Imitation of Basic Hand Preshapes by Fluid Based Method: Fluidic Formation Control

Umut Tilki[1], Ismet Erkmen[1], and Aydan M. Erkmen[1]

[1]Middle East Technical University, Faculty of Engineering, Department of Electrical and Electronics Engineering, 06531, Çankaya, Ankara, Turkey
utilki@metu.edu.tr, erkmen@metu.edu.tr, aydan@metu.edu.tr

Abstract

In this paper, a new approach is developed handling the correspondence problem due to the difference in embodiment between imitator and demonstrator in imitation learning. In our work the imitator is a fluidic system of dynamics totally different than the demonstrator which is human hand gestures. In this work we demonstrate the fluidic formation control so as to generate basic hand preshaping features. Our fluidic formation control is based on Smoothed Particle Hydrodynamics (SPH), which is a particle based lagrangian method. The controller adjusts fluid parameters such as body force (f), density and velocity of particles (V) to generate particle formation mimicking hand preshapes.

1. Introduction

Learning to act by observing others that may not be kinematically the same is the recent focus of machines imitating human or animals, or machines imitating other machines [1]. Many problems and difficulties have been at the motivation of imitation learning. The primary difficulty is due to the correspondence problem, or in other words, the mapping of actions or action sequences between a demonstrator (teacher) and an imitator (student) because of embodiment (kinematic structure) differences. For imitation between two human beings, although same in dynamics a disease may hamper the correspondence problem and the organ maybe faulty: the demonstrator lifting a hand, the imitator may lift a leg. In imitation between two systems different in kinematics, this disease is inherent. For example imitation of human hand motions by a 3 – fingered robot hand has an inherent correspondence problem, will no one to one organ matching, the imitator being under actuated with less limbs than that of the demonstrator. Despite the correspondence problem, many researchers speculated that imitative learning has valuable characteristics such as speeding up the learning process, not requiring the communication between teacher and student. [2]

In this work, we focus on imitation of a human preshaping by a particle colony of a totally different dynamics. More specifically this imitation is the colony formation control such as to resemble basic hand preshapes. The colony of particles that has a strikingly different dynamics so as to provoke an inherent correspondence problem, since the colony has no human like organs, is a mass of fluid composed of fluid particles. We achieve formation control of this colony of fluid particles using the fluid parameters of fluid dynamics equations and Smoothed Particle Hydrodynamics (SPH) which is a mesh free computational method not only used for simulating fluid flows but also in astrophysics, ballistics and in tsunami studies. It is a particle based Lagrangian method, and the resolution of the method can easily be adjusted with respect to variables of the fluid such as density. The most important advantage is the adaptive mesh free nature of the SPH method. Because of this advantage, the formulation of SPH is not affected by the arbitrariness of the particle distribution. Therefore, it can handle the problem concerned with large deformation. We have found that SPH allows the fluidic imitator, imitating human hand posture, to have a certain degree of freedom in the interpretation of the observed movement and thus to naturally generalize over varying movements by changing the boundary conditions of the particles.

2. Related Work and Motivation

Imitation learning requires a complex set of mechanisms that detect what to learn from a teacher by observing his/her movements and map them onto its own movements by transforming the teacher limbs into its own dynamical features. Such a process includes movement recognition, pose estimation and tracking, body correspondence, coordinate transformation, matching of observed movements, etc [3]. Alissandrakis et al. [4] introduced ALICE the “Action Learning for Imitation via Correspondence between Embodiments”, a generic imitation framework that can be used by an imitator agent to find corresponding actions that produces similar states and effects as a model agent. In another work, laser scanner data are used to determine rough human body positions, body orientation and pose information. [5] For tracking and velocity control, particle filter approach is used in this research. Minato et al. [6] use a grid-based approach in their approach on an android by mapping human posture in three-dimensional position space. They attempt to naturally animate a robot to maintain social interaction. For posture transformation from human subject to android they use a motion capture system which can measure the posture of human subject and the android by attaching markers on the android so that all joint motions can be discriminated. Then the same markers are attached to the subject’s body. In this kind of experimental setups, the major constraint is that there must be a large set of sensory data, collected from data gloves, magnetic trackers, stereo vision systems etc. to track and understand the movements of the demonstrator.

Our main objective in our work is to deal with imitation by observation between two dynamically different systems that carries a total mismatch of organ correspondence (inherent idemotor apraxia disease in the imitator system) so that imitation requires the understanding of what to imitate, when and how to imitate without the case of pattern matching for organ correspondence. Our imitator is a fluid body imitating through sensing, the human hand preshapes. We provide in this paper, the "proof of concept" demonstration that through our proposed fluidic formation control can make the fluidic body assume postures that mimic basic hand preshapes. Our approach is based on generating the control of flow field variables in order to get desired behaviors and shapes of the fluid body.

Fluid based modeling has been used in robotics, generally for swarms and recently in the formation of geometric patterns with multiple robots [7], [8]. In these works, mobile robots are modeled as fluid particles and are controlled by the help of fluid dynamics parameters. There are various characteristics of fluid flows which are desirable in swarm robotics, such as obstacle avoidance and source to sink optimal path finding behaviors of the fluid. In this work, we combine these behaviors to get the desired shape of the colony of particles. In [7] the swarm group was controlled by the help of the SPH parameters to avoid obstacles. In another work [8], again mobile robots were modeled as fluid particles and their motions were generated using some geometric patterns. There, besides the SPH theorem, Finite Element Method (FEM) and electrostatic fields are also used to derive the fluid particles towards the geometric pattern and also for treating the environmental obstacles in a proper manner.

3. Fluidic Formation Control

In SPH method the problem domain is represented by a set of arbitrary distributed particles and no specific discretization connectivity for these particles is needed (mesh-free). In our work these particles are controlled by the help of fluid parameters to get the desired shapes looking like human hand preshapes. In our colony each particle is affected by a finite set of neighboring particles forming the "support domain" of that particle (Fig. 1). All calculation of the field variables depend on these neighboring particles. Since SPH is an approximation method, integrals are approximated based on field functions representation method is used for field function approximation. This is also known as kernel approximation in the SPH method. The kernel approximation is then discretized based on particle units (particle approximation). It is done by replacing the integration in the integral representation of the field function with summations over all the corresponding values at the neighboring particles the support domain. The particle discretization generate discrete iterations at every time step, and hence the use of the particles depends on the current local distribution of the particles. The particle approximations are performed to all terms related to field functions in the partial differential equations to produce a set of ordinary differential equations in discretized form with respect to time. [9] By combining these items, we can say that SPH is a mesh-free, adaptive, lagrangian, and particle numerical method.

3.1. Our Problem Characteristics

There are two types of fluid flow: One is the incompressible fluid flow and the other is the compressible fluid flow. Since human hand preshaping is generated from directional movements, in our problem our fluid particles have to be modeled as incompressible fluid flow. We control these incompressible fluid particles by changing fluid parameters to get the desired hand preshapes. For example we deduce that branching control for the particle colony can mimic separation of fingers. To generate the hand gesture meaning "perfect" (Fig. 8) the controller first separate the colony into two branches and then attracts the tip of the branches on a point and this shape looks like finger tips, approaching each other like in a finger tip grasping.

3.2. Basic SPH Concept and Formulation

Our SPH formulation is adapted from [9]. It solves the momentum equation to determine the particle acceleration based on values of parameters, such as density, pressure, viscosity, which are obtained from neighbor particles in the support domain.

The integral representation of a function f(x) in the SPH method;

$$f(x) = \int_{\Omega} f(x')\delta(x-x')dx' \tag{1}$$

where Ω is the volume of integral that contains three-dimensional position vector, f is a field function like viscosity or pressure, and $\delta(x-x')$ is the Dirac delta function. For the approximation of the integral representation in equation (1), Dirac delta function is replaced by a smoothing (kernel) function W.

$$< f(x_i) >= \int_{\Omega} f(x')W(x-x',h)dx' \tag{2}$$

where the angle brackets, < >, indicate the approximation and W is the smoothing function which is not only determines the pattern for the function approximation but also defines the dimensions of the support domain of particles. This smoothing function is used in the calculation of fluid variables approximation. For smoothing function, there are various choices such as Gaussian kernel, the cubic spline kernel, the quadratic smoothing function etc. Due to its smoothness, stability, and accuracy we choose the Gaussian kernel in our simulations (3).

$$W(R_{ij}) = \begin{cases} \alpha_d e^{-R_{ij}^2} & R_{ij} \le Kh \\ 0 & otherwise \end{cases} \tag{3}$$

where $\alpha_d = \dfrac{1}{\pi h^2}$ for two-dimensional space, K = 2, $R_{ij} = | x_i - x_j | / h$.

For the particle approximation, the continuous integral representation of the kernel approximation is converted to a discretized form of summation over all the particles in the support domain shown in Figure 1.

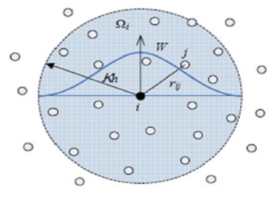

Fig. 1. The support domain for particle i and the 1D projection of a smoothing function over it

The particle approximation of the function is obtained as;

$$< f(x_i) >= \sum_{j\in\Omega_i} \frac{m_j}{\rho_j} f(x_j) W(x_i - x_j, h) \quad (3)$$

where m_j and ρ_j are the mass and density of particle j. Eq.(3) states that the value of a function at particle i is approximated using the average of those values of the function at all particles in the support domain of the particle i weighted by the smoothing function.

Since density approximation (4) determines the particle distribution and the smoothing length evolution it is really important in the SPH method and simply states that the density of a particle can be approximated by the weighted average of the densities of the particles in the support domain of that particle.

$$\rho_i = \sum_{j=1}^{N} m_j W_{ij} \quad (4)$$

where ρ is density, N is the number of particles which are in the support domain of particle i, m is the mass of particle j and W_{ij} is the smoothing function of particle i evaluated at particle j.

$$W_{ij} = W(x_i - x_j, h) = W(| x_i - x_j |, h) = W(R_{ij}, h) \quad (5)$$

where R_{ij} is the relative distance between particle i and j.

Although density equation is one of the important equations for fluid flow, we get the velocity values for the particles from solution of the momentum equation (6). The momentum equation calculates the time rate of change of velocity using substantial derivative D / Dt .

$$\frac{Du_i}{Dt} = -\sum_{j=1}^{N} m_j \left(\frac{p_i + p_j}{\rho_i \rho_j} + \Pi_{ij}\right) \frac{\partial W_{ij}}{\partial x_i}$$

$$+\sum_{j=1}^{N} m_j \left(\frac{\mu_i \varepsilon_i^{xx} + \mu_j \varepsilon_j^{xx}}{\rho_i \rho_j} \frac{\partial W_{ij}}{\partial x_i} + \frac{\mu_i \varepsilon_i^{xy} + \mu_j \varepsilon_j^{xy}}{\rho_i \rho_j} \frac{\partial W_{ij}}{\partial y_i}\right) + f_i^x$$

(6)

$$\frac{Dv_i}{Dt} = -\sum_{j=1}^{N} m_j \left(\frac{p_i + p_j}{\rho_i \rho_j} + \Pi_{ij}\right) \frac{\partial W_{ij}}{\partial y_i}$$

$$+\sum_{j=1}^{N} m_j \left(\frac{\mu_i \varepsilon_i^{xy} + \mu_j \varepsilon_j^{xy}}{\rho_i \rho_j} \frac{\partial W_{ij}}{\partial x_i} + \frac{\mu_i \varepsilon_i^{yy} + \mu_j \varepsilon_j^{yy}}{\rho_i \rho_j} \frac{\partial W_{ij}}{\partial y_i}\right) + f_i^y$$

where u and v velocities for the particle i along x and y directions respectively. P is the pressure, ρ is the density of the specified particle, Π_{ij} is the artificial viscosity (7), W_{ij} is the smoothing kernel, and ε stands for the stress factor.

In the momentum equation (6) there are three terms on the right hand side. The first term is the major portion of this equation due to the pressure gradient with the dissipative artificial viscosity which is mainly used in order model the shock waves in the tube in fluid flow simulations. The second term shows the viscosity and stress parameters. One of the particles starts to move, the other particles are affected because of this motion. ε^{xx} and ε^{yy} stands for the normal stress, ε^{xy} is for the shearing deformation, for generating dragging effect (8). The last term, f, is the body force. Since it directly enters the momentum equation, it has a direct effect on the flow. In fluid physics the typical body force is the gravitational force. It is suitable for the guidance of the particles. In this paper we used the body force to get the desired trajectories of the particles.

The artificial viscosity term in eq. (6) can be written as in the eq. (7).

$$\Pi_{ij} = \begin{cases} \dfrac{\beta_\Pi \phi_{ij}^2}{\bar{\rho}_{ij}} & v_{ij}.x_{ij} < 0 \\ 0 & v_{ij}.x_{ij} \geq 0 \end{cases} \quad (7)$$

where

$$\phi_{ij} = \frac{h_{ij} v_{ij}.x_{ij}}{| x_{ij} |^2 + \varphi^2} \qquad \varphi = 0.1 h_{ij} \qquad \bar{\rho}_{ij} = \frac{1}{2}(\rho_i + \rho_j)$$

$$h_{ij} = \frac{1}{2}(h_i + h_j)$$

β_Π , which is intended to suppress particle interpenetration, is a constant and typically set around 1. [9]

And shear stress rates in eq. (6) are like in eq. (8).

$$\varepsilon_i^{xx} = \frac{2}{3} \sum_{j\in\Omega_i} \frac{m_j}{\rho_j} \left(2u_{ji} \frac{\partial W_{ij}}{\partial x_i} - v_{ji} \frac{\partial W_{ij}}{\partial y_i}\right)$$

$$\varepsilon_i^{xy} = \sum_{j\in\Omega_i} \frac{m_j}{\rho_j} \left(v_{ji} \frac{\partial W_{ij}}{\partial x_i} + u_{ji} \frac{\partial W_{ij}}{\partial y_i}\right) \quad (8)$$

$$\varepsilon_i^{yy} = \frac{2}{3} \sum_{j\in\Omega_i} \frac{m_j}{\rho_j} \left(2v_{ji} \frac{\partial W_{ij}}{\partial y_i} - u_{ji} \frac{\partial W_{ij}}{\partial x_i}\right)$$

Besides these differential equations, there is a suitable state equation between pressure (p) and density (ρ) for modeling compressible and incompressible fluid flow (9) and (10).

$$p_i = \rho_i R_i T_i \quad \text{for compressible flow} \quad (9)$$

$$p_i = B_i\left(\left(\frac{\rho_i}{\rho_0}\right)^\gamma - 1\right) \quad \text{for incompressible flow} \quad (10)$$

where R is the specific gas constant, T is the temperature, B is a constant, ρ_0 is the reference density and γ is a constant around 7.

As a state equation, we used eq. (10), since incompressible flow is much more suitable for our purpose. We are mainly interested in the directions and trajectories of the particles, to get the desired shape. The particle acceleration is calculated from the momentum equation eq. (6) by using eqs. (7-10). For calculation of the particle velocity time marching method is used, since the Navier – Stokes Equations have no analytical solution.

3.3. Our Proposed Approach

Up to this point the mathematical background is given about fluid dynamics and SPH. The ultimate goal of the SPH-based algorithm is to control the colony formation by updating the position of all particles. The flow chart of our algorithm is in the Fig. 2. First, a particle i initializes its fluid dynamic parameters for a given initial value. Then particle i needs to know the fluid variables of its neighbor particles which are inside the support domain. To solve the governing equations, it collects the information of position, velocity, and density of these neighbor particles. As we stated the formulation of density, shear stress, artificial viscosity, particle i collects these information from its neighbors. To solve the momentum equation in (6), the particle i need the values of pressure (10), density (4), and viscous stress (8). After calculation of these fluid variables, we can calculate the acceleration of the particle from the momentum equation. Before updating the position of the particle, by changing the velocity values and the body force values which have the direct effect on the motions of particles according to the desired trajectories, we can control the fluid flow.

4. Simulation Results

For a particle i;

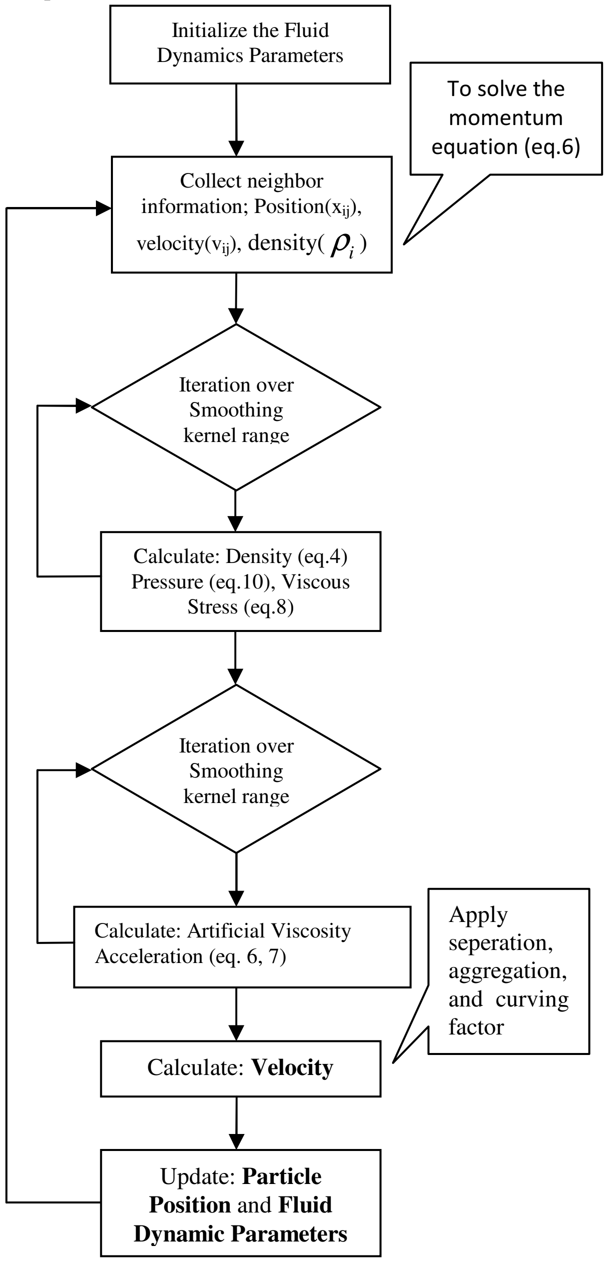

Fig. 2 Flow Chart

In our simulations, we placed the fluid particles randomly in the environment (Fig.3) and applied different body forces to demonstrate the effect of the body force on fluid flow.

At first we applied body force $f = [f_x\, f_y] = [2\ 0]$ to the particles initially shown in Fig. 3. The result of this motion can be seen in Fig.4.

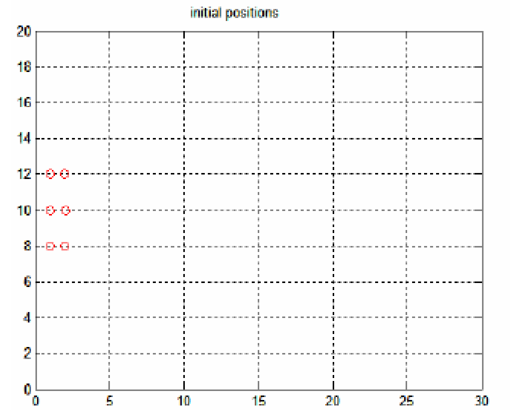

Fig.3. Initial positions of particles.

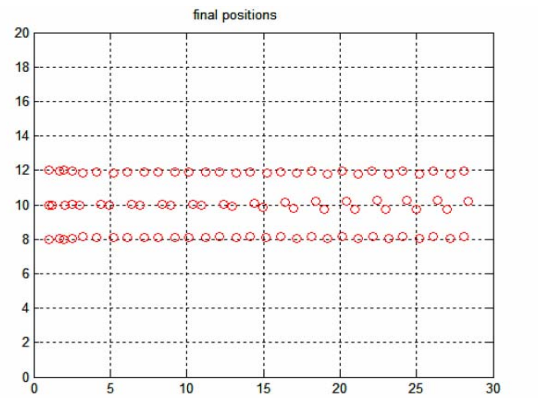

Fig.4 The effect of [2 0] body force

This result demonstrates the importance of the body force to fluid flow and the influence this parameter has in orienting and fingering effects of formation control. Our another experiment is the changing the body forces which we applied to each fluid particle during the simulation and get different fluid motions which are resemble the human hand gestures. In here we again used the same initial positions for the particles. The first one is to show the separation of the fluid particles which looks likes branching of the human fingers. (Fig.5)

At the beginning of the simulation we applied (2 0) as the body force but to get the separation we changed the body force which are in Fig. 6 and Fig. 7.

Another desired trajectory is first separating the fluid particles and then aggregating them in a point like in Fig.8. To generate this finger tip grasping like motion we again used the body forces. As one can see from the figure, after separating the colony into two branches the controller curves further the braches so that they converge to a point. This branch curving control behavior is highly desired to make the shape much more natural when compared human hand preshapes.

5. Conclusions

In this paper, we introduced fluidic formation controller to tune fluid flow parameters to get the desired colony formations which resemble human hand preshapes. Our algorithm is based on the Smoothed Particle Hydrodynamics (SPH) which is mainly used for simulation of the fluid motions.

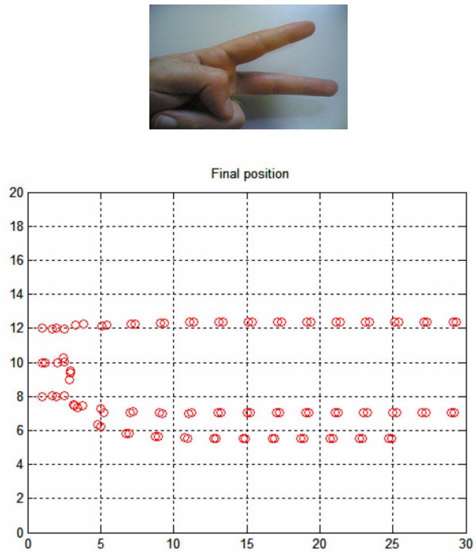

Fig.5 Separation of fluid particles

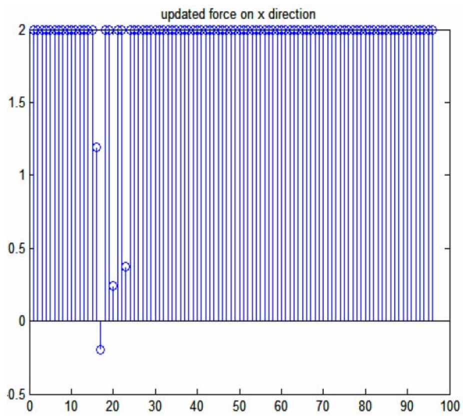

Fig.6 Body Force along x direction

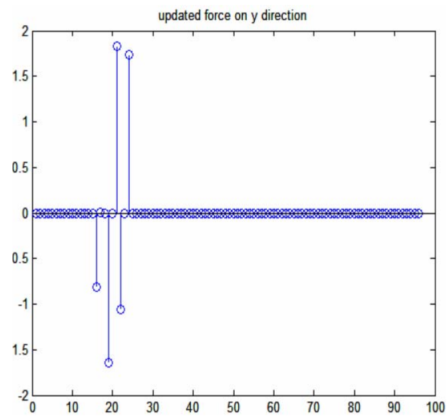

Fig.7 Body Force along y direction

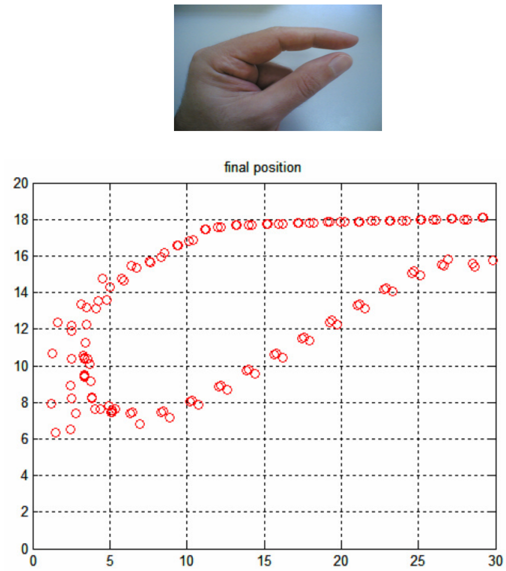

Fig. 8 Separating and aggregating the fluid particles

7. References

[1] A. Billard, "Learning motor skills by imitation: a biologically inspired robotic model", in Cybernetics & Systems, special issue on "Imitation in animals and artifacts", C. Nehaniv & K. Dautenhahn editors. Summer 2000.

[2] J. Demiris and G. Hayes, "Imitative Learning Mechanism in Robots and Humans", in Proceedings of the 5th European Workshop on Learning Robots, pp. 9-16, Bari, Italy, July 1996

[3] Schaal, S.; Ijspeert, A.; Billard, A. "Computational approaches to motor learning by imitation" in Frith, C.D.;Wolpert, D. (eds.), The Neuroscience of Social Interaction, 1431, pp. 199-218, Oxford University Press, 2004.

[4] Alissandrakis, A., Nehanic, C. and Dautenhahn, K. "Imitation with ALICE: Learning to Imitate Corresponding Actions across Dissimilar Embodiments", IEEE Transaction on Systems, Man, and Cybernetics – Part A: Systems and Humans, Vol.32, No.4, July 2002

[5] Glas, D.F., Miyashita, T., Ishiguro, H., Hagita, N. "Laser tracking of human body motion using adaptive shape modeling". IEEE/RSJ International Conference on, IROS 2007, San Diego, California, USA

[6] Minato, T., Ishiguro, H. "Generating natural posture in an android by mapping human posture in three-dimensional position space". IEEE/RSJ International Conference on, IROS 2007, San Diego, California, USA

[7] Pac, M.R., Erkmen, A.M., Erkmen, I. "Control of Robotic Swarm Behaviors Based on Smoothed Particle Hydrodynamics", IEEE/RSJ International Conference on, IROS 2007, San Diego, California, USA

[8] Pimenta, L.C.A., Mendes, M.L., Mesquita, R.C., Pereira, G.A.S. "Fluids in Electrostatic Fields: An Analogy for Multirobot Control", IEEE Transactions on Magnetics, April 2007

[9] G. R. Liu and M. B. Liu, "Smoothed Particle Hydrodynamics: A Meshfree Particle Method", World Scientific, 2003