

# Bio-Inspired Rough Terrain Contact Patch Perception

Dimitrios Kanoulas and Marsette Vona

**Abstract**—We present a new bio-inspired system for automatically finding foot-scale curved surface patches in rough rocky terrain. These patches are intended to provide a reasonable set of choices for higher-level footfall selection algorithms, and are pre-filtered for several attributes—including location, curvature, and normal—that we observed humans to prefer. Input is from a  $640 \times 480$  depth camera augmented with a 9-DoF inertial measurement unit to sense the direction of gravity. The system is capable of finding approximately 700 patches/second on commodity hardware, though the intention is not to find as many patches as possible but to reasonably sample upcoming terrain with quality patches. Sixty recordings of human subjects traversing rocky trails were analyzed to give a baseline for target patch properties. While the presented system is not designed to select a single foothold, it does find a set of patches for possible footholds which are statistically similar to the patches humans select.

## I. INTRODUCTION

Recent advancements in control and perception have enabled bipeds to walk on flat [1] and uneven indoor terrains [2]. Major advances have also been made for outdoor quadrupeds in rough terrain where the probability of blindly landing footholds is high [3]. Online footfall selection has been considered for quadrupeds and hexapods [4]–[8]. Still, attaining animal-like legged locomotion on rough outdoor terrain with sparse foothold affordances—arguably a primary use-case for legs vs other forms of locomotion—still remains a largely open problem. New advances in both perception and control will be required; here we attempt to disentangle these two aspects to a degree and focus on perception.

The foothold selection problem is particularly interesting for bipeds with non-point feet that make contact with patches of terrain. In prior work [9], [10] we presented algorithms for modeling, fitting, and perceptually validating curved surface patches that can succinctly model terrain at foot scale. Here we use paraboloids with elliptic, circular, or rectangular boundaries (Fig. 1). Our new contribution is a real-time bio-inspired system for automatically finding and fitting *salient* patches. We take data from a forward-facing depth camera<sup>1</sup> augmented with an inertial measurement unit (IMU) that measures the gravity direction (Fig. 4).

In this work we manually carry the sensor over the terrain to enable study of perception algorithms in isolation from control, though we acknowledge that the two cannot be completely decoupled. The intention of this system is to

condense and sparsify bulk point-cloud data from a depth camera (here up to 12MB/sec) into a higher-level form, about 700 patches (roughly 23kB) per second. These constitute a sampling of upcoming terrain from which a footfall selection algorithm, considering task-specific dynamics and control, could then make its choice. (Because the patches do not sample e.g. vertical surfaces, swing leg motion planning could use a supplemental octree or other representation for collision avoidance.)

A key aspect of our approach is that we do not just fit as many patches as possible, but instead attempt to balance patch quality with sufficient sampling of the appropriate parts of upcoming terrain. We use four saliency measures at different stages for this purpose, involving aspects of patch orientation, curvatures, and location. Our approach is bio-inspired both in that one of these relates to a known biomechanical property—humans tend to fixate about two steps ahead in rough terrain [11]—and also because we used observations of the patches humans actually select as a baseline for setting parameters of the measures.

The system is implemented in C++ using the Point Cloud Library [12], and the source code is available on our website [13]. All datasets are available upon request.

Next we review the research context, followed by a summary of our patch models and the bio-inspired measures we use to determine patch saliency. We then present our real-time algorithm for finding, fitting, and validating salient patches. Finally we describe our method for observing patches selected by human subjects and compare statistics of these with automatically selected patches.

### A. Related Work

To date only a few works have used on-line perception for bipedal foot placement in uneven or rough terrain. In [14] planar segments were fitted to point cloud data for indoor scenes with slopes and steps, and in [2] a laser scanning system is used in a similar context. In [15] KinectFusion [16] was used in simulation to avoid harsh impacts. A number of other works (e.g. [17]) introduce perception mainly for obstacle detection but not 3D foot placement.

On-line perception for foot placement has also been implemented for quadrupeds and hexapods [4]–[8]. In some cases foot placement has been done without perception by using a known 3D terrain map and on-line motion capture (e.g. [18]). It is also common here to use perception for obstacle avoidance, terrain categorization, or gait selection without specific 3D foot placement [19], [20]. Quadrupedal and hexapedal studies are related to the bipedal case but often use a point-like contact model, whereas many bipeds have

The authors are with the College of Computer and Information Science, Northeastern University, Boston, Massachusetts dkanou@ccs.neu.edu, vona@ccs.neu.edu

<sup>1</sup>We use a Kinect structured light depth camera with  $640 \times 480$  resolution. While this device does not work well in full sunlight, it can be used outdoors in the shade or at twilight. The IMU is a CH Robotics UM6.

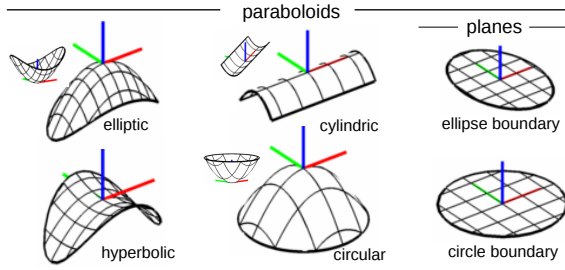


Fig. 1. Paraboloid patch types and local coordinate frames.

extended feet to support torques for balance and may need to consider foot-sized terrain patches.

Research on human locomotion shows that visual information is crucial when walking on uneven terrain [11], [21]–[25]), but so far only a few works (e.g. [26]) have specifically applied this to perception for bipedal robots.

## II. PATCH MODELING AND SALIENCY

In [9], [10] we introduced a set of bounded curved surface patch models and algorithms to fit and validate them to points sampling environment surfaces. Here we use a subset of these, the paraboloids (Fig. 1), which balance expressiveness with compactness of representation.<sup>2</sup> Various choices are possible with regard to the size and shape of the patch boundary curves, which can be adjusted in applications depending on the geometry of foot contact surfaces and the different contact configurations a particular footfall selection algorithm might consider. Here we use a generic approach with rectangle bounds for cylindric paraboloids and ellipse (possibly degenerating to circular) bounds otherwise, and we fit the bounds to probabilistically contain 95% of the points in a spherical foot-scaled neighborhood with  $r = 10\text{cm}$  [9].

Each patch has two intrinsic parameters, the principal curvatures  $\kappa_{x,y}$ , and six extrinsic parameters  $\mathbf{r} \in \mathbb{R}^3$ ,  $\mathbf{t} \in \mathbb{R}^3$  giving the spatial pose of a local coordinate frame attached to the patch. Eq. 1 gives an implicit form for the patch surface in local frame where  $\mathbf{p}_1 \in \mathbb{R}^3$  is a point on the patch, see [9] for details.

$$\mathbf{p}_1^T \text{diag}(\kappa_x, \kappa_y, 0) \mathbf{p}_1 - 2\mathbf{p}_1^T [0 \ 0 \ 1]^T = 0 \quad (1)$$

Using 4-byte floats for the parameters requires 32 bytes per patch,<sup>3</sup> highlighting the compression of information from a dense  $640 \times 480$  depth image (about 0.4MB with the Kinect’s 11 bit depth resolution) to a corresponding sparse representation using e.g. 100 patches (about 3kB).

### A. Saliency

The term saliency has been used in computer graphics (e.g. [27], [28]) to describe parts of surfaces that seem perceptually important to humans. Often these are essentially locations of curvature extrema. Such a definition may also be relevant here, as humans do sometimes step on e.g. the peak of a rock. However, this seems relatively uncommon. We thus introduce four new measures of saliency that relate to patches that humans commonly select for stepping.

<sup>2</sup>Modeling patches using paraboloids and other design choices are described in more detail in [9].

<sup>3</sup>The boundary parameters are fixed here.

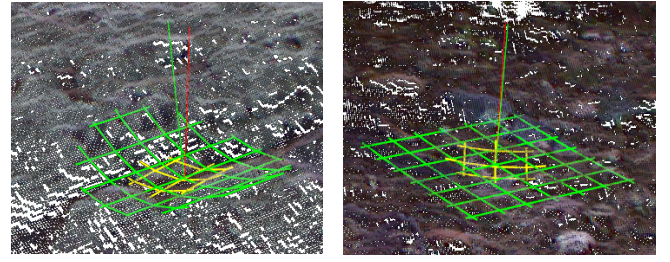


Fig. 2. Illustration of the difference-of-normals measure.

*Difference of Normals (DoN):* The difference of normals operator was introduced in [29] as the angle between the normals of fine scale vs coarse scale neighborhoods of a point (Fig. 2). This value relates to the irregularity of the surface around the point, and also to the local uniqueness of the point (following the same idea as the difference of Gaussians operator in 2D images). We conjectured that points with *low* DoN may be salient for the purpose of footfall selection. The coarse scale neighborhoods we use are of radius  $r = 10\text{cm}$  and the fine scale are  $r/2$  (for this and the next measure square neighborhoods are actually used to enable fast normal computation with integral images, see Section III).

*Difference of Normal-Gravity (DoNG):* The angle between the  $r$ -neighborhood normal vector of each point and the reverse of the gravity vector  $-\hat{\mathbf{g}}$  (from the IMU,  $\hat{\mathbf{g}}$  points down) gives a measure of the slope of that area. For fairly obvious reasons, points with low DoNG can be considered more salient for footfall selection.

*Distance to Fixation Point (DtFP):* Various publications on vision for human locomotion (e.g. [11]) find that humans fixate approximately two steps ahead when locomoting in rough terrain. We thus estimate a spatial fixation point near the ground approximately two steps ahead of the current position. We define points with smaller Euclidean distance from the fixation point to have higher saliency.

*Minimum and Maximum Principal Curvature:* The smaller of the two principal curvatures  $\kappa_{\min} \triangleq \min(\kappa_x, \kappa_y)$  at a point is the inverse of the radius of the smallest osculating circle tangent to the surface there; similarly the largest osculating circle has radius  $1/\kappa_{\max}$ . The signs of the principal curvatures also indicate whether the surface is concave (both positive), convex (both negative), or saddle (opposite signs) at that point. These values can be used in a few different ways depending on the shape of the robot foot. For example, for a flat footed robot (or to a rough approximation, for a human wearing a hiking boot), concave regions with more than slightly positive  $\kappa_{\max}$  could be considered less salient, because the foot can’t fully fit there. A robot with spherical feet might prefer areas that are not too convex (as the foot would only make contact at a tangent point) but also not too concave to fit.

We present algorithms for calculating these four measures next. The first three can be calculated quickly for all points and so are useful to identify good patch neighborhoods before fitting. We apply the fourth later when curvatures are available after patch fitting. (We are not aware of a method

for finding true principal curvatures, short of fitting patches, that is as fast as we would like.)

### III. ALGORITHM

The algorithm proceeds in four stages (Fig. 3): (I) preprocessing; (II) DtFP saliency; (III) DoN and DoNG saliency; (IV) patch fitting, curvature saliency, and postprocessing.

#### Stage I: Input and preprocessing.

*Step 1:* Receive  $640 \times 480$  image  $Z$  from the depth camera and absolute orientation quaternion  $\hat{\mathbf{q}}$  from the IMU.

*Step 2:* Convert  $Z$  to an organized<sup>4</sup> point cloud  $C$  in camera frame and  $\hat{\mathbf{q}}$  to a unit gravity vector  $\hat{\mathbf{g}}$  pointing down in camera frame.

*Step 3:* Apply a discontinuity-preserving bilateral filter to  $C$  to reduce noise effects [30].

*Step 4:* Create a new  $320 \times 240$  organized point cloud  $D$  by downsampling  $C$  with a  $2 \times 2$  median filter kernel.

**Stage II:** DtFP saliency. Parameters  $l_d = 1\text{m}$ ,  $l_f = 1.2\text{m}$  are the distances down and forward from the camera to the estimated fixation point ( $l_d$  is the approximate height at which we held the camera;  $l_f$  is an approximation of two human step lengths [25], minus the approximate distance from the body to the camera as we held it); parameter  $R = 0.7\text{m}$  can be adjusted to capture the ground area to be sampled for upcoming steps.

*Step 5:* Estimate the fixation point  $\mathbf{f}$  in camera frame

$$\mathbf{f} \triangleq l_d \hat{\mathbf{g}} + l_f ([1 \ 0 \ 0]^T \times \hat{\mathbf{g}})$$

using the properties that  $\hat{\mathbf{g}}$  points down and  $[1 \ 0 \ 0]^T$  points right in camera frame.

*Step 6:* Initialize seed points  $S$  as all points in  $D$  within an  $R$ -ball region of interest of  $\mathbf{f}$ .

**Stage III:** DoN and DoNG saliency. Parameter  $r = 10\text{cm}$  is the patch neighborhood radius, which can be adjusted to match foot contact geometry;  $f = (525/2)\text{pixels}$  is the focal length of the depth camera at the downsampled  $320 \times 240$  resolution;  $\phi_d = 15^\circ$  and  $\phi_g = 35^\circ$  are DoN and DoNG angle thresholds estimated from human-selected patches (Section IV-B).

*Step 7:* Compute  $320 \times 240$  surface normals  $N, N_s$  corresponding to  $D$  using integral images [31]. The normal  $N(i)$  uses window size  $2rf/Z(i)$  where  $Z(i)$  is the  $z$  coordinate (depth) of point  $i$  in camera frame, and  $N_s(i)$  uses window size  $rf/Z(i)$ .

*Step 8:* Remove from  $S$  all points  $i$  for which

$$N(i)^T N_s(i) < \cos(\phi_d).$$

*Step 9:* Remove from  $S$  all points  $i$  for which

$$-N(i)^T \hat{\mathbf{g}} < \cos(\phi_g).$$

The same integral image algorithm used for fast normal estimation can also produce “surface variation” values [32] which are often related to local curvature, but this relation depends on the input and is not guaranteed. We thus defer

curvature saliency until after patch fitting, which does give estimates of the true principal curvatures.

**Stage IV:** Patch fitting, curvature saliency, and postprocessing. Parameter  $n_s = 100$  is the maximum number of seed points and can be adjusted to balance coverage and patch overlap in relation to patch and region of interest size;  $n_f = 50$  is the maximum neighborhood size for patch fitting, which can also be adjusted depending on patch size;  $\kappa_{\min} = -13.6\text{m}^{-1}$ ,  $\kappa_{\max} = 19.7\text{m}^{-1}$  are the min and max principal curvatures estimated from human selected patches (Section IV-B);  $d_{\max} = 0.01\text{m}$  is the maximum RMS Euclidean patch residual.

*Step 10:* If  $|S| > n_s$  then discard  $|S| - n_s$  points randomly from  $S$ . We experimented with a non-maximum suppression algorithm [33] instead of random subsampling, using a weighted average of the DoN and DoNG angles. However the results were not clearly preferable.

*Step 11:* Use an organized search to find a neighborhood  $H(i)$  with at most  $n_f$  points from  $C$  randomly distributed within an  $r$  ball of each seed  $S(i)$ . In [10] we studied different neighborhood methods including breadth-first search on a triangle mesh and K-D tree spatial search. Here we used the organized point cloud search method in PCL which is optimized using back-projections to the image plane.

*Step 12:* Fit a patch  $P(i)$  to each neighborhood  $H(i)$ . Our fitting algorithm [9] uses a Levenberg-Marquardt nonlinear iteration to minimize an algebraic surface residual. It can accommodate covariance matrices for input and output uncertainty, but we don’t use that feature here. The boundary curve is fit probabilistically (using moments) to include 95% of  $H(i)$  projected on the patch local frame  $xy$  plane.

*Step 13:* Discard patches with min principal curvature less than  $\kappa_{\min}$  or max principal curvature greater than  $\kappa_{\max}$  (curvature saliency). This step could be adjusted depending on the application.

*Step 14:* Compute Euclidean patch residual [10], [34] and discard patches where this is greater than  $d_{\max}$ .

*Step 15:* Our patch coverage algorithm [10] could also be applied to discard patches with areas not sufficiently supported by data, but we did not integrate it here yet.

#### A. Time Complexity and Runtime

Stages I-III are  $O(|Z|)$ , i.e. linear in the input. The runtime of stage IV is dominated by  $O(n_s n_f^2)$  for step 12; steps 11<sup>5</sup>, 14, and 15 are  $O(n_s n_f)$ . The worst case time complexity for the whole algorithm is thus  $O(|Z| + n_s n_f^2)$ .

In practice on commodity hardware (one 2.50GHz core, 8GB RAM) the bilateral filter and downsampling (stage I) run in  $\sim 20\text{ms}$  total. Normal computation, DtFP, DoN, and DoNG saliency in stages II and III take  $\sim 35\text{ms}$  combined, dominated by  $\sim 30\text{ms}$  for integral image computation (the main reason for downsampling is that the required integral

<sup>5</sup>The implementation of step 11 is actually  $O(n_s r^2)$ , but could be improved to  $O(n_s n_f)$  by switching to breadth-first search on a triangle mesh.

<sup>4</sup>Organized points have a 1:1 correspondence to an  $M \times N$  image.

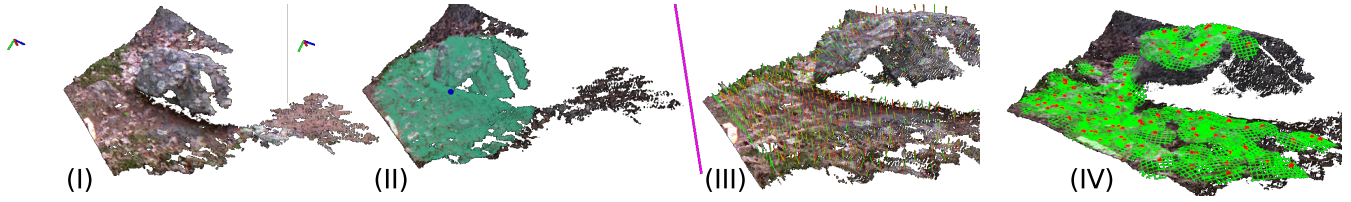


Fig. 3. Algorithm overview: (I) dense point cloud input; (II) candidate seeds (green) near an estimated fixation point (blue); (III) dense normals are computed and used for saliency measures along with the IMU-derived gravity direction (purple); (IV) 100 patches are fit to a random subsampling of salient seeds (red) and are validated for quality of fit and acceptable curvature.

images take  $\sim 150\text{ms}$  at  $640 \times 480$  [31]). Neighborhood finding in stage IV takes  $\sim 0.03\text{ms}$  per seed, and patch fitting and Euclidean residual are  $\sim 0.8\text{ms}$  total per neighborhood with  $n_f = 50$ . For  $n_s = 100$  stage IV thus takes  $\sim 83\text{ms}$  total, and the full algorithm is  $\sim 20 + 35 + 83 = 138\text{ms}$  per frame.

#### IV. HUMAN SUBJECT DATA

We measured the patches that humans actually selected on several sections of rocky trail to establish the baseline saliency thresholds  $\phi_{d,g}$  and  $\kappa_{\min,\max}$ . Depth and IMU data collected from these trails was also used to test the algorithm.

##### A. Method

The trail sections were located in the the Middlesex Fells in Melrose, MA and were 9, 4, and 10.5 meters long. All included rocks and other types of solid surfaces normally encountered outdoors. We put strips of colored tape on the ground to mark nominal routes and to help establish visual correspondence among multiple video and RGB-D<sup>6</sup> recordings. The tape strips are intended to give subjects a rough idea of which route to pick but not the exact spots to place their feet.

We collected 30Hz  $640 \times 480$  RGB-D recordings of all trails with spatiotemporally coregistered<sup>7</sup> 100Hz IMU data using a handheld Kinect camera with a CH Robotics UM6 9-DoF IMU (3-axis accelerometers, 3-axis gyroscopes, and 3-axis magnetometers) attached, including a Kalman filter to estimate absolute geo-referenced orientation (Fig. 4).<sup>8</sup> The structured light method used by the Kinect does not work well in full sunlight so we took this data at twilight. Sunlight operation could be possible with other types of depth camera or stereo vision. The camera was held facing  $\sim 45^\circ$  forward and down and  $\sim 1\text{m}$  above the ground by a human operator who walked at normal pace along each trail section. The data were saved in lossless PCLZF format.

We also took video recordings of the feet of five healthy human volunteers walking on these trails. For each trail participants were asked to walk at normal pace twice in each direction, following the nominal marked route (60



Fig. 4. Our sensing apparatus is a Microsoft Kinect RGB-D camera with a CH Robotics UM6 9-DoF IMU affixed to it. It can be battery powered and works outdoors in shade or twilight. Recording software runs on a laptop while a tablet strapped to the back of the sensor gives a heads-up display.

recordings). We visually matched all footsteps (total 867) in these recordings to corresponding (pixel, frame) pairs in the RGB-D+IMU data, and we fit patches (algorithm steps 11 and 12) at these locations (Fig. 5).

##### B. Results and Threshold Estimation

We took statistics<sup>9</sup> of properties of the human selected patches including the max and min curvatures, the difference angle between the two-level normals (DoN) and the difference angle (DoNG) between the full patch normal and the upward pointing vector  $-\hat{g}$  from the IMU (Fig. 6 top and rows labeled “man” in Table I). Thresholds  $\phi_{d,g}$  and  $\kappa_{\min,\max}$  for the saliency algorithm were set to the corresponding averages from the human selected patches plus (minus for  $\kappa_{\min}$ )  $3\sigma$ , where  $\sigma$  is the standard deviation.

We ran the full algorithm on the same data frames as the human-selected patches and collected statistics on the same patch properties (Fig. 6 bottom and rows labeled “auto” in Table I). The results are similar to the human-selected patches. In a way this is by construction,<sup>10</sup> but it does help establish that the algorithm can work as intended. In total 82052 patches were fit across 832 data frames, meaning (since  $n_s = 100$ ) that about 1.4% of patches were dropped due to the curvature or residual checks (algorithm steps 13 and 14). This relatively low number indicates that the saliency checks performed prior to fitting (DoN, DoNG, and DtFP) have an additional benefit in that they help reduce time wasted fitting bad patches. In prior work where patches were fit purely at random either 3% (for triangle mesh-based

<sup>6</sup>Depth and coregistered color images; the color data was used only for visual correspondence.

<sup>7</sup>Though calibration methods have been developed (e.g. [35]), here spatial coregistration of IMU and depth data was based on the construction of the sensor apparatus. Spatial registration of the depth and RGB data used built-in calibration in the Kinect sensor. Temporal registration of all three datastreams was approximate.

<sup>8</sup>Accelerometers alone would not be able to distinguish the acceleration due to gravity from accelerations due to motion of the camera.

<sup>9</sup>Min, max, median (med), average (avg), and standard deviation (std).

<sup>10</sup>All values for “auto” are defined to fall within the corresponding “man” average plus (minus for  $\kappa_{\min}$ )  $3\sigma$ .



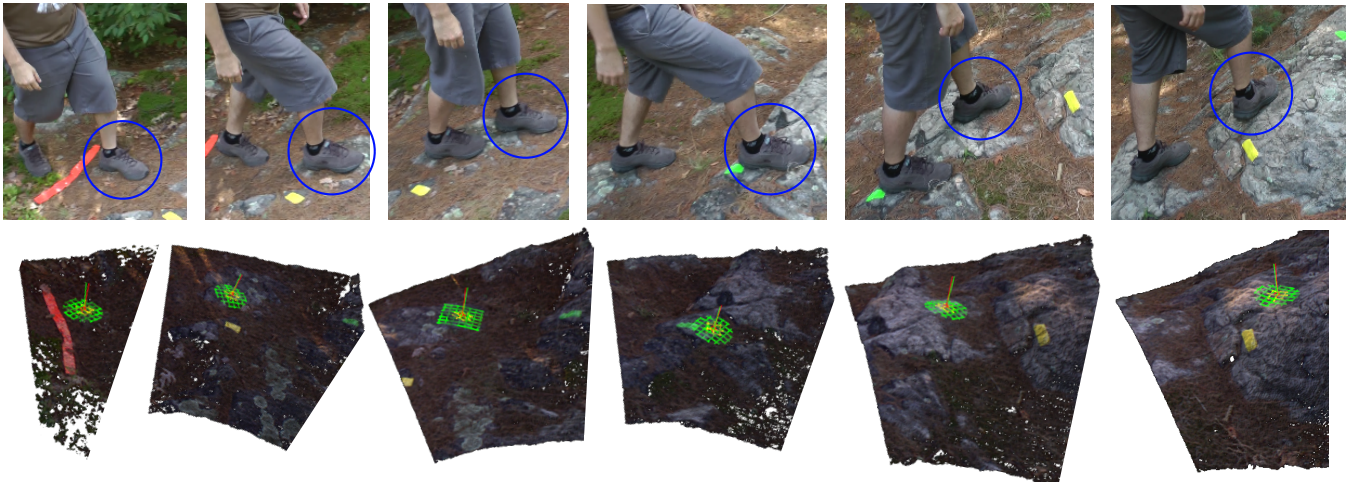
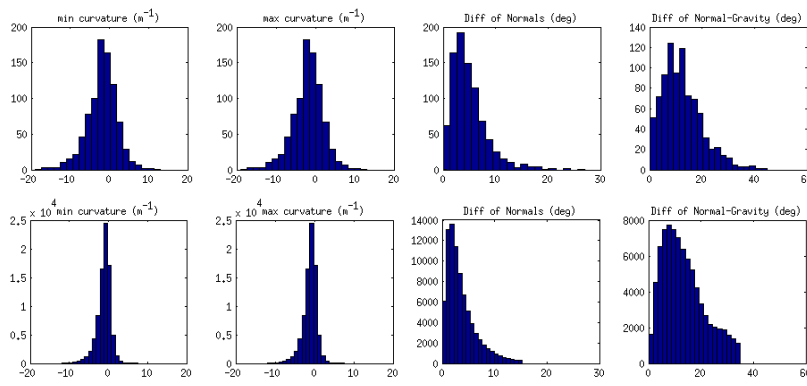


Fig. 5. Human-selected patches, manually identified in video and RGB-D recordings.



**Human Selected Patches**  
867 total patches  
832 depth+IMU data frames

**Automatically Fit Patches**  
82052 total patches  
832 depth+IMU data frames  
(same frames as above)

Fig. 6. Comparison of histograms of principal curvatures and our DoN and DoNG measures for human-selected and automatically identified patches.

measure	min	max	med	avg	std	
DoN (°)	0.00	26.94	4.17	4.95	3.45	man
	0.00	15.31	2.83	3.65	2.85	auto
DoNG (°)	0.24	44.85	11.37	12.34	7.54	man
	0.00	34.96	11.89	13.40	8.08	auto
$k_{min}$ (m <sup>-1</sup> )	-19.07	13.04	-1.70	-1.91	3.89	man
	-11.97	7.64	-0.87	-1.14	1.75	auto
$k_{max}$ (m <sup>-1</sup> )	-7.87	28.94	3.78	4.62	5.02	man
	-7.81	16.97	1.01	1.32	1.76	auto

TABLE I

neighborhoods) or 10% (for K-D tree neighborhoods) of patches were dropped due to residual alone [10].

## V. CONCLUSIONS AND FUTURE WORKS

In this paper we presented a bio-inspired approach to perceiving foot-sized contact patches in rough terrain. Our real-time system takes dense point cloud inputs from a depth camera augmented with an IMU and outputs a sparse stream of salient patches that could be used by a task-specific footfall selection algorithm. We observed the patches that humans actually select in such terrain and showed that the patches found by the system are statistically comparable.

A next step is to combine this work with our moving volume KinectFusion system [36], which integrates multiple observations from a moving depth camera into a coherent dense model of local terrain in front of and underneath the sensor. This will help enable *patch tracking* where salient

patches are found in upcoming terrain, tracked as locomotion proceeds, and then dropped when they are left behind.

We are also developing a mini biped with a depth+IMU camera and feet to negotiate rough terrain. Our long term goal is to apply these perception algorithms as part of a real-time foothold selection system on this robot.

## ACKNOWLEDGMENT

This material is based upon work supported by the National Science Foundation under Grant No. 1149235.

## REFERENCES

- [1] J. Chestnutt, M. Lau, G. Cheung, J. Kuffner, J. Hodgins, and T. Kanade, "Footstep Planning for the Honda ASIMO Humanoid," in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation (ICRA)*, 2005, pp. 629–634.
- [2] K. Nishiwaki, J. E. Chestnutt, and S. Kagami, "Autonomous Navigation of a Humanoid Robot over Unknown Rough Terrain using a Laser Range Sensor," *Int. J. Robotic Research*, vol. 31, no. 11, pp. 1251–1262, 2012.
- [3] M. Raibert, K. Blankespoor, G. Nelson, R. Playter, and the BigDog Team, "BigDog, The Rough-Terrain Quadruped Robot," *Proceedings of the 17th World Congress of the International Federation of Automatic Control*, pp. 10 822–10 825, July 2008.
- [4] E. Krotkov and R. G. Simmons, "Perception, Planning, and Control for Autonomous Walking with the Ambler Planetary Rover," *Int. J. Robotics Research*, vol. 15, no. 2, pp. 155–180, 1996.
- [5] J. Bares and D. Wettergreen, "Dante II: Technical Description, Results, and Lessons Learned," *Int. J. Robotic Research*, vol. 18, no. 7, pp. 621–649, 1999.

- [6] J. Z. Kolter, Y. Kim, and A. Y. Ng, "Stereo Vision and Terrain Modeling for Quadruped Robots," in *Proceedings of the 2009 IEEE International Conference on Robotics and Automation (ICRA)*, 2009, pp. 3894–3901.
- [7] C. Plagemann, S. Mischke, S. Prentice, K. Kersting, N. Roy, and W. Burgard, "A Bayesian Regression Approach to Terrain Mapping and an Application to Legged Robot Locomotion," *Journal of Field Robotics*, vol. 26, pp. 789–811, 2009.
- [8] M. Kalakrishnan, J. Buchli, P. Pastor, M. Mistry, and S. Schaal, "Learning, Planning, and Control for Quadruped Locomotion over Challenging Terrain," *International Journal of Robotics Research*, no. 2, pp. 236–258, 2010.
- [9] M. Vona and D. Kanoulas, "Curved Surface Contact Patches With Quantified Uncertainty," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2011, pp. 1439–1446.
- [10] D. Kanoulas and M. Vona, "Sparse Surface Modeling with Curved Patches," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2013.
- [11] D. S. Marigold and A. E. Patla, "Gaze Fixation Patterns for Negotiating Complex Ground Terrain," *Neuroscience*, pp. 302–313, 2007.
- [12] R. B. Rusu and S. Cousins, "3D is here: Point Cloud Library (PCL)," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, China, 2011.
- [13] D. Kanoulas and M. Vona, "Surface Patch Library (SPL)," 2013, <http://ccis.neu.edu/research/gpc/spl>.
- [14] J.-S. Gutmann, M. Fukuchi, and M. Fujita, "Stair Climbing for Humanoid Robots Using Stereo Vision," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2004, pp. 1407–1413.
- [15] O. E. Ramos, M. García, N. Mansard, O. Stasse, J.-B. Hayet, and P. Souères, "Towards Reactive Vision-Guided Walking on Rough Terrain: An Inverse-Dynamics Based Approach," in *Workshop on Visual Navigation for Humanoid Robots (IEEE ICRA)*, Karlsruhe, Germany, May 2013.
- [16] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohli, J. Shotton, S. Hodges, and A. Fitzgibbon, "KinectFusion: Real-Time Dense Surface Mapping and Tracking," in *Proceedings of the IEEE ISMAR*, 2011.
- [17] D. Maier, C. Lutz, and M. Bennewitz, "Integrated Perception, Mapping, and Footstep Planning for Humanoid Navigation Among 3D Obstacles," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems IROS*, 2013.
- [18] F. Doshi, E. Brunskill, A. C. Shkolnik, T. Kollar, K. Rohanimanesh, R. Tedrake, and N. Roy, "Collision Detection in Legged Locomotion using Supervised Learning," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2007, pp. 317–322.
- [19] D. Wooden, M. Malchano, K. Blankespoor, A. Howard, A. A. Rizzi, and M. Raibert, "Autonomous Navigation for BigDog," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2010, pp. 4736–4741.
- [20] A. Stelzer, H. Hirschmüller, and M. Görner, "Stereo-Vision-Based Navigation of a Six-legged Walking Robot in Unknown Rough Terrain," *Int. J. of Robotics Research*, vol. 31, no. 4, pp. 381–402, 2012.
- [21] M. Hollands and D. Marple-Horvat, "Visually Guided Stepping under Conditions of Step Cycle-Related Denial of Visual Information," *Experimental Brain Research*, pp. 343–356, 1996.
- [22] A. Patla, "Understanding the Roles of Vision in the Control of Human Locomotion," *Gait and Posture*, pp. 54–69, 1997.
- [23] A. Patla and J. Vickers, "Where and When do we Look as we Approach and Step over an Obstacle in the Travel Path?" *Neuroreport*, pp. 3661–3665, 1997.
- [24] S. Rietdyk and C. K. Rhea, "Control of Adaptive Locomotion: Effect of Visual Obstruction and Visual Cues in the Environment," *Experimental Brain Research*, pp. 272–278, 2006.
- [25] D. S. Marigold and A. E. Patla, "Visual Information from the Lower Visual Field is Important for Walking Across Multi-Surface Terrain," *Exp Brain Res*, vol. 188, pp. 23–31, 2008.
- [26] M. A. Lewis, H.-K. Lee, and A. Patla, "Foot Placement Selection using Non-geometric Visual Properties," *The International Journal of Robotics Research*, vol. 24, pp. 553–561, July 2005.
- [27] C. H. Lee, A. Varshney, and D. W. Jacobs, "Mesh Saliency," in *ACM SIGGRAPH*. New York, NY, USA: ACM, 2005, pp. 659–666.
- [28] Y.-S. Liu, M. Liu, D. Kihara, and K. Ramani, "Salient Critical Points for Meshes," in *Proceedings of the 2007 ACM Symposium on Solid and Physical Modeling (SPM)*, New York, NY, USA, 2007, pp. 277–282.
- [29] J. Lam and M. Greenspan, "On the Repeatability of 3D Point Cloud Segmentation Based on Interest Points," in *Computer and Robot Vision (CRV), 2012 Ninth Conference on*, 2012, pp. 9–16.
- [30] S. Paris and F. Durand, "A Fast Approximation of the Bilateral Filter using a Signal Processing Approach," in *In Proceedings of the European Conference on Computer Vision*, 2006, pp. 568–580.
- [31] S. Holzer, R. B. Rusu, M. Dixon, S. Gedikli, and N. Navab, "Adaptive Neighborhood Selection for Real-Time Surface Normal Estimation from Organized Point Cloud Data Using Integral Images," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2012, pp. 2684–2689.
- [32] M. Pauly, M. Gross, and L. P. Kobbelt, "Efficient Simplification of Point-Sampled Surfaces," in *Proceedings of the Conference on Visualization (VIS)*. Washington, DC, USA: IEEE Computer Society, 2002, pp. 163–170.
- [33] T. Q. Pham, "Non-Maximum Suppression using Fewer than two Comparisons per Pixel," in *ACIVS (1)*, 2010, pp. 438–451.
- [34] G. Taubin, "An Improved Algorithm for Algebraic Curve and Surface Fitting," in *Computer Vision, 1993. Proceedings., Fourth International Conference on*, 1993, pp. 658–665.
- [35] J. Lobo and J. Dias, "Relative pose calibration between visual and inertial sensors," *The International Journal of Robotics Research*, vol. 26, pp. 561–575, jun 2007.
- [36] H. Roth and M. Vona, "Moving Volume KinectFusion," in *British Machine Vision Conference (BMVC)*, September 2012.