[0041] The robot instantaneous velocity and acceleration obtained 310 from the ALRobotPose module 230 are continuously processed to compute 320 a robot stopping distance, and a robot frontal security distance is calculated 330 therefrom.

[0042] From this output and a calculated robot position, a robot safety region is determined 340.

[0043] The invention defines regions around the robot that must be clear of obstacles for the robot to move. The size and shape of these regions depend on the robot instantaneous velocity, in translation and rotation. The shape of the safety region is determined so that:

- No point of the robot should come closer to an obstacle than a certain distance called *minimum distance*, 10 cm in a purely exemplary implementation;
- No point should be inside the region swept by the stopping robot, should the robot stop immediately;
- The safety region size in the motion direction is increased by a certain distance, called *frontal distance*, which depends continuously on the robot stopping distance; in an implementation of the invention, taken purely by way of example, the frontal distance is taken to be equal tot the minimum distance at low speed (i.e. 10cm), and 40 cm at full speed, but depending on the actual speeds of the robot, different values may be selected and calculated using the formula below which returns the results in the table.

[0044]The extra distance included in the maximum frontal distance is there to account for sensor uncertainty, sensor delay, and the comfort of people around the robot.

Frontal security distance F is computed from the following formula :

$$F = MD + \frac{SD}{MSD} \times (MFD - MD)$$

Wherein:

MD = Minimum Distance

SD = Stopping Distance

MSD = Maximum Stopping Distance

MFD = Maximum Frontal Distance

[0045] Examples of the frontal security distance (in meters) as a function of the speed of the robot (in meters per second) are given by the table below:

Speed (m/s)	Frontal security distance (m)
0,1	0,12
0,2	0,17
0,3	0,25
0,35	0,30
0,4	0,35
0,5	0,40
0,55	0,40

[0046] As can be seen, the minimum frontal security distance for a robot moving at its minimum speed of 0,1 m/s is just above the minimum distance of 10 cm which is the absolute limit for a robot which is not moving. The frontal security distance then increases up to 30 cm for the speed of 0,35 m/s which is a most common default speed for a humanoid robot of the type disclosed by this invention.

[0047] The frontal security distance does not increase above the maximum speed of the robot (0,55 m/s) for which a frontal security distance of 40 cm is necessary.

[0048] Figures 4a, 4b, 4c, 4d, 4e and 4f illustrate in a more detailed manner the computation of the safety areas around the robot in a number of embodiments of the invention.

[0049] On these figures, an autonomous robot 410a, 410c, 410e is moving at oriented speed 420a, 420c or 420e. It can be seen that speed 420c is greater than speed 420a. Also, the movement of the robot 410e is oriented to the left of the robot, at approximately the same frontal speed 420e as the frontal speed 420c or the robot 410c.

[0050] Safety regions 410b, 410d and 410f are defined around the robot and these safety regions have different sizes which depend on the speed of the robot. In figures 4b, 4d, 4f, the orientation of the triangles 420b, 420d, 420f depend on the orientation of the movement of the robot. Meanwhile the sizes of triangles 420b, 420d, 420f, depend on the instant speed of the robot, and their respective sizes increase with the speed of the robot. The safety region is defined around the robot and not around each obstacle as it is in the prior art. Also, the safety region rotates with the robot as illustrated by figure 4f.

CLAIMS

1. A method for controlling a trajectory (530a, 730) of at least one of upper and lower members of a humanoid robot (100), said method comprising:

- Storing an initial trajectory with a target point in a memory of the robot;
- Acquiring, from at least a sensing procedure controlled from on-board the robot, data representative of a position of one or more obstacles (540b, 910c);
- Calculating, by a processor on-board the robot:

an envelope (410b, 810) of said robot, said envelope depending on a speed and a direction of the robot;

a relative position of the envelope and the one or more obstacles;

a probability of collision of the envelope with an obstacle; and,

a series of commands in order to: i) avoid collision of the envelope with an obstacle; and ii) at least one of rejoin the target point of the initial trajectory stored in memory within an initial timing, and preserve the general direction and content of an initial gesture, to change in at least one of space and time at least one of the trajectory or~~and~~ a speed of the at least one of upper and lower members of the robot~~;~~

~~said method being characterized in that the series of commands is conditioned in at least one of space and time to: i) avoid collision of the envelope with an obstacle; and ii) at least one of rejoin the target point of the initial trajectory stored in memory within an initial timing, and preserve the general direction and content of an initial gesture~~.

2. The method of claim 1, wherein the series of commands to rejoin the target point comprises:

- calculating a point of the initial trajectory;
- calculating~~from where the robot will reach the target point in~~ a straight line between the point of the initial trajectory, and the target point;
- calculating a~~the~~ speed of the robot in the straight line to rejoin the target point at the initial timing;

- defining the trajectory of the robot as the initial trajectory until the point of the initial trajectory, then the straight line between the point of the initial trajectory, and the target point.

3. The method of claim 1, wherein the series of commands to preserve the general direction and content of the initial gesture comprises calculating a target position for a point of the robot, said point on a trajectory, computing a maximum joint velocity for a chain containing the point, said maximum joint velocity depending of the distance between the point and an obstacle.

4. The method of claim 1 to 3, wherein the initial trajectory is referenced in space and time.

5. The method of one of claims 1 to 4, wherein the sensing procedure is performed by at least one of a plurality of laser lines generators, a plurality of imaging sensors, a plurality of acoustic sensors and a plurality of contact detectors.

6. The method of claim 5, wherein the sensing procedure is performed by at least two of a plurality of laser lines generators, a plurality of imaging sensors, a plurality of acoustic sensors and a plurality of contact detectors.

7. The method of one of claims 1 to 6, wherein the data representative of a location of one or more obstacles is an extraction of a number of characteristic points from a map of pixels representing a probability of absence of an obstacle in said pixels.

8. The method of claim 6 or~~and~~ claim 7, depending on 6, wherein the map of pixels is produced at the output of at least a second sensing procedure and a data fusion process which increases an estimate of a confidence level of the probabilities of absence of an obstacle in said map of pixels in relation to a first sensing procedure.

9. The method of one of claims 7 and 8, wherein the probability of absence of an obstacle in a pixel of the map decreases over a preset time parameter down to 0,5 unless updated by an output of a sensing procedure with a higher probability.

10. The method of one of claims 1 to 9, wherein an envelope of a footprint of the robot is calculated as a function of the speed of the robot and predetermined guards around the robot.

11. The method of claim 10, wherein the series of commands is calculated to determine a changed trajectory to avoid collision of the envelope with any obstacle.

12. The method of claim 11, wherein the series of commands is further calculated to determine a changed trajectory and a changed speed to rejoin a target point of the initial trajectory at a time when the robot should have reached the target point on the initial trajectory.

13. The method of one of claims 1 to 9, wherein an upper member of the robot comprises a chain of segments articulated together by motored joints.

14. The method of claim 13, wherein an envelope of a member of the robot is calculated as a function of a predetermined security distance~~guard~~ around the articulated segments.

15. The method of one of claims 13 or 14, wherein the series of commands is calculated to determine a reduction in angular speeds of the motors of the joints of the articulated segments when the envelope approaches an obstacle.

16. The method of claim 15, wherein the reduction in angular speeds of the motors of the joints is calculated to saturate a ~~maximum~~ velocity of the chain to

a maximum safety value, said maximum safety value being calculated based on a target velocity of a point in the chain, and positions of obstacles relative to said point in the chain.

17. A humanoid robot comprising:

- At least an upper member and a lower member;
- A memory storing an initial trajectory with a target point and computer code instructions;
- A plurality of sensing modules configured to acquire data representative of a position of one or more obstacles;
- A processor configured to execute said computer code instructions to calculate:

an envelope of said robot, said envelope depending on a speed and a direction of the robot;

a relative position of the envelope and the one or more obstacles;

a probability of collision of the envelope with an obstacle; and,

a series of commands in order to: i) avoid collision of the envelope with an obstacle; and ii) at least one of rejoin the target point of the initial trajectory stored in memory within an initial timing, and preserve the general direction and content of an initial gesture, to change in at least one of space and time at least one of the trajectory or~~and~~ a speed of the at least one of upper and lower members of the robot~~;~~

~~said robot being characterized in that the series of commands is conditioned in at least one of space and time to: i) avoid collision of the envelope with an obstacle; and ii) at least one of rejoin the target point of the initial trajectory stored in memory within an initial timing, and preserve the general direction and content of an initial gesture~~.