

Human-like Visual Grasp of Unknown Objects

Vincenzo Lippiello, Fabio Ruggiero, Bruno Siciliano and Luigi Villani

Abstract—In this paper a method to achieve human-like grasps in unstructured environments is presented. The algorithm is composed of an object surface reconstruction algorithm and a local grasp planner, evolving in parallel. The former uses an elastic elliptical reconstruction surface, with axes and dimensions assigned by a preshaping process, that is let to evolve dynamically under the action of reconstruction forces. The reconstruction surface shrinks toward the object until some parts of the surface intercept the object visual hull. The latter moves the fingertips on the current available reconstruction surface towards points which are optimal (in a local sense) with respect to a certain number of indices weighting both the grasp quality and the kinematic configuration of the hand. A control module must ensure that the references given by the planner are correctly followed by the robotic hand. Experiments are presented, showing the effectiveness of the proposed algorithm.

I. INTRODUCTION

A challenging question in the robotic research field is how to perform human-like grasps in unstructured environments. In general, humans can grasp and manipulate a large variety of objects with a high level of dexterity. An elaborated taxonomy of human grasps can be found in [6].

Throughout the literature, two main approaches among the others can be recognized in the process of transferring humans manipulation skills to robotic multi-fingered hands. Namely, they are *programming by demonstration* and *neural networks and genetic algorithms* techniques.

Considering the former case, the humans movements are recorded and analyzed off-line using a motion capture system, and therefore the motion is transferred to a robotic hand [11]. Further, it has been demonstrated that humans perform different grasps in reason of the task specification, even if the orientation and location of the objects are kept the same. A programming by demonstration system which shows how fine manipulations tasks, e.g. screw moves, can be recognized, is presented in [13], where the recorded trajectory is analyzed, interpreted and mapped to a manipulator.

In the latter case, the space of all feasible grasp configurations is analyzed using genetic algorithms [5]. Since these last are not suitable for real-time applications, neural

networks are adopted. In this way, a neuro-genetic architecture is exploited in the sense that the genetic algorithms are used to create a training set for the neural network. In [7], a human-like grasp is recognized by a biologically plausible neural network. This last is built upon a hierarchical model for motion detection using a view-based recognition approach that is consistent with principles in the human cortex.

Further approaches to detect and perform human-like grasps are presented in [1], [2], where a *qualitative reasoning* approach to the synthesis of dexterous grasps is provided. An intelligent planner has been developed in order to perform this synthesis, advantageously adopting qualitative methods instead of analytical or numerical models. However, only coarse solutions can be provided, since this approach is an attempt to strike a compromise in the use of qualitative and quantitative resources.

When human-like grasps must be achieved in unstructured environments, real-time performances are necessary and no pre-recorded trajectories are available. Hence, some visual sensor has to be considered to reconstruct the object to be grasped and manipulated. In [8] kinematic parameters of the human grasp, such as path and preshape, are determined by the three dimensional geometric structure of the target object, and not by the two dimensional projected image of the same object. Moreover, human object recognition is based on identifying coarse structures rather than specific features, as underlined in [3].

In [4], it is stated that the task of autonomous manipulation can be generally divided into object detection, recognition, coarse end-effector alignment, preshaping, vision-guided grasping and the execution of the desired grasp action. Starting from this concept, in this paper a method for fast visual grasping of unknown objects using a camera mounted on a robot in an eye-in-hand configuration is presented. This method is composed of an *object surface reconstruction algorithm* and of a *local grasp planner*, which evolve in a synchronized parallel way. Differently from a classical serial approach in which the object is first completely reconstructed and then the evaluation of the optimal grasp under a selected global criterion is performed, the presented parallel approach may represent a valid alternative in cases where real-time grasping and not powerful hardware are required. In fact, although the modern technologies allow a fast object reconstruction, the investigation of all the possible combinations of the grasp points or of the set of surfaces which approximate the object (depending on the reconstruction method adopted) could generally require a considerable amount of time. With the proposed approach, the total computational time is given

The research leading to these results has been supported by the DEX-MART Large-scale integrating project, which has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement ICT-216239. The authors are solely responsible for its content. It does not represent the opinion of the European Community and the Community is not responsible for any use that might be made of the information contained therein.

The authors are with PRISMA Lab, Dipartimento di Informatica e Sistemistica, Università degli Studi di Napoli Federico II, via Claudio 21, 80125, Naples, Italy vincenzo.lippiello@unina.it, fabio.ruggiero@unina.it, siciliano@unina.it, lvillani@unina.it

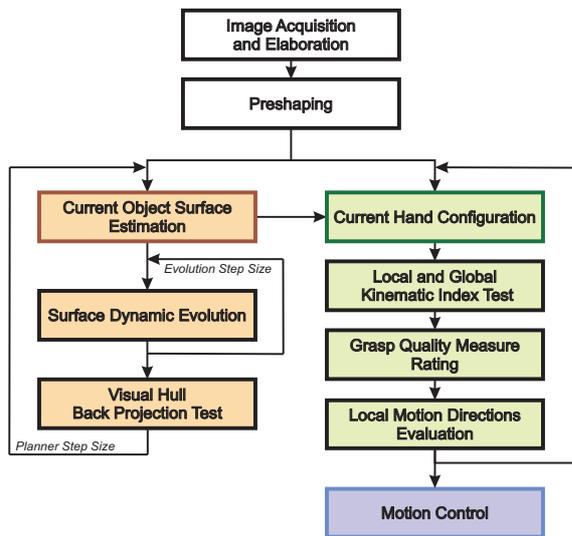


Fig. 1. Block diagram of the visual grasp algorithm.

by the slower between the reconstruction and the planning stage and these two parallel processes are independent and can be allocated under different computational resources. The main drawback of the proposed parallel approach is that the final grasp is optimal only in a local sense. Further, the local grasp planner is guided by some heuristic quality measures to achieve a human-like grasp.

Since the grasp planning is executed on-line on the base of the acquired data and of the current state of the robotic hand, a kinematic control of motion can be employed upon the dynamic of the mechanical system [12]. In fact, if the robotic system is equipped with high-gain motion controllers at a low level, the so called resolved-velocity control can be performed, in which the effects of dynamic or disturbances are neglected. In this case the system is considered as an ideal positioning device and the high controller can act at a velocity level. Since only the kinematics is exploited to derive such control law, often this approach is called *kinematic motion control* and it is well known in the robotic literature [9], [12] and used also in manipulation tasks [10].

Finally, notice that the proposed method is not dependent on the particular choice of the hand: it is a general approach and it is suitable not only for human-like robotic hands.

II. VISUAL GRASP ALGORITHM

The block diagram in Fig. 1 shows the data flow and the main elaboration steps of the proposed visual grasp algorithm. The elaboration processes may be arranged into four main groups: *image acquisition and preshaping*, *object surface reconstruction algorithm*, *local grasp planner* and *motion control*.

During the first stage a set of n images suitable for the reconstruction process is acquired. Then, the silhouette of the object for each acquired image is evaluated, and the object center of mass, assuming a homogeneous mass distribution, is estimated using a least-squares triangulation method.

The preshaping algorithm, as explained in the next section, starts to compute, from the bounding box of each silhouette, a polyhedron in the Cartesian space, which represents an overestimation of the visual hull. Then, the initial *reconstruction surface*, with elliptical shape and centered at the estimated center of mass of the object, is built on the base of the dimensions of the polyhedron. Further, the initial grasp configuration of the hand is evaluated and it depends on the initial reconstruction surface.

After this preshaping step, both the object model reconstruction process and the local planner start in parallel and cooperate to the final goal. In particular, as shown in Fig. 1, the reconstruction algorithm updates in real-time the estimation of the current reconstructed object surface, while the local planner, on the base of the current estimation, computes the fingertips trajectories toward a local optimal grasp configuration.

The controller guides the robotic multi-fingered hand imposing the joints velocity, in such way that the trajectories given by the planner are correctly tracked by the fingers.

The assumptions made throughout the paper are that an eye-in-hand configuration with a calibrated camera is available for the reconstruction stage. The observed object has to be fixed in the space during the images acquisition and distinguishable with respect to the background and other objects —from a topological point of view, the object must be an orientable surface with genus 0.

III. PRESAPING MODULE

For each of the n silhouettes and from the relative bounding box, one can build four planes in the Cartesian space, each containing the camera origin and two adjacent vertices of the corresponding silhouette's bounding box, resulting in $4n$ Cartesian planes. Obviously, each plane splits the Cartesian space into two regions, one of which contains the visual hull. The intersections of all these planes create a polyhedron which contains the object visual hull, or in other words, is a polyhedral overestimation of this last.

The vertices of this polyhedron \mathcal{P} can be quickly computed by solving a linear programming problem. Since each side of each bounding box is associated with a plane, if the normal unit vector to the plane is chosen pointing outwards with respect to the interior side of the bounding box, the inner space of this plane is represented by the following set of inequalities:

$$A_i \mathbf{x} \leq \mathbf{d}_i,$$

where the subscript i denotes the i -th image, with $i = 1, \dots, n$, A_i is a 4×3 matrix whose rows are the transpose of the normal unit vectors, and \mathbf{d}_i is a 4×1 vector whose elements define uniquely the positions of the planes in the space. Stacking all the A_i and \mathbf{d}_i in the matrices A and \mathbf{d} , the inner space of the polyhedron is represented by the following set of inequalities:

$$A \mathbf{x} \leq \mathbf{d}. \quad (1)$$

Considering (1) as a set of constraints in a minimization problem, the vertices of the corresponding polyhedron are

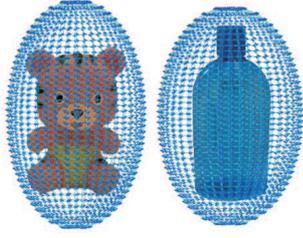


Fig. 2. Examples of preshaped elliptical reconstruction surfaces.

the so called *basic feasible solutions*, whose computation is well known in literature. Notice that, since the problem has been formulated as a linear programming problem, the computational time is very short and it depends only on the number n of images.

Once all the n_v vertices $\mathbf{x}_v = [x_{v_x} \ x_{v_y} \ x_{v_z}]^T$ of the polyhedron \mathcal{P} have been computed, the central moments can be evaluated as:

$$\mu_{i,j,k} = \sum_{\mathbf{x}_v \in \mathcal{P}} (x_{v_x} - \bar{x}_{v_x})^i (x_{v_y} - \bar{x}_{v_y})^j (x_{v_z} - \bar{x}_{v_z})^k,$$

where $\bar{\mathbf{x}}_v = [\bar{x}_{v_x} \ \bar{x}_{v_y} \ \bar{x}_{v_z}]^T = \frac{1}{n_v} \sum_{i=1}^{n_v} \mathbf{x}_{v_i}$.

Finally, a pseudo-inertia tensor of the polyhedron can be defined as:

$$I = \begin{bmatrix} \mu_{2,0,0} & \mu_{1,1,0} & \mu_{1,0,1} \\ \mu_{1,1,0} & \mu_{0,2,0} & \mu_{0,1,1} \\ \mu_{1,0,1} & \mu_{0,1,1} & \mu_{0,0,2} \end{bmatrix},$$

where its eigenvalues and eigenvectors define the principal axes of inertia of an ellipsoid, which is employed here as the initial shape of the reconstruction surface. Finally, the ellipsoid is suitably enlarged ensuring that the object is wrapped, as shown in the examples of Fig. 2.

Depending on the object shape, the ellipsoid may have one axis bigger than others, one axis smaller than others, or all axes of a similar dimension. For all these cases, a good choice for the grasp configuration depends also on the task to accomplish (e.g. pick-and-place, manipulation, assembling, etc.), on the type of grasp to perform (firm or fine), on the environmental constraints (e.g. the ground plane), and on the hand kinematics and the number of fingers. In this paper, for simplicity and considering the previous assumptions of fine manipulation, the initial grasp configuration is chosen as an equilateral grasp in a plane parallel to the two minor axes of the ellipsoid, when it is reachable with respect to the hand and environmental constraints— see [6] for precision grasps in the circular case.

IV. OBJECT SURFACE RECONSTRUCTION

As described in the previous sections, from the set of n silhouettes of the object, an elliptical initial reconstruction surface is generated, virtually placed around the object and sampled by n_s reconstruction points. A virtual mass is associated to each sample point, and four links are imposed by springs connecting the closest cross points, resulting in

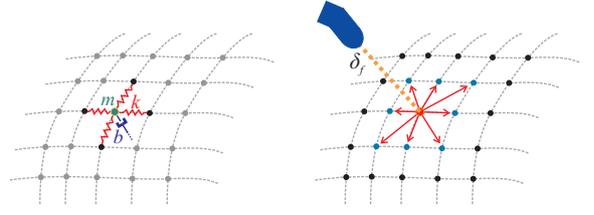


Fig. 3. Cross network topology of the reconstruction surface (on the left), and the contour of neighbor points (on the right).

a cross reticular topology for the reconstruction surface (see Fig. 3).

Each parallel of the ellipsoid should have the same number n_m of points, corresponding to the number of meridian, allowing the construction of a fully linked cross reticulum. In other words, for each point, the existence of a couple of corresponding points on the closest parallels of the grid is guaranteed. Without loss of generality, the number n_p of parallels is chosen equal to the number of meridians $n_p = n_m = \sqrt{n_s - 2}$. To avoid that the parallels nearest to the poles determine an unnecessary initial thickening of sample points around the poles, a suitable angular distribution of the parallels has been imposed, by reducing (augmenting) the density of the parallels near the poles (equator).

The model of the system, defining the deformation motion of the reconstruction surface, is described by the following dynamic equations:

$$m\ddot{\mathbf{x}}_{i,j} + b\dot{\mathbf{x}}_{i,j} + k(4\mathbf{x}_{i,j} - \mathbf{c}(\mathbf{x}_{i,j})) = \mathbf{f}_{i,j}, \quad (2)$$

for $i = 1, \dots, n_m$ and $j = 1, \dots, n_p$, where $\mathbf{c}(\mathbf{x}_{i,j}) = \mathbf{x}_{i-1,j} + \mathbf{x}_{i,j+1} + \mathbf{x}_{i+1,j} + \mathbf{x}_{i,j-1}$, and $\mathbf{x}_{i,j}$ is the position in the workspace of the sampling point at the intersection of the i -th meridian with the j -th parallel. The parameters m , k , and b represent the mass associated to the point, the constant spring linking the point with its nearest four cross points, and the constant spatial damper, respectively.

Vector $\mathbf{f}_{i,j}$ is the *reconstruction force* acting on the mass associated to the sample point (i, j) , which is attractive with respect to the border of the visual hull and is progressively reduced every time the corresponding point comes in or goes out from the visual hull:

$$\mathbf{f}_{i,j} = \alpha_{i,j}(t_{i,j}) F_a \mathbf{n}_{i,j}, \quad (3)$$

where $\mathbf{n}_{i,j}$ is the unit vector pointing from the current point (i, j) to the estimated centroid of the object, and $\alpha_{i,j}(t_{i,j}) F_a$ is the amplitude of the force. F_a is the maximum amplitude of the force and $\alpha_{i,j}(t_{i,j}) \in (-1, 1]$ is a discrete sequence of scale factors defined as follow:

$$\alpha_{i,j}(t_{i,j}) = -\epsilon \alpha_{i,j}(t_{i,j} - 1), \quad (4)$$

where $\epsilon \in (0, 1)$, $\alpha_{i,j}(0) = 1$, and $t_{i,j}$ is an integer index which starts from zero and is incremented every time the point (i, j) comes in or goes out the visual hull.

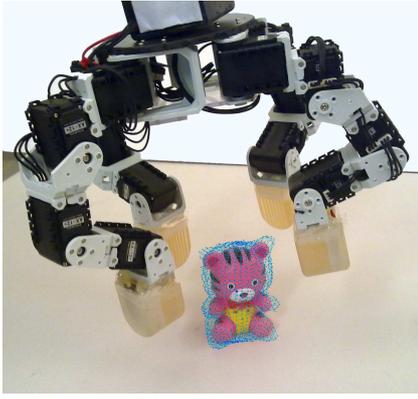


Fig. 4. Visual grasp concept.

V. PLANNER AND CONTROLLER MODULES

The current estimation of the object surface is stored in a proper buffer, which is continuously updated during the dynamic evolution of the elastic surface, and it is employed by the local grasp planner for updating the fingertips trajectories. The local grasp planner, in accordance with the current reconstructed object surface, generates the fingertips trajectories on the basis of suitable quality indices, but keeping a fixed *floating safety distance* δ_f between the fingertips and the corresponding reconstruction points, along the outgoing normal to the surface. The distance is exploited like a security parameter to avoid undesired collision between the fingers and the object before the final grasp. See Fig. 4 for an illustration of the visual grasp concept.

Namely, starting from the initial grasp configuration, the planner generates the motion of the fingertips from the current position to a new set of points of the updated surface, according to a force field associated at each contact point, until no improvements in the quality of grasp are reached. This new configuration of the contact points will be the new initial grasp configuration for the next iteration of the local grasp planner. The process ends when the object reconstruction algorithm reaches an equilibrium and the planner computes the final grasp configuration.

In the next subsections, the adopted quality indices employed to generate the force field, the finger trajectory planner and the kinematic motion controller are presented.

A. Motion field of forces

In this paper, planar grasps in the 3D space are considered, assuming that the moments and transversal forces acting on the object can be neglected. In particular, the desired optimal grasp is characterized by having all the contact points of the considered n_f -fingered hand lying on the same grasping plane in an equilateral configuration [6]. This choice guarantees the force closure property for a large number of situations and simplifies the computation of good grasps, although it may exclude a number of grasp configurations that can be more effective. Moreover, the area of the grasp polygon, resulting from the projection of the contact points

on the grasp plane, has to be maximized to improve the quality of the grasp with respect to possible external moments normal to the grip plane. Finally, if it is required by the desired application, it can be also imposed that the current surface reconstruction center of mass (that is equivalent to the reconstructed object center of mass at the end of the process) has to be contained in the current grasping plane, enhancing the minimization of gravity and inertial effects during manipulation tasks.

To reach this goal, a field of forces defined as the sum of suitable force contributions is associated at each contact point.

First, the interpolating plane Π of the current contact points —i.e., the plane which minimizes the distance from all the contact points—, and the projections \mathbf{p}_i^Π of the contact point \mathbf{p}_i on Π , with $i = 1, \dots, n_f$, are computed (see Fig. 5). Then, the force associated to the i -th contact point is defined as:

$$\mathbf{f}_i = \mathbf{f}_{\Pi i} + \mathbf{f}_{c_m} + \mathbf{f}_{e_i} + \mathbf{f}_{a_i} + \mathbf{f}_{b_i}, \quad (5)$$

where each contribution of force, with reference to Fig. 5, is defined as follows:

- $\mathbf{f}_{\Pi i} = k_\Pi (\mathbf{p}_i^\Pi - \mathbf{p}_i)$ is the force which moves \mathbf{p}_i to \mathbf{p}_i^Π , so that all contact points belong to the same grasp plane;
- $\mathbf{f}_{c_m} = k_{c_m} (\mathbf{c}_m - \mathbf{c}_m^\Pi)$ is the force, equal for all the contact points, which attracts the grasp plane Π to \mathbf{c}_m , where \mathbf{c}_m is the center of mass of the current reconstruction surface and \mathbf{c}_m^Π is the projection of the center of mass on the interpolating plane;
- $\mathbf{f}_{e_i} = k_e (\theta_i - \frac{2\pi}{n_f}) \mathbf{t}_i$ is the tangential force which is in charge of producing an equilateral grasp configuration, where θ_i is the angle between the vectors $\mathbf{c}_m^\Pi - \mathbf{p}_i^\Pi$ and $\mathbf{c}_m^\Pi - \mathbf{p}_j^\Pi$, with $j = i + 1$ for $i = 1, \dots, n_f - 1$, and $j = 1$ for $i = n_f$, and \mathbf{t}_i is the tangential unit vector normal to $\mathbf{c}_m^\Pi - \mathbf{p}_i^\Pi$, lying on Π and pointing toward \mathbf{p}_j^Π ;
- $\mathbf{f}_{a_i} = k_a (\mathbf{p}_i^\Pi - \mathbf{c}_m^\Pi) / \|\mathbf{p}_i^\Pi - \mathbf{c}_m^\Pi\|$ is a force component which tends to enlarge the area of the grasp polygon. The Euclidean norm is considered;
- \mathbf{f}_{b_i} represents the kinematic barrier force, depending on the local and global kinematic index, described in the next subsection.

The parameters k_Π, k_{c_m}, k_e, k_a are positive constant coefficients, suitably designed to weigh the single force contributions with respect to the requirements of the single situations and/or tasks to accomplish.

The force \mathbf{f}_i is then projected onto the tangential plane to the current reconstruction surface at the contact point i , determining the direction of the motion for the i -th contact point:

$$\mathbf{f}'_i = \mathbf{f}_i - (\mathbf{f}_i^T \mathbf{v}_i) \mathbf{v}_i,$$

where \mathbf{v}_i is the unit normal vector to the reconstruction surface at the point \mathbf{p}_i .

In particular, the direction of \mathbf{f}'_i individuates one of the points of the surface close to the current one, as shown in

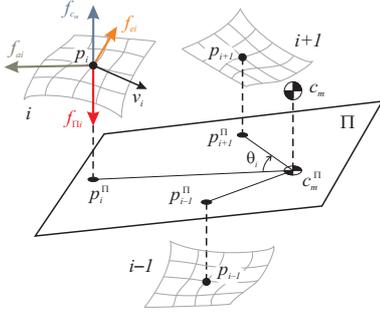


Fig. 5. Force field for the i -th floating contact point.

Fig. 3, employed by the planner to produce the floating motion of the finger. When $\|\mathbf{f}'_i\|$ is higher than a given threshold σ_f , the current grasp configuration changes according to the directions of \mathbf{f}'_i . Again, the Euclidean norm is here considered. The choice of σ_f means that forces \mathbf{f}'_i whose norm is under this threshold can be neglected, and when this happens for all the contact points, then the reached configuration is the local optimal grasp for the current iteration. Obviously σ_f affects both the accuracy of the grasp solution and the computational time, determining the number of iterations required to converge to the local optimum, and thus it must be suitably tuned considering this trade-off.

B. Kinematic barrier forces

The kinematic barrier force \mathbf{f}_{bi} for the i -th floating contact point is aimed at avoiding the motion of the finger along directions that cause the reaching of joint limits, joint or hand singularities, and collisions between fingers or with the palm. In detail, the barrier force is equal to

$$\mathbf{f}_{bi} = \mathbf{f}_{ji} + \mathbf{f}_{si} + \mathbf{f}_{ci},$$

where each term is related to one of the neighbor points of the contour, directed from the corresponding contour point towards the actual contact point:

- \mathbf{f}_{ji} is zero when the finger joint positions are far from their limits, while it quickly increases its magnitude, with a hyperbolic law, when one or more joint limits are approached at least for one of the contour points. Therefore, the force \mathbf{f}_{ji} is in charge to move the contact point far from unreachable positions.
- \mathbf{f}_{si} is zero when the finger configuration is far from kinematic singularities, while it quickly increases, with a hyperbolic law, when a kinematic singularity is approached. Therefore, \mathbf{f}_{si} represents a force that is repulsive with respect to the directions leading to finger singularities.
- \mathbf{f}_{ci} is zero when the fingers are far from each others and from the palm, while its magnitude is increased when a safety distance is violated.

Obviously when the sum of each contribution in (5) for a finger results in a zero force field, the corresponding contact point does not change its position in the actual step of

the current iteration of the planner stage. Finally, notice that the barrier forces can be also employed to cope with environmental constraints, e.g. object ground plane or other surrounding objects.

C. Finger trajectory planner and motion controller

The local grasp planner produces a sequence of intermediate target grasp configurations at each iteration of the object reconstruction algorithm, which ends with the optimal grasp configuration (in local sense). The intermediate configurations are used to generate the fingertip paths.

Namely, the sequence of intermediate configurations is suitably filtered by a spatial low-pass filter in order to achieve a smooth path for the fingers on the object surface. Notice that only the final configuration needs to be reached exactly, while the intermediate configurations can be considered as via points for finger trajectory generation, that can be computed in real-time with a one step delay.

With respect to the smooth paths through the points of the filtered configurations, the actual finger paths generated by the finger trajectory planner keep a distance δ_f along the normal to the surface. When the final configuration is reached, the safety distance is progressively reduced to zero, producing the desired grasp action, with directions of grasp perpendicular to the object reconstructed surface.

A kinematic control is used to allow a correct tracking of the trajectories given by the planner. In particular, a closed-loop inverse-kinematic algorithm [12] has been exploited to reach such goal. Such method require as input the desired position for each finger, and as output it gives the joints velocity for each finger. High gain low-level controllers are necessary to physically accomplish the task. Moreover, an interaction control (e.g. an impedance control) can be used to touch the object with a desired dynamic, and a force optimization algorithm could be used for a proper distribution of the grasp forces.

VI. EXPERIMENTS

The proposed method has been experimentally tested on different objects considering a different number of fingers of the available robotic hand of Fig. 4. Obviously, since it is not an anthropomorphic hand, a human-like grasp here means that it must be stable and the hand should be in a feasible and dexterous configuration. In the following, the results for the objects shown in Fig. 2, a teddy-bear and a little bottle, are presented.

Images in a number equal to $n = 13$ have been taken for all the objects by a common webcam mounted on a robot manipulator. The elliptical reconstruction surface is sampled with $n_s = 1500$ points while the dynamic reconstruction parameters have been chosen as follows: $m = 10^{-3}$ kg, $k = 0.3 \cdot 10^{-3}$ N/m, $b = 0.09 \cdot 10^{-3}$ Ns/m, and $F_a = 5$ N for both the objects. The $k_{\Pi}, k_{c_m}, k_e, k_a$ parameters for the grasp planner have been chosen all equal to 1, so to have an equivalent weight for all the contributions, while the threshold σ_f has been tuned to a value of 0.002 N.



Fig. 6. Steps of the object surface reconstruction process for the teddy-bear (left) and for the little bottle (right).

The multi-fingered hand used to simulate the grasp and to perform the experiments is depicted in Fig. 4. The floating security distance δ_f has been set to 2 cm, which is deliberately a huge value for a better visualization of the finger trajectories. The computational time for the whole process results in about 1.5 s on a Pentium 1.7 GHz.

In Fig. 6 some intermediate steps of the reconstruction algorithm are shown, while the finger trajectories and the final grasp configurations, respectively for the teddy-bear and for the little bottle, are shown in Fig. 7. Both cases of $k_{cm} = 1$ (on the left) and $k_{cm} = 0$ (on the right) are considered (the black/blue bold point represents the position of the object center of mass of the reconstructed object). In particular, for the case $k_{cm} = 1$ it is evident that both the grasp planes of the final grasps contain the center of mass of the objects, as desired, while for the case $k_{cm} = 0$ the plane of the final grasp is far from the center of mass to achieve a more extended areas of the grasp polygon.

More in detail, Figure 7 shows how the teddy-bear is grasped with three fingers achieving a desirable stable planar equilateral grasp (120° apart) for both cases of $k_{cm} = 1$ and $k_{cm} = 0$. The yellow lines represent the sequence of reconstruction points selected by the planner during the evolution of the reconstruction surface. The green lines represent the trajectories that the planner generates for the fingertips after spatial filtering and considering the safety distance. Finally, the red lines show the last part of the grasp trajectory, when the safety distance is progressively reduced achieving a perpendicular approach to the object surface.

For the case of the little bottle, four fingers of the hand has been considered. The final grasp configuration corresponds to the equilateral best grasp (90° apart) for the object. Moreover, the achieved trajectories are very regular due to the good choice of the initial grasp configuration evaluated by the preshaping module. This result is common when the object is symmetric with respect to one or more axes, and so it is well represented by an ellipsoidal surface. Of course, for the particular shape of the bottle, the results do not change significantly when k_{cm} is set to 0.

VII. CONCLUSION AND FUTURE WORK

A method to control the motion of a multi-fingered hand to achieve a human-like grasp of unknown objects has been presented, which is composed of an iterative object surface

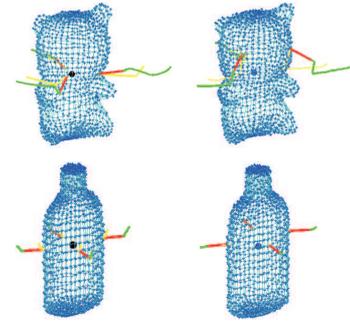


Fig. 7. Finger trajectories evaluated by the local grasp planner (green: approach, red: grasp) and the corresponding sequence of floating grasp points achieved during the reconstruction process (yellow) for the teddy-bear (top) and for the little bottle (bottom), both evaluated with $k_{cm} = 1$ (left) and $k_{cm} = 0$ (right).

reconstruction algorithm and of a local optimal grasp planner combined with a kinematic motion controller, evolving in a synchronized parallel way. The effectiveness of the proposed method has been confirmed by a number of case studies.

Future work is aimed at exploiting touch sensors to refine the final grasp, so as to deal also with possible local minima.

REFERENCES

- [1] S. P. Ananthanarayanan and A. A. Gldenberg, "Dexterous manipulation using qualitative reasoning, part 1: determination of a desired object displacement", *Fifth International Conference on Advanced Robotics*, Pisa, I, pp. 625-630, 1991.
- [2] S. P. Ananthanarayanan and A. A. Gldenberg, "Dexterous manipulation using qualitative reasoning, part 2: modelling and syntehsis of finger manipulative synergies", *Fifth International Conference on Advanced Robotics*, Pisa, I, pp. 631-636, 1991.
- [3] I. Biederman, "Recognition-by-components: a theory of human image understanding", *Psychological Review*, pp. 115-147, 1987.
- [4] F. Bley, V. Schmirgel and K.-F. Kraiss, "Mobile manipulation based on generic object knowledge", *15th IEEE International Symposium on Robot and Human Interactive Communication*, Hatfield, UK, pp. 411-416, 2006.
- [5] A. Chella, H. Dindo, F. Matraxia and R. Pirrone, "A neuro-genetic approach to real-time visual grasp synthesis", *International Joint Conference on Neural Networks*, Orlando, FL, 2007.
- [6] M. R. Cutkosky, "On grasp choice, grasp models, and the design of hands for manufacturing tasks", *IEEE Transactions on Robotics and Automation*, vol. 5, n. 3, pp. 269-279, 1989.
- [7] F. Fleischer, A. Casile and M. A. Giese, "View-independent recognition of grasping actions with a cortex-inspired model", *IEEE-RAS International Conference on Humanoid Robot*, Paris, F, pp. 514-519, 2009.
- [8] Y. Hu, R. Eagleson and M. A. Goodale, "Human visual servoing for reaching and grasping: the role of 3-D geometric features", *IEEE International Conference on Robotics and Automation*, Detroit, MI, pp. 3209-3216, 1999.
- [9] Y. Kim and M.A. Minor, "Distributed kinematic motion control of multi-robot coordination subject to physical constraints", *The Intrnational Journal of Robotic Research*, vol. 29, n.1, pp. 92-109, 2010.
- [10] D. Montana, "The kinematics of multi-fingered manipulation", *IEEE Transactions on Robotics and Automation*, vol. 11, pp. 491-503, 1995.
- [11] B. Ozyer and E. Oztop, "Task dependent human-like grasping", *IEEE-RAS International Conference on Humanoid Robots*, Daejeon, ROK, pp. 227-232, 2008.
- [12] B. Siciliano, L. Sciacivco, L. Villani and G. Oriolo, "Robotics. Modelling, Planning and Control", *Springer*, 2008.
- [13] R. Zollner, O. Rogalla, R. Dillmann and M. Zollner, "Understanding users intention: programming fine manipulation tasks by demonstration", *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Lausanne, CH, pp. 1114-1119, 2002.