PROCEEDINGS

55. IWK

Internationales Wissenschaftliches Kolloquium
International Scientific Colloquium

13 - 17 September 2010

Crossing Borders within the ABC

Automation,

Biomedical Engineering and

Computer Science

Faculty of
Computer Science and Automation

www.tu-ilmenau.de

TECHNISCHE UNIVERSITÄT
ILMENAU

Impressum
Published by

Publisher: Rector of the Ilmenau University of Technology
Univ.-Prof. Dr. rer. nat. habil. Dr. h. c. Prof. h. c. Peter Scharff

Editor: Marketing Department (Phone: +49 3677 69-2520)
Andrea Schneider (conferences@tu-ilmenau.de)

Faculty of Computer Science and Automation
(Phone: +49 3677 69-2860)
Univ.-Prof. Dr.-Ing. habil. Jens Haueisen

Editorial Deadline: 20. August 2010

Implementation: Ilmenau University of Technology
Felix Böckelmann
Philipp Schmidt

USB-Flash-Version.

Publishing House: Verlag ISLE, Betriebsstätte des ISLE e.V.
Werner-von-Siemens-Str. 16
98693 Ilmenau

Production: CDA Datenträger Albrechts GmbH, 98529 Suhl/Albrechts

Order trough: Marketing Department (+49 3677 69-2520)
Andrea Schneider (conferences@tu-ilmenau.de)

ISBN: 978-3-938843-53-6 (USB-Flash Version)

Online-Version:

Publisher: Universitätsbibliothek Ilmenau
ilmedia
Postfach 10 05 65
98684 Ilmenau

Home / Index:
http://www.db-thueringen.de/servlets/DocumentServlet?id=16739

AN INTERNET PROTOCOL FOR THE BRAIN: TOWARD A UNIFIED MESSAGE FORMAT AS AN INFORMATION REPRESENTATION IN COGNITIVE ROBOTICS

Gerd-Jürgen Giefing

Department of Electrical Engineering and Information Technology
Laboratory of Information Technology
Technische Fachhochschule Georg Agricola, 44787 Bochum, Germany

ABSTRACT

In cognitive robotics, embodied modules like sensors, actors and processing units are combined in a very flexible and dynamic way to achieve specific tasks. Such modules still are not standardized.

We discuss an intra-module information representation supporting the global workspace and the communication between modules by considering findings in neuroscience, robotics, computer networks communication and results in automation. We then propose a routable unified message format as a matrix data object based information representation for an autonomous robot.

A demonstrator is presented proving the applicability of the proposed message format. Modules suitable for a humanoid robot head are task-driven integrated. A basic set of tasks has been implemented. The demonstrator incorporates augmented reality techniques with a focus on 3D stereo computer vision and graphics.

Index Terms – cognitive robotics, information representation, unified message format, augmented reality, 3D stereo vision

1. INTRODUCTION

Researchers in cognitive robotics have to spend much time on integrating sensors, actors, processes and data. To reduce complexity, guarantee upward scalability, control communication bandwidth and to support core work it is useful to establish standards.

Standards are accepted easier if they are simple and not complicating developments. They do not need to be technically optimal for a certain solution but a good compromise, e.g. Internet Protocol (RFC 791).

Test environments and test data sets to compare results and mathematics or image processing libraries for instance are commonly shared.

An information representation mostly independent of audio, vision, etc. for processing in modules and for the communication between modules still lacks standardization in cognitive robotics. Such a unified information representation is in focus of this paper.

2. RELATED WORK

Some of the referenced robot frameworks, toolkits and middleware approaches claim to set the standard in cognitive robotics because they have integrated work done before and are available as open source.

A three tier architecture and an abstract communication by generalized functions or adapters is typical for all these approaches. Many concepts rely on a central controller or mediator which performs a transformation of messages and coordinate systems (device centric, robot centric, world centric), e.g. MARIE [1] and CoSy Architecture Schema Toolkit [2] and CARMEN [3].

Miro [4] is biologically motivated and optimized for wireless real-time communication and monitoring multi robots. Components are hot-pluggable.

DMRAI [5] focuses on distributed system services in a platform independent Service Based Architecture. A CORBA like trader supports a synchronous real-time transfer of objects. Various messages are defined for super types, service types, interface types, property types.

The Player 2.0 [6] framework offers a queue-based message passing configured by a Task Description Language. The hierarchical addressing concept includes host, robot, interface and node. The message space name uses type and subtype. Data marshalling is done via XDR of the underlying RMI/CORBA layer.

ROS [7] can define a number of processes in a peer-to-peer topology supported by a name service. ROS provides over 400 message types which can be nested arbitrarily deep.

In both Player 2.0 and ROS, nodes send messages for a certain topic via an asynchronous publish-subscribe mechanism or via a server-like synchronous request/response scheme.

AuRA [8] controls low-level components by the superposition of motor and perceptual vector field maps (schemas). A complex robot behaviour is a sequence of schema interactions.

In this paper we follow a structural approach which does not make use of learning semantics, e.g. robot language games [9].

3. INFORMATION REPRESENATION IN RELATED DISCIPLINES

In this section we discuss relevant findings for cognitive robotics systems in related disciplines.

3.1. Information Representation and Processing in Neuroscience

Cognitive and computational neuroscience focuses on spatiotemporal representation and processing derived from synapses and electrical spikes [10]. Structural principles are considered to be the same everywhere in brain.

3.1.1. Networked Cortical Modules

Lesion experiments and neuroimaging yield a brain atlas, showing that processing in the brain is organized in cortical areas dedicated to specific functions.

Cortical pathways are dense parallel connections of projection neurons forwarding information to other cortical areas (modules). An afferent counterpart mostly exists for every efferent path.

Modules which are synchronized and interact in the “current state of consciousness” are regarded to be part of the global workspace and have full access to its information pool [11].

3.1.2. Layered Space-Variant Information Processing

Neurons are arranged in layers in most areas of the brain. Within a module, interneurons can form a cortical hypercolumn containing a full set of information values for a given receptive field.

Temporal synchronization to oscillations of about 25 ms binds assemblies of neurons by complex electrochemical dynamics to stimuli [12]. Nonlinear stability mechanisms are necessary to solve problems like binocular rivalry and change blindness.

3.1.3. Place Cells

Maps for spatial information do have neural correlates in place cells located in the hippocampus and grid cells located in the entorhinal cortex [13]. Their mental representation is scene centered along landmarks in cognitive comprehensive-maps. These maps can be described by embedded view graphs well suitable for problems like homing [14][15].

3.2. Classical Robotics

Device, robot and world coordinates are main issues in classical robotics. Coordinate transformation in homogeneous coordinates for translation, rotation or perspective projection is calculated by matrix multiplication. Alternatively, we can use quaternions.

To accelerate processing, information is provided at different scales, e.g. image pyramids [16] and MIP maps. Processing steps may be documented in the history field of the information representation’s header section, e.g. HIPS [17].

3.3. Communication in Computer Networks

The ISO model of data communication (ISO/IEC 7498-1:1994) defines seven layers and three planes: user, control and management. Protocols and protocol data units can be created and dynamically deployed within the ISO framework [18].

3.3.1. Communication Schemes

These communication schemes are serviceable in cognitive robotics:

- Connection-oriented synchronous client-server communication with a pull mechanism for several clients
- Peer-to-peer communication adequate for data streaming from module to module
- Publisher-subscribe communication pushes data to subscribers bound to the respective service

The last two communication schemes can be operated in synchronous/blocking or asynchronous/ non-blocking I/O and are common to transaction oriented middleware infrastructures, e.g. MQSeries or Java Messaging Service. Features like monitoring, scalability, fault tolerance or hot-pluggable may be advantageous.

3.3.2. Real-time Communication

To establish a distributed real-time communication with determined latency for a closed loop control, all protocol layers have to provide real-time control natively, e.g. EtherCAT (IEC 61784-2) [19]. Near real-time can be achieved by absolute timestamps in support protocols, e.g. RTP and RTCP (RFC 1889).

3.3.3. Interface Definition

CORBA and COM among others marshal objects written in different programming languages and distribute them across different computer platforms. CORBA uses an interface definition language (IDL) to define the object data format.

By means of ASN.1 (ITU-T X.680ff) and encoding rules (ITU-T X.690ff), systems can exchange messages platform independently.

Additional options are XML-RPC which utilizes XML data or HTTP client-server calls to forward information.

3.4. Automation and Manufacturing

In manufacturing, control systems traditionally follow a hierarchical design. The direction emerging in industrial robotics shows communication and hardware abstraction (virtual manufacturing device), an open communication and transparent process visualization. Typically, not all direct sensors and actors combinations are supported.

The Manufacturing Automation Protocol (MAP, IEC 61850-7-4) defines a flexible standard object data format. This object structure for encoding is sent along with the message. About 80 unified services are provided to read, write and manage attributes and supervise connections. MAP is able to route messages across different networks and uses a deeply nested addressing scheme for devices and components.

MAP precedes the Manufacturing Message Specification (MMS, ISO 9506). MMS is the basis of the Utility Communication Architecture (IEEE TR 1550), Inter Control Center Communication (IEC 60870) and Communication Networks and Systems in Substations (IEC 61850).

4. UNIFIED MESSAGE FORMAT

Derived from the previous findings, we think in images and propose a routable unified message format as a matrix data object based information representation. Therefore we name the respective communication protocol Matrix Transfer Protocol (MTP).

4.1. Design Aspects

The following design aspects form the basis of the proposed unified message format.

4.1.1. Communication

For simplicity and easy testing, we assume a sparsely connected coarse-grain architecture. Unlike HTTP (RFC 1945), the communication scheme we propose for MTP is a peer-to-peer one-way communication between processing modules via data streaming. A directed graph or flow path describes the links between the modules. A multiple fan-in and fan-out is also applicable.

Hence, an I/O module must be capable to act as gateway and route or modify MTP messages, respectively. The addressing scheme includes unicast (single device), multicast (group of devices) and special addresses (broadcast, no address).

4.1.2. Distributed System Modes

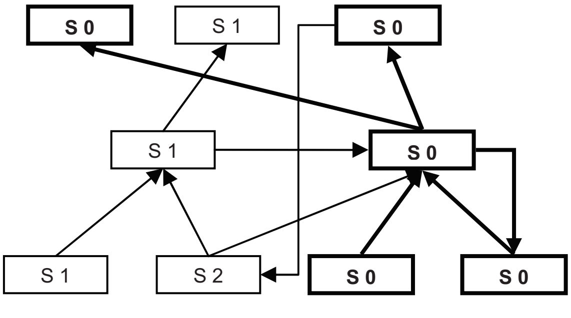

Figure 1. Connected modules running in different system modes (S0-S2). Modules and paths of the global workspace (system mode S0) are in bold.

Each module runs in a certain system mode (state) at any time. A module can choose its appropriate system mode from its inputs. Different system modes can run in parallel throughout the distributed system.

The active task follows from the external visible main behavior of the robot system to solve a specific problem. The active task is determined by one of the running system modes. For each task a respective system mode is defined. Modules compete for the control of the main behavior.

Modules contributing to the active task of the robot system are interacting in the global workspace and are regarded to be in the focus of attention (see Figure 1).

4.1.3. Unified Matrix Data Object

The core of the proposed message format is a matrix data object containing structured elements ("hypercolumns") in a one dimensional array interpreted as a two dimensional field (see Figure 2).

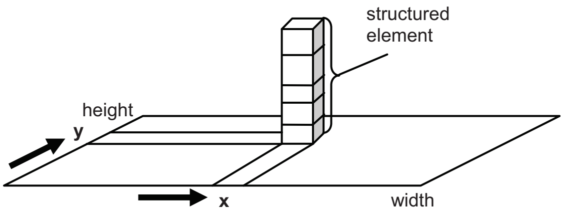

Figure 2. Unified matrix data object

This data object is able to represent the major data types relevant to cognitive robotics and geometric computing: position values, images, sound, maps, transformations/mappings and graphs.

4.1.4. Coordinate Systems

System-wide known coordinate systems have to take three kinds of dimensions into account: spatial, temporal and "ordinal".

- Sensor centered, actor centered (e.g., camera PTU, robot) and world centered spatial coordinate systems
- Coordinate systems with temporal dimensions convenient for e.g. text-to-speech, instead of message sequences
- "Ordinal" dimensions for ordered sets of information (e.g. dimension of a graph)

The coordinate system also determines scaling, range and structure of the matrix elements.

4.2. Protocol Data Unit Format

The header of the proposed unified message format for MTP combines patterns of PDU fields known from IP (version, time to live), MMS (address, object type) and RTP (absolute timestamp). Security and safety aspects are neglected.

The data encoding utilizes human readable ASCII text to a large extent to support simple debugging and to achieve restricted platform independence. Standard techniques may be deployed for data marshalling of binary encoded values.

Table 1. Proposed PDU format

field name	data type	encoding
version	int	ASCII string
src_address	int	ASCII string
dst_address	int	ASCII string
seq_number	int	ASCII string
time_to_live	int	ASCII string
timestamp	int	ASCII string
info	int	ASCII string
system_mode	int	ASCII string
coord_system	int	ASCII string
width	int	ASCII string
height	int	ASCII string
elements	ELEMENT[]	ASCII string /binary

The fields of the proposed PDU format are denoted in Table 1. Most fields are self-explanatory. The proposed PDU format is not intended to represent a pyramidal image structure, e.g. MIP images.

Table 2. ELEMENT structure of matrix data objects

matrix type	dimension	element structure
single position	space	x_x, x_y, x_z (,v_x,..(,a_x,..))
single rotation	space	φ, θ, Ψ
RGB image	space	r, g, b
intensity image	space	i
vector field	space	v_x, v_y, v_z
string	time	text
transformation	order	a
graph (dense)	order	adjacency weights
graph (sparse)	order	adjacency list

Table 2 lists the options of the field *elements* containing the matrix data object's values and shows that only a few matrix types need to be defined. A 1-by-1 matrix data object represents a single position value. An orthogonal set of matrix types, element structures, dimensions, scales, etc. is not planned.

5. A ROBOT HEAD SYSTEM USING AUGMENTED REALITY TECHNIQUES

The proposed unified message format is implemented in a robot head system. The robot head is utilized and controlled by a human supervisor. It is well suited for both a smart interactive man machine interface and a robot which extends the supervisor's sensing capabilities in a transparent way.

5.1. Hardware Components

The robot head side consists of

- **Eyes:** active stereo camera system
- **Ears:** microphone audio input
- **Mouth:** loudspeaker audio output

Among many camera system options tested, most in use are two HD Gigabit Ethernet cameras on a fast pan-tilt unit. The base distance of the stereo cameras are twice of the human interpupillary distance to get an enhanced stereo depth sensation.

The robot system provides the following interfaces to the human supervisor

- robot head (see above)
- keyboard, mouse, joystick controller
- head pointing sensor (head tracking device)
- graphics output displays

Figure 3. Head tracking device on HMD

The graphics output can be presented to the human supervisor by stereo monitors, stereo beamers or a SVGA head mounted stereo display (see Figure 3).

5.2. System Design

Based on hardware components mentioned, a demonstrator is realized (see Figure 4). The message format is used with an appropriate flat addressing scheme and adequate coordinate systems.

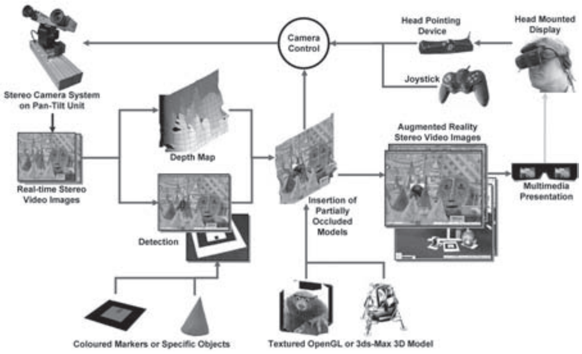

Figure 4. System overview

3D stereo augmented reality techniques combine real world data and virtual data in real-time to support system introspection und debugging.

The system is organized in a layered architecture:

1. library packages for vision, graphics, TCP/IP communication, message-passing, etc.
2. hardware drivers for low-level device control
3. a) modules for hardware abstraction
   b) modules for processing and controlling

Each module is implemented as a network process and capable to seamlessly communicate peer-to-peer to adjacent modules. Modules can be distributed across different hosts. Basically, they can hold multiple inputs and outputs. Modules route via the central controlling module at present to ensure system stability.

5.3. Tasks

Although the proposed data format is supposed to support arbitrary modules in cognitive robotics, we have chosen fairly simple tasks to validate the concept. Five tasks have been implemented so far:

- **System calibration:** adjusting camera gaze and identifying left and right devices
- **Manual camera pointing:** joystick controller determines camera gaze
- **Telepresence:** supervisor's head orientation controls camera gaze
- **Tracking:** color blob or colored marker visible in the scene controls camera gaze
- **Reading:** written block letters are optically recognized and pronounced by a synthetic voice

In most tasks, a depth map is calculated. Depth coded (2.5D) stereo video images are provided to the human supervisor for augmented depth perception. These tasks could be combined e.g. on a mobile platform for an autonomous intelligent cameraman [20].

5.4. Test and Debugging Facilities

Data streams from or to modules can simply be piped to files or from files. UNIX tools, e.g. netcat can be used as feeding test drivers and to save data as well. Single modules and arbitrary combinations of modules can be tested. Modules which evaluate timestamps are able to emulate the original timing.

A LabVIEW interface to the unified message format is key to an extensive online monitoring and visualization of dynamic data.

6. RESULTS

The robot system is realized with a partially implemented unified message format on a Linux PC platform in C, C++ and Java. The platform works near real-time without any real-time patches applied. Routing is configured manually.

6.1. Performance and Stability

The system creates and processes a maximum of about 1,200 output messages per second. Each message size can differ strongly. Messages containing color images are about 15,000 times the size of position value messages. Adaptations and token bucket mechanisms suppress output messages to save bandwidth and processing performance.

Modules operate at different speeds and resolutions. Only one feedback loop is active at a certain time. Stability is ensured by choosing different time constants in sensing and controlling actors.

6.2. Vision and Graphics

The vision and graphics module of the robot system currently processes 10 SVGA stereo images per second.

Figure 5. 3D stereo object insertion

Figure 5 shows an example to support the human supervisor by augmented reality techniques. Two 3D objects are inserted in correct pose and position according to identified colored markers.

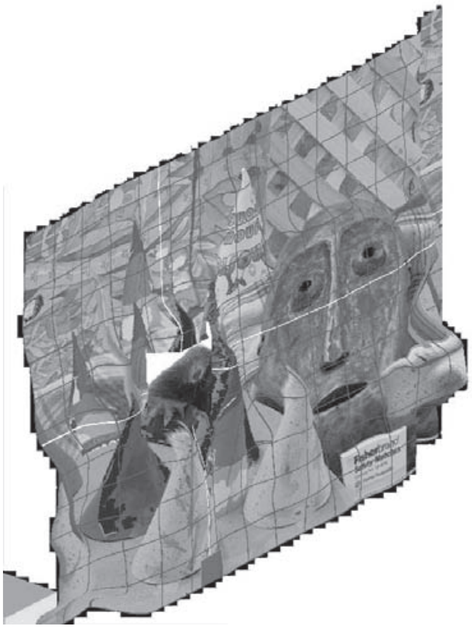

Figure 6. Inserted 3D object in 2.5D image

A partially occluded 3D object insertion into a 2.5D image is demonstrated in Figure 6.

7. CONCLUSION AND FUTURE WORK

We have proposed a unified message format for a Matrix Transfer Protocol and demonstrated that this message format is suitable for distributed modules in cognitive robotics to accomplish specific tasks.

Some proposed message format options and system configurations still have to be validated. Additional challenge is to verify this approach by integrating more devices and more modules e.g., face recognition, gesture recognition and grasping.

8. REFERENCES

[1] C. Côté, Y. Brosseau, D. Létourneau, C. Raïevsky, F. Michaud, “Robotic Software Integration Using MARIE”, *International Journal of Advanced Robotic Systems - Special Issue on Software Development and Integration in Robotics*, vol. 3, no. 1, INTECH, pp. 55-60, 2008

[2] N. Hawes, A. Sloman, J. Wyatt, M. Zillich, H. Jacobsson, G.-J. Kruijff, M. Brenner, G. Berginc and D. Skocaj, “Towards an Integrated Robot with Multiple Cognitive Functions”, in *Proc. Of the 22nd Conf. on Artificial Intelligence*, R.C. Holte, A. Howe (eds.), AAAI Press, Vancouver CDN, pp. 1548-1553, Jul. 2007

[3] M. Montemerlo, N. Roy, S. Thrun, “Perspectives on standardization in mobile robot programming: The Carnegie Mellon Navigation (CARMEN) Toolkit,” in *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, Las Vegas, Nevada, pp. 2436–2441., Oct. 2003

[4] H. Utz, S. Sablatnög, S. Enderle, G.K. Kraetzschmar, “Miro – middleware for mobile robot applications”, *IEEE Transactions on Robotics and Automation, Special Issue on Object-Oriented Distributed Control Architectures*, vol. 18, no. 4, pp. 493-497, Aug. 2002

[5] E. Woo, B.A. MacDonald, F. Trépanier, “Distributed mobile robot application infrastructure”, in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003)*, pp. 1475-1480, 2003

[6] T.H.J. Collett, B.A. MacDonald, B.P. Gerkey, “Player 2.0: Toward a Practical Robot Programming Framework”, in *Proceedings of the Australasian Conference on Robotics and Automation (ACRA 2005)*, Sydney AUS, Dec. 2005

[7] M. Quigley, B. Gerkey, K. Conley, J. Faust, T. Foote, J. Leibs, E. Berger, R. Wheeler, A. Ng, “ROS: an open-source Robot Operating System,” in *Proc. Of IEEE International Conference on Robotics and Automation*, 2009.

[8] R.C. Arkin, T.R. Balch, “AuRA: principles and practice in review”, *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 9, no. 2, pp. 175 – 189, Taylor and Francis, 1997

[9] L. Steels, "Language Games for Autonomous Robots", *IEEE Intelligent Systems*, vol. 16, no. 5, IEEE Press, pp. 16-22, Sep./Oct. 2001

[10] M.F. Bear, B.W. Connors and M.A. Paradiso, *Neuroscience: Exploring the Brain*, 3rd ed., Lippincott, Philadelphia, PA, 2006

[11] B.J. Baars, “The conscious access hypothesis: Origins and recent evidence”, *Trends in Cognitive Science*, vol. 6, no. 1, Elsevier Inc., pp. 47-52, Jan. 2002

[12] F. Rieke, D. Warland, R van Steveninck and W. Bialek, *Spikes: Exploring the Neural Code*, MIT Press, Cambridge MA, Jun. 1999

[13] E.R. Wood, P.A. Dudchenko, H. Eichenbaum, “The global record of memory in hippocampal neuronal activity”, *Nature*, vol. 397, Nature Publishing Group, pp. 613–616, 1999

[14] H.A. Mallot, K. Basten, “Embodied spatial cognition: Biological and artificial systems”, *Image and Vision Computing*, vol. 27, no. 11, Butterworth-Heinemann, Newton MA, pp. 1658-1670, Oct. 2009

[15] H.A. Mallot, M. Franz, B. Schälkopf, H.H. Bülthoff, “The view-graph approach to visual navigation and spatial memory”, in *Proc. Of the 7th International Conference on Artificial Neural Networks*, Springer Berlin / Heidelberg D, pp. 751-756, Oct. 1997

[16] A.P. Witkin, “Scale-space filtering”, in *Proc. 8th Int. Joint Conf. Art. Intell.*, Karlsruhe D, pp. 1019–1022, 1983

[17] M.S. Landy, Y. Cohen, G. Sperling, “HIPS: A unix-based image processing system.”, *Computer Vision, Graphics, and Image Processing*, vol. 25, no. 3, Elsevier Inc., pp. 331-347, Mar. 1984

[18] D.J. Wetherall, J. Guttag and D.L. Tennenhouse, “ANTS: A Toolkit for Building and Dynamically Deploying Network Protocols”, in *IEEE OPENARC’98*, San Francisco CA, Apr. 1998

[19] L. Sha and D. Seto, “A Distributed Real Time Coordination Protocol”, in *IPDPS 2000 Workshops*, LNCS 1800, J. Rolins et al. (Eds.), Springer-Verlag, Berlin Heidelberg D, pp. 671-677, 2000

[20] C. Schröter, M. Höchemer, S. Müller, H.-M. Gross, “Autonomous Robot Cameraman - Observation Pose Optimization for a Mobile Service Robot in Indoor Living Space”, in *Proc. Pf the 2009 IEEE Int. Conf. on Robotics and Automation*, IEEE Press, Piscataway NJ, pp. 2199-2204, 2009