports have much benefit for our customers.

4. EFFORT AND BENEFIT OF OUR EVALUATION PROCESS

Our evaluation process can fit any kind of development process unless development process for ready to use software products. We have already adapted on our evaluation process to 7 development projects to release our software products. We can use the process both major release and minor release without no trouble.

The process has traceability with quality characteristics for key value indicator between MTP with test reports. This process is unsynchronized to development process, therefore, it is easily to add the process to usual development process, as like V-model, scrum process or agile process. We think other companies which develop software products can use our evaluation process with their own development process.

With using our evaluation process, we can reduce bugs found by our customers after releasement of our products and also total evaluation cost. We can estimate our evaluation cost by writing MTP and LTP in detail. QA teams can also develop test cases and execute the cases smoothly referred to MTP. QA teams reduce rework according to lack of quality.

The benefit of our evaluation process can easily provide quality analysis and metrics data based on quality characteristics. Our customers who receive our test reports can judge our products' quality fairly to fit their corporate systems. We believe our test reports reduce cost of customers' evaluation and validation whether our software products are suitable for their systems.

5. LIMITATION

Our evaluation process use quality characteristics based on ISO/IEC 9126-1. ISO/IEC 25051 is based on ISO/IEC 25000 series, SQuaRE [5]. Therefore, Compatibility or Security characteristics on ISO/IEC 25010 [6] are not fit our evaluation process. We need to change our quality characteristics to support ISO/IEC 25010.

6. CONCLUSION AND FUTURE WORK

Our evaluation process starts how to provide quality data of our software products with easy understanding and fairly to our customers. We want to reduce our customers spend much cost and time for evaluation whether our software products fit their corporate systems. We use quality characteristics at ISO/IEC 9126-1 to solve this issue and develop unsynchronized evaluation process toward development process from test planning process for MTP to writing test reports. Our evaluation process uses quality characteristics for key value indicator and acceptance test for entrance criteria to verify progress of quality at each test level. By definition of test level, our evaluation process is unsynchronized toward development process and the process has flexibility to meet variety of development process. We believe many companies can use our way easily to provide quality information by their test reports.

We have already started to support quality characteristics based on ISO/IEC 25010 to satisfy ISO/IEC 25051. According to ISO/IEC 25010, "Quality in use model" is defined with sub-characteristics. We need to research mapping the characteristics at 'Quality in Use' to our use cases testing. Also, we need to improve manual based testing and define metrics for the testing. Approach of the testing method can improve quality both software and users manuals, but we believe much of unclear test description in the method.

We have also developed detail process to comply ISO/IEC/IEEE 29119-2 [7] and also test documents are based on ISO/IEC/IEEE 29119-3 [8]. Our internal evaluation process, as like test planning process and monitoring and controlling process to comply this software test process.

After our evaluation process supports those new standard, as like SQuaRE and ISO/IEC/IEEE 29119, our process becomes key solution to comply ISO/IEC 25051.

REFERENCES

[1] ISO/IEC 25051:2014, Software engineering -- Systems and software Quality Requirements and Evaluation (SQuaRE) -- Requirements for quality of Ready to Use Software Product (RUSP) and instructions for testing

[2] PSQ Certification System, The Computer Software Association of Japan (CSAJ), http://www.psq-japan.com/

[3] ISO/IEC 9126-1, Software engineering -- Product quality -- Part 1: Quality model

[4] IEEE 829, IEEE Standard for Software and System Test Documentation

[5] ISO/IEC 25000, Systems and software engineering -- Systems and software Quality Requirements and Evaluation (SQuaRE) -- Guide to SQuaRE

[6] ISO/IEC 25010, Systems and software engineering -- Systems and software Quality Requirements and Evaluation (SQuaRE) -- System and software quality models

[7] ISO/IEC/IEEE 29119-2, Software and systems engineering -- Software testing -- Part 2: Test processes

[8] ISO/IEC/IEEE 29119-3, Software and systems engineering -- Software testing -- Part 3: Test documentation

[9] M. Azuma, (1996), "Software Products Evaluation System: Quality Model, Metrics and Process - International Standards and Japanese Practice", Inf. & Software Tech., Elsevier, Vol.38, No. 3.

[10] M. Azuma, (2000), "QUALITY IN USE; Its Concept, Metrics and Methodology", Proceedings 2nd WCSQ 2000

[11] M. Azuma, (2011), "The impact of ICT evolution and application explosion on software quality: a solution by ISO/IEC 250nn square series of standards", WoSQ'11 Proceedings of the 8th international workshop on Software quality, Pages 1-2

[12] S. Ishikawa, T.Komiyama, (2013), "Systems and Software Quality: 6. Establishment of the International Standard based Software Quality Certification Scheme", Journal of Information Processing, the Information Processing Society of Japan, in Japanese.

INTENTIONAL BLANK

CROSS CORRELATION ANALYSIS OF MULTI-CHANNEL NEAR INFRARED SPECTROSCOPY

Raul Fernandez Rojas[1], Xu Huang[1], Keng Liang Ou[2] and Jehu Lopez-Aparicio[3]

[1]Faculty of Education, Science, Technology and Mathematics, University of Canberra, Australia
{raul.fernandezrojas,xu.huang}@canberra.edu.au
[2]College of Oral Medicine, Taipei Medical University, Taiwan
klou@tmu.edu.tw
[3]Faculty of Sciences, National Autonomous University of Mexico, Mexico
jehu@ciencias.unam.mx

ABSTRACT

In this paper we present the use of a signal processing technique to find dominant channels in near infrared spectroscopy (NIRS). Cross correlation is computed to compare measuring channels and identify delays among the channels. In addition, visual inspection was used to detect potential dominant channels. The results showed that the visual analysis exposed pain-related activations in the primary somatosensory cortex (S1) after stimulation which is consistent with similar studies and the cross correlation analysis found dominant channels on both cerebral hemispheres. The analysis also showed a relationship between dominant channels and neighbouring channels. Therefore, our results present a new method to detect dominant regions in the cerebral cortex using near-infrared spectroscopy. These results have also implications in the reduction of number of channels by eliminating irrelevant channels for the experiment.

KEYWORDS

Signal processing, functional response, time dependant, medical imaging analysis, fNIRS.

1. INTRODUCTION

Near-infrared spectroscopy (NIRS) is an optical topography technique that enables non-invasive continuous monitoring of local hemodynamic response associated with cortical activity. NIRS has been applied in studies to assess cerebral functioning (functional NIRS or fNIRS) such as tasks on motor skills [1], processing of faces [2, 3] and language skills development in infants [4, 5], or perception of pain [6,7]. However, the large number of channels (24, 48, or 52) used to measure simultaneously the brain activity in patients makes the analysis and post processing of brain signals a demanding task. For that reason, region of interest (ROI) of NIRS signals are of high importance.

David C. Wyld et al. (Eds) : ITCS, JSE, SIP, ARIA, NLP - 2016
pp. 23–33, 2016. © CS & IT-CSCP 2016 DOI : 10.5121/csit.2016.60303

In medical imaging, region of interest can be defined as a specific area (anatomical location) on a digital image that is designated to implement further examination. ROIs are of common use in functional magnetic resonance imaging (fMRI) where activated regions are of particular interest. ROIs are defined on the hypothesis that the information is retained in activity patters among groups of neurons[8] and that across different patients there are regions in the human brain that exhibit the same functionality[9]. Poldrack[10] provides three reasons to select specific regions for further analysis. The first reason to perform a ROI analysis is to detect activity patterns across different conditions or variables. The second reason is to decrease the number of statistical tests to control for Type I error. The third purpose is to limit testing to a smaller region that is selected on the basis of functionally specific regions such as retinotopically organized regions in primary visual cortex or motion-sensitive regions in middle temporal cortex. In fMRI ROIs are defined either in terms of specific structural or functional features[8], however the best practice is to define ROIs for each subject based on their own anatomy[10].

In functional near-infrared spectroscopy, regions of interest are also of use. ROI analysis can be done through analysis of channels (also called Channels of Interest or COI) which show a significant increase in the hemodynamic concentration after external stimulation compared to the base-line measurement. In the neurosciences field the changes in blood circulation by evoked stimulation is also known as "hemodynamic response"[11]. In the literature, there are diverse methods proposed to obtain region of interest in functional NIRS. For example, probabilistic mapping methods[12], contrast-to-noise-ratio (CNR)[13], principal component analysis (PCA)[5], or Cross-correlation analysis[7]. Nonetheless, these methods do not provide the dominant channel, and neither the relationship between the dominant channel and adjacent channels; these two features are important to reduce the number channels for further processing.

For these reasons, this paper proposes a technique to identify the dominant channel in each cerebral hemisphere by using cross correlation analysis. In this study, we use a data set of cerebral hemodynamic responses from an acupuncture stimulation test on six subjects [7]. The analyses showed the existence of dominant channels, the association with neighbouring channels and the extent of the activated area in the brain cortex.

2. MATERIALS AND METHODS

2.1 Topography Equipment

Data was obtained using the Hitachi ETG-4000 (Hitachi Medical Corporation) to investigate cerebral hemodynamics by NIRS. The measuring equipment utilises near infrared light to monitor cortical hemodynamics. NIR light is transmitted to the patient's head using multiple optical fibre emitters (Fig. 1, red circles). The near infrared light penetrates head tissue and bone to reach the cerebral cortex. Some NIR light is absorbed by hemoglobin, while the non-absorbed light is reflected to the source, where it is sampled by an optical detector (Fig. 1, blue circles). The NIR light between emitters and detectors is sampled at a given time point named channels (Fig. 1, numbered squares). Because the absorption of NIR light of Oxy-hemoglobin (HbO) and Deoxy-hemoglobin (HbR), two wavelengths of light (695 and 830 nm) are needed for their continuous monitoring; while total hemoglobin (HbT) is calculated as the difference between HbO and HbR. The sample frequency used in this experiment was of 10 samples per second. The probe configuration used for this study was using two probes of 12 channels to measure neurologic activity. The measurement area examined is the bilateral motor cortex area, as we

expected to obtain hemodynamic response in the somatosensory cortex area (S1)[14, 15]. According to the international EEG 10-20 system[16], the probes were centred on the C3 and C4 position. Fig. 1 shows the 24-channel configuration used in the study, channels 1 to 12 represent the right hemisphere, while channels 13 to 24 represent the left hemisphere.

2.2 Participants and Experimental Protocol

Six healthy right-handed subjects (2 females, 4 males) were examined in the experiments, age range 25 to 35 years (mean age 27.8) old were included. Written consent was obtained from all participants prior to initiation of the experiments. Subjects with a history of a significant medical disorder, a current unstable medical condition, or currently taking any medication, were excluded. All participants were explained about the acupuncture procedure and manipulations, and the subjects had the opportunity to stop the procedure at any time if needed. In addition, all the subjects reported that never had used any type of acupuncture treatment in the past.

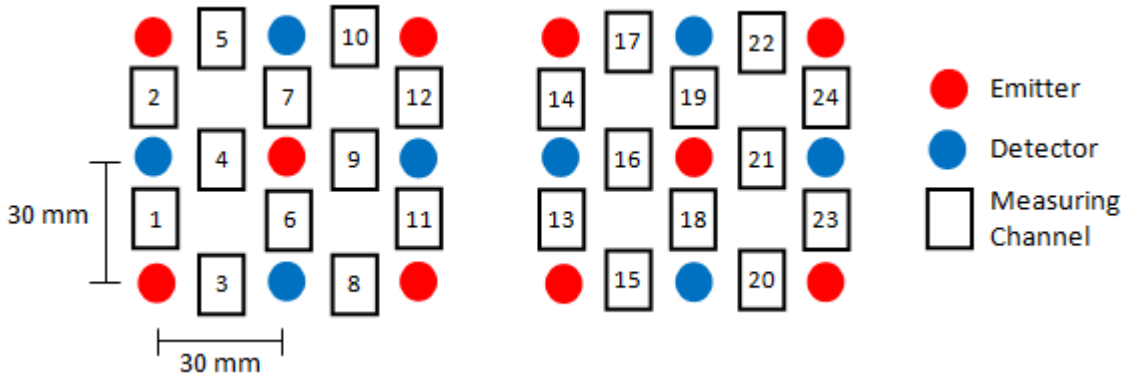


Figure 1. Channel configuration, right hemisphere (channels 1-12) and left hemisphere (channels 13-24).

The experiments were designed by the School of Oral Medicine of Taipei Medical University (TMU, Taiwan) in collaboration with the University Of Canberra (UC, Australia). The study and methods were carried out in accordance with the guidelines of the declaration of Helsinki and approved by full-board review process of the TMU-Joint Institutional Review Board under contract number 201307010. All experiments were carried out at TMU in a quiet, temperature (22-24o) and humidity (40-50%) controlled laboratory room. The experiments were done in the morning (10:00am-12:00pm) and each experiment lasted around 30 minutes. Quantitative data was collected using the ETG-4000 with the patients sat down in an ergonomic chair near the topography system (Fig. 2, left panel).

In order to obtain stimulation-related activation in the cerebral cortex, acupuncture was used to induce pain stimulation in a safe manner. Brand new acupuncture needles were used for each experiment, and using traditional Chinese acupuncture techniques that were performed by an acupuncturist of TMU Hospital. The puncture point used for stimulation was the “Hegu Point”, located on top of the hand, between the thumb and forefinger (Fig. 2, right panel). This point was used because it is an area of easy access and the hand can be set aside while the patient is relaxed on the chair. Each patient was punctured on both hands, each hand on separate days; each hand was treated as a separated experiment[7]. The acupuncture procedure consisted of three types of acupuncture stimulations (tasks). Each task lasted 5 seconds and after each stimulation a resting time (Rt) of 30 seconds was done. The first stimulation is needle insertion (T1), the three

following stimulations are needle twirl to increase Qi (T2), and the last stimulation is needle removal (T3). The complete data set was used as primary source to assess our dominant channel identification method.

2.3 Time-Dependant Cross Correlation

In signal processing, cross correlation is used to calculate the temporal similarity between two signals. In the NIRS context, cross correlation is employed to compare NIRS signals and to identify the dominant channels on both hemispheres. The cross correlation function was computed between channels 1-12 in the right probe and 13-24 in the left probe. This measure of temporal similarity of two signals can be done by computing a time-shifting along one of the input signals. The cross correlation between two waveforms x(t) and y(t) can be defined as follows: $r_{xy}(\tau) = \sum_{-\infty}^{\infty} x(t)y(t-\tau)$. Where τ is the time-lag between x(t) and y(t), the value of r_{xy} denotes the difference (lag/lead) between channel signal y(t) and channel signal x(t). The cross correlation value between two channels in the same probe is done after each stimulation from -40 sec to +40 sec at a rate of 10 samples per second.

Figure 2. Left image shows a subject wearing the multi-channel probe next to the spectroscopy equipment. Right image presents the acupuncture needle in the “Hegu point”.

2.4 Evaluation Approach

To evaluate our experiment, visual inspection can be done on activation patterns in both hemispheres and will be focus on locating potential dominant areas. The visual analysis will be carried out by configuring the NIRS data following the channel configuration shown in Fig 1. This analysis will also provide evidence of cortical activity that is associated with the stimulation-related sensory processing in the brain cortex.

3. RESULTS

3.1 Visual Pattern Identification

It was possible to depict activated regions in the NIRS data by visual inspection. Fig. 3 presents the colour representation of three sequential images from the right (Ch1-Ch12) and left (Ch13-Ch24) hemispheres of subject four (male, 26 year old); only needle insertion task (T1) is presented in this study because it was the strongest stimulation reported by all subjects. The images were taken every five seconds after external stimulation, subject and sampling time were

chosen for graphic purposes. Two events were observed, the existence of a dominant area and the propagation delay of activated areas in the brain cortex. These findings were consistent in all participants and on both hemispheres.

In the identification of dominant areas two situations can be noticed, the single-dominant region and the multi-dominant region. In the first case, the single-dominant region is when the dominant area is very evident and stands out among other regions. An example of this can be seen in Fig. 3A, the area around Ch7 (dotted oval) is very evident that this region has a higher response to external stimulus as compared to other areas. The second case is when the data have a multi-dominant response to evoked pain. Fig. 3B shows an example of this case, two main regions can be observed; the brain response is greater around Ch19 and Ch18. In this case the selection of a dominant channel is not as evident as the single-region case. In both cases, the selection of a dominant channel is made by visual perception; however in the multi-dominant response, another method is needed to identify the dominant channel.

After the initial visual analysis, a delay between the dominant area and surrounding areas was also eminent. This is evident in Fig. 3A, around Ch7 the magnitude of HbO traces is higher and seems to be the centre of propagation towards adjacent channels in posterior images. For example, in Fig. 3A-I, the area around Ch7 appears to be slightly more active than the rest of the channels. However, after five seconds (Fig. 3A-II) the activation has increased and reached Ch5 and Ch10. In the last image (Fig. 3A-III), the activation is stronger than previous image and radiation pattern expanded to neighbouring channels Ch2, Ch4, Ch9, and Ch12. This phenomenon suggests a strong relationship between Ch7 and surrounding channels.

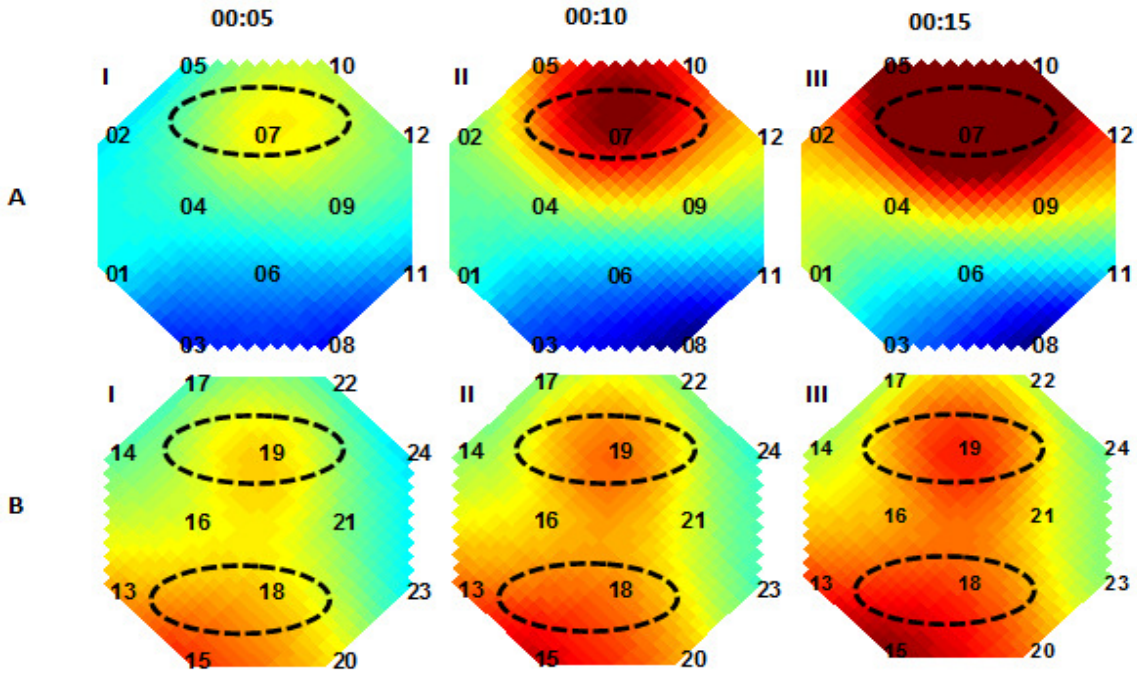


Figure 3. Visual analysis. A) Single dominant pattern. B) Multi-dominant pattern. Three images were taken every five seconds after stimulation. Dotted circles show the dominant region. Cortical activity was initially started in a small region and extended to surrounding areas.

3.2 Cross Correlation Signal Processing

Based on the assumption that a dominant channel and a relationship among the channels exist, the cross correlation analysis was computed to examine these two ideas. Fig. 4 shows the results of two cross correlation comparisons of two channels, Ch7 (Fig. 4A) and Ch16 (Fig. 4B) on right and left cerebral hemispheres respectively, from subject four after the acupuncture stimulus.

Detecting the dominant channel in dominant regions was done by comparing the lags between potential dominant channel and the rest of the channels. The detection was carried out by correlating the delay times between possible dominant channel and the rest of channels in that hemisphere. Fig. 4A shows the cross correlation analysis and confirmed that the region around Ch7 was the dominant area. It was also found that Ch5, and Ch10 have no time delay (τ =0.0 sec) with Ch7, which suggests that these three channels are positioned over a potential region of interest due to the strong activation after stimulation. Moreover, cross correlation analysis of the opposite hemisphere (Fig. 4B) clarified the impression of having two dominant regions after visual analysis. This result illustrated that Ch16 lead the rest of the channels for this particular subject and experiment. It was also found that Ch18 (τ =+0.8 sec) and Ch19 (τ =+0.3 sec) have minimum lags against Ch16, which rejects the idea of having two separate dominant areas with high response to the external stimulation. Therefore, we can tell that for subject 4 (n=4), Ch16 is the dominant channel on the left hemisphere. As a result, it is possible to tell that the dominant channel is the measuring channel with the fastest answer after external stimulation; and in most of the cases within this experiment, the dominant channel is the channel with the strongest HbO response after noxious stimulation.

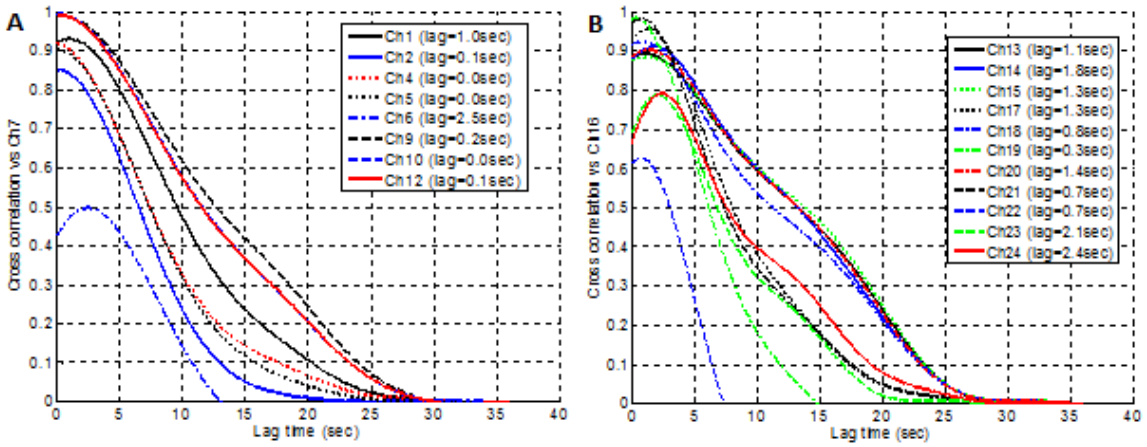


Figure 4. Cross correlation comparison of dominant channel and surrounding channels from subject four. A) Time-dependant evaluation of Ch7 and neighbouring channels. B) Time-dependant comparison between Ch16 and neighbouring channels.

Cross correlation signal processing was also used to determine the relationship between dominant channels and neighbouring channels. For instance, the results from subject four (n=4) showed that Ch7 and surrounding channels have a small lag time that confirms that NIRS signals in dominant regions travel to surrounding regions. Fig. 5 presents the delays between channels on both hemispheres, we can see (Fig.5A) that region formed by Ch2, Ch4, Ch5, Ch7, Ch9, Ch10 and C12 has near cero delay ($\tau \approx 0$ sec), while opposite channels Ch1, Ch3, Ch6, Ch8, Ch11 present a bigger delay against Ch7. This propagation is a clear indication of the relationship of

NIRS signals between dominant channel and neighbouring channels. We can also see that channels (red squares) with small delays with dominant channel present a higher activation, while channels with larger delays belong to areas with limited activation (blue squares). This can be observed similarly in the left hemisphere as showed in Fig. 5B.

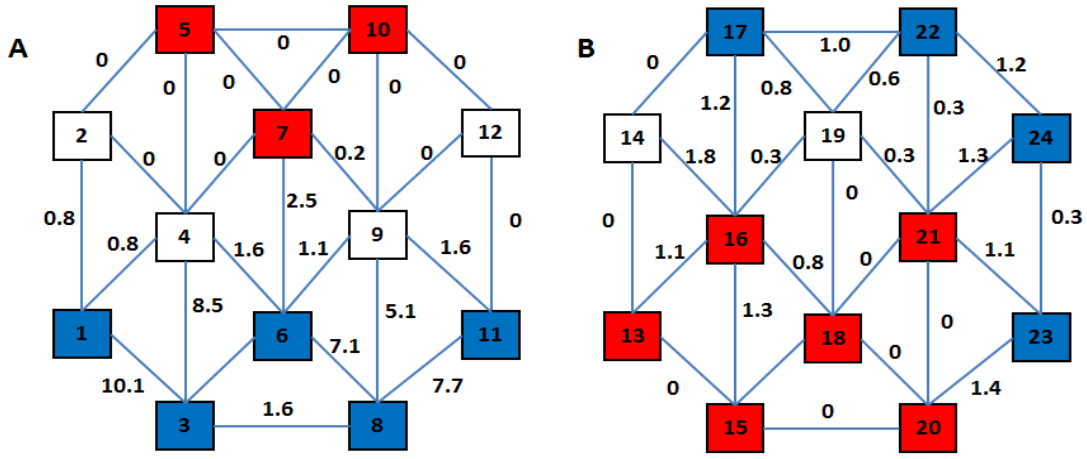


Figure 5. Cross correlation signal processing. Propagation time between activated areas (red) and less active channels (blue). A) right hemisphere. B) left hemisphere.

These results suggest that brain hemodynamic signals have a progressive movement from dominant regions towards peripheral regions. It is also important to note that the results after following stimulations (tasks) remained very similar, however, the lag and lead times between channels were slightly smaller. This situation can be explained by the fact that the first stimulation produced the strongest response in all the acupuncture experiment.

5. DISCUSSION

In the present study, we have used cross correlation signal processing to identify dominant channels in functional near-infrared spectroscopy (fNIRS). Cross correlation analysis finds the dominant region based on the lag times between channels and also provides an estimation of the relationship between channels. The visual inspection also corroborated the propagation effect and dominant patterns in the NIRS data.

We first used visual inspection of the stimulation-related activation patterns in the brain cortex to obtain an initial assessment of dominant channels. It was expected to obtain activation in functional areas where the cortical representation of pain is involved. It was noted that the brain response increased around the area of channel 7 (Ch7) on the right hemisphere and the area of channel 19 (Ch19) on the left hemisphere in all the subjects; being these two areas part of the postcentral gyrus in the parietal lobe. The postcentral gyrus is the location of the primary somatosensory cortex (S1), area that has been reported to be involved with the perception and modulation of painful somatosensory sensations[15]. It is also important to note that the cortical activation following the acupuncture stimulation was obtained on both hemispheres (bilateral S1 activation). These findings are in line with previous results [15, 17]. However, other studies in pain stimulation have reported that pain activation can be also detected in the secondary somatosensory cortex (S2), anterior cingulate cortex (ACC), and insular cortex (IC)[14, 18, 19].

Based on this preliminary analysis, we expected to find ROIs in these areas and also confirmed the validity of our overall acupuncture experiment.

Cross correlation between channels was calculated to identify dominant channels and the relationship among channels. The analysis exhibited the existence of delays between channels, not only in the region of interest but also in the whole measuring probe. These delays between signals indicate that leading signals are dominant channels and based on the delays between channels, it is possible to say that the delays represent the movement of activated areas to surrounding areas. Moreover, our results showed that the dominant channel is one of the channels with the highest response to external stimulation.

An important result from the cross correlation analysis is that the number of control signals can be reduced by eliminating irrelevant channels. For example, channels with very small activation after stimulus can be excluded from the data base since they fail to represent any interaction within the experiment. Therefore, signal processing techniques such as cross correlation are of importance to find relevant channels that truly represent activations after stimulus.

6. CONCLUSIONS

In conclusion, this study presents an analysis of near infrared spectroscopy (NIRS) signals by cross correlation. The cross correlation analysis exhibited the dominant channels in the region of interest on both hemispheres, and found that is the channel with the fastest response to external stimuli and with a very strong activation after stimulus. The results also showed that the time lags of the hemoglobin signals represent a relationship between dominant channel and neighbouring channels; these delays can be seen as the delay of localized cerebral blood flow from dominant areas traveling to surrounding areas. In addition, after a visual inspection, it was found that the acupuncture stimulation resulted in bilateral activation in the primary somatosensory region (S1) in all subjects. Based on the evaluations, our analysis successfully showed the region of interests and visualizes them as the apparent motion of cortical activity from dominant channel. Therefore, our results present a new method to detect region of interest in functional near-infrared spectroscopy.

REFERENCES

[1] Hatakenaka M, Miyai I, Mihara M, Sakoda S, Kubota K. Frontal regions involved in learning of motor skill—a functional NIRS study. NeuroImage. 2007;34:109-16.

[2] Kobayashi M, Otsuka Y, Nakato E, Kanazawa S, Yamaguchi MK, Kakigi R. Do infants represent the face in a viewpoint-invariant manner? Neural adaptation study as measured by near-infrared spectroscopy. Front Hum Neurosci. 2011;5.

[3] Honda Y, Nakato E, Otsuka Y, Kanazawa S, Kojima S, Yamaguchi MK, et al. How do infants perceive scrambled face?: A near-infrared spectroscopic study. Brain Res. 2010;1308:137-46.

[4] Gervain J, Macagno F, Cogoi S, Peña M, Mehler J. The neonate brain detects speech structure. Proc Natl Acad Sci USA. 2008;105:14222-7.

[5] Kovelman I, Shalinsky MH, Berens MS, Petitto L-A. Shining new light on the brain's "bilingual signature": a functional Near Infrared Spectroscopy investigation of semantic processing. NeuroImage. 2008;39:1457-71.

[6] Bartocci M, Bergqvist LL, Lagercrantz H, Anand K. Pain activates cortical areas in the preterm newborn brain. Pain. 2006;122:109-17.

[7] Rojas F.R, Huang X, Ou KL, Tran D, Islam SMR. Analysis of pain hemodynamic response using near-infrared spsctroscopy (NIRS). Int J Mult Appl. 2015;7:31-42.

[8] Etzel JA, Gazzola V, Keysers C. An introduction to anatomical ROI-based fMRI classification analysis. Brain Res. 2009;1282:114-25.

[9] Nieto-Castanon A, Ghosh SS, Tourville JA, Guenther FH. Region of interest based analysis of functional imaging data. NeuroImage. 2003;19:1303-16.

[10] Poldrack RA. Region of interest analysis for fMRI. Soc Cogn Affect Neurosci. 2007;2:67-70.

[11] Ortiz E. Use of Neuroimaging to Clarify How Human Brains Perform Mental Calculations.Online Submission. 2010.

[12] Verner M, Herrmann MJ, Troche SJ, Roebers CM, Rammsayer TH. Cortical oxygen consumption in mental arithmetic as a function of task difficulty: a near-infrared spectroscopy approach. Front Hum Neurosci. 2013;7.

[13] Song X, Pogue BW, Jiang S, Doyley MM, Dehghani H, Tosteson TD, et al. Automated region detection based on the contrast-to-noise ratio in near-infrared tomography. Appl Opt. 2004;43:1053-62.

[14] Hofbauer RK, Rainville P, Duncan GH, Bushnell MC. Cortical representation of the sensory dimension of pain. J Neurophysiol. 2001;86:402-11.

[15] Bushnell M, Duncan G, Hofbauer R, Ha B, Chen J-I, Carrier B. Pain perception: is there a role for primary somatosensory cortex? Proc Natl Acad Sci USA. 1999;96:7705-9.

[16] Homan RW, Herman J, Purdy P. Cerebral location of international 10–20 system electrode placement. Electroencephalogr Clin Neurophysiol. 1987;66:376-82.

[17] Sutherland MT, Tang AC. Reliable detection of bilateral activation in human primary somatosensory cortex by unilateral median nerve stimulation. NeuroImage. 2006;33:1042-54.

[18] Coghill RC, Sang CN, Maisog JM, Iadarola MJ. Pain intensity processing within the human brain: a bilateral, distributed mechanism. J Neurophysiol. 1999;82:1934-43.

[19] Apkarian AV, Bushnell MC, Treede RD, Zubieta JK. Human brain mechanisms of pain perception and regulation in health and disease. Eur J Pain. 2005;9:463-.

AUTHORS

Raul Fernandez Rojas received his B. Eng. (Electronics) at the Universidad Tecnologica de la Mixteca, Mexico and his M. Eng. (Hons) at The Australian National University, Australia. Raul's research interests are robotics, computer vision, medical imaging, and signal processing.

He is currently pursuing a PhD degree in Information Sciences and Engineering at the University of Canberra, Australia.

Xu Huang received his B.E. and M.E. degrees and first Ph.D. in Electrical Engineering and Optical Engineering prior to 1989 and his second Ph.D. in Experimental Physics in the University of New South Wales, Australia in 1992. He has earned the Graduate Certificate in Higher Education in 2004 at the University of Canberra, Australia. Prof Xu's fields of research includes: cybersecurity, network security, Internet of Things (IoT), wireless and optical communications, cloud computing, digital signal processing, bio signal processing, brain computer interface (BCI), intelligent system, and smart networks.

He has worked at the Australia National University from 1988 to 1990, the University of New South Wales from 1990 to 1995, also University of New England from 1995 to 2001. He is currently the Head of the Engineering at the Faculty of Information Sciences and Engineering, at the University of Canberra, Australia. He has been the Chair, Co-Chair, and TCM at various high quality International Conferences, and Editor for various high quality Journals. He has edited seven books, nine Book Chapters, 45 Journal Articles, and more than two hundred papers in high level of the IEEE and other international conferences (within ERA ranking); he has been awarded 17 patents in Australia in 2010 and 2013.

Prof. Huang has been a senior member of IEEE in Electronics and Computer Society since 1989, a Fellow in the Institute of Engineering Australian (FIEAust), Chartered Professional Engineering (CPEng), a Member of Australian Institute of Physics. He has been a member of Committee of the Institution of Engineering Australia at Canberra Branch for last 10 years.

Keng-Liang Ou graduated from the Mechanical Engineering Ph.D. program at the National Chiao Tung University, Taiwan. Professor Ou focuses his research on Biomaterials, Bioengineering, Biomedical Devices and Nanotechnology.

He went to Taipei Medical University to work on biomaterials research and development, and he is the current elected Dean of College of Oral Medicine in Taipei Medical University, Taiwan. He is also in charge of the Graduate Institute of Biomedical Materials and Tissue Engineering, Research Center for Biomedical Implants and Microsurgery Devices and Research Center for Biomedical Devices and Prototyping Production.

Professor Ou is the President of Institute of Plasma Engineering Taiwan, the leader of The Taiwan society for metal heat treatment and the Head Taiwan Oral Biomedical Engineering Association. He is the leader and organizer for the biomedical product design, production, manufacturing, testing, legalization and market planning, with supports from team of scientists and researchers with different expertise. Professor Ou has extensive collaborations with industry and has played a major role in developing medical devices for health service professions in the world. Professor Ou was honorably awarded with the 49th Ten Outstanding Young Persons of Taiwan on 2011.

Jehu Lopez-Aparicio received his Master degree in Electronics Sciences from the Meritorious Autonomous University of Puebla, Mexico in 2013. Jehu is currently completing his PhD at The National Autonomous University of Mexico. Jehu's research interests are: microelectronics design, biosensors design, nanoelectronics, and quantum physics.

He has been working at The National Autonomous University of Mexico since 2014 as an academic and electronics laboratory demonstrator. Jehu has also worked for CIDEC/Delphi technical center as a system engineer for two years.

Mr Lopez-Aparicio has been awarded a patent for an electronic system to count photons, a project were he designed this system to complete his Master degree at the Meritorious Autonomous University of Puebla. Recently, Jehu also co-invented a biosensor to monitor in real-time glucose and insulin through a blood sample. This biosensor was awarded with the 2015 Latin America Google Research Award.

INTENTIONAL BLANK

A FAST SEARCH ALGORITHM FOR LARGE VIDEO DATABASE USING HOG BASED FEATURES

Qiu Chen[1], Koji Kotani[2], Feifei Lee[3] and Tadahiro Ohmi[4]

[1]Department of information and Communication Engineering,
Kogakuin University, Japan
[2]Department of Electronics, Graduate School of Engineering,
Tohoku University, Japan
[3]School of Optical-Electrical and Computer Engineering,
University of Shanghai for Science and Technology, China
[4]New Industry Creation Hatchery Center, Tohoku University, Japan
chen@cc.kogakuin.ac.jp, feifeilee1701@163.com

ABSTRACT

In this paper, we propose a novel fast video search algorithm for large video database. Histogram of Oriented Gradients (HOG) has been reported which can be reliably applied to object detection, especially pedestrian detection. We use HOG based features as a feature vector of a frame image in this study. Combined with active search, a temporal pruning algorithm, fast and robust video search can be achieved. The proposed search algorithm has been evaluated by 6 hours of video to search for given 200 video clips which each length is 15 seconds. Experimental results show the proposed algorithm can detect the similar video clip more accurately and robust against Gaussian noise than conventional fast video search algorithm.

KEYWORDS

Fast search, Video database, HOG Features

1. INTRODUCTION

In recent years, video content becomes commonplace on the web and the size of video database quickly increases due to rapid developments of internet connection and disk storage technology in recent years. Video retrieval has become a hot area of research in current. Video search is an important problem in this area because it has a wide range of applications such as TV commercials detection [1], video copyright enforcement [2],[3], video clustering and so on. In this paper, video search means when a user presents a query video clip to the search engine, the search engine should identify all similar ones, that is to say, accurately locate the position of query video clip if it exists in the video database.

David C. Wyld et al. (Eds) : ITCS, JSE, SIP, ARIA, NLP - 2016
pp. 35–41, 2016. © CS & IT-CSCP 2016 DOI : 10.5121/csit.2016.60304

Many video search algorithms [7]-[10] have been proposed, and achieves successes to a certain extent. But such algorithms, however, are computational-power hungry for the exhaustive search of large video database. For large video database, Search speed is an important issue of video search. Base on active search [4], a temporal pruning algorithm, Kashino et al. [1] improved the conventional multimedia search algorithm. Nevertheless, their feature extraction utilizes intensity features of the frame image, so the results may be sensitive to small change of luminance and motion in the frame. In this paper, we utilizes a new feature based on Histogram of Oriented Gradients (HOG) [5] features, which had been reliably applied to object detection, especially pedestrian detection [6][11]. It has the following advantages: computational simplicity, motion-insensitivity and luminance-insensitivity. Because such a feature is compatible with active search algorithm, fast search speed can also be achieved by combining HOG based features and active search.

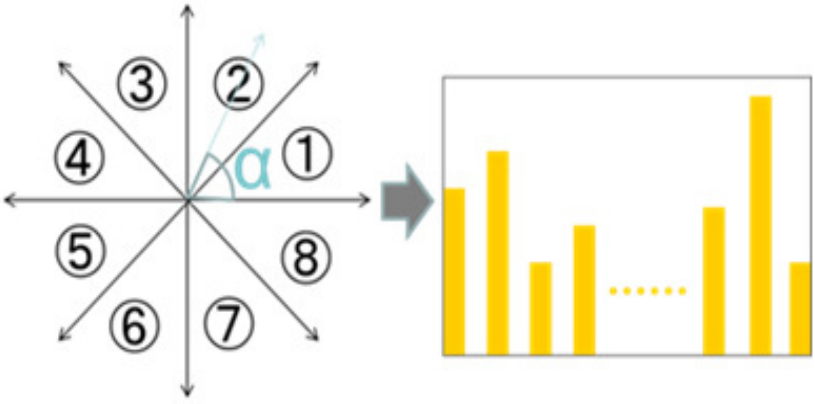


Figure 1. Processing steps of HOG based features.

In section 2, we will first introduce HOG based features, and then describe fast video search algorithm we employ in section 3. Experimental results compared to conventional search approach will be discussed in section 4. Finally, conclusions are given in section 5.

2. HOG BASED FEATURES

Histogram of Oriented Gradients (HOG) [5] features has been developed for object recognition previously. Figure 1 shows the processing steps of HOG based features. In HOG, for each pixel of an input image, the intensity difference of the horizontally adjacent pixels (*fx*) and the intensity difference of the vertically adjacent pixels (*fy*) are first calculated by using simple subtraction operations shown as formula (1), (2).

$$fy = f(i, j+1) - f(i, j-1) \tag{1}$$

$$fx = f(i+1, j) - f(i-1, j) \tag{2}$$

$$\alpha = tan^{-1}\frac{fy}{fx} \tag{3}$$

Each intensity variation vector is then quantized simply by the oriented angle calculated by formula (3). The number of vectors quantized in each quantization region is counted and a histogram is generated.

The essence of HOG based features above can be considered that the operation detects and quantizes the direction of intensity variation in the image blocks. Hence HOG based features contain very effective image feature information. We will describe how to apply it as feature vector of frame to solving the fast video search problem in next section.

3. PROPOSED VIDEO SEARCH ALGORITHM

The procedure of proposed fast search algorithm is shown in figure 2. In the pre-processing stage, the feature vectors are calculated from the query video clip and the video database HOG method described in section 2. The feature vectors are then quantized using VQ algorithm which applied quantization by combinations of scalar quantization (SVQ) for each feature dimension [1]. In the search stage, the windows are applied to both the query feature vectors and the feature vectors of video database. In the next step, the number of vectors quantized in the windows of the query video clip and video database are counted and feature vector histograms are created respectively. The similarity between these histograms is then calculated. If the similarity exceeded a threshold value given previously, the query video clip will be detected and located. Otherwise, the window on the video database will be skipped to the next position determined by the similarity in current position and the threshold value. In the last step, the window on the video database is shifted forward in time and the search proceeds.

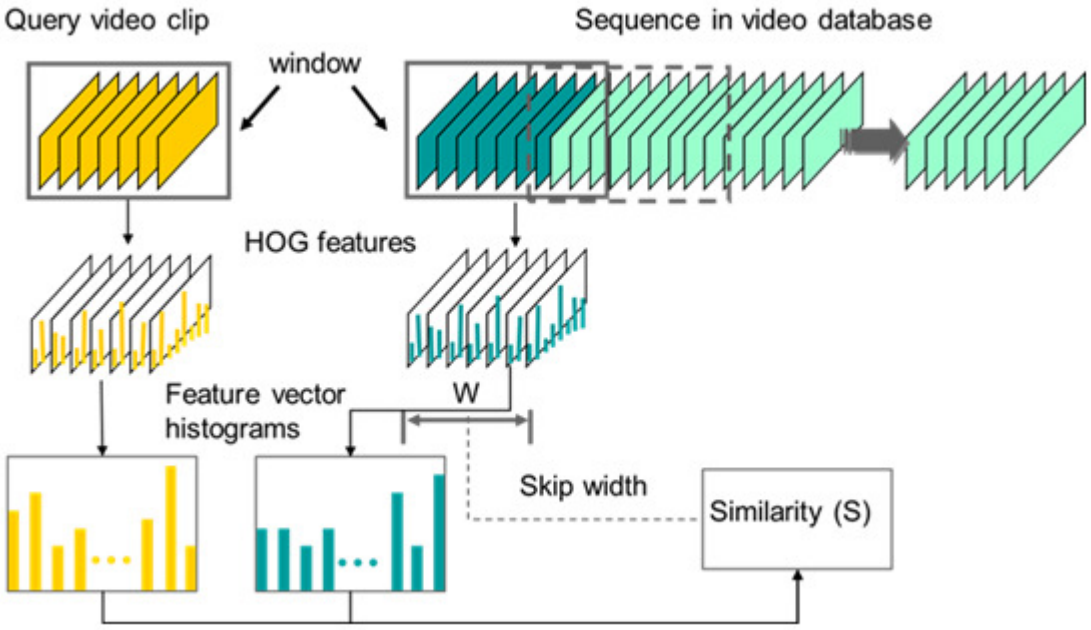


Figure 2. The procedure of proposed fast search algorithm

Here, histogram intersection is used as the similarity measure [4], and is defined as formula (4).

$$S_{QD} = S(H_Q, H_D) = \frac{1}{N}\sum_{l=1}^{L} \min(h_{Ql}, h_{Dl}) \quad (4)$$

where h_{Ql}, h_{Dl} are the numbers of feature vectors contained in the *l-th* bin of the histograms for the query and the stored signal, respectively, L is the number of histogram bins, and N is the total number of feature vectors contained in the histogram. The skip width w is shown by formula (5).

$$w = \begin{cases} floor(N(\theta - S_{QD})) + 1 & (S_{QD} < \theta) \\ 1 & otherwise \end{cases} \quad (5)$$

where *floor(x)* means the greatest integral value less than x, and θ is a given threshold.

4. EXPERIMENTAL RESULTS AND DISCUSSIONS

We performed all of the experiments on a conventional PC @ 3.2GHz (4G memory). The algorithm was implemented in ANSI C. We used 6 hours of video captured from TV program. In the experiment, the video frame rate was 14.97 fps, and image size was 80*60 as shown in table I.

Tabel I: Parameters of Video dataset.

Video content	News, drama, sports etc.
Video length	Query video clips: 15s * 200 Video database sequence: 6 hours
Frame rate	14.98 fps
Frame number	Query video clips: 15s * 200 Video database sequence: 6 hours
Image format	PPM
Capture size	80*60

We captured 6 hours of video twice, one for video database sequence and the other for query video clips. Query video clips were generated by selecting video clips randomly for 200 times from the second video. Then we can perform search for 200 video clips from 6 hours of video. The threshold θ is 0.7 determined by preparing experiments according to FAR and FRR curve, which will be discussed later.

We utilized a 1-hour video sequence by selecting randomly from the second video to determine boundary threshold which were used to implement scalar VQ process (SVQ).

To suit the search task, quantization levels of HOG are set at 8 in θ -axis (totally 9) in the feature extraction stage. Thus, the number of histogram bins is total 512. Similarity calculation between the feature vector histograms will be quite faster compared with conventional algorithm which number of histogram bins is 4096.

4.1. Image Features of Conventional Algorithm

In conventional algorithm [1], they use small scaled images as video features. An image feature vector is defined as formula (6).

$$g(k) = (g_1(k), \cdots, g_j(k) \cdots, g_W(k)) \quad (6)$$

where k is the frame number, j is the division number of the subimages, and W is the number of subimages. The $g_j(k)$ is the normalized intensity and is defined as formula (7), where $\bar{x}_j(k)$ is the average intensity in the j-th subsection.

$$g_j(k) = \frac{\bar{x}_j(k) - \min_i \bar{x}_i(k)}{\max_i \bar{x}_i(k) - \min_i \bar{x}_i(k)} \tag{7}$$

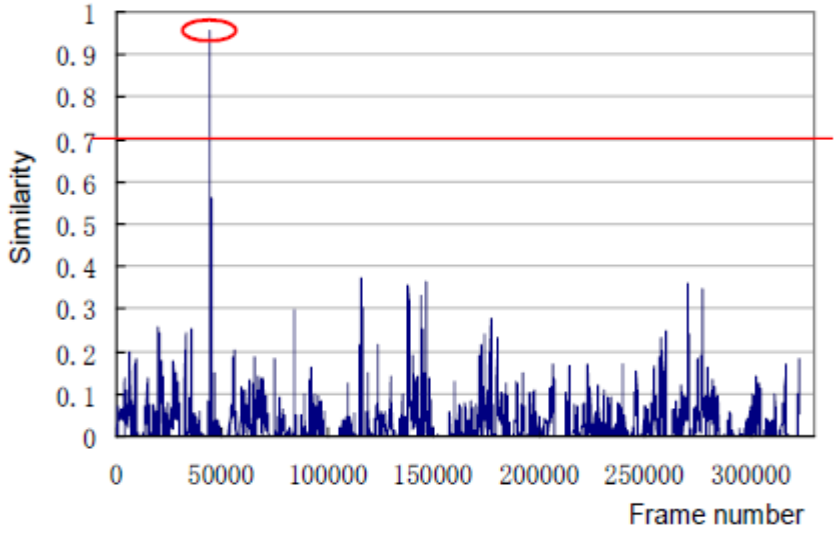


Figure 3. The example of search result.

4.2. Experimental Results

Figure 3 shows an example of result in 200 times search. The red ellipse marks the correct location of detected query video clip. Results of search accuracy are shown in figure 4. Search accuracy is shown as a function of window size. The perfect accuracy of 100% is obtained when window size is given 30sec. But even if window size decreases to 2sec, the accuracy still remains 98%.

We also compared our algorithm with the algorithm which does not utilize active search (full search), and conventional fast search algorithm described in section 4.1. Table II gives the approximate computational cost of the algorithms. As descried above, the number of histogram bins is total 512 in our proposed algorithm, 8 times smaller than that of conventional algorithm. From Table I, we can see the search time costs only 70ms, which is 271 times faster than full search, and also 6.7 times faster than the conventional fast search algorithm.

We also investigated the robustness of image features used in respective algorithms by adding Gaussian noise to the query video clips. Figure 5 shows how the search accuracy changes with the amount of noises. The curves with trigonal mark and foursquare mark stand for proposed algorithm and conventional algorithm, respectively. Our proposed algorithm achieves higher search accuracy than conventional algorithm. Although the search accuracy decreases with

increase the amount of Gaussian noises, it can be said that proposed algorithm is more robust for video search task than the conventional approach.

Table I: Approximate Computational Cost (CPU time).

Stage	Full search	Conventional	Proposed algorithm
Feature Extraction	540sec	540sec	550sec
VQ processing	50ms	50ms	35ms
Search	19sec	470ms	70ms

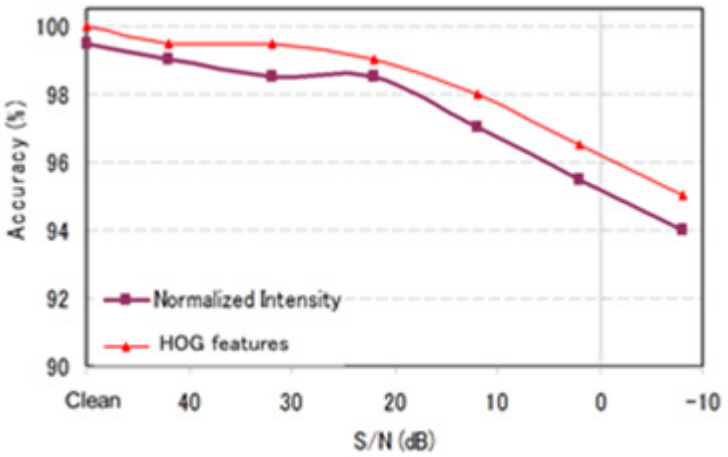


Figure 5. Gaussian noise vs search accuracy.

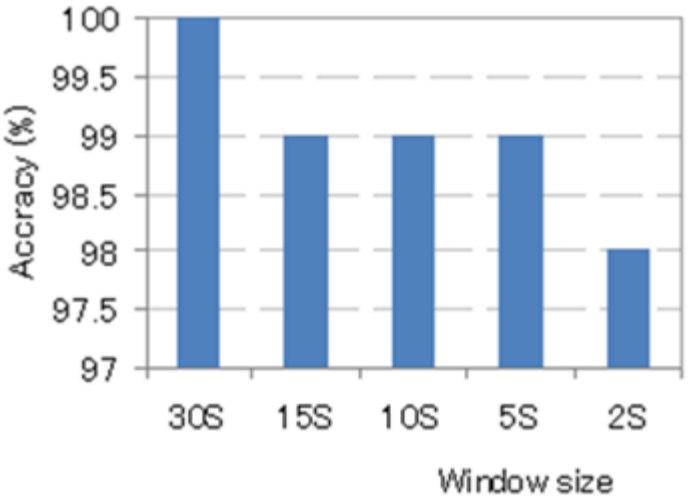


Figure 4. Window size vs. search

5. CONCLUSIONS

By using a new feature based on HOG based features, we present a fast and robust video search algorithm for video clips from large video database. The proposed search algorithm has been

evaluated by 6 hours of video to search for 200 video clips. Experimental results show that search time costs only 70ms, which is 271 times faster than full search, and also 6.7 times faster than the conventional fast search algorithm. Furthermore, proposed algorithm is more robust against Gaussian noise for video search task than the conventional approach.

REFERENCES

[1] K., Kashino, T. Kurozumi, and H. Murase, "Quick AND/OR search for multimedia signals based on histogram features", IEICE Trans., J83-D-II(12), 2000, pp. 2735-2744.

[2] S.S. Cheung and A. Zakhor, "Efficient video similarity measurement with video signature", IEEE Trans. on Circuits and System for Video Technology, 13(1), 2003, pp. 59-74.

[3] A. Hampapur, K. Hyun, and R. Bolle, "Comparison of sequence matching techniques for video copy detection", SPIE. Storage and Retrieval for Media Databases 2002, 4676, San Jose, CA, USA, 2002, pp. 194-201.

[4] V.V. Vinod, H. Murase, "Focused color intersection with efficient searching for object extraction", Pattern Recognition, 30(10), 1997, pp. 1787-1797.

[5] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection", In CVPR, pp. 886–893, 2005.

[6] C. Hou, H. Ai, and S. Lao, "Multiview pedestrian detection based on vector boosting", In Asian Conference on Computer Vision, pp.18–22, 2007.

[7] L. Agnihotre, N. Dimitrova, T. McGee, S. Jeannin, S. Schaffer, J. Nesvadba, "Evolvable visual commercial detector", In Proceedings of IEEE. International Conference on Computer Vision and Pattern Recognition, vol. 2, 2003, pp. 79-84.

[8] R. Lienhart, C. Kuhmunch, W. Effelsberg, "On the detection and recognition of television commercials", In Proceedings of IEEE Conference on Multimedia Computing and Systems, 1997, pp. 509-516.

[9] R. Mohan, "Video sequence matching", In Proceedings of the International Conference on Audio, Speech and Signal Processing, vol. 6, 1998, pp. 3679-3700.

[10] A. Araujo, M. Makar, V. Chandrasekhar, D. Chen, S. Tsai, H. Chen, R. Angst, and B. Girod, "Efficient video search using image queries", In Proceedings of International Conference on Image Processing, pp. 3082-3086, 2014.

[11] Q. Zhu, M. C. Yeh, K. T. Cheng, and S. Avidan, "Fast human detection using a cascade of histograms of oriented gradients" In CVPR, pp. 1491–1498, 2006.

INTENTIONAL BLANK

MULTIAGENT SYSTEM FOR SCRUTINY OF ADMISSION FORMS USING AUTOMATIC KNOWLEDGE CAPTURE

Puneet Modgil[1] and M. Syamala Devi[2]

[1]Department of Computer Science and Applications,
Panjab University, Chandigarh, India
Punmod@rediffmail.com
[2]Department of Computer Science and Applications,
Panjab University, Chandigarh, India
Syamala@pu.ac.in

ABSTRACT

Multiagent Systems are autonomous intelligent systems. In many academic institutions student admissions are performed after generating merit lists. Generation of merit lists is preceded by manual scrutiny of admission forms. This manual scrutiny is a knowledge-intensive, tedious and error-prone task. In this paper the design, implementation and testing of Multiagent System for Scrutiny of Admission Forms (MASAF) using Automatic Knowledge Capture is presented. MASAF consists of three agents namely: Form agent, Record agent, and Scrutiny agent. These three agents, using ontology, cooperatively fulfill the goal of highlighting the discrepancies in filled forms. MASAF has been tested by scrutinizing about 1000 forms and all of discrepancies found were correct as verified by human scrutinizer. Thus it can be concluded that using Multiagent system for scrutiny of forms can reduce human intervention, improve performance in terms of speed and accuracy. The system can be enhanced to automatically correct the discrepancies in forms.

KEYWORDS

Multiagent system, Knowledge Capturing, Scrutiny of admission forms, Ontology

1. INTRODUCTION

An Agent is a software entity that can automatically perceive environment and can behave reactively as well as proactively with social ability. Multiagent System is distributed artificial intelligence system in which a number of agents interact with each other to achieve the desired objectives. A global solution arises from cooperation amongst these agents but no one agent is capable of solving the problem. Thus agents are well suited for solving the problems which are knowledge intensive and distributive in nature.

In academic institutions before actual admissions take place a merit list of admission seekers is generated. Merit list is generated on the basis of certain parameters such as marks in qualifying

David C. Wyld et al. (Eds) : ITCS, JSE, SIP, ARIA, NLP - 2016
pp. 43–51, 2016. © CS & IT-CSCP 2016 DOI : 10.5121/csit.2016.60305

examination. These parameters are course specific. The information about these parameters is obtained from the forms. The next step is to verify the information provided in the forms. This scrutiny is done by responsible persons of institution on the basis of documents attached along with forms. Thus scrutiny is a tedious, error-prone and knowledge intensive task. There is a need to automate the above task and this is feasible by employing agents to capture knowledge automatically from the web and to compare with the information available in the forms. Such an automated system based on agents with knowledge capturing capabilities is presented in this paper and referred to as MASAF in the rest of the paper. Section 2 highlights the review of literature. Design of MASAF is presented in section 3. Section 4 includes implementation details, testing and results. Section 5 contains conclusions and scope for future work.

2. RELATED WORK

In this section work performed by many researchers related to Multiagent system and applications in academics and knowledge capture is reviewed and presented.

The authors in [1] present knowledge and information network approach for managing research assets in a knowledge-oriented organization using a Multiagent system. The integration of existing knowledge and information technologies to develop new ways to manage scientific and technical knowledge in modern organizations is the challenge addressed in this article.

Multi-Agent System for University Course Timetable Scheduling: MAS_UP-UCT is described in [2]. The problem is divided into two sub problems: 1. Faculty course timetable scheduling (which involves only allocation of course day and time), and 2. University course rooms allocation (which involves allocation of rooms for courses). Four types of agents are used by the system: Main Scheduler Agent (MSA), Faculty Scheduler Agent (FSA), Expert Assistant Agent (EAA), and Personal Agent (PA). Each faculty has a FSA which has to schedule the courses of that faculty.

The details of Campus Automation Web Information System (CAWIS) project is discussed in [3]. The Multi agent System is used for course planning, resource allocation, class timetabling, and office allocation, exam scheduling and other administrative and organizational activities of the university. The six basic agents defined are: (i) Students Agent (ii) Training and Learning Agent (iii) Lecturer Agent (iv) Infrastructure Agent (v) Task Agent (vi) Strategical / Goal Agent .

In [4] authors explain how wrappers are able to accept a query against the source and return a set of structured results, thus enabling applications to access web data in a similar manner to that of information from databases. A significant problem in this approach arises as Web sources may undergo changes that invalidate the current wrappers. Thus authors have presented novel heuristics and algorithms to address this problem.

In [5] Authors present their experience in building Information Retrieval (IR) systems using Multiagent technology. In particular they presented their results in using X.MAS, a generic Multiagent architecture aimed at retrieving, filtering and reorganizing information according to user interests.

3. DESIGN OF MULTIAGENT SYSTEM FOR SCRUTINY OF ADMISSION FORMS (MASAF)

In this section roles and responsibilities of agents, ontology design, agents' design and agent interaction are discussed.

3.1 Agents and their Responsibilities

Based on requirement to achieve goals as specified in goal diagram (created using agent tool [6]) shown in Figure 1, three agents are defined in MASAF model: Form agent, Record agent and Scrutiny agent. These three agents together perform the job of persons who are responsible for checking a list of application forms for validity of various fields by verifying certificates, marks lists and other documents and pointing out the discrepancies observed.

The Multiagent model of MASAF is shown in Figure 2. MASAF mainly consists of three agents but fourth agent gateway agent is kept as mediator between user and system thus it is not considered as part of the system itself. In the model the capabilities possessed by all the three agents are shown with 'possess' tag. Three agents are chosen because there are three cohesive tasks which need to be performed namely: retrieving applications, capturing knowledge about key fields and comparison of information. Individual responsibilities of these three agents are as follows:

Form Agent:

This agent retrieves the applications one by one from the website (in case third party is employed), or database of the organisation which is going to admit the students and save it in the database specifically created for MASAF. The key fields like roll number of qualifying examination, year of passing are communicated to the next agent i.e. record agent using Agent Communication Language (ACL) messages.

Record Agent:

Based on the key field values received from form agent the record agent logs on to relevant websites such as the board/university in which student studied. From these websites record agent captures knowledge about various attributes of the candidate like total marks, subjects studied etc. These attributes contribute to assignment of different weightages in the creation of merit list. The captured knowledge is kept in database as well as communicated to scrutiny agent using ACL messages.

Scrutiny Agent:

The scrutiny agent performs the comparison of information in the application forms, stored in local database, with that communicated by record agent. On the basis of comparison Scrutiny agent points out discrepancies (if any) present in the application forms.

Thus the scrutiny of applications by accessing relevant data, verifying and highlighting discrepancies is performed with the cooperative and joint efforts of all the three agents in MASAF.

Figure 1. Goal diagram for MASAF

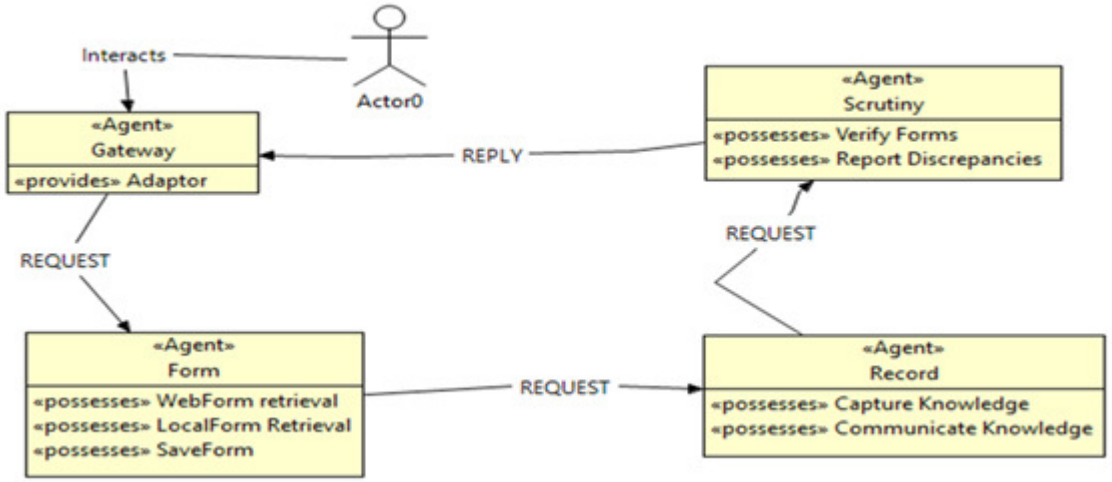


Figure 2. Agent Model for MASAF

3.2 Ontology Design

In the context of knowledge sharing and agent communication, the term ontology is used as a specification of a conceptualization. That is, an ontology is a description (like a formal specification of a program) of the concepts and relationships that can exist for an agent or a community of agents. In MASAF Ontology is stored in a frame-based knowledge model [7] .This model consists of classes i.e. concepts in the domain of discourse, slots i.e. properties or attributes of these classes and Facets which describe properties of slots. The design of the ontology for MASAF is called AcademicOntology. Conforming the model the following Agent Actions are defined:

- StartRetrievingForms, i.e. start retrieval of forms from website.
- StartRetrievingRecords i.e. start capturing knowledge about admission seekers.
- StartScrutiny i.e. start finding discrepancies in forms (if any)

The agent actions use both concepts and relations. The following primitive concepts are defined:

- Form: An entity which contains entire information about individual admission seeker.
- PrimInfo: Primitive Information about admission seeker required to capture knowledge about him/her from web.

- CourseWeightage: Contains information about which course has weightage related to which particular subject of previous class.

- Discrepancy: Contains information about kind of discrepancy (if any) present in filled form.

- CaptKnowledge: Contains knowledge captured about particular candidate from web.

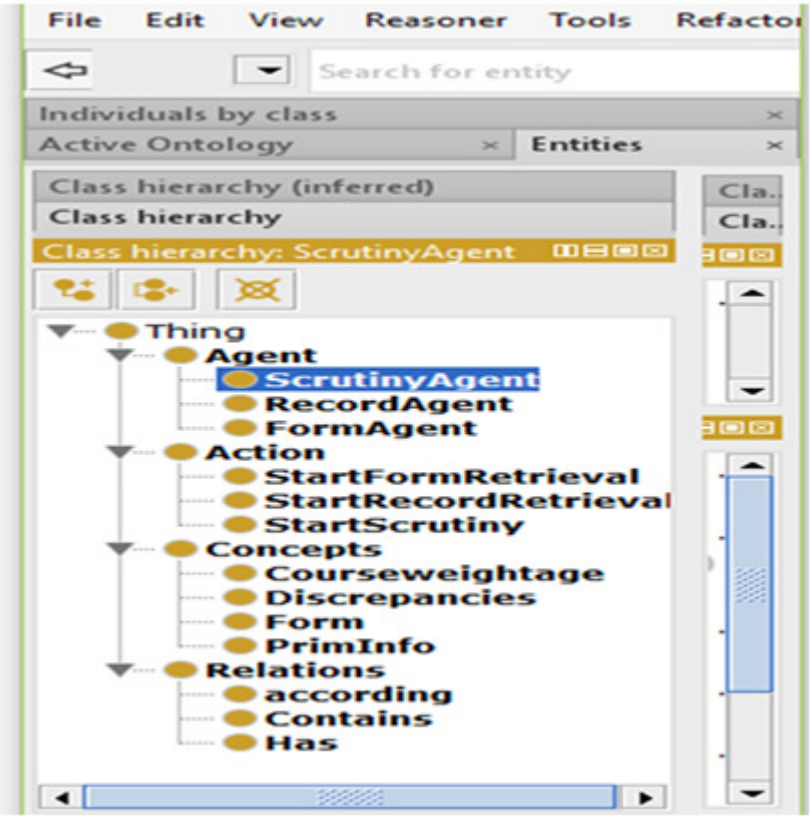


Figure 3. Partial Ontology for MASAF

The following relations are defined: 'According' between Form and Course i.e. form is given weightage according to particular course. 'Has' between Form and Discrepancy i.e. form has a particular discrepancy. 'Contains' between Form and PrimInfo. The ontology classes developed using protégé tool are shown in Figure 3.

3.3 Agent Design

The design of three agents internal to the system along with design of gateway agent which is external to the system is separately presented as follows:

Form Agent: Form agent is comprised of one shot behavior which is triggered whenever user asks to start the scrutiny. Form agent's one shot behavior when triggered goes to website or database of organisation to find out whether any new forms have arrived and if they have, it retrieves those forms and store them in system specific database. Form agent decides whether a

form is new or not by checking whether form with same form number is present in system's database already or not. However while retrieving forms form agent keeps sending values of key fields to record agent so that it can keep working in parallel.

Record Agent: This agent implements the cyclic behavior i.e. whenever form agent sends key fields this agent starts its work on the basis of these key fields. Firstly this agent on the basis of previous board/university wraps up the website of that board/university. Wrapping simply means controlling the website by means of code rather than by human [4]. After that it retrieves detailed record of the student from website on the basis of roll number. Detailed record retrieved contains knowledge on the basis of which scrutiny takes place. The attributes in detailed record include total marks, subjects studied, subjects passed and so on. In order to retrieve information about various attributes of student this agent has to parse the detailed record page of different board websites. Parsing is required because data presented in websites is semi-structured in nature. So agent makes use of defined algorithm to capture required knowledge from semi-structured information in web page. This knowledge is passed to scrutiny agent as well as saved in system's database for future reference.

Scrutiny Agent: This agent implements cyclic behavior, i.e. whenever request for scrutiny comes this agent executes behavior of verifying information in application form with that provided by record agent. If there are any discrepancies they are reported to the user and also saved in database for future reference.

Gateway Agent: This agent acts as a mediator between user interface and MASAF. This agent passes requests made on user interface to the system by converting the request into proper ACL messages and returns results to user interface from MASAF which are in different format than to be shown on user interface.

3.4 Agent Interaction

The three agents in MASAF communicate with each other using Agent communication Language (ACL) messages on the basis of predefined ontology. ACL is a specification by Foundation for Intelligence Agents (FIPA) for intercommunication of agents.

In MASAF model firstly gateway agent sends ACL message to form agent with intention REQUEST, action specifying 'StartFormRetrieval' and content containing ontology concept 'CourseWeightage' class which specify course for which scrutiny to take place.

Further Form agent sends ACL message to Record agent with intention REQUEST, action specifying 'StartRecordsRetrieval' and content containing ontology concept 'PrimInfo' which contains key fields.

In next interaction record agent sends ACL message to scrutiny agent with intention REQUEST, action specifying 'StartScrutiny' and content containing ontology class 'CaptKnowledge' which contains values of key attributes. On the basis of provided information scrutiny agent verifies application form and saves discrepancies and sends them to gateway agent as ACL message with content having 'Discrepancy' ontology class. The gateway agent finally retrieves information from this class and shows it on user interface. Interaction amongst agents expressed in Agent Unified Modelling Language is shown in Figure 4.

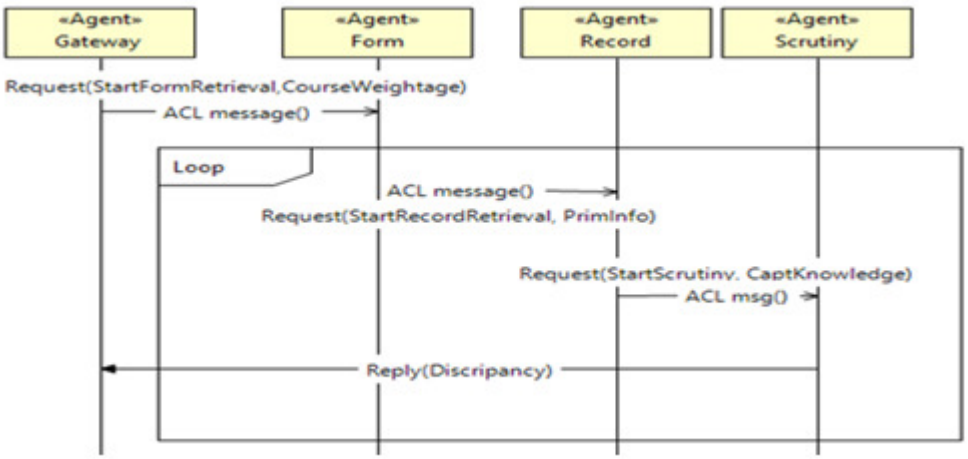


Figure 4. Interaction diagram of agents

4. IMPLEMENTATION

In section 4.1 tools and techniques used for creation of MASAF are discussed and in section 4.2 testing and results are presented.

4.1 Tools and Techniques

Agents are created in Java Agent Development Environment (JADE). User interface of the system is provided by means of Java Enterprise Edition (Java EE) Servlets. JADE Gateway agent is working as a mediator between Java EE user interface and multiagent system created on JADE platform. Agents Communicate with each other using messages created using Agent Communication Language (ACL) framework which is available with JADE.

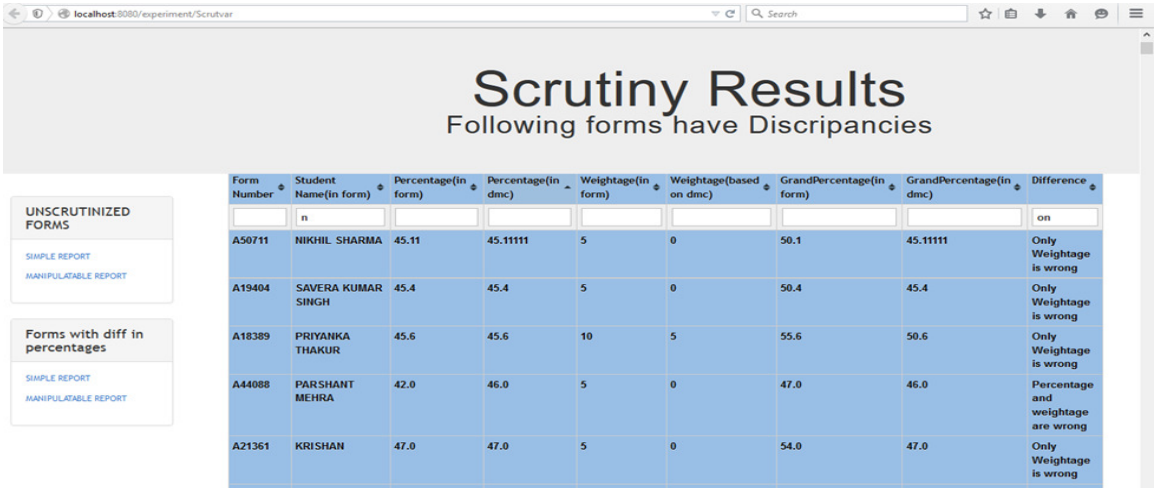


Figure 5. Results of Scrutiny

In order to retrieve data from websites both form and record agent use similar technique. This technique involves wrapping the website [4] and parsing the semi-structured data in HTML and

retrieving the required information from it. Tool used for wrapping the websites is java package Selenium. For efficient parsing of semi-structured data JavaSoup package for java is used. The user interface provided by the system is web-interface.

Thus requests can be made to system and results can be seen from system through a web browser. Agents run in the backend on the server in order to deliver the results on the user Interface. A screenshot of User Interface is shown in Figure 5. Analysis and design of MASAF is depicted using agenttool [6]. Ontology is developed using open source software tool protégé [7].

4.2 Testing and Results

MASAF is tested for the admissions of Bachelor of computer Applications (BCA) an undergraduate course run in Government colleges, Chandigarh, India. 997 student applications are processed by the system and discrepancies are highlighted correctly. MASAF with the help of Form Agent read all the filled in forms and saved them in local database according to pre-specified ontology. After retrieving necessary information from the form, record agent retrieved information of all admission seekers from three different online sites, namely: Central board of School Education (CBSE), Haryana Board of School Education (HSEB), and Punjab School Education board (PSEB).

The sample results of scrutiny are shown in Figure 5 highlighting the discrepancies. For example student with form number ‘A50711’ and name Nikhil sharma has filled different weightage in form than captured from the web. Weightage (in form) column specifies weightage mentioned in the form, Weightage (based on dmc) column specifies weightage on the basis of knowledge captured from the website and discrepancy is specifically mentioned in difference column as for this instance ‘only weightage is different’.

5. CONCLUSIONS AND SCOPE OF FUTURE WORK

Multiagent system due to the properties of proactive, reactive and social ability can perform a number of knowledge intensive tasks without much human intervention. The MASAF performed scrutiny of 997 applications in about one hour which otherwise would have completed by seven persons in one week. Thus the system is dependable and efficient. The system can be extended to rectify any discrepancies if found in forms thus generating the merit list automatically. For this correction agent could be introduced with required functionality which would remove all the discrepancies in the forms.

REFERENCES

[1] Cantú j, Ceballoss G. “A multiagent knowledge and information network approach for managing research assets”. Expert Systems with Applications. Elsevier 2010; 37:5272–5284.

[2] Oprea M. “MAS_UP-UCT: A Multi-Agent System for University Course Timetable Scheduling”. International Journal of Computers, Communications & Control 2007; 2:904-102.

[3] Taskin H, Gozetepe K, Taskin M, Canay O. “Agent Based University Planning of Sakarya University: “CAWIS” Project”. Journal of Theoretical and Applied Information Technology 2008; 4:753-759.

[4] Raposo J, Pan A, lvarez A, Hidalgo J. “Automatically maintaining wrappers for semi-structured web sources”, Data & Knowledge Engineering 61 (2007) 331–358

[5] Addis A, Armano G, Vargiu E. "Multiagent systems and information retrieval our experience with X.MAS". Expert Systems with Applications. Elsevier 2012; 39:2509-2523.

[6] Juan C. Garca-Ojeda, Scott A. DeLoach, and Robby, agentTool III: From Process Definition to Code Generation, Proc. of 8th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2009), Decker, Sichman, Sierra, and Castelfranchi (eds.), May, 10–15., 2009, Budapest, Hungary, pp. 1393-1394.

[7] Gennari J, Musen M, et al. "The evolution of Protégé: an environment for knowledge-based systems development". International journal of Human-Computer studies 2003; 58(1):189-213.

[8] Tu SW, Eriksson H, Gennari JH, Shahar Y, Musen MA. "Ontology-based configuration of problem-solving methods and generation of knowledge-acquisition tools: application of PROTEGE-II to protocol-based decision support." Artif. Intell. Med. 1995 Jun; 7(3):257-89.

AUTHORS

Puneet Modgil is a research scholar in the Department of Computer Science and Applications, Panjab University (PU), Chandigarh. He has served as Assistant Professor in Government College, sector - 11, Chandigarh, and Panjab University, Chandigarh. His Academic qualifications include M.Sc (Information Technology) M.Phil (Computer Science and Applications). He has teaching experience of 10 years.

Dr. M. Syamala Devi is a professor in the department of Computer Science & Applications, Panjab University, Chandigarh. She received her Ph.D degree in Computer Science & Systems Engineering from Andhra University, Visakhapatnam and M.E. in Computer Science & Engineering, from NIT, Allahabad. She has completed M.Sc in Applied Mathematics from Andhra University, Visakhapatnam. Before joining Panjab University, she served Indian Space Research Organization, Sriharikota, and National Institute of Technical Teachers' Training & Research, Chandigarh. Her areas of expertise include algorithms, Image Processing, Distributed Artificial Intelligence and Educational Computing.

INTENTIONAL BLANK

COMPUTER VISION BASED SMOKE DETECTION METHOD BY USING COLOUR AND OBJECT TRACKING

Ung Hwang and Jechang Jeong

Department of Electronic and Computer Engineering,
Hanyang University, Seoul, Korea
hwangung@gmail.com, jjeong@hanyang.ac.kr

ABSTRACTS

To reduce the damage from fire disaster, demand for automatic detection system by using computer vision technique is increasing. But because of false detections that are caused by various situations, it is hard to use in real. In fire detection area, to overcome this problem, the algorithm using several temporal and spatial information of object is proposed. Colour, brightness, and movement information is used to make relevant smoke detection algorithm. And continuous monitoring for several constraints is used to avoid false detections that are caused by unexpected behaviour of object. In experimental result, total 11 videos that have smoke and the other 2 videos that have no smoke is used for test performance of proposed method. It shows relevant performance for false detection and could detect almost fire in video.

KEYWORDS

Smoke Detection, Computer Vision, Object Tracking, Pattern Recognition

1. INTRODUCTION

In the research report of South Korea in 2014 [1], most of the fire accidents are unpredictable and approximately10% of them is caused by unknown source. So there is need to find unexpected fire by unmanned machine for early alarming to reduce the damage of resource. In Korea, there are several researches on early detection system for automatic disaster alarming system. The reason that they were not detected early is the detection system is not fully installed in certain area, and the degree of completion of disaster focused detection algorithm is not quiet qualified to use it in real monitoring system. Various fire detection algorithms were developed to detect fire. They developed find the features of fire and smoke by various methods [2-11].

Fillips et al. proposed colour based fire detection algorithm that use difference of brightness of pixels [2]. During fire consume, fire shows wide brightness range and its brightness changed always. They use this temporal difference of pixel brightness to find fire. Chen et al. proposed algorithm that use colour range. It include equation like “Red > Green > Blue”, and R has very high value. Basic colour of fire is good feature to find fire, because its colour is quiet strong that little hard to see in everywhere.

David C. Wyld et al. (Eds) : ITCS, JSE, SIP, ARIA, NLP - 2016
pp. 53–59, 2016. © CS & IT-CSCP 2016 DOI : 10.5121/csit.2016.60306

When people find fire, they usually see smoke first. So smoke detect algorithm is better than fire detection algorithm for early alarming. Fire detection system usually uses both smoke and fire detection method, but smoke detection algorithm is more important for long range smoke detection, because fire usually detected by people who saw smoke that generated from the fire. By this reason, the machine monitor fire instead of human eyes also need to have more sensitive algorithm for early fire detection.

Toreyin et al. proposed smoke detection algorithm that can use temporal variation of wavelet domain. They use background of video, edge, and colour to determine the features of smoke [4]. Chen proposed a method for smoke detection algorithm by using the total number of smoke pixel that is determined by pixel differences between frames and chromatic value model of smoke [5].Yuan proposed motion based smoke detection algorithm is specially concentrated on upward moving of smoke[6]. It uses chromatic model to define smoke pixel and the accumulation of motion to distinguish between smoke and non-smoke moving object.

Smoke has similar colour and movement with non-smoke object, therefore conventional smoke detection algorithms usually have false detections. It is necessary to define more characteristic of smoke. Mainly this paper consist of two parts, one is for long distance detection, and the other one is for short range. For long distance smoke detection, Chen et al.'s method is used that count all smoke picture in the frame [4]. Smoke pixel is defined empirically in specific RGB colour range that can find all smoke in test video. For short range smoke detection, more specific moving detection algorithm is applied. These concept is deeply covered in section 2.2-2.4. Experiment results are showed in section 3.

2. PROPOSED SMOKE DETECTION ALGORITHM

2.1 Pixel based smoke detection by using colour and brightness

Fig.1. Long distance smoke

When fire occur, long distance smoke has a little difference between the frames even wildfire is occurred. So we have to use more sensitive detection method than short distance smoke detection. Block based algorithm that is used for fire detection is easy to analyses local area and

classify each local area's characteristic for smoke detection. But it is hard to detect small amount of variation of frame. So in this paper we use pixel based detection algorithm to find long distance fire detection. First we find pixels that satisfy next equation:

$$\left|Y_{x,y,n-1} - Y_{x,y,n}\right| < TH_y \tag{1}$$

where Y_n is brightness of pixel of nth frame. TH_y is empirically selected as 20.

In human eyes, normally smoke show achromatic colour that has similar r, g, and g value. But in videos, smoke does not has achromatic colour always. So smoke colour model is designed to can include all kinds of smoke colours in videos. The equation is shown in eq. (2). Smoke normally shows achromatic colours, but these kinds of colour are easily found on everywhere.

Black car, grey shirt and white wall are example of potential smoke candidates in the view of smoke colour. So colour based smoke detection is not efficient itself, So motion or appearance based detections have proposed definition by colour more than fire, monitored behaviours of smoke is used to detect fire in this paper.

$$\begin{aligned} \left|\mathrm{R_{x,y}} - \mathrm{G_{x,y}}\right| &< \mathrm{TH_{CS}} \\ \left|\mathrm{G_{x,y}} - \mathrm{B_{x,y}}\right| &< \mathrm{TH_{CS}} + 80 \\ \left|\mathrm{B_{x,y}} - \mathrm{R_{x,y}}\right| &< \mathrm{TH_{CS}} + 80 \end{aligned} \tag{2}$$

2.2 Object based smoke detection

To partition each object, labelling algorithm distinguish each object is used. And for relevant partitioning, open and close filter which use min and max filter are used. The open filter is the filter which uses min filter after max filter. The close filter is opposite. By using open filter, it is easy to remove noise pixels that has smoke feature in the frame without big data losses of original shape. By using close filter, divided smoke pixel can be joined as one object. Fig 2 shows open and close filter reduced noise and centralized and organized smoke. By this partitioning object method, groundwork for using method of Section 2.3 is ready.

2.3 Behaviour based smoke detection

For long distance detection, there are not many objects that cause false detection. But for short distance detection, there are many things to exclude to find the real smoke. One of major cause is smoke has colour that can be easily found everywhere, so colour based smoke detection method cannot occupy big portion of smoke detection algorithm.

Smoke shows uncommon behaviour that becomes large and goes upward. These kinds of moving are used to detect smoke in [5-6]. These principles of smoke detection are very helpful to find smoke and this concept is also applied in the proposed algorithm. But if there is object that has same colour and do same move (going upward in frame), that object is easy to occur false detection. To avoid this, in this proposed algorithm, there are some constraints that can remove false candidate of smoke by finding objects that have not smoke behaviour.

(a) (b)

Fig 2. (a) Smoke pixel image (b) Filtered image by using open and close filter

2.3.1 Natural smoke attribute

Smoke is diffused well because it has high temperature, and burning materiel makes smoke continually. In [1], they concluded that smoke is in the video if smoke pixel in the image increase continually for a moment. In this paper, long distance smoke detection use this method. And smoke pixels have similar brightness between near pixel. Its colour is not changed immediately like fire (fire shows various colour) in the small region. So if some object has various colour in a same object and each near pixels brightness are not similar each other, those object is not consider as same object. In [2], they use this attribute to find a smoke.

2.3.2 Estimate smoke movement by using camera position

Usually to find smoke, camera is installed in higher position than monitored area. So if some object is coming to under camera, it become bigger in frame, but its position become lower continually. So we can assume it is not smoke, but another object.

2.3.3 Position of fire

We assume fire is located in camera frame or under camera frame. So the object that comes from side of frame is not considered as smoke. And position that smoke appear first is should be remain for a moment. Because we assume fire become bigger and bigger, position that smoke appear first is should be maintained and not moved if smoke is being generated.

2.3.4 Sudden changing

We assume to use 15-30 f/s camera for fire detection. This is very short time to make big difference in frame. So we can make threshold to limit the size changing of object or moving distance of object. Normal object cannot go far within 1/30 second and even smoke cannot bigger than specific amount. So unacceptable cases are considered as error and ignored.

2.4 Final Decision

In the final decision, alarm can occur if all condition is satisfied. The object continually become bigger for a second, and that satisfy rule of 2.3 is decided as smoke object and alarm occur immediately.

Fig 3. Block diagram of proposed algorithm

3. EXPERIMENTAL RESULT

In Fig. 4, there are 11 videos that have smoke and 4 videos that do not have smoke. The main causes of false detection in this proposed algorithm is the object in front of smoke. Because the proposed algorithm uses object based detection, if shape of smoke is ruined then proposed algorithm can makes error. Usually smoke is on the sky and there is nothing between smoke and detector, so camera installed in proper position, it can overcome. The other false detection is caused by two object that has similar colour. If they meet and divide frequently, the proposed algorithm cannot distinguish there two object and they become one object (become bigger than before). The proposed algorithm only make false detections for these situation, so it will be the future work.

Table 1. Experimental result.

Video	Detection	Comment	Number of false detection
Smoke 1	True	-	0
Smoke 2	True	-	0
Smoke 3	True	-	0
Smoke 4	True	-	0
Smoke 4	True	-	0
Smoke 5	True	-	0
Smoke 6	True	-	0
Smoke 7	False	Cannot detect smoke in the sky, (Too little amount of smoke, similar colour with sky)	0
Smoke 8	False	Smoke is divided two because of an object in front of smoke	0
Smoke 9	True	-	0
Smoke 10	True	-	0
Smoke 11	True	-	0
Non-smoke 1	True	-	0
Non-smoke 2	True	-	0
Non-smoke 3	False	Smoke divide two because of an object in front of smoke	2
Non-smoke 4	False	Two object shows similar characteristics. They united and divide frequently.	5 (long size video)

Fig 4. Experimental Result – First 4 Video has smoke, last 4 video do not has smoke

3. CONCLUSIONS

There are many methods to detect fire for early detection that can reduce damage of accident. In this paper, smoke detection method is proposed to decrease the false detection for better use in real. To reduce the false detection rate, we use labelling and object tracking that can monitor behaviour of each object. By monitored data, smoke object is determined as smoke when all movement and characteristic are satisfied to decide. By using proposed algorithm, false detection rate is lowered than before, however in some unexpected condition there are still some errors. If we can use better object tracking method and computing power is enough to use this, it can be covered in future work.

ACKNOWLEDGEMENTS

This work was supported by Institute for Information & communications Technology Promotion(IITP) grant funded by the Korea government(MSIP) (No.2015-B0101-15-1377,Development of 16-channel Multi-viewer, H/W Encoder, and Transcoder Systems for 4K UHD Broadcasting)

REFERENCES

[1] Fire report in Korea Oct / 2014, http://usafe.namwon.go.kr/boardread_ifr.asp?g=1&idx=754&cnt=9.

[2] W. Phillips Iii, M. Shah, and N. da Vitoria Lobo, "Flame recognition in video," Pattern recognition letters, vol. 23, pp. 319-327, 2002.

[3] T.-H. Chen, P.-H. Wu, and Y.-C. Chiou, "An early fire-detection method based on image processing," in Image Processing, 2004. ICIP'04.2004 International Conference on, 2004, pp. 1707-1710

[4] B. U. Toreyin, Y. Dedeoglu, and A. E. Cetin, "Contour based smoke detection in video using wavelets," European Signal Processing Conference, pp. 123-128, 2006.

[5] T.-H. Chen, Y.-H. Yin, S.-F. Huang, and Y.-T. Ye, "The smoke detection for early fire-alarming system based on video processing," in Intelligent Information Hiding and Multimedia Signal Processing, 2006. IIH-MSP'06. International Conference on, 2006.

[6] F. Yuan, "A fast accumulative motion orientation model based on integral image for video smoke detection," Pattern Recognition Letters, vol. 29, pp. 925-932, 5/1/ 2008.

[7] Y. Hakan Habiboglu, O. Gunay, and A. Enis Cetin, "Real-time wildfire detection using correlation descriptors," in Signal Processing Conference, 2011 19th European, 2011, pp. 894-898.

[8] A. Benazza-Benyahia, N. Hamouda, F. Tlili, and S. Ouerghi, "Early smoke detection in forest areas from DCT based compressed video," in European Signal Processing Conference, 2012, pp. 2752-2756.

[9] M. Stula, D. Krstinic, and L. Seric, "Intelligent forest fire monitoring system," Information Systems Frontiers, vol. 14, pp. 725-739, 2012/07/01 2012

[10] B. Ko, J. Park, and J.-Y. Nam, "Spatiotemporal bag-of-features for early wildfire smoke detection," Image and Vision Computing, vol. 31, pp. 786-795, 2013

[11] D. Krstinić, D. Stipaničev, and T. Jakovčević, "Histogram-based smoke segmentation in forest fire detection system," Information Technology and Control, vol. 38, 2015.

INTENTIONAL BLANK

AUTOMATIC KURDISH DIALECTS IDENTIFICATION

Hossein Hassani[1,2] and Dzejla Medjedovic[3]

[1]Department of Computer Science and Engineering,
University of Kurdistan Hewlêr, Erbil, Kurdistan Region, Iraq
[2]Department of Computer Science, Sarajevo School of Science and Technology,
Sarajevo, Bosnia and Herzegovina
hosseinh@ukh.edu.krd
hossein.hassani@stu.ssst.edu.ba
[3]Department of Computer Science, Sarajevo School of Science and Technology,
Sarajevo, Bosnia and Herzegovina
dzejla.medjedovic@ssst.edu.ba

ABSTRACT

Automatic dialect identification is a necessary Language Technology for processing multi-dialect languages in which the dialects are linguistically far from each other. Particularly, this becomes crucial where the dialects are mutually unintelligible. Therefore, to perform computational activities on these languages, the system needs to identify the dialect that is the subject of the process. Kurdish language encompasses various dialects. It is written using several different scripts. The language lacks of a standard orthography. This situation makes the Kurdish dialectal identification more interesting and required, both form the research and from the application perspectives. In this research, we have applied a classification method, based on supervised machine learning, to identify the dialects of the Kurdish texts. The research has focused on two widely spoken and most dominant Kurdish dialects, namely, Kurmanji and Sorani. The approach could be applied to the other Kurdish dialects as well. The method is also applicable to the languages which are similar to Kurdish in their dialectal diversity and differences.

KEYWORDS

Dialect identification, NLP, Kurdish language, Kurmanji, Sorani

1. INTRODUCTION

Dialectology has not received a considerable attention in Computational Linguistics (CL) and Natural Language Processing (NLP). However, it has been part of sociolinguistics and linguistics in general for a long time. For example, dialectal classification study for English [1] and other European languages started several decades ago. It has produced different results such as preparation of "Atlas Linguarum Europae" and "European linguistic map" [2]. Although the situation for the languages such as English and German that have been the focal subject of CL, does not show any radical change with regard to computational dialectology, it seems to be

David C. Wyld et al. (Eds) : ITCS, JSE, SIP, ARIA, NLP - 2016
pp. 61–78, 2016. © CS & IT-CSCP 2016 DOI : 10.5121/csit.2016.60307

slightly shifting for other languages. To illustrate, recent works express some interests in computational dialectology for languages such as Arabic and Chinese [3, 4].

Several reasons could have caused the mentioned "ignorance". First, a language such as English, which has received a high degree of attention from computational perspective, has a standard (or two main standards: American and British) format. Neither the speakers nor the readers of several dialects of this language have serious difficulties in understanding each other. Second, the language is written with one script and follows a standard orthography. Third, the dialects are constructed on a common architecture of the language with insignificant variation in their structures.

Although other reasons could be counted, for this research, these three can justify the necessity for the development of the methods and the tools, as part of the Language Technology (LT), for a multi-dialect language such as Kurdish for which one or more of the mentioned reasons are not applicable.

The LT follows the same principles that the technology generally follows in almost all cases. That is, its development and improvement are rooted in different needs such as usual day-to-day, sociological, and business related requirements. By the same analogy, it is also developed and evolved based on scientific, experimental, and laboratory demands that usually happen in the research activities. This might justify why the dominant languages in NLP have not attracted such attention, while other languages such as Chinese and Arabic have noticeably been of interest for dialectal studies from computational perspective.

The same analogy could be used for Kurdish, which is the language of Kurds. Kurdish is spoken in divergent dialects. The speaking population of this language is uncertain. There is a discrepancy in the reports of population ranging from 19 million [5] to 28 million [6]. In fact, the Kurds' lives as well as their language have been extremely affected by their political and geopolitical situation. Indeed, they have remained marginalized geographically, politically, and economically, during the last two centuries [7]. Consequently, the negligence of their language has been propagated to many other fields. Therefore, it is not of any surprise to see that the Kurdish computational study is not by any mean an established sector of CL and NLP.

Diversity of Kurdish dialects, grammatical distances, vocabulary differences, and mutual unintelligibility are some main factors in guiding most of the scarce CL and NLP research and development into two main dialects of Kurdish, namely Kurmanji and Sorani. In fact, in most cases for the reasons that we will address in section 3, these activities have focused on Sorani alone. This research has also focused on the two stated dialects for the same reasons. Regardless of the number of the dialects, dialect identification is a requirement in Kurdish CL and NLP. The reason is, when one wants to perform a computational process on Kurdish text, such as Machine Translation (MT), sentiment analysis, Part of Speech Tagging (POST) or a similar process, one cannot proceed without being aware of the dialect of the context. We will discuss this case in more detail in the following sections.

The rest of the article has been organized in five sections. The first section discusses the dialects, languages and their relations. The second section provides an overview of Kurdish, its dialects, scripts, grammar, orthography, and some linguistic aspects of the language. The third section explains the related works and the methodology of the research. The forth section discusses the

experiments, the results and their analysis, and some found issues. The Conclusion section summarizes the findings and suggests some areas that need more study in the future.

2. DIALECTS AND LANGUAGES

Although the definition of the language seems to be clear, when one wants to distinguish between "dialects" and "languages", the borderlines seem to be blurred [8]. Linguists have different opinion about dialects and languages. Nevertheless, they mostly agree on referring to two sets of criteria, one social and the other political, based on which one can distinguish whether what is spoken in a specific community is a dialect or a language [9].

Despite the fact that the similarity in the vocabulary, the pronunciation, the grammar, and the usage are important parameters in making distinction between dialects and languages, the central concept of this distinction is suggested to be the concept of mutual intelligibility [9]. That is, if two dialects are mutually unintelligible, they are considered two different languages, otherwise, two dialects of the same language. However, there are dialects that are mutually intelligible which are considered languages and there are others which are mutually unintelligible yet considered dialects of a language [9].

In the case of Kurdish dialects, the definition is arguable from some linguists' point of view [10]. This situation makes research activities on Kurdish dialects rather challenging. This is because although the basis of CL is common among dialects and languages, it might significantly affect the research approach and methodology according to whether you tackle the problem area *interlingually* or *intralingually*. The following section provides more information on this subject.

3. KURDISH LANGUAGE

Kurdistan is the homeland of the Kurds. This is a region located across Iran, Iraq, Turkey, and Syria. The term Kurdistan has been used to name a province or to call a wider region, both in Iran and Iraq. For some political reasons, the case is different in Turkey and Syria. Kurdish people are sometimes called "a nation without state" [11]. However, the Kurds have been in the middle of many battles over the past centuries, hence, their geopolitics situation has always been a matter of concern to the world's policy.

Surprisingly, this situation has not benefited them as much as it has the other communities with the similar status. For example, regardless of the utmost cruelty that happened, several countries received benefits out of both World Wars, while the Kurdish population never received such benefits until recent times. As a result, this situation has affected the Kurdish usage and its popularity as well. It seems that the circumstances are going to be different since the Iraqi Kurdistan region has started to have its regional government under the new federal Iraq.

In the following sections a background on Kurdish, its dialects, scripts, orthography, and its current situation with regard to CL and NLP will be discussed.

3.1. Overview

Kurdish is the name given to a number of distinct dialects of a language spoken in the geographical area touching on Iran, Iraq, Turkey, and Syria. However, Kurds have lived in other

countries such as Armenia, Lebanon, Egypt, and some other countries since several hundred years ago. They also have large diaspora communities in some European countries and North America.

There are some opinions about the Kurdish root that state that the Kurds have come from different origins, that they have changed their language, and their first language was rather different from the current one [12]. However, those who believe in this theory do not make this clear that how people from different origins have spoken an unknown language and why they have changed it. More accurate figure on this has been given by [13].

Kurdish studies, though not very popular, has an almost a century of history. McCarus provides an informative background on Kurdish studies. Although his work dates back to the 1960s, it still can be seen as a major resource about the Kurdish studies [14]. A very recent finding based on a different approach "to try to prove with inter-disciplinary scientific methods explained, that indigenous aborigine forefathers of Kurds (speakers of the 'Kurdish Complex') existed already B.C.E. and had a prehistory in their ancestral homeland (mainly outside and Northwest of Iran of today)" [15]. However, research on Kurdish has been biased for different reasons.

Kurdish language includes different dialects. Dialect diversity is an important characteristic of Kurdish. Kurdish is written using four different scripts. The popularity of the scripts differ according to the geographical and geopolitical situations. There is no consensus among the Kurdish linguists upon the number of letters in the Kurdish alphabet. Latin script uses a single character while Persian/Arabic and Yekgirtû in some cases use two characters for one letter. The Persian/Arabic script is even more complex with its right-to-left and concatenated writing style.

Kurdish is spoken in different dialects, which are not following the same grammar. The level of the differences vary for every pair of dialects. I addition, an important feature of current Kurdish is the lack of a standard orthography [16].

The above brief overview shows the complexity of Kurdish from different perspectives. This complexity, particularly, affects the language computation, which in turn makes hindrance in front of studying Kurdish in the context of CL and NLP. It also makes the development of LT for this language rather challenging. Below we discuss these issues in a more detail.

3.2. Dialects

As it was mentioned, Kurdish is a multi-dialect language. Since the 1960s several major scholars, including Westerns and Kurds have published influential research outcomes about Kurdish and its dialects [16, 19–21]. But neither the nomenclature of these dialects have been standardized nor there is a solid agreement on their relation to the language [13, 22, 23].

In a recent research, Haig and Öpengin identify the Kurdish dialects as Northern Kurdish (Kurmanji), Central Kurdish (Sorani), Southern Kurdish, Gorani, and Zazaki. For each one of these dialects they mention the main sub-dialects [18]. The populations that speak different dialects of the language differ significantly. The majority of Kurmanji speakers are located in different countries, such as Turkey, Syria, Iraq, Iran, Armenia, and Lebanon, just to name the main lands. The second popular dialect is Sorani, which is mainly spoken among Kurds in Iran and Iraq. Zazaki is spoken in Turkey. Gorani is primarily spoken in Iran and Iraq [16, 21]. In

addition, as a result of long-term conflicts in the region, Kurds also have a large diaspora community in different western countries, where almost all Kurdish dialects are spoken to different extents.

It is worth mentioning that as Leezenberg describes, the reason that we stick with the internationally accepted categorization of the language is to keep the harmonious environment of the scholarly research [24], while we are aware and actually prefer, at least due to the local usage, to use Hawrami, or Hawramani, instead of Gorani, in most of the situations. In fact, as we have observed, sometimes, the term Gorani, is rather restricted, if it is not considered as unknown at all, in most of Sorani and Hawrami dominated areas.

3.3. Alphabet and Scripts

Kurdish is written using four different scripts, which are modified Persian/Arabic, Latin, Yekgirtû(unified), and Cyrillic [25]. The popularity of the scripts differ according to the geographical and geopolitical situations. There is no consensus among the Kurdish linguists upon the number of letters in the Kurdish alphabet. The main reason for the disagreement seems to be mainly on the phonetic aspects (and to a great extent acoustic features) rather than lexical aspects, though clearly these two affect each other. For example, Bedir_Xan and Lescot suggested 31 letters in their Latin script proposal for Kurmanji, arguing that the Kurdish did not have a separate sound to distinguish between 'خ' and 'غ', hence in their Latin script they used the letter "x" for both sounds [26]. However, these two sounds are written with two different letters in Persian/Arabic and Yekgirtû scripts. As a result, some sounds are lost if an utterance is captured using Latin script. In order to address this issue, the current Latin script has been augmented to capture the mentioned sounds [25]. Nonetheless, as an advantage, Latin script uses a single character while Persian/Arabic and Yekgirtû, in some cases, (e.g., 'وو' in Persian/Arabic and 'sh' in Yekgirtû for 'û' and 'ş' in Latin, respectively) use two characters for one letter. Although Yekgirtû is phonetically more complete (it includes 37 "letters"), its double character representation for a single phoneme makes it computationally more difficult. The Persian/Arabic script is even more complex with its right-to-left and concatenated writing style.

Latin script, mainly, is used for writing in Kurmanji dialect. But this is not applied for Kurdish communities in Armenia and former Soviet countries, whom they use Cyrillic script. Furthermore, until recently, the Kurmanji community of the Iraqi Kurdisatn, was mostly using modified Persian/Arabic script. For Sorani, the main script is modified Persian/Arabic. Zazaki, mainly, is written in Latin. Gorani (Hawrami) is, mainly, written in modified Persian/Arabic. We stressed on word "mainly", because there are considerable exceptions in the usage of these scripts, particularly Latin and modified Persian/Arabic. The former is used in Turkey, because the Kurd community is already familiar with the script through Turkish. In Iran, Syria, and Iraq, the dominance is with the Persian/Arabic script. The reason is obvious. Persian is the national and formal script in Iran, and Arabic has been the national and formal script for Iraq and Syria. Generally speaking, Persian/Arabic script has a longer history in writing Kurdish, while Latin script was suggested and introduced by Mir Celadet Bedir-Xan around 1930s [27].

However, in the recent years the situation has been changing. That is, the Latin script is growing in the usage and is becoming more popular in the areas that it was not before. But it is not the same for Persian/Arabic or Cyrillic script. That is, the Persian/Arabic has been continuing to be the dominant script in the areas that it was and the Cyrillic script has been restricted to the

communities in Armenia and the former Soviet countries. As an example for the latter, Persian/Arabic script is the official script in the Iraqi Kurdistan Region, though the usage of Latin script is growing, particularly, by different Kurdish media.

3.4. Grammar and Orthography

Despite having the same root, Kurdish dialects grammatically differ from each other. The differences are vary in terms of grammatical features and the level that they differ [16–18]. In some cases the grammatical differences are trivial, while in some others they are considerable. We show this with two samples.

As the first sample, Sorani speakers do not apply gender differentiation, while Kurmanji applies gender. To be more precise, there are restricted sub-dialects, which is spoken by a small community of a population of less than a few thousand people. The authors, in their research, have recently come across a small Sorani speaking community, where the gender is used, to just differentiate between male and female human-being. Indeed, Hassanpour has already addressed the issue of genders and its usage in some sub-dialects of Sorani [16], which does not seem that is in use by the current speakers anymore.

In another case, the authors learned that similar situation is true for one of the sub-dialects of Laki, which is called Jafar-aabadi. Laki, itself, is a sub-dialect of Southern Kurdish, which in general does not include genders. However, the authors were told, by an informant about the dialect, that Jafar-aabadi speakers use gender, not only for human-being but also for other subjects. As an example, in the small community where this dialect is spoken, the Moon is masculine. The reason that is mentioned for this assignment is that the Moon dares to come out at nights, while the Sun is feminine, because it comes out during the day. Further investigation on this case is a linguistic endeavor. However, authors are following the case as it is related to their other areas of research in CL.

The second sample is the negation, where in Kurmanji one says "**ne** li nêzîkê", which means "it is not close", while in Sorani it is said a "le nêzîk **niye**" (The negations were shown in bold). These examples show the difficulty of dealing with Kurdish as "a language" and not as "a group of languages".

Kurdish has different issues from an orthographic point of view. First and foremost, there is no standard orthography for Kurdish. Hassanpour gives a brief history of how an orthography was suggested in Iraq based on the Arabic language alphabet and the challenges that it faced during the 1920s [16]. Finding the reasons for why a language that is spoken by a massive population had not have its own orthography, which in turn sparkles other questions such as why the written Kurdish only has a history of no more than a few centuries, is not an easy task.

For example, some sources, orally, have talked to one of the authors about the correspondence between the Arab army and Kurds defending Banah (a Kurdish city located in the Kurdistan province in Iran) around the 670s. These sources based their "story" upon information that they have received from a descendant family, whom were involved in that correspondence. Although the author could not ascertain the case at this stage, it was worth mentioning for further investigations. Even if this story is not true, it is still not clear what kind of alphabet and orthography have been used by the Kurds at the time. Although this is basically a matter that is

related to linguists and historians, if we find some reliable answers, we will share it with the interested researchers.

3.5. Kurdish and Language Technology

In this research we use the "computationally-enabled" as a technical term to distinguish the languages that are enabled with the minimum tools of Language Technology, which in turn allows those languages and their products, whether in written or spoken format, to be processed by computers. Although there are frameworks for this accounting and assessment such as BLARK (The Basic Language Resource Kit) [28], this is beyond the scope of this article. We just mention the case in the capacity of the current article.

Despite having a large speaking population, there is no or limited computational research with regard to Kurdish. Indeed, a simple search on the Internet regarding computational activities on the language provides no more than a few results, which either they are at the preliminary stages of the study, or they cover very specific areas such as text to speech concepts, to discuss some limited corpus, or to provide some comparison between the dominant Kurdish dialects.

Moreover, even this small amount of research mostly covers one dialect of Kurdish, which is Sorani. Therefore, currently Kurdish cannot be considered as a computationally-enabled language. Although Hassani and Kareem discuss the case with regard to assistive technologies for Kurdish [29] and there are also other appreciable attempts by some scholars which have taken place [30], the overall figure has neither been progressing significantly nor is promising. In spite of the fact that there are some evidence showing a slight growth in the interest in this area (for example, see [31]), to become a computationally-enabled language, Kurdish needs extensive scholarship and professional efforts.

3.6. Current Situation

As a consequence of the establishment of the Iraqi Kurdistan Regional Government, Kurdish has become one of the two official languages. This has been declared under Article 4 of the Iraqi Constitution [32]. Neither this article nor other part of the constitution specifies a particular dialect of Kurdish in this regard. Similar approach has been followed in the Draft Constitution of Kurdistan Region, which has been approved by the Parliament of Kurdistan Region [33, 34] (The Draft Constitution of Kurdistan Region is in Arabic and Kurdish; a translation into English can be found here [35]). The document can become official if it is approved in a referendum, which has not been held yet.

As a result of the above steps, Kurdish has become the main teaching medium for the entire pre-university education. Even though there are some exceptions for the private schools, which might use some foreign languages such as English or French, in these schools too learning Kurdish is an obligatory educational element. Furthermore, the language is used to a varying extent in most of the universities in the region as well.

However, the decision on making a dialect official depends on the population who speak the dialect in the specific area/governorate. For instance, Kurmanji is the official dialect for communication and education (up to the end of high school) in Duhok governorate of the Iraqi Kurdistan Region, while in the other two governorates Sorani dialect plays the same role.

There is no precise demographic report accessible to show the population who speak different dialects, but the figure can be loosely extracted from the population who live in different governorates. A report on the Iraqi Kurdistan Population Forecast for 2009-2020 period shows that about 26% of the Iraqi Kurds are currently living in the Kurmanji dominant areas [36]. These demographic facts vary significantly in different countries where Kurds live. As an illustration, the figures for the language shows that Kurmanji is spoken by around 20 million Kurds [37], Sorani is spoken by around 7 million [38], and other dialects or sub-dialects (for example, Hawrami, Kalhori, Feyli, and others) are spoken by around 3 million Kurds [39].

Nevertheless, alongside promoting linguistic diversity and rights in general, "the [Iraqi] Kurdistan Regional Government's policy is to promote the two main dialects [of Kurdish] in the education system and the media" [40]. Consequently, the majority of satellite TVs in the Iraqi Kurdistan Region has at least news programs broadcasting in both dialects, either at the same session or as the separate sessions. In fact, some TV channels display news tickers (crawlers) in both dialects and sometimes in both Latin and Persian-Arabic scripts (e.g., Kurdistan TV, Rudaw, Kurdsat, NRT, and KNN). The websites of some of these TV channels are also provided in both dialects [41-43]. But, perhaps because the majority of Iraqi Kurds speak Sorani and this dialect has a long rich historical and literature background in the Iraqi Kurdistan Region, the de facto dialect of the conversations and the documents of the Iraqi Kurdistan Region is Sorani [44].

However, in spite of the emerging usage of the language, both regionally [45] and worldwide, Kurdish is not yet official in other countries where Kurds live. The reason behind this brief explanation is not to highlight political issues and motivations but to outline the fact that Kurdish might play a more significant role in the coming years. Without considering other parts of Kurdistan, being the official language of the Iraqi Kurdistan region only, suggests that Kurdish Computational Linguistics needs considerable attention.

Nevertheless, as the context of languages has changed tremendously because of the emergence and rapid spread of information technology, to become a well-known language in the world, Kurdish needs to be studied in light of the paradigm of CL and NLP. In fact, Kurdish needs to be understood not only by other people throughout the world but also among the Kurds themselves who speak different dialects that are not mutually intelligible.

Furthermore, Kurdish has a low visibility among the Internet users. Also currently there are no machine translator, no optical character recognizer, no commercialized text to speech, and no speech to text systems available for the language. In fact, crucial issues such as lack of grammatical and orthographical standards for the language, would affect any attempt towards the development of such utilities for the *intralanguage/interlanguage* purposes.

In summary, many obstacles stand in the way of advancing the preparation of Kurdish in order to be computationally processed. Moreover, working on Kurdish from a Computational Linguistics and Natural Language Processing perspectives would require some fundamental elements. For example, developing a corpus, as a core element that is required for many aspects of CL and NLP such as machine translation, dictionary preparation, text classification, discourse analysis, and text summarization, needs a substantial amount of time, budget, and effort. This becomes more challenging if one thinks about having a specific corpus for each special domain of the language study and processing. The challenge can grow if this corpus should be kept up-to-date and accessible to different users.

Equally important, for some reasons, which are beyond the scope of this article, written literature for the Kurdish language does not have a diverse and lengthy background. For some dialects/sub-dialects such as Kalhori and Hawrami, the case might be even more serious.

Moreover, Kurdish CL and NLP have not yet been established as academic disciplines. A quick survey on the available websites of universities, which are located in the Kurdish speaking areas in Iraq, Turkey, Iran, Syria, and Armenia, shows no fact that these subjects have been taken seriously except in one case, University of Kurdistan – Sanandaj, which one can find some valuable studies, though focusing mainly on one of the Kurdish dialects [46]. Indeed, current academic research on the Kurdish language in terms of CL and NLP is neither established nor seems to be promising as a scientific research area.

In this research we have tried to take one step towards an important issue with regard to Kurdish CL and NLP, which is automatic dialect identification. The following chapter explains the methodology of the research.

4. METHODOLOGY

Dialectology has been one of the research areas in traditional linguistics for almost as long as linguistics has been recognized as an independent field of science. However, the same is not the case in the Computational Linguistics context, at least for the dominant languages in the field such as English, German, and French. Therefore, when one is interested in computational dialectology, soon finds that the major works in this area have been carried out for some languages which, in computing sense, are not very popular.

To illustrate, Kessler has provided a method for computational dialectology in Irish Gaelic [47]. Similarly, Nerbonne and Heeringa have worked on Dutch [48]. They have computationally compared and classified 104 Dutch dialects. This type of dialectology assumes that the dialects under the investigation are mutually intelligible. In most of the cases, the focus is on the *phone* differences or slight changes that happen in the language morphology, from one dialect to the other.

In a different context, Tang and Heuven have performed a series of thorough experiments on some Chinese dialects in which they have provided some methods for these dialects classification [4]. In this latter case, intelligibility among the dialects is the main concern of the research.

Another research has been carried out in order to identify the Arabic dialects, which has resulted in suggesting an annotator that is used to annotate the Arabic texts according to their identified dialects [49].

Text classification is a well-studied area in Natural Language Processing, yet it still is a very demanding research subject [50–52]. Most of the text classification methods concentrate on the context classification. Different methods are used in text classification, most of which are based on Machine Learning techniques [53].

In the current research, we adapted a text classification method in order to classify Kurdish texts into the dialects that the texts are written in. We have targeted two main Kurdish dialects: Kurmanji and Sorani. The adapted method was applied in several steps, namely, data collection,

transliteration, and weighting list creation. Finally, the outcomes were tested in order to investigate the accuracy of the dialect identification. These steps will be explained in the following sections.

4.1. Transliteration

As it was mentioned, Kurmanji texts are, mainly, written in Latin script, while for Sorani texts the main script is Persian/Arabic. Persian/Arabic script has a longer history, while Latin script was suggested and introduced by Mir Celadet Bedir-Xan around the 1930s [27]. However, in both cases exceptions exist. That is, one can find texts in Kurmanji that have been written in Persian/Arabic script and texts in Sorani that have been written in Latin script. Again, as it was mentioned, currently no standard orthography exists for either dialect.

For this research, we collected the texts from different Kurdish media. In addition, it was decided to use the Latin script as the base for the dictionaries, the training set, and for the test data as well. But, because the Sorani texts were mainly written in Persian/Arabic script, the texts had to be transliterated into Latin script. In order to do so, we have developed a tool (a transliterator) in Python that transliterates the texts which are written in Persian/Arabic script into Latin script.
The main challenge of this transliteration process is the lack of a standard orthography in writing Kurdish. This case was discussed in section 3.4. Also there are ambiguous cases that an automatic transliterator is not able to produce what one might be able to produce by manual transliteration.

Our Python transliterator uses three compact Python dictionaries in order to cover the three different cases, which occur in the Kurdish writing using Persian/Arabic script. The first Python dictionary includes digits and single characters that can be transliterated into a single equivalent Latin character. For example, ‘ک’ and ‘ك’ both would be transliterated to 'k'. The second Python dictionary includes double characters, which have been concatenated using a special connector. It also handles the situations where the code and the shape of the concatenated characters are changed due to the participation in the concatenation. In this situation, in some cases, a double character must be transliterated to one character, and in some others, to two equivalent characters. For example, ‘ئا’ would be transliterated to 'a', while 'پ' would be transliterated to 'p'. The third Python dictionary is used for the situations where a character is concatenated to its predecessor or successor using two concatenation connectors on its both sides or it includes a postfix space such as 'ب' and 'په', which would be transliterated to 'b' and 'pe' respectively.

The transliterator was tested and tuned to cover all cases which are known to be special. The mentioned Python dictionaries were ordered and tuned manually. However, lack of standard orthography causes that one cannot expect that the result of the transliteration to be correct in all cases. Nevertheless, the result of the transliteration was tested manually in different situations in order to make sure that the transliterator produces the correct results when one compares the results against the original texts.

Figure 1 shows a sample in Kurmanj, which has been written using Persian/Arabic script.

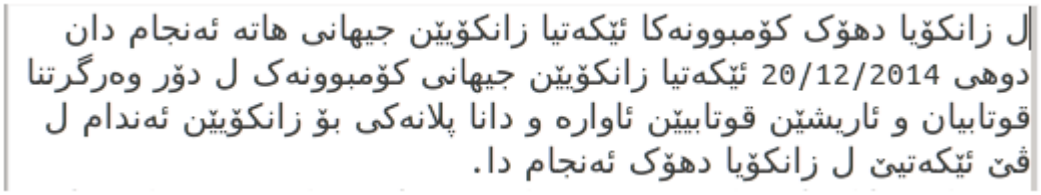


Figure 1. A sample text of Kurmanji in Persian-Arabic script

Figure 2 shows the result of the transliteration of the text of Figure 1 using the developed transliterator.

```
l zankoya dhok kombûneka êketya zankoîên cîhany hate
encam dan
dwhy 20/12/2014 êketya zankoîên cîhany kombûnek l dor
wergrtna qutabian u arîşên qutabiên aware u dana
planeky  bû zankoîên endam l vê êketîê l zankoya dhok
encam da.
```

Figure 2. The transliterated text of Figure 1

4.2. Weighting List Creation

In this research, we have used an adaptation of Support Vector Machines (SVM) [54], [55]. For the training and test phases, we collected data from different resources available on the Internet. For this purpose, we used the websites of several Kurdish media. The fundamental reason for this approach was because we decided to restrict our study to the most contemporary concepts that were widely understandable by the target dialects speakers.

At this step, which can be interpreted as the training phase, the classifier reads the texts and "sanitizes" the text to remove non-alphabet characters from the text, using regular expressions. It then extracts the vocabulary of the text and inserts them into a weighting matrix. We decided to include only the words with the length of at least two characters in this weighting matrix. Obviously, duplication is prevented.

The list keeps two weighting measures for each vocabulary. Each one of these two measures represent the closeness/distance of the word to one of the two dialects. At this stage, classifier assigns a value of 100 or 0 as the weighting measure (closeness/distance) to the vocabulary.

During this phase, the training phase, the classifier might find a word that is already in its weighting matrix. If, for example, this word has previously been assigned to Kurmanji, and now it has been found in a Sorani text as well, the weighting entry for Sorani would be set to 100 too. In other words, it means that the word is equally considered as Kurmanji and Sorani. At the end of this phase the required knowledge of the classifier has been generated.

Figure 3 shows a piece of the Weighting List file. In this list, the first column shows the row number. The second column shows the vocabulary. The third column shows the Kurmanji weight of the vocabulary. The fourth column shows the Sorani weight of the vocabulary.

In this sample there is no common words between the two dialects. However, there are commonalities between the vocabularies of these dialects. We have shown this briefly in section 5 (Experiment). The importance of this commonality and how it would affect the NLP in Kurdish is out of the scope of this research.

5302	rêka	100	0
5303	rêkany	100	0
5304	rêknekewtnî	0	100
5305	rêkupêkî	0	100
5306	rêkupêkîyek	0	100
5307	rêkxrawe	0	100
5308	rêkxrawî	0	100
5309	rêkxstneweî	0	100
5310	rêkên	100	0
5311	rêvebirina	100	0
5312	rêwcê	0	100
5313	rêyan	100	0

Figure 3. A sample of Weighting List

It is worth mentioning that the measures of 100 was used for the later developments. At this stage, this seems to be a binary function, returning *true* if a vocabulary belongs to a certain dialect, *false*, otherwise. However, it was not used this way. Instead, it was used as a measure which participated in the dialect classification cumulatively. In fact, we are interested in further research about this case and to assign different closeness/distance weightings that show the affinity of a word to a particular dialect more precisely. Therefore, the Weighting List could be updated in the future to accommodate different values between 0 and 100. This would require more data which must be manually labelled and used in the training of the classifier. Obviously, the other parts of the research environment would remain unchanged.

4.3. Classification Process

In the classification process, first, the classifier reads the input text and extracts what is, usually, called the "features" in the classification context. Again, during this process, the text processor removes all non-letter characters from the text. The feature extraction happens in two steps. First, the text processor tokenizes the text, counts the words, and updates the vocabulary vector by setting the number of occurrences of each word that it finds in the text. Second, it sets a two entry vector by calculating the weight of each entry.

This process has been formulated as below:

$$W^2_{i=1}[i] = \sum_{j=1}^{n} WL[i,j] \times VC(j), \qquad VC(j) > 0 \tag{1}$$

$$DC^2_{i=1}[i] = min\left(\left(W^2_{i=1}[i] \div 100\right), \qquad 100\right) \tag{2}$$

Given:

W is a vector with two entries corresponding to the two dialects. **WL** is the weighting matrix or feature matrix. **VC** is the number of occurrences of each word in the text that has a corresponding entry in **WL**. Each entry of the **DC** vector in (2) shows the percentage (probability of the text being of a specific dialect) that classifier assigns to the text.

The classifier was developed using Ocatve. Octvae was chosen because it was powerful in handling vectors, arrays, and matrices. It was also a proper open source replacement for this kind of experiment, which otherwise should be performed using MATLAB.

5. EXPERIMENT

The Weighting List creation process produced 6792 words. In the testing phase, several pieces of texts were tested for each dialect. Table 1 shows the result.

Table 1. Dialect classification result

Text Dialect	Best Guess		Worst Guess	
Kurmanji	Kurmanji	92%	Kurmanji	52%
	Sorani	26%	Sorani	26%
Sorani	Sorani	91%	Sorani	50%
	Kurmanji	24%	Kurmanji	24%

Table 2 shows the number of words attributed to each dialect alongside the common words among the dialects. It also shows the percentage of the common words to all words and the total words in each dialect.

Table 2. Dialect classification result

Count	Total	Kurmanji	Sorani
Words	6792	2632	4160
Common Words	208	208	208
Percentage	3%	7%	5%

5.1. Analysis and Discussion

The experiment showed that with a reasonable number of vocabulary of about 7,000 entry, the classifier is able to correctly classify the texts that are found in the media. It also showed that in all cases, the classifier assigns a significant magnitude to the dialect that was not the main dialect of the text. This case could have been considered a usual result of the commonality between the dialects with regard to their vocabulary.

To investigate this case, the common vocabularies between the dialects were counted. The common vocabularies were selected based on their weighting measure in the Weighting List. Obviously, the common vocabularies were those with the same weighting measure. Table 2 shows, the percentage of the common vocabularies. It shows that although based on the collected data there is an insignificant difference between the percentages of the presence of the common vocabularies in each dialect, the rate of commonality is far less than the one that is observable in the dialect identification percentage.

This fact is of interest from different point of views. For instance, it shows that Kurmanji and Sorani dialects are sharing common vocabularies that although do not form a large portion of their lexicon, play an important role as the basis for their lexicon structure. In other words, although Kurmanji and Sorani are considered as two dialects that are mutually unintelligible, their case might not be similar to the case of two different languages.

5.2 Issues

There are several issues that our research has not addressed at this stage. We continue our study to investigate these issues that we believe that they are crucial for the advancement of CL and NLP for Kurdish. These issues are listed below:

- What would be the results of the experiment, if we use a "stemmer" during the Weighting List generation and during the classification process?
- How the lack of a standard orthography affects the entire study?
- How the issue of the proper nouns (proper names) [56, 57] and Named Entity Recognition would affect the entire approach?

Regarding the last item, one may suggest that an immediate remedy could be to find those words that start with a capital letter. Unfortunately, this is not an option, because in Kurdish no uniform rule exists about capitalization of the proper nouns. Even if such a rule existed, it could not be applied to Persian/Arabic script.

6. CONCLUSIONS

The researchers of Computational Linguistics and Natural Language Processing for the dominant languages such as English and German, have not focused on the computational dialectology. However, this subject is important in languages with a diverse and linguistically long distant dialects. In some cases, these dialects might be considered mutually unintelligible. For the languages with this specification, automatic dialect classification/identification becomes a necessary part of Language Technology.

Kurdish language includes several dialects. The two widely spoken dialects are Kurmanji and Sorani. These two dialects are considered by professionals and linguists as mutually unintelligible. This research applied an adapted technique of classification based on an adaptation of Support Vector Machine (SVM) approach for dialect classification of Kurdish texts which are written in the mentioned dialects.

The research showed that with a proper vocabulary list that is used to train the system, the text's dialect could be identified with a high degree of accuracy. The experiments also showed that there is a small number of common lexicon that plays an important role in the forming of the context of each dialect.

However, the area of the research has several unexplored topics. For instance, how developing a "stemmer" that is able to find the *stems* of the text words would affect the classification result, both from the efficiency and the accuracy point of views. Esmaili, Salavati, and Datta [30] have

introduced a rule-based "stemmer". We have also developed a "stemmer" with some differences with the one that Esmaili, Salavati, and Datta have developed. However, this "stemmer" has not been incorporated at this stage of the research.

REFERENCES

[1] C. G. Clopper and D. B. Pisoni, (2007) "Free classification of regional dialects of American English," Journal of Phonetics, vol. 35, no. 3, pp. 421–438. [Online].
Available: http://www.sciencedirect.com/science/ article/pii/S0095447006000301.

[2] G. Tuaillon, (1986) "How the French dialectal data enter the Atlas Linguarum Europae," English, Computers and the Humanities, vol. 20, no. 4, pp. 247– 252. [Online].
Available: http://dx.doi.org/10.1007/BF02400111.

[3] S. Harrat, K. Meftouh, M. Abbas, S. Jamoussi, M. Saad, and K. Smaili, (2015). "Cross-dialectal arabic processing," in Computational Linguistics and Intelligent Text Processing, Springer, pp. 620–632.

[4] C. Tang and V. J. van Heuven, (2009). "Mutual intelligibility of Chinese dialects experimentally tested," Lingua, vol. 119, p. 24.

[5] P. G. Kreyenbroek and S. Sperl, (1992) The Kurds: a contemporary review. New York: Routledg

[6] Kurdish Academy of Languages, The Kurdish Population, (2008). [Online].
Available: http://www.kurdishacademy.org/?q=node/199 (visited on 10/05/2014).

[7] D. McDowall, (2005). A Modern History of Kurds. New York: I.B.Tauris.

[8] J. Benesty, M. M. Sondhi, and Y. Huang, (2008). Springer Handbook of Speech Processing. Secaucus, NJ, USA: Springer-Verlag New York, Inc.

[9] M. Gasser, (2006). How Language Works, Ed3.0. [Online].
Available: http: //www.indiana.edu/~hlw/book.html (visited on 12/26/2015).

[10] The dialects of Kurdish / home, (2015). [Online].
Available: http://kurdish.humanities.manchester.ac.uk/ (visited on 02/20/2015).

[11] J. Huggler, (2001). The world's largest nation without a state seeks a new home in the west, The Independent. [Online]. Available: http://www.independent.co.uk/news/world/europe/the-worlds-largest-nation-without-a-state-seeks-a-new-home-in-the-west-692440.html (visited on 02/20/2015).

[12] A. Burhan, (2011). "Kurds and Kurdish language," Turkish Studies, vol. 6, no. 03, pp. 43–57. [Online]. Available: http://www.turkishstudies.net/Makaleler/1793359710_4_ahmet_buran.pdf (visited on 02/27/2015).

[13] G. Haig and Y. Matras, (2002). "Kurdish linguistics: a brief overview,".

[14] E. R. McCarus, (1960). "Kurdish language studies," The Middle East Journal, pp. 325–335.

[15] F. Hennerbichler et al., (2012). "The origin of Kurds," Advances in Anthropology, vol. 2, no. 02, p. 64.

[16] A. Hassanpour, (1992). Nationalism and language in Kurdistan, 1918-1985. Edwin Mellen Pr.

[17] T. Jügel, (2014). “On the linguistic history of Kurdish,” Kurdish Studies, vol. 2, no. 2, pp. 123–142.

[18] G. Haig and E. Öpengin, (2014). “Introduction to special issue-Kurdish: a critical research overview,” Kurdish Studies, vol. 2, no. 2, pp. 99–122.

[19] D. N. MacKenzie, (1962). Kurdish Dialect: Studies. Oxford University Press, vol. 2.

[20] J. Nebez, (1976). Ziman-i Yekgirtû-i Kurdi (’Towards a Unified Kurdish Language)[in Kurdish]. Bamberg: NUKSE.

[21] M. R. Izady, The Kurds: A concise handbook. Taylor & Francis, 1992.

[22] Kurdish language | Kurdish academy of language. (2014). [Online]. Available: http://www.kurdishacademy.org/?q=node/41 (visited on 02/27/2015).

[23] L. Paul, (2014). KURDISH LANGUAGE, Encyclopaedia Iranica, online edition. [Online]. Available: http://www.iranicaonline.org/articles/kurdish-language-i (visited on 09/20/2014).

[24] M. Leezenberg, (2015). “Gorani influence on central Kurdish,” [Online]. Available: http://www.kurdishacademy.org/?q=node/10 (visited on 05/29/2015).

[25] KAL featured articles. (2014). [Online]. Available: http://www.kurdishacademy.org/?q=ku/book/export/html/5 (visited on 02/28/2015).

[26] C. A. Bedirxan and R. Lescot, (1986). Kurdische Grammatik: Kurmancî-Dialekt. Kurdisches Institut, vol. 1.

[27] Y. Matras and G. Reershemius, (1991). “Standardization beyond the state: the cases of Yiddish, Kurdish and Romani,” UIP-BERICHTE UIE REPORTS DOSSIERS IUE, p. 103.

[28] S. Krauwer, (2003). “The basic language resource kit (BLARK) as the first milestone for the language resources roadmap,” Proceedings of SPECOM 2003, pp. 8–15, 2003.

[29] H. Hassani and R. Kareem, “Kurdish Text to Speech (KTTS),” in Designing for Global Markets 10 Proceedings of the Tenth International Workshop on Internationalisation of Products and Systems IWIPS 2011, 2011, pp. 79–89.

[30] K. S. Esmaili, S. Salavati, and A. Datta, (2014). “Towards Kurdish information retrieval,” ACM Transactions on Asian Language Information Processing (TALIP), vol. 13, no. 2, p. 7.

[31] B. O. Mohammed, (2013). “Handwritten Kurdish character recognition using geometric discretization feature,” IJCSC, vol. 4, pp. 51–55.

[32] The Republic of Iraq - Ministry of Interior - General Directorate for Nationality, (2005). Constitution of Iraq. [Online]. Available: http://perleman.org/files/sitecontents/070708095356.pdf (visited on 02/19/2015).

[33] Kurdistan Parliament voted for draft constitution. (2014). [Online]. Available: http://www.perlemanikurdistan.com/Default.aspx?page=article&id=5593&l=1 (visited on 02/20/2015).

[34] Draft Constitution of Kurdistan Region: Kurdistan Parliament. (2014). [Online]. Available: http://www.perlemanikurdistan.com/files/sitecontents/100809083313.pdf (visited on 02/20/2015).

[35] M. J. Kelly, (2010). "The Kurdish regional constitution within the framework of the Iraqi federal constitution: a struggle for sovereignty, oil, ethnic identity, and the prospects for a reverse supremacy clause," Penn State Law Review, vol. 114, no. 3, pp. 707–808.

[36] K. R. S. Office, (2014). Iraqi Kurdistan Population Forecast for 2009-2020 [in Kurdish], Erbil.

[37] Kurdish, northern, (2015). Ethnologue. [Online].
Available: http://www.ethnologue.com/language/kmr (visited on 02/20/2015).

[38] Kurdish, central, (2015). Ethnologue. [Online]. Available: http://www.ethnologue.com/language/ckb (visited on 02/20/2015).

[39] Kurdish, southern, (2015). Ethnologue. [Online].
Available: http://www.ethnologue.com/language/sdh (visited on 02/20/2015).

[40] The Kurdish language. (2015). [Online]. Available: http://cabinet.gov.krd/p/p.aspx?l=12&p=215 (visited on 02/20/2015).

[41] Kurdistan TV. (2015). [Online]. Available: http://www.kurdistantv.tv/kurs/Home
(visited on 02/20/2015).

[42] Rudaw. (2015). [Online]. Available: http://http://rudaw.net/sorani (visited on 02/20/2015).

[43] KNN. (2015). [Online]. Available: http://knnc.net/ (visited on 02/20/2015).

[44] Kurdistan Parliament [in Kurdish]. (2015). [Online].
Available: http://www.perlemanikurdistan.com/Default.aspx?l=3 (visited on 02/20/2015).

[45] Turkey 'to allow Kurdish lessons', (2014). BBC News. [Online].
Available: http://www.bbc.com/news/world-europe-18410596 (visited on 02/20/2015).

[46] KLPP - main (EN). (2015). [Online].
Available: http://eng.uok.ac.ir/esmaili/research/klpp/en/main.htm (visited on 02/20/2015).

[47] B. Kessler, (1995). "Computational dialectology in Irish Gaelic," in Proceedings of the Seventh Conference on European Chapter of the Association for Computational Linguistics, ser. EACL '95, Dublin, Ireland: Morgan Kaufmann Publishers Inc., pp. 60–66. [Online].
Available: http://dx.doi.org/10.3115/976973.976983 (visited on 02/20/2015).

[48] J. Nerbonne and W. Heeringa, (2001). "Computational comparison and classification of dialects," Dialectologia et Geolinguistica, vol. 9, no. 2001, pp. 69–83.

[49] O. F. Zaidan and C. Callison-Burch, (2014). "Arabic dialect identification," Computational Linguistics, vol. 40, no. 1, pp. 171–202.

[50] K. Nigam, A. Mccallum, S. Thrun, and T. Mitchell, (2000). "Text classification from labeled and unlabeled documents using em," English, Machine Learning, vol. 39, no. 2-3, pp. 103–134, [Online].
Available:http://dx.doi.org/10.1023/A:1007692713085 (visited on 03/22/2015).

[51] J. Burstein, D. Marcu, S. Andreyev, and M. Chodorow, (2001). "Towards automatic classification of discourse elements in essays," in Proceedings of the 39th annual Meeting on Association for Computational Linguistics, Association for Computational Linguistics, pp. 98–105.

[52] J. Staš, J. Juhár, and D. Hládek, (2014). "Classification of heterogeneous text data for robust domain-specific language modeling," English, EURASIP Journal on Audio, Speech, and Music Processing, vol. 2014, no. 1, 14, 2014. [Online]. Available: http://dx.doi.org/10.1186/1687-4722-2014-14 (visited on 03/22/2015)

[53] A. Danesh, B. Moshiri, and O. Fatemi, (2007). "Improve text classification accuracy based on classifier fusion methods," in Information Fusion, 2007 10th International Conference on, IEEE, 2007, pp. 1–6.

[54] T. Joachims, (1998). Text categorization with support vector machines: Learning with many relevant features. Springer.

[55] S. Tong and D. Koller, (2002). "Support vector machine active learning with applications to text classification," The Journal of Machine Learning Research, vol. 2, pp. 45–66.

[56] Y. Ravin and N. Wacholder, (1997). Extracting names from natural-language text.

[57] G. Walther, B. Sagot, and K. Fort, (2010). "Fast Development of Basic NLP Tools: Towards a Lexicon and a POS Tagger for Kurmanji Kurdish," in International conference on lexis and grammar, [Online]. Available: https://hal.archives-ouvertes.fr/hal-00510999/document (visited on 02/27/2015).

AUTHORS

Hossein Hassani is a lecturer at the University of Kurdistan Hewlêr since 2007. He holds a BSc in Computer (Software), and an MSc in Information Management. He is also a PhD candidate in Computer Science.

Dzejla Medjedovic is an Assistant Professor and Vice Dean of Graduate Program at the Sarajevo School of Science and Technology. She has obtained her PhD in Computer Science from the Stony Brook University.

AUTHOR INDEX