2010 IEEE-RAS International Conference on Humanoid Robots
Nashville, TN, USA, December 6-8, 2010

The BERT2 infrastructure: An integrated system for the study of human-robot interaction

Alexander Lenz*, Sergey Skachek*, Katharina Hamann†, Jasmin Steinwender†,
Anthony G. Pipe* and Chris Melhuish*
*Bristol Robotics Laboratory, University of Bristol and University of the West of England,
Coldharbour Lane, Bristol, BS16 1QD, UK, Email: alex.lenz@brl.ac.uk
†Max Planck Institute for Evolutionary Anthropology, Deutscher Platz 6, 04103 Leipzig, Germany

***Abstract*— Bristol Elumotion Robot Torso Version 2 (BERT2) is a humanoid robot currently in development at Bristol Robotics Laboratory (BRL). In this paper we present the current state of development and demonstrate how the integration of several advanced subsystems (of commercial and non-commercial nature) within a heterogeneous computing infrastructure enables us to construct a unique platform ideally suited to investigate complex human-robot interaction (HRI). We particularly focus on two important domains of non-verbal communication, namely gaze and pointing gestures in a real-world 3D setting and outline our thinking in terms of safety, ambiguities and further experimental work.**

I. INTRODUCTION

RECENT robotics research and development have greatly advanced our engineering abilities in terms of object manipulation, verbal and non-verbal communication, motion, cognition and interaction. Most projects tackle these issues separately from each other in a well defined but limited context in order to make specific advances. Although these approaches are clearly important and have provided the community with a plethora of insights, we believe that at this stage it is also of great significance to combine several technologies and engineer a complete complex HRI system.

Part of the work presented here takes inspiration from a number of human-human interaction (HHI) studies which show that communication between two human agents incorporates several modalities which complement and enhance each other. Besides speech, there are facial and body gestures which humans combine and integrate into their communication stream. Often, humans use these additional modalities to express concepts which are difficult to represent with speech, e.g. spatial indications [1]. Furthermore, studies of children of a very young age describe usage of gaze following, visual perspective taking and pointing to cooperate in novel situations [2],[3],[4].

Studies from psychology have shown a close coupling between human visual attention and auditory perception or verbal utterances. Knowing what your communication partner is looking at during the course of a conversation can greatly enhance comprehension. Work by Griffin [5] has shown that gaze direction plays a role in the planning of speech production, and objects are typically gazed at $\approx 900ms$ before the beginning of the verbal utterance. Also, Altman and Kamide [6] present evidence that the likelihood of gazing at a particular object in a scene is modulated by what humans hear. Consequently, *our robot should be able to direct its head and eyes toward objects that score high in an internal attention system.* Furthermore, studies by Senju and Csibra [7] show that six month old infants follow an adult's gaze only when coupled with ostensive cues, including infant-directed gaze and the raising of eyebrows. This leads us to conclude that *a robot should be able generate a set of facial gestures, including eye-brow movements,* in order to communicate naturally with humans.

Now, gaze communication works both ways and *our robot should also be able to sense the human's head and eye gaze direction.* Recent work by Staudte and Crocker [8] has shown that human gaze is modulated by both the robot's speech and gaze[1]. A robot capable of evaluating human gaze ("What is the human presently looking at?") should display better cognitive performance when combining speech inputs with visual attention estimates. The ability of a robotic system to exploit the human face and body gestures could also greatly enhance behavioural safety aspects during HRI tasks when operating in a co-located space. For example, the robot's action velocities can be modulated by the human's gaze direction, indicating a change in the level of shared attention. Moreover, assuming that the human's perception system is still superior to the robot's, a sudden drastic change of human gaze or posture during interaction could be interpreted by the robotic system as a sign of caution. The human may have sensed something of high importance that was not perceived by the robot. The perception and evaluation of the human's gaze in this situation (as a secondary event) can be seen as an extension of the robot's senses. Here, it may be important to distinguish between head orientation and eye gaze. While sudden changes in the latter happen often during conversation (random eye saccades) and might often be ignored, a momentary change of head direction (associated with a greater energy expenditure) quite probably indicates an important shift of attention, that would be worth following.

The more explicit way of sharing attention and interest in HHI is achieved using body gestures. Liszkowksi et al.

[1]It remains to be shown if the results of these studies, conducted with humans looking at a film sequence of a robot on a screen, can be transferred into the "real 3D world" of HRI.

[4] have shown that 12-month-old children already point[2] declaratively at objects in order to share attention with an adult. Furthermore, pointing gestures are repeated when the adult does not share attention, suggesting not only that children perceive others as having psychological states that can be shared, but also demonstrating pointing as the preferred (non-verbal) method to initiate shared attention. In addition, earlier research on body gestures [9] has more formally affirmed that gestures and speech are synchronised. Consequently, for naturalness, intuitiveness and effectiveness of communication, *a humanoid robot should be able to do both, produce comprehensible human-like gestures and interpret human gestures*.

In the following sections, we will introduce selected attributes of our robotic infrastructure as work-in-progress toward a system capable of communicating efficiently using a similar range of channels available to humans. We will outline the systems engineering aspects necessary to combine several perception and action systems and introduce experimental scenarios which will be tackled in the future in order to deepen our understanding of HRI.

Our aims are twofold. Firstly, our platform provides a means to investigate multi-modal HRI. We are particularly interested in gaining an understanding of the engineering of a safe system in physical, cognitive and behavioural terms. Secondly, the systems installed are available to collaborators from psychology and anthropology for various experimental investigations.

II. HETEROGENEOUS COMPUTING AND COMMUNICATION INFRASTRUCTURE

From the onset of the work presented here it was clear that our goals could only be achieved by combining proprietary sub-systems with (free) open-source software and our own hardware and software developments. We decided to use YARP [10] as the "glue" in our heterogeneous computing infrastructure. YARP allows us to run all required software modules on a cluster of PCs running Windows™or Linux operating systems. YARP is an open source C++ library that allows the development of independent software modules that communicate with each other using named TCP/IP or UDP channels. Connection and disconnection of modules can be established during run-time. The YARP communication channels are named ports (e.g. /bert2/perception/eyeGaze) which can be employed as a data streaming interface or as remote-procedure-call (RPC) facilities.

When a software product with its own proprietary communication interface requires integration (e.g. the gaze tracker, section IV-A) the development of a YARP wrapper allows all data of interests to be made available to the rest of the system on clearly defined YARP ports.

[2]Pointing was coded as an extended arm (either fully or slightly bent) and index finger or open hand with palm down.

III. TORSO, ARMS AND HANDS

BERT2 (see Fig. 1) is an upper-body humanoid robot designed and, currently still under construction, at BRL in close co-operation with our mechanical engineering partner Elumotion[3]. The torso comprises four joints (hip rotation, hip flexion, neck rotation and neck flexion) and the hip rotation forms the most proximal joint to the rigid mounting base. Each arm is equipped with seven[4] degrees-of-freedom (DOF). The shoulder flexion joint forms the mounting point of the arm to the torso and the wrist flexion joint is the most distal joint of the arm. The wrist provides a mounting interface for a sophisticated humanoid hand (see below and Fig. 2) or a simple gripper.

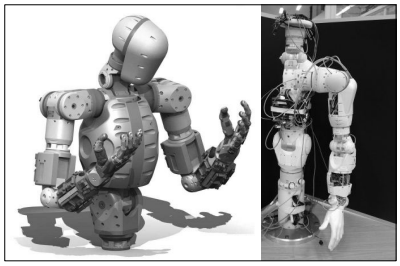

Fig. 1. 3D CAD model of BERT2 and the partially assembled torso at the current state of development. Head (see also Fig. 3), right arm and hands (see Fig. 2) are have been designed and build and we are currently in the process of assembly.

Each of these 18 joints is actuated by a brushless DC motor via a Harmonic Drive™gear box. Low level motor control is achieved through EPOS[5] motor controllers from Maxon Motor Company. In addition to the incremental encoders for accurate positioning, each joint is equipped with an absolute position sensor which eliminates the need for a "homing" phase during start-up. In order to detect contact with its environment, and to employ sophisticated control schemes, each joint is also equipped with a torque sensor. Both the absolute position sensor and the torque sensor are analogue devices which are interfaced to the EPOS controller via its two analogue input ports. The EPOS controllers are connected to a *controller area network* (CAN) which forms, in conjunction with a central-controller (essentially a Linux PC with a PCI to CAN interface), the joint-level control infrastructure.

The central-control PC's task is to provide a joint-level error detection and communication monitoring service as well as the implementation of an interface to the YARP infrastructure. Joint positions (absolute and encoder based) and torques are available as YARP streams and BERT2 can be driven from higher-level motor control modules via a position/velocity input stream.

[3]For further information on Elumotion's robotic hardware visit www.elumotion.com or enquire via Bristol Robotics Laboratory.

[4]The seven degrees are: shoulder flexion, shoulder abduction, humeral rotation, elbow flexion, wrist pronation, wrist abduction and wrist flexion.

[5]We employ the EPOS 70/10 for the larger motors and EPOS 24/5 for the smaller loads.

The hands (Fig. 2) are quite anthropomorphic in their mechanical design. Like the torso and arms, each of the nine[6] actuators for each hand is driven by a brushless DC motor and low level motor control is again established via EPOS controllers. All motors and gearing are fully integrated into the hand and (in contrast to many cable driven hands) no mechanical parts must be integrated into the robot arm. This approach makes the hand particularly modular and allows for simple integration into other robotic infrastructures. To facilitate integration and to allow higher-level hand movement commands (e.g. `grasp(.)`,`open(.)`,`approach(.)`,etc.), a drive system based on the dsPIC30F6014A digital signal controller from Microchip™has been developed. This embedded system provides the CAN interface to the individual EPOS controllers to drive each joint whilst taking care of issues of cross-coupling, homing and coordination. Access from and to higher level systems can be established via the on-board RS232, SPI or Ethernet interfaces. Additionally, this subsystem features 16 analogue channels which will be utilised with the employment of touch sensor on fingers and palm in the future.

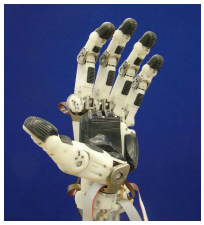

Fig. 2. The fully assembled anthropomorphic hand of BERT2

IV. HEAD

With the design of BERT2's head we implemented an important non-verbal communication channel, namely facial expression, with a particular emphasis on gaze, as used in human-human interaction. Although, in human-human communication the eyes are a bi-directional communication channel, we decided to separate the directions of communication in the case of BERT2's head for reasons of practicability, performance and robustness (see Fig. 3).

A robust face[11] has been designed and built that is capable of changing expression quickly, by mixing static face characteristics moulded into a plastic head, with dynamic characteristics of the eyes, eyebrows, mouth and lips provided by embedded colour LCD screens. A wide range of expression and fast changes are possible (including eye blinks and eye saccades) without the problems of motor noise or the unreliability inherent in miniaturised moving parts. The facial animations have been implemented using Max 5[7] and we have written a YARP wrapper application that provides a standard interface to control all facial parameters via RPCs.

[6]The nine DOF comprise gross movement for all 5 fingers, "trigger" action for index and middle finger, opposing of the thumb and finger spread.

[7]Max is a graphical development environment for music and multimedia developed and maintained by the software company Cycling '74.

Here, it is important to note that this realisation of a robot head does not aim to create an artefact with close resemblance to a human. In contrast to some of our previous work, where we used heads supplied by David Hanson[8] [12], the purpose of the work reported here is the design of a natural communication interface whilst maintaining an artificial, "robotic" look.

Fig. 3. BERT2's expressive head with embedded LCD screen to generate facial gestures. The eyes are 3D animations and are used to communicate the direction of attention to the human. The stereo vision system faceLAB from Seeing Machines™is mounted on the side of the head and is used as a head and gaze tracker. A webcam is mounted centrally in the forehead and can be used as a wide angle view of the scene.

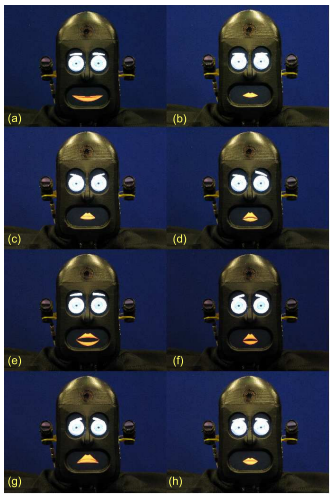

Fig. 4. The eight basic facial expressions are shown above: (a) happy, (b) thinking, (c) angry, (d) disgusted, (e) surprised, (f) afraid, (g) sad, (h) tired. The software implementation allows for a weighted combination of all basic expressions and smooth morphing between states.

A. Head and Gaze Tracking

In order to estimate the human's focus of attention we integrated the head and gaze tracking system faceLAB™from Seeing Machines. The gaze tracking system employs an infra-red stereo vision system to detect head pose and position, and high accuracy eye tracking is achieved using cornea reflection techniques[9]. As above, a YARP wrapper application has been written in order to provide a streaming interface of parameters like head gaze direction, head position and eye gaze direction.

[8]Web site: www.hansonrobotics.com.

[9]For further information see www.seeingmachines.com.

One problem when using gaze tracking techniques is their inherently narrow field of view. To overcome this problem we combined the gaze tracker with our own (wide-angle) face finding application, which we called FaceFinder. FaceFinder is based on the openCV face tracking library and uses the webcam that is mounted inside BERT2's the forehead as shown in Fig. 2. FaceFinder provides a YARP port that streams the relative position in x and y (range +/- 0.5). Additionally, FaceFinder provides a YARP port on which the actual camera image is streamed. The resolution is 320x240 pixels (8-bit RGB). The webcam FaceFinder combined with the faceLAB gaze tracking system resembles the human peripheral and foveal vision system. The detection of a human face in the peripheral areal of the webcam can be used as a trigger to re-direct the robot head towards the human. If, after this movement, the human is detected by the stereo vision system, head and gaze tracking starts automatically.

B. Speech Perception and Production

In order to enable spoken language interaction we have integrated the CSLU Toolkit [13] Rapid Application Development (RAD) into the BERT2 infrastructure. RAD allows the use of the TCL scripting language which permits binding with the YARP "backbone" of our system and consequently the fusion of action and perception with spoken dialogue.. RAD employs the Festival speech synthesis system and recognition is based on Sphinx-II. The construction of speech dialogues is supported via a state-based graphical programming environment.

V. THE 3D SCENE PERCEPTION

BERT2 uses the VICON motion capture (MoCap) system to detect and localise interaction objects and the human's body parts in 3D space. Although using a MoCap system as main vision facility for a humanoid robot may seem farfetched (as it is an external system that cannot be fitted to the robot), it allows us to divert away from the machine vision challenges typically encountered when using robot-mounted vision systems. Instead, we can directly focus on the subject of HRI. Our MoCap system is set up to have an accuracy sufficient to follow the motion of human body parts and environmental features and object using retro-reflective markers. However, details of the human hand proved to be difficult to track. The small distances between finger joints lead to false marker detection and reliable finger poses were impossible to obtain. Consequently, we opted for an alternative solution using a data glove from 5DT^{TM}.

Our modular perception software design follows a bottom up approach, starting with the development of the lower level detection modules towards gesture extraction.

Beneath the 3D perception software there is proprietary VICON hardware and software together with VICON object model templates. Additionally, a data glove system from 5DT^{TM}, including Application Programming Interface (API), enhances the MoCap in terms of hand gesture detection. The VICON software stores information about the marker topology of the objects to be captured. The 5DT API provides software access to the data glove features. Our 3D perception subsystem sits on top of this 3rd party software and comprises three modules: *GestureTracker*, *ObjectTracker* and *ViconLink*. *ViconLink* functions as a wrapper module which translates real time VICON data into the BERT2 data exchange infrastructure via a group of YARP ports. Any BERT2 module requiring raw MoCap data can connect to *ViconLink* ports and obtain it. Due to the different nature of algorithms necessary for the tracking of interaction objects and the human, these two aspects are handled by separate software modules. Both *ObjectTracker* and *GestureTracker* are largely based on YARP classes. The main purpose of *ObjectTracker* is to update the *EgoSphere* (see section VI-B) with the most recent object positions/orientations and to perform some low pass filtering to reduce the visual noise from the MoCap system. In its main loop the *ObjectTracker* constantly receives recent data from the *ViconLink*, smooths the data, and only updates the *EgoSphere* when the object translation or rotation exceeds certain thresholds. These thresholds are obtained automatically during system start-up by analysing the variance of the MoCap noise.

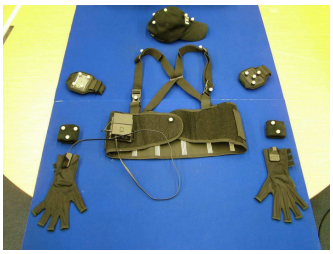

Fig. 5. Interaction garment.

The *GestureTracker* module requires the human interacting with BERT2 to wear a garment equipped with retro-reflective markers in order to be captured by the MoCap system. Hand gestures are sensed with the 5DT data gloves (see Fig. 5). The *GestureTracker* not only performs the gesture detection routines but also implements the calibration algorithms for the calculation of skeleton parameters, e.g. relative positions of shoulder and elbow joints and the centre of rotation of the neck. Calibration takes place during natural subject movements when the human enters the interaction area wearing the garment and is corrected during the interaction. The data glove calibration takes place simultaneously and the human is only required to perform a sequence of open palm to clenched fist movements.

VI. STATIC AND DYNAMIC KNOWLEDGE

During interaction with a human in the physical environment, the robotic system represents the state of the world via several databases handling static and dynamic knowledge.

A. Object Property Data Base (OPDB)

The *OPDB* is the common namespace manager and stores the static components of all objects in the interaction scenario. Here, the term object is applied to any rigid body that can be perceived by the system, including furniture, tools,

the robot's body parts, as well as the human's limbs, head and torso. The OPDB is implemented as a YARP wrapper around a relational database with queries and responses being communicated through a YARP port. As such, all other modules can access the information in the database. The database contains physical parameters of the objects, such as their cuboid bounding box (see Fig. 6 for an illustration) and symbols like the object's name for spoken language interaction. The database can also provide storage for sets of grasping points for object handling if appropriate. Each object which is known to the system (and can be perceived and represented in the EgoSphere) has a unique object identification number referred to as the *objectID*, which serves as a key across BERT2's databases.

B. EgoSphere

The first layer of abstraction between the sensory perception systems, higher level cognitive architectures and motor control elements is formed by the *EgoSphere*. Unlike the sensory ego-sphere (SES) by Peters[14] which implements short term memory, associations and direction of attention in addition to localisation, our simpler implementation solely acts as a fast, dynamic, asynchronous store of object positions and orientations.

The object positions are stored in spherical coordinates (radius, azimuth and elevation) and the object orientation is stored as rotations of the object reference frame about the three axes (x,y,z) of a right-handed Cartesian world-frame system. The origin of the world frame can be chosen arbitrarily and, for our experimental work, we located it at the centre of the robot's base-frame. Other stored object properties are a visibility flag and the *objectID*.

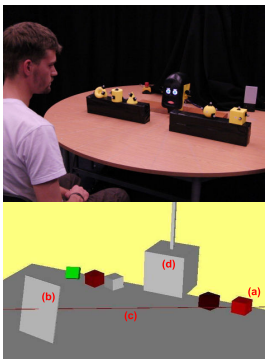

Fig. 6. Photograph of the actual interaction situation and our real-time visualisation of the EgoSphere content: Several objects (a) are represented by their cuboid bounding box. The human head interacting with the robot is represented by object (b). The human gaze, intersecting with an object is represented by the line (c) and the robot's head is labeled (d).

Any perception system adds objects to the *EgoSphere* when they are perceived and varies position, orientation or visibility according to its latest position and orientation estimates. The *EgoSphere* automatically assigns an *egoSphereID* to an added object. Any part of the robot's control infrastructure requiring spacial information from objects can query the *EgoSphere*. No assumptions are made about the nature of an object and any further information will have to be gathered from possible additional databases using the *objectID*. This architecture makes the *EgoSphere* particularly useful for storing multi-modal information.

We understand that such a platform-independent abstraction layer introduces additional computation between the perception and reaction modules. However, our experience when integrating several software modules leads us to conclude that the benefits of such an abstraction (unified access, platform independence) outweigh the costs of latencies associated with additional computational layers.

The *EgoSphere* is implemented in C++ as a client-server system using the YARP infrastructure. Any software module requiring access to the *EgoSphere* includes a client class which provides methods like `addObject(.)`, `setObject(.)`, `getObject(.)` or `getNumberOfObjects(.)`, etc. Clearly, at the current state, the *EgoSphere* is merely a convenient abstraction layer. With increasing complexity of HRI tasks during the course of our research, we plan to add further complexity (focus of attention, confidence, timeliness etc.) whilst preserving modularity.

C. Gaze - Object - Intersection

The *gazeObjectIntersection* module we have developed is perhaps one level up the "reasoning chain". This part of our system provides the robot with information about the object in the environment which the human is currently looking at (represented by its *egoSphereID*) and an estimation of the coordinates of intersection.

In order to achieve this, the *gazeObjectIntersection* system polls the *egoSphere* cyclically ($f_{egoPoll} = 50Hz$) to obtain the current position and orientation of all objects. This information is combined with the object properties from the OPDB. The momentary scene is then dynamically modelled as a selection of cuboids (with the size of the bounding box of each object) in a 3D Cartesian space.

From the gaze tracking module, the eye gaze direction (yaw and pitch in the human head centric coordinate system) is obtained as well as the position of the human head (represented in robot head centric coordinates). With this information, the appropriate coordinate transformations are performed in real-time and the human head and gaze vector are added to the 3D world model (see Fig. 6). With all elements in place, the *gazeObjectIntersection* system now calculates if and where the human gaze intersects with the boundary box of an object, taking into account mutual object occlusion (e.g. a small object placed behind a large object may not be visible). Finally, the *egoSphereID* of the object and the coordinates of intersection (in the world coordinate system) are provided as a stream on a YARP port. An update is provided every time the gaze tracking module provides a new gaze estimate (typically $f_{gaze} = 60Hz$).

In addition to this real-time information, the *gazeObject-Intersection* module can be re-configured during runtime to provide a list of objects the human could see from his or her position. This ability of "Level 1" [15] perspective taking is achieved by obtaining object position, orientation, size and

human head position, as described above. Instead of taking the current estimate of the human gaze direction, the gaze directions are simulated in yaw and pitch in a range of $\pm 45°$ with a resolution of $1°$. The result of these calculations is the set of objects visible to the human (identified by their *egoSphereID*) which is made available as a YARP stream.

VII. CONCLUSIONS AND FUTURE WORK

We have presented an architecture comprising hardware and software of commercial and non-commercial nature that is combined with our own developments in order to provide an open and extensible integrated system to study human-robot interaction. Due to the brevity of this paper we focused on perception aspects of non-verbal communication, namely head and eye gaze and pointing gestures, omitting motor-control issues. Our work is strongly motivated by studies from psychology, particularly with respect to infants and children.

As a next step we have begun experimental work with adults interacting with our robot. The first two sets of experiments will create a pool of human behavioural data consisting of eye gaze (see figures 6 and 7) and gestures directed at external entities. Situated in the context of human-robot interaction, the data will be used to find algorithms enabling the robot to extract meaning from the human's non-verbal communicative means of gaze and pointing. Key questions focus on the duration of eye gaze; the difference between eye gaze and head gaze; the joint angles constituting pointing gestures and measures of urgency in both gaze and pointing.

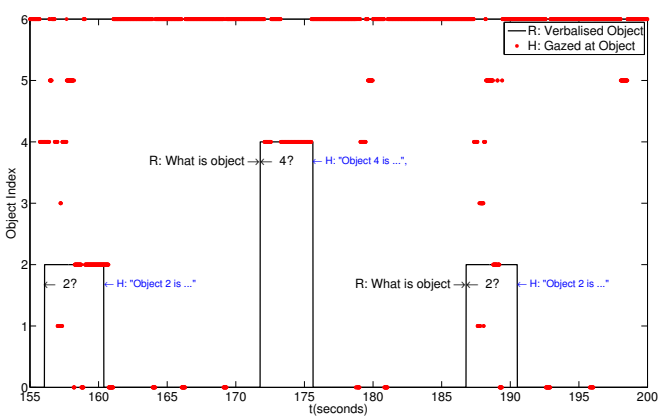

Fig. 7. Example of HRI data: In this experiment the robot (R) asks the human (H) to describe a particular object, i.e. "R: What is object 4?". The fine solid line represents the verbalised object and the dotted line represents the objects the human looks at. We are interested in the timing of the human gaze with respect to the robots verbal reference. Object 6 is the robot head and objects 1 to 5 are the ones shown in figure 6.

Further experiments will address the issue of non-verbal human-robot interaction from a different angle, namely, the perspective of the user. Here, the human's reactions to different types of visual communication produced by BERT2 will be assessed.

In the longer term, and with the completion of BERT2, gaze and pointing (from the robot and the human) will be combined with verbal communication, to allow safe human-robot interaction in a co-located space, addressing ambiguity, shared attention and context awareness in an uncertain environment.

VIII. ACKNOWLEDGMENTS

The authors gratefully acknowledge the funding provide by the European Commission under the Robotics and Cognitive Systems, ICT Project CHRIS (FP7-215805). Furthermore, we would like to thank our collaborators on the CHRIS project for their insights, useful comments and challenging questions.

REFERENCES

[1] A. Kendon, *Gesture: Visible Action as Utterance*. Cambridge University Press, Oct. 2004.

[2] M. Tomasello, M. Carpenter, J. Call, T. Behne, and H. Moll, "Understanding and sharing intentions: The origins of cultural cognition," *Behavioral and Brain Sciences*, vol. 28, no. 05, pp. 675–691, 2005.

[3] F. Warneken, F. Chen, and M. Tomasello, "Cooperative activities in young children and chimpanzees," *Child Development*, vol. 77, no. 3, pp. 640–663, Jun. 2006, PMID: 16686793. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/16686793

[4] U. Liszkowski, M. Carpenter, A. Henning, T. Striano, and M. Tomasello, "Twelve-month-olds point to share attention and interest," *Developmental Science*, vol. 7, no. 3, pp. 297–307, Jun. 2004, PMID: 15595371. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/15595371

[5] Z. M. Griffin and K. Bock, "What the eyes say about speaking," *Psychological Science*, vol. 11, no. 4, pp. 274–279, Jul. 2000. [Online]. Available: http://pss.sagepub.com/content/11/4/274.abstract

[6] G. T. Altmann and Y. Kamide, "Discourse-mediation of the mapping between language and the visual world: Eye movements and mental representation," *Cognition*, vol. 111, no. 1, pp. 55–71, 2009. [Online]. Available: http://ukpmc.ac.uk/articlerender.cgi?artid=1736498

[7] A. Senju and G. Csibra, "Gaze following in human infants depends on communicative signals," *Current Biology*, vol. 18, no. 9, pp. 668–671, 2008. [Online]. Available: http://www.cell.com/current-biology/abstract/S0960-9822(08)00455-7

[8] M. Staudte and M. W. Crocker, "Visual attention in spoken human-robot interaction," in *Proceedings of the 4th ACM/IEEE international conference on Human robot interaction*. La Jolla, California, USA: ACM, 2009, pp. 77–84. [Online]. Available: http://portal.acm.org/citation.cfm?id=1514111

[9] D. McNeill, *Psycholinguistics: A New Approach*. Harpercollins College Div, Feb. 1987.

[10] P. Fitzpatrick, G. Metta, and L. Natale, "Towards long-lived robot genes," *Robot. Auton. Syst.*, vol. 56, no. 1, pp. 29–45, 2008. [Online]. Available: http://portal.acm.org/citation.cfm?id=1327705

[11] D. Bazzo, R. Vaidyanathan, A. Lenz, and C. Melhuish, "Design and testing of a hybrid expressive face a humanoid robot," accepted for the Proceedings of the 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2010.

[12] P. Jaeckel, N. Campbell, and C. Melhuish, "Facial behaviour mapping-From video footage to a robot head," *Robot. Auton. Syst.*, vol. 56, no. 12, pp. 1042–1049, 2008. [Online]. Available: http://portal.acm.org/citation.cfm?id=1458886

[13] S. Sutton, R. Cole, J. D. Villiers, J. Schalkwyk, P. Vermeulen, M. Macon, Y. Yan, E. Kaiser, B. Rundle, K. Shobaki, P. Hosom, A. Kain, Johan, J. Wouters, D. Massaro, and M. Cohen, "Universal speech tools: The cslu toolkit," *IN PROCEEDINGS OF THE INTERNATIONAL CONFERENCE ON SPOKEN LANGUAGE PROCESSING (ICSLP*, vol. 7, pp. 3221—3224, 1998. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.40.7220

[14] I. R. A. Peters, K. A. Hambuchen, and R. E. Bodenheimer, "The sensory ego-sphere: a mediating interface between sensors and cognition," *Auton. Robots*, vol. 26, no. 1, pp. 1–19, 2009. [Online]. Available: http://portal.acm.org/citation.cfm?id=1487424

[15] J. H. Flavell, "Perspectives on perspective taking," in *Piaget's theory*. Routledge, 1992, pp. 107–139.