

Generic method for generating blended gestures and affective functional behaviors for social robots

Greet Van de Perre^{1,2}  · Hoang-Long Cao^{1,2} · Albert De Beir^{1,2} · Pablo Gómez Esteban^{1,2} · Dirk Lefeber^{1,2} · Bram Vanderborght^{1,2}

Received: 10 May 2016 / Accepted: 23 June 2017 / Published online: 5 July 2017
© Springer Science+Business Media, LLC 2017

Abstract Gesturing is an important modality in human–robot interaction. Up to date, gestures are often implemented for a specific robot configuration and therefore not easily transferable to other robots. To cope with this issue, we presented a generic method to calculate gestures for social robots. The method was designed to work in two modes to allow the calculation of different types of gestures. In this paper, we present the new developments of the method. We discuss how the two working modes can be combined to generate blended emotional expressions and deictic gestures. In certain situations, it is desirable to express an emotional condition through an ongoing functional behavior. Therefore, we implemented the possibility of modulating a pointing or reaching gesture into an affective gesture by influencing the motion speed and amplitude of the posture. The new implementations were validated on virtual models with different configurations, including those of the robots NAO and Justin.

Keywords Generic gesture system · Pointing · Gestures · Upper body postures · Affective gesture

1 Introduction

Body language is a crucial feature in human communication. Facial expressions, body posture and gestures all convey information about a person's internal state, and contribute to the overall effectiveness of communication. It has been

shown that these features also benefit the interaction between humans and robots by ensuring a more fluent and natural communication (Park et al. 2011; Scheutz et al. 2007; Salem et al. 2013; Breazeal et al. 2005). Different research teams have implemented gestures in robots, in the light of investigating different aspects of communication. Since these gestures are generally preprogrammed off-line for a specific robot configuration (Sugiyama et al. 2007; Ido et al. 2006; Zecca et al. 2009), or generated by mapping motion capture data to the robot's configuration (Matsui et al. 2005; Tapus et al. 2012; Do et al. 2008), they are robot-specific and not easily transferable to other robots. To offer a solution to this issue, which is known as the correspondence problem (Dautenhahn and Nehaniv 2002; Alissandrakis et al. 2002), we developed a generic method to generate gestures for social robots. By storing target gestures independently of a configuration and calculating a mapping based on a random configuration chosen by the user, gestures can be calculated for different robots.

Since for different types of gestures, different features are important, our method was designed to work in two modes (Fig. 1). The *block mode* is used to calculate gestures whereby the overall arm placement is crucial, like for emotional expressions (Van de Perre et al. 2015). The *end effector mode*, on the other hand, is developed for end-effector depending gestures, i.e. gestures whereby the position of the end-effector is important, like for manipulation and pointing (Van de Perre et al. 2016). The working principles and results of the block and end-effector mode were presented in detail in previous publications. In this paper, we focus on how these two modes are combined to generate blended deictic gestures and emotional expressions, and how information about the current emotional condition can be used to modify functional behaviors calculated by the end-effector mode into affective motions.

✉ Greet Van de Perre
Greet.Van.de.Perre@vub.ac.be

¹ Robotics and Multibody Mechanics Research Group, Vrije Universiteit Brussel, Brussels, Belgium

² Flanders Make, Brussels, Belgium

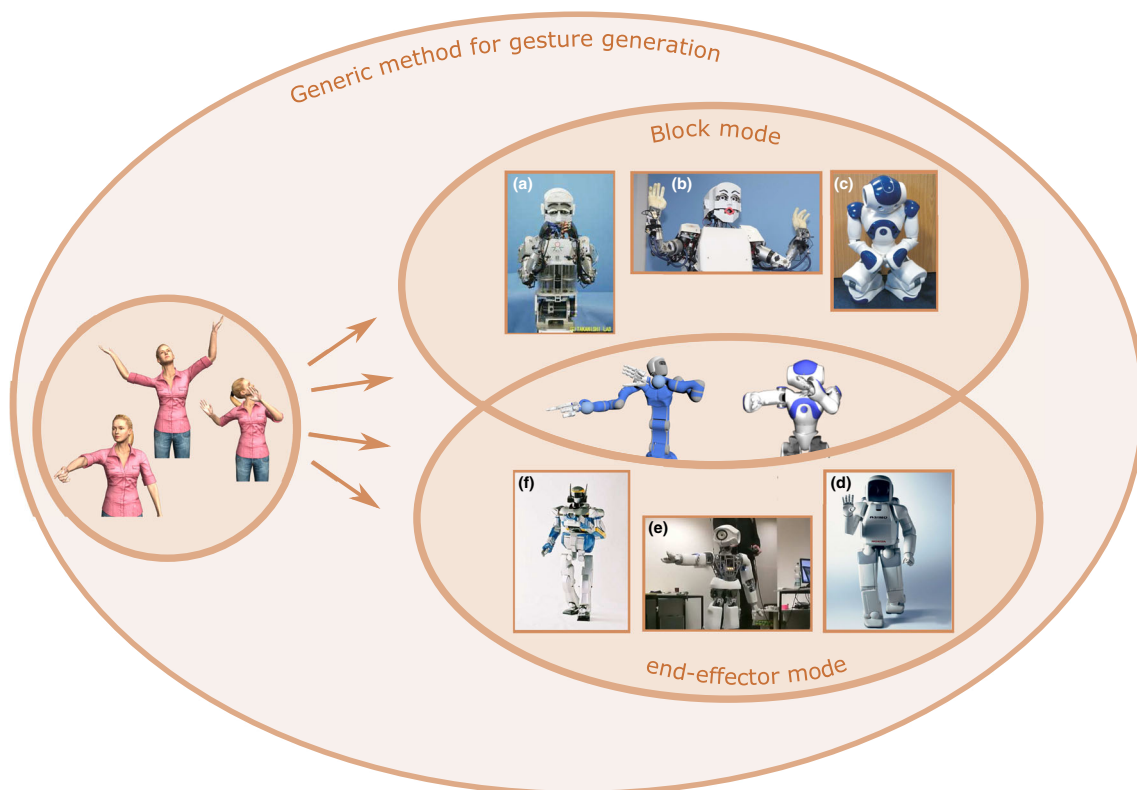


Fig. 1 This schema represents the methodology of the developed gesture method, aiming to overcome the limitations of the current state of art where gestures are implemented for specific robot. The method uses a human base model to store target gestures independently of any configuration in a database, and to calculate a mapping at runtime, based on the robot configuration specified by the user. Two modes are used to allow for different types of gestures to be calculated. The *block mode* is used to calculate gestures whereby the overall arm placement is cru-

cial, like for emotional expressions, while the *end-effector mode* was developed for end-effector depending gestures, like for deictic gestures. This paper focusses on how the two modes can be combined to generate blended emotional and deictic gestures, and how information concerning the emotional state can be used to modulate functional behaviors into affective motions. Robots: **a** WE-4RII (Itoh et al. 2004), **b** KOBIAN (Zecca et al. 2009), **c** NAO (Belpaeme et al. 2012), **d** ASIMO (Salem et al. 2009), **e** Myon (Hild et al. 2012), **f** HRP-2 (Hirukawaa et al. 2004)

2 Related work

Different attempts are made to ease the animation of social robots. Balit et al. (2016) suggested to use the knowledge of animation artists to generate lifelike robotic motions by providing a generic software, whereby different types and combinations of gestures can be created by keyframing or by 3D character articulation. However, since the generated motions are still dependent on the used joint configuration, this does not address the correspondence problem.

A possible solution lies in the field of developmental robotics, by using neural networks to learn the correspondence between a posture and the robot's joint angles (Andry et al. 2001). A technique to teleoperate a humanoid robot without an explicit kinematic modeling by using neural networks was proposed by Stanton et al. (2012). Mühlig et al. (2012) ease the correspondence problem between a human tutor and robot in imitation learning by representing demonstrated movement skills using a flexible task space

representation. Another approach of addressing the correspondence problem in imitation learning was suggested by Azad et al. (2007), by using a reference kinematic model, the Master Motor Map, to convert motion capture data to an arbitrary robot morphology. This is a similar strategy as we use to map target gestures from a database to a robot configuration (see Sect. 3.1). In a later stage, the Master Motor Map was extended with a dynamic model and improved to allow for on-line reproduction of human motion to a humanoid robot (Terlemez et al. 2014). In Koga et al. (1994), a semi-general approach for generating natural arm motions, specifically for manipulation tasks is presented. Their inverse kinematics algorithm is based on neurophysiological findings, and decouples the problem of calculating joint angles for the arm from calculating those for the wrist. The sensorimotor transformation model of Soechting and Flanders (1989) is used to determine the arm posture, while the wrist angles are found by assuming a spherical wrist and using orientation inverse kinematics.

In both Salem et al. (2010) and Le et al. (2011), a gesture framework initially developed for virtual agents is applied on a humanoid robot. In Salem et al. (2010), speech-accompanying gestures are generated for ASIMO by using the speech and gesture production model initially developed for the virtual agent MAX. For a specified gesture, the end effector positions and orientations are calculated by the MAX system and used as input for ASIMO's whole body motion controller (Gienger et al. 2005). Similarly, in Le et al. (2011), speech-accompanying gestures are generated for NAO by using the GRETA system. The gestures are described independently of the embodiment by specifying features as the hand shape, wrist position and palm orientation. However, to obtain the corresponding joint values, a predetermined table listing values for the shoulder and elbow joints for all possible wrist positions is used. So although the gestures are described independently of the robot configuration, mapping these gestures to the robot requires hard coded joint information.

An interesting feature of this framework however, is the possibility of generating affective motions by modulating a neutral behavior using a set of expressivity parameters. In that way, it is possible to convey an emotional state through an ongoing functional behavior. This is indeed an important feature in human communication. Lots of research has been performed on how an emotional state is reflected in the motions generated by a human. Wallbott (1998) found that both the quantity as the quality of the emotion influences the generated body movements. A number of studies investigated the effect of different emotions on human gait (Montepare et al. 1987; Crane and Gross 2007), while other focussed on addressing affect to static body postures (James 1932; Coulson 2004; Atkinson et al. 2004). A number of researchers focussed on affective arm movements (Pollick et al. 2001) and whole body motion (Meijer 1989; Castellano et al. 2007), whereof a number of researches were directed to the effect of affect on dance (Dittrich et al. 1996; Castellano et al. 2007). Using the knowledge obtained by these numerous researches, it is possible to create behaviors conveying emotional information by modifying neutral motion patterns (see Sect. 5).

An interesting aspect of our method, is that, in contrast to learning techniques, no training is required. The method can directly calculate gestures for a chosen configuration. Furthermore, the method allows an impressive amount of flexibility, both for the desired robot configuration as for the targeted body motion. The method can be used for a collection of robots and virtual models that consist at least of one arm, a body, or a head (see Sect. 3). Regarding the body motions, our developed system uses two separate modules to calculate different types of gestures. Next to modulating a certain neutral gesture into affective motions, it is also possible to combine different types of gestures into one blended

gesture. An emotional expression in the sense of an explicit, full body action as calculated by our block mode, can take place in combination with a deictic gesture as calculated by our end-effector mode, by assigning each gesture to other body parts. How the modes are combined to generate such a blended body motion is handled in Sect. 4. But first, to get a better understanding of the method, the working principles of the two modes are briefly repeated in Sect. 3. In situations where it is desirable to express an emotional condition not by using explicit bodily expressions, it can be expressed by modifying an ongoing functional behavior. How this is implemented in the method is handled in Sect. 5. A number of results are discussed throughout Sects. 4 and 5. We conclude this paper by a summary and some details about current developments in Sect. 6.

3 Working principles of the method

To ensure a generic method usable for different kind of robots, the framework was developed without using any kind of robot morphology. Instead, a simplified model of the rotational possibilities of a human is used; the human base model. Firstly, a set of Body Action Units (BAU's) was defined, based on the human terms of motion. The defined BAU's are listed in Table 1. The units were grouped into different *blocks*, corresponding to one human joint complex, such as the shoulder or the wrist. These blocks can subsequently be grouped into three body parts, namely the head, body and arm, which we refer to as *chains*. In that way, our human base model was defined. A standard reference frame was defined, whereby the x -axis is located in the walking direction and the z -axis is pointing upwards, and subsequently, a reference frame was assigned to each joint block (see Fig. 2). When a user desires to generate gestures for a certain robot or model, its morphological information is specified by inputting a limited amount of rotational information and the configuration's Denavit–Hartenberg (DH) parameters into the program, whereby the different joints of the robot are grouped into the chains and blocks of the human base model. As such, the method can be used for any robot that consists at least of one arm, a body, or a head.

3.1 Block mode

The block mode is used for gestures whereby the overall placement of the arms is important, such as for emotional expressions. In this mode, the method uses a set of target gestures stored in a database and maps them to a selected configuration. To ensure a good overall posture, it is not sufficient to only impose the pose of the end effector, since inverse kinematics for robots with a different configuration and different relative arm lengths could result in unrecognisable global postures. Therefore, the orientation of every

Table 1 The body action coding system

Chain	Block	BAU	Description
Head	Head	1	Flexion/extension of neck
		2	Abduction/adduction of neck
		3	Rotation of neck
Body	Body	4	Flexion/extension of spinal column
		5	Lateral flexion of spinal column
		6	Transversal rotation of spinal column
Arm	Clavicle	7	Abduction/adduction of shoulder girdle
		8	Elevation/depression of shoulder girdle
	Shoulder	9	Flexion/extension of shoulder
		10	Abduction/adduction of shoulder
		11	Inward/outward medial rotation
	Elbow	12	Flexion/extension of elbow
	Wrist	13	Pronation/supination of elbow
		14	Flexion/extension of wrist
15		Abduction/adduction of wrist	

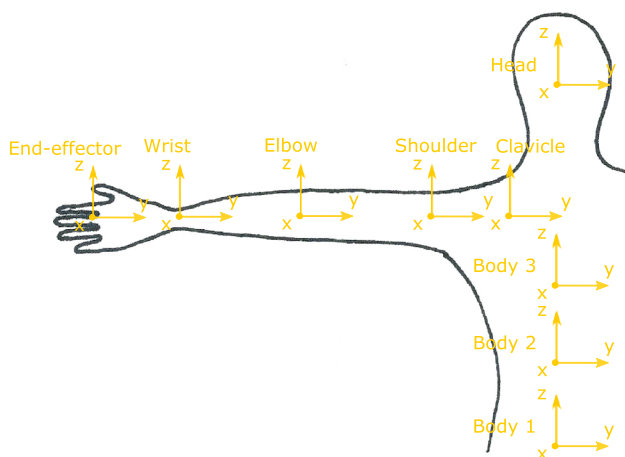


Fig. 2 A reference frame was assigned to each block. For the body 1 block, the reference frame is the standard reference frame. The body 2 and body 3 axes are respectively, the body 1 and body 2 embedded axes. The head and clavicle's reference axes are the body 3—embedded axes. For all other blocks of the arm, the axes are the embedded axes of the previous block

joint complex the robot has in common with a human needs to be imposed. Hence, the target gestures are stored in the database by specifying the orientation of every joint block i of the base model using the orthopaedic angles (Kadaba et al. 1990) of frame $i + 1$ (the base frame of block $i + 1$) with respect to frame i (the base frame of block i) (see Fig. 2). To make a robot or model perform a selected expression, a mapped rotation matrix for every present joint block is calculated by combining the information from the database and the morphological data specified by the user:

$$R_i = {}^{b,i}R_{st} \cdot R_{i,des} \cdot {}^{st}R_{e,i} \quad (1)$$

Here, R_i is the mapped rotation matrix for block i , ${}^{b,i}R_{st}$ the rotation matrix between the base frame of block i and the standard reference frame, $R_{i,des}$ the target rotation matrix in standard axes for block i , loaded from the database and ${}^{st}R_{e,i}$ the rotation matrix between the standard reference frame and the end frame of block i , i.e. the base frame of block $i + 1$.

These mapped matrices serve as input for an inverse kinematics algorithm to calculate the necessary joint angles to make the specified robot configuration perform the desired expression. Using the Runge–Kutta algorithm (Ascher and Petzold 1998), for every block, the angle values q are obtained from their derivatives \dot{q} , calculated by the following algorithm (Sciavicco 2009):

$$\dot{q} = J_A^\dagger(q) (\dot{x}_d + K(x_d - x_e)) + (I - J_A^\dagger(q)J_A(q)) \dot{q}_0 \quad (2)$$

Here, x_d is the desired end effector pose. Since the maximum number of joints in one block is three, it is not necessary to use all six parameters of the pose; the consideration of the orientation of the end effector is sufficient. Therefore, x_d is reduced to the zyx –Euler angles corresponding to the mapped rotation matrix. $J_A^\dagger(q)$ is the Moore–Penrose pseudo inverse of the analytical jacobian $J_A(q)$. Since only rotational information is imposed, $J_A(q)$ is reduced to its rotational part only. x_e is the current end effector pose; i.e. the current zyx –Euler angles, and K a positive definite gain matrix. Since the different blocks are treated separately, no redundancy is present, causing the second term $(I - J_A^\dagger(q)J_A(q)) \dot{q}_0$ to be zero.

3.2 End-effector mode

The end-effector mode is used for gestures whereby the position of the end-effector is crucial, like for deictic gestures. In some situations, for example when reaching for an object, the position of the right and/or left hand is important and specified by the user. This situation is called the *place-at* condition. The specified position then serves as a basis to calculate the necessary end-effector position for the selected chain, which is used as input for the same inverse kinematics algorithm as used in the block mode (Eq. 2). While in the block mode, a constraint is imposed on the end-effector of every block and the inverse kinematics algorithm is used to calculate the joint angles of every block separately, in the end-effector mode a constraint is imposed on the end-effector of the chain, and the algorithm is used to calculate the joint angles of the chain as a whole. Since in the end-effector mode the position is specified, the desired end effector pose x_d is limited to positional information only, reducing $J_A(q)$ to its

translational part only. In the highly probable case of an arm chain consisting of more than three degrees of freedom, the functional redundancy is used to guide the configuration into a natural posture. In that case, the second term of Eq. 2 will differ from zero, activating the influence of \dot{q}_0 on the calculated joint speeds. \dot{q}_0 introduces the cost function $w(q)$:

$$\dot{q}_0 = k_0 \left(\frac{\partial w(q)}{\partial q} \right)^T \quad (3)$$

with k_0 a positive weight factor. For the cost function w , we decided to work with a slightly adapted form of the joint range availability (JRA) criterion (Jung et al. 1995), whereby an optimal human like posture is calculated by keeping the joints close to a selected set of minimum posture angles (see our previous publication Van de Perre et al. 2016):

$$w = \sum_{i=1}^n w_{0,i} \frac{(q_i - q_{mi})^2}{(q_{max,i} - q_{min,i})^2} \quad (4)$$

Here, q_i is the current value of joint i and q_{mi} the minimum posture angle for that joint. $q_{max,i}$ en $q_{min,i}$ are the maximum and minimum joint limits, and $w_{0,i}$ a weight factor for joint i . The *pointing* condition functions in a same way as the *place-at* condition, apart from the fact that by specifying a desired pointing position, no direct constraint is imposed on the end-effector. A series of configurations with a specific combination of end-effector position and orientation can fulfil the pointing constraint. When pointing to an object, the index finger is directed towards the object, implying that for a certain position of the end-effector, the orientation needs to be chosen along the connection line between the object and the last wrist joint. Or with other words, the extension of the end-effector needs to pass the selected target position. To calculate the different possible postures, the end-effector is gradually virtually extended and the pointing position is imposed on the virtual end-effector. For every virtual length, the optimal configuration is calculated. From the resulting collection of postures, the cost function finally selects the optimal result by comparing the total cost of every configuration.

Before calculating a trajectory to the desired end-effector position, the possibility of reaching this position by the current configuration is checked by using an approximate calculation of the workspace. If the desired end-effector position is indeed located in the workspace of the robot, a suitable trajectory towards this position is calculated. In case of a reaching gesture towards a position located outside the workspace, the pointing-condition can be activated, and a trajectory towards a suitable posture for a pointing gesture is calculated instead.

4 Blended gestures

4.1 Priority levels

During natural communication, humans use and combine different types of gestures. By combining the two modes of our method presented above, it is possible to generate blended emotional expressions and deictic gestures. In order to do so, priority levels for each chain are assigned to both gesture types and a *mode mixer* was designed. If the mode mixer is turned off, all gestures are treated separately; starting a new gestures entails a previously started gesture to be aborted. By enabling the mode mixer, different gestures are blended by considering for every chain, only the end-effector condition(s) corresponding to the gesture with the highest priority level. The priority levels are defined using a number of rules:

- For an emotional expression, the priority level for each chain is set on a basic level (level 1)
- A deictic gesture has a higher priority than an emotional expression: the chain corresponding to the pointing/reaching arm receives a higher priority level (level 2)
- Similarly, gazing has a higher priority than an emotional expression: the head chain receives a higher priority level (level 2)

For every separate chain, the highest priority level present determines which gesture needs to be considered for that chain. The corresponding calculation principles are enabled, and the required constraints are loaded for the different chains: orientational information for every block composing the chain for the *block mode*, or the desired end-effector position for the complete chain for the *end-effector mode*.

When, for example, an emotional expression is performed in combination with a left handed deictic gesture, the left arm chain has a level 1 priority for the emotional expression but a level 2 for the deictic gesture. Therefore, for that chain, the pointing position is considered and the end-effector mode will calculate the corresponding joint angles. For the other chains, only priority level 1 is present. Therefore, the block mode will calculate the joint angles for all blocks in the remaining chains.

Figure 3 schematically summarizes how the mode mixer and the priority levels determine the imposed constraints, while Fig. 4 visualizes the work flow of one iteration, depending on the priority levels.

4.2 Examples of blended gestures

Figure 5 illustrates the calculation of a blended gesture for both the robots NAO and Justin. The left part of Fig. 5 shows

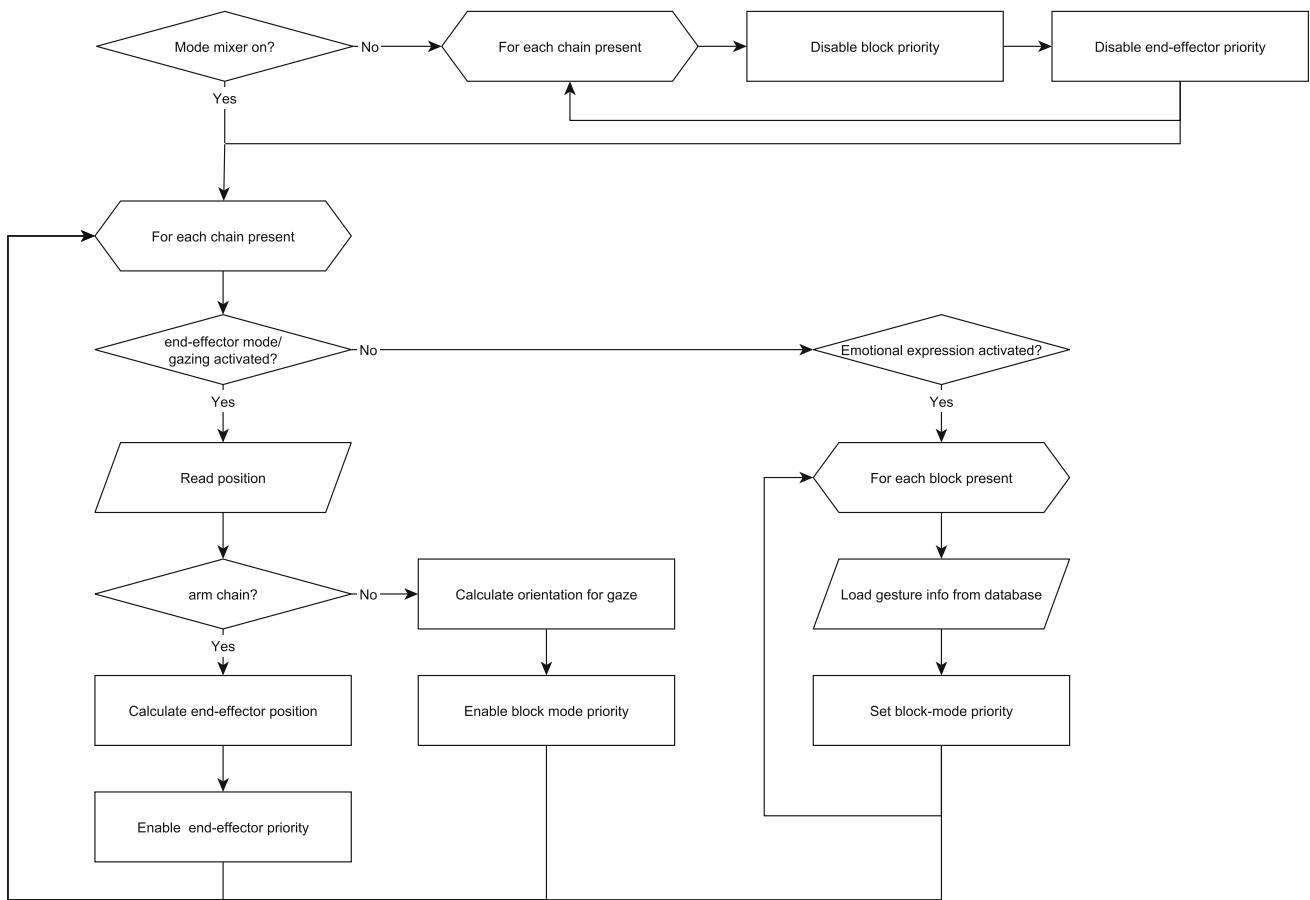
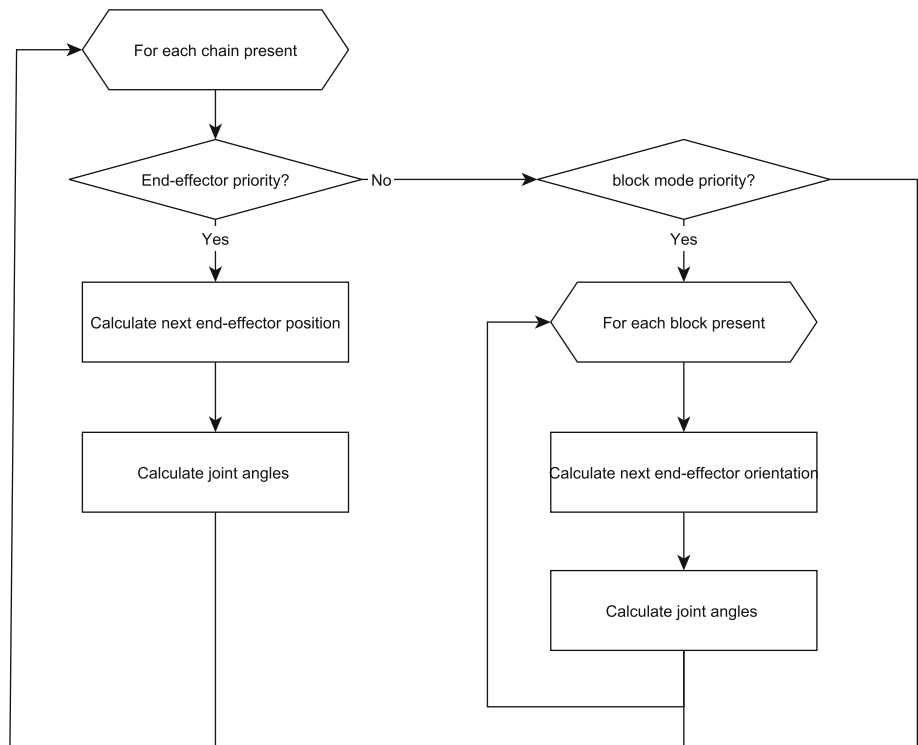


Fig. 3 Schematic representation of how the end-effector constraints are determined by the motion mixer and the priority levels

Fig. 4 Work flow of one iteration, depending on the priority levels



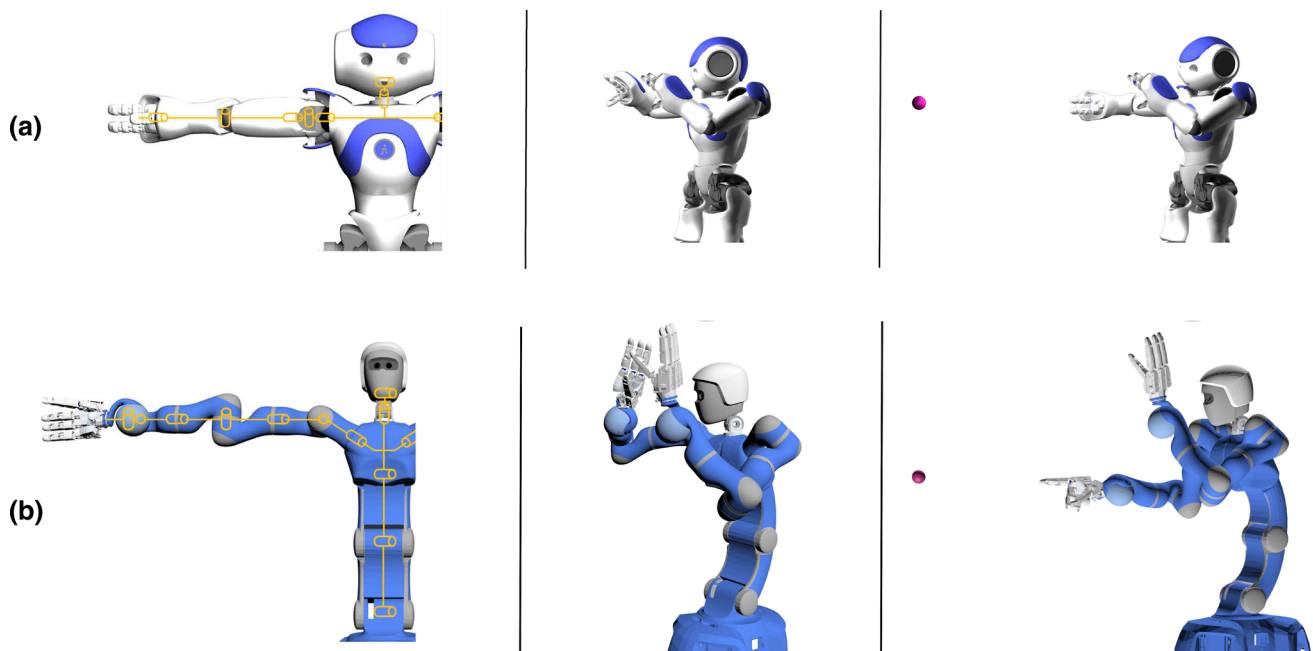


Fig. 5 Example illustrating the calculation of a blended gesture for **a** NAO **b** Justin. *Left* joint configuration of the robot. *Middle* calculated end posture for the emotional expression of fear. *Right* calculated end posture for a combination of a pointing gesture with the right arm, and the emotional expression of fear

the joint configuration of the robots. NAO has an actuated head, and a left and right arm of 5 DOF. Justin's arms, on the other hand, contain 7 DOF, with a remarkable different configuration. In addition, Justin has an actuated body with 3 DOF. The middle of Fig. 5 displays the end posture for the emotional expression of fear, calculated by the block mode. For the right part of Fig. 5, the mode mixer was enabled and a combination of gestures was demanded. Next to the expression of fear, a pointing gesture with the right arm was desired, accompanied by gazing towards the pointing location. As explained above, the priority levels determine which calculation principle is activated for every chain, and which corresponding end-effector conditions need to be used. For the expression of fear, all present chains have priority level 1. However, the priority of the pointing gesture for the right arm is higher than the basic level; level 2. Therefore, for the right arm chain, the end-effector mode is activated, whereby the end-effector condition is determined by the desired pointing position (see Sect. 3.2). For all the other chains present, the block mode is activated. Since the priority of gazing towards a specified position for the head overrules that of the emotional expression, the necessary rotation matrix to obtain the desired gazing direction is imposed. For the left arm chain, and body chain in case of Justin, the mapped rotation matrices, calculated using data from the gesture database, are imposed as end-effector condition for every present block in the corresponding chains (see Sect. 3.1).

5 Affective functional behaviors

5.1 Expressivity models

In some situations, it is desirable to express an emotional condition in a different manner than by using explicit bodily expressions as calculated by the block mode. It is possible, for example, that both arms are involved in a functional behavior, and therefore not available for performing an emotional expression. On the other hand, the recognizability of an emotional expression can decrease severely when one arm is used for a deictic gesture. In such cases it can be useful to express an emotional state through an ongoing functional behavior by modulating it, using a certain set of characteristic performance parameters. In literature, different expressivity models have been developed to reach that goal. Amaya et al. (1996) proposed a model to generate an emotional animation from neutral motions by calculating an emotional transform based on the difference in speed and spatial amplitude of a neutral and emotional motion. In Pelachaud (2009), six parameters, namely spatial extent, temporal extent, fluidity, power, overall activation and repetition were used to modify behavior animations for the virtual agent Greta. Yamaguchi et al. (2006) found that the amplitude, position and speed are relevant parameters in modifying basic motions to express joy, sadness, anger and fear, while Lin et al. (2009) found that the stiffness, speed and spatial extent of the motion, can effectively generate emotional animations

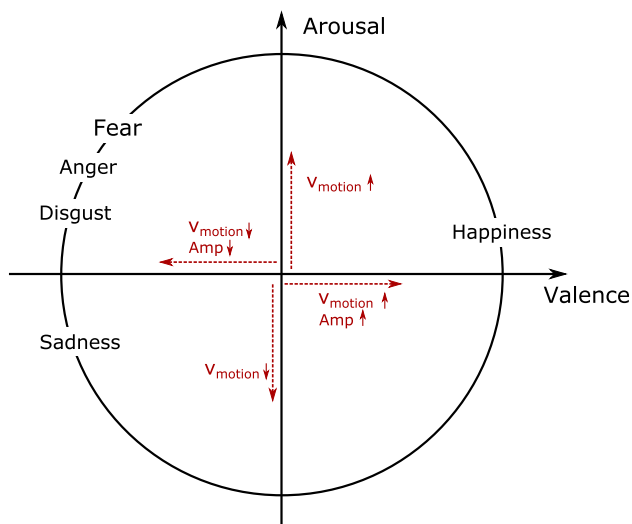


Fig. 6 Dependency of the modification factors *motion speed* (v_{motion}) and *Amplitude* (*Amp*) on the valence and arousal value, depicted on the circumplex model of affect (Posner et al. (2005))

from an initial neutral motion. Xu et al. (2013a) proposed a method for bodily mood expression, whereby a set of pose and motion parameters modulate the appearance of an ongoing functional behavior. Results indicated that the spatial extent parameters, including hand-height and amplitude, head position and the motion speed are the most important parameters for readable mood expressions (Xu et al. 2013b). Since in all these discussed expressivity models, the motion speed and the amplitude are important recurring factors, we decided to focus on these modification parameters in our method.

5.2 Generating affective gestures by influencing the motion speed

In both (Xu et al. 2013b) and Yamaguchi et al. (2006), it was experimentally confirmed that the motion speed influences the perceived level of both valence and arousal; a fast motion is associated with a high arousal and valence, while a slow motion is attributed to low arousal and valence values. By considering the two dimensional emotion space of valence and arousal, based on the circumplex model of affect (Posner et al. 2005), we obtained an appropriate speed scaling factor for each emotion (see Fig. 6). When calculating a deictic gesture with the end-effector mode of our method, a suitable trajectory between the initial posture and the end posture is generated by calculating intermediate key frames. The timing between two consecutive frames is fixed, but the amount of frames, and therefore the total duration of the gesture is determined by the speed scaling factor, in order to add affectional content to it.

5.3 Generating affective postures using the nullspace

The second modification parameter, the amplitude of the motion, refers to the spatial extent; the amount of space occupied by the body. (Xu et al. 2013b) found that this parameter is only related to the valence; open postures with a high amplitude are coupled with affective states with high valence, while closed, low amplitude postures are related to states with a low valence (see Fig. 6). As discussed in Sect. 3, the necessary joint angles to reach a desired posture are calculated by the inverse kinematics algorithm of Eq. 2 with as cost function w , a slightly adapted form of the joint range availability criterion (see Eq. 4). In that way, an optimal humanlike posture is calculated by keeping the joints q close to a selected set of minimum posture angles q_{mi} . Instead of using the fixed minimum posture angles, it is possible to express them as a function of the current valence level. Hence, the resulting calculated posture becomes dependent of the current affective state. The Body Action Units mostly influencing the openness of a posture are BAU 10 and 13; the units corresponding to the abduction/adduction of the shoulder and the flexion/extension of the elbow joint (see Table 1). For the joints corresponding to these BAU's, a linear function of the valence is provided instead of the fixed minimum posture angle as used before. When scaling the valence level val for each emotion as read on the circumplex model of affect (see Fig. 6) between 0 and 1, the following linear function can be used to select the current appropriate value for the minimum posture angle, which we now call the *affective posture angle* q_{ai} :

$$q_{ai} = q_{ai,min} + val \times (q_{ai,max} - q_{ai,min}) \quad (5)$$

The minimum value $q_{ai,min}$ of the affective posture angle corresponds to the value associated to the minimum valence value, i.e. a value generating a closed posture with low amplitude. The angle value is defined in the corresponding reference frame connected to the human base model, and relatively to the T-pose as visualized in Fig. 2. Therefore, for BAU 10, a value of 90° is a suitable choice, since it corresponds to a posture whereby the upper arm is touching the flank of the body. Regarding BAU 13, a small amplitude posture is reached when keeping the forearm as close as possible to the upper arm. A value of 170° is therefore an appropriate choice. Similarly, the maximum value $q_{ai,max}$ of the affective posture angle corresponds to the value associated to the maximum valence value; the value generating an open posture with high amplitude. This should be a posture whereby both the elbow and wrist are located far away from the body. A suitable choice is therefore 0° for BAU 10, and 80° for BAU 13.

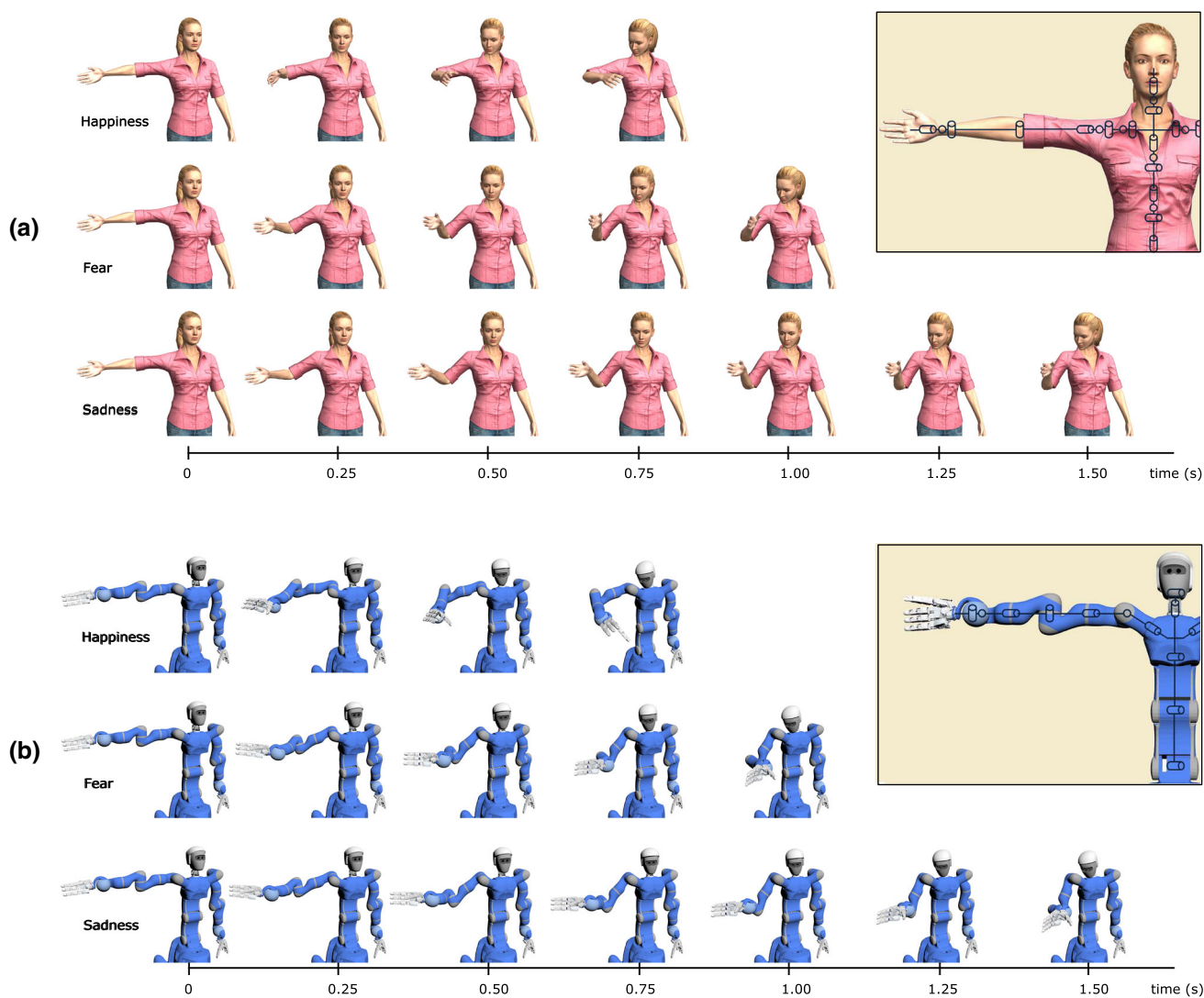


Fig. 7 Example illustrating expressing affect during a functional behavior for **a** a human virtual model **b** the robot Justin. A reaching gesture was calculated during different affective states: happiness, fear and sadness. *Right top* corresponding joint configuration. *Main figure*

time line showing a set of postures for every affective state, illustrating the effect of the motion speed modification factor on the calculated gesture. The effect of the amplitude modification factor is visible when comparing the end postures for every mood

5.4 Example: deictic gesture during different states of affect

Figure 7 illustrates the results of the two subsections discussed above. A right-arm reaching gesture during different states of affect was calculated for two different configurations; a human virtual model with a 9 DOF arm (Fig. 7a), and the robot Justin (Fig. 7b). This example shows that the developed method cannot only be used for existing robots, but also for any virtual model by assigning suitable joint configuration to it.

The right top of Fig. 7 shows the corresponding joint configuration of the used models, while the main figure visualizes a set of calculated postures for every affective state on a time

line. As discussed in Sect. 5.2, the total timing of the gesture is influenced by the speed factor, of which the current value is determined by the current affective state. Since the motion speed increases with both valence and arousal, a high value is obtained for the happy state, resulting in a short total timing of the gesture of 0.75 s. For the same pointing gesture performed during a sad state, a low speed factor and long duration (1.5 s) is calculated, while for the fearful state, the values are located somewhere in between (duration of 1.0 s).

The influence of the amplitude modification factor is visible when comparing the end postures for each affective state. Since the amplitude of the posture increases with higher valence values, an open posture is calculated for the happy

state, whereby the elbow is located far away from the body. For the sad state, the elbow is placed close to the body, generating a closed posture as expected. Since the valence values for fear and sadness are close to each other (see Fig. 6), the difference in posture during the the corresponding states is small (a difference of approximate 10° for BAU 10), but here the total timing of the gesture is the main differentiator.

6 Conclusions and current work

In this paper, we presented the new developments of our generic method to generate gestures for social robots. The method was designed to work in two modes, to allow the calculation of different types of gestures. The *block mode* is used to calculate gestures whereby the overall arm placement is crucial, like for emotional expressions, while the *end effector mode* was developed for end-effector depending gestures, such as deictic gestures. The working principles of both modes were discussed in previous publications (Van de Perre et al. 2015, 2016). During human communication, different types of gestures are used and combined. In this paper we discussed how the two modes can be combined to generate blended emotional expressions and deictic gestures. To achieve this, a mode mixer was developed, and for every mode, priority levels were assigned to each chain. The priority levels decide which end-effector constraints need to be considered for each chain. In that way, when gestures with different priority levels are selected with the mode mixer enabled, the imposed end-effector conditions originating from the different gestures result in a blended posture. A combination of a pointing gesture with the emotional expression of fear was calculated for both the robots NAO and Justin to illustrate this new functionality. When one arm is used for a deictic gesture, it is possible that the recognizability of the emotional expression decreases. In that case, it can be interesting to express an emotional condition not by using explicit bodily expressions as calculated by the block mode, but through an ongoing functional behavior. We implemented the possibility to modulate a pointing or reaching gesture into an affective gesture by influencing the motion speed and amplitude of the posture. To illustrate the results of this new implementation, an affective gesture was calculated for two different configurations for three affective states. Differences in motion speed and posture could be clearly distinguished. However, for configurations with very low DOF's, these differences can diminish. In that case, it can be interesting to implement supplementary modification parameters.

The new implementations were validated on the virtual model of different robots. Current work includes evaluating the method on the physical model of a set of robots, including Romeo and Pepper.

Acknowledgements The first author is funded by the Fund for Scientific Research (FWO) Flanders. This work is partially funded by the EU-Project DREAM (611391). Robotics and Multibody Mechanics Research Group is partner of the Agile and Human Centered Production and Robotic Systems Research Priority of Flanders Make. The authors would like to thank DLR for sharing the virtual model of Justin.

References

- Alissandrakis, A., Nehaniv, C. L., & Dautenhahn, K. (2002). Imitation with alice: Learning to imitate corresponding actions across dissimilar embodiments. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, 32(4), 482–496.
- Amaya, K., Bruderlin, A., & Calvert, T. (1996). Emotion from motion. *Graphics Interface*, 96, 222–229.
- Andry, P., Gaussier, P., Moga, S., Banquet, J., & Nadel, J. (2001). Learning and communication via imitation: An autonomous robot perspective. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 31, 431–442.
- Ascher, U. M., & Petzold, L. R. (1998). *Computer methods for ordinary differential equations and differential-algebraic equations*. Philadelphia: Siam.
- Atkinson, A. P., Dittrich, W. H., Gemmell, A. J., Young, A. W., et al. (2004). Emotion perception from dynamic and static body expressions in point-light and full-light displays. *Perception*, 33, 717–746.
- Azad, P., Asfour, T., & Dillmann, R. (2007). Toward a unified representation for imitation of human motion on humanoids. In: *IEEE International Conference on Robotics and Automation*, (pp. 2558–2563), IEEE.
- Balit, E., Vaufreydaz, D., & Reigner, P. (2016). Integrating animation artists into the animation design of social robots. In: *ACM/IEEE Human-Robot Interaction 2016* (pp 417–418)
- Belpaeme, T., Baxter, P. E., Read, R., Wood, R., Cuayáhuil, H., Kiefer, B., et al. (2012). Multimodal child-robot interaction: Building social bonds. *Journal of Human-Robot Interaction*, 1(2), 33–53.
- Breazeal, C., Kidd, C. D., Thomaz, A. L., Hoffman, G., & Berlin, M. (2005). Effects of nonverbal communication on efficiency and robustness in human-robot teamwork. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2005)*, (pp 708–713).
- Castellano, G., Villalba, S. D., & Camurri, A. (2007). Recognising human emotions from body movement and gesture dynamics. In: A.C.R. Paiva, R. Prada, R.W. Picard (Eds.), *Affective computing and intelligent interaction* (pp .71–82), Springer.
- Coulson, M. (2004). Attributing emotion to static body postures: Recognition accuracy, confusions, and viewpoint dependence. *Journal of Nonverbal Behavior*, 28(2), 117–139.
- Crane, E., & Gross, M. (2007). Motion capture and emotion: Affect detection in whole body movement. In: A.C.R. Paiva, R. Prada, R.W. Picard (Eds.), *Affective computing and intelligent interaction* (pp. 95–101), Springer.
- Dautenhahn, K., & Nehaniv, C. L. (2002). *The correspondence problem*. Cambridge: MIT Press.
- De Meijer, M. (1989). The contribution of general features of body movement to the attribution of emotions. *Journal of Nonverbal Behavior*, 13(4), 247–268.
- Dittrich, W. H., Troscianko, T., Lea, S. E., & Morgan, D. (1996). Perception of emotion from dynamic point-light displays represented in dance. *Perception*, 25(6), 727–738.
- Do, M., Azad, P., Asfour, T., & Dillmann, R. (2008). Imitation of human motion on a humanoid robot using non-linear optimization. *8th IEEE-RAS International Conference on Humanoid Robots, Humanoid* (pp. 545–552).

- Gienger M., Janssen H., & Goerick C. (2005). Task-oriented whole body motion for humanoid robots. In: *2005 5th IEEE-RAS International Conference on Humanoid Robots* (pp. 238–244), IEEE.
- Hild, M., Siedel, T., Benckendorff, C., Thiele, C., & Spranger, M. (2012). Myon, a new humanoid. In: L. Steels, M. Hild (Eds.), *Language grounding in robots* (pp. 25–44), Springer.
- Hirukawaa, H., Kanehiroa, F., Kanekoa, K., Kajitaa, S., Fujiwaraa, K., Kawaia, Y., et al. (2004). Humanoid robotics platforms developed in HRP. *Robotics and Autonomous Systems*, 48(4), 165–175.
- Ido, J., Matsumoto, Y., Ogasawara, T., & Nisimura, R. (2006). Humanoid with interaction ability using vision and speech information. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2006)* (pp 1316–1321).
- Itoh, K., Miwa, H., Matsumoto, M., Zecca, M., Takanobu, H., Roccella S., et al. (2004). Various emotional expressions with emotion expression humanoid robot WE-4RII. In: *IEEE Technical Exhibition Based Conference on Robotics and Automation* (pp. 35–36).
- James, W. T. (1932). A study of the expression of bodily posture. *The Journal of General Psychology*, 7(2), 405–437.
- Jung, E. S., Kee, D., & Chung, M. K. (1995). Upper body reach posture prediction for ergonomic evaluation models. *International Journal of Industrial Ergonomics*, 16(2), 95–107.
- Kadaba, M. P., Ramakrishnan, H., & Wootten, M. (1990). Measurement of lower extremity kinematics during level walking. *Journal of Orthopaedic Research*, 8(3), 383–392.
- Koga, Y., Kondo, K., Kuffner, J., & Latombe, J. C. (1994). Planning motions with intentions. In: *Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques* (pp. 395–408), ACM.
- Le, Q. A., Hanoune, S., & Pelachaud, C. (2011). Design and implementation of an expressive gesture model for a humanoid robot. In: *11th IEEE-RAS International Conference on Humanoid Robots* (pp. 134–140), IEEE.
- Lin, Y. H., Liu, C. Y., Lee, H. W., Huang, S. L., & Li, T. Y. (2009). Evaluating emotive character animations created with procedural animation. In: Z. Ruttkay, M. Kipp, A. Nijholt, H. H. Vilhjálmsón (Eds.), *Intelligent virtual agents* (pp. 308–315), Springer.
- Matsui, D., Minato, T., MacDorman, K., & Ishiguro, H. (2005). Generating natural motion in an android by mapping human motion. In: *IROS* (pp. 3301–3308).
- Montepare, J. M., Goldstein, S. B., & Clausen, A. (1987). The identification of emotions from gait information. *Journal of Nonverbal Behavior*, 11(1), 33–42.
- Mühlig, M., Gienger, M., & Steil, J. J. (2012). Interactive imitation learning of object movement skills. *Autonomous Robots*, 32(2), 97–114.
- Park, E., Kim, K. J., & del Pobil, A. P. (2011). The effects of robot body gesture and gender in human–robot interaction. *Human-Computer Interaction*, 6, 91–96.
- Pelachaud, C. (2009). Studies on gesture expressivity for a virtual agent. *Speech Communication*, 51(7), 630–639.
- Pollick, F. E., Paterson, H. M., Bruderlin, A., & Sanford, A. J. (2001). Perceiving affect from arm movement. *Cognition*, 82(2), B51–B61.
- Posner, J., Russell, J., & Peterson, B. (2005). The circumplex model of affect: An integrative approach to affective neuroscience, cognitive development, and psychopathology. *Development and Psychopathology*, 17, 715–734.
- Salem, M., Kopp, S., Wachsmuth, I., & Joublin, F. (2009). Towards meaningful robot gesture. In: H. Ritter, G. Sagerer, R. Dillmann, M. Buss (Eds.), *Human centered robot systems* (pp. 173–182), Springer.
- Salem, M., Kopp, S., Wachsmuth, I., & Joublin, F. (2010). Generating multi-modal robot behavior based on a virtual agent framework. In: *Proceedings of the ICRA 2010 Workshop on Interactive Communication for Autonomous Intelligent Robots (ICAIR)*.
- Salem, M., Eyssele, F., Rohlfing, K., Kopp, S., & Joublin, F. (2013). To err is human (-like): Effects of robot gesture on perceived anthropomorphism and likability. *International Journal of Social Robotics*, 5(3), 313–323.
- Scheutz, M., Schermerhorn, P., Kramer, J., & Anderson, D. (2007). First steps toward natural human-like hri. *Autonomous Robots*, 22(4), 411–423.
- Sciavicco, L. (2009). *Robotics: Modelling, planning and control*. Berlin: Springer.
- Soechting, J. F., & Flanders, M. (1989). Errors in pointing are due to approximations in sensorimotor transformations. *Journal of Neurophysiology*, 62(2), 595–608.
- Stanton, C., Bogdanovych, A., Ratanasena, E. (2012). Teleoperation of a humanoid robot using full-body motion capture, example movements, and machine learning. In: *Proceedings of Australasian Conference on Robotics and Automation*.
- Sugiyama, O., Kanda, T., Imai, M., Ishiguro, H., & Hagita, N. (2007). Natural deictic communication with humanoid robots. *IROS, 2007*, 1441–1448.
- Tapus, A., Peca, A., Aly, A., Pop, C., Jisa, L., Pintea, S., et al. (2012). Children with autism social engagement in interaction with nao, an imitative robot. a series of single case experiments. *Interaction Studies*, 13(3), 315–347.
- Terlemez, O., Ulbrich, S., Mandery, C., Do, M., Vahrenkamp, N., & Asfour, T. (2014). Master Motor Map (MMM)—Framework and toolkit for capturing, representing, and reproducing human motion on humanoid robots. In: *14th IEEE-RAS International Conference on Humanoid Robots (Humanoids)* (pp. 894–901), IEEE.
- Van de Perre, G., De Beir, A., Cao, H. L., Esteban, P. G., Lefeber, D., & Vanderborght, B. (2016). Reaching an pointing gestures calculated by a generic gesture system for social robots. *Robotics and Autonomous Systems*, 83, 32–43.
- Van de Perre, G., Van Damme, M., Lefeber, D., & Vanderborght, B. (2015). Development of a generic method to generate upper-body emotional expressions for different social robots. *Advanced Robotics*, 29(9), 59–609.
- Wallbott, H. G. (1998). Bodily expression of emotion. *European Journal of Social Psychology*, 28(6), 879–896.
- Xu, J., Broekens, J., Hindriks, K., & Neerincx, M. A. (2013a). Mood expression through parameterized functional behavior of robots. In: *IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN 2013)* (pp. 533–540).
- Xu, J., Broekens, J., Hindriks, K., & Neerincx, M. A. (2013b). The relative importance and interrelations between behavior parameters for robots' mood expression. In: *2013 Humaine Association Conference on Affective Computing and Intelligent Interaction (ACII)* (pp. 558–563), IEEE.
- Yamaguchi, A., Yano, Y., Doki, S., & Okuma, S. (2006). A study of emotional motion description by motion modification and adjectival expressions. *IEEE Conference on Cybernetics and Intelligent Systems, 2006*, 1–6.
- Zecca, M., Mizoguchi, Y., Endo, K., Iida, F., Kawabata, Y., Endo, N., et al. (2009). Whole body emotion expressions for KOBAN humanoid robot: Preliminary experiments with different emotional patterns. In: *The 18th IEEE International Symposium on Robot and Human Interactive Communication. RO-MAN 2009* (pp. 381–386).



Greet Van de Perre studied Mechanical Engineering at the Vrije Universiteit Brussel. After graduating in 2011, she started her Ph.D. at the Robotics and Multibody Mechanics Research Group, funded by an FWO fellowship. Her research focusses on the development of a generic gesture system for social robots and the design of an actuated upper-torso for the social robot Probo.



Hoang-Long Cao received his B.Eng. in Mechatronics from Can Tho University (Vietnam) in 2010, and M.Sc. in Electromechanical engineering from the Vrije Universiteit Brussel (Belgium) and Universite Libre de Bruxelles (Belgium) in 2013. From September 2013, he has been working as a PhD researcher at the Multibody Mechanics and Robotics research group (VUB). His research interest is social robotics. Currently he is involved in

DREAM project for the development of robot-enhanced therapy for children with autism spectrum disorders.



Albert De Beir received in 2014 his Master degree in electromechanical engineering from the Vrije Universiteit Brussel as well as his Bachelor degree in Psychology from the Universite Libre de Bruxelles. He is now pursuing a Master in neuropsychology. His research interest focus on the development of artificial intelligence for social robots.



Pablo Gómez Esteban received, in March 2014 from the Rey Juan Carlos University of Madrid, a Ph.D. degree for his thesis entitled “Cooperation and Competition in Emotional Robot Societies” with highest distinction. He is currently a post-doctoral researcher working at Vrije Universiteit Brussel in Belgium. His research was addressed towards decision making processes within competitive and cooperative environments; the impact that affective factors have

in such processes; and designing and implementing those models to be used in non-expensive platforms within domestic environments. But his research interests also lie on cognitive architectures for social robots and human-robot interaction. He is the author or co-author of several chapters in books and published articles in international journals and conferences. He has also been a reviewer for journals and conferences including *Frontiers in Psychology*, *Autonomous Robots* and *ICSR 2015*.



Dirk Lefeber was born in 1956. He received a degree in Civil Engineering in 1979 and a Ph.D. in Applied Sciences in 1986 from the Vrije Universiteit Brussel. Currently he is full professor, former head of the department of Mechanical Engineering and head of the Robotics and Multibody Mechanics Research Group, Vrije Universiteit Brussel, which he founded in 1990. At that time the research focused on dynamical simulations of walking, hopping and running of

mono-, bi-and quadruped robots. From the beginning, the group had a strong interest in adaptable compliant actuators. The group invented 2 different adaptable compliant actuators: the pleated pneumatic artificial muscle and the Maccopa actuator. With these they have built the walking robots Lucy and Veronica. The past 7 years the main research focus moved to physical and cognitive human-robot interaction. The group developed the AMP-foot, an actuated transtibial prosthesis and the Hekta, a passive transfemoral prosthesis. The ALTACRO-rehabilitation robot is a 12 active-DOF lower-limb rehabilitation exoskeleton. Probo is a social robot with 19 DOF in the head, allowing for emotional interaction with people. Currently the R&MM group is partner in 5 FP-7 projects. The research interests of the group are new actuators with adaptable compliance, dynamically balanced robots, robot assistants, rehabilitation robotics and multibody dynamics, all in the context of physical and cognitive human robot interaction with emphasis on improving the quality of life of human.



Bram Vanderborght focused his Ph.D. research on the use of adaptable compliance of pneumatic artificial muscles in the dynamically balanced biped Lucy. In May-June 2006 he performed research on the humanoid robot HRP-2 at the Joint Japanese/French Robotics Laboratory (JRL) in AIST, Tsukuba (Japan) in the research “Dynamically stepping over large obstacles by the humanoid robot HRP-2”. From October 2007–April 2010 he worked as post-doc researcher

at the Italian Institute of Technology in Genova (Italy) on the humanoid robot iCub and compliant actuation. Since October 2009 he is appointed as professor at the VUB. He has an ERC starting grant on SPEA actuation concept. He is member of the Young Academy of the Royal Flemish Academy of Belgium for Science and the Arts. His research interests include cognitive and physical human robot interaction, social robots, humanoids, robot assisted therapy and rehabilitation/assistive robotics.