AS-84.3200
Automation- and systems technology project work

Autumn 2010
Final Report

- Study points : 2
- Name of the projects work : Ceilbot-Odometry
- Starting date : 01/10/2010
- Ending date : 07/12/2010
- Author : Riccardo Bruno, Automation departement, bruno.riccardo@gmail.com
- Supervisor : Tomi Ylikorpi, David Leal Martinez
- Worksheet : Realize trolley odometry

November 30, 2010

Content

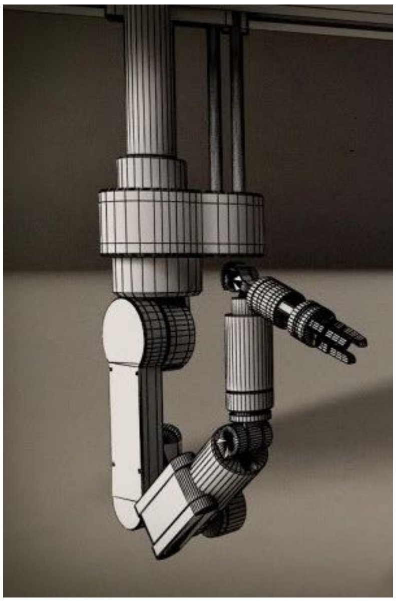
1 Preface

In this work I had to study a lot of new and unpracticed things, then I follow the following procedure :

1. Get the point of the problem
2. Get and analyze some documents about it
3. Read the solutions already proposed (study the documentation)
4. Think about a possible solution
5. Implement it

I spent most of my time finding and reading the articles that usually were more complicated than my needs. Also I miss the last point, in part because I have been slower than expected and in part because a little lack of experience in practical work and team work (had been difficult get the specifics). However I can say that it have been a very useful experience.

2 Introduction



The ceilbot project is one part of the home automation and then of the services robotics. Build a robot that can move on the ceiling of houses permits to avoid the problem of the obstacles on the ground, fast changing environments, people avoidance and also make it useful for some activities like patrolling, move things, help to get up disabled people, etc etc.. Obviously it can be also do the work of the ground robots like clean or move/take thing. The past project groups of ceilbot determined that the best way to move on the ceiling is a rail system mounting a trolley on the robot. They also choose to mount a manipulator to give more versatility.
This year the goal of group project is create a prototype of the robot.

2.1 The goal

My group goal was realize an Odometry system that, with the mapping system, allow the robot to move on the rail knowing his exact position on the track (on the ceiling) and then allow the possibility of planning actions.

3 Description of the work

3.1 Preliminary information

We agreed that the robot will not be faster than 2m/s and the trolley group said that maybe it will be slower. I also supposed that we will have a map of the track realized using an algorithm done by the other part of localization group (we divide the work in subgroups).

3.2 Odometry

The Odometry is a basic solution, used by most robots: it consist in using the Knowledge of the wheel's motion to estimate the entire vehicle motion.

Main Problems:

- Sensor error: If is not used a quadrature encoder there can be some noise in the measurament.
- Slippage: when the robots turn a corner a wheel usually slip a little bit, then even if the odometry is perfect we cannot recover the exact path.
- Error in the estimation of the wheel diameter : even small error in the estimation of the diameter can bring large navigation errors with time.

3.3 My solutions

The redundancy in the data is a must in the case of something so important like the position the robot, then my idea for the odometry is use two incremental optical encoders : one in the motor axe, and another one mounted in an apposite wheel (mounted on the same line with the center of the robot as explained in [9], to be sure that we are going to read the right distance). Two encoders are the only solution to reduce eventually errors due to slippage that otherwise are unrecoverable. I did a simple simulation to achieve some experimental data to have an idea

3.3.1 Simulation

```
clear
clf('reset')
d= 0 ;
t = 0 ;%% time inizialing
sigma = 0 ;
d(1)=1 ;
for t=2:200
   d(t)=d(t-1)+0.5*rand-0.5*rand;
end
%d=2*rand(1,200) ;
tot=0 ;
totr=0 ;
for t=1:200
   sigma = tot*0.03 ;
   en = normrnd(d(t),sigma) ; %%datas from the encoder
   vt(t,:,:)=[t,d(t),en] ;
   tot = tot+en ;
   vtot(t) = tot ;
   totr = totr+d(t) ;
   vtotr(t)= totr ;
end
figure(1)
plot(vt(:,1),vt(:,2),vt(:,1),vt(:,3))
xlabel('Time')
ylabel('Distance Traveled in 1s')
figure(2)
plot(vt(:,1),vtotr,vt(:,1),vtot)
xlabel('Time')
ylabel('Total Distance')
```

I simulated 200 seconds of working, with a maximum voleocity of 2 m/s. I supposed a standard deviation of 3% from the average that is set as real value.

Between every simulated variation there cannot be more variation than the 50% (excessive acceleration is not possible in the reality).

In the following figure the blu line show the real value and the green the one read from the incremental encoder.
It show the total distance traveled until that time.

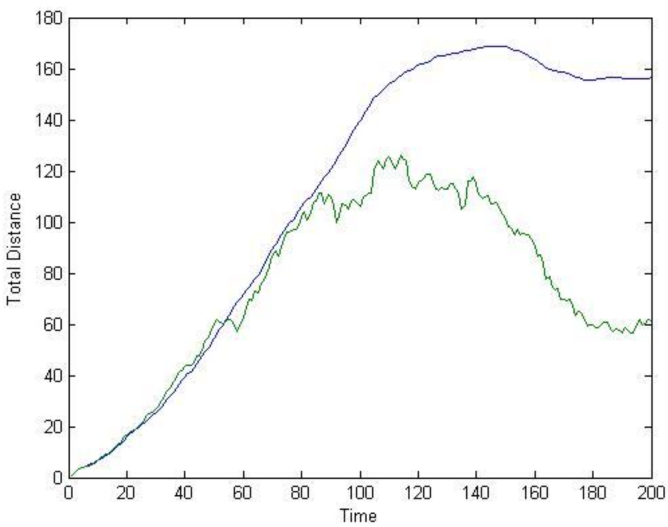


Figure 2: Difference in estimated and real distance traveled respect one point P

From the graph is evident that after 80 seconds of work the measurements aren't any more reliable, to be sure, I would fix the threshold on the 30th second.

3.3.2 Landmark for absolute positioning and error correction

A landmark/check point system is necessary for two purposes :

1. Know the absolute position on the track
2. Correct the estimated position at least every 30 seconds

On the initialization the robot would seek one landmark to know its position. It should do also this procedure after a determined time of work without meet any landmark. If the robot maximum velocity is 2m/s that means that the check points have to be at least one every 60 meters, but in the optic of redundancy and also because usually the robot will operate in smaller distance I suggest to put one every 10 meters. In that way the maximum position initialization time would be 5s. My suggestion is that we can implement a simple barcode on the rails: the velocity is not excessive and the barcode can be invisible to the human eyes.
A barcode with 12 bit would be more than sufficient, it give us the possibility to use 3 different ASCII character.

3.3.3 Practical implementation

To be sure of avoid error due to the sensors noises I would choose incremental optical quadrature encoders, I checked online some models for the encoder on the wheel [9], one can be normally bought for ~50€.
The Trolley group leader show me that when you choose a motor to buy you can also buy an encoder together, it always cost ~60€. One of the proposed motors mount a 201937: Encoder M

About the barcode system is difficult found a simply article about it but it shouldn't be difficult to implement.

4 Conclusion

The suggested system should be very accurate, I tried to apply the principle that I found reading about the odometry : redundancy. The total cost of the odometry system should be more or less about 200€. It is a shame that I could not found more interesting material about the barcode system.

4.1 Used time

Topic	Time spent	T. Estimated
Get the point	6 hours	3 hours
Read Articles	15 hours	15 hours
Go to the weekly meeting	30 hours	30 hours
Write documentation	25 hours	15 hours
Work on the prototype	X	20 hours
Total	76 hours	83 hours

4.2 Propose for the future

Can be useful implement also a little more complex odometric system, with two encoders (as done in most of the paper studied), that allow to draw the exact pattern followed (2D). So it can be overlapped with the map of the rail to get the exact position (it can be useful in case of malfunctions of the check points). Moreover can be added also a compass or another system [1,2,4,5] and the implement a data fusion with a Kalman filter.

4.3 The experience

I think that the project course have been a very interesting activity. I had to face new and challenging problems, most of them of about the organization and the groups works. It have been my first experience, maybe we missed our goals because we lack of experience in similar works (my university do something similar only at the end of the master course) and also some basic knowledge. Maybe it would give a better throughput work only on this course without follow other lectures. However I really appreciate it because had show me the applications of my studies in the reality.

References

[1]	Olson, Edwin. "*A Primer on Odometry and Motor Control*.", 2004. Web. Oct. 2010.
[2]	Hardt, V.D., Arnould, P., Wolf, D., and Dufaut, M.: "*Method of mobile robot localization by fusion of odometric and magnetometric data*", Int. J. Adv. Manuf. Technol., 1994, pp.65-69
[3]	Riccardo Bruno, Ceilbot Project Plan Localization Autumn 2010, 2010
[4]	L. Ojeda and J. Borenstein, "*Reduction of odometry errors in overconstrained mobile robots*" in Proc. UGV Technology Conf. 2003 SPIE AeroSense Symp., Orlando, FL
[5]	CAI, W., LEE, B., AND CHEN, L. 1999. "*An auto-adaptive dead reckoning algorithm for distributed interactive simulation*". In Proceedings of the Thirteenth Workshop on Parallel and Distributed Simulation (Atlanta, GA). 82–89
[6]	http://usdigital.com/products/e4p/ US-E4P miniature Optical kit encoder, datasheet
[7]	Wildi, T., R. Glaus (2002): A Multisensor Platform for Kinematic Track Surveying. 2nd Symposium on Geodesy for Geotechnical and Structural Engineering, Berlin, 2002
[8]	Products details and datasheet
[9]	R.Bruno, "*Ceilbot Intermediate report*" Autumn 2010,2010
[10]	Lu Xiangju, Fan Guoliang, Wang Yunkuan, "*A robust barcode reading method based on image analysis of a hierarchical feature Classification*", 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, Oct. 2006, pp. 3358 – 3362.
[11]	Photo of the arm from http://www.gavinrothery.com/site/bulletin.php