Superquadrics Objects Representation for Robot Manipulation

Eliana Costa e Silva*, M. Fernanda Costa†, Wolfram Erlhagen† and Estela Bicho**

*CIICESI, ESTGF, Polytechnic of Porto and Centre Algoritmi, University of Minho, Portugal
†Centre of Mathematics/Dept. of Mathematics and Applications, University of Minho, Portugal
**Centre ALGORITMI/Dept. of Industrial Electronics, University of Minho, Portugal

Abstract. Superquadric are mathematically quite simple and have the ability to obtain a variety of shapes using low order parameterization. Furthermore they present closed-form equations and therefore can be used in the formulation of robotic movement planning problems, in particular in obstacle-avoidance and grasping constraints. In this paper we explore the modeling of objects using superquadrics. The classical nonlinear optimization problem for fitting shapes is extended by adding nonlinear constraints. The numerical results obtained by two different optimization methods are presented and a comparison of the volume of the superquadrics to the volume of simple ellipsoids is made.

Keywords: Nonlinear Optimization, Superellipsoids, Robot Manipulation, Obstacle Avoidance
PACS: 02.60.Pn

INTRODUCTION

Modeling objects is essential in several areas such as assistance and services robotics, computational mechanics and computer graphics (e.g. [1]; [2]; [3]; [4]). Superquadrics have been extensively used in object modeling. The main reason is that superquadrics: **(i)** present a close-form equation; **(ii)** are mathematically quite simple, involving a few sines, cosines, and exponents; **(iii)** have the ability to obtain a variety of shapes using only a small number of parameter; **(iv)** the control parameters affect global properties of the shapes in a comprehensible manner. Additionally, the solids can be easily modified by bending and twisting, and have the potential to become widespread in three-dimensional geometric design [5]. The closed-form equations obtained when using superquadrics can be used in the formulation of movement planning problems for robots, in particular in obstacle-avoidance constraints. In this paper we explore the modeling of objects using superquadrics. The classical nonlinear optimization problem for fitting 3D shapes is extended by adding nonlinear constraints. Traditionally fitting superquadric models to 3D data is performed by formulating an unconstrained nonlinear least squares problem and using the well known Levenberg-Marquardt method to solve it (see e.g. [1]; [2]). However, in this work the nonlinear least squares problem has constraints, and methods for solving constrained nonlinear optimization problems must be used. The numerical results are obtained using two optimization methods, provided by the MATLAB OPTimization Interface (OPTI) Toolbox [6], and a comparison of the volume occupied by the superquadrics and the volume of ellipsoids for modeling the same objects is made.

SUPERQUADRIC FOR OBJECTS REPRESENTATION

Superquadrics are a family of parametric shapes which can be classified into superellipsoid, supertoroid and superhyperboloid with one and two parts [5]. Superellipsoids and supertoroids are useful for a volumetric part-based object description. In fact they are compact in shape and have a closed surface. In this paper we focus on superellipsoids. The implicit form to describe a superellipsoid is:

$$f(\Lambda;x,y,z) = \left(\left(\frac{x}{a_1}\right)^{2/\varepsilon_2} + \left(\frac{y}{a_2}\right)^{2/\varepsilon_2} \right)^{\varepsilon_2/\varepsilon_1} + \left(\frac{z}{a_3}\right)^{2/\varepsilon_1} \quad (1)$$

where $\Lambda = (a_1,a_2,a_3,\varepsilon_1,\varepsilon_2)$ are the parameters of the superellipsoid, being $\varepsilon_1,\varepsilon_2 > 0$ the shape parameters and $a_1,a_2,a_3 > 0$ the scale parameters along the x,y and $z-$axis of the superquadric. The function f is known as the inside-outside function. For an object in a general position, (p_x, p_y, p_z) and orientation given by the Euler angles,

(ϕ,ψ,θ) , the new inside-outside function is calculated by inverting the transformation and substituting into the old inside-outside function (see [5] for details). Therefore, without loss of generality, here we consider objects that are located at the origin of the world frame and are aligned with its axis, i.e. $(p_x,p_y,p_z)=\vec{0}$ and $(\phi,\psi,\theta)=\vec{0}$. The volume of a superellipsoid is given by $V=2a_1a_2a_3\varepsilon_1\varepsilon_2 B\left(\frac{\varepsilon_1}{2}+1,\varepsilon_1\right)B\left(\frac{\varepsilon_2}{2},\frac{\varepsilon_2}{2}\right)$, where B is the beta function.
The inside-outside function is used to determine if a point $P_k(x_k,y_k,z_k)$ lies outside or inside the superquadric surface. The inside of each solid is given by $f(\Lambda;P_k)<1$; the surface is indicated by $f(\Lambda;P_k)=1$; and the outside by $f(\Lambda;P_k)>1$. Therefore, determining the parameters of the superquadric that better fits a set 3D-points $P_k,\ k=0,\ldots,n$, is equivalent to solving a minimization problem whose objective function is $\sum_{k=0}^{n}(f(\Lambda;P_k)-1)^2$. In order to avoid false approximations when the set of points is not closed to the superquadric surface but some of them still approximate it correctly, a special coefficient $\sqrt{a_1a_2a_3}$ is used. Additionally, for making the process time efficient, in [2] the authors propose to use the mean distorsion per point, $\frac{\sqrt{a_1a_2a_3}}{n+1}(f^{\varepsilon_1/2}(\Lambda;P_k)-1)^2$. Thus, the problem of determining the superquadric that better fits the 3D points is solved by determining the parameters Λ such that:

$$\min_{\Lambda\in\mathbb{R}^+}\sum_{k=0}^{n}\frac{\sqrt{a_1a_2a_3}}{n+1}(f^{\varepsilon_1/2}(\Lambda;P_k)-1)^2,\qquad \text{s.t.}\qquad \Lambda_{min}\le\Lambda\le\Lambda_{max}. \tag{2}$$

This is a bounded nonlinear optimization problem where Λ_{min} and Λ_{max} are the vectors of lower and upper bounds. For the problem of a robot grasping and manipulation objects, in order to guarantee collision-free movement (see e.g. [4]), we impose that every point P_k lies inside the superquadric. This is accomplished by adding the nonlinear constraints $f(\Lambda;P_k)<1,\ k=0,\ldots,n$, to the above optimization problem:

$$\begin{array}{ll}\min\limits_{\Lambda\in\mathbb{R}^+} & \sum\limits_{k=0}^{n}\frac{\sqrt{a_1a_2a_3}}{n+1}(f^{\varepsilon_1/2}(\Lambda;P_k)-1)^2\\ \text{s.t.} & \Lambda_{min}\le\Lambda\le\Lambda_{max}\\ & f(\Lambda;P_k)<1,\quad k=0,\ldots,n.\end{array} \tag{3}$$

RESULTS

Here we focus on modeling objects in the context of human-robot collaboration (see Figure 1 and e.g. [4]). The numerical results were obtained using an Intel(R)Core(TM)2Duo-2.13GHz, 4Gb RAM. The choice of the optimization method took into account that: the objective function is highly nonlinear and the analytic expression of its derivatives is not easy to determine. For these reasons we consider two different optimization methods, provided by the OPTI Toolbox [6], one applies a global derivative-free technique and the other a local derivative-based method. The solvers used were: (i) NLOPT_GN_ISRES - Improved Stochastic Ranking Evolution Strategy algorithm for nonlinearly-constrained global optimization which is a population-based stochastic derivative-free algorithm that implements a evolution strategy that is based on a combination of a mutation rule and a Nelder-Mead-like update rule; (ii) Ipopt - an interior point filter line search method that aims to find a (local) solution of a twice continuously differentiable nonlinear problem. OPTI Toolbox[6] was used with its default options, with the exception of the maximum number of function evaluations and maximum number of iterations that were set to 5.00e+4 and 5.00e+3, respectively. The problems were coded in MATLAB and no information of the first derivatives was supplied, therefore for Ipopt, the first derivatives were approximated using finite-differences and the BFGS update was used to approximate the Hessian matrix.
We consider that the objects are at the origin of the world frame and are aligned with its axis. We denote its dimensions on the main three axis by R_x,R_y and R_z. The 3D point coordinates are given by a CAD model of each object (see Table 1). As initial guess the following values were used: $a_1=R_x/2=d_x$, $a_2=R_y/2=d_y$, $a_3=R_z/2=d_z$, $\varepsilon_1=\varepsilon_2=1$. The bounds, Λ_{min} and Λ_{max}, were set to: $\Lambda_{min}=(0.49d_x,.49d_y,0.49d_z,0.1,0.1)$ and $\Lambda_{max}=(0.51d_x,0.51d_y,0.51d_z,2,2)$. Table 2 shows the numerical results of the ISRES and Ipopt solvers concerning superellipsoid fitting of the several objects, namely, a column, a base, a wheel, a nut and the torso of **ARoS** (see Figure 1). The first columns are relative to the solutions of problem (2) and the last are relative to problem (3). In this table we present the superquadrics parameters, $a_1,a_2,a_3,\varepsilon_1$ and ε_2, the objective function value, obj, the computational time in seconds, cpu, the percentage of points of the object outside the superquadric, o.p., the maximum distance to the superquadric of a point outside the superquadric to the superquadrics, m.d., and finally the percentage of volume decrease relatively to the ellipsoid model (see also Table 1). For each object it was possible to determine the superellipsoid that better fits its 3D points. For all objects the best results to problem (2), in terms of the objective function

TABLE 1. Dimensions of the objects considered in the numerical tests. In brackets, are presented the semi-axes of the ellipsoids and of the elliptic cylinder depicted in Figure 1

	Column	**Base**	**Wheel**	**Nut**	**Torso**
R_x	80 (60)	320 (180)	140 (85)	100 (60)	156 (120)
R_y	80 (60)	250 (145)	140 (85)	90 (60)	506 (300)
R_z	340 (190)	15 (27.5)	40 (35)	40 (35)	800
Number of points*	92	392	101	112	400
Volume of the ellipsoid	2.865133e+06	1.345858e+07	1.059240e+06	5.277876e+05	–

* Number of points on the 3D CAD model of each object

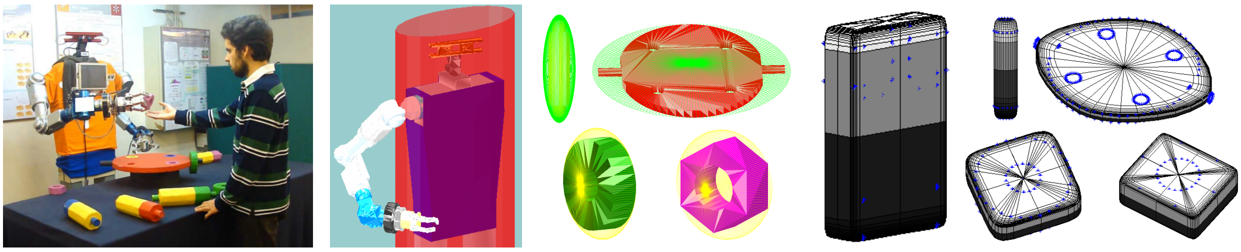

FIGURE 1. The image on the left corresponds to **ARoS** - an Anthropomorphic robot designed and built at University of Minho; the plots in the middle show **ARoS**' torso modeled as an elliptic cylinder and objects (column - yellow; red - base; green wheel; nut - purple) modeled using ellipsoids; the plots on right show objects modeled using superquadrics.

value and the computational cost, were obtained by Ipopt. The solutions of problem (2) present a significant reduction of the volume of the enveloping surface (approximately 58%, 51%, 71% and 35%) when compared to the ellipsoid used in [4]. However, a significant percentage of the data points are outside the superquadric. This is not desired if we consider that the closed-form equation of the objects is going to be used in the definition of the obstacle avoidance for movement planning of the robotic arms and hands. The numerical solutions for problem (3) present a slightly higher volume of the superquadric when compared to the volume obtained for the solutions of problem (2), but still significantly bellow the volume of the enveloping ellipsoid. Once more, Ipopt presents better results in terms of objective function value and cpu time. We can also observed that $\varepsilon_1 < 1$ and $\varepsilon_2 \approx 1$ for the objects similar to cylindroids (i.e. for the column and base) and $\varepsilon_1, \varepsilon_2 < 1$ for the cuboids (i.e. the torso). Furthermore, by observing Figure 1 we can see that for the wheel (green object) and for the nut (purple object), superellipsoid are not the best superquadric form. Supertoroids:

$$f(\Lambda; x,y,z) = \left(\left(\left(\left(\frac{x}{a_1}\right)^{2/\varepsilon_2} + \left(\frac{y}{a_2}\right)^{2/\varepsilon_2} \right)^{\varepsilon_2/2} - a_4 \right)^{2/\varepsilon_1} + \left(\frac{z}{a_3}\right)^{2/\varepsilon_1} \right), \tag{4}$$

with $\Lambda = (a_1, a_2, a_3, \hat{a}, \varepsilon_1, \varepsilon_2)$, where $a_4 = \hat{a}/\sqrt{a_1^2 + a_2^2}$ and $\hat{a}$ is the torus radius, is an alternative that we will explore in the future.

DISCUSSION AND FUTURE WORK

In this paper we have explored the modeling of objects using superellipsoids. We extended the classical nonlinear least squares problem for fitting shapes by adding nonlinear constraints. The numerical results were obtained by two different optimization methods provided by the MATLAB OPTI toolbox [6]. The results show that when superellipsoid are used a significant decrease of the volume enveloping the object is observed. However, we need to extend to other superquadric, as for example supertoroids. In the future we expect to include the superquadric object representation in the movement planning of an anthropomorphic robot.

TABLE 2. Numerical results. The best objective function value is highlighted.
$^{i)}$ $\Lambda_{max} = (0.55d_x, 0.55d_y, 0.55d_z, 2, 2)$; $^{ii)}$ Exceeded Iterations/Function Evaluations/Time

Algorithm	**ISRES**	**IPOPT**	**ISRES**	**IPOPT**
	Numerical solutions for (2)		*Numerical solutions for* (3)	
Column				
a_1	4.0800e+01	4.0800e+01	4.0800e+01	4.0800e+01
a_2	4.0800e+01	4.0800e+01	4.0626e+01	4.0626e+01
a_3	1.6660e+02	1.6660e+02	1.7062e+02	1.7062e+02
ε_1	5.7277e-01	5.7281e-01	2.4528e-01	2.4528e-01
ε_2	6.7914e-01	6.7870e-01	9.9990e-01	9.9990e-01
obj / cpu / vol.	4.8836e+00 / 3.029	**4.8836e+00** / 0.068	8.0368e+00 / 13.837	**8.0368e+00** / 0.095
o.p. / m.d.	70% / 1.4 / 58%	70% / 1.4 / 58%	0% / — / 60%	0%/ — / 60%
Base			$^{i)}$	$^{i)}$
a_1	3.1361e+02	3.1360e+02	3.3242e+02	3.3232e+02
a_2	2.5452e+02	2.5460e+02	2.7192e+02	2.7457e+02
a_3	1.4976e+01	1.4979e+01	1.5494e+01	1.5498e+01
ε_1	1.0017e-01	1.0000e-01	1.0000e-01	1.0000e-01
ε_2	1.2222e+00	1.2126e+00	1.3419e+00	1.3696e+00
obj / cpu	5.2984e+00 / 6.855	**5.2970e+00** / 0.231	8.3835e+00 / 47.779	**8.3808e+00** / 0.483
o.p. / m.d. / vol.	88% / 2.5 / 51%	88% / 2.5 / 51%	0% /—/ 56%	0% /—/ 56%
Wheel				
a_1	7.1400e+01	7.1400e+01	8.4995e+01	8.5000e+01
a_2	7.1009e+01	7.1009e+01	8.4988e+01	8.5000e+01
a_3	2.0113e+01	2.0113e+01	2.0031e+01	2.0031e+01
ε_1	1.0000e-01	1.0000e-01	1.0004e-01	1.0000e-01
ε_2	4.6997e-01	4.6881e-01	9.9970e-01	1.0000e+00
obj / cpu	3.3172e-02 / 3.466	**3.3168e-02** / 0.129	4.2373e-04 / 10.942	**4.2090e-04** /0.119
o.p. / m.d. / vol.	37% / 1.6 / 71%	37% / 1.6 / 71%	0 / — / 85%	0 / — / 85%
Nut			$^{i)}$	$^{i)}$
a_1	4.0800e+01	4.0800e+01	4.4000e+01	4.4000e+01
a_2	3.5334e+01	3.5334e+01	3.8105e+01	3.8105e+01
a_3	1.6089e+01	1.6089e+01	1.6139e+01	1.6139e+01
ε_1	1.0000e-01	1.0000e-01	1.0000e-01	1.0000e-01
ε_2	1.2846e-01	1.0000e-01	1.2594e-01	1.0335e-01
obj / cpu	2.6463e-02 / 2.684	**2.6460e-02** / 0.082	8.6036e-03 / 9.489	**8.5996e-03** / 0.123
o.p. / m.d. / vol.	29% / 1.6 / 35%	29% / 1.6 / 35%	0% / — / 40%	0% / — / 40%
Torso			$^{i)}$	$^{i)}$
a_1	7.6440e+01	7.6440e+01	8.0641e+01	8.0646e+01
a_2	2.4792e+02	2.4792e+02	2.6255e+02	2.6253e+02
a_3	3.9200e+02	3.9200e+02	4.1531e+02	4.1528e+02
ε_1	7.6135e-01	7.6099e-01	1.0000e-01	1.0000e-01
ε_2	1.0000e-01	1.0000e-01	1.0001e-01	1.0000e-01
obj / cpu	7.1563e+01 / 13.302	**7.1563e+01** / 0.179	1.9288e+02$^{ii)}$ / 52.359	**1.9288e+02** /0.496
o.p. / m.d.	31% / 2.1	31% / 2.1	0% / —	0% / —

REFERENCES

1. G. Biegelbauer, and M. Vincze, "Efficient 3D object detection by fitting superquadrics to range image data for robot's object manipulation," in *International Conference on Robotics and Automation*, IEEE, 2007, pp. 1086–1091.
2. T. T. Cocias, S. M. Grigorescu, and F. Moldoveanu, "Multiple-superquadrics based object surface estimation for grasping in service robotics," in *Int. Conf. on Optimization of Electrical and Electronic Equipment*, IEEE, 2012, pp. 1471–1477.
3. H. Dang, and P. K. Allen, *Autonomous Robots* **36**, 309–330 (2014).
4. E. Costa e Silva, M. F. Costa, J. P. Araújo, D. Machado, L. Louro, W. Erlhagen, and E. Bicho, *Appl. Math* **9**, 619–629 (2015).
5. A. H. Barr, *IEEE Computer graphics and Applications* **1**, 11–23 (1981).
6. J. Currie, and D. I. Wilson, *Foundations of computer-aided process operations, Savannah, Georgia, USA* pp. 8–11 (2012).