184 D. Page, Y. Fougerolle, A. Koschan, A. Gribok, M Abidi, D. Gorsich, and G. Gerhart, "SAFER Vehicle Inspection: A Multimodal Robotic Sensing Platform," in Proc. SPIE Unmanned Ground Vehicle Technology VI, Vol. 5422, Orlando, FL, USA, pp. 549-560, April 2004

# SAFER Vehicle Inspection: A Multimodal Robotic Sensing Platform

David L. Page,[a] Yohan Fougerolle,[a,b] Andreas F. Koschan,[a] Andrei Gribok,[a] Mongi A. Abidi,[a] David J. Gorsich,[c] and Grant R. Gerhart[c]

[a]Imaging, Robotics, and Intelligent Systems Laboratory, The University of Tennessee, Knoxville, TN, USA 37996-2100;
[b]LE2i-CNRS Université de Bourgogne, 12 rue de la fonderie, 71200 Le Creusot, France;
[c]U. S. Army RDECOM Tank-Automotive Research, Development and Engineering Center, Warren, MI, USA 48397-5000

## ABSTRACT

The current threats to U.S. security both military and civilian have led to an increased interest in the development of technologies to safeguard national facilities such as military bases, federal buildings, nuclear power plants, and national laboratories. As a result, the Imaging, Robotics, and Intelligent Systems (IRIS) Laboratory at The University of Tennessee (UT) has established a research consortium, known as SAFER (Security Automation and Future Electromotive Robotics), to develop, test, and deploy sensing and imaging systems for unmanned ground vehicles (UGV). The targeted missions for these UGV systems include—but are not limited to—under vehicle threat assessment, stand-off check-point inspections, scout surveillance, intruder detection, obstacle-breach situations, and render-safe scenarios. This paper presents a general overview of the SAFER project. Beyond this general overview, we further focus on a specific problem where we collect 3D range scans of under vehicle carriages. These scans require appropriate segmentation and representation algorithms to facilitate the vehicle inspection process. We discuss the theory for these algorithms and present results from applying them to actual vehicle scans.

**Keywords:** unmanned ground vehicles, vehicle inspection, computer vision, 3D segmentation, 3D representation

## 1. INTRODUCTION

Unmanned ground vehicles (UGV) are becoming a key element of security and surveillance in the post-9/11 environment. Currently, a wide variety of mobile platforms are available to meet the UGV mobility needs of both military and civilian community. These platforms include Andros,[1] Odis,[2] and PackBot[3] among others. These systems are robust and mature technologies that end-users can deploy today, but their imaging and sensing packages are typically a single video camera. Although video is essential for operator and driver awareness, other sensor modalities offer the potential to extend these existing platforms to become more than mobile cameras. This goal is the research objective of the Security Automation and Future Electromotive Robotics (SAFER) Program. The Imaging, Robotics, and Intelligent Systems (IRIS) Laboratory at the The University of Tennessee (UT) has established SAFER to develop, test, and deploy sensing and imaging systems that enhance the missions of current and future UGVs. In essence, SAFER seeks to deploy "sixth-sense" technologies such as nuclear, chemical, and biological (NCB) sensors, thermal imaging cameras, laser range scanners, and other advanced sensors and to incorporate autonomous intelligence into these sensors through the development of fusion and processing algorithms.

Further author information: (Send correspondence to D.L.P.)
D.L.P.: E-mail: dpage@utk.edu, Telephone: 1 865 974 8520, Fax: 1 865 974 5459



Unmanned Ground Vehicle Technology VI, edited by Grant R. Gerhart, Chuck M. Shoemaker, Douglas W. Gage, Proceedings of SPIE Vol. 5422 (SPIE, Bellingham, WA, 2004) · 0277-786X/04/$15 · doi: 10.1117/12.542808


**Figure 1.** The fundamental elements of a robotics platform. The SAFER program focuses on the processing component through fusion of multiple sensors.

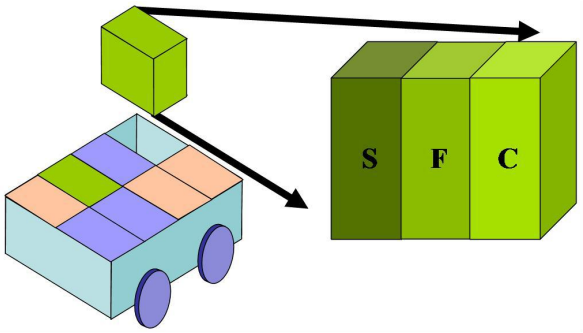



**Figure 2.** The mobile robot serves as a configurable platform that allows different sensor bricks to be added or removed. Each "SFC" brick is self contained with a sensor, fusion, and communications module.

### 1.1. SAFER Overview

As shown in Figure 1, the three fundamental elements of a robotics platform are: sensing, processing, and mobility. The main focus of the SAFER program is the development of processing algorithms. Currently, a variety of sensors are available and a variety of mobile platforms are available. The link that requires additional research is the processing to bring these elements together. Specific technologies that SAFER has targeted include processing and fusion of 3D geometry data from laser scanners, 2D video from visual and thermal cameras, and 1D signals from NCB detectors.

For the development of these technologies SAFER promotes the notion of "SFC bricks" to achieve an interchangeable sensor suite. A sensing, fusion, and communications (SFC) brick is a three-module concept. The sensor module contains one—or integrates multiple sensors—to collect data around the robot environment. The fusion module processes this data and incorporates reasoning and analysis. Finally, the communications module transmits this information to appropriate end users. The SFC brick concept allows the user to easily deploy and upgrade the system as new sensor bricks become available.

Figure 2 illustrates the SFC design strategy to achieve modularity, scalability, and reconfigurability. This particular example shows a wheeled platform with multiple SFC bricks that are independent, self-contained, drop-in units. Each brick "sees," "thinks," and "reports" such that these three actions are modules within the brick. The sensing, fusion, and communication modules, respectively, gather and log the data for a particular sensor and transmit this information to the appropriate data collection point. This collection point may or may not be the user controlling the mobile platform. For example, if the sensor detects nuclear contamination threats, then the brick may transmit directly to a group of nuclear experts, perhaps via satellite, to a remote location beyond the operational environment of the platform. This self-contained model means that information flows to the most effective data collection points without passing through a potential bottleneck or overloading unnecessarily the robot operator. Further, this design model allows distributed dissemination of information to more than one collection point.

### 1.2. Targeted UGV Mission

To focus research efforts, SAFER initially targets a specific UGV mission for the inspection of vehicle undercarriages. The key design of this system is for the robot and sensors to have a low profile for navigation underneath a vehicle such as a car or truck. Figure 3 illustrates this mission. This figure shows a prototype platform that has the flexibility to house a variety of different sensors. The platform is able to navigate under a vehicle through a remote operator interface. The opening just above the IRIS logo contains a mirror system to allow different


(a) (b)

**Figure 3.** These images depict the first prototype for the SAFER Mobile Sensor Platform. (a) The configurable sensor bay has a view portal just above the IRIS logo in this image. (b) The low-profile platform can navigate remotely under vehicles.

sensors to view up and under a vehicle. This prototype is able to maneuver completely under standard cars and trucks.

The SAFER program is using this platform to experiment with different sensor configurations to study the under-vehicle inspection problem. Vehicle inspection is traditionally accomplished through security personnel walking around a vehicle with a mirror on the end of a stick. That person is able to view underneath a vehicle with the mirror to identify contraband such as weapons, bombs, or other security threats. The challenge however is that the mirror-on-a-stick approach only allows partial coverage under a vehicle, and the mirror is often restricted by ambient lighting conditions such as poor lighting or sunlight glare. The prototype above seeks to overcome these issues by allowing complete coverage with the low-profile robot and extending beyond visible inspection by using NCB detectors and thermal cameras. Additionally, the stand-off capabilities that the prototype offers is an attractive alternative where potential harm to security personnel is possible. A mirror-on-a-stick solution puts personnel in harms way, but the remote wireless links of the prototype in the figures allows the user to remain at a safe distance.

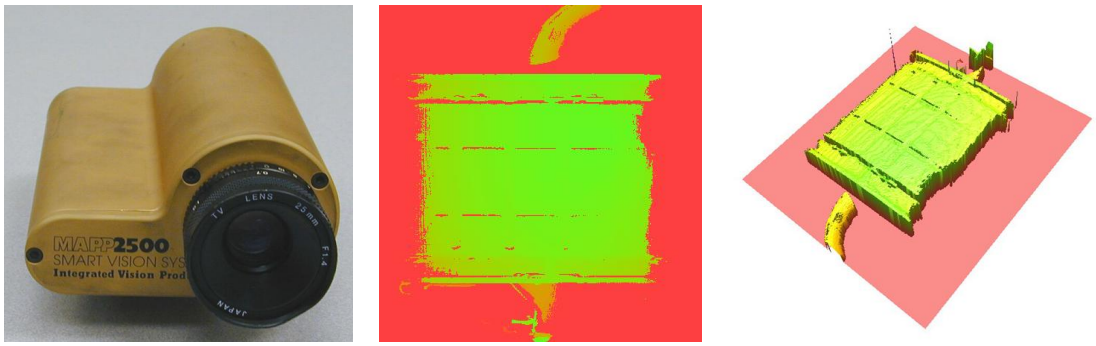
### 1.3. Development of a 3D Sensor Brick

One area that the SAFER program has identified as an important research topic for under vehicle inspection is the development of a 3D sensor brick. Following the SFC brick concept outlined above, the intent is to develop a modular brick based on a 3D laser range scanner. We note that SAFER is exploring other sensors for the SFC brick concept such as mosaics for video,[4] but in this paper we specifically focus on 3D scanners. A laser range scanner is a sensor that is similar to a stereo vision camera in that the scanner generates 3D images of a scene, but laser scanners provide more detailed and accurate data than more familiar passive stereo systems. This brick should contain the scanner and processing algorithms to allow collection of 3D data underneath a vehicle. The profile of this brick should fit within the profile of the platform in Figure 3. For the development of this brick, we have identified three key research elements: data collection, surface reconstruction, and object segmentation.

In this paper, we present these elements and give special emphasis to the surface reconstruction component. Specially, we present a superquadrics-based representation to aid in reconstructing and storing 3D information with regard to vehicle undercarriages. The next section explores these topics in detail as this paper now shifts away from introducing SAFER and focuses on a specific aspect of the 3D sensor brick problem.

## 2. 3D MODELING OF VEHICLE UNDERCARRIAGE

In this section, we present the three main processing phases from the data collection to the modeling of a vehicle undercarriage. This processing supports the development of the 3D range sensor brick. We first present issues




(a) (b) (c)

**Figure 4.** Data collection with the IVP Ranger System. (a) The camera for the IVP system. (b) Color-coded range image example of a muffler. The colors indicate distances to the muffler from the camera. (c) A 3D model generated from calibration of the camera and the range image in (b).

and practical solutions for acquiring data in the tight constraints under a vehicle, and then we present a scheme based on superquadrics to model these various components. Superquadrics are a highly compact representation, which makes them an attractive representation for the large quantities of data that we anticipate. Further, the superquadrics representation offers potential for future development of object matching algorithms. Thus, for data reduction and the potential for matching (with perhaps automotive databases), we seek to develop the following pipeline for collecting under vehicle data and then modeling that data with superquadrics.

## 2.1. Data Collection

The first phase of the 3D range sensor brick is data collection. For our prototype system, we have selected the IVP Ranger System,[5] which is a sheet-of-light profile scanner. A photograph of the camera appears in Figure 4(a). The basic output of the scanner is a single range profile. This profile lies in the plane of the sheet of light cast by the laser. The Ranger camera captures an image of the laser line as it warps across the scene and then converts this line into a 3D profile. The Ranger is able to scan over 500 profiles per second. For our tests we stack 512 profiles together to form a $512 \times 512$ range image with 256 range bins. An example of a range image for a muffler appears in Figure 4(b). We are testing a variety of configurations for scanning under the vehicle, but in general, these configurations offer a few millimeter resolution per range bin. With proper system calibration, we convert these range images into appropriate triangle-mesh data structures. An example appears in Figure 4(c). These 3D triangle meshes are the basic representation for the raw data from a range scanner.

Under ideal laboratory conditions, data collection with a range scanner is straightforward. Underneath a vehicle however, we must address a several challenges. One such challenge is the ambient illumination beneath the vehicle. Typically, the lighting conditions are poor. For active range imaging, which is the case we are discussing in this paper, this problem is minimal since we are using a laser source for illumination. However, for other applications such as video, lighting conditions are important factor for consideration. Although ambient lighting is not a major issue with laser scanners, spectral reflections are. The metallic surfaces under a vehicle exhibit strong spectral reflection properties. A laser can further complicate this problem as inter-reflections lead to errors in range estimation. A promising approach for this problem involves the use of an optical filter tuned to the frequency of the laser. The filter allows the data collection to isolate the initial reflection of the laser and thus improve the range sensing capabilities. The final noise issue that we discuss involves the trajectory of the robot itself. We assume a linear trajectory under the vehicle but in practice, the robotic platform must navigate over uneven terrain with irregularities such as holes and bumps. These irregularities combined with the small distance between the scanner and the vehicle undercarriage create and amplify distortions during the data acquisition phase.

Besides measurement noise, the next problem to consider is that range scanning under a vehicle suffers from significant view occlusions. Consider that the narrow and tight space under a vehicle restricts the robot to

maneuver the camera in a horizontal plane. The most obvious occlusion is that the camera can only view one side of a component—the bottom side facing straight down towards the ground. The muffler for example in the figure above is a one-sided view. Without dismantling the car, we can not move the range scanner to the other side of the muffler. Such an occlusion should illustrate the potential of other occlusions such as one object partially covering another object. The objects underneath a vehicle have various shapes and scales located at different depths. For example, the presence of a transmission axle close to the scanner occludes components mounted closer to the vehicle body. The solution to this problem is to use multiple scans to fill as much as possible the areas without information. This solution is only a partial one, though. The different views and scanning angles are extremely restricted by the low-profile under a vehicle. Thus an integration and fusion of multiple scans only partially fills the occlusion holes, but multiple scans significantly enhances the data. As a result, we scan underneath a vehicle with multiple passes and at different angles.

The final challenge that we consider with the data collection is the data redundancy inherent to laser range scanning. A single range image with $512 \times 512$ pixels yields over 250K data points. With additional scans to overcome occlusions (as noted above) and to achieve full coverage under a vehicle, this number quickly grows to several million data points. This large data set allows high-fidelity geometry that other 3D sensors do not offer, but the price is that of data redundancy and a potential data overload. The solution is to remove the redundancy. Superquadrics are an answer since they model complex objects and surfaces with just a few parameters. For our 3D sensor brick research, we have chosen to explore superquadrics as a representation for under vehicle data.

## 2.2. Superquadric Representation

In this section, we discuss our scheme for representing the 3D geometry data for under vehicle inspection. The processing proposed in this section aids the development of a 3D sensor brick for SAFER. We begin this section by first discussing a segmentation algorithm to partition the data since a single superquadric is insufficient—and somewhat counterproductive—for representing the entire undercarriage of a vehicle. Superquadrics however are quite suitable for representing individual components and parts such as the canister of a muffler or the piping for the exhaust system. Segmentation is necessary to first divide the under vehicle data into individual components before fitting superquadrics to those components. We then follow with subsequent sections that formalize superquadrics and demonstrate their recovery through fitting.

### 2.2.1. Surface Segmentation

Several segmentation algorithms are available in the literature that are specific to the superquadric problem[6,7] and to the more general mesh segmentation.[8–14] The approach that we will take has been proposed by Page et al.[15] and is known as "Fast Marching Watersheds." This algorithm is not specific to superquadrics but rather is a perceptually based algorithm that segments a triangle mesh into visual parts. This segmentation is based on a human vision theory known as the minima rule[16] and thus yields segmentations that are similar to how a human observer might segment a scene. An example segmentation with this algorithm appears in Figure 5. Zhang et al.[17] have used a similar segmentation algorithm for superquadric representation. The underlying principle is the watershed segmentation from image processing[18] and found in mesh processing.[8,14]

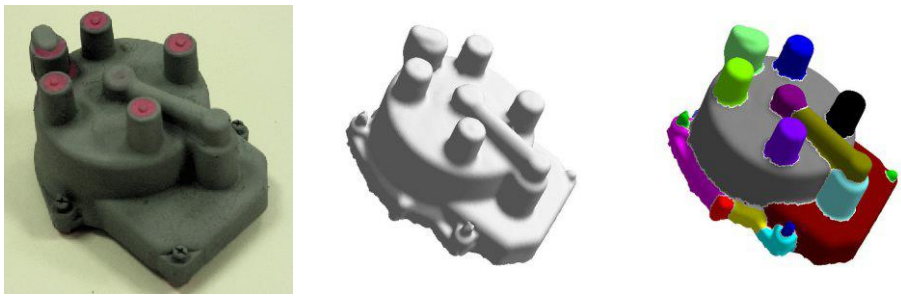
To improve the segmentation results, we have enhanced Page et al.[15] by using discrete watershed-height levels as opposed to the original continuous definitions. The algorithm in Page et al.[15] defines a directional curvature height $\tilde{h}_{uv}$ between two vertices ($u$ and $v$) of the mesh data using Euler's formula as follows:

$$\tilde{h}_{uv} = \kappa_1 \cos^2(\theta_{uv}) + \kappa_2 \sin^2(\theta_{uv}) \tag{1}$$

where $\kappa_1$ and $\kappa_2$ are the maximum and minimum principal curvatures, respectively, at $u$ as estimated by the algorithms in Taubin[19] or Page et al..[20] The angle $\theta_{uv}$ is the angle between the maximum principal direction and the vector connecting $u$ to $v$ in the tangent plane of $u$. This value $\tilde{h}_{uv}$ is a continuous height map across the mesh.

Our proposed enhancement is to set $\tilde{h}_{uv}$ to one of $L$ discrete levels. Thus, we specify a discrete directional height $h_{uv}$ as

$$h_{uv} = \left\lfloor L \frac{\tilde{h}_{uv} - \tilde{h}_{\min}}{\tilde{h}_{\max} - \tilde{h}_{\min}} \right\rfloor \tag{2}$$



(a) (b) (c)

**Figure 5.** Segmentation of an automotive distributor cap. (a) The original object. (b) Computer rendering of the cap after scanning into computer with IVP Ranger. (c) Segmentation of (b) into visual parts by algorithm in Page et al..[15]

where $\lfloor \cdot \rfloor$ denotes the floor function and $\tilde{h}_{\max}$ and $\tilde{h}_{\min}$ are the maximum and minimum values for each $\tilde{h}_{uv}$. Both $h_{uv}$ and $L$ are integers.

### 2.2.2. Superquadrics Formulation

After we have segmented a 3D data set into parts, the next step is to fit superquadrics to these individual parts. We begin with a brief background into superquadrics. Superquadrics have been introduced by A. H. Barr[21,22] and are used as a quantitative model for various applications in computer environments, both in computer graphics and in computer vision.[6,7,23,24] These primitives have parametric formulation (3), which is useful to efficiently and accurately polygonize a surface for efficient display on common graphical systems, and an implicit formulation (4), which is useful during the recovery and reconstruction process.

$$\begin{aligned} x &= a_1 \cos^{\varepsilon_1} \eta \cos^{\varepsilon_2} \omega \\ y &= a_2 \cos^{\varepsilon_1} \eta \sin^{\varepsilon_2} \omega \\ z &= a_3 \sin^{\varepsilon_1} \eta \end{aligned} \tag{3}$$

where $-\frac{\pi}{2} \leq \eta \leq \frac{\pi}{2}$, $-\pi \leq \omega \leq \pi$, and parameters $a_1$, $a_2$ and $a_3$ determine the size of the superquadric in the $x$, $y$ and $z$ directions. The parameters $\varepsilon_1$ and $\varepsilon_2$ determine the shape of the cross section. Superquadrics can represent a large set of shapes with various $\varepsilon_1$ and $\varepsilon_2$, ranging from star shapes to cubes. In practice, only convex shapes are considered, which implies $0 < \varepsilon_1 \leq 2$ and $0 < \varepsilon_2 \leq 2$, generating shapes from a diamond to a cube.

From the explicit definition, the following equation can be deduced:

$$F(x,y,z) \equiv \left( \left( \frac{x}{a_1} \right)^{\frac{2}{\varepsilon_2}} + \left( \frac{y}{a_2} \right)^{\frac{2}{\varepsilon_2}} \right)^{\frac{\varepsilon_2}{\varepsilon_1}} + \left( \frac{z}{a_3} \right)^{\frac{2}{\varepsilon_1}} = 1 \tag{4}$$

In order to place and orient a superquadric in space, a homogeneous transformation—composed of a rotation and translation—has to be applied. Finally, to increase the variety of shapes that can be represented by superquadrics, two global deformations are performed. A linear tapering can be applied to represent shapes such as cones and the bending proposed by Jaklič and Solina[25] is used to model more complex shapes such as the handle of a cup or any other bent object. The tapering is defined as follows:

$$\begin{aligned} X &= f_x(x) = \left( \frac{k_x}{a_3} z + 1 \right) x \\ Y &= f_y(y) = \left( \frac{k_y}{a_3} z + 1 \right) y \\ Z &= z \end{aligned}$$

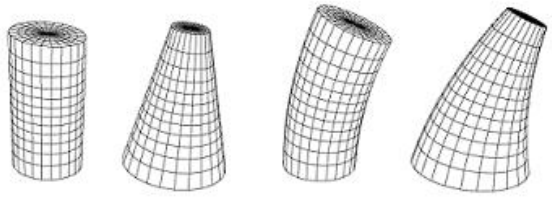


**Figure 6.** Globally deformed superquadrics. From left to right: original object, tapered object, bent object and combination of both deformations

where variables $k_x$ and $k_y$ represent the tapering factors along the $x$ and $y$ axes respectively with $-1 \leq k_x, k_y \leq 1$. Coordinates $(x, y, z)$ define a point on the surface of the original superquadric, and $(X, Y, Z)$ define the corresponding point on the tapered surface.

The bending deformation defined by Jaklič and Solina[6] differs from the one defined by Barr,[22] by the introduction of an additional parameter $\alpha$ that allows for bending in any plane that goes through the $z$ axis of the object coordinate system. A bent superquadric is shown in Figure 6. The bending deformation is defined by the following equations:

$$\begin{aligned} X &= x + (R - r)\, cos\alpha \\ Y &= y + (R - r)\, sin\alpha \\ Z &= \left(\frac{1}{k} - r\right) sin\gamma \end{aligned}$$

where $k$ represents the curvature of the bending plane, and $\alpha$ the bending angle around the $z$-axis. Other intermediate parameters $\gamma$, $r$ and $R$ are evaluated using:

$$\begin{aligned} \gamma &= z \times k \\ r &= \sqrt{x^2 + y^2} cos\theta \\ R &= \frac{1}{k} - \left(\frac{1}{k} - r\right) cos\gamma \\ \theta &= \alpha - \beta \\ \beta &= \arctan\frac{y}{x} \end{aligned}$$

Finally, considering the shape coefficients, scaling factors, coefficient for rotations, translation and the global deformations, we can describe a globally deformed superquadric positioned in space by a set of 15 parameters.

$$\Lambda = \{a_1, a_2, a_3, \varepsilon_1, \varepsilon_2, \phi, \theta\, \psi, T_x, T_y, T_z, k_x, k_y, k, \alpha\}$$

### 2.2.3. Superquadrics Recovery

With the above formulation, we are now ready to recover superquadrics from our under vehicle data sets. We refer to the vertices of a triangle mesh as a point cloud. If we take the output of a 3D range image and convert it into a triangle mesh, we can simply use the vertices of the mesh as a point cloud. Recovering superquadrics from point clouds is a common practice in the literature. Superquadrics can be recovered from cloud points using standard linear optimization methods such as the Levenberg-Marquard method. Depending on the cost function chosen for the optimization process, the reconstructed superquadric can fit incomplete data. Parameter recovery is achieved by minimizing an appropriately defined objective function. The objective function serves as an error metric to evaluate the accuracy of the recovered model. Both the accuracy and the efficiency of the data fitting

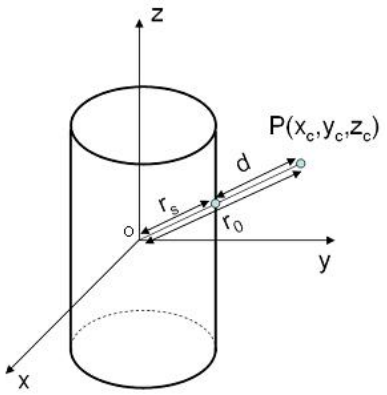



**Figure 7.** Radial Euclidian distance between a point P and a superquadric.

process heavily depend on the objective function used. Two objective functions have been primarily used in the literature. Solina and Bajcsy[25] presented an objective function based on the implicit definition of superquadrics.

$$G(\Lambda) = a_1 a_2 a_3 \sum_{i=1}^{n} (F^{\varepsilon_1}(x, y, z) - 1)^2 \tag{5}$$

Gross and Boult[26] proposed another objective function based on the radial Euclidean distance (9) and concluded that the objective function based on the Euclidean distance had less bias on the recovered shape parameters $(\varepsilon_1, \varepsilon_2)$ than the objective function proposed by Solina and Bajcsy[6] but fails to capture accurately the dimension of the object. The radial distance can be evaluated by

$$d = |r_0||1 - F^{\frac{-\varepsilon_1}{2}}(x_c, y_c, z_c)| = |r_s||F^{\frac{\varepsilon_1}{2}}(x_c, y_c, z_c)| \,, \tag{6}$$

where $r_0 = \|\overrightarrow{OP}\|$ represents the radius of a point $P$ in the canonical system and $r_s$ the radius of the intersection point of the superquadric surface and the line $[O, P)$ as shown in Figure 7.

Using this expression leads to the following equation:

$$F^{\varepsilon_1}(x_c, y_c, z_c) - 1 = \frac{d}{|r_s|}\left(\frac{d}{|r_s|} + 2\right) \,, \tag{7}$$

where $d$ is the radial euclidian distance of the point $(x_c, y_c, z_c)$. Finally, the objective function given in (5) can be rewritten as

$$G_1(\Lambda) = (a_1 a_2 a_3) \sum_{i=1}^{N} \left(\frac{d_i}{|r_s|} + \left(\frac{d_i}{|r_s|} + 2\right)\right)^2 \,, \tag{8}$$

and Gross and Boult function is defined by

$$G_2(\Lambda) = \sum_{i=1}^{N} d_i^2 = \sum_{i=1}^{N} \left(|r_0| \left|1 - F^{\frac{\varepsilon_1}{2}}(x_c, y_c, z_c)\right|\right) \,. \tag{9}$$

Both of these cost function can be used. In practice, we use the one proposed by Solina in equation (5). From this recovery method and the data collection in the previous section, we are able to scan underneath a vehicle and develop superquadric models for that data. In the next section, we present experimental results.

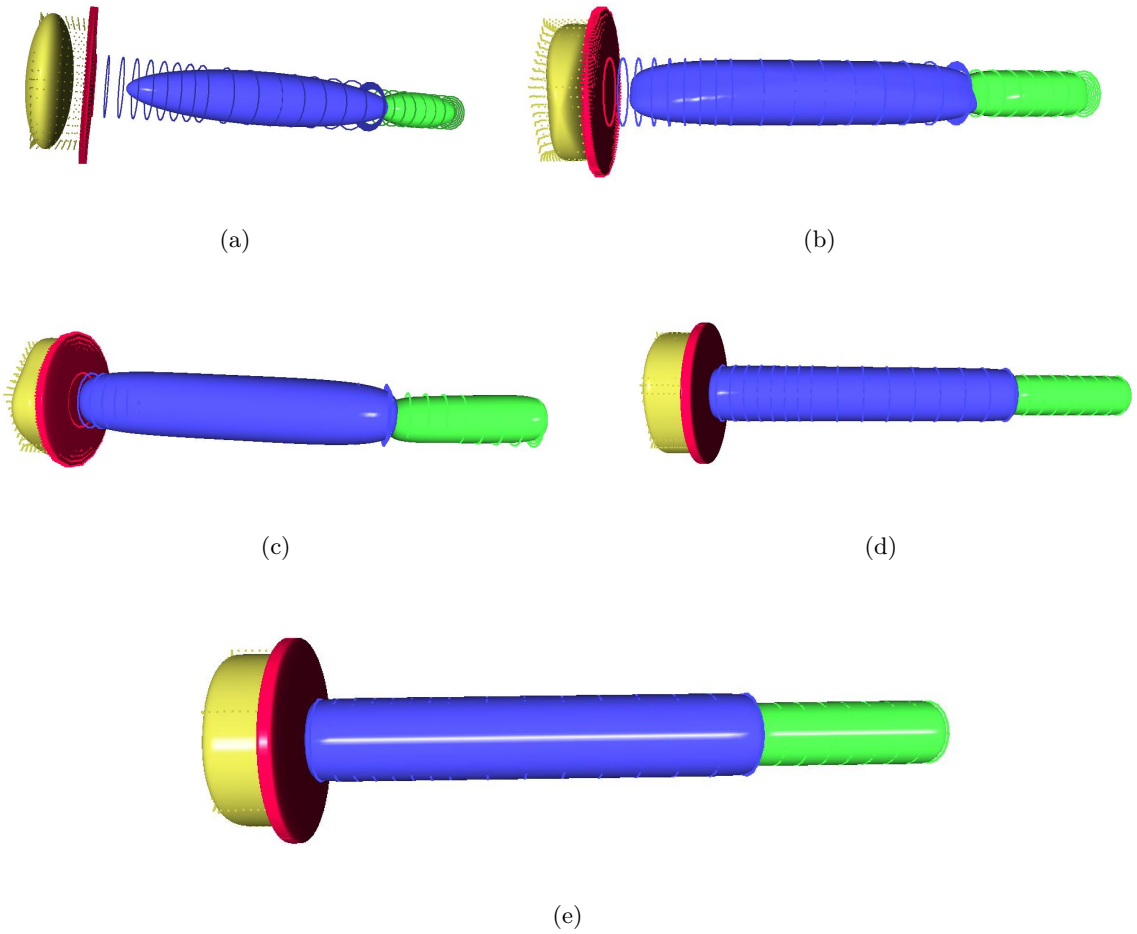



**Figure 8.** Superquadric recovery process on synthetic data. (a) At initialization, ellipses are positioned into the oriented bounding box of each point cloud. At each iteration, the global fitting error is reduced. (b) After 5 iterations. (c) After 10 iterations. (d) After 20 iterations. (e) After 50 iterations. The optimization process stops when a limit number of iterations is reached or when the error is below a certain threshold.

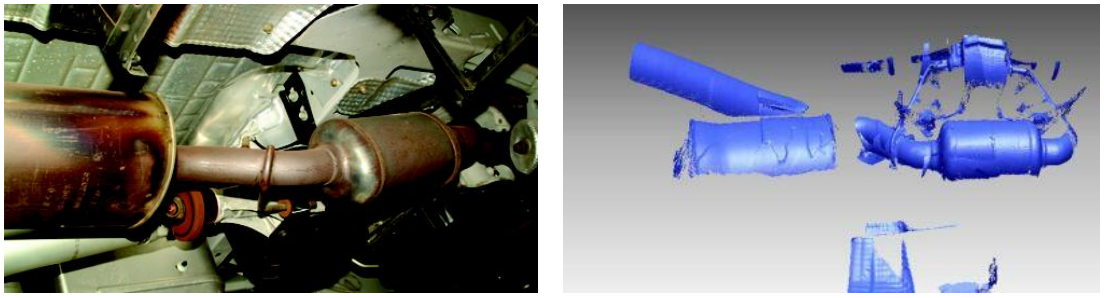
## 3. EXPERIMENTAL RESULTS WITH RANGE SCANNERS

We have implemented the above data collection and superquadric representation pipeline using the previously mentioned IVP Ranger System and processing algorithms in C++ on a desktop computer. This configuration is the first step towards an SFC 3D sensor brick. In this section, we present the experimental results obtained with this system, but first begin with a synthetic data set to illustrate the superquadric recovery process.

The sequence in Figure 8 demonstrates the iterative recovery of superquadrics for a synthetic car axle. The point clouds in Figure 8(a) appear as dots and the ellipsoids represent the initialization. In this case, the axle has been segmented into four regions (yellow, red, blue, and green). Each region has its own point cloud, and they are color-coded as well. The initialization involves seeding each region with an initial superquadric. Then, using the optimizations outlined in the previous section, each superquadric is iteratively recovered until reaching a stopping criteria. Figures 8(a)–8(e) illustrate this iterative sequence.

Following this illustration with synthetic data, we next begin with data collection for real under vehicle data sets. Figure 9 shows a sequence of eleven different range scans from under a vehicle. These range images are from the IVP scanner and are centered around the exhaust system. In this figure, the range images have grey values that represent distances from the range scanner to the vehicle. The darker the pixel, the further away the surface

**Figure 9.** This sequence of images shows the multiple scans necessary to acquire a complete image of the exhaust system. The mosaic of scans is an overlay of eleven different 3D range images from the IVP Ranger.

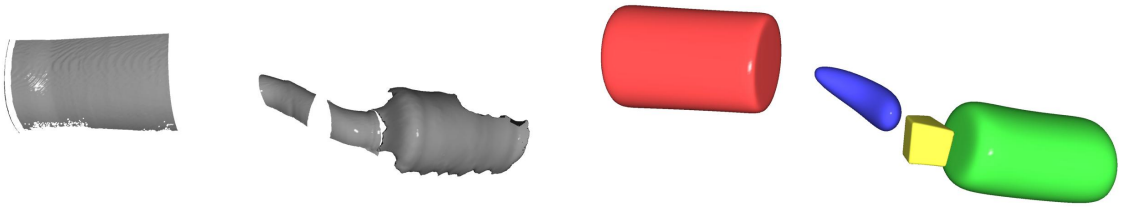


(a) (b)

**Figure 10.** View of the exhaust system underneath a vehicle. (a) Color image showing the muffler and exhaust piping. (b) Surface mesh obtained from the integration of the eleven range images. This image is a screen shot from a 3D interactive viewer.

is from the scanner. A full black pixel represents no range measurement available, which occurs when a surface is beyond the range of the sensor. We have mosaicked these images together to show their relative position to each other and to give the viewer a sense of the need for multiple views. The photograph in Figure 10(a) shows a color image of the exhaust system to give a context for the data set. The major components of interest are the muffler and catalytic converter. After registering and combining the eleven views from Figure 9, we can create a 3D model of the exhaust system as shown in Figure 10(b). This latter figure serves as the raw data for the segmentation algorithm and for superquadric fitting. We note that this raw mesh consists of over 125K vertices—a large data structure.

With the initial mesh in Figure 10(b), we next apply the segmentation algorithm and remove small regions after segmentation. The small regions occur because the scanner is unable to view the entire component due to significant occlusion. As a result, segmentation yields small regions that are not practical for superquadric



(a) (b)

**Figure 11.** Parts and superquadrics representation of the muffler. (a) Muffler and catalytic converter after cleaning, smoothing and segmentation. (b) Superquadric representation.

recovery. We filter these regions with a threshold. This post processing leads to the mesh shown in Figure 11(a) where the parts represent the muffler, catalytic converter, and sections of the exhaust pipe. We note that again occlusions lead to incomplete modeling of the exhaust piping, but enough data is available to recover some superquadric shape. The next step is to fit superquadrics to these parts. Figure 11(b) shows the recovered superquadrics corresponding to each part. We used the vertices of the 3D mesh in Figure 11(a) to form a point cloud data set. Then, we recovered the superquadrics as outlined with equations in the previous section.

To consider the data reduction, the four superquadrics represent 15 parameters each. If we implement each parameter as a floating point number, we require a total of 60 floating point numbers to store this 3D geometry data. By comparison, the prior triangle mesh with over 125K vertices requires three floating point numbers for each vertex for a total of over 375K floating point numbers just to store the vertex information. This number does not include the storage required for the triangle connectivity of the mesh. With these results, we can see the significant reduction—on the order of one to two magnitudes—that superquadrics provide.

## 4. CONCLUSIONS

With this paper, we have introduced the SAFER Program. The main goal of SAFER is to develop sensing and processing algorithms to enhance UGV security and surveillance missions. SAFER proposes the SFC brick concept as a modular solution to the development of sensor packages. With modular sensor bricks, end users have the choice to select different sensor bricks that best suit the needs of a given UGV mission. As an illustration for the development of a sensor brick, we have also presented in this paper research for algorithm development in a 3D range sensor brick. We have focused this research on data representation to reduce redundancy and to overcome data overload associated with range scanners. Specifically, we have discussed a superquadrics based representation and shown experimental results.

## ACKNOWLEDGMENTS

The authors would like to thank Sreenivas Rangan for his help with data collection. This work is supported by the University Research Program in Robotics under grant DOE-DE-FG02-86NE37968, by the DOD/ RDE-COM/NAC/ARC Program, R01-1344-18, by the US Army under grant Army-W56HC2V-04-C-0044, and by FAA/NSSA Program, R01-1344-129/130 and R01-1344-088.

## REFERENCES

1. Remotec. Andros family of tracked vehiles, Oak Ridge, TN, www.remotec-andros.com.
2. N. S. Flann, K. L. Moore, and L. Ma, "A small mobile robot for security and inspection operations," *Control Engineering Practice* **10**, pp. 1265–1270, Nov. 2002.
3. iRobot. PackBot family of tracked vehiles, Burlington, MA, www.irobot.com.

4. A. Koschan, J.-C. Ng, and M. Abidi, "Multiperspective mosaics for under vehicle inspection," in *Proceedings of the SPIE Unmanned Ground Vehicle Technology*, **5422**, 2004.
5. Integrated Vision Products, *User Documentation: MAPP Ranger System.* Sweden, 2000.
6. A. Jaklič, A. Leonardis, and F. Solina, *Segmentation and Recovery of Superquadrics*, Kluwer Academic Publisher, Dordrecht, 2000.
7. D. N. Metaxas, "Deformable models for segmentation, 3D shape and motion estimation and recognition," in *Proceedings of the British Machine Vision Conference*, 1999.
8. A. P. Mangan and R. T. Whitaker, "Partitioning 3D surface meshes using watershed segmentation," *IEEE Transactions on Visualization and Computer Graphics* **5**, pp. 308–321, Oct.–Dec. 1999.
9. K. Wu and M. D. Levine, "3D part segmentation using simulated electrical charge distributions," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **19**, pp. 1223–1235, Nov. 1997.
10. L. Vincent and P. Soille, "Watersheds in digital spaces: an efficient algorithm based on immersion simulations," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **13**, pp. 583–598, June 1991.
11. B. Falcidieno and M. Spagnuolo, "Polyhedral surface decomposition based on curvature analysis," in *Modern Geometric Computing for Visualization*, T. L. Kunii and Y. Shinagawa, eds., pp. 57–72, Springer-Verlag, 1992.
12. N. S. Sapidis and P. J. Besl, "Direct construction of polynomial surfaces from dense range images through region growing," *ACM Transactions on Graphics* **14**, pp. 171–200, Apr. 1995.
13. Y. Yu, A. Ferencz, and J. Malik, "Extracting objects from range and radiance images," *IEEE Transactions on Visualization and Computer Graphics* **7**, pp. 351–364, Oct.–Dec. 2001.
14. M. E. Rettmann, X. Han, and J. L. Prince, "Automated sulcal segmentation using watersheds on the cortical surface," *NeuroImage* **15**, pp. 329–344, Feb. 2002.
15. D. L. Page, A. Koschan, and M. Abidi, "Perception-based 3D triangle mesh segmentation using fast marching watersheds," in *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, **II**, pp. 27–32, June 2003.
16. D. D. Hoffman and W. A. Richards, "Parts of recognition," *Cognition* **18**, pp. 65–96, 1984.
17. Y. Zhang, J. K. Paik, A. Koschan, M. A. Abidi, and D. Gorsich, "A simple and efficient algorithm for part decomposition of 3d triangulated models based on curvature analysis,," in *Proceedings of the International Conference on Image Processing*, **III**, pp. 273–276, Sept. 2002.
18. R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, Addison–Wesley Publishing Company, Reading, MA, second ed., 2002.
19. G. Taubin, "Estimating the tensor of curvature of a surface from a polyhedral approximation," in *Proceedings of the Fifth International Conference on Computer Vision*, pp. 902–907, 1995.
20. D. L. Page, Y. Sun, A. Koschan, J. Paik, and M. Abidi, "Normal vector voting: Crease detection and curvature estimation on large, noisy meshes," *Graphical Models* **64**, pp. 1–31, 2003.
21. A. H. Barr, "Superquadrics and angle-preserving transformations," *IEEE Transactions on Computer Graphics and Applications* **1**, pp. 11–23, Jan. 1981.
22. A. H. Barr, "Global and local deformation of solid primitives," *Computer Graphics Proceedings* **18**(3), pp. 21–30, 1984.
23. M. E. Montiel, A. S. Aguado, and E. Zaluska, "Surface subdivision for generating superquadrics," *Visual Computer* **14**(1), pp. 1–17, 1998.
24. A. Jaklič and F. Solina, "Moments of superellipsoids and their application to range image registration," *IEEE Transactions on Systems, Man, and Cybernetics, Part B* **33**(4), pp. 648–657, 2003.
25. F. Solina and R. Bajcsy, "Recovery of parametric models from range images: The case for superquadrics with global deformations," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **12**(2), pp. 131–147, 1990.
26. A. Gross and T. Boult, "Error of fit measures for recovering parametric solids," in *Proceedings of the International Conference on Computer Vision*, pp. 690–694, 1988. Invited paper.