Unstructured Terrain Navigation and Topographic Mapping with a Low-cost Mobile Cuboid Robot

Andrew S. Morgan[1]*, Robert L. Baines[1]*, Hayley McClintock[1], and Brian Scassellati[2]

***Abstract*— Current robotic terrain mapping techniques require expensive sensor suites to construct an environmental representation. In this work, we present a cube-shaped robot that can roll through unstructured terrain and construct a detailed topographic map of the surface that it traverses in real time with low computational and monetary expense. Our approach devolves many of the complexities of locomotion and mapping to passive mechanical features. Namely, rolling movement is achieved by sequentially inflating latex bladders that are located on four sides of the robot to destabilize and tip it. Sensing is achieved via arrays of fine plastic pins that passively conform to the geometry of underlying terrain, retracting into the cube. We developed a topography by shade algorithm to process images of the displaced pins to reconstruct terrain contours and elevation. We stitch together images collected over the cube's rolling trajectory to render detailed topographic maps.**

I. INTRODUCTION

The demand for increasingly autonomous robotic systems has motivated extensive research on robot navigation and environmental mapping. One common approach, simultaneous localization and mapping (SLAM), reconstructs an environment in 3D and estimates robot pose in that environment by acquiring sensory data from on-board Lidar, cameras, GPS, or other sound and signal-based range finders [1].

Although SLAM and its variants have been shown to depict obstacles and topography with a sufficient degree of accuracy for most navigation purposes, their construction of environmental maps comes at great computational expense and ignores finer details, limiting their applications in cost-conscious or small-scale projects [2]. For instance, a 3D reconstruction of the environment is typically derived from triangular meshes which, beyond being computationally expensive to construct and store [3], [4], have hard constraints on the resolution of features they can map [5]. Additionally, the suite of sensors required for accurate topographic reconstruction ramps up the monetary expense, complexity, and fragility of a system. [5].

A compelling alternative to optical sensor- and range sensor-driven mapping techniques, like SLAM, is to leverage tactile-based sensors to map an environment through direct physical interaction. Minute features and textures, indiscernible to range-based systems, could be tracked with tactile sensing, all with lower computational and monetary expense. Examples of established tactile sensing schemes include electro-active tactile sensors, which have been used in robotic hands to classify objects of diverse shapes and sizes [15], and tactile sensors based on deforming fiducials on a compliant membrane [11], [12] or incident light changes [13], [14]. Both techniques have been shown to elicit accurate surface and texture classifications under a variety of object conditions.

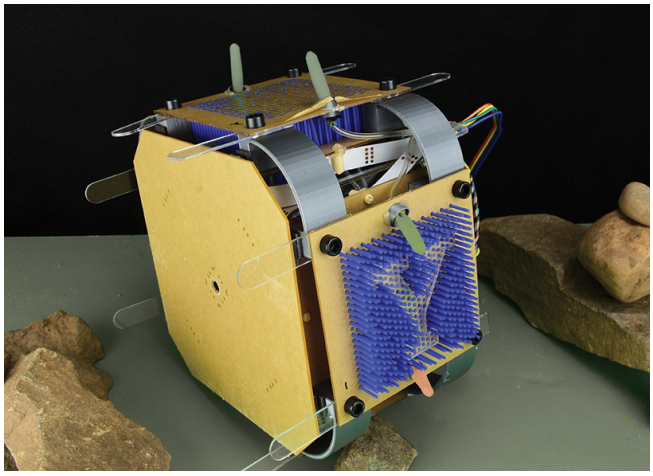

Fig. 1. A cuboid robot that can move across and map the topography of unstructured terrain. Each face of the rolling surface is equipped with pin arrays that provide sensing and inflatable bladders to facilitate locomotion. LED strips are strung throughout the robot's interior to illuminate deformed pins for the topography by shade approach.

Herein, we showcase a cuboid robot that can roll through varied terrain and construct accurate contour and elevation maps in real-time using tactile-based sensing (Figure 2). The cuboid robot moves by sequentially inflating two bladders on each of its four faces to destabilize itself and roll to a subsequent face. To sense and map the terrain over which it rolls, the robot utilizes a passive tactile-based sensing mechanism that comprises of an array of plastic pins which deform to the shape of objects underneath. On the inside of the cube, lights are flashed in a coordinate fashion on different sides of the pins while pictures are taken with an embedded wide-angle camera. Via our topography by shade algorithm, inspired by a classical computer vision techniques, shape from shading and photometric stereo [7], [9], the topography of underlying terrain is re-constructed from these

[1]Andrew Morgan, Robert Baines, and Hayley McClinktock are with the Department of Mechanical Engineering & Materials Science, Yale University, 9 Hillhouse Avenue, New Haven, CT 06520, USA. (email: Andrew.Morgan, Robert.Baines, Hayley.McClintock@yale.edu).

[2]Brian Scassellati is with the Department of Computer Science, Yale University, 51 Prospect Street, New Haven, CT 06511, USA. (email: Brain.Scassellati@yale.edu).

∗ denotes that authors contributed equally to the creation of this work.

images. Our topography by shade algorithm is discussed in detail in Section IIC.

The outline of this paper is as follows: Section II describes the hardware and software systems of the robot, Section III outlines three experiments performed on the cuboid robot to quantify its locomotion and mapping abilities, and Section IV concludes the paper with a discussion of future work.

II. Systems of the Cuboid Robot

A. Mechanical Hardware

The body of the cuboid robot (Figure 3A) is made primarily from laser cut acrylic sheets and 3D-printed brackets. Four of the six faces of the cuboid contain an array of plastic pins that are free to slide between two parallel acrylic plates, serving as the passive tactile-based sensing elements. The two remaining faces are made from opaque acrylic to block out ambient light. Blocking the light renders optimal conditions for the topography by shade algorithm, as discussed in Section III. The robot is connected to an external pneumatic line and a control PC.

Figure 3B shows an exploded view of one sensory face of the cuboid robot. The face constitutes three distinct levels. The outermost level (1) contains exposed sensory pins. The pins are routed through small holes (1.35 mm diameter) so that they are free to translate vertically, but have minimal side-to-side movement. In addition, there are two bores where press-fit 3D printed inserts hold the latex pneumatic bladders. These inserts are angled to direct the inflation trajectory of the bladders. The second level (2) has pin routing holes that provide a secondary constraint preventing lateral movement. Additionally, four holes contain press-fit 3.65 mm wooden dowels that route an addressable, flexible LED strip around the spacers between the second and third levels. The third level (3) is made from a clear acrylic sheet, which prevents the pins from falling while providing a view of the pins to the camera.

The three levels of the face are stacked and aligned using four plastic shoulder screws and 3D printed spacers. The spacers between levels one and two also serve as brackets that connect the four pin array faces together and are rounded to aid with rolling. Additional acrylic stability supports lie between the spacer and level one to prevent the cuboid robot from rolling onto its non-sensing faces, as seen in Figure 1.

The camera is fixed to a 6.35 mm wooden dowel that runs between the two pin-less faces of the robot. This design allows the camera to passively rotate about the central axis of the robot during locomotion, relying on gravity to reorient the lens toward the sensory face contacting the ground. The entire robot weights 911 g and is 20.4 cm about each of its edges. The total cost of the robot at time of construction is approximately $100.

B. Electronic Infrastructure

The cuboid robot's interior contains individually addressable LED strips (NooElec, WS2801) to illuminate the displaced pins and a wide-angle camera (ELP, megapixel Super Mini) to capture images. The remaining electronic infrastructure is off-board for the proof-of-concept prototype described in this paper. An array of 8 addressable miniature pressure regulators with built-in bang-bang control are used to control the inflation of the pneumatic bladders [6].

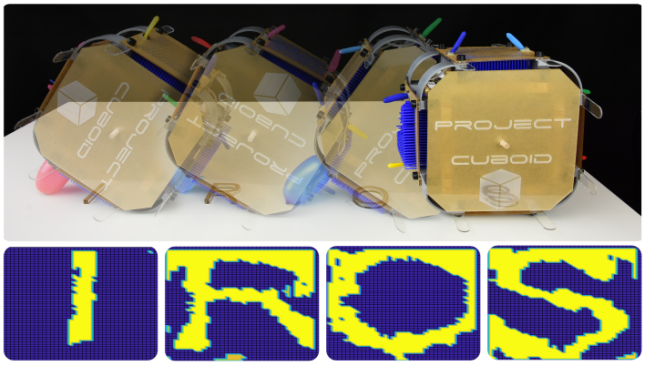

Fig. 2. The cuboid robot rolls across obstacles in the midst of performing topographical mapping. The robot is able to capture a high degree of object detail and traverse unstructured terrain.

Both the LEDs and the pressure regulators interface directly to the Arduino, which is in turn controlled by a PC via serial communications. The camera is directly connected to the PC via a USB cable.

C. Software Architecture

The software architecture (Figure 4) contains four primary ROS nodes, each with its own set of sub-modules that execute robot-specific functionality. Low-level code running on the Arduino for the I2C bus and serial LED communication is accessed by the master ROS controller. Commands from the master control node are sent to the Arduino via serial, commanding the pressure regulators and the LEDs. The camera node publishes a 640x480px image at 30Hz, which was found to be an optimal balance between precision and frame rate. In the subsequent sections, we discuss the main two nodes, control and vision processing, in detail.

Algorithm 1 Mapping Topography by Shade in Cuboid

Output: d_{face} ▷ Map for all Faces

1: **for** $face \in faces$ **do**
2: $inflateSequence(face)$
3: **for** $edge \in edges$ **do**
4: $illuminateEdge(edge)$
5: $im[edge] \leftarrow cuboidCamera()$ ▷ HSV
6: $d_{face}[face] \leftarrow blurInterpolation(im)$
7: **return** d_{face}

1) Control node: The master controller coordinates sequential tasks within the ROS communication framework by keeping a global clock and waiting for specific events to occur. Algorithm 1 presents the outline of this sequence. After the cuboid robot rolls onto a face via the *inflateSequence(face)* method, which commands the addressable pressure regulators to inflate and deflate systematically for

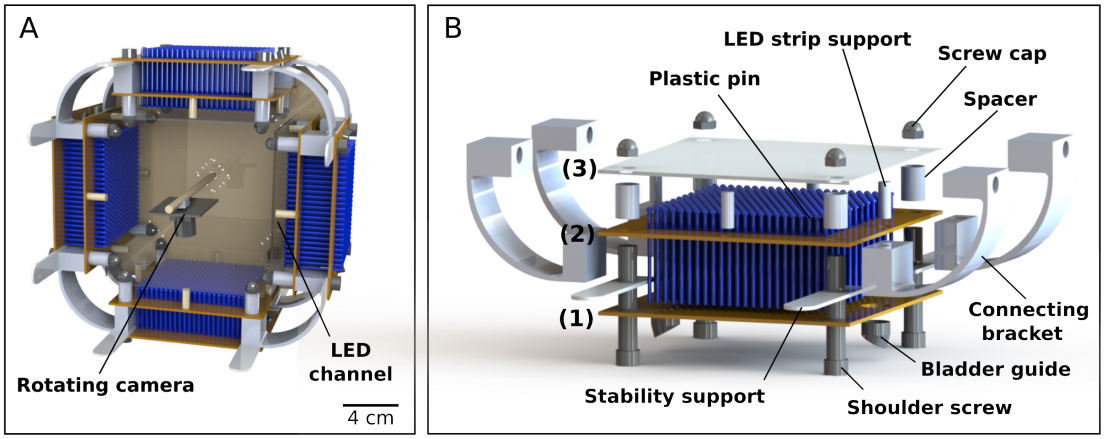

Fig. 3. Schematic of the cuboid robot. (a) Open side view, including the camera and camera mounting system (center of the robot on a dowel rod) (b) Exploded view of one face. Labels (1), (2), and (3) denote the outer, middle, and inner laser cut acrylic levels, respectively.

a desired face, the controller initiates a hard-coded illumination sequence for internal LED sections in *illuminateEdge(edge)*. During the illumination process, the master controller captures hue saturation value (HSV) images from *cuboidCamera()* when each of the four edges are illuminated individually. The four images are then sent through the *blurInterpolation(im)* method for further processing, as described in a subsequent section. This process continues until a predefined halt is triggered in its operation, presenting the final topographic map, d_{face}, for the user.

Algorithm 2 blurInterpolation() Method

Input: im ▷ Four Images in HSV
Output: m_{face} ▷ Topographical Map of Face
1: $imAvg \leftarrow averageImgs(im)$ ▷ Avg Data
2: $datBlur \leftarrow blurNeighbors(imgAvg)$
3: $datFilt \leftarrow rmvPeakExtr(dataBlur)$
4: $valleys \leftarrow findValleys(datFilt)$
5: $peaks \leftarrow findPeaks(datFilt)$
6: $m_{face} \leftarrow interpolatePeaks(peaks, valleys)$
7: **return** m_{face}

2) Vision processor node and "topography by shade" algorithm: The vision processing node receives and processes HSV images provided by the camera node. It also implements the topography by shade algorithm, which reconstructs the terrain under the displaced pins. The algorithm is inspired by the early computer vision technique "Shape from Shading", which initially used the 2D intensity map of an image to reconstruct its 3D geometry [7]. Several variations and extensions on the original technique have been proposed, insipiring our topography by shade algorithm [16].

Our topograpahy by shade algorithm draws inspiration from yet another early computer vision technique from the late 1980s, photometric stereo, which was used to determine an object's surface orientation through incident illumination of the object over successive images, holding the viewing angle constant [9]. Although current, more sophisticated photometric techniques exist for entire 3D surface reconstruction, such as accounting for arbitrarily positioned lights and multiple camera locations [8], our adaptation leverages simplifying assumptions to reduce computation time.

Like shape from shading, we assume that the reflected light intensity of some terrain manifested in the pins is a function of its proximity to a viewing camera and the angle of the light source on the object. We do not consider the material reflectivity function and hold the position of the light source relative to the deformation of interest constant. As in classical photometric stereo, we assume the image projection to the camera is orthographic (due to mechanical constraints in our system).

To illuminate the displaced pins, we use four equal-intensity light sources symmetrically located around the sensor area. This is accomplished by individually addressing sections of the continuous LED strands around the perimeter of the pins on each of the cuboid faces. We capture images of the displaced pins in each lighting condition, convert those images to HSV format, and average them. From the average image, we are able to easily extract a relative elevation profile because elevated areas are more illuminated in comparison to lower areas.

The algorithm for transforming face images to a topographical map is summarized in Algorithm 2. First, we average the images into one composite via *averageImgs(im)* and use a Guassian blurring kernel to reduce high frequency noise in the image via *blurNeighbors(imgAvg)*. Then, we filter out lone extrema values in our data through *rmvPeakExtr(datBlur)* and find general valleys and peaks in our data through the *findValleys(datFilt)* and *findPeaks(datFilt)* methods, respectively. Finally, we interpolate between these peaks and valleys, partitioning object surface from baseline in the *interpolatePeaks(peaks,valleys)* method to render our final map.

III. Experiments and Results

We performed a number of tests on the cuboid robot to validate its efficacy in moving across and mapping unstruc-

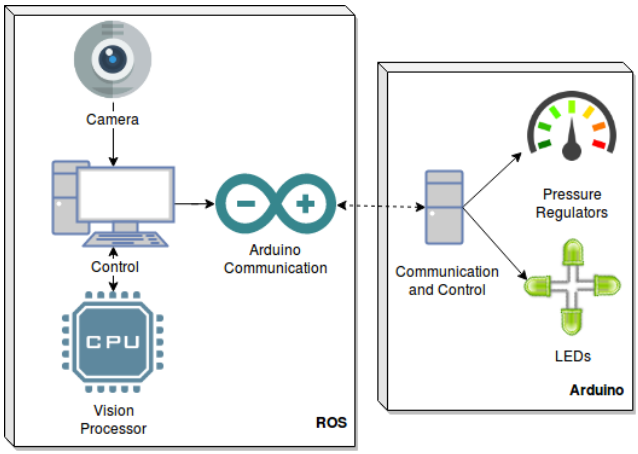

Fig. 4. The cuboid robot's software architecture comprises a collection of ROS nodes which interface via serial to an Arduino Mega microcontroller, which executes lower-level controls. The PC-based control node receives and processes images of the topographical data. Pre-specified commands are sent to the Arduino to activate rolling capabilities and lighting sequences.

tured terrain:

1) A characterization of the cube's actuator volume to tip angle relationship and determination of maximum environmental slope for locomotion.
2) The energetic cost of transport for the robot up different grades.
3) The robot's capacity to accurately distinguish topography in two lighting conditions and to discern heights of different objects.
4) Assessment of the robot's efficacy in simultaneously rolling and mapping topography.

A. Analysis of cuboid locomotion capabilities

To test the robot's capability to navigate across varying terrain, we characterized the minimum volume input to the pneumatic bladders, V, required for it to move up an angular grade, α, spanning from $\alpha = 0 - 9°$. The analysis was conducted in a quasi-static manner, namely, the bladders were slowly inflated with precision syringes to not impart significant angular momentum on the cube. In addition to tracking the volume of air input to the bladders, we monitored the pressure of the system using a pressure sensor in parallel to pneumatic source (Honeywell, ASDXRRX030PDAA5).

Figure 5 depicts the minimum total input volume of the two inflatable bladders that statically destabilized the cuboid about its center of mass, allowing it to roll. Expectantly, for a higher the grade angle, a greater volume had to be input to the bladders for the cube to tip.

As illustrated in the schematic at the top of the Figure 5 and as evidenced in the supplementary video, the second inflatable bladder (closest to the contact of rotation) did not need to inflate as much as the first bladder due to its proximity to the pivot point. In fact, the volume of the second bladder never reached above 80 mL, including that of our highest grade, 9°.

We attribute the variance in minimum input volume between sides to subtly different alignment angles of the bladders with respect to the cuboid chassis. A possible remedy would be to ensure the bladder guides (Figure 3B) are all press-fit into the hole in exactly the same orientation. An additional factor that contributed to variance was likely viscoelastic effects present in each latex bladder. Since each bladder was subjected to different pressures, external forces from terrain, and was actuated a number of times through the course of our tests, they each experienced different amounts of strain relaxation and hysteresis. Thus, their inflation trajectories became less productive in generating motion (*i.e.* bent outward instead of directly downward) for the same volume input.

In practice, *a priori* knowledge of maximum grade angle across a swath of terrain would indicate the baseline required bladder inflation volume to traverse that terrain. One could leverage the angular momentum of the cube system by delivering short bursts of air to the bladders and tip with far less volume input.

B. Robot cost of transport

We calculated the total energy input required to tip the cuboid to calculate its cost of transport (COT), a dimensionless metric often used as a comparison between robot locomotion efficiency and that of humans. COT has a lower bound of 0 and an upper bound of infinity; a lower numbers signifies a more efficient system. It is well-established that the COT of a human is $\approx 0.2 - 0.4$. For comparison, the COT of the Honda humanoid robot, Assimo, is ≈ 3.2 [17], [18]. The COT of the cuboid robot can be expressed as:

$$COT = \frac{E_{input}}{mgd} \tag{1}$$

where E_{input} is the input energy required to roll the system to another face, m is the mass of the system, g is acceleration due to gravity, and d is the total lateral distance covered during one step of locomotion, also known as stride (equal to the length of the edge of the robot, since it is a cube). The amount of energy required for one stride is that which moves the cuboid to the point of static instability (for an ideal cube, this is any value past 45° tip). Assuming ideal gas relations, the total input energy from pneumatic airlines to the robot is defined by:

$$E_{input} = -P\Delta V \tag{2}$$

Assuming the rotational axis to be a point at the nearest corner of the cuboid to the tip, and assuming pure rotation, we can solve for the relationship between COT and V for the cuboid which follows the form:

$$COT = 8.473V - 0.011 \tag{3}$$

Therefore, the COT for traversing an incline of $4.4°$ is 1.13, where the COT increases to 1.89 at grade of $9°$. It is also interesting to note the theoretical COT relationship

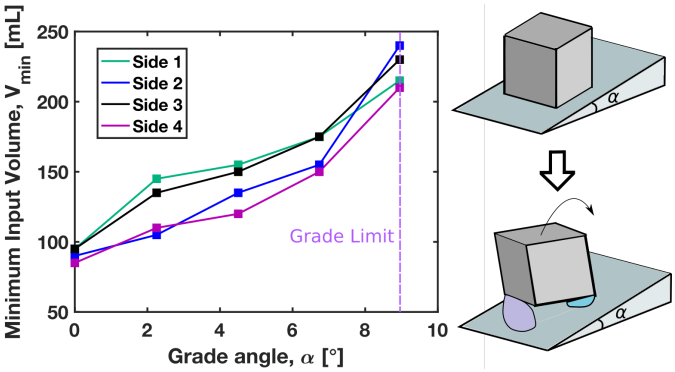

Fig. 5. The minimum volume input required to traverse sloped terrain of angle α. The plotted value denotes the sum of volume required for both balloons combined, where each color denotes a different face of the cuboid. The cuboid reaches a hard limit at 9° where the balloons cannot inflate enough to induce a tip.

for an n-sided polygon robot system with similar mass and radius, r, which can be represented as:

$$COT = \frac{-P\Delta V}{mgr\sqrt{2-2cos\frac{2\pi}{n}}} \tag{4}$$

The slope of the COT curve with respect to volume increases when additional sides are added. However, intuitively, the shorter stride length associated with a polygon with more sides necessarily requires less initial input volume to roll.

C. Characterization of topography by shade algorithm

We executed our topography by shade algorithm in two distinct lightings to ascertain its transfer-ability to different environmental conditions representative of real-world scenarios. The first lighting condition was total darkness, which we hypothesized would facilitate the most accurate topographic reconstruction due to minimal external light noise sources. The second lighting condition was ambient room lighting. In each lighting condition, three distinctly shaped objects were assessed. From top to bottom in Figure 6A: a hexagonal nut (18 mm dia.), multiple dice (16 mm width), and a small screwdriver (20 mm dia.), for a total of 6 independent tests (Figure 6A). As a metric of accuracy, we observed the reconstruction based on its ability to capture contours, internal geometry such as holes, and continuous surfaces and boundaries of the tested objects.

As predicted, the topography by shade algorithm performed most accurately in total darkness, where only directional, controlled lighting patterns influenced topographic shading of the deformed pin array. In evaluating the hexagonal nut, we note that the right upper-quadrant of its top surface is less pronounced in ambient lighting. The blob appearing in the top left of both pictures was found to be a cluster of stray pins sticking up (without any object underneath). This finding indicates that the topography by shade approach is heavily influenced by the quality and resolution of the mechanical deforming pins. Over the other two objects, there was no marked differences observed between the lighting conditions, suggesting our sensing and algorithmic approach is robust to various types of lighting.

Our second test for robustness sought to evaluate how well the algorithm projects surface heights from objects of different geometries. We found that we can detect correct relative elevations. In Figure 6B, we show a particular surface plot that was rendered in ambient light. The objects reconstructed were a single die (16 mm tall) and a flat fish-shaped cutout (9 mm tall).

We found that the intensity of the internal LED light drastically influenced correct elevation segmentation; even more so than harsh ambient lighting. By calibrating the internal LEDs appropriately, we were able to mitigate effects of internal reflections on the acrylic from the LEDs. The importance of considering internal distribution of light, as well as external lighting in cuboid operation conditions is thus underscored.

D. A robot obstacle course

As a final assessment of the cuboid robot, we constructed an obstacle course, consisting of sequences of uniquely shaped objects. Each object– a metal cylindrical rod, several nuts, a wrench, a 3D-printed L-slot bracket, a small screwdriver, and a die–was adhered to the flat test bench. The metal rod and the nut were of the same vertical height. All other grouped objects were of disparate heights. We commanded the cuboid to roll across the terrain and reconstruct a topographic map. We evaluated the reconstructed map based on ground truth images of the data and prior knowledge of their relative positions and elevations.

Figure 6C showcases the actual topographic features (left), the raw camera image of the deformed pin array with manually sketched object contours (middle), and those perceived by the cuboid robot (right). As a whole, the reconstruction qualitatively resembles the ground-truth objects fairly well. Contours are generally correct, and the reconstruction captures both internal and external geometric features like the center of the nut and the slot in the L-bracket. Distinct elevations (illustrated by the different colors of objects in the reconstruction map) were all identified successfully, save the second frame from top, in which the algorithm incorrectly put the small nut and wrench at the same elevation. We suspect this is the case because the objects did not have a large enough height difference to be appropriately segregated.

Marked deviation from ground-truth topography occurs about the periphery of the cuboid's captured image, probably resulting from glare from nearby LEDs compounded by an imperfect alignment angle of the cube on top of the terrain. Additionally, some artifacts present in the reconstruction were the fault of the pin mechanism in that a few of its pins got stuck and persisted throughout the rolling duration, falsely indicating the presence of an object (as in Figure 6A top). We reiterate that the resolution of the map is restricted by that of the diameter of the pin heads. A higher resolution pin array would likely result in higher fidelity maps. We will leave increased resolution mapping as future work.

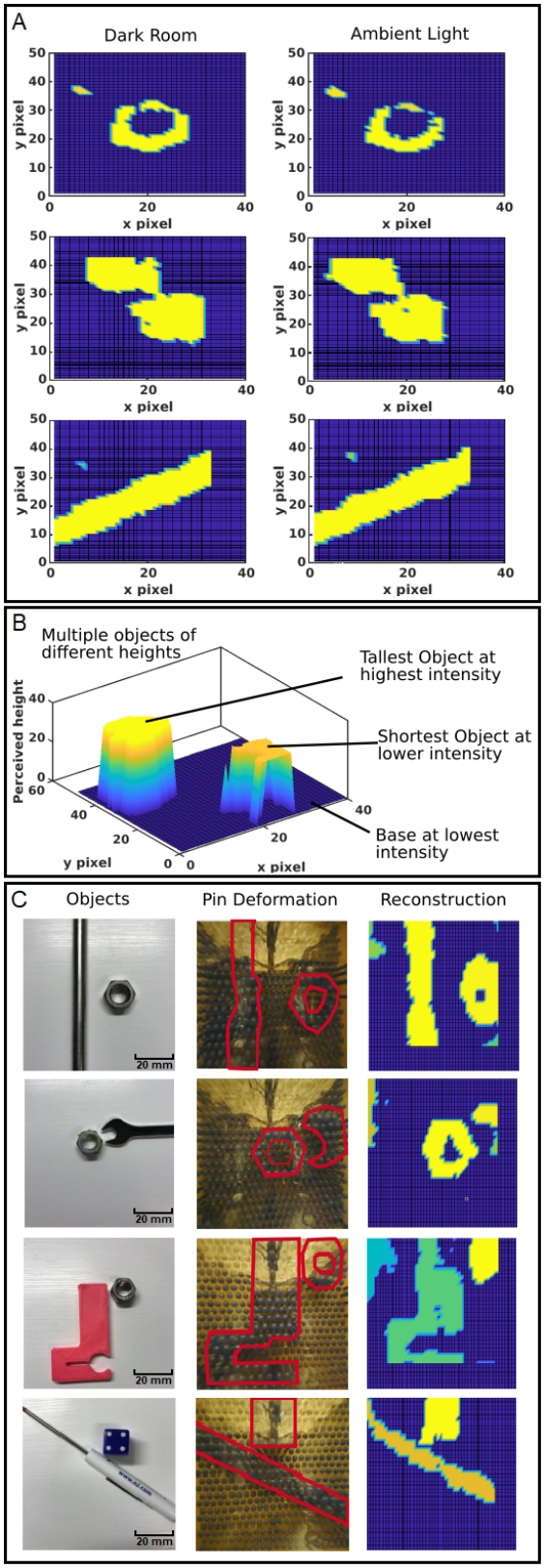

Fig. 6. (A) ambient light test for single hexagonal nut, two die, and one small screwdriver. (B) map constructed from a single die and a fish shaped cutout, showing our algorithm's ability to parse relative elevations correctly. (C) example mapping of multiple objects from an obstacle within the same frame. The algorithm is able to discern unique contours, as well as elevations of objects in close proximity (different elevations are denoted by colors of shading in the Reconstruction column).

IV. CONCLUSIONS

This study presented a cuboid robot which simultaneously moves across and maps diverse terrain. The robot consists of commercially available, resilient, inexpensive components, and low-cost algorithms, in contrast with other mapping robots that utilize suites of complex sensors. Shortcomings of the robot include the propensity of the mechanical pins to stay stuck and the resolution and size of the features that can be mapped due to the size of the pin face. We intend to address these challenges in a future iteration of the robot design.

Due to its low-cost, we envision that the presented design could be used to make swarms of robots that collaboratively construct detailed topographic maps where they are deployed, or in robotic applications that are subject to confined spaces with low power resources. The sensing apparatus and algorithms developed for the cuboid are transferable to larger scale robotic systems, too. We foresee that the tactile-based mapping strategy, topography by shade, will be a logical supplement to optical and range-finding sensors for next-generation mobile robots, giving them heightened feature identification and reconstruction capabilities.

ACKNOWLEDGMENT

The authors would like to thank Sarah Sebo for her mentorship throughout the project. This work was funded by the Department of Computer Science at Yale University.

REFERENCES

[1] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit, FastSLAM: A Factored Solution to the Simultaneous Localization and Mapping Problem, In. Proc. Eighteenth National Confernce on Artificial Intelligence, page 593-598, Aug. 2002.

[2] R. Triebel, P. Pfaff, and W. Burgard, Multi-Level Surface Maps for Outdoor Terrain Mapping and Loop Closing, in 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, Beijing, China, 2006, pp. 22762282.

[3] M. Levoy, K. Pulli, B. Curless, S. Rusinkiewicz, D. Koller, L. Pereira, M. Ginzton, S. Anderson, J. Davis, J. Ginsberg, J. Shade, and D. Fulk. The digital michelangelo project: 3D scanning of large statues. In Proc. SIGGRAPH, pages 131144, 2000.

[4] C.F. Olson. Probabilistic self-localization for mobile robots. IEEE Transactions on Robotics and Automation, 16(1):5566, 2000.

[5] S. Thrun, Robotic Mapping: A Survey, p. 31.

[6] J. W. Booth, J. C. Case, E. L. White, D. S. Shah, and R. Kramer-Bottiglio, An addressable pneumatic regulator for distributed control of soft robots, in 2018 IEEE International Conference on Soft Robotics (RoboSoft), Livorno, 2018, pp. 2530.

[7] B. K. P. Horn, Obtaining Shape from Shading Information, Psychology of Computer Vision. p. 42. 1975.

[8] H. Jin, D. Cremers, D. Wang, E. Prados, A. Yezzi, and S. Soatto, 3-D Reconstruction of Shaded Objects from Multiple Images Under Unknown Illumination, International Journal of Computer Vision, vol. 76, no. 3, pp. 245256, Mar. 2008.

[9] R. J. Woodham. "Photometric method for determining surface orientatoin from multiple images," Optical Engineering, vol. 19, no 1, pp. 139-144, Jan. 1980.

[10] Biomimetic and biohybrid systems: Second International Conference, Living Machines 2013, London, UK, July 29 - August 2, 2013. Proceedings, 1st edition. New York: Springer, 2013.

[11] A. Yamaguchi and C. G. Atkeson, Optical Skin For Robots: Tactile Sensing And Whole-Body Vision, RSS17 Workshop on Tactile Sensing for Manipulation: Hardware, Modeling, and Learning, Cambridge, MA, USA. p. 6.

[12] K. Kamiyama, H. Kajimoto, N. Kawakami and S. Tachi, Evaluation of a vision-based tactile sensor, IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004, New Orleans, LA, USA, 2004, pp. 1542-1547 Vol.2.
[13] B. McInroe, C. Chen, K. Goldberg, R. Bajcsy, and R. Fearing, Towards a Soft Fingertip with Integrated Sensing and Actuation, p. 8.
[14] H. Maekawa, K. Tanie, K. Komoriya, M. Kaneko, C. Horiguchi and T. Sugawara, "Development of a finger-shaped tactile sensor and its evaluation by active touch," Proceedings 1992 IEEE International Conference on Robotics and Automation, Nice, France, 1992, pp. 1327-1334 vol.2.
[15] H. Yousef, M. Boukallel, and K. Althoefer, Tactile sensing for dexterous in-hand manipulation in roboticsA review, Sensors and Actuators A: Physical, vol. 167, no. 2, pp. 171187, Jun. 2011.
[16] R. Zhang, P. Tsai, J. E. Cryer, and M. Shah, Shape-from-shading: a survey, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 21, no. 8, pp. 690706, Aug. 1999.
[17] S. Collins, A. Ruina, R. Tedrake, and M. Wisse, Efficient Bipedal Robots Based on Passive-Dynamic Walkers, Science, vol. 307 5712, pp. 10821085, 2005.
[18] J. M. Donelan, R. Kram, and A. D. Kuo, Mechanical work for step-to-step transitions is a major determinant of the metabolic cost of human walking, Journal of Experimental Biology, vol. 205, pp. 3717 - 3727, Dec. 2002.