REPORTS AND STUDIES IN
FORESTRY AND
NATURAL SCIENCES

ROMAN BEDNARIK (TOIM.)

Tietojenkäsittelytieteen päivät 2010:

The Computer Science Days 2010

PUBLICATIONS OF THE UNIVERSITY OF EASTERN FINLAND
Reports and Studies in Forestry and Natural Sciences

UNIVERSITY OF
EASTERN FINLAND

ROMAN BEDNARIK

Tietojenkäsittelytieteen päivät 2010

The Computer Science Days 2010

Publications of the University of Eastern Finland.
Reports and Studies in Forestry and Natural Sciences.
2

University of Eastern Finland
Joensuu
2010

Joensuu, 2010
Pertti Pasanen
University of Eastern Finland
ISBN: 978-952-61-0130-9 (PDF) ISSNL: 1798-5684 ISSN: 1798-5692

Editor's welcome

Dear computer scientists,

Welcome to Joensuu and Eastern Finland, to the annual meeting of the Finnish Society for Computer Science. This year the theme is: "Internationalization of Finnish Computer Science. Science, Technology, and Development: Infinite possibilities?"

I hope that the conference program will provide you with enough space to engage in discussions and interactions. The keynote talks and panel will certainly enthuse you, and the social program will allow for building new networks and extending the community.

Numerous people helped with preparing and organizing the conference, chairing sessions, attracting sponsors and other funding, proofreading, reassuring, and with other support. Without their generosity, this event would have not been possible.

All submissions were reviewed. Thanks go to the reviewers for providing feedback on the submissions:

Pasi Fränti

Jaakko Hollmén

Tomi Kinnunen

Ari Korhonen

Ville Leppänen

Tarja Systä

Enjoy your time in Joensuu.

Roman Bednarik

Joensuu, April 2010

Index

Financing and advancing international research: the role of Academy of Finland

Erki Oja

erkki.oja@tkk.fi

Aalto University, School of Science and Technology

Finland is close to the top of the OECD countries in R&D investments, measured as percentage of the GDP. The Academy of Finland is a major player, channeling over 16% of the government research expenditure. In 2009, this sum amounted to 309 million Euros. International activities are at the core of the Academy strategy. International cooperation is seen as crucial in internationally competitive basic research, high-quality infrastructures, and research environments. Three major aspects exist: global cooperation with the Academy's key partner countries, joint actions within the European Research Area (ERA), and Nordic cooperation.

Bottom-up cooperation, starting from the researcher level, is the basic form of international collaboration, but the Academy also helps to promote research using the funding opportunities specified in international agreements. The strategic priority countries are (in alphabetic order) Brazil and Chile, China, India, Japan, Russia, and USA and Canada. The cooperation takes various forms. Within the ERA, the main channels are the EU Framework Programs, especially the "ideas" program and ERC, as well as ESFRI and ESF. Joint Programming is a new for of cooperation. In the Nordic arena, the central operators are the NOS committees in various fields of science.

Unleashing the potential: ICTs for development

Tim Unwin

tim.unwin@rhul.ac.uk

Department of Geography, University of London

This address explores the innovative roles that computer scientists can play in helping people from poor countries transform their lives, especially in Africa. It considers some of the reasons why so many Information and Communication Technologies for Development (ICT4D) initiatives have failed, and suggests that this is often because they are top-down, supply-led and driven by the private sector or civil society, without sufficiently understanding the needs of the intended beneficiaries. While recognizing that many computer scientists are driven by the lure of lucrative contracts with international corporations, it concludes that some of the most innovative computer science research might actually be inspired by listening to the voices of the poor and marginalized.

Internationalization of Finnish Computer Science - progress or stagnation?

Esko Ukkonen

ukkonen@cs.helsinki.fi

Department of Computer Science, University of Helsinki

Internationalization is currently seen as one of the most important objectives of the development of Finnish universities. We should have more international students, researchers and teachers, and we should be able to provide top-quality research environments that attract the best talent from abroad to our departments. These goals are not any more new, and they have turned out difficult to achieve. We seem to be lagging behind of many other countries we want to compare ourselves with.

The talk reports some international activities as they have been implemented at the Department of Computer Science of the University of Helsinki. A summary of the state of Computer Science is also presented as it shows up in the recent Academy of Finland report 'The State and Quality of Scientific Research in Finland 2009'. According to the report, computer science in Finland is making decent but not outstanding progress.

Panel: *Internationalization of Finnish Computer Science*

Erkki Sutinen, Matti Tedre

erkki.sutinen@uef.fi, matti.tedre@uef.fi

University of Eastern Finland

Jorma Paukku, Ilari Lindy

Jorma.Paukku@formin.fi, Ilari.Lindy@formin.fi

Ministry for Foreign Affairs of Finland

Finnish universities are increasingly being assessed by their level of internationalization. The assessment criteria include the number of outgoing and incoming visits, journal publications, project coordination, awards, invited lectures, person-months, and staff hiring - all these have to have an international dimension now.

To the question "how to be international", one can distinguish four overlapping and complementary approaches. The first one, "home-based internationality", refers to collaborating with foreign researchers over the Internet. The second one, "conference internationality", refers to publishing, presenting, and networking in conferences. The third one, "expatriate internationality", refers to long exchanges or projects where some team members live outside their home country. The fourth one, "everyday internationality" refers to a situation where hosting and sending visitors is a normal routine at the unit, where the unit's staff is international, and where the working language in the unit is flexible.

None of the above-mentioned four approaches to internationality are in any inherent way better than the others, but they lead to different results. Some approaches suit some fields better than others. Joint theoretical research may not always necessitate time spent together face to face. An established research team may not need to build networks in conferences. And some research fields are international by their very nature. As a field, ICT4D (ICT for development) works on particular development issues in particular research contexts. ICT4D requires multiperspectival understanding of contextual dynamics, which takes a significant time to develop. In addition, validity of ICT4D research is established in situ, monitoring and evaluation entail long-term relationship between research communities, a broad systems view can only be developed through long involvement, and the normative goals often implied by ICT4D research can only be achieved through long commitment on site. ICT4D research teams are also typically international. Taking today's assessment criteria and ICT4D's nature into consideration, ICT4D should be the favorite field of today's academics.

Role of IT in the Future of Indonesia

H. E., Harry Purwanto

Embassy of the Republic of Indonesia

Abstract

Indonesia's geographical configuration consists of 17.508 islands, stretching along 5.300 km and a place of 235 million populations coming from more than 350 ethnic groups, is really a unique challenge for Indonesian Government not only in maintaining cohesion and unity of the nation, but also in providing prosperity to the people throughout Indonesia. Extending telecommunications to all 33 provinces and each of its main islands is a daunting task. The need to provide telecommunication for development and modernization has been recognized even since the first years of its independence and culminating with its launching domestic satellite "Palapa" in 1976. In the days ahead, in order to transform Indonesia towards an advanced and knowledge-based society, and closing gaps between big cities and distance places, Indonesia should optimally take advantage of ICT in enhancing education, health, general election system, transparency, bureaucracy's efficiency up to developing early warning system and setting up emergency preparedness.

Robot education for the children

Chih-Ming Wu

My Robot Institute, Taiwan

Abstract

Robotics technology is common scientific interest around the world of our kids. It is just like the art and music regardless of language. Robots not only can prompt children to study science and technology, but also can help them to think about the future. Because children always interested on robot. So it can be a good interface for kids make new friends form different country or language. Our project is starting in 2007. Scifest Just like a platform for the kids. We let the children learn about robots science. And help them to learn how to guide others to build a small robot to become a teach assistant of robot. And the little TA also make friends at the same time when their teaching. Then we can help children to organize an international community in robot hobby RUN (Robot United Nation).

Computer Scientists Professional Identity: ICT for Development

Mike Joy

M.S.Joy@warwick.ac.uk

University of Warwick, UK

Abstract

Professionalism in a relatively new discipline such as Computer Science is difficult to identify clearly. Not only are the technologies changing rapidly, but the relationship of the discipline to society is fluid. In this talk, we present a case study of one Western country's path to a professional framework for the IT Profession, and consider how the principal components of such a framework might apply for a developing country.

Ohjelmistometriikat arkkitehtuuritasolla

Sami Hyrynsalmi

sthyry@utu.fi

Software Engineering, Department of IT, University of Turku

Abstract

Vuosikymmenten saatossa on esitelty satoja erilaisia ohjelmistomittoja, jotka pyrkivät kvantisoimaan abstraktit ohjelmaosat yksinkertaisiksi luvuiksi tai symboleiksi. Pelkästään olio-ohjelmointiparadigmalle määriteltyjä mittoja on useita satoja. Mielenkiintoisesti korkeamman abstraktiotason moduuleille, kuten luokkajoukoille ja joukkojen muodostamille komponenteille, ei kuitenkaan ole esitelty kuin muutamia mittoja. Ohjelmistojen koon kasvaessa myös mielekkäiden käsiteltävien yksiköiden koko on kasvanut. Tämän vuoksi myös käytettävien ohjelmistomittojen on kasvettava vastaamaan kiihtyvän kehityksen tarpeita. Esitelmässä käsitellään olio-ohjelmointiparadigman mukaisia mittoja sekä erityisesti arkkitehtuuritasolle määriteltyjä ohjelmistomittoja, niiden oikeaksi todistamista ja mitattavia ominaisuuksia, joille voitaisiin kehittää korkeamman tason ohjelmistomittoja.

MOPSI Location-based Search Engine: Concept, Architecture and Prototype

Pasi Fränti, Juha Kuittinen, Andrei Tabarcea, Lomanzi Sakala

pasi.franti@uef.fi

Abstract

Traditional *location-based services* use databases where all entries have been explicitly georeferenced beforehand. We propose an alternative approach based on web search and using ad-hoc georeferencing of web-pages. We denote it as *location-based search engine* and emphasize its seemingly small but significant distinction from traditional location-based services. We outline how to construct such search engine and prove its effectiveness using a prototype called MOPSI search.

Elliptic Curve Cryptography Processor for High-Speed Applications — Extended Abstract —

Kimmo Järvinen*
Aalto University, School of Science and Technology
Department of Information and Computer Science

Abstract

Elliptic curve cryptosystems are public-key cryptosystems offering, e.g., shorter keys and faster performance compared to other alternatives, such as, RSA. Scalar multiplication on an elliptic curve is the main operation of every elliptic curve cryptosystem. This paper describes an elliptic curve processor designed specifically for field-programmable gate arrays. The processor takes use of special curves called Koblitz curves and its design combines results from several of the author's previous publications. To the best of the author's knowledge, the processor computes scalar multiplications faster than any other implementation (on any platform). This paper is an extended abstract of [Järvinen 2009b].

Keywords: Elliptic curve cryptography, Koblitz curves, field-programmable gate arrays

1 Introduction

The use of elliptic curves for public-key cryptography was proposed independently by N. Koblitz and V. Miller in 1985. Since then, *elliptic curve cryptosystems* have been studied intensively because they offer both shorter keys and higher performance than, e.g., RSA. Elliptic curves are nowadays used in numerous practical applications and their popularity is constantly increasing. Efficient realizations of elliptic curve cryptosystems require fast arithmetic on elliptic curves and, in particular, fast computation of an operation called scalar multiplication. For this reason, accelerating elliptic curve arithmetic with dedicated computing units has received a lot of attention. *Field-programmable gate arrays* (FPGA) have proven to be highly feasible alternatives because they combine fast performance and reprogrammability. FPGA-based designs can be tailored for specific algorithms or parameters because reprogrammability ensures that designs can be changed later, e.g., if the key size needs to be changed. In this respect, FPGA-based designs compare to software; performance-wise they are closer to dedicated hardware implementations. We present an FPGA-based elliptic curve cryptography processor optimized for a specific type of elliptic curves called Koblitz curves.

2 Preliminaries

Let $E(\mathbb{F}_{2^m})$ be the *additive Abelian group* formed by the points on an *elliptic curve* E defined over a *finite field* $\mathbb{F}_{2^m}$. *Point addition* is the operation $P_1 + P_2$ where P_1 and P_2 are two distinct points in $E(\mathbb{F}_{2^m})$. *Point doubling* is the operation $2P_1$. Both operations are computed with several operations in $\mathbb{F}_{2^m}$: additions $x + y$, multiplications xy, squarings x^2, and inversions x^{-1}. *Scalar multiplication*, $Q = kP$, where k is a (large) integer and $P \in E(\mathbb{F}_{2^m})$, is the main operation of elliptic curve cryptosystems. The security of these cryptosystems relies on the assumption that it is infeasible to solve the *elliptic curve discrete logarithm problem*, i.e., to find k if Q and P are known. Scalar multiplication is computed with successive point additions and point doublings analogously with exponentiation algorithms, such as, *square-and-multiply*: one computes point doubling (square) for all bits of k and point addition (multiplication) if a bit is one ($k_i = 1$). Hence, the average cost for an ℓ-bit k is ℓ point doublings and $\ell/2$ point additions.

The cost of point addition and point doubling can be reduced by representing points with different coordinates. In traditional *affine coordinates* ($P = (x, y)$), point addition and point doubling both require one inversion, which is significantly more expensive than addition, multiplication, or squaring in $\mathbb{F}_{2^m}$. The inversions can be exchanged for other operations in $\mathbb{F}_{2^m}$ by representing points in *projective coordinates* as $P = (X, Y, Z)$. This typically offers significant speed improvements; however, an inversion is still needed for mapping the result point, Q, to the affine coordinates in the end of scalar multiplication.

The number of point additions can be reduced with *precomputations*. One computes and stores $P_i = iP$ where $i \in \{1, 3, 5, \ldots, 2^{w-1} - 1\}$. Scalar multiplication is then performed by scanning several bits of k at once and by utilizing the precomputed points, P_i. The average number of point additions reduces to $\ell/(w + 1)$.

All point doublings can be replaced with cheap Frobenius maps on so-called *Koblitz curves* [Koblitz 1991]. They are included, e.g., in standards by NIST. The *Frobenius map* is $\phi(P) = (x^2, y^2)$ where $P = (x, y) \in E(\mathbb{F}_{2^m})$. Utilizing Frobenius maps requires that k is first converted into a so-called τ*-adic representation*. Both projective coordinates, e.g. [Al-Daoud et al. 2002], and precomputations, e.g. *width-w τ-adic non-adjacent form* (τNAF) [Solinas 2000], can be utilized on Koblitz curves.

To summarize, an efficient scalar multiplication on a Koblitz curve requires: (1) conversion of the integer k into a τ-adic representation, (2) precomputations with the point P, (3) scalar multiplication with a square-and-multiply(-like) algorithm, and (4) conversion from projective coordinates to affine coordinates.

3 Architecture

The processor is designed for a standardized Koblitz curve, *NIST K-163*, defined over $\mathbb{F}_{2^{163}}$. As shown in Fig. 1, the processor consists of four components: (1) τ-adic converter, (2) preprocessor, (3) main processor, and (4) postprocessor and they are, respectively, devoted for the tasks listed in the end of Sec. 2. The components form a three-stage pipeline and the processor is, therefore, capable of computing three scalar multiplications simultaneously.

The τ*-adic converter* is based on the algorithms and architecture presented in [Brumley and Järvinen 2010]. It converts k into width-4 τNAF and encodes it with (k_i, f_i)-tuples, where $k_i \neq 0$ gives the precomputed point used in a point addition and f_i is the number of Frobenius maps between two point additions.

The *preprocessor* computes the precomputed points, P_i, required in scalar multiplication with width-4 τNAF.

The *main processor* computes the scalar multiplication, i.e., successive point additions and Frobenius maps. A single point addi-

*e-mail: kimmo.jarvinen@tkk.fi

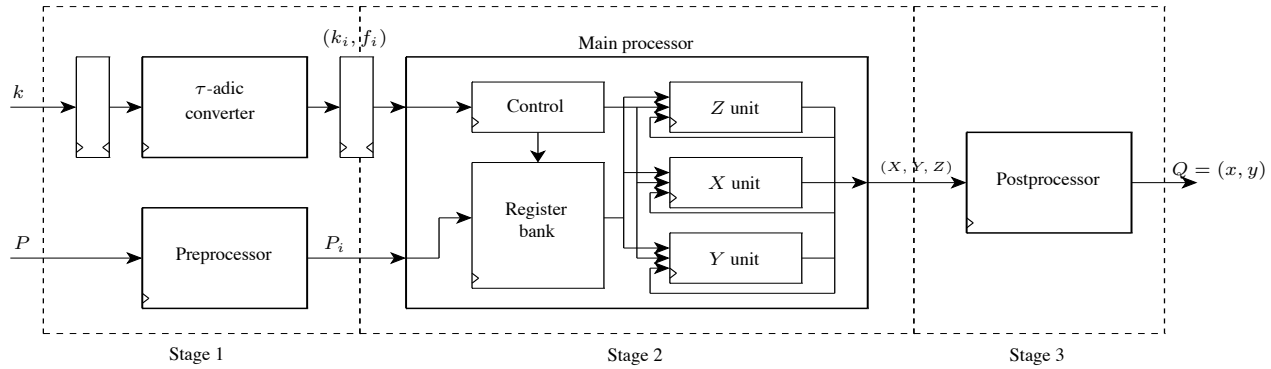


Figure 1: *Toplevel view of the processor*

Table 1: *Comparison to other processors available in the literature*

Reference	Device	Curve	Area	Memory	Time (μs)	Throughput
This work	Stratix II S180C3	K-163	14280 ALMs	25 M4Ks	11.71	235550
[Ahmadi et al. 2008]‡	Virtex-II 4000	K-233	15916 slices	Some BRAMs	7.22	138504
[Dimitrov et al. 2008]	Stratix II S180C3	K-163	28328 ALMs	52 M512s, 66 M4Ks	17.15	58309
[Järvinen and Skyttä 2008]	Stratix II S180C3	K-163	16930 ALMs	21 M4Ks	16.36	161290
[Järvinen and Skyttä 2009]†	Stratix II S180C3	K-163	26148 ALMs	—	4.91	203665
[Lutz and Hasan 2004]†	Virtex-E 2000	K-163	10017 LUTs	—	75	13333

† The values do not include area or time of the τ-adic conversion
‡ The values do not include area or time of the τ-adic conversion, precomputations, or coordinate conversion

tion has a critical path of four multiplications in $\mathbb{F}_{2^m}$ if at least three parallel multipliers are available. In the main processor, point additions are computed with *coordinate-specific units* as proposed in [Järvinen and Skyttä 2009] where two successive point additions are interleaved so that they have an effective critical path of only two multiplications in $\mathbb{F}_{2^{163}}$ per point addition. This interleaving is possible because the computation of the Z coordinate can be started already before the X and Y coordinates from the previous point addition are available. The Frobenius maps are computed with a *repeated squarer*, a component introduced in [Järvinen 2009a] that computes x^{2^e} in one clock cycle; hence, mapping a precomputed point with f_i Frobenius maps requires two clock cycles. The repeated squarer is attached to the register bank containing the precomputed points and it maps the points simultaneously while the coordinate-specific units compute point additions. Therefore, Frobenius maps are not on the critical path which consists of only $\ell/(w+1) \approx 163/5 = 32.6$ point additions, on average.

The *postprocessor* maps the result point, Q from projective coordinates to affine coordinates by computing an inversion, two multiplications and a squaring in $\mathbb{F}_{2^m}$: $(x, y) \leftarrow (X/Z, Y/Z^2)$.

4 Results

The processor presented above was described in VHDL and compiled for Altera Stratix II S180C3 FPGA with Altera Quartus II, ver. 8.1. The results are collected in Table 1 together with other processors from the literature that use Koblitz curves.

Table 1 shows that the processor achieves higher throughput (scalar multiplications per second) than other works. The faster timings reported for a single scalar multiplication in other works do not include τ-adic conversion [Ahmadi et al. 2008; Järvinen and Skyttä 2009], precomputations, and coordinate conversion [Ahmadi et al. 2008] or consume significantly more area [Järvinen and Skyttä 2009]. Scalar multiplication typically takes about a millisecond on a modern microprocessor. Hence, to the best of the author's knowledge, the processor computes scalar multiplications faster than any other implementation on any platform.

References

AHMADI, O., HANKERSON, D., AND RODRÍGUEZ-HENRÍQUEZ, F. 2008. Parallel formulations of scalar multiplication on Koblitz curves. *J. Univers. Comput. Sci. 14*, 3, 481–504.

AL-DAOUD, E., MAHMOD, R., RUSHDAN, M., AND KILICMAN, A. 2002. A new addition formula for elliptic curves over $GF(2^n)$. *IEEE Trans. Comput. 51*, 8, 972–975.

BRUMLEY, B. B., AND JÄRVINEN, K. U. 2010. Conversion algorithms and implementations for Koblitz curve cryptography. *IEEE Trans. Comput. 59*, 1, 81–92.

DIMITROV, V. S., JÄRVINEN, K. U., JACOBSON, M. J., CHAN, W. F., AND HUANG, Z. 2008. Provably sublinear point multiplication on Koblitz curves and its hardware implementation. *IEEE Trans. Comput. 57*, 11, 1469–1481.

JÄRVINEN, K. U., AND SKYTTÄ, J. O. 2008. High-speed elliptic curve cryptography accelerator for Koblitz curves. In *Field-progr. Custom Computing Machines, FCCM '08*, 109–118.

JÄRVINEN, K., AND SKYTTÄ, J. 2009. Fast point multiplication on Koblitz curves: Parallelization method and implementations. *Microproc. Microsyst. 33*, 2, 106–116.

JÄRVINEN, K. U. 2009. On repeated squarings in binary fields. In *Selected Areas in Cryptography, SAC '09*, LNCS 5867, 331–349.

JÄRVINEN, K. U. 2009. Optimized FPGA-based elliptic curve cryptography processor for high speed applications. *Integration—VLSI J.*. Submitted in July, 2009.

KOBLITZ, N. 1991. CM-curves with good cryptographic properties. In *Adv. in Cryptology, CRYPTO '91*, LNCS 576, 279–287.

LUTZ, J., AND HASAN, A. 2004. High performance FPGA based elliptic curve cryptographic co-processor. In *Information Technology: Coding and Computing, ITCC '04*, vol. 2, 486–492.

SOLINAS, J. A. 2000. Efficient arithmetic on Koblitz curves. *Des. Codes Cryptography 19*, 2–3, 195–249.

Suunnattujen etäisyyksien määrittäminen pyyhkäisyviivatekniikalla

Mika Murtojärvi, Ville Leppänen, Olli S. Nevalainen
Informaatioteknologian Laitos ja TUCS (Turku Centre for Computer Science), Turun Yliopisto

1 Johdanto

Etäisyyttä merellä olevasta tarkastelupisteestä lähimpään tietyssä suunnassa olevaan maa-alueeseen voidaan käyttää muodostuvan aallokon voimakkuuden arviointiin [2]. Lisäksi tarvitaan tieto tuulen voimakkuudesta. Näihin tekijöihin perustuvia yksinkertaisia aaltomalleja on käytetty melko hiljattain [5]. *Suunnattujen etäisyyksien* määrittämiseen on työkaluja [5, 6], mutta niiden suorituskyky ei ole aina riittävä. Tässä kuvataan yksinkertainen ja tehokas ongelman ratkaiseva algoritmi. Menetelmä kuvataan tarkemmin artikkelissa [7].

2 Ongelmanasettelu

Ongelma on määritelty 2-ulotteisessa tasossa. Annettuna on tarkastelupisteiden joukko P, janojen joukko L ja suuntien joukko Θ. Joukon L janat esittävät maa-alueiden reunaviivoja. Kullekin pisteelle $p \in P$ ja suunnalle $\theta \in \Theta$ on laskettava lyhin etäisyys $L_{p,\theta}$ pisteestä p maa-alueelle suunnassa θ pisteestä p katsoen. Mikäli piste p on maalla, $L_{p,\theta} = 0$. Reunaviivan pisteelle $L_{p,\theta} = 0$, mikäli suunnassa θ on maata. Muussa tapauksessa olkoon $d_{p,\theta,l}$ etäisyys pisteestä p janan l ja p-päätepisteisen θ-suuntaisen puolisuoran leikkauspisteeseen. Mikäli leikkauspistettä ei ole, $d_{p,\theta,l} = \infty$. Haluttu etäisyys on $L_{p,\theta} = \min_{l \in L} d_{p,\theta,l}$.

3 Algoritmit

Suunnattujen etäisyyksien määrittäminen muistuttaa ongelmaa, jota kutsutaan englanniksi nimellä *ray shooting problem* [1]. Tarkasteltava ongelma on kuitenkin helpompi, koska etäisyydet lasketaan kaikista pisteistä samoihin suuntiin. Tarkastelemme suuntaa $\theta = \pi$ ts. etäisyyksiä etsitään tarkastelupisteistä suoraan vasemmalle. Muille suunnille etäisyydet saadaan pyörittämällä koordinaatistoa.

Ongelma ratkaistaan *pyyhkäisyviivatekniikalla*, kts. [3]. Käsitteellisesti menetelmässä liikutetaan vaakasuoraa pyyhkäisyviivaa alhaalta ylös. Kullakin hetkellä tietorakenteessa S ovat tarkalleen ne joukon L janat, jotka pyyhkäisyviiva

leikkaa. Pyyhkäisyviivan kohdatessa tarkastelupisteen $p \in P$ lasketaan vaakasuuntainen etäisyys vasemmalle pisteestä p siihen janaan $l \in L$, joka on vaakasuunnassa lähimpänä p:tä. Lisäksi tutkitaan, onko p maalla eli jonkin polygonin sisällä. Tätä varten lasketaan, moniko tietorakenteen S janoista on pisteen p vasemmalla puolella. Piste p ei ole maalla, jos tämä lukumäärä on parillinen. Kun joukon S esittämiseen käytetään *järjestysstatistiikkapuuta*, em. lukumäärä saadaan tehokkaasti. Kyseessä on tasapainotettu järjestetty binääripuu (punamusta puu), jonka kukin alkio e sisältää tiedon siitä, montako alkiota on e-juurisessa alipuussa, kts. [4]. Tarkasteltaessa pistettä p etsitty lukumäärä on sama kuin löydetyn janan l järjestysluku joukossa S.

Algoritmi järjestää aluksi y-koordinaattien mukaan joukon Q, joka sisältää janojen kaikki päätepisteet sekä tarkastelupisteet $p \in P$. Joukko Q käydään läpi y-koordinaattien mukaisessa järjestyksessä. Kun kohdataan janan alempi päätepiste, jana lisätään joukkoon S, ylemmässä päätepisteessä jana poistetaan joukosta S. Kohdattaessa piste $p \in P$ lasketaan suunnattu etäisyys. Algoritmin aikakompleksisuus on $O((m+n)log(m+n))$, missä $m = |P|$ ja $n = |L|$

Työssä kehitettiin myös likimääräinen algoritmi, jonka kuvailu sivuutetaan. Menetelmän suorituskykyetu oli melko pieni tarkkaan menetelmään verrattuna, ja reunaviivan lähellä olevat pisteet ovat menetelmälle ongelmallisia.

Viitteet

[1] Agarwal, P.K., 1989, Ray shooting and other applications of spanning trees with low stabbing number. Proceedings of the fifth annual symposium on Computational geometry, s. 315–325.

[2] Anon., Shore Protection Manual, 1984, 4th edn, vol. 1 (U.S. Army Corps of Engineers, Coastal Engineering Research Center).

[3] Bentley, J.L., Ottmann, T.A., 1979, Algorithms for reporting and counting geometric intersections. IEEE Transactions on Computers, C-28, s. 643–647.

[4] Cormen, T.H., Leiserson, C.E., Rivest, R.L., Stein, C., 2009, Introduction to algorithms. The MIT press.

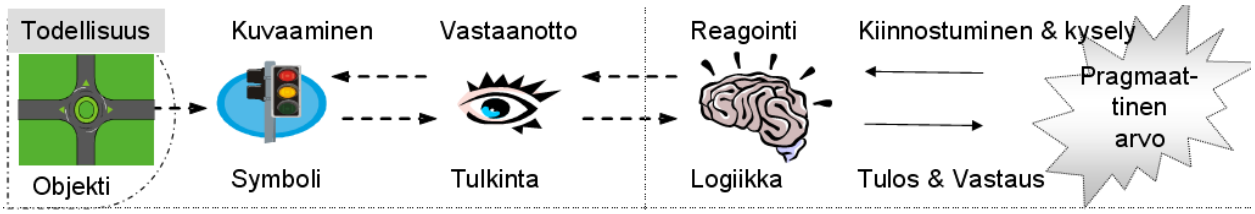
[5] Ekebom, J., Laihonen, P., Suominen, T., 2003, A GIS-based step-wise procedure for assessing physical exposure in fragmented archipelagos. Estuarine, Coastal and Shelf Science, 57, s. 887–898.

[6] Murtojärvi, M., Suominen, T., Tolvanen, H., Leppänen, V., Nevalainen, O.S., 2007, Quantifying distances from points to polygons — applications in determining fetch in coastal environments. Computers & Geosciences, 33, s. 843–852.

[7] Murtojärvi, M., Leppänen, V., Nevalainen O.S., 2009, Determining directional distances between points and shorelines using sweep line technique. International Journal of Geographical Information Science, 1362-3087, 23(3), s. 355–368.

Symbolinen analyysi: teoriasta käytäntöön

Erkki Laitila SwMaster Oy
erkki.laitila@swmaster.fi



Kuva 1. Symbolin tulkintaa liikenteessä. Sama ketju sopii ohjelmakoodiinkin virheen etsintään.

Abstrakti

Kaikki tiede perustuu symbolien käyttöön. Tietokoneohjelmat ovat myös symboleilla aikaan saatuja ratkaisuja. Näinollen on perusteltua tutkia miten tarkkoja ja laajamittaisia tulkintoja voidaan tehdä pelkästään symboleista lähtien - käyttämättä muita konstruktioita tai ulkopuolisia apuja. Tästä pelkistetystä lähtökohdasta on rakentunut symbolisen analyysin tutkimus sekä sen pohjalta ohjelmisto-ongelmien ratkaisemisen tekniikka, ***ORT****.*

Avainsanat*: program comprehension, source code analysis, core computer science*

1. Johdanto

Perinteinen tapa analysoida kohteita on ulkoinen: otetaan käyttöön kirjasto tai malli tai kieli, joka mallintaa kohdetta ulkoa päin – välillisesti. Tämä on sikäli hankalaa, että kaikki kohteet riippuvuuksineen joudutaan kuvaamaan epäsuorasti ja monivaiheisesti ja yksilöimään tilannekohtaisella tavallaan.

Hankaluuksia analyysien tekoon tulee usein työkaluissa valittaessa tapoja kuvata semantiikkaa. Niinpä on syntynyt monimutkaisia kirjastoja ja kuvauskieliä validointeineen kuten XML-johdannaiset ja Lispiin perustuva denotational-semantiikka. Klassinen staattinen analyysi ja dynaaminen analyysi ovat myöskin tällaisia ulkoa tarkastelevia menetelmiä. Niiden puutteena on tarkan tiedon saanti mallintamiskohteen sisäisistä asioista ja semantiikasta.

Symbolinen analyysi pyrkii vastaamaan näihin haasteisiin esittämällä sisäisen *virtuaalisen* mallinta-mistekniikan, missä jokaisella symbolilla on oma identiteetti ja semantiikka sekä jopa oma käyttäytymis-malli eli automaatti tilakoneineen. Työkalussa se on ohjelmoitu hybridioliona, joten se pystyy käyttämään oliokielten valmista olioviitesemantiikkaa [1]. Näin on saatu ajokelpoinen yleinen kognitiivinen virtuaalijärjestelmä, *oma pienoismaailmansa*.

Käytännön hyötyä symbolisesta analyysistä saadaan sen tarjoamien formalismien ja ajattelutapojen kautta. Sen avulla voidaan rakentaa työkaluja, jotka hakevat automaattisesti esiin kriittisiä piirteitä tai mallintavat alueita, jotka liittyvät käyttäjän ongelmakohtiin.

2. Symbolisen analyysin teknologiat

Tutkimusvaiheessa vuosina 2005-2008 valmistui teorioiden joukko tukemaan symbolista käsittelyä [1]. Se koostuu toisiinsa linkittyneistä teknologia-avaruuksista seuraavasti. Koodi luetaan koneen muistiin GrammarWare-osuudella, missä koodi abstrahoidaan samalla tehokkaaseen symbolisen kielen muotoon. Mallintamisosuudella (ModelWare) siitä kudotaan atomistinen malli, missä koodin riippuvuudet on purettu atomilauseiksi. Jokaiselle atomilauseelle on mallissa yksi hybridiolio, joka siten vastaa symbolisen analyysin käsitettä symboli. Sitä voidaan simuloida (SimulationWare), mikä jäljittelee ohjelman suorittamista. Simuloinnin tulos vastaa dynaamisen analyysin tulosta, vaikkakin se usein kertoo vain osan loppuympäristön informaatiosta. Niinpä ohjelmasta saatavan tietämyksen poimimiseksi (KnowledgeWare) tarvitaan seuraavaa teknologiaa. Ohjelman

ymmärtämisen tehostamisen tarkoituksessa symbolinen muoto tulee vastaan tarkkana ja tehokkaana ilmaisutapana, sillä atomilauseisiin sijoittuva tieto yhdistää kutsuvia symboleita loogisella tasolla kuten matematiikassa: A=f(g(Y)). Näin suoritusketju symbolista toiseen saadaan perusteltua ohjelman semantiikan käsitteillä tarvittavine parametreineen. Sen avulla voidaan kaivaa esiin virheitäkin – kehittäjän toimiessa ylituomarina mikä meni vikaan.

3. Teoriasta käytäntöön

Edellä kuvattu tekniikka tuottaa oikeansisältöisiä tulkintoja symboleista seuraavasti. Kuvassa (**Kuva 1**) esitetään tiedonkulku tavoiteltavasta pragmaattisesta tavoitteesta tulkinnan vaiheeseen ja takaisin kysymysten ja vastausten toimiessa siirtomediana. Nämä tarkastelut ovat vain osatotuuksia, joten "suurempaa totuutta" on etsittävä kuten alla kerrotaan.

3.1 Ohjelmisto-ongelmien ratkaisemisen tekniikka

Tutkimuksen soveltavana vaiheena vuosina 2008-2010 on kehitetty ORT-tekniikka, jossa on seuraavat osa-alueet: vikaluokittelu (12 tapaa), jolla ongelmat luokitellaan sekä menetelmäkehikko (14 eri tasoa), jolla ongelmat formuloidaan ennen ratkaisua eli syväanalyysiä todentamis- ja päättelyvaiheineen.

3.2 Ongelman formulointi

Formulointiin on viisi erillistä menetelmätasoa, joilla tilanteeseen päästään kiinni. Ne muodostavat sujuvan ketjun, mikä johtaa todentamaan ongelma tapahtuma-paikalla tai laboratoriossa tai muuten simuloiden.

3.3 Ongelman syväanalyysi

Todentamisessa, esimerkiksi debuggerilla, ongelma on voitu jo maadoittaa tiettyyn symboliin tai sekvenssiin. Tästä eteenpäin ORT tarjoaa joukon menetelmiä, joiden taustalla on jo vakiintunut tieto siitä, mitä koodista on tarpeenkin tietää [4]. Näiden lisäksi tiettyjä holistisia hakuja ja käsitteiden muodostamiseen liittyviä työvaiheita tarvitaan. Nämä kaikki ovat mukana ORT-paketissa.

3.4 Virtuaaliarkkitehtuuri ja megamalli

Mallien käsittelyn hankaluutena on tyypillisesti ollut se, että mallin eläessä on vaikea tunnistaa sen rajoja. Suuri syy UML:n kaavioiden vähäiselle käytölle on siinä, että sen kaaviot eivät liity saumattomasti yhteen ja että UML ei tue kooditason tarkastelua. Siten on hankala ymmärtää koodia ja malleja UML:n avulla.

MDE:stä poimittu käsite **megamalli** [5] poistaa näitä ongelmia. Sen ansiosta edellä kuvatut menetelmätasot voidaan abstrahoida niin, että samaan kaavaan sopii sekä ihmisen tarkastelu että työkalun toimintojen speksaus. Tämä määritelmän ansiosta kaikki menetelmät voidaan tehdä monellakin eri tavalla, joten kehittäjä ei joudu ongelmiin työkalun puuttuessa.

3 Yhteenveto

Symbolisen analyysin metodologia on onnistunut yhdistämään keskenään tietojenkäsittelytieteen ytimen alueet kuten kielioppitekniikka, mallintaminen, suorittamisen automaatit sekä tietämyksen keruu. Näin on syntynyt laajennus ja täsmennys Chomskyn hierarkiaan: jokainen symboli on oma automaattinsa, jonka itsenäiselle mallintamiselle löytyy nyt perusteet.

Kehitetty formalismi normalisoi ohjelmakoodista saatavan tietoaineiston, joten sen varaan on mahdollista rakentaa kehittyneitä alustoja tuleville työkaluille. ORT-tekniikan selkeä käsitteistö palvelee vianhakua ja tyypillistä muutosten suunnittelua siinäkin mielessä, että sen varaan kehittäjä voi rakentaa omat mentaaliset mallinsa – ja jatkaa itse miten parhaaksi näkee.

Kiitokset

Tutkimus on tehty Jyväskylän yliopiston COMAS-ohjelmassa. Jatkohanketta on tukenut SwMaster Oy.

Viitteet

[1] Laitila, E., *Symbolic Analysis as a Basis for a Program Comprehension Methodology*. Väitös. Jyväskylän yliopisto, 2008.

[2] Laitila, E., *Symbolic Analysis as a Basis for Program Comprehension*, VDM Publishing, Location, 2009.

[3] Laitila, E., *Opas ohjelmisto-ongelmien ratkaisemiseen*, SwMaster & Uusi IT, Jyväskylä, 2010.

[4] von Mayrhauser, A., Vans, A.M., Program *UnderstandingBehavior During Adaptation of Large Scale Software.* Int Conf. on Program Comprehension 1998.

[5] Favre, J-M. Nguyen, T.. Towards a Megamodel to Model Software Evolution through Transformations (2004). http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.58.8491

Eye-tracking Setups for Diverse Collaborative Environments

Sami Pietinen*
University of Eastern Finland
Joensuu Campus
Department of Computer Science and Statistics

Abstract

Previous research of visual attention has mostly considered situations in which a single person performs a specific task. To our knowledge, the current eye-tracking devices and software support only this research situation, although some analysis software do support the feature where multiple eye movement recordings can be visualized on to a common stimuli. In this paper, we 1) discuss the different environments that are used in common collaborative work situations and 2) show what kind of requirements they raise and 3) what kind of setups is needed to meet the requirements, and 4) where we plan to use the setups that we present here.

CR Categories: H.5.2 [Information interfaces and presentation (I.7)]: User Interfaces (D.2.2, H.1.2, I.3.6)—Input devices and strategies (e.g., mouse, touchscreen);

Keywords: eye tracking, collaborative work, pair programming

1 Introduction

Previous research of visual attention has mostly considered situations in which a single person performs a specific task. To our knowledge, the current eye-tracking devices and software support only this research situation. Closest to support for studying collaborative work is given by some analysis software by supporting the feature where multiple eye movement recordings can be visualized on to a common stimuli.

Setting up a research environment to acquire the visual attention data from two or more persons when all are making use of a single stimulus requires more effort than in the typical case of just one person. We report on the lessons learnt from setting up such an environment with the hope that such information can help the future eye-tracking research that studies the visual attention of two or more participants in a collaborative task.

2 Eye Movements and Eye-Tracking

One of the important skills in collaborative work and learning is the ability to attend to information of mutual interest. In fact, we are experts in coordination of our activities based on the information we get about the point of regard of our peers. Together with other social communication cues, such as facial expressions, gestures, or posture, we continuously monitor gaze direction of other collaborators, and we use these to contribute to our understanding of joint attention as we communicate and work with others. To better understand collaboration in e.g. programming, i.e what could be possible indicators of efficient collaboration; we applied eye-movement tracking to capture the visual attention patterns of two programmers. Most of the eye-tracking research rests on the eye-mind assumption [Just and Carpenter 1980], stipulating that there exists a link between visual attention and visual information, and assuming that attention is linked to foveal gaze direction. We however also acknowledge that it may not always be so [Duchowski 2003]. One can focus visually on a particular point and at the same time cognitively attend to something else. Also, we can use our peripheral system to direct our visual attention focus in the field of view, instead of employing foveal system.

[Horvitz et al. 2003] claim that attention, its signaling and recognition, are the central parts in successful communication and thus play a central role in collaboration. Applying eye tracking to study attention and to support communication in collaborative processes is not a new discipline, but yet a relatively recent field of applied eye-tracking research. Since Velichovsky's work [Velichkovsky 1995], many other researchers investigated the interplay between visual attention and collaboration. For space constrains we do not review the whole body of literature, but we concentrate on the recent advances. A recent line of research developed by [Cherubini et al. 2008] and [Sangin et al. 2008] focuses on collaborating dyads in distance situations. The researchers apply machine learning algorithms and verbal protocol analysis to detect stages of the collaborative (learning) process. These might include, for example, misunderstanding, negotiation, or uncertainty. Others have applied information highlighting to cue visual attention with a hope to improve problem solving processes or knowledge transfer. We thus suggest that similar methods could be applied also to study the collaborative processes in programming. We hypothesize that the knowledge of the links between visual attention and aspects of collaboration can help us 1) in understanding of the role of visual attention in this context, 2) in improving and developing better methodologies, and 3) in building gaze-aware integrated development environments, which can automatically adjust the information and improve the collaboration processes in programming.

3 Goals and objective of the work

The objectives of this work are to be able to 1) record eye movements in collaborative settings in which a shared monitor screen is used 2) analyze recorded eye movements efficiently and cost effectively 3) find candidates for collaborative eye-tracking metrics. The goal is to be able to reliably and efficiently track and analyze eye movements in collaborative settings in order to infer different aspects of collaborations, such as efficiency, so that we could use this information to create new theories for eye movements behavior in collaborative tasks and give tracked users feedback on their success. In short, the goal is to produce a framework for eye-tracking in collaborative settings.

4 Methods used to achieve the goals

We developed an eye-tracking setup for tracking (two) collaborators eye movements simultaneously and then used this setup for recording eye movements during a two-month long empirical software

*e-mail: firstname.lastname@uef.fi

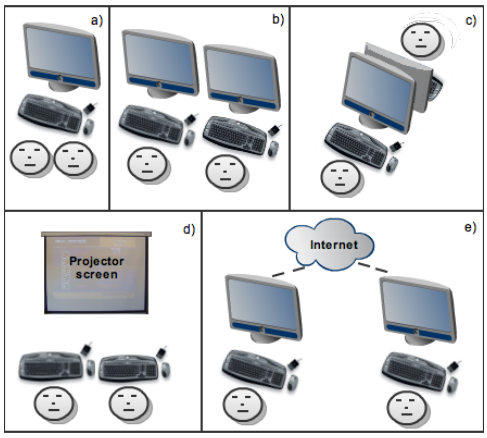


Figure 1: *Diverse collaborative environments. a) Single monitor setup, b) Dual monitor setup, c) Frontal setup, d) White-screen setup and e) distributed setup.*

project. We are constructing tools for eye-movement analysis and are currently able to superimpose two persons eye movements on to a recorded scene video. The setup can incorporate eye tracking of more than two persons eye movements.

5 Diverseness of collaborative environments

Figure 1. represents the usual setups that are used in order to make collaboration possible. We have mainly used the setup a), with an eye-tracking setup described in [Pietinen et al. 2008]. This setup consist of monitor-integrated tracker model Tobii 1750 or T120 and table-mounted tracker model ASL 504. With these devices, also setups b) and c) are possible, although ASl 504 provides very limited head movement therefore being useless in empirical long lasting recording sessions without chin rest. This is because in longer recording sessions participants tend to forget that they are eye-tracked and gradually start to move more than the setup allows. An integrated model would bring more accurate results still allowing quite much of freedom of head movement. Another option would be to replace ASL 504 with Tobii X120, but in this model there is need for environment mapping process between the stimuli source and the tracker, - i.e. angles and distances between the tracker and stimuli source and the size of stimuli source - which is prone to measurement errors. The setup d) can only be accommodated by using table mounted models.

One could also use head-mounted models in setup d), but they are more obtrusive, because one has to wear the device (as is not the case with table mounted remote tracker models) and do not without additional devices offer stable location of the stimuli. Head movements would make the stimuli closer to dynamic stimuli by moving the location of otherwise static stimuli. Head mounted eye-trackers would also impose the need of putting them on and getting them of from the head when larger movement is needed inside the office space and even more when there is need for going outside the office.

The setup e) is more demanding to construct since the stimuli that is represented to participants, needs to be delivered over the net to both participants and preferably with as low latency as possible. We can use shared desktop applications for this or simulate this by just using same monitor signal source from the computer by using a DVI divider for example. Same goes with the input devices. In near future, multi-pointer and multi-touch technologies will be in common use and this will also change the nature of collaboration. In conclusion, to accommodate all the setups, one could use table mounted trackers, but if integrated model would be available, they should be preferred. This evaluation of appropriateness of different eye-tracker models to different setup is limited to the ones we have opportunity to use. The eye-tracker models that we currently have in our lab are Tobii T120, X120, ASL 504 and ASL 501, and also an mobile version from ASL. A more throughout comparison of different eye-tracker models can be found from [Spakov 2008].

6 Analysis of Eye movements in Collaborative Work

What we can currently do is a descriptive analysis of the recorded eye movements, but we propose new metrics that could be used for inferring different aspects of collaboration, such as overlapping fixations and their average fixations lengths. The idea is to extend and validate the current eye tracking metrics used in eye tracking single persons eye movements to collaborative settings. A setting that contains two or more persons working on a shared task. We are also considering the use of superimposed eye-movements as an additional communication medium, which might be useful especially in distributed settings, as claimed by other researchers. Using different manufacturers' eye-trackers poses a problem of differences in the raw eye movement data. In high level of analysis with a need for low amount of frame-rate in data, this might not be such of a big problem, but in more detailed analysis, this can be one source of an error. This claim needs to be investigated quantitatively, but at this stage, it seems to be an issue revealing itself for example in the stableness of the raw eye movement data. This leads to an need to normalize the data before using common fixation algorithm for example or use a windows size that is adjusted based on tracker model.

7 Main outcome and its significance

The main outcome of this paper is to be able to track multiple persons eye movements simultaneously and choose a proper setup for recording the eye movements. The best current eye tracking setup for studying human collaboration varies depending on the purpose of use. Table mounted remote models seem to be the best eye-tracker models for diverse environments, but if ease of setup is of high importance and multiple monitors can be used, integrated models should be used.

References

CHERUBINI, M., NÜSSLI, M.-A., AND DILLENBOURG, P. 2008. Deixis and gaze in collaborative work at a distance (over a shared map): a computational model to detect misunderstandings. In *ETRA '08: Proceedings of the 2008 symposium on Eye tracking research & applications*, ACM, New York, NY, USA, 173–180.

DUCHOWSKI, A. T. 2003. *Eye Tracking Methodology: Theory & Practice*. Springer-Verlag, Inc, London, UK.

HORVITZ, E., KADIE, C., PAEK, T., AND HOVEL, D. 2003. Models of attention in computing and communication: from principles to applications. *Commun. ACM 46*, 3, 52–59.

JUST, M. A., AND CARPENTER, P. A. 1980. A theory of reading: From eye fixations to comprehension. *Psychological Review 87*, 4, 329–354.

PIETINEN, S., GLOTOVA, T., TENHUNEN, V., AND TUKIAINEN, M. 2008. A method to study visual attention aspects of collaboration: Eye-tracking pair programmers simultaneously. In *Proceedings of ETRA 2008*, ACM Press, New York, ACM, 39–42.

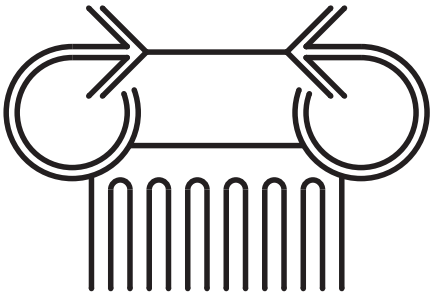
SANGIN, M., MOLINARI, G., NÜSSLI, M.-A., AND DILLENBOURG, P. 2008. How learners use awareness cues about their peer's knowledge?: insights from synchronized eye-tracking data. In *ICLS'08: Proceedings of the 8th international conference on International conference for the learning sciences*, International Society of the Learning Sciences, 287–294.

SPAKOV, O. 2008. *iComponent - Device-Independent Platform for Analyzing Eye Movement Data and Developing Eye-Based Applications*. PhD thesis, University of Tampere, Interactive technology.

VELICHKOVSKY, B. M. 1995. Communicating attention: Gaze position transfer in cooperative problem solving. *Pragmatics and Cognition 3*, 2, 199–222.

Roman Bednarik

Tietojenkäsittelytieteen päivät 2010: The Computer Science Days 2010



This report contains the papers and abstracts presented at the annual conference of The Finnish Society for Computer Science, held in Joensuu, Finland, between April 15th and 16th, 2010. All papers were reviewed and the authors presented their posters during the conference.

UNIVERSITY OF
EASTERN FINLAND

Publications of the University of Eastern Finland
Reports and Books in Forestry and Natural Sciences

ISBN: 978-952-61-0130-9 (PDF)
ISSNL: 1798-5684
ISSN: 1798-5692