

NAO Humanoid Robot Obstacle Avoidance Using Monocular Camera

Walaa Gouda^{*1}, Randa Jabeur Ben Chikha²

¹Computer and Engineering Networks Department, Jouf University, Saudi Arabia

²Computer Science Department, Jouf University, Saudi Arabia

ARTICLE INFO

Article history:

Received: 07 December, 2019

Accepted: 08 January, 2020

Online: 10 February, 2020

Keywords:

Motion planning

Humanoid

Object recognition

ABSTRACT

In this paper, we present an experimental approach that allows a humanoid robot to effectively plan and execute whole body motions, like climbing obstacles and straight stairs up or down, besides jumping over obstacles using only on-board sensing. Reliable and accurate motion sequence for humanoid employed in complex indoor environments is a necessity for high-level robotics tasks. Using the robot's own kinematics will construct complex dynamic motions. A series of actions to prevent the object from being performed on the basis of the identified object from the database of the robot, obtained using the robot's own monocular camera. As shown in real world experiments beside simulation using NAO H25 humanoid, the robot can effectively perform whole body movements in cluttered, multilevel environments containing items of various shapes and dimensions.

1 Introduction

Humanoid robots have become a common research platform because they are considered the future of robotics because of their creativity. Humanoid robots, however, are fragile mechanical robotic systems and the main challenge is to maintain the balance of the robot [1]. Human like design and locomotion require complex motions to be performed by humanoids. Humanoids are well adapted for human-designed mobile manipulation activities such as walking, reaching various types of terrain, moving in complex environments such as environments with stairs and/or narrow passages, navigating in cluttered environments without colliding with any barriers, etc. Such capabilities would make humanoid robots perfect assistants to humans, such as disaster management or housekeeping [2]–[5].

Humanoid service robots need to deal with a variety of objects. For example, avoiding objects by turning away from them, stepping over, onto or down an obstacle, climbing up or down stairs. For any humanoid robot, these are considered challenging tasks. Humanoids usually perform motion instructions improperly [2, 3, 6]. That is because they possess rough odometry estimates; they might slip on the ground surface depending on the ground friction, as well as the frequency of its joint back-lashing. In fact, their light-weighted and small sensors are influenced by noise inherently. All these

circumstances can lead to uncertain estimates of pose and/or inaccurate execution of movement [7, 8]. Nonetheless, there are other explanations that justify why in practical applications humanoids are not used often. For example, humanoids are expensive as they are produced in small quantities and consist of complex hardware parts [3].

Many researchers use navigation algorithms that use wheel robots rather than legs, but the disadvantage of this model is that it does not respect all the collision avoidance capabilities of humanoids; therefore, more appropriate methods are needed to navigate in cluttered and multilevel circumstances [3, 9]–[11].

At first, Humanoid research has concentrated on some variables such as basic walking, but current systems are now becoming more knotted. Most humanoids already have full body control prototypes and advanced sensors such as sound, laser, stereo vision, and touch sensor systems. That provide the circumstances required and important to deal with complex issues for instance grasping and walking. Movement planning is a positive way of dealing with complex issues, because planning enables the versatility of meeting different criteria. The design of complex dynamic motions is accomplished only by *robot kinematics*, which transforms from the common space where the kinematic chains are described into the Cartesian space where the robot's manipulators travel and

*Corresponding Author: Walaa Gouda, Computer and Engineering Networks Dept., Jouf University, Saudi Arabia, wgaly@ju.edu.sa

vice versa [3, 12]–[14].

As it can be used to plan and execute movements, robot kinematics is important, it is split into *forward* and *inverse* kinematics. Forward kinematics corresponds to using the kinematic equations of the robot to calculate the position of the end effector from the specified parameter values [12, 15, 16]. On the other hand, reverse kinematics corresponds to using the kinematics equations of a robot to determine the joint parameters that provide the target position of the end effector. That is clear why kinematics is necessary in any aspect of complex movement design [12, 14, 16, 17]. The relationship between inverse and forward kinematics is shown in Figure 1.



Figure 1: Representation of inverse and forward kinematics

Whole body motions that fulfill a variety of constraints are required for Humanoids to perform complex motions activities, where the robot has to maintain its balance, avoid self-collisions and collisions with environmental obstacles. However, consideration must be given to the ability of humanoids to move over/ down/ onto objects and maneuver in multi-level environments. All these restrictions and the high degree of freedom of the humanoid robot make the whole body motion planning a challenge [2]. The main goal of *whole body balance motion* is to make consistent motions and adapt the robot's behavior to new situations [18].

Based on our Knowledge, human-like movements depend on the characteristics and the objectives of given tasks. A possible solution to the evolution of human-like motions or attitudes is to investigate and solve a given task with software of a human robot. In order to reduce these human efforts and perform more complex tasks, many researchers have studied another approach based on learning theories. Demonstration robot programming, also known as learning imitation, tedious manual programming seemed to automate the manipulation of robots [19, 20, 14].

For many supported tasks to be performed autonomously and safely, a robot must be aware of the task restrictions and must prepare and carry out motions that take into account these constraints while avoiding obstacles [19, 20]. Such motion planners, however, typically require manually programming the task constraints, which requires a domain knowledge programmer. On the other hand, methods focused on learning from demonstrations are highly effective in automatically learning task restrictions and controllers from people demonstrations without programming knowledge and experience [20].

Humanoid motion design has been examined thoroughly in recent years. For example, the approach provided by [7] allows a NAO humanoid to climb spiral staircases efficiently, fitted with a 2D laser sensor and a monocular camera..

Although [9] proposed a method for locating NAO humanoids using only on-board sensing in undefined complicated indoor

environments. Approach (Nishiwaki et al.[21] allows NAO to ascend single steps after manually aligning the robot in front of them without any sensory information to detect the stairs. While in [22], the authors provides a three-step action plan for ascending staircases for HRP-2. Furthermore in [23], the authors proposed a technique for climbing stairs using efficient image processing techniques with single camera fixed at the top of the robot at a height of 60 cm.

In addition, Burgetet al. [4] introduce an approach to the whole body movement design by manipulating interactive items such as doors and drawers. Their experiments with a NAO unlocking a drawer, a lock, and carrying up an object showed their power to solve complicated planning problems. Gouda et al. [24] proposes another approach to entire body motion planning for humanoids using only on-board sensing. NAO humanoid has been used to verify the sequence of actions posed in order to avoid obstacles, step over acts and step up or down.

Thus in [3], the researchers developed a motion named T-step, which helps the robot to move over actions as well as to parameterize steps on or down actions. The paper presented by Gouda et al.[1] suggested a NAO humanoids method for climbing up stairs as well as jumping up or down obstacles placed on the ground.

Measurement of distance for robots is very important as the robot needs to go to the exact position to perform additional tasks, such as playing with a ball, observing the house etc. The authors in [25] used four different measurement methods to compare the results. Such methods include the live Asus Xtion Pro, sonar, NAO-owned cameras for achieving stereo vision, and NDI for obtaining the 3D coordinate and ground truth as well. They conclude by selecting an infrared tool such as Asus Xtion Pro Live for short range calculation of distance. They also deduced that both sonar and infrared can be easily used to obtain the depth information without much intervention, but more studies needs to be done with stereo vision.

This paper is a major overhaul and extension of earlier [24, 1] conference versions. We have developed a NAO humanoid movement system that generates solutions that meet all the necessary constraints and apply reverse kinematics to joint chains.

The designed framework allows the robot to intelligently perform whole body balancing action sequences, involving walking over, up or down obstacles, as well as climbing up or down straight stairs in a 3D environment, shown in Figure 2. Based solely on the on-board sensors and joint encoders of the robot, we built an effective whole body movement strategy that performs safe motions to maneuver robustly in challenging scenes with obstacles on the ground, as shown in Figure 2. Our approach uses a single camera to determine the appropriate movement consisting of a series of actions based on the observed obstacle. As shown in the practical experiments with a NAO humanoid H25 as well as a series of simulation experiments using Webots (a simulation applications for programming, modeling and simulating robots [26, 27]), our system leads to vigorous whole body

movements in cluttered multilevel environments comprising barriers of various sizes and shapes.

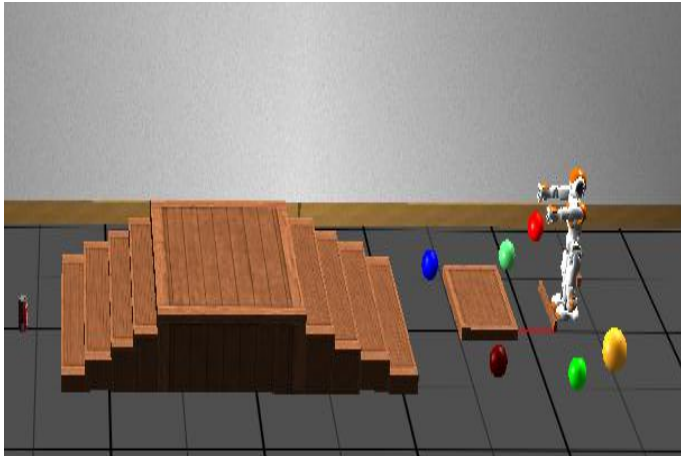


Figure 2: The simulated environment similar to the real world environment

2 Methodology

The obstacle avoidance device problems presented in this paper are generally applied to the humanoid robot of NAO. NAO robot (shown in Figure 3) is the world’s premier research and learning humanoid. It is the perfect platform for all forms of science, technology, engineering and mathematics. [28, 29].

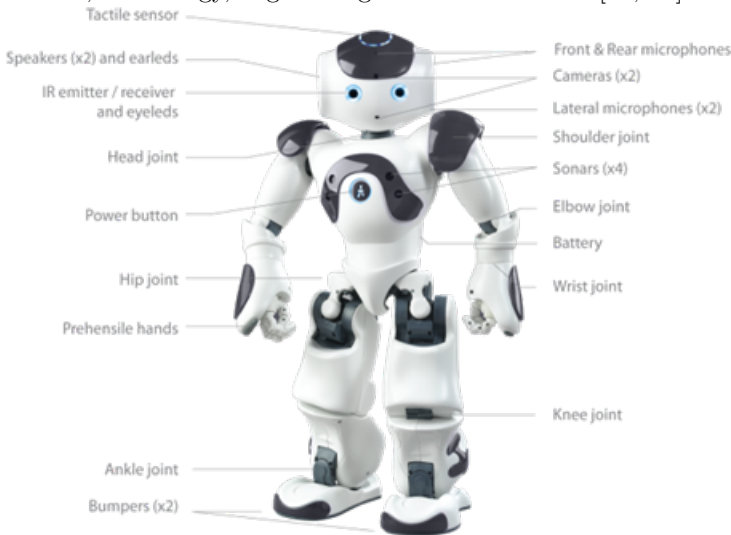


Figure 3: Aldebaran NAO H25 [18]

NAO humanoid robot is an integrated, programmable, medium-sized humanoid experimental platform with five kinematic chains (head, two arms, two legs) developed by Aldebaran Robotics in Paris, France. The NAO project began in 2004. NAO officially replaced Sony’s quadruped AIBO robot in the RoboCup in August 2007. NAO has produced many prototypes and several models over the past few years [18, 14, 30].

Table 1 summarized NAO Humanoid Robot’s technical

specification information, where NAO H25 is about 58 cm in height; 5.2 kg weighs and includes a programming framework called NaoQi that offers a user-friendly interface for controlling sensors on the NAO and sending control commands to achieve higher robot behaviour. The NAO robot used has 25 degrees of freedom (DOF), each joint is fitted with location sensors, it has 11 DOF for its lower parts and 14 DOF for its upper parts [18, 31, 32].

Table 1: Technical specification of NAO Humanoid Robot

NAO’s Descriptions	NAO’s Specifications
NAO’s Height	58 cm
NAO’s Weight	5.2 kg
NAO’s Autonomy	90 min const. walk
NAO’s Degree of freedom	25 DOF
NAO’s CPU	X86 AMD Geode 500 MHz
NAO’s OS (Built-in)	Linux
NAO’s Compatible OS	Mac OS, Windows, Linux
NAO’s Programming languages	C, C++, .Net, Urbi, Python
NAO’s Vision	Two CMOS 640x480 cameras
NAO’s Connectivity	WiFi, ethernet

NAO incorporates a variety of sensors that can be used to identify the face and form with two similar video cameras on the forehead. Nonetheless, the two cameras do not overlap, as shown in Figure 4, And each time only one is active and the view can be switched almost instantly from one to the other.

Both cameras provide a 640x480 pixel video with a speed of 30 frames per second, which can be used to distinguish visual items such as goals and balls [18]. Four sonars on the chest (2 receivers and 2 receivers) allow NAO to detect obstacles in front of it. Furthermore, NAO has a wealthy torso inertial system with a 2-axis gyroscope and a 3-axis accelerometer that can provide real-time information about its instantaneous body movements. Two bumpers at each foot’s tip are basic off/on switches and can provide obstacle information on foot collisions. There is a set of strength-sensitive resistors for each foot that support the forces applied to the feet, while encoders on all stepper motors record the real measurements of all joints for each moment. [14, 30].

Eventually, a GEODE 500 MHz board with 512 flash memory and optional extension via a USB bus is the CPU located in the robot’s head. If necessary, a Wifi connection connects NAO to any local network or other NAOs. NAO is powered by Lithium Polymer batteries with time ranging from 45 minutes to 4 hours depending on their operation [18, 31].

In general, the robot should be completely symmetrical, but it is noteworthy, according to the manufacturer, that some joints on the left side differ from the corresponding joints on the right side [14, 30]. In addition, certain joints appear to be able to shift within a wide range, the robot’s hardware controller forbids access to the limits of these

ranges due to possible NAO shell collisions [14]. Kinematics is very useful for developers of NAO software because it can be used to plan and execute these complex movements [14, 6, 30].

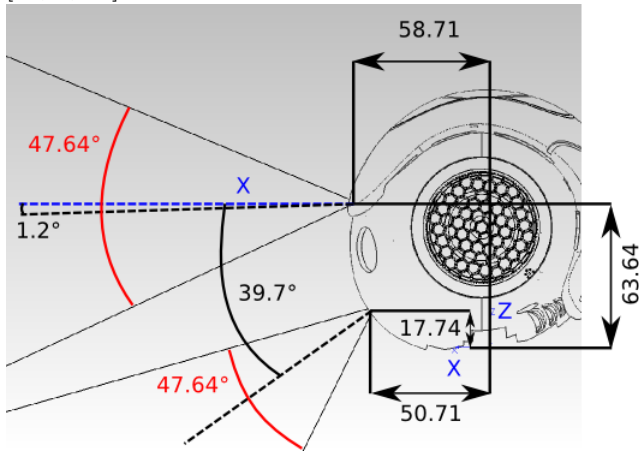


Figure 4: Cameras Location [18].

NAO's geometric model gives the location of the effector ($X = [P_x, P_y, P_z, P_{wx}, P_{wy}, P_{wz}]$) relative to an absolute space according to all the common positions ($q = [q_1, \dots, q_n]$).

$$X = f(q) \tag{1}$$

The direct kinematic model is presented in equation (2), which is the time derivative of equation (1).

$$\dot{X} = \frac{\delta}{\delta t} f(q) \dot{q} = J(q) \dot{q} \tag{2}$$

Where $J(q)$ is the Jacobian matrix. It is necessary to check the end effector and deduct the joint position, so that the inverse kinematic model shown in equation (3) is needed.

$$\dot{q} = J^{-1} \dot{X} \tag{3}$$

In many cases, J is not explicitly invertible directly (matrix not square), this problem is solved mathematically using Moore-Penrose pseudoinverse [18].

Aldebaran Robotics includes shared values in the documentation of the robot [18]. For each link / joint, the center of mass is represented by a point in the three-dimensional space that assumes the joint's zero pose.

The obstacle avoidance model problems suggested in this work are generally applied to the humanoid robot of NAO. Since NAO H25 has 25 DOF, it can perform many complex motions including walking, kicking a ball, going up and down obstacles, etc. [14, 6]. NAO's walk uses a simple dynamic model (Linear Inverse Pendulum) which is stabilized by feedback from its joint sensors, which makes the walk stable against minor disturbances and absorbs frontal and lateral torso oscillations [18].

The swing foot can be positioned on the front and 16 cm on the side at most 8 cm and the peak elevation is 4 cm using the walking controller given. The robot's feet are about 16

cm x 9 cm in size. From these statistics, it is clear that NAO can not perform complex motions such as avoiding obstacles (excluding turning itself around) using the standard motion controller as shown in Figure 5 [18, 3, 14].

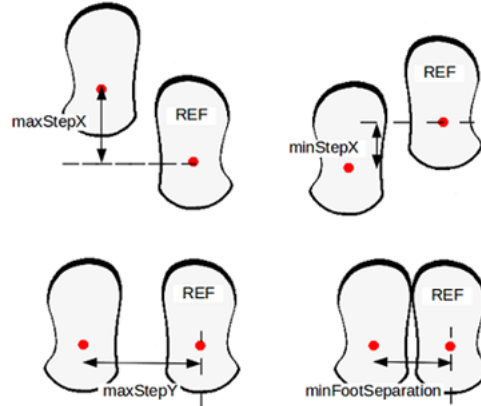


Figure 5: Clip with maximum outreach [18].

Only the on-board sensor of the robot, the lower camera, is used in this paper. Another limitation for NAO is its camera, as the two cameras of NAO do not overlap and only one of them is active each time and the view can be switched almost instantly from one to the other [18]. That means that it is not possible to form a 3D image, i.e. details about the distance between the obstacle and the robot.

In order to avoid obstacles, NAO must identify objects in the environment if they encounter some, so that it must first know specific objects by using the Choreographic Vision Monitor [31] (Choreography is a graphical device created for NAO humanoids by Aldebaran Robotics).

Algorithm 1: Recognize objects

```

while There exist some objects to learn do
    capture image using NAO's cameras ;
    draw the contour of the learned object manually ;
    give a name and other details to the object ;
end
storing learned images in NAO's database ;
    
```

Object learning is shown in Figure 6 and described in Algorithm 1. The image should be obtained using NAO's cameras as shown in Figure 6a, then the contour of the learning object is drawn, segment by segment, manually as shown in Figure 6b. After that, in order to differentiate between different objects, details about the object like the name is entered as shown in Figure 6c.

Then a message appears to show the learning process's success (see Figure 6d). Photos are stored in NAO's database after learning all the necessary items. Once all photos are placed in NAO's database, during its active deployment, NAO will be able to perform object recognition. The method of recognition is based on visual key points detection and not on the object's external form, so NAO can only identify

objects that have been previously studied.

The recognition method is partly resilient to distance, varying from half to twice the distance used for learning, and angles up to an inclination of 50° for something observed facing the camera, rotation and light conditions [18]. Each key point observed in the current image is matched only with one of the key points learned in the database. So, if the score is too small to choose between two items, none of them will be correlated with the key point. This means that the object can not be learned twice, because learning the same area of an object twice reduces its detection rate as the score is the same for both objects and the object is not recognized [18].

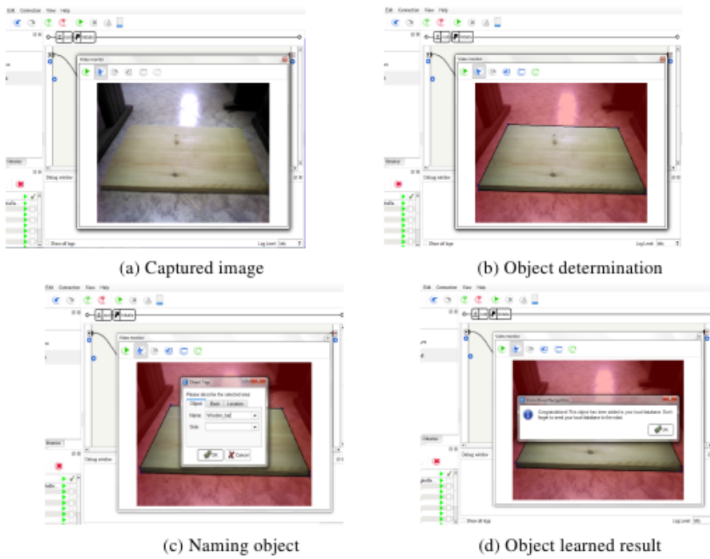


Figure 6: Object learning phase

2.1 Motion design

As shown in Figure 5, it is obvious that NAO can not use its typical motion controller to perform complex motions (other than walking) such as moving over/onto/ down obstacles. In order to allow the robot to overcome these limitations, a *kinesthetic teaching* is applied. Here, the humanoid NAO H25 is programmed using the choreograph [31] and the python programming language. We implement a special movement design (described in Algorithm 2), inspired by [3] and extends the work in [24, 1].

This design allows the robot to walk over iteratively as well as to climb up or down obstacles (including straight staircases) depending on the shape of the obstacle. The right leg's foot (L_R) is positioned in the intended motion at an angle of 30°, which is the basis for the other acts. The robot then shifts his balance to that leg, (L_R), (the leg with the angle 30°) and moves freely the other leg, left leg (L_L). Then the balance is shifted to L_L , then L_R moves freely next to L_L and then the balancing is performed on both legs as shown in Figure 7. The development of such complex dynamic movements can only be accomplished by using the robot's own kinematics [2, 12].

Algorithm 2: Actuate motion

```

go to initial position;
place right foot at angle 30°;
move balance to right foot;
let left foot move freely;
if obstacle recognized is a long bar then
    move left foot forward;
    set the foot height to zero, i.e., put it on the ground;
else if obstacle recognized is a wide bar then
    move left foot forward;
    if robot on the ground then
        set the foot height to 2cm, i.e., onto the bar;
    else [robot on the bar]
        set the foot height to -2cm, i.e., down the bar;
    end
else the recognized object is stairs
    move left foot forward;
    if robot facing stairs up then
        set the foot height to 4cm, i.e., climbing up stairs;
    else [robot on the bar]
        set the foot height to -4cm, i.e., climbing down stairs;
    end
end
move balance to left foot;
let right foot move freely;
move right foot forward beside the left foot;
    
```

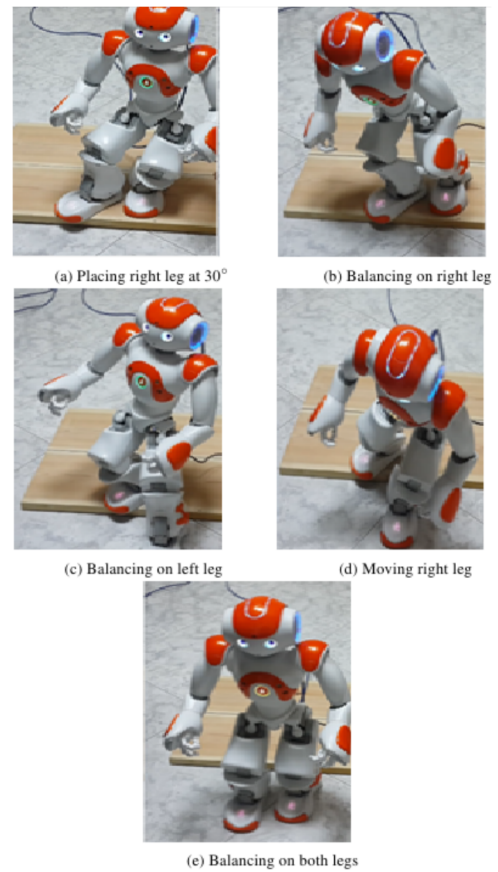


Figure 7: Designed motion step

The motive for using this series of motions is to manipulate, when stepping forward, the greater lateral foot displacement. From such a position, the robot can perform step over

movements to overcome obstacles with an elevation of up to 5 cm which can not be done with the standard motion controller.

The sequence of motions for ascending and descending stairs or wooden bar is similar to the sequence of motion for stepping over actions except for the positioning the swing foot, as it is positioned closer to the placed foot but at a different height where the height is changed using reverse kinematics depending on the recognized item.

The movements designed for full body stability use NAO's original kinematics to directly control the effectors in Cartesian space using an inverse kinematics approach to solving. The model *Generalized Inverse Kinematics* is used; it addresses cartesian and joint power, stability, redundancy, and role priorities.

This formulation takes into account all the joints of the robot in a single problem. The obtained motion guarantees several specified conditions, such as balancing, holding down one foot, etc. The developed motion system's capabilities are then demonstrated in a series of simulation experiments using Webots (a simulation platform for modeling, programming and simulating robots [26]) for the NAO robot, as well as real-world experiments with the real NAO H25 robot.

The main differences between the motion built in this work and the motions defined by *Maier et al.* [3] are in the angle of placement of the robot's foot. Because our model movement helps the robot to position its foot at a 30° angle, This allows the robot to achieve the equilibrium in a shorter time and more securely than the motion given by *Maier et al.* [3], while *Maier et al.* [3] equals 90° to the robot's positioning angle.

Our work further expands the work outlined in [1] as it executes the entire scenario starting with walking in search of a target, including avoiding obstacles by ascending straight stairs, climbing up stairs as well as walking over, on or down obstacles until the goal is reached.

3 Experimental Evaluation

The developed framework allows the robot to robustly execute entire body balancing action sequences, including stepping over and ascending or descending obstacles, as well as ascending or descending straight stairs in a 3D environment, shown in Figure 2. Based solely on the onboard sensors and joint encoders of the robot, we built an effective whole body movement strategy that performs safe movements to maneuver robustly in demanding scenes with obstacles on the ground as shown in Figure 2.

Our method uses monocular camera to determine the appropriate motion consisting of a series of actions based on the detected obstacle. As shown in practical experi-

ments with a NAO humanoid H25 as well as a series of simulation experiments using Webots, the framework results in efficient motions for the whole body in congested multilevel environments with artifacts of various shapes and sizes.

The scenario consists of a world in which the humanoid has to navigate through arbitrarily cluttered obstacles (see Figure 2). The robot is going to navigate the environment until it reaches the goal of recognizing the cane.

The target is far from the robot's reachable space at the start of the experiment. The robot starts to walk in quest of its target to avoid obstacles. Once the obstacle is in front of the robot, the entire body movement must be carried out in order to prevent the robot from colliding with it.

It's worth noting that if the robot gets lost and encounters items that it hasn't learned instead of locating the previously learned objects or target, it's going to notify a message that notifies it is lost and stop walking.

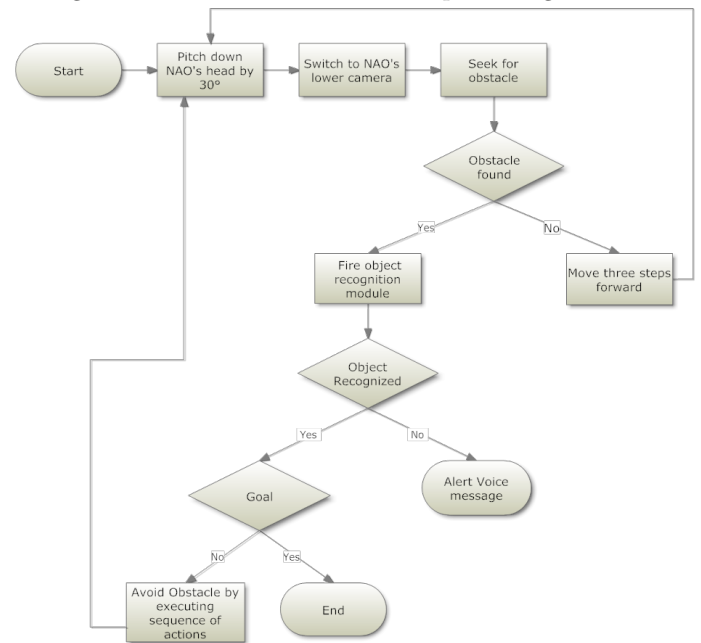


Figure 8: The main process of our experimentation on NAO H25

Figure 8 illustrates the main process that describes our approach. The robot must move three steps forward in the experiments presented, then stop moving and pitch his head down at an angle of 30° and switch to lower camera in his head to look for obstacles on the ground in front of his feet. The object recognition module will be activated until an obstacle is detected; then the robot must take three more steps forward. In the case of an obstacle being recognized, the robot compares the object being recognized with the objects being observed and stored in its database.

If it is the robot's goal, it must stop moving and alert a voice message that notifies it reaches its goal; otherwise it will execute stable whole body motions to deal with the

recognized object and then move three steps forward and so on until the goal is achieved. If the robot does not know the object, it triggers a voice message notifying that it is approaching an object that it has not previously known or seen and then stops moving.

Figure 9 shows the operation of choreographic [31] identification of items [31], the top right of the photo shows a video monitor showing what the robot actually sees. It is clear that through the robot's vision recognition module, the robot can recognize the long wooden bar previously taught and saved in NAO's database. The vision recognition module returns the name of the recognized object to the next stage when a match occurs to take an action based on the object. Figure 10 is the action module expansion shown in Figure 9, where the robot has to perform step by step sequence of motions to overcome this barrier when the wooden bar is recognized.

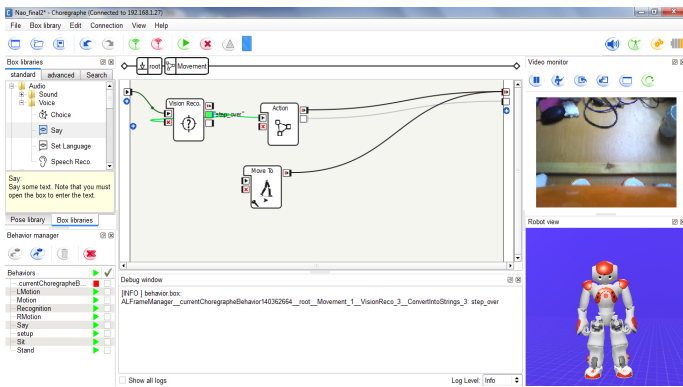


Figure 9: Wooden bar is recognized

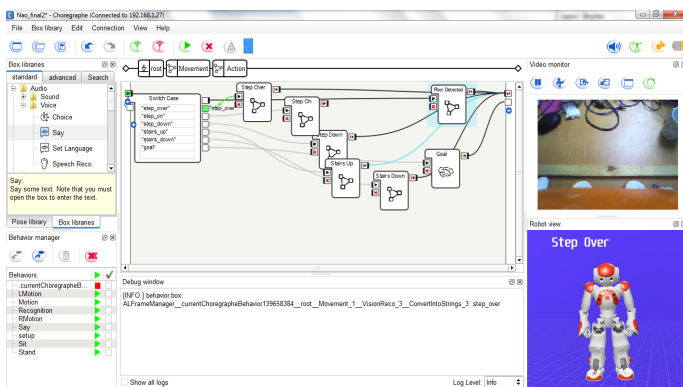


Figure 10: Step over actions are executed

The key challenge is to accurately determine the distance from NAO to the obstacle. As mentioned earlier, the NAO camera does not provide the distance between the robot and the obstacle. The robot's vision recognition module is also partly robust to distance the robot from the target, as it ranges from half to twice the distance used to learn [18]. It is also based on the identification of visual key points, not on the object's external shape, which means that if the robot recognizes the wooden bar, it will perform step-over activities regardless of the distance between them.

If the distance between his feet and the target is not sufficient, the robot may struggle to overcome the obstacle. The distance between the robot and the obstacle is not known because the camera of the robot has a limitation in providing depth information. If the obstacle is placed below the correct margin of the robot, the robot will strike the obstacle while moving its leg, resulting in a shift in the angle of the foot, disrupting its balance and the robot will fall. A further situation if the obstacle is placed at a distance greater than a suitable distance, the robot can put his swing foot on the bar, which will also cause a disturbance in his balance and fall.

A simple solution to this problem is to allow the robot to learn the image of the wooden bar on the floor at a distance near his head as shown in Figure 11a, and learn the image of the floor in the learning phase near the robot's feet as shown Figure 11b.

If the wooden bar is located at a large distance in front of the robot as shown in Figure 11d and Figure 11c, both the floor image and the long wooden bar image will be recognized by the vision recognition module. But as the vision recognition module gives high visual key points to the floor over the long wooden bar, it will return the floor as the recognized object. In this situation, the robot pushes three steps forward, until the long wooden bar's visual key points are higher than the floor's, which means the robot is now close to the wooden bar and must be avoided. In deciding the distance between NAO and the barrier, this solution proves to be very efficient.

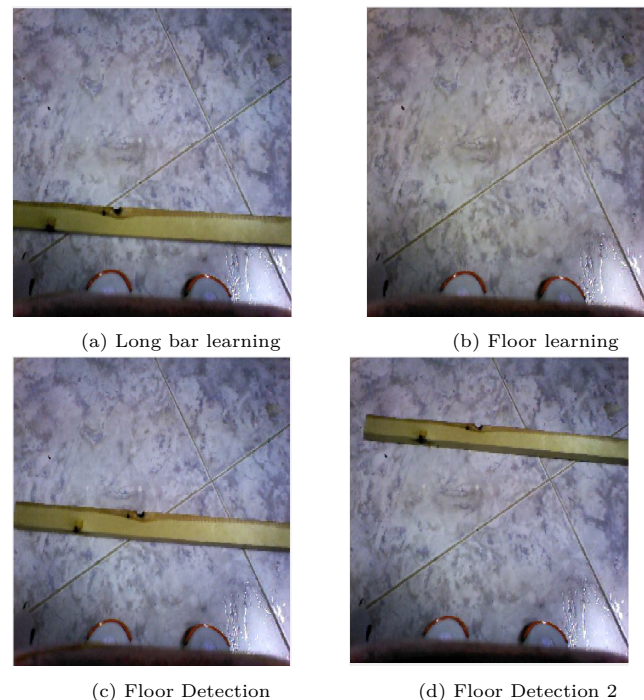


Figure 11: Learning floor image
 The experiments are conducted on: (a) the robot stepping over a 40 cm wide wooden bar, 3.5 cm high and 2 cm deep; see Figure 12, (b) Step on a wooden bar 40 cm wide, 2 cm

high and 40 cm deep; see Figure 14, and (c) move down to the ground from that bar as shown in Figure 15. Figure 16's first row shows the process of climbing up a single step of a 40 cm wide, 4 cm high and 20 cm deep straight staircase. The second row shows the remaining four steps of the staircase climbing up. Similarly, Figure 17's first row shows the process of climbing down a single step of a 40 cm wide, 4 cm high and 20 cm deep straight staircase. The second row demonstrates the descent down the staircase's remaining four steps.

These figures show still frames of a video sequence where our robot successfully walks up or down a wooden bar and a straight staircase climbs up or down.



Figure 15: NAO descending from a 2 cm height and 40 cm depth wooden obstacle using the expected whole body movement



Figure 12: NAO steps over a 3.5 cm height and 2 cm depth wooden obstacle using planned whole body movement



Figure 16: NAO ascending 4 cm high and 20 cm deep stairs using the expected entire body movement

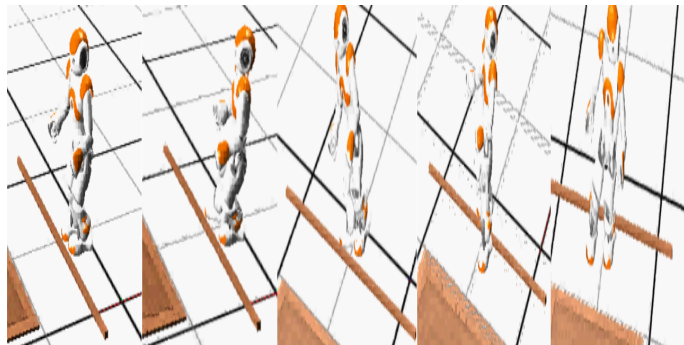


Figure 13: Simulated NAO walking over a 2 cm depth and 3.5 cm height wooden obstacle using expected whole body movement



Figure 17: NAO descending 4 cm high and 20 cm deep stairs using the expected whole body movement



Figure 14: NAO step 2 cm high and 40 cm deep on a wooden obstacle using the expected whole body movement

The algorithm applied for all motions is the same. The only exception is the height of the leg of NAO and the position of the swing foot, as in the case of walking up or down stairs and walking on or down obstacles, the swing foot is positioned closer to the standing foot. The justification for reducing the angle of positioning of the foot is the time and the opportunity to comfortably carry out more actions.

Our work performs a whole scenario starting with walking in search of a goal, including avoiding obstacles by climbing down straight stairs, climbing up stairs, as well as climbing over, on / down obstacles, until the goal is reached.

Table 2: Comparison between the proposed approach and other approaches

Motion Design	Foot Placement	Average Time	Obstacle Avoided	Vision
<i>Maier et al.</i> [3]	90°	2 min.	small bar, large bar	Depth Camera
<i>Gouda et al.</i> [24]	60°	43 sec.	Long bar, Wide bar	Monocular Camera
Designed Motion	30°	29 sec.	Long bar, Wide bar, straight staircase	Monocular Camera

The discrepancy between the suggested approach and the [3] approach is shown in the table 2. Declining foot displacement angle reduces the time as well as the robot's ability to perform more safely actions. The angle of the displacement of the foot can no longer be reduced as after 30°, since the robot could not perform actions.

The robot steps over it when a long bar is recognized and transfers his leg to the ground after the obstacle. Whereas the robot steps on/down the bar or stairs in the case of a broad or staircase and lifts his leg on or down it. The period for all motions to be performed is quite close. It takes the robot 30 seconds to execute step over movement, 29 seconds to execute step on / down movement, and 28 seconds to ascend one step up or down.

We perform a systematic evaluation of our approach to correctly walking over/onto / down an obstacle and climbing up stairs. Only on-board sensors are used to determine the success rate of these actions. In ten real world runs on our straight staircase consisting of four steps, approximately 96% of the straight staircase was successfully climbed up / down by the robot. Only one of the climbing steps contributed to the robot's crash.

The robot also successfully moved a total of eight subsequent times over/onto/down the wooden bar. The joints are then heated for an extended period of time by applying a

stress on them. Joint overheating changes the parameters of the joints, especially rigidity, which affects the robot's balance; thus movements can no longer be performed effectively and the robot does not overcome the obstacle and sometimes falls. Through reducing the time spent in critical positions or setting stiffness to 0 after each operation, it is possible to reduce the heating in the joints.

Another problem is the execution time of the motion, as the robot must have enough time to reach balance after each action is performed in the motion or it will fall. The robot will not be able to finish the action it is doing in the event of the time being too short, so equilibrium will not be achieved and the robot will crash. However, if the execution time is too long to allow the robot to finish the preforming operation, its joints may easily get hot and may not be able to keep their balance in each position for a long time, so it may also fall.

These results show that our method makes it possible for the humanoid NAO H25 to climb up or down reliably four subsequent steps of straight staircases that are not visually detectable easily. In addition, by stepping over/onto/down them efficiently, the robot can avoid colliding with ground obstacles.

4 Conclusion

Throughout this research, we presented an innovative approach that allows a humanoid robot, particularly NAO H25, to design and implement entire body stability set of actions such as stepping over and overcoming obstacles, as well as ascending or descending a staircase using only on-board sensing. NAO is capable of walking on multiple floor surfaces such as carpets, tiles and wooden floors, it is capable of walking between these surfaces.

Large obstacles, however, can still cause him to fall because it assumes that the ground is more or less flat. The construction of complex dynamic movements can only be accomplished by using the own kinematics of the robot. Based on the recognized object obtained from the robot's database using the robot's own monocular camera (using NAO's vision recognition module), a sequence of actions to avoid the object is being executed.

One of NAO's most important challenges is to accurately identify the proper sequence of motions that help it to safely avoid obstacles without colliding with them. NAO is not able to use its standard motion controller to perform complex motions such as obstacle avoidance, two motion designs are investigated and presented to overcome this limitation.

Another difficulty is to assess the distance between the robot and the obstacle. To robots, distance measurement is

very important as the robot has to go to the exact position to perform additional tasks, such as playing with a ball, controlling the room, etc.

The robot's vision recognition module is partly reliable as it varies from half to up to twice the distance used for learning to distance between the robot and the target. The camera of the robot also has the limitation that it can not provide the distance between the robot and the obstacle; in order to overcome this limitation, a simple solution is implemented to allow the robot to learn images for the object on the floor in a place close to the robot's feet in the learning phase next to images for the floor itself. In deciding the distance between the robot and the obstacle, this solution proves to be very efficient.

This work is a major extension of the work discussed in [24, 1]. Shown in the simulation as well as actual world experiments with NAO H25 humanoid, in congested, multilevel environments with obstacles of different shapes and sizes, the robot can effectively execute whole-body movements.

References

- [1] W. Gouda and W. Gomaa, "Complex motion planning for nao humanoid robot," in *11th International Conference on Informatics in Control, Automation and Robotics (ICINCO)*, 2014.
- [2] C. Graf, A. Härtl, T. Röfer, and T. Laue, "A robust closed-loop gait for the standard platform league humanoid," in *Proceedings of the Fourth Workshop on Humanoid Soccer Robots in conjunction with the*, pp. 30–37, 2009.
- [3] D. Maier, C. Lutz, and M. Bennewitz, "Integrated perception, mapping, and footstep planning for humanoid navigation among 3d obstacles," in *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, pp. 2658–2664, IEEE, 2013.
- [4] F. Burget, A. Hornung, and M. Bennewitz, "Whole-body motion planning for manipulation of articulated objects," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pp. 1656–1662, IEEE, 2013.
- [5] M. Mattamala, G. Olave, C. González, N. Hasbún, and J. Ruizdel Solar, "The nao backpack: An open-hardware add-on for fast software development with the nao robot," in *Robot World Cup*, pp. 302–311, Springer, 2017.
- [6] S. Shamsuddin, L. I. Ismail, H. Yussof, N. Ismarrubie Zahari, S. Bahari, H. Hashim, and A. Jaffar, "Humanoid robot nao: Review of control and motion exploration," in *Control System, Computing and Engineering (ICCSCE), 2011 IEEE International Conference on*, pp. 511–516, IEEE, 2011.
- [7] S. Oßwald, A. Gorog, A. Hornung, and M. Bennewitz, "Autonomous climbing of spiral staircases with humanoids," in *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, pp. 4844–4849, IEEE, 2011.
- [8] A. Choudhury, H. Li, D. Greene, S. Perumalla, et al., "Humanoid robot-application and influence," *Avishek Choudhury, Huiyan Li, Christopher M Greene, Sunanda Perumalla. Humanoid Robot-Application and Influence. Archives of Clinical and Biomedical Research*, vol. 2, no. 2018, pp. 198–227, 2018.
- [9] A. Hornung, K. M. Wurm, and M. Bennewitz, "Humanoid robot localization in complex indoor environments," in *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pp. 1690–1695, IEEE, 2010.
- [10] W. Gouda, W. Gomaa, and T. Ogawa, "Vision based slam for humanoid robots: A survey," in *Electronics, Communications and Computers (JEC-ECC), 2013 Japan-Egypt International Conference on*, pp. 170–175, IEEE, 2013.
- [11] G. Majgaard, "Humanoid robots in the classroom.," *IADIS International Journal on WWW/Internet*, vol. 13, no. 1, 2015.
- [12] S. Kucuk and Z. Bingul, "Robot kinematics: forward and inverse kinematics," *Industrial Robotics: Theory, Modeling and Control*, pp. 117–148, 2006.
- [13] M. Gienger, M. Toussaint, and C. Goerick, "Whole-body motion planning—building blocks for intelligent systems," in *Motion Planning for Humanoid Robots*, pp. 67–98, Springer, 2010.
- [14] N. Kofinas, *Forward and inverse kinematics for the NAO humanoid robot*. PhD thesis, Diploma thesis, Technical University of Crete, Greece, 2012.
- [15] M. Szumowski, M. S. Żurawska, and T. Zielińska, "Preview control applied for humanoid robot motion generation," *Archives of Control Sciences*, vol. 29, 2019.
- [16] A. Gholami, M. Moradi, and M. Majidi, "A simulation platform design and kinematics analysis of mrl-hsl humanoid robot," 2019.
- [17] M. Assad-Uz-Zaman, M. R. Islam, and M. H. Rahman, "Upper-extremity rehabilitation with nao robot," in *Proceedings of the 5th International Conference of Control, Dynamic Systems, and Robotics (CDSR'18)*, pp. 117–1, 2018.
- [18] AldebaranRobotics, "Nao software 1.14.5 documentation @ONLINE," 2014.
- [19] C. Bowen, G. Ye, and R. Alterovitz, "Asymptotically optimal motion planning for learned tasks using time-dependent cost maps," 2015.
- [20] S. Shin and C. Kim, "Human-like motion generation and control for humanoid's dual arm object manipulation," 2014.
- [21] K. Nishiwaki, S. Kagami, Y. Kuniyoshi, M. Inaba, and H. Inoue, "Toe joints that enhance bipedal and fullbody motion of humanoid robots," in *Robotics and Automation, 2002. Proceedings. ICRA'02. IEEE International Conference on*, vol. 3, pp. 3105–3110, IEEE, 2002.
- [22] J. E. Chestnutt, K. Nishiwaki, J. Kuffner, and S. Kagami, "An adaptive action model for legged navigation planning.," in *Humanoids*, pp. 196–202, 2007.
- [23] W. Samakming and J. Srinonchat, "Development image processing technique for climbing stair of small humanoid robot," in *Computer Science and Information Technology, 2008. ICCSIT'08. International Conference on*, pp. 616–619, IEEE, 2008.
- [24] W. Gouda and W. Gomaa, "Nao humanoid robot motion planning based on its own kinematics," in *Methods and Models in Automation and Robotics (MMAR), 2014 19th International Conference On*, pp. 288–293, IEEE, 2014.
- [25] J. Guan and M.-H. Meng, "Study on distance measurement for nao humanoid robot," in *Robotics and Biomimetics (ROBIO), 2012 IEEE International Conference on*, pp. 283–286, IEEE, 2012.
- [26] Cyberbotics, "Webots: the mobile robotics simulation software @ONLINE," 2014.
- [27] L. D. Evjemo, "Telemanipulation of nao robot using sixense stem-joint control by external analytic ik-solver," Master's thesis, NTNU, 2016.
- [28] RobotsLAB, "Nao evolution - v5," 2015.
- [29] T. Deyle, "Aldebaran robotics announces nao educational partnership program," 2010.
- [30] N. Kofinas, E. Orfanoudakis, and M. G. Lagoudakis, "Complete analytical inverse kinematics for nao," in *Autonomous Robot Systems (Robotica), 2013 13th International Conference on*, pp. 1–6, IEEE, 2013.

- [31] E. Pot, J. Monceaux, R. Gelin, and B. Maisonnier, "Choregraphe: a graphical tool for humanoid robot programming," in *Robot and Human Interactive Communication, 2009. RO-MAN 2009. The 18th IEEE International Symposium on*, pp. 46–51, IEEE, 2009.
- [32] L. Ismail, S. Shamsuddin, H. Yussof, H. Hashim, S. Bahari, A. Jaafar, and I. Zahari, "Face detection technique of humanoid robot nao for application in robotic assistive therapy," in *Control System, Computing and Engineering (ICCSCCE), 2011 IEEE International Conference on*, pp. 517–521, IEEE, 2011.