Deep Convolutional Terrain Assessment for Visual Reactive Footstep Correction on Dynamic Legged Robots

Octavio Villarreal[1], Victor Barasuol[1], Marco Camurri[1,4], Michele Focchi[1], Luca Franceschi[2], Massimiliano Pontil[2], Darwin G. Caldwell[3] and Claudio Semini[1]

***Abstract*— In this work an on-line, dynamic foothold correction strategy based on visual feedback for legged robots is presented. Images representing terrain surfaces in the vicinity of a foothold are evaluated. This is done off-line based on terrain roughness and the robot's morphology to select the best possible foothold. We train a terrain classifier based on a Convolutional Neural Network (CNN) using the pairs of terrain images and selected footholds. We evaluate the strategy on the quadruped robot HyQ while traversing challenging terrain.**

I. Introduction

Mobile legged robots are able to traverse rough and unstructured terrain in a robust fashion [1], [2], [3], [4], [5], [6], [7], [8]. However, precise foothold placement and fast reactions are often necessary to avoid motions and postures that may result in failure, such as collisions, slippage or falls during dynamic locomotion. In this work, an on-line, dynamic foothold correction strategy based on visual feedback for legged robots is presented. We implement a terrain classifier based on a Convolutional Neural Network (CNN), that maps images of the terrain to foothold corrections.

We test the strategy on the quadruped robot HyQ while traversing challenging terrain. We based our motion generation on the Reactive Controller Framework (RCF) [1]. Herein, the feet trajectories are computed periodically based on synchronized Central Pattern Generators (CPGs). We apply the correction computed by the CNN classifier on top of the predefined trajectory, in a continuous fashion during swing phase for each of the legs.

The main contributions of this work are: 1) a set of **heuristic criteria** that considers the surface and the robot kinematics to select the best foothold; 2) a **foothold correction** strategy that is executed while performing fast gait motions, allowing the robot to cope with rough terrain and 3) the use of a low-dimensional CNN that is more computationally efficient with respect to the continuous evaluation of the heuristics used for training.

II. Approach

In this work, we aim to create an "intermediate layer" that links the robustness of the reactive strategies (namely, the RCF [1] with the preventive behavior when visual information is available. Specifically, we propose an on-line foothold correction, which uses only on-board sensing to be executed while the robot traverses difficult environments.

Our method extends the work of [9], and it is devised in three stages: 1) template recollection; 2) heuristic foothold selection; and 3) CNN training.

To implement our strategy, we first estimate the location of the next foothold for each leg. In the case of a statically stable gait (e.g., a crawling gait) the predicted foothold position only depends on the tracking of the trajectory of the foot. On the other hand, while executing a trotting gait, the velocity of the trunk also influences the next landing position of the feet.

After obtaining a predicted foothold, a heightmap around the vicinity of the predicted foothold is acquired from the vision system, to then compute the foothold correction using the CNN-based corrector. We then apply the necessary changes in trajectory to execute the corrections. This is done continuously during swing phase for each of the legs. Being able to correct a foothold continuously during swing phase is of key importance. If the robot is perturbed (e.g., by an external force applied on the trunk) while traversing a rough scenario, the continuous correction will allow the leg to change its landing position to a safe foothold, increasing the robustness capabilities during locomotion.

Template recollection. We collect a series of templates consisting of images where each pixel represents the height of a specific point of the terrain (namely, heightmaps). Templates can be acquired from three main sources: simulation, experience or specific templates generated artificially. Each heightmap consists of a grid of 15×15 pixels with a size of $2\,\mathrm{cm}^2$ per pixel. Each pixel represents a possible foothold and the center pixel of the grid is the expected landing position without correction (**nominal foothold**). Figure 1 shows an example of the quadruped robot HyQ collecting maps in simulation.

Heuristic foothold selection. We initially create a training set from the collected maps and select the best foothold based on surface roughness, robot kinematics and default swing trajectory. Rougher sections of the terrain are avoided by discarding cells (pixels in the heightmap) that have large values of standard deviation and mean of the height with respect to its neighboring cells. Cells that lead to a foot frontal collision, shin collision or are not reachable according to the robot leg kinematics are also discarded. Finally, from the remaining cells, the one that is closest to the **nominal**

[1]Dynamic Legged Systems Lab, Istituto Italiano di Tecnologia, Via Morego 30, 16163 Genoa, Italy. firstname.lastname@iit.it
[2]Computational Statistics and Machine Learning, Istituto Italiano di Tecnologia, Via Morego 30, 16163 Genoa, Italy. firstname.lastname@iit.it
[3]Department of Advanced Robotics, Istituto Italiano di Tecnologia, Via Morego 30, 16163 Genoa, Italy. darwin.caldwell@iit.it
[4]Oxford Robotics Institute, University of Oxford, 23 Banbury Rd, OX2 6NN Oxford, UK. mcamurri@robots.ox.ac.uk

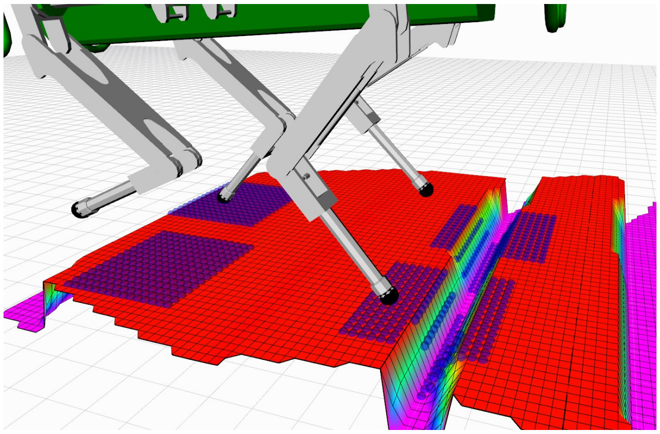

Fig. 1: Simulation of the quadruped robot HyQ collecting templates while executing a trotting gait. Templates are represented by the squared areas around future footholds, where each of the blue spheres is a potential foothold within the grid.

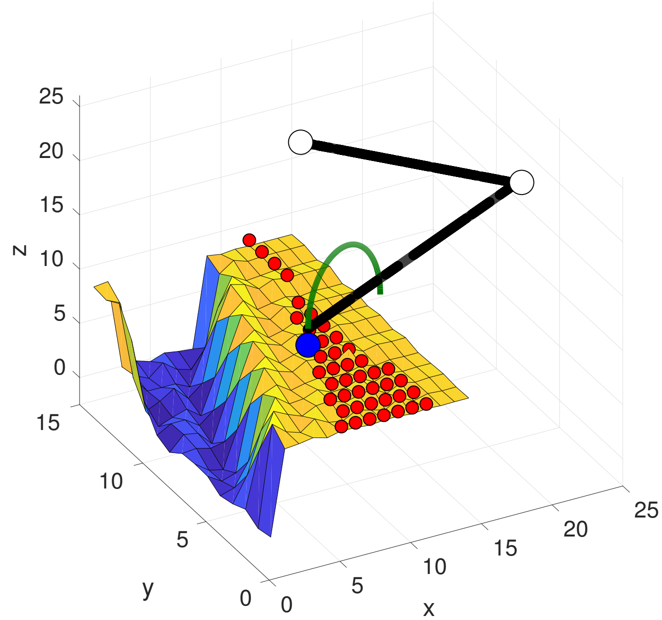

Fig. 2: Example of an evaluated terrain template. Red circles represent the set of available safe footholds, the blue circle represents the best foothold among the set of safe footholds (i.e., the closest to the center of the grid). The lower and upper limb of the leg are reprensented by the black solid lines and the white circles represent the location of the joints. The foot follows the trajectory indicated by the solid green line.

foothold is deemed as the optimal foothold. Namely, from the non-discarded cells, the optimal foothold is the one that yields the smallest 2-norm of the vector directed from the nominal foothold to itself. Figure 2 shows an example of an evaluated template.

It is important to take into account that having a drift-free, accurate map computed on-board is still a very difficult task for mobile robots. In legged robots, the task becomes more challenging when executing dynamic gait motions, due to the influence of constant impacts and higher velocities with respect to static gaits. This adds a high level of uncertainty to the computation of the map. In our case, we deal with this uncertainty by adding a "safety margin" around the evaluated footholds. This safety margin defines a radius of cells that have to comply with the previously mentioned criteria around an evaluated foothold. If any of the surrounding footholds within this radius does not comply with the criteria, the currently evaluated foothold is discarded. It is also worth noting, that if the system faces a scenario that was not considered when building the training set, unexpected foothold locations may be computed by the CNN-based corrector, which may lead to unsafe landing positions.

CNN training. We train a low-dimensional CNN based on the set generated using the foothold selection heuristics. A CNN-based foothold corrector was chosen, due to its proven effectiveness for processing image data [10], [11]. We achieve a gain in computational efficiency with respect to the continuous evaluation of the aforementioned heuristics, thanks to the proposed architecture chosen as a trade-off between speed and accuracy. Figure 3 shows the chosen architecture.

III. Results

In this section, we present the results corresponding to the accuracy and effectiveness of the CNN-based foothold corrector. Also, the results of the implementation of the foothold correction strategy are presented.

CNN design and evaluation. Based on the heuristics we generated 12687 examples. The examples were both created artificially (i.e., defining the height values of each element of the 15×15 matrix) and through simulation. The set includes bars, gaps, blocks and stairs, of various heights and at different orientations. Half of the examples are used for training and the other half for testing. We define three types of footholds: **an optimal foothold** corresponds to the best foothold according to the heuristics; **safe footholds** are the ones that are not discarded by the heuristics but are further away from the nominal foothold than the optimal; **unsafe footholds** are the ones discarded by the heuristics. The data

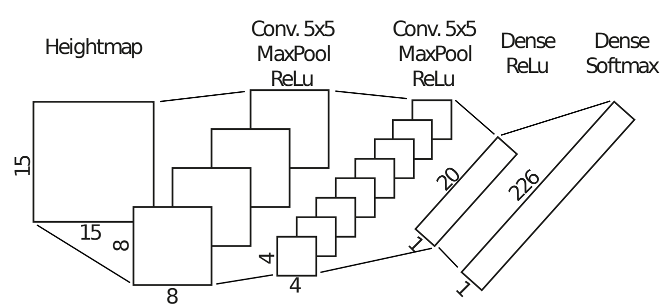

Fig. 3: Architecture of the trained CNN used to map terrain templates to foothold corrections.

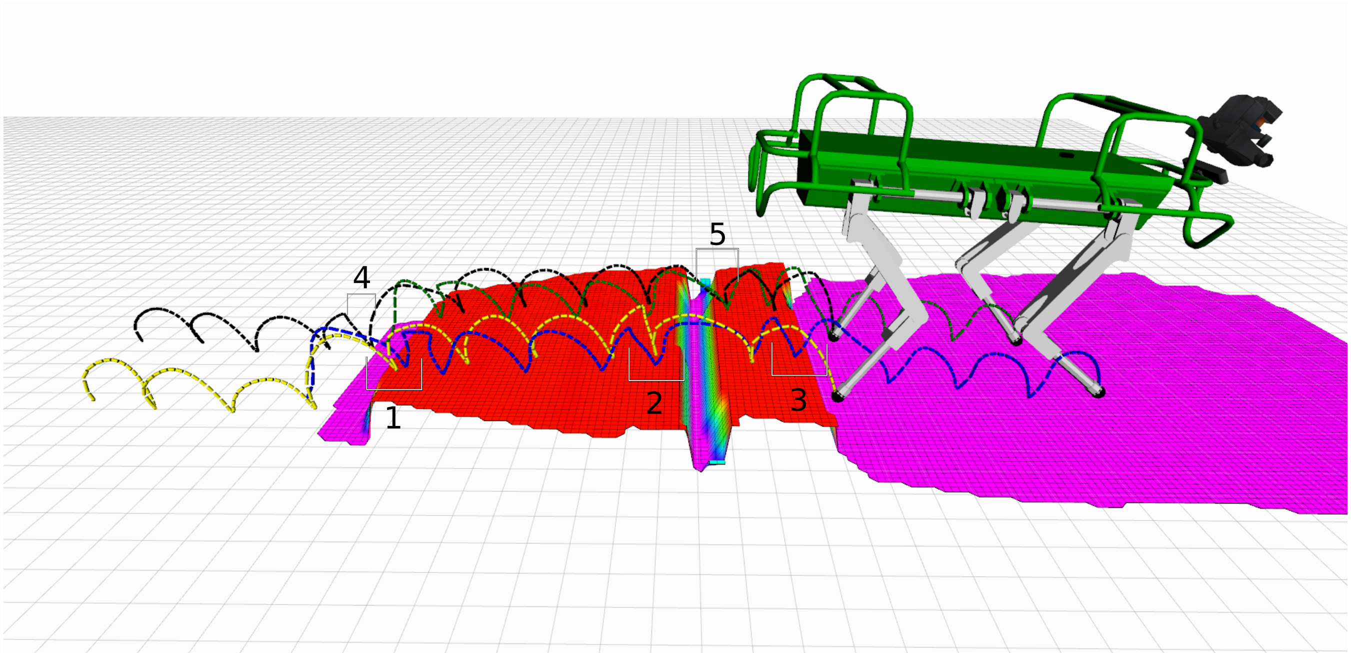

Fig. 4: Test scenario composed of a pallet with 15 cm height and 90 cm large, followed by a 15 cm height and 30 cm large wooden beam, separated by a 10 cm gap. The dashed lines correspond to the foot trajectories that are corrected by the vision-based feedback, during 0.5 m/s trotting gait: left-front (green line), right-front (blue line), left-hind (black line) and right-hind (yellow line).

was split equally for training and testing. After training, the CNN-based foothold corrector was tested, achieving approximately 85% of perfect prediction (i.e, choosing the optimal foothold). It is worth noting that even if the optimal foothold was not always chosen by the CNN, approximately 99% of the footholds coming from the corrector corresponded to safe footholds according to the heuristics based on surface evaluation and robot kinematics.

To assess the computational gain of our method, we use the number of clock tick counts divided by the clock frequency of the processor. We compare the time taken by both the full-blown heuristics and the CNN-based foothold corrector. In the case of the full blown heuristics, the time taken to evaluate one template ranges from 0.1 ms to 20 ms. On the other hand, the CNN-based foothold corrector takes between 0.072 ms and 0.1 ms. Therefore, the CNN-based predictor is 1.5 to 200 times faster than the computation of the full-blown heuristics. Furthermore, the robot receives control commands at an update rate of 4 ms. Hence, we are able to send foothold corrections continuously at any given time during swing phase without compromising the robot stability due to real-time constraints.

Simulation results. Figure 4 depicts the details of a simulation scenario where it is possible to see the heightmap computed by the perception system and the resulting foot trajectories while the HyQ robot is trotting using the RCF [1]. To build the map, we use an RGB-D sensor located at the front of the robot and generate a local map using the Grid Map library from [12]. The map is built as the robot moves, keeping the past information collected from the vision sensor and placing the map based on the robot state estimator [13]. For this task, the robot is asked to trot at a constant forward velocity of 0.5 m/s. This task was chosen because it is particularly difficult for blind locomotion, since it may lead to "deadlocks" if one of the legs falls into the gap.

Paying close attention to the footprints (dashed lines), it is possible to see the effect of the foothold corrections on the nominal foot trajectories. Examples of the changes of trajectory triggered by the foothold corrections are:

- The correction labeled as number 1 shows two corrections, one corresponding to the right front leg (blue line) and one associated with the right hind leg (yellow line). In the case of the right front leg, the leg changes its nominal foothold avoiding a foot frontal collision with the pallet. Regarding the right hind leg, it also changes its foothold avoiding both foot frontal and shin collisions.
- Correction number 2 shows a pair of corrections for the same legs as correction number 1. In this case, both legs decide to change footholds before their nominal landing positions. This prevents the situation corresponding to the robot stepping close to the edge (avoiding slippage due to uncertainty in the map) and into the gap (avoiding shin collision).
- In correction number 3, the right front leg takes a shorter step, instead of going up to the nominal landing position (which is outside of the wooden beam). This helps to avoid a shin collision due to its proximity to the edge of the wooden beam. This is similar to the correction

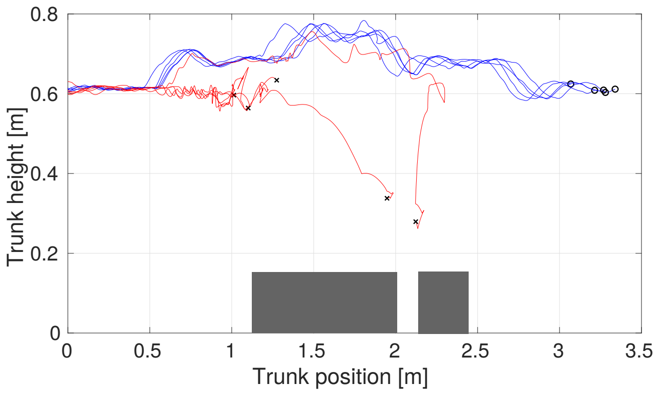

Fig. 5: Plot depicting the trunk height of the robot during multiple trials of the gap crossing scenario simulation at $0.5\,\mathrm{m/s}$. Red lines indicate the trials without visual reaction activated, while blue lines indicate the trials where the visual reaction was active.

executed by the left front leg (green line), labeled with number 5.

- Correction labeled as 4, shows how the left hind leg (black line) takes a shorter step before the pallet avoiding shin collision.

Thanks to the corrections, the robot is able to accomplish the task without performing motions that lead to an unsafe situation. A similar trial to the one shown in Figure 4 can be seen in the associated video [1].

Figure 5 shows a series of trials of the gap crossing task using the RCF with (blue lines) and without (red lines) visual corrections, in order to test the repeatability of the strategy. It can be noted that when visual feedback is provided to the robot, the task is completed successfully, contrary to the case when no visual corrections are executed.

IV. Conclusions

We have presented a novel strategy for on-line foothold correction based on a Deep Convolutional Neural Network from visual feedback. The simulation results showed that the robot was able to traverse a difficult scenario, avoiding undesired motion trajectories that may lead to harmful situations (e.g., slippage, collisions or falls). As future work, we will test experimentally our visual reaction strategy on the HyQ robot while performing both static and dynamic gaits. Furthermore, we aim to show how the vision-based reaction can aid to keep the stability when the robot is subject to disturbances on the trunk.

References

[1] V. Barasuol, J. Buchli, C. Semini, M. Frigerio, E. R. De Pieri, and D. G. Caldwell, "A reactive controller framework for quadrupedal locomotion on challenging terrain," in *2013 IEEE International Conference on Robotics and Automation (ICRA)*, 2013.

[2] H.-W. Park, P. M. Wensing, and S. Kim, "High-speed bounding with the mit cheetah 2: Control design and experiments," *The International Journal of Robotics Research*, vol. 36, no. 2, pp. 167–192, 2017. [Online]. Available: https://doi.org/10.1177/0278364917694244

[1]https://youtu.be/bQDK5Kda3RA

[3] C. D. Bellicoso, F. Jenelten, P. Fankhauser, C. Gehring, J. Hwangbo, and M. Hutter, "Dynamic locomotion and whole-body control for quadrupedal robots," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sept 2017, pp. 3359–3365.

[4] C. Mastalli, A. Winkler, I. Havoutis, D. G. Caldwell, and C. Semini, "On-line and on-board planning and perception for quadrupedal locomotion," in *IEEE International Conference on Technologies for Practical Robot Applications (TEPRA)*, May 2015.

[5] M. Kalakrishnan, J. Buchli, P. Pastor, and S. Schaal, "Learning locomotion over rough terrain using terrain templates," in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Oct 2009, pp. 167–172.

[6] D. Wahrmann, A. C. Hildebrandt, R. Wittmann, F. Sygulla, D. Rixen, and T. Buschmann, "Fast object approximation for real-time 3d obstacle avoidance with biped robots," in *2016 IEEE International Conference on Advanced Intelligent Mechatronics (AIM)*, July 2016, pp. 38–45.

[7] D. Belter and P. Skrzypczyński, "Rough terrain mapping and classification for foothold selection in a walking robot," *Journal of Field Robotics*, vol. 28, no. 4, pp. 497–528, 2011. [Online]. Available: http://dx.doi.org/10.1002/rob.20397

[8] P. Fankhauser, M. Bjelonic, C. D. Bellicoso, T. Miki, and M. Hutter, "Robust rough-terrain locomotion with a quadrupedal robot," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 2018.

[9] V. Barasuol, M. Camurri, S. Bazeille, D. Caldwell, and C. Semini, "Reactive trotting with foot placement corrections through visual pattern classification," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct 2015.

[10] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[11] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, *Deep learning*. MIT press Cambridge, 2016, vol. 1.

[12] P. Fankhauser and M. Hutter, "A Universal Grid Map Library: Implementation and Use Case for Rough Terrain Navigation," in *Robot Operating System (ROS) - The Complete Reference (Volume 1)*, A. Koubaa, Ed. Springer, 2016, ch. 5. [Online]. Available: http://www.springer.com/de/book/9783319260525

[13] S. Nobili, M. Camurri, V. Barasuol, M. Focchi, D. G. Caldwell, C. Semini, and M. Fallon, "Heterogeneous sensor fusion for accurate state estimation of dynamic legged robots," in *Proceedings of Robotics: Science and Systems*, Boston, USA, July 2017.