Automated Staircase Detection, Alignment & Traversal

Mike Fair & David P. Miller
School of Aerospace & Mechanical Engineering
University of Oklahoma
mwfair@yahoo.com, dpmiller@ou.edu

Introduction:

Lola is a stair climbing robot developed at the OU Autonomous Robot Lab. Lola [1] is built around the chassis of a *Quest* Access wheelchair. The control electronics have been replaced and several sensors including a laser scanner [2] and a set of Sharp range sensors [3], have been added. The original wheelchair hardware was capable of negotiating a staircase, but it required the human operator to detect the staircase, put the chair into stair-climbing mode, align the chair with the staircase, and then manually control the rate of ascent or decent. The sequencing of the chair tilt and easy downs (the hydraulic actuators which gently allowed the chair to transition from level ground to inclines [4]) was done automatically.

Despite this significant automation, the alignment of the chair to the staircase was difficult for many operators. This was especially true when going up a stairwell since that was done, from the operator's perspective, backwards.

This paper describes a technique for detecting the top and bottom of a staircase to allow automated descents and ascents of staircases. Additionally, the detection algorithm allows the robot to easily align itself with the stairwell.

A blind person, assuming foreknowledge of a nearby staircase, advances slowly while feeling for the dropoff of the top step. Having located the top step, the person gingerly advances over the threshold, determining the steps orientation and feeling for the next step. The characteristics of the staircases are then clear enough to proceed at a normal pace, with the only remaining task being to prepare for the end of the staircase. This exact process was used by the robot to negotiate its way down a staircase that is known to be nearby.

The first part of this simple control scheme requires only the measurement of current position. A classical controller of some kind would form a position error and act accordingly. This sensing task could potentially be accomplished using only encoders as a dead reckoning setup. A more complete solution would involve an absolute locator such as the global position system.

The second part of this scheme, dealing with the staircase, requires at least a set of sensors that can sense an obstacle blocking the path, or the onset of a drop-off. This task could be accomplished with only a few ranging devices configured to monitor the space to the front and rear, both horizontally, as well with a vision cast downward.

It is desired in this research to pursue a more intelligent and robust layer of control that can plan ahead, bringing safety and efficiency to the task of climbing stairs. In order to accomplish this as simply as possible, a minimal sensor set is desired and the integration of sensing and control is carried out in a modular fashion. Whenever possible, actuators are made to respond autonomously from higher level planning functions.

Feature recognition: **Wall Identification**

Data from the laser scanner is used to identify and characterize features of interest to the robot. Locating a staircase is the primary function of the laser.

Looking at a staircase from the bottom, the stair treads are not readily noticeable. It is primarily the stair risers that are scanned by the laser, and these appear to be short walls that are progressively farther away as the scanner is tilted up. In order to find a staircase, and align to it, the laser looks for the wall like feature of the stair risers.

In order to obtain a scan of one complete stair riser from end to end, uninterrupted by the stair tread, the laser is positioned horizontally. Starting from the starboard side of the scan, a sequence of range data is chosen four points long. This data is then analyzed for linearity.

$$C_1 = \tan(-Angle) = \frac{n\sum XY + \sum X \sum Y}{n\sum X^2 - \sum X \sum X} \qquad C_2 = Range = \frac{\sum X^2 \sum Y - \sum X \sum XY}{n\sum X^2 - \sum X \sum X}$$

$$NonLinearity = \frac{\sum \left| X - C_1 Y - C_2 \right|}{n}$$

$$Discontinuity = \max_i \left(\sqrt{(X_i - X_{i-1})^2 + (Y_i - Y_{i-1})^2 + (Z_i - Z_{i-1})^2} \right)$$

If these characteristics satisfy the criteria, the line segment is accepted. The port end of the segment is then extended by one point, and the process is repeated. If a criteria test is failed, the entire segment is rejected, and a new segment, is chosen, again four points long. The third data point from the end of the rejected segment becomes the beginning point of the new segment.

If the line segment is accepted, its length is measured, and the segment is logged if the length is longer than any previously accepted one. After the entire scan has been parsed and analyzed, the resulting line segment is the longest one which satisfies all the criteria.

Top step detection

Lola advances slowly until its front mounted infrared sensors notice a dropoff. If both sensors see the ground disappear simultaneously, it is assumed that the robot is square to the stairs. Otherwise the robot continues to advance until the other infrared sensor also notices the dropoff. Given the known distance between the two sensors, and the assurance that the yaw rate remains zero, the angle between the staircase and the robot can be calculated. The robot must only watch for a maximum distance traveled while the second sensor looks for the dropoff. This maximum should be roughly two thirds the overall length of the robot, since any further movement would risk one tread falling over the stair case. Having detected such a situation, the proper response would be to back up, turn toward the known direction of the stairwell, and repeat the search procedure.

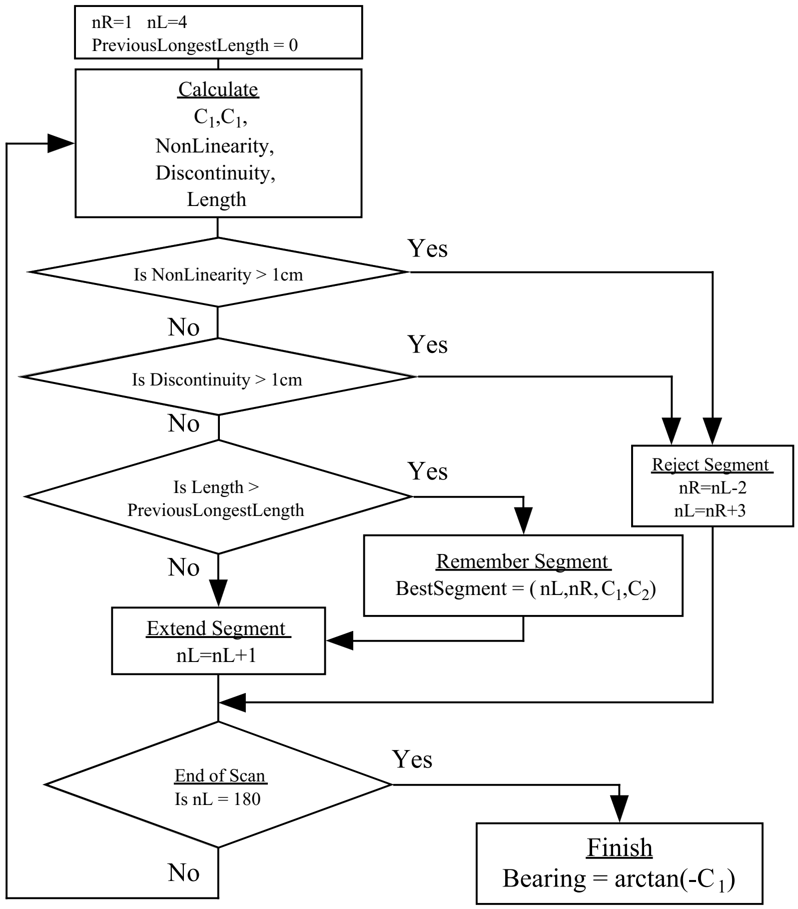

Figure 1 Line fitting algorithm

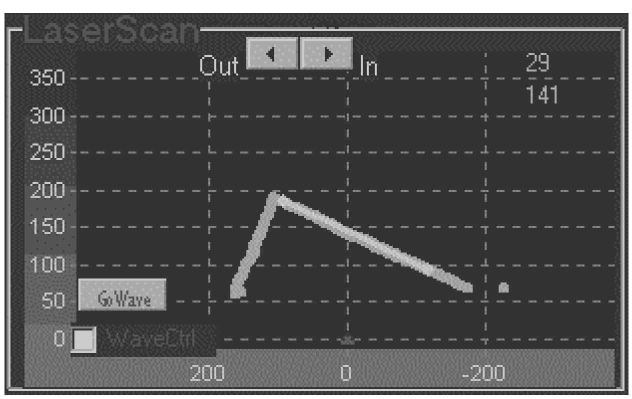

Figure 2 Line fitting results

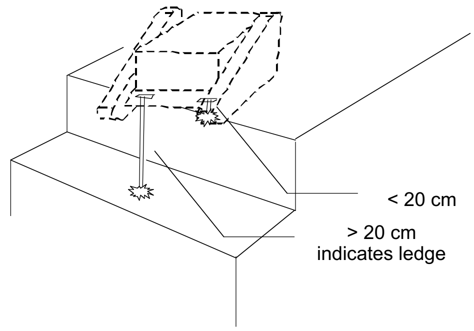

Figure 3 Top Step Detection

Stair negotiation

Both stair negotiating algorithms utilize the same sequence, but with different triggers and actions associated with each step. This is feasible because the stair sequence is basically symmetrical.

Stair Ascent			Stair Descent	
Action	Completion Criteria	Common Sequence	Action	Completion Criteria
Single scan; Set Hdg	Wait for HdgErr = 0	Find Stairs	Call sequence FindTopStep	Subroutine finishes and HdgErr =0
xScan, Find slope	20<Slope<60	Verify staircase	Scan for stairwell	Hole 1m wide
About Face	Wait for HdgErr = 0	Align to Stairs	Not used	
Backward slow	Pitch < -30	Move until rotate	Forward slow	Pitch < -30
Backward medium	Pitch > -10	Move along stairs	Stop until stable, then fwd	Pitch > -25
Backward slow	Pitch > -5	Rotate to flat	Forward medium	Pitch > -5
Backward slow	Move > 0.5m	Clear staircase area	Forward medium	Move > 0.5m
About Face	Wait for HdgErr = 0	Complete sequence	Not used	

Table 1 Stair sequence

Ascent

From the bottom, the stairs look like a series of short straight walls that are progressively farther away. The robot, making the assumption that previous commands have brought it close to a staircase and oriented it nearly facing the stairs, scans the area and chooses the longest continuous section of obstacle data. Inferring that this is the stair case, the robot then turns to face this stair riser directly. In order to verify that it has indeed faced a staircase, and not a wall, the robot tilts its laser up and down the staircase, calculating the average gradient directly ahead. If this matches an acceptable staircase, with an angle between 20 and 65 degrees, the robot accepts the situation and moves to the next step.

Since the robot must go up stairs backwards, it performs an about face, after which it assumes it is aligned to the staircase and is free to travel up them. The robot then backs up until it senses that its pitch down has exceeded 30 degrees, the criteria for being mostly on the staircase. It must be noted here that the roll controller is active for this entire sequence. This guarantees that, even if the robot is not well aligned to the stairs, contact with the first step of only one tread will cause a roll that will in turn cause the robot to correct the misalignment automatically. This feature remains active while the treads are fully on the staircase as well.

Once fully on the stairs, the robot moves backward slowly, watchful for the end of the staircase. This is indicated by the rear looking range sensors which detect the disappearance of the last riser, and the maximum distance above the upper ground surface. Fulfillment of such a condition causes the release of the rear easy down, and the robot continues its backward motion until the pitch changes to near horizontal.

Once on level ground again, the robot backs up to clear the stairs, a preset distance, and then turns around. This completes the stairclimbing sequence, and the controller signals the robots readiness for the next task.

Descent

Again assuming that previous commands have brought the robot nearly to the staircase, the robot begins its stair descent by ‘feeling’ for the staircase threshold. It slowly moves forward until one of the front mounted infrared sensors detects the disappearance of the ground. The robot then moves forward and measures the distance until the other sensor detects the top of the staircase. This motion data, together with the known distance between the two sensors, yields the position and orientation of the top step. The robot then backs up and aligns itself to the dropoff.

The robot then uses its laser to verify that there is indeed a stairwell of sufficient width to negotiate. Upon verification, the robot releases its front easy down and begins moving forward until the pitch down occurs as the center of gravity travels over the ledge. It should be noted here that, since gravity and robot motion are in the same direction for the descent sequence, motor controls must be somewhat delicate. A full charge forward would result in a bumpy ride. The robot therefore halts and the brake is set as the chassis pitches forward on the easy down lever, settling smoothly onto the staircase. Once the pitch data indicates the robot is fully on the staircase, and after pausing briefly, the robot commands the easy down to be retracted, and then eases itself down the staircase. The sequence continues on the staircase until the pitch is seen to be increasing toward level, indicating that the robot has reached the ground. This position requires more power, so the motor speed is increased and the robot travels until it achieves a level attitude. The final step to be executed on flat ground is to move forward 0.5m, clearing the area.

Results:

The algorithm described above has been implemented on Lola and exercised numerous times. The robot relies on having some approximate knowledge of the location of stairwells. This is required because the robot can easily outrun its stairwell detection sensors when moving at full speed. However, when combined with ded-reckoning using motor encoders and a floor plan map, or GPS and a campus map, Lola is capable of traversing complicated terrain autonomously.

Bibliography:

1. M. Fair. *Autonomous Stairclimbing With A Mobile Robot.* M.S. Thesis, University of Oklahoma, School of Aerospace & Mechanical Engineering, September 2000.
2. SICK Ag., *Laser Measurement System* LMS 200,posted at www.sickoptic.com/pdfs/laser/pilms2e.pdf ,1999.
3. Sharp Inc., *GP2D12/GP2D15 Distance Measuring Sensors*, posted at www.sharp.co.jp/ecg/NewProducts/sensor/gp2d12.pdf , May 1997.
4. Watkins, B. R., *Cushioning mechanism for stair-climbing wheelchair*, U.S. Patent 4,915,184, April 10, 1990.

Acknowledgements:

The authors wish to thank Austiaj Limited Partnership and KISS Institute for Practical Robotics for providing the Quest chair from which *Lola* was created.