

Active Camera Stabilization to Enhance the Vision of Agile Legged Robots

Stéphane Bazeille*, Jesus Ortiz*, Francesco Rovida[†], Marco Camurri*,
 Anis Meguenani[‡], Darwin G. Caldwell*, Claudio Semini*

* Department of Advanced Robotics, Istituto Italiano di Tecnologia, via Morego, 30, 16163 Genova.

<first name>.<last name>@iit.it

[†] Robotics, Vision and Machine Intelligence Lab, Aalborg University of Copenhagen AC Meyers
 Vaenge 15 DK-2450 Copenhagen. francesco@m-tech.aau.dk

[‡] Institut des Systèmes Intelligents et de Robotique (ISIR), Université Pierre et Marie Curie Pyramide -
 Tour 55, 4 Place JUSSIEU 75005 Paris. meguenani@isir.upmc.fr

Abstract

Legged robots have the potential to navigate in more challenging terrains than wheeled robots. Unfortunately, their control is more demanding, because they have to deal with the common tasks of mapping and path planning as well as more specific issues of legged locomotion, like balancing and foothold planning. In this paper, we present the integration and the development of a stabilized vision system on the fully torque-controlled hydraulically actuated quadruped robot (HyQ). The active head added onto the robot is composed of a fast pan and tilt unit and a high resolution wide angle stereo camera. The pan and tilt unit enables camera gaze shifting to a specific area in the environment (both to extend and refine the map) or to track an object while navigating. Moreover, as the quadruped locomotion induces strong regular vibrations, impacts or slippages on rough terrain, we took advantage of the pan and tilt unit to mechanically compensate for the robot's motions. In this paper, we demonstrate the influence of legged locomotion on the quality of the visual data stream by providing a detailed study of HyQ's motions, which are compared against a rough terrain wheeled robot of the same size. Our proposed IMU-based controller allows us to decouple the camera from the robot motions. We show through experiments that, by stabilizing the image feedback, we can improve the onboard vision-based processes of tracking and mapping. In particular, during the outdoor tests on the quadruped robot, the use of our camera stabilization system improved the accuracy on the 3D maps by 25 %, with a decrease of 50 % of mapping failures.

Index Terms

Stereo vision, Camera stabilization, Active Head, Simultaneous Localization and Mapping (SLAM), Color Tracking, Quadruped Robot.

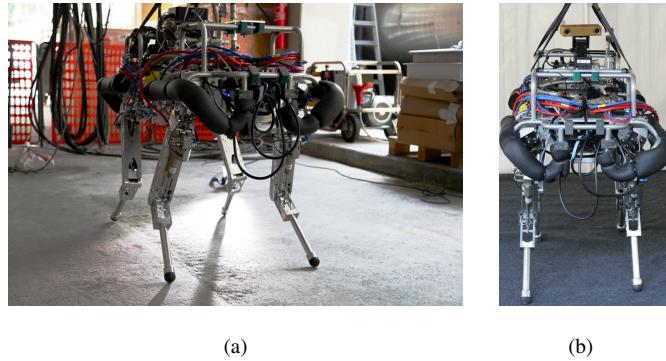


Figure 1: Picture of IIT’s HyQ robot in its outdoor test area. (a) The robot is shown before the integration of the active head (b) HyQ with its active head.

I. INTRODUCTION

Legged robots are naturally superior at accessing a large variety of surface conditions when compared to wheeled robots. This is partially due to the ability of legged systems to decouple the path of the robot from the sequence of footholds and to their inherent ability to use a range of locomotion strategies, tailored to the situation at hand.

The *Hydraulic Quadruped*, HyQ (Fig. 1(a)) developed at the Department of Advanced Robotics at the Istituto Italiano di Tecnologia (IIT) [1] is a versatile robot with hydraulic actuation. HyQ is fast, robust, actively compliant and built for dynamic locomotion in both indoor and outdoor environments. Hydraulic arms for HyQ are under development and will soon add a manipulation capability. Possible future application domains are search and rescue, forestry and agriculture, inspection and exploration.

As of today, the robot is able to perform highly dynamic tasks, like trotting with speeds up to 2 m/s, walking, jumping, rearing, performing step reflexes, and keeping its balance on inclined terrain or on unstable ground and under external disturbances. HyQ has demonstrated successful operations with minimal exteroceptive feedback [2], [3], [4] relying only on the onboard *Inertial Measurement Unit* (IMU), torque and joint position sensing.

The next milestone of the project is to improve the robot’s autonomy by perceiving the environment to be able to walk over unknown and rugged terrains in a planned manner. To this end, we added a stereo vision sensor to provide the robot with higher-level feedback, which can be used in a number of ways, *e.g.*, localization, mapping, ground recognition, or terrain modeling. Indeed, the camera is a ubiquitous sensor for autonomous robotic applications as it has many advantages such as small size, low weight, low energy consumption, and above all it provides rich environmental information. It delivers essential sensory data in order to: (i) decide a path or compute footholds based on three dimensional terrain reconstruction and classification, (ii) grasp an object while trotting and produce rich maps with semantic information (*e.g.*, objects, pedestrians, holes, ground surfaces). In our previous work we have successfully shown some preliminary vision-enhanced locomotion behavior where the frame-by-frame visual information was used to modify the locomotion parameters of the robot gait [5], [6] and we presented some foothold

planning results on pre-built maps [7].

However, to perform such complex tasks as SLAM and path planning, additional challenges arise in the context of highly dynamic locomotion performed by a quadruped robot in an unknown unstructured environment. On one hand, legged locomotion creates regular body motion, which induces vibrations, and high impacts that lead to important changes between consecutive camera frames. On the other hand, navigation in rough terrain can induce large motions (slippage, loss of balance, fall) which increase the difference between frames and occasionally add motion blur in the images. These issues decrease the performance of image processing algorithms such as feature matching and thereby weaken the results of tracking. One solution could be to increase the frame rate but this would drastically increase the computational load. Hence, to keep a robust real time position tracking with a frame rate of 10 Hz, it is crucial to stabilize the camera.

In order to improve the robustness and the versatility of the vision system, we mounted the camera onto a fast Pan and Tilt Unit (PTU). These two additional degrees of freedom pave the way towards: (a) enabling the robot to perform tasks such as object tracking or manipulation while trotting, (b) improving mapping and localization by looking around and (c) compensating for the body motion and thus providing a smoother video feed to the robot itself or to a tele-operator.

Contributions: in this paper, we present the integration and the development of an onboard mechanically stabilized vision system for the robot HyQ. We assess our system by comparing the camera motion with and without stabilization during trotting and walking experiments. A fast optical based camera pose estimation algorithm has been specifically developed to validate our method. Moreover, we demonstrated the improvements of our mechanical motion stabilization system both on mapping and tracking.

Contents: the remainder of this paper is organized as follows. In Section II, we present a review of related work on quadruped robots equipped with vision sensors and camera stabilization with active heads. In Section III we present the details of the hardware, the sensors and the method we used to validate the quality of the motion stabilization. In Section IV, we compare the motion of our dynamic platform to the motion of a rough terrain wheeled robot to outline the particular issues introduced by our robot. Section V explains the developed camera stabilization method. In Section VI we demonstrate the effectiveness of our approach during tracking and mapping experiments, and discuss the obtained results. Finally, in Section VII we conclude the paper and present our future work.

II. RELATED WORK

In this section we survey the related work on quadruped robots equipped with vision sensors. Then, we discuss different platforms, mainly wheeled and humanoid robots, equipped with an active head, *i.e.*, with a camera mounted on a motorized neck. Later on we present the related work on camera stabilization systems. Finally, we detail the commercially available stabilization devices.

A. Legged robots with visual capabilities

Research on quadrupedal locomotion is an active field in the robotic community. However, up to now only a few groups have worked on the integration of vision sensors into quadruped platforms. Most research is focused on the development of step controllers, rather than on higher-level, perception-based processes. Legged locomotion over challenging terrain requires precise and failsafe perception capabilities, regardless of issues like stability, dynamic motion, impact, or complex visibility conditions. As a consequence, a large part of studies in quadrupedal locomotion often simplifies the problem of perception by doing onboard terrain mapping with the support of external localization or use accurate previously acquired maps while solving the localization of the robot onboard.

Kolter *et al.* [8] presented the most autonomous approach by removing the dependence on given maps and external state input. In their control framework they use a stereo camera and an ICP-based technique (Iterative Closest Point) for point cloud registration. Then, they perform motion planning with their quadruped *LittleDog*. Although the camera was on the robot, the vision processing and path planning were performed on an external computer.

Again on *LittleDog*, Filitchkin and Byl [9] used a monocular camera to perform terrain classification, and in turn select between predetermined gaits to traverse terrain of varying difficulty. It has to be noted that the high gear ratio and the low control bandwidth of *LittleDog*'s joints make it very difficult to implement well controlled dynamic gaits.

More recently, Stelzer *et al.* [10] developed a complete visual navigation framework for their small hexapod robot. The algorithm computes pose by fusing inertial data with relative leg odometry and visual odometry measurements. A traversability map is built using the 3D point cloud from the stereo system and the previously computed pose. Then paths are planned and the robot chooses the appropriate gait according to the terrain.

In the different papers introduced previously the research was performed on small electrically-actuated robots using statically stable gaits. Rough terrain mapping and navigation for highly dynamic robots of the size of HyQ or larger have not yet been much explored in the scientific community. *BigDog* is the most well known example, where in [11], [12] the authors fused the information from stereo vision, leg odometry, and IMU in order to obtain an accurate state estimate of the robot. They also developed a registration pipeline and an obstacle detection algorithm using stereo vision and a LIDAR scanner.

Bajracharya *et al.* [13] recently showed terrain mapping for vision-in-the-loop walking on the *LS3* robot from Boston Dynamics. The vision system is used to map the environment in the vicinity of the robot and inform the gait generation process about possible changes in the surface where the robot is locomoting. Finally, Shao *et al.* [14] presented an obstacle avoidance approach for their electrically actuated quadruped robot that uses stereo vision for terrain modeling algorithm.

In the context of quadrupedal locomotion, only a few results have been shown because the use of a vision sensor is more complicated due to strong and fast unwanted motions in the image sequence that make standard algorithms inefficient. As a consequence, state-of-the-art algorithms must be improved, or image stabilization needs to be added as vision processing algorithms often reach optimal performance with a stable image. When dealing with image stabilization, two approaches have to be considered: digital stabilization and mechanical stabilization.

B. Digital image stabilization

Image stabilization is usually done digitally because it is simpler and less expensive, but mechanical stabilization usually gives better performances, and compensates a much wider range of jitter. Mechanical stabilization can normally compensate up to 50 pixel jitter, while digital stabilization methods can only compensate up to 20 pixel jitter with the same performance criterion [15].

On quadruped robots, image stabilization has been implemented digitally by Karazume and Hirose [16] on their quadruped robot TITAN-VIII. The only mechanical motion stabilization developed on a quadruped has been shown in [17] and [18], where the authors have equipped Sony's 1.5 kg robot AIBO with a system that mimics the inertial sensing mechanisms of the vestibular system of mammals. Their approach can learn how to compensate for the stabilization of head movements.

On other kinds of robots, mechanical motion stabilization has been studied more extensively, as shown in the next section.

C. Mechanical motion stabilization with active head

Although digital stabilization is cheaper and simpler to integrate, mechanical stabilization provides two main advantages. First, there is no extra computational cost, which is an important benefit on a robot with limited resources. Second, the PTU makes a vision system more versatile and efficient, because it enables scanning of regions of interest and obstacle tracking while avoiding uninteresting areas.

In the research state-of-the-art, the works described in [19], [20] show an active head on a wheeled vehicle that tracks and follows a leader vehicle on rough terrain. Among the recent active camera systems for wheeled vehicles, it is worth mentioning the work of Gasteratos *et al.* [21], who developed a robotic head with four degrees of freedom, an IMU and two digital cameras. In their approach, they use inertial and optical flow data for mechanical and digital image stabilization, respectively.

The largest quantity of work on camera stabilization with active heads has been done in the humanoid field with robots that also perform dynamic manipulation based on vision. The authors of [22] mounted an active head on a humanoid robot, which performs ego motion compensation to stabilize the vision system during walking.

In [23] vision-based grasping is demonstrated on a humanoid robot. Also, a manipulation system for a highly dynamic task is presented in [24], their robot catches a thrown ball by continuously tracking it and adjusting the pose of the full body. The authors combine smooth pursuit of an object with active compensation of vibration motion.

In [25], [26], high speed tracking on an active vision system with fewer degrees of freedom has been developed. Another example is presented in [27] where an active vision head as part of a humanoid platform performing grasping and manipulation tasks while tracking the current object of interest and keeping it in the field of view.

Some groups worked on developing fast stabilization systems that are called oculomotor reflexes. For example [28] using predictive filters to compensate for the periodic motions of a humanoid robot. Also [29] and [30] developed oculomotor reflexes to stabilize the gaze when facing unknown perturbations of the body. These last mentioned approaches used on humanoids in the context of mobile manipulation platforms shows good results. But,

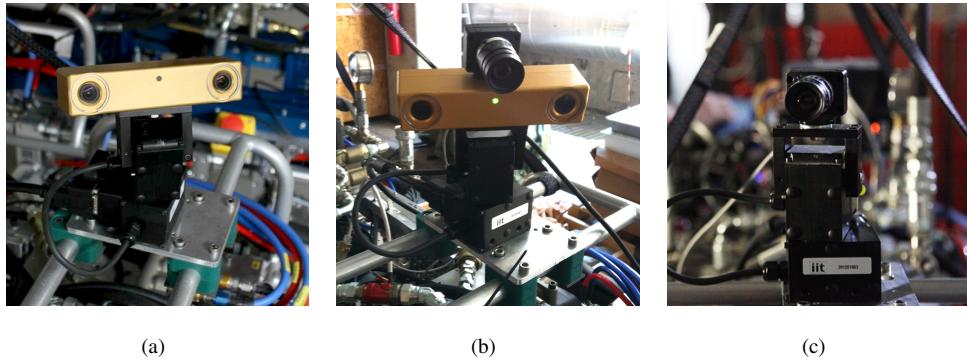


Figure 2: The HyQ’s different active head (a) HyQ’s active head (b) Active head setup used for mapping experiment (stereo camera, monocular camera and the PTU); (c) Active head setup used for the tracking experiment (monocular camera and the PTU).

as they use learning to anticipate the compensation, they are applied on “stable” platforms. In our case we focus on dynamic locomotion of a highly mobile platform, improving the image processing results keeping the whole system onboard and fully reactive. As a consequence we do not try to completely compensate the camera motions as the aforementioned humanoid robots. Instead, we smooth the erratic motions of our dynamic platforms in order to improve the robustness of our stereo vision based image processing.

Finally, some stabilized cameras or stabilization systems for cameras are commercially available since few years. They are according to our knowledge only monocular and mainly used by photographers and video makers to improve the rendering. Boats or flying drones are typical platforms which required camera stabilization. Examples are: the Ocular Robotics REV-25 stabilized cameras [31] or stabilization systems such as the Flir Pan-Tilt Unit-D48E-ISM [32], the Stabilgo Rig, the Tarot T-2D Brushless Gimbal, and the Nebula 4000. These systems differ from ours in two aspects: first, they are usually much bigger. Second, their only purpose is only the stabilization, while our platform can also be used for active scanning or tracking.

III. SYSTEM OVERVIEW

A. *HyQ structure and control*

The experimental platform used in this study is the versatile quadruped robot HyQ [1] (Fig. 2). It is a fully torque-controlled, hydraulic actuated machine that weighs 85 kg or 110 kg with external or onboard hydraulic power supply, respectively. The robot is 1 m long and has legs that are 0.78 m when fully extended. Each leg has three active degrees of freedom, and all the joints are equipped with torque sensors, high-resolution encoders and high-performance servo valves, which drive the joints with a bandwidth of 250 Hz. An onboard IMU, placed in the middle of the torso, is used for robot balancing. The selection of these components, in combination with model-based control techniques, enables high-performance control in torque and position, alongside the implementation of active compliance.

Two onboard computers are in charge of robot control and perception: a Pentium PC104, running real-time Linux (Xenomai) with the joint-level control loop at 1 kHz; and a vision-dedicated computer equipped with a 4-core Intel™ processor at 2.5 GHz and a NVIDIA GPU GeForce GTX 640, running Ubuntu with a low latency kernel. All the computation is performed onboard with the above mentioned machines.

B. Active head components

The new active head system shown in Fig. 2(b) is composed of the following components:

- Pan-Tilt Unit (Flir PTU-D46-17). Range pan: $\pm 159^\circ$, Range tilt: $-47^\circ/31^\circ$, Maximum speed: $145^\circ/\text{s}$, Maximum control rate: 60 Hz.
- Inertial Measurement Unit (Microstrain 3DM-GX3-25). Measurements: Linear accelerations, angular velocities, Euler angles, Maximum output rate 1 kHz.
- Color stereo camera (Point Grey Bumblebee). Focal length of 2.5 mm, field of view of 97° , 1024 pixel \times 768 pixel, 20 fps.
- Monocular color camera (Point Grey Firefly). Focal length of 6 mm, field of view of 60° , 640 pixel \times 480 pixel, 60 fps (used only to test and assess the system).

C. Visual motion estimation from optical flow

To assess the performance of our camera stabilization system, we developed an algorithm to visually estimate the horizontal and vertical displacement of the scene with respect to the camera frame of reference. Since we assume a static scene, the effects of impacts and undesired movement on video data can be directly evaluated by measuring the output of the visual motion estimator, as detected velocity.

Many algorithms exist to estimate the motion using vision. They can be grouped mainly with respect to whether they work in the spatial domain or in the frequency domain. Frequency-domain-based motion estimation algorithms detect motions using phase information between the frames, as Based Phase Correlation [33].

On the other hand, time-domain-based motion estimation algorithms find the motions in the spatial domain. We chose the Lucas and Kanade pyramidal algorithm [34] as the most suitable for our case, in terms of time, accuracy and performance. Based on that, our visual motion estimation algorithm has been designed as follows:

- **Feature extraction.** The first step consists of finding features to track between two consecutive images. We used a modified version of the Harris algorithm, extended by Shi and Tomasi [35]. We reduced the computational requirements to keep the fastest rate of 60 Hz using the monocular camera. To do that, we constrained the extraction to 200 features in 9 sub-windows of 91 x 69 pixels covering the whole image.
- **Feature tracking.** Then the features are tracked in consecutive images using the Lucas and Kanade pyramidal algorithm. To optimize the tracking we extended the sub-windows to a size of 171 pixel \times 149 pixel, in order to decrease the probability of a feature leaving the region of interest.
- **Compute optical flow histogram.** After the feature matching, we obtain a Set of Local Optical Flow Vectors (SLOFV) describing the motion. The vectors with a magnitude lower than 1.5 pixel are set to zero and we

compute the histograms. The resolution of the angle histogram is set to 0.5° . If the number of zero vectors is above a certain threshold we consider the Global Optical Vector (GOV) equal to zero.

- **Global motion extraction.** In order to extract the global motion from the histogram, we first remove outliers in magnitude, *i.e.*, histogram bins with too few vectors. Then, we look for a strong peak in the angle distribution (more than 70 % in a single histogram bin). In rare cases where we cannot find one, the optical flow is set to zero. Otherwise, the obtained value is selected as the optical flow direction and the corresponding magnitude is calculated as the average magnitude of all vectors inside this histogram bin.

During our experiments in static environment (*i.e.* without dynamic obstacle) our method was stable and always providing a good estimation of the motion. In case of dynamic changes in the scene, results degrade quickly as this method has been designed for static scene.

Having the global vector estimate, we compute the horizontal and vertical displacement $dx(k)$, $dy(k)$ of camera frame k with respect to the camera frame $k - 1$. We can then compute the angular velocity $\dot{\epsilon}$ (along yaw axis), $\dot{\psi}$ (along pitch axis) with the following derivatives:

$$\dot{\epsilon}(k) = -A \frac{dx(k)}{dt} \quad \dot{\psi}(k) = -A \frac{dy(k)}{dt} \quad (1)$$

A is the parameter (in degree/pixel) to convert from pixels to degrees. It is computed empirically for both the bumblebee and the firefly camera, by rotating the camera using the pan motor, and measuring the values obtained from optical flow. It takes into account the focal length and the size of the CCD chip. As an example for the Firefly camera we calculated 845 pixels for 45 degrees that is to say $A = 45/845$. Finally, we estimate the angular position ϵ , ψ by discrete integration. By using our onboard computer described in Sec.III-A, the camera motion can be estimated at the camera frame rate. Qualitative results are presented in the following section.

D. Validation of our visual motion estimation

In Fig. 3(a) and 3(c), we show the comparison between the angular rate and angular position obtained by our visual motion estimation and the IMU motion estimation. To obtain these results we mounted the Firefly camera on the PTU (Fig. 2(c)) aligned with the IMU, and then moved the robot in random directions. Fig. 3(c) shows that visual motion estimation gives a reasonable approximation of the inertial sensors values. Nonetheless, its performance is affected by a non-neglectable delay that makes it unsuitable for real-time control. This delay is induced by the camera acquisition time and the processing time of the optical flow. It is highlighted in Fig. 3(b). The delay could be reduced using a camera with a faster frame rate but in our case we preferred the IMU more accurate and with a much higher frequency update rate.

It has to be noted that in all the following graphs only the pan axis motion will be presented for clarity. The tilt behaves similarly, just within a smaller range.

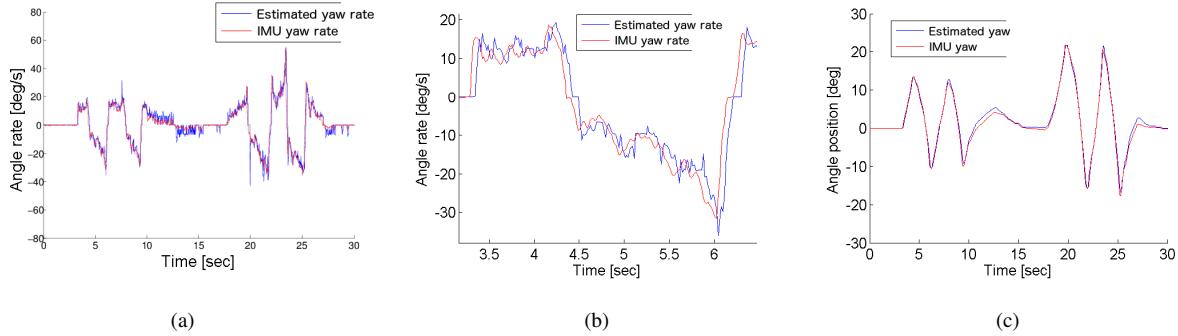


Figure 3: Comparison of the yaw angular rate and position estimated by using our optical based method and obtained by the IMU. (a) Angular rate estimated (blue) and angular rate of IMU (red) (b) Zoom on the previous plot (a) to highlight the delay produced by the estimation based method. (c) Estimated camera motion (blue) and the robot motion (red) when the PTU is off (random robot motion). This figure illustrates that the angular position estimated with the camera is giving a good approximation of the IMU angular position.

IV. MOTION ANALYSIS OF ROUGH TERRAIN ROBOTS DURING NAVIGATION

Motions of large size robots designed to navigate on rough terrains introduce particular issues. Problems appear in particular when dealing with vision-based tasks like state estimation and SLAM. First, navigation on rough terrain usually induces vibrations and large motions (slippage, collision with obstacles). Second, highly dynamic robots induces typical motions (impacts, loss of balance). These important motions produce large differences and blur in the frames. As a consequence they decrease the performance of feature matching algorithms needed in many vision based processes.

The motions of wheeled vehicles have been widely studied in literature as wheeled robot are extensively used by the research community for years. These days, dynamic legged robots are becoming increasingly more prevalent but the challenges are still focused on low-level control, so the high level studies of the motion have not really been done yet.

While the wheeled motion is mainly due to the terrain and to a lesser extend to the frequency of the suspension system, legged motion is both due to gait parameters and to the terrain. In order to outline the particular motions due to rough terrain navigation and above all to legged robot locomotion we will analyze in this section the motions of HyQ for two different gait and we will compare its motions to another large size rough terrain robots: the wheeled robot ARTOR. We will then show the influence of the robot motions on the optical flow to highlight the issues in image processing.

A. The wheeled platform ARTOR

The motion of a wheeled vehicle is very well characterized: longitudinal accelerations are caused by the traction or braking, while the lateral acceleration is caused by the centripetal force when the vehicle is steering.



Figure 4: (a) ARTOR (Autonomous Rough Terrain Outdoor Robot) by ETH Zürich [36]. This robot vehicle has comparable size to HyQ and it is equipped with the same sensors. (b) An image extracted from the dataset in Thun (Switzerland)

The rotation of the vehicle can be separated into orientation (yaw) and inclination (pitch and roll). The orientation depends on steering and speed of the vehicle, while the inclination has two causes: terrain slope and horizontal forces. The frequency of the inclination of the vehicle is conditioned by the suspension system: the natural frequency of the vibrations of the inclination and vertical displacement in road vehicles has typical values between 1 Hz and 2 Hz. This range of frequencies remains within the comfort zone for people. Racing cars have frequencies over 2 Hz, to increase the contact between wheels and ground, and thus increase the general performance of the vehicle. Nevertheless, some vehicles such as forklifts have rigid suspension systems and hence higher natural frequencies.

For our study we analyzed the motion of the Autonomous Rough Terrain Outdoor Robot (ARTOR) from ETH Zürich, a wheeled robot of size similar to HyQ and equipped with comparable sensors (IMU, stereo camera, Velodyne) but without any camera stabilization system [36]. ARTOR is powered by two electrical motors, it has six wheels and a skid-steering system. It is roughly 1 m tall and 1 m long, weighs 350 kg and it has a top speed of 3.5 m/s. It can navigate in rough terrain (see Fig. 4(a)) at a relatively high speed.

The dataset used for our analysis was shared by ETHZ. It contains IMU and stereo camera data recorded in 2013 in Thun (Switzerland), on a military test track for off-road vehicles. ARTOR was navigating at low speed (10 km/h) on a rough road with small rocks and potholes (see Fig. 4(b)). The detailed conditions of these experiments are available on the website [36]. The recorded data covers a distance of about 1 km with a duration of more than 20 minutes. Roughly, all acquired data exhibit similar spectral characteristics, hence, in the following, the analysis will be demonstrated using a representative 40 seconds of the recorded log.

In Fig. 5 we show the angular rate captured by the inertia sensor (XSens MTi IMU) mounted on ARTOR during the off-road experiments, where the values are plotted in the frequency domain for the sake of clarity. As mentioned above, the natural frequency of a wheeled vehicle is between 1 Hz and 2 Hz. Since ARTOR has a rigid suspension system, higher frequencies are also present in the spectrum. In particular, there are two characteristic peaks, one

below 1 Hz and the other around 2 Hz for roll and yaw, and 2.5 Hz for pitch.

B. The legged platform HyQ

On legged platforms, the motion strongly depends on the selected gait. We show in this Section the HyQ's motions for two different gaits on a flat stiff terrain: a walk and a trot. Fig. 6(a) shows the frequencies of the angular rate of HyQ during a walking gait with a peak around 3.5 Hz. In Fig. 6(b) we show the same angular rate representation for the trotting gait on flat and stiff terrain (outdoor). Although the terrain is the same for the two gaits, the spectrum is totally different. For the walking gait we obtained strong peaks in the multiples of 0.82 Hz, a fraction of the trotting frequency. In the case of the trotting the motion has higher frequency components than during a walk and the intensity start decreasing from around 10 Hz. Note that during the experiments the camera was fixed on the PTU, with the controller off and the PTU fixed at the reference position.

C. A camera motion comparison between a wheeled and a legged robot

As explained above, the motions of a wheeled off-road robot and the two characteristic gaits of HyQ differ. In particular, on HyQ the frequency of motion is higher than the wheeled robot ARTOR even with a walking gait. The comparison with locomotion on flat terrain (asphalt) has been chosen to highlight the motion that is naturally induced by the robot gait, regardless of the ground morphology. On a rougher terrain, HyQ motions would just be amplified at lower frequencies.

To investigate the consequences on the image quality, we measured the motion from our optical flow based method on both platforms. In order to compare the camera motion between HyQ and ARTOR under the exact same conditions (*i.e.*, same camera, same frame rate and resolution) we emulate the motion of ARTOR directly on HyQ by adapting the algorithm from [37]. The roll and pitch angles of ARTOR are reproduced by HyQ, while the yaw is high pass filtered. In practice this means that the angular velocity of ARTOR is reproduced with a good level of fidelity. The linear acceleration however has to be high-pass filtered and converted to position. This is because we are doing the reproduction of the movement in place. Using this method we lose the low frequency movements of ARTOR, but these are the ones which have a smaller effect on the optical flow.

In Fig. 7 we show the GOV magnitude (see Sec. III-C) on HyQ during walking and trotting, and HyQ replaying the recorded ARTOR's movements. The figure shows that HyQ's trot and walk create more motion on the camera than wheeled navigation. The average magnitude of the optical flow is around 8 pixels for HyQ navigating on a flat terrain and below 3 for the ARTOR robot navigating on a rough terrain. This experiment attest the particularly high motions induced by our highly dynamic robot.

V. CAMERA STABILIZATION

As mentioned before, camera stabilization on highly dynamic legged platforms is important to improve the performance of all the image processing required for high level computation. By minimizing the frame by frame variations it reduces the loss of pose tracking or object tracking. We chose to mechanically compensate for the

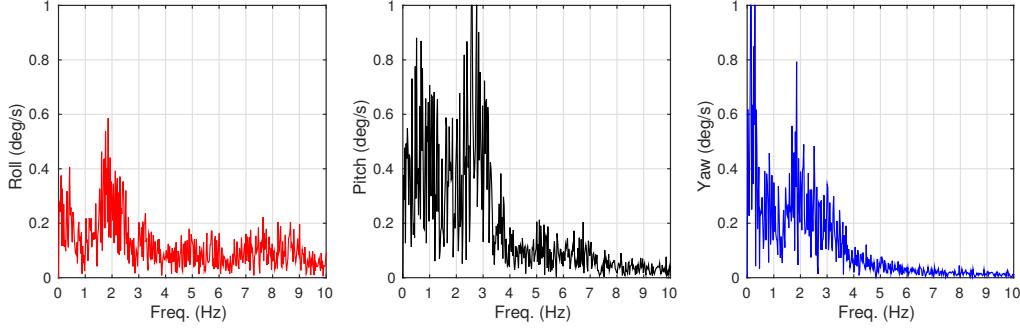


Figure 5: Frequency representation of the angular rate on the wheeled vehicle ARTOR during off-road experiments. We can see that the characteristic frequencies are around 2 Hz for roll and yaw, and 2.5 Hz for the pitch.

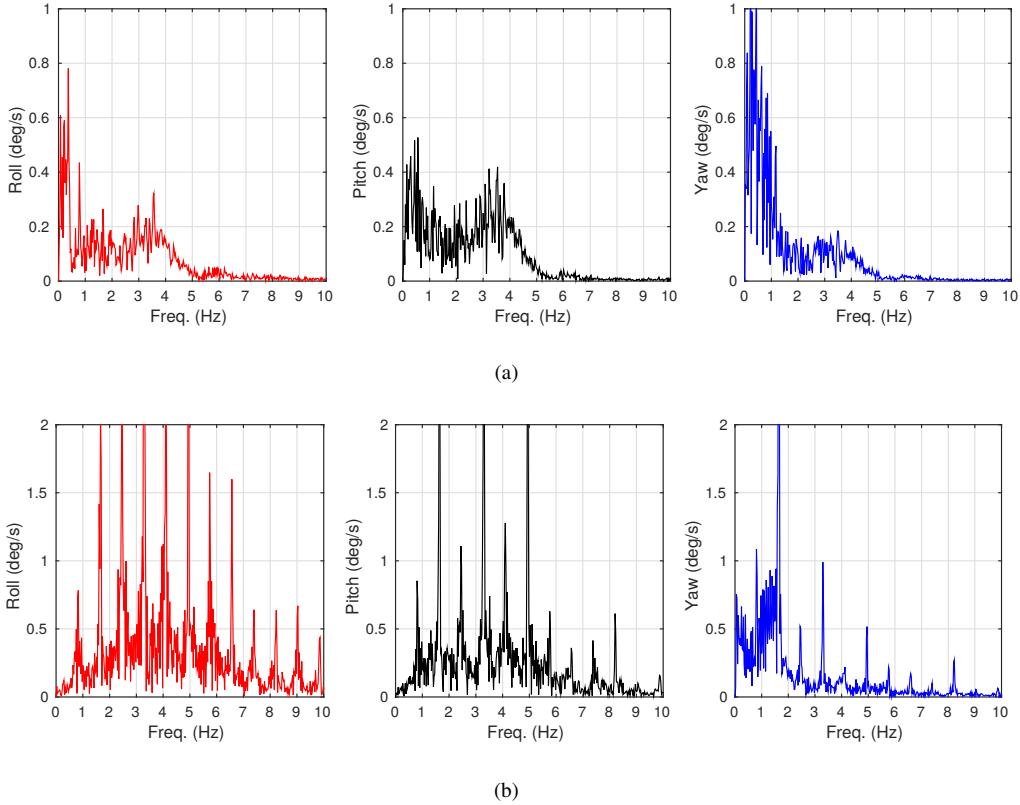


Figure 6: (a) Frequency representation of the angular rate on HyQ during walking. The characteristic frequencies are higher than in ARTOR with a peak around 3.5 Hz (b) Frequency representation of the angular rate on HyQ during trotting. We can see peaks in the multiples of 0.82 Hz, a fraction of the trotting frequency.

motion using the IMU data as an input. Another possible input for the stabilization could have been the visual input, however it was not suitable for our purposes because it had a considerable delay (see Sec. III-C). The IMU based

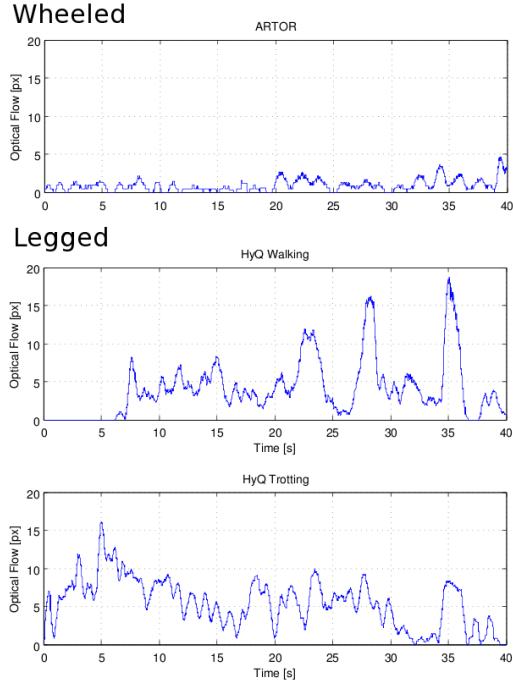


Figure 7: Optical flow magnitude on ARTOR navigating and on HyQ walking and trotting. These graphs correspond to the motions shown in Fig. 5, 6(a), 6(b). The two different gaits of HyQ show higher optical flow than ARTOR.

motion is a lot faster and we will show in this section how the motion of the camera is smoothed while the robot is trotting.

A. Robot motion and PTU speed

To ensure the PTU motors are fast enough to compensate for the angular oscillations produced by HyQ's motion we identify the PTU system using the Empirical Transfer Function Estimate (ETFE) method [38].

The motor bandwidth has been estimated around 18 Hz. As explained in the previous section, the main robot motions do not exceed 10 Hz and are strong only below 6 Hz. The selected PTU is thus sufficiently fast to deal with the motion disturbances on HyQ.

B. Controller design

To control the PTU motors we implemented a PD controller, which has been used for most of the active head systems, for example Gasteratos *et al.* [21]. For each axis of the PTU we close the feedback loop in position, reading the actual angular position from the motor encoder, and subtracting it from the reference signal to compute the error $e(k)$:

$$e(k) = \theta^{\text{ref}}(k) - \theta(k) \quad (2)$$

The error $e(k)$ feeds into the PD block, which computes the angular velocity for the motors. In the control scheme (see Fig. 8) a feedforward term based on the angular velocities of the IMU has been added to the PD output to enhance the performance of the camera stabilization.

C. Background execution

The PTU is a resource shared between several high-level algorithms. In our solution, we decided to put the position inputs coming from higher modules into the variables $\theta^{\text{IN}}, \phi^{\text{IN}}$, defined as the “external” input. This allows other processes to control the PTU while the stabilization is still active. For instance, an external module, *e.g.*, active mapping, tracking or recognition, can concurrently use the PTU with the stabilization system to achieve a more intelligent behavior.

The camera stabilization input is accounted for separately and is directly added to the external input to form a single pair $(\theta^{\text{ref}}, \phi^{\text{ref}})$ for the controller (see Eq. 3).

In the following, α, β, γ will refer to the angle positions of roll, pitch and yaw and $\dot{\alpha}, \dot{\beta}, \dot{\gamma}$ to their corresponding rates (*i.e.*, angular velocities). We will denote with θ, ϕ the angular positions of pan and tilt joints, respectively.

D. IMU based stabilization

The goal of the stabilization is to decouple the camera orientation from the body orientation. This system tracks the orientation of the camera frame to keep it aligned with the orientation of the world frame. The IMU measurements provide roll, pitch and yaw of the body with respect to the world frame, α, β, γ . The pan and tilt angles must respect the following constraint:

$$\begin{bmatrix} \theta \\ \phi \end{bmatrix}^{\text{ref}} = \begin{bmatrix} \theta \\ \phi \end{bmatrix}^{\text{in}} - \begin{bmatrix} \theta \\ \phi \end{bmatrix}^{\text{mc}} = \begin{bmatrix} \theta \\ \phi \end{bmatrix}^{\text{in}} - \begin{bmatrix} 0 & 0 & 1 \\ \sin(\theta) & \cos(\theta) & 0 \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix} \quad (3)$$

The block diagram of the overall control system is presented in Fig. 8, where $T(\theta) = [\sin(\theta) \cos(\theta)]^T$ and K the feedforward gain. The latter has been decreased below 1 to keep a margin of compensation for the controller.

In Fig. 9(a) and 9(b), we can see that the camera motion is reduced by the camera stabilization. The first experiment shows a random robot motion, the second one a trot.

VI. EXPERIMENTAL RESULTS

In this section we describe the experiments performed to assess our stabilization system running while executing two tasks with the robot: tracking a colored object while trotting, and mapping the environment while trotting. In both cases the trotting velocity was about 0.2 m/s.

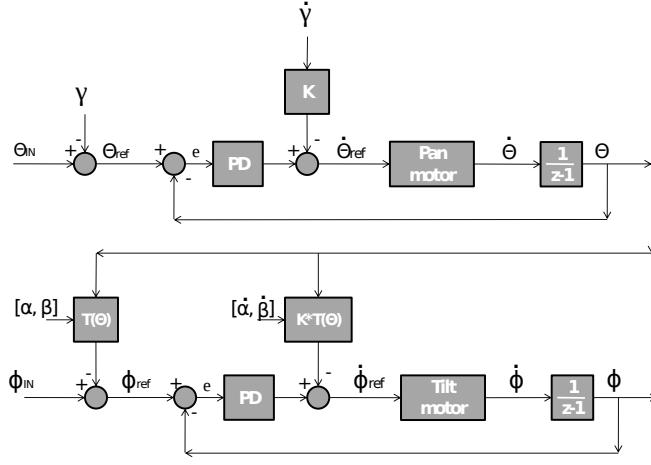


Figure 8: Control loop schema. α, β, γ are respectively roll, pitch, yaw measured from IMU. θ, ϕ are the PTU joint angles, pan and tilt.

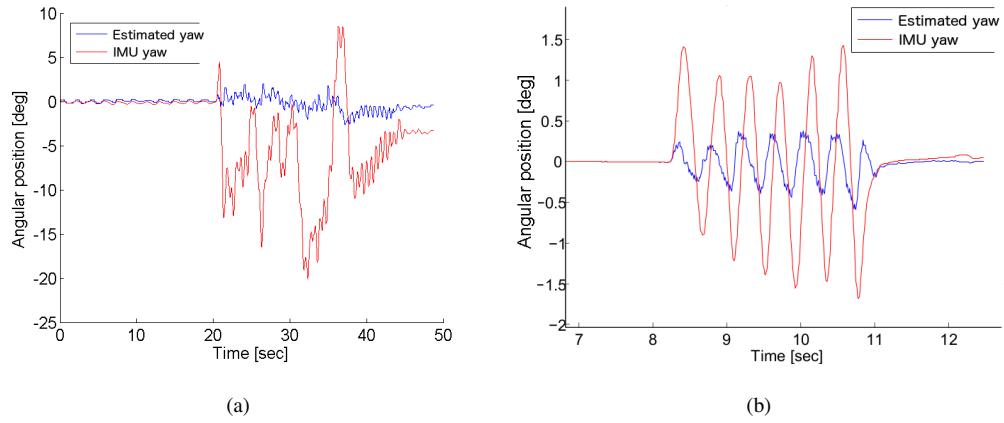


Figure 9: (a) Estimated camera motion along yaw axis (blue) and robot motion along yaw axis (red) during random moves with the PTU stabilization running. The motion of the camera is obtained by using our visual motion estimation, and the robot motion is obtained with the IMU. Compared to Fig. 3(c) where the PTU stabilization was off, the estimated camera motion is drastically reduced. (b) Estimated camera motion along yaw axis (blue) and robot motion along yaw axis (red) during trotting with the PTU stabilization running.

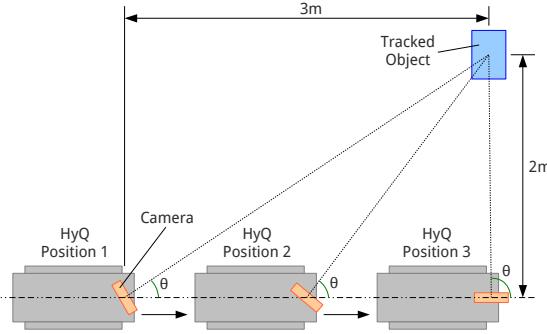


Figure 10: Schematic representation of the tracking experiments. As the robot trots in a straight line, the camera turns to track the object.

A. Color tracking

For the color tracking we used a modified version of the Continuously Adaptive Mean Shift (CAMShift) algorithm [39]. CAMShift combines *Mean Shift* algorithm with an adaptive region-sizing step. The algorithm builds a confidence map in the new image based on the color histogram of the object in the previous image, and uses the Mean Shift to find the peak of a confidence map near the old position of the object.

The color is represented using the Hue component of the HSV color model as this color space is more consistent than the standard RGB color space under illumination changes. In our implementation, as Hue is unstable at low saturation, the color histograms do not include pixels with saturation below a threshold.

Experimental setup: The tracking experiments were performed outdoor on a 25 m long track with the monocular camera setup (Fig. 2(c)) at a frame rate of 60 Hz.

As shown in Fig. 10, the object was placed around 3 m away from the robot's starting point and 2 m on the left of the longitudinal axis. The robot was trotting straight on a flat outdoor terrain (asphalt) at 0.2 m/s while the camera was tracking the object. Then, it was stopped after passing the object, *i.e.*, when the camera reached a pan angle greater than 100°.

Results: Fig. 11 illustrates the improvement of the tracking by using the PTU compensation. Without stabilization the pan motion is stronger, the differences between consecutive frames are bigger and the corrections to keep the object in the center of the image are larger. With compensation, the pan motion is smooth, as the image was already decoupled from the body motion, producing smaller differences between consecutive frames.

The experiment shows the benefit of the stabilization on a tracking task. It is worth mentioning that the tracking is working on top of the stabilization (see Sec. V-C), as a completely decoupled process; this capability is important to keep the PTU available for other tasks, *e.g.*, active scanning and object tracking.

B. Mapping

To build maps of the environment we used a stereo camera and a modified version of the RGB-D SLAM method [40] to register the point clouds. For the depth extraction from stereo, we opted for the Sum of Absolute Differences

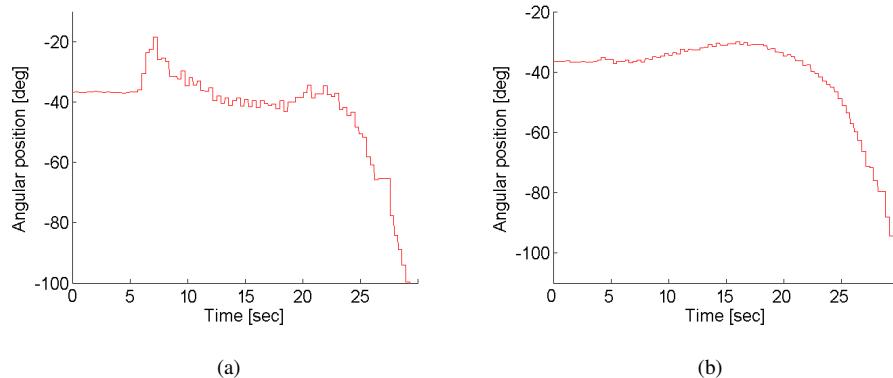


Figure 11: Tracking improvements when using the PTU stabilization system: (a) tracking while trotting without stabilization, (b) tracking while trotting with the camera fixed.

(SAD) correlation based-method because of the small computational load, which makes it suitable for onboard real-time implementations. We apply the SAD on edge images and we used sub pixel interpolation [41] with a matching mask of $21 \text{ pixel} \times 21 \text{ pixel}$.

Then, the disparity map was spatially and temporally filtered to fill the gaps and remove speckle noise. Finally for each valid disparity pixels we estimate a 3D position. In our system, a point cloud with 640 pixel \times 480 pixel 3D points with their associated RGB values can be computed at 10 Hz using four threads. On top of that the RGB-D SLAM method uses three processing steps to build a consistent 3D model.

- 1) Speeded Up Robust Features (SURF) are extracted from the incoming color images and matched against features from the previous image;
 - 2) Then, by evaluating the point clouds at the location of these feature points, a set of point-wise 3D correspondences between two frames is used to estimate the relative transformation using the RANdom SAMpling Consensus (RANSAC) algorithm;
 - 3) The third step involves a refinement of the initial estimation with the Iterative Closest Point (ICP) algorithm.

Experimental setup: The mapping experiments were carried out using the Bumblebee stereo camera with the Firefly mounted on top for camera motion assessment (see 2(b)).

To evaluate the quality of the maps generated with our system, we carried out several tests with the robot trotting at 0.2 m/s on a flat surface with 11 box-shaped obstacles of varying sizes lying on the robot's path (see Fig. 12(a) and 12(b)).

Since path-planning or foot placement control are beyond the scope of this work, the experiments were executed on flat ground. Maps were computed onboard and were transformed into a height map with a resolution of 1 cm to make them easily comparable with each other. For these experiments the tilt angle input was set to 10° down.

Fig. 13 and 14 depict the generated maps (color and height respectively) for the six trials. We measured the position and dimension of the obstacles to generate a ground truth height map. As we shown in Fig. 12(b), the test

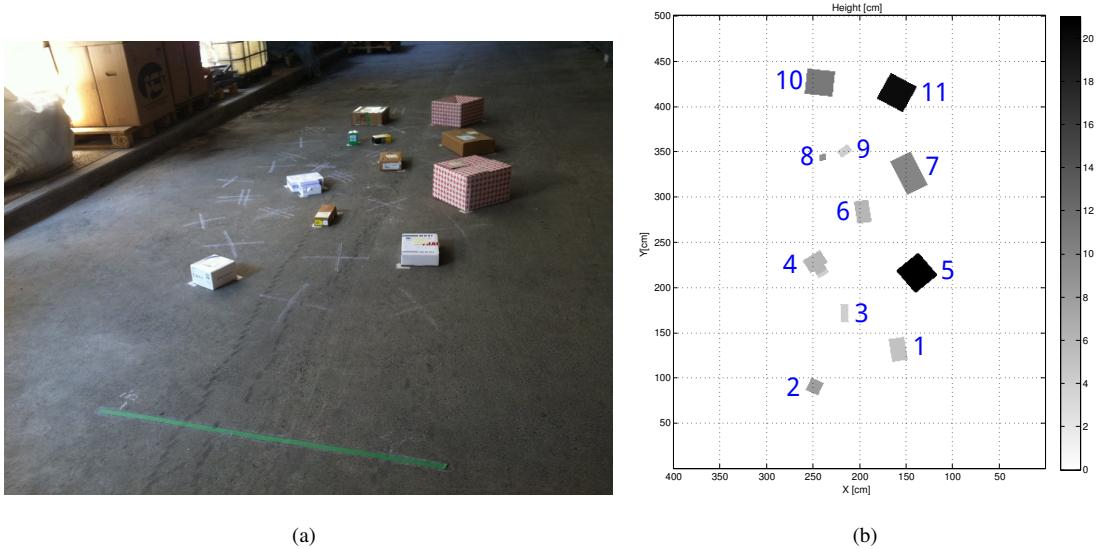


Figure 12: (a) Set up for the mapping experiment. 11 boxes of different sizes were placed on the robot route. The corresponding ground truth height map is shown in Fig. 12(b). (b) Ground truth height map used for mapping evaluation. This map was obtained by measuring the dimensions and positions of the obstacles. All dimensions are expressed in [cm], from 0cm (white) to 20cm (black).

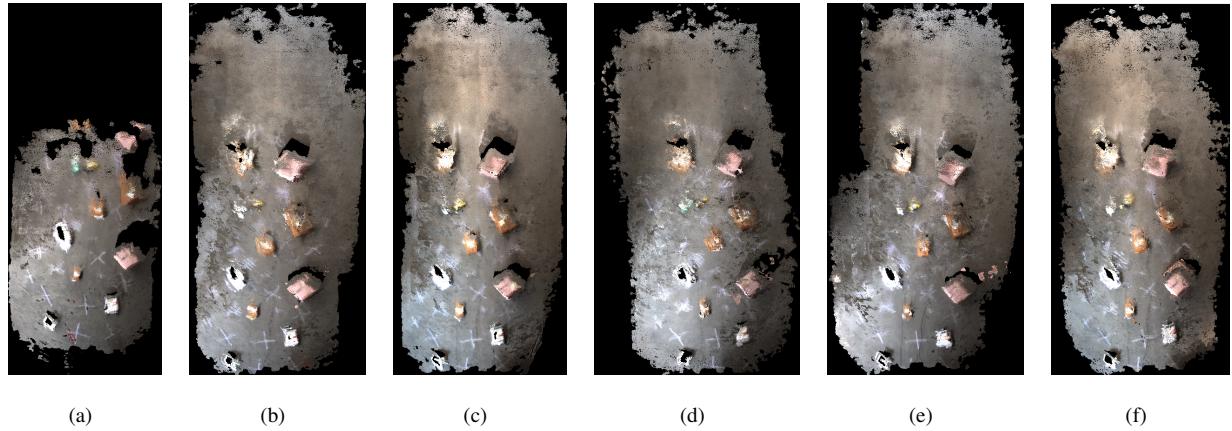


Figure 13: 2D maps with color information. (a) (b) (c) the mapping was performed with stabilization running, (d) (e) (f) the mapping was performed with the camera fixed.

scenario has a dimension around $4\text{ m}^2 \times 5\text{ m}^2$ and it contains obstacles with a height between 4 cm and 21 cm.

Results: The height map extracted from the point cloud takes the camera starting position as a reference. To be comparable with the ground truth, we calculated the height map from the map and we estimated the ground level taking the most repeated value (since the ground represents the biggest surface in the scenario). Then we inverted the values to have the height expressed as distance from the ground plane to the top of the obstacle (the same

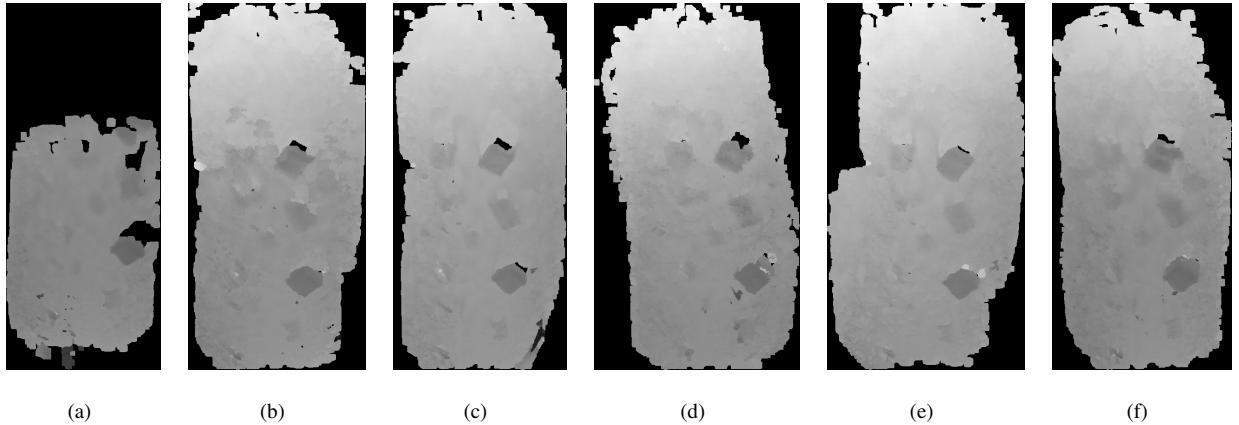


Figure 14: 2D height maps. (a) (b) (c) the mapping was performed with stabilization running, (d) (e) (f) the mapping was performed with the camera fixed.

reference as the ground truth) and we aligned the height map with the ground truth applying the required rotation and translation (the rotation was calculated empirically, and the translation was computed using a two dimensional correlation algorithm to match the corners of the obstacles with the ones of the ground truth).

Table I summarizes the errors (Sum of Squared Differences) between 6 trials and the ground truth. Each time we compute errors on different regions: ground and obstacles 1-11. Trials *a*, *b* and *c* were performed using our active stabilization, while *d*, *e* and *f* were performed without. The blue text indicates the map trial with the smallest error, and the red text the one with the maximum error. The last line shows the average error on the 11 obstacles for each trial. The two values presenting a star in the trial A are biased as the map is cut before the end. As a consequence a part of the ground and the obstacle 10 cannot be properly compared. We decided to show this map (Fig. 13(a) and 14(a)) to illustrate a loss of tracking mentioned previously in the paper. During experiment the robot fell down at the end because of a system failure on the control computer. In a case such as this, it is usually impossible even with stabilization to maintain the visual localization as the robot was on the ground and the camera nearly occluded. The results listed in the table demonstrate how in most of the cases the trials with the pan and tilt unit have the smallest errors, while the trials without show the maximum error values.

Note that the experiments have been performed on flat terrain removing the obstacles before the robot slips or walks on them. This means that the robot motions were smaller than in real experiments on rough terrain. On rough terrain with slippages and loss of balance the stabilization should be even more effective. These experiments on rough terrain outdoor will be explored in the near future.

To highlight the differences between the two cases, we also provide the mean error of the trials listed in Table I.

The results depicted in Fig. 15 show the average errors over 3 trials for the ground plane, the map, and almost all the obstacles. Except for obstacle n.5 which is slightly worse, our stabilization system improves the mapping in any case.

Table I: Average errors for the generated height maps in [cm] on 6 trials (a-f). Stabilization vs. Fixed camera. Blue and red values indicates the best and the worst performance within the set of trials, respectively; starred values are from the incomplete map and are not taken into account for the comparison.

Trial	Stabilization			Fixed camera		
	a	b	c	d	e	f
Ground	2.5*	15.6	14.9	16.3	16.6	15.8
1	1.4	2.0	2.3	3.8	3.0	1.7
2	6.0	8.3	7.7	7.0	7.2	8.8
3	2.1	2.5	1.7	4.1	3.1	2.6
4	3.2	4.5	7.7	6.8	10.3	3.9
5	3.7	1.6	1.2	1.7	1.1	2.7
6	1.2	2.0	5.6	5.5	7.8	1.8
7	2.4	2.2	4.5	4.7	6.1	2.3
8	6.4	4.7	8.4	13.4	13.0	4.9
9	3.4	3.9	9.2	10.2	13.4	3.9
10	11.3*	17.6	19.0	21.3	25.4	13.7
11	8.4	8.1	12.7	16.0	14.8	8.5
Mean						
obstacles	4.5	5.2	7.2	8.6	9.6	5

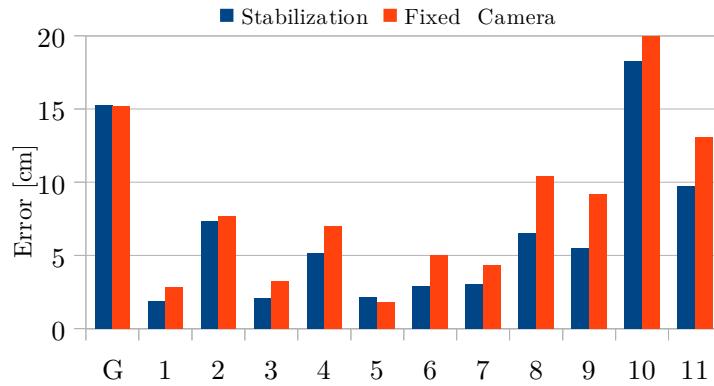


Figure 15: Average errors obstacle by obstacle for the generated height maps. The error is expressed in cm on the vertical axis, while on the horizontal axis the ground (G) and the obstacles are listed. We can estimate from this graph an average improvement of 25%.

Another criterion we used to show the improvement of the mapping while using the active compensation is assessing the accuracy in the reconstruction of obstacle shapes. Inaccurate matches generate small drifts that distort the shapes or cause ghost replicas. Fig. 16 depicts a detail of obstacle 5 (color and height map) in all the six trials. The top three figures are the trials with compensation, and the bottom ones without it. The reconstructed maps of

Table II: Failure rate over 10 trials. Left: Average failure rates with stabilization (S) and with Fixed Camera (FC); Right: Failures of the SLAM system over 10 trials with stabilization (S) and with Fixed Camera (FC). The improvement is greater than 50%.

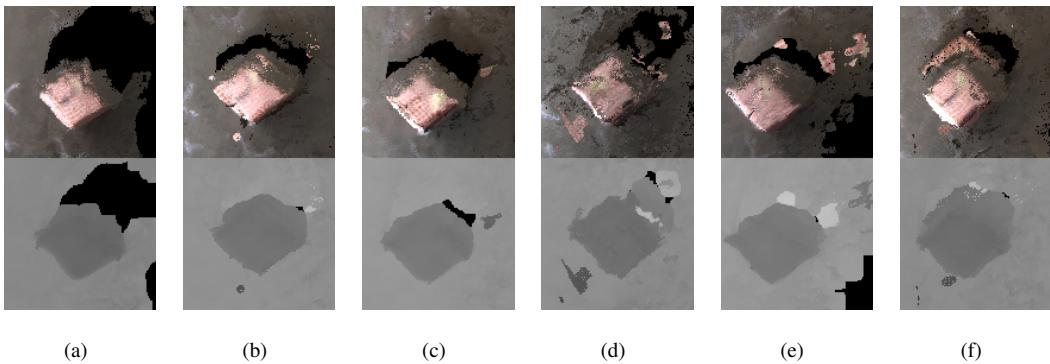
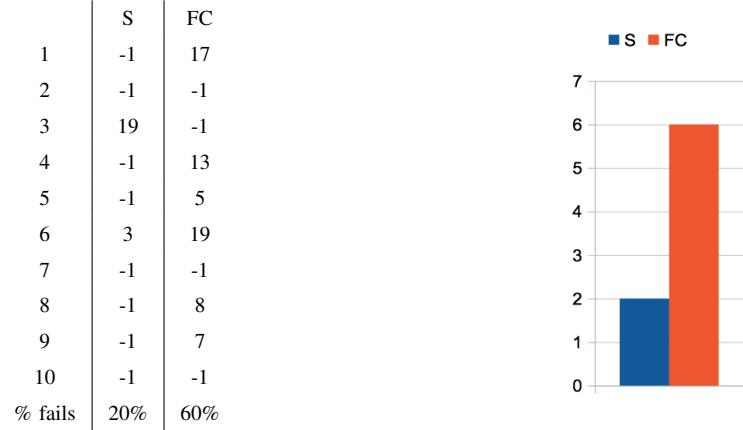


Figure 16: Detail of obstacle 5 with the color image on the top and heightmap in the bottom. (a) (b) (c) the mapping was performed with stabilization running, (d) (e) (f) the mapping was performed with the camera fixed. The trials without stabilization present more artifacts.

the trials including stabilization are more accurate, while the non-stabilized maps contain artifacts. As we show, the stabilization improves the accuracy of the map, but this is not the only advantage of such a system. Another set of experiments has been done to show the significant reduction of mapping failures using the stabilization.

Table II shows the failure rate of the mapping pipeline with and without the pan and tilt unit stabilization over 20 trials. As mentioned above, an example of the loss of tracking problem is shown in Fig. 13(a) and 14(a). During the experiments we built maps while trotting and in case of failure we recorded the time when the tracking was lost. We recorded also the relative transformation between two clouds at a processing speed of 10 Hz, and we checked for a loss of tracking event. A loss of tracking was triggered if the transformation computed by the ICP was above a threshold of 0.1 rad along one axis between the two clouds.

When a loss of tracking was detected the elapsed time and the map were saved. In this table, -1 stands for a mapping completed without failure, a value greater than 0 stands for the seconds elapsed before the loss of tracking. These results show that the stabilization decreases the failure rate or simply helps to maintain the consistent mapping longer. At the same frequency the same SLAM system is more robust when the stabilization is applied. In the future we plan to use an FPGA to compute the 3D point cloud onboard at 25 Hz, this combined with our camera stabilization will help to perform some fully online planned tasks.

C. Discussion

In the two previous sections, we presented the results of image processing computed during trotting with and without active compensation. As expected, the stabilization reduces the shift between consecutive frames and enhances the image processing algorithms in terms of reliability and accuracy. Apart from the qualitative results presented above, we observed how the use of active compensation reduces the rate of failures, (*i.e.*, lost tracking).

As a robust behavior is crucial for legged robotic navigation, and since the state-of-art methods in image processing are not sufficient to be used on a highly dynamic legged platform like HyQ, we believe that the development of a mechanical motion compensation system is a required feature for legged robot SLAM, which is expected to work even in the case of slippages and falls.

VII. CONCLUSION AND FUTURE WORK

In this paper, we presented the integration and the development of an actively stabilized vision system on the quadruped robot HyQ. The new vision system is composed of a wide-angle stereo camera and a monocular high frame-rate camera (added for assessment purposes), mounted on a fast pan and tilt unit (PTU). The PTU enhances the capabilities of the vision system, since it enables (i) gaze shifting to a specific area (*e.g.*, to extend a 3D map of the environment in front of the robot), (ii) to track an object while the robot is in motion, (iii) to compensate for the robot motion.

We analyzed to what extent the visual data stream is affected by the motion of HyQ during locomotion. The walking and trotting rhythmic movements, in combination with fast and irregular motions due to external disturbances on the robot, lead to large differences in consecutive images, which decreases the quality of the extracted features and makes the localization or tracking inaccurate.

To solve this problem, we opted for a mechanical image-stabilizing system, which is based on the input of the onboard IMU data that controls a PTU where the stereo camera is attached to. The developed control loop is based on a PD controller, it works in hard real-time, and other vision algorithms can be used on top of it, as it is implemented as a completely independent process. In order to validate the improvement on visual processing we performed two kind of experiments: object tracking and 3D mapping.

We showed how our system improves the accuracy of the generated maps by 25% and reduced the failures by more than 50%. In the future, we will use the generated height map to perform path and foothold planning in order to autonomously navigate on unknown rough terrain. Afterwards, we would like to focus on improving the

robustness of the mapping in cases of dynamics changes in the scene: for instance a person crossing the robot path or an obstacles moved by the robot.

ACKNOWLEDGMENTS

The authors would like to thank Philipp Krusi and Paul Furgale from the Autonomous System Lab (ASL) at ETH Zürich for providing us with a dataset of the ARTOR project [36]. We also thank Jonas Buchli from ETH Zürich for his valuable advice in the early stages of this work. Successful experiments on HyQ are the fruit of many people's contributions during the last years. We also thank Victor Barasuol, Michele Focchi, Marco Frigerio, Jake Goldsmith and Ioannis Havoutis, from DLS Lab in IIT, Genova. This research has been funded by the Fondazione Istituto Italiano di Tecnologia.

REFERENCES

- [1] C. Semini, N. G. Tsagarakis, E. Guglielmino, M. Focchi, F. Cannella, and D. G. Caldwell, "Design of HyQ - a hydraulically and electrically actuated quadruped robot," *Journal of Systems and Control Engineering*, vol. 225, no. 6, pp. 831–849, 2011.
- [2] 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 *IEEE International Conference on Robotics and Automation (ICRA)*, 2013.
- [3] I. Havoutis, C. Semini, J. Buchli, and D. G. Caldwell, "Quadrupedal trotting with active compliance," *IEEE International Conference on Mechatronics (ICM)*, 2013.
- [4] M. Focchi, V. Barasuol, I. Havoutis, J. Buchli, C. Semini, and D. G. Caldwell, "Local reflex generation for obstacle negotiation in quadrupedal locomotion," in *Int. Conf. on Climbing and Walking Robots (CLAWAR)*, 2013.
- [5] S. Bazeille, V. Barasuol, M. Focchi, I. Havoutis, M. Frigerio, J. Buchli, C. Semini, and D. G. Caldwell, "Vision enhanced reactive locomotion control for trotting on rough terrain," in *IEEE International Conference on Technologies for Practical Robot Applications (TePRA)*, 2013.
- [6] I. Havoutis, J. Ortiz, S. Bazeille, V. Barasuol, C. Semini, and D. G. Caldwell, "Onboard perception-based trotting and crawling with the hydraulic quadruped robot (hyq)," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2013.
- [7] A. Winkler, I. Havoutis, S. Bazeille, J. Ortiz, M. Focchi, R. Dillmann, D. G. Caldwell, and C. Semini, "Path planning with force-based foothold adaptation and virtual model control for torque controlled quadruped robots," in *IEEE ICRA*, 2014.
- [8] J. Z. Kolter, K. Youngjun, and A. Y. Ng, "Stereo vision and terrain modeling for quadruped robots," in *Robotics and Automation, IEEE International Conference on*, 2009, pp. 1557 –1564.
- [9] P. Filitchkin and K. Byl, "Feature-based terrain classification for littledog," in *Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on*, 2012, pp. 1387 –1392.
- [10] A. Stelzer, H. Hirschmüller, and M. Görner, "Stereo-vision-based navigation of a six-legged walking robot in unknown rough terrain," *The International Journal of Robotics Research*, vol. 31, no. 4, pp. 381–402, 2012.
- [11] J. Ma, S. Susca, M. Bajracharya, L. Matthies, M. Malchano, and D. Wooden, "Robust multi-sensor, day/night 6-dof pose estimation for a dynamic legged vehicle in gps-denied environments," in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, 2012, pp. 619–626.
- [12] D. Wooden, M. Malchano, K. Blankespoor, A. Howard, A. A. Rizzi, and M. Raibert, "Autonomous navigation for bigdog," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*. IEEE, 2010, pp. 4736–4741.
- [13] M. Bajracharya, J. Ma, M. Malchano, A. Perkins, A. Rizzi, and L. Matthies, "High fidelity day/night stereo mapping with vegetation and negative obstacle detection for vision-in-the-loop walking," in *IEEE/RSJ IROS*, 2013.
- [14] X. Shao, Y. Yang, and W. Wang, "Obstacle crossing with stereo vision for a quadruped robot," in *Mechatronics and Automation (ICMA), International Conference on*, 2012, pp. 1738 –1743.
- [15] P. Rawat and J. Singh, "Review of motion estimation and video stabilization techniques for hand held mobile video," *International Journal of Signal and Image Processing (SIPIJ)*, 2011.

- [16] R. Karazume and S. Hirose, "Development of image stabilization system for remote operation of walking robots," *Proc. IEEE Int. Conf. on Robotics and Automation*, 2000.
- [17] M. Marcinkiewicz, M. Kunin, S. Parsons, E. Sklar, and T. Raphan, "Towards a methodology for stabilizing the gaze of a quadrupedal robot," ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2007, vol. 4434, pp. 540–547.
- [18] M. Marcinkiewicz, R. Kaushik, I. Labutov, S. Parsons, and T. Raphan, "Learning to stabilize the head of a walking quadrupedal robot using a bioinspired artificial vestibular system," *Proc. IEEE Int. Conf. on Robotics and Automation*, 2008.
- [19] J. Giesbrecht, P. Goldsmith, and J. Pieper, "Control strategies for visual tracking from a moving platform," in *Proc. of Canadian Conference on Electrical and Computer Engineering (CCECE)*, 2010, pp. 1 –7.
- [20] B. Jung and G. Sukhatme, "Real-time motion tracking from a mobile robot," *International Journal of Social Robotics*, vol. 2, pp. 63–78, 2010.
- [21] A. Gasteratos, "Tele-autonomous active stereo-vision head," *International Journal of Optomechatronics*, vol. 2, pp. 144–161, 2008.
- [22] T.-K. Kang, H. Zhang, and G.-T. Park, "Stereo-vision based motion estimation of a humanoid robot for the ego-motion compensation by type-2 fuzzy sets," in *Proc. of the IEEE Int. Symposium on Industrial Electronics (ISIE)*, 2009, pp. 1785 –1790.
- [23] A. Ude and E. Oztop, "Active 3-d vision on a humanoid head," in *Proc. of the IEEE Int. Conference on Advanced Robotics (ICAR)*, 2009.
- [24] B. Bauml, O. Birbach, T. Wimbock, U. Frese, A. Dietrich, and G. Hirzinger, "Catching flying balls with a mobile humanoid: System overview and design considerations," in *International Conference on Humanoid Robots*, 2011, pp. 513 –520.
- [25] G. Xiang, "Real-time follow-up tracking fast moving object with an active camera," in *Proc. of the Int. Congress on Image and Signal Processing (CISP)*, 2009, pp. 1 –4.
- [26] M. Akhloufi, "Real-time target tracking using a pan and tilt platform," in *Proc. of the Int. Conference on Machine Vision, Image Processing, and Pattern Analysis (MVIPPA)*, 2009, pp. 437–441.
- [27] K. Huebner, K. Welke, M. Przybylski, N. Vahrenkamp, T. Asfour, D. Kragic, and R. Dillmann, "Grasping Known Objects with Humanoid Robots: A Box-based Approach," in *Proc. of the Int. Conference on Advanced Robotics*, 2009, pp. 1–6.
- [28] S. Gay, A. Ijspeert, and J. Santos-Victor, "Predictive gaze stabilization during periodic locomotion based on adaptive frequency oscillators," in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, May 2012, pp. 271–278.
- [29] T. Shibata and S. Schaal, "Biomimetic gaze stabilization based on feedback-error-learning with nonparametric regression networks," *Neural Networks*, vol. 14, no. 2, pp. 201–216, 2001.
- [30] F. Panerai, G. Metta, and G. Sandini, "Learning visual stabilization reflexes in robots with moving eyes," *Neurocomputing*, vol. 48, no. 14, pp. 323 – 337, 2002.
- [31] "Ocular Robotics." [Online]. Available: <http://www.ocularrobotics.com/products/camera-pointing-devices/rev25-st/>
- [32] "FLIR." [Online]. Available: <http://www.flir.com/mcs/view/?id=53679>
- [33] O. Kwon, J. Shin, and J. Paik, "Video stabilization using kalman filter and phase correlation matching," in *Image Analysis and Recognition*. Springer Berlin Heidelberg, 2005, vol. 3656, pp. 141–148.
- [34] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," 1981, pp. 674–679.
- [35] J. Shi and C. Tomasi, "Good features to track," in *Computer Vision and Pattern Recognition, International Conference on*, 1994, pp. 593–600.
- [36] "ARTOR (Autonomous Rough Terrain Outdoor Robot)." [Online]. Available: <http://www.artor.ethz.ch/>
- [37] J. Ortiz, C. Tapia, L. Rossi, J. Fontaine, and M. Maza, "Description and tests of a multisensorial driving interface for vehicle teleoperation," in *Intelligent Transportation Systems, 2008. ITSC 2008. 11th International IEEE Conference on*. IEEE, 2008, pp. 616–621.
- [38] R. Aarts, *System Identification and Parameter Estimation*, 12nd ed., 1998, vol. 1.
- [39] G. Bradski, "Computer video face tracking for use in a perceptual user interface," *Intel Technology Journal*, 1998.
- [40] F. Endres, J. Hess, N. Engelhard, J. Sturm, D. Cremers, and W. Burgard, "An evaluation of the rgbd slam system," in *Robotics and Automation, IEEE International Conference on*, 2012, pp. 1691–1696.
- [41] R. Szeliski and D. Scharstein, "Symmetric sub-pixel stereo matching," in *ECCV 2002*. Springer, 2002.