Annals of DAAAM for 2012 & Proceedings of the 23rd International DAAAM Symposium, Volume 23, No.1, ISSN 2304-1382
ISBN 978-3-901509-91-9, CDROM version, Ed. B. Katalinic, Published by DAAAM International, Vienna, Austria, EU, 2012
Make Harmony between Technology and Nature, and Your Mind will Fly Free as a Bird
Annals & Proceedings of DAAAM International 2012

MOBILE PLATFORM FOR ENVIRONMENT ANALYSIS

IONESCU, G[abriel] & BOIANGIU, C[ostin] A[nton]

Abstract: *This project studies the possibility of developing a mobile robot that scans the environment using 3D sensors. The project also focuses on developing an algorithm that analyzes the 3D data received from the 3D sensors and runs an image stitching algorithm based on the features found in the images. The goal is obtaining a large 3D panorama from all the pictures that the 3D sensor has sent.*
Keywords: *image stitching, 3D robot algorithms, robot communication*

1. INTRODUCTION

As technology progresses, all the industries that are directly or indirectly linked to it are changing. One of these industries is the entertainment industry. In this field there is a constant need for making the human-computer interaction more impressive and this is done by the means of technology. One of the ways of interacting with a modern electronically device is done graphically. In games, for example, the user sees 3D objects that have been modeled, usually by having a real object as a reference and imitating its shape. The problem with 3D modeling is that it requires a considerable amount of time to obtain a highly detailed object by starting from nothing. Currently there are a few technologies that help in the modeling process by obtaining a rough 3D shape, but none of these are based on gathering data through robotized systems. In the last few years, 3D sensors have evolved in such a rhythm that made them affordable to the general public. The improvements can be used in robotics and in gathering 3D data from the environment. This project studies the possibility of developing an algorithm that analyzes data from a 3D sensor and sends commands, based on the results from the analysis, to a robot that moves through the environment and scans all the nearby objects.

At this moment there are studies that focus on optimizing the acquisition of 3D images using time-of-flight cameras [9] and 3D reconstruction [10]. Opposed to current approaches, which focus on 3D reconstruction using proposed algorithms and solutions, this study has the goal of implementing a fast high-level algorithm that uses well established techniques for image analysis (e.g. feature detection [11]) and based on the results will generate a 3D panorama based on the input data. Another goal that this project aims to realize is the movement of a robot using input from the 3D sensor.

2. PROJECT REQUIREMENTS

The purpose of this project is building a mobile platform that communicates through any of the wireless technologies available today. This mobile platform has to be equipped with a device that captures information about the environment in 2D and 3D format and passes them to a central device that processes sequential data and matches the data through image processing algorithms, eventually forming one large piece of data that can be visualized in a 3D data viewer.

As stated before, this project should work without human intervention by sending movement commands to a mobile platform. This mobile platform should be composed of the following type of components:

1. A wireless receiver/transmitter.
2. Device that is capable of capturing data about the environment.
3. A powerful device that can run an operating system in order to provide more complex capabilities.
4. Electric motors for moving the platform.
5. Interface between the wireless receiver and the electric motors.

Currently there are a few large scale wireless technologies that can be used in transferring data between two distant points, but in order to obtain a high speed communication between two clients (at least 1 Mbit/s) we should look towards one of the most wide-spread technologies available: Wi-Fi. Current Wi-Fi technologies and standards support up to 150 Mbit/s [5] which should pose no problem in transferring data wirelessly even if the signal quality is lower than normal.

The device that captures information about the environment has to provide both color and depth information about the objects it faces. These two requirements can be fulfilled by having a device that performs two parallel tasks: providing 2D pictures of the environment (color) and 3D data (the depth information). The two pieces of data should also be closely synchronized so that one depth pixel should have an accurate corresponding color pixel.

From the computational point of view the project should have two machines that monitor and command the robotic platform. One of the machines should be powerful enough to analyze and rotate the 2D and 3D frames and take decisions based on the latest received frame and the other should be able to process commands fast enough to provide a reliable system and also to process the raw data from the 3D sensor. The two machines should also be able to communicate using the wireless technology described above.In order to make the robotic platform move through the environment it needs wheels and motors. As the devices installed on the robot will be mostly powered by batteries, the easiest way of moving a platform is by using electric motors. The interface between the wireless receiver and the electric motors will be made through a device that will receive the commands from the machine installed on the robotic platform. To avoid having large power consumption the best option for this device is a microcontroller.

2.1. Project Principles

The basic principle of this project consists of capturing 2D and 3D information about the environment and scanning two successive frames of data. For example, given N sets of images, any pair (k, k + 1), with k < N, that have common distinctive elements, meaning that these elements can be identified in both images, should provide information about the relative position of the image set k+1, relative to k. After N images the algorithm should output a result similar to a 3D panorama. Further processing can be done on the final dataset in order to obtain isolated 3D objects or textures from the environment.

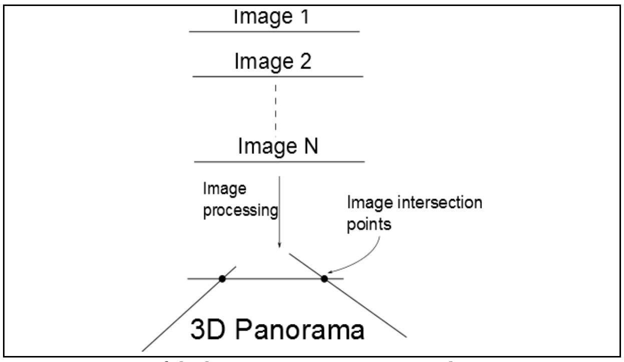

Fig. 1. Top view of the how a panorama is generated

The frames must be preprocessed in order to synchronize the 2D image to the 3D depth points. Often, cameras induce distortions in the captured images and can be problematic when there is a need for accuracy, but this problem can be solved by calibrating the camera and preprocessing each frame.

3. THE ADOPTED HARDWARE SOLUTION

The method chosen for finding the hardware solutions in building this robot starts with analyzing the mechanical components: the electric motors. One of the problems in choosing the electric motors is estimating how heavy the whole platform will be when all the components will be installed. By making a rough estimation of the final weight of the robotic platform the chosen solution was the “Dagu Wild Thumper 6WD” because it features 6 electrical motors that have a stall torque of roughly 11 kg/cm and it also has a large surface for installing all the other devices [3].

The second component for which a solution is needed is the device that will command the electric motors. One of the most popular platforms in recent years is the Arduino platform. The devices that use this platform span from simple input/output microcontrollers to more complex solutions that perform time-critical operations. Other fields that use Arduino in building devices are the power control field. As most of the devices based on this platform communicate through an USB cable that usually translates the data from the USB protocol to RS232 by using an FTDI RS232 chip, reduces the complexity in implementing a protocol for controlling the electric motors. This device is the “Wild Thumper Controller”. The main electrical characteristics of the motor controller are [2]: Dual 15A fuse protected H-bridges, Commands the motors via PWM, Controlled by an AtMega168 IC with Arduino bootloader.

As stated in the first chapter, in the last few years technologies have begun focusing more and more on developing 3D devices. One of the recent developed devices is the Microsoft Kinect. One of the great advantages that the Kinect brings is the fact that it does not need to be monitored continuously, thus saving processing power. The interface with the PC is done through one USB cable and a well-established software protocol. The Kinect can deliver 2D pictures and 3D depth information in a synchronized manner. The 2D pictures can vary in size from 320x240, 640x480 to 1280x1024, but the 3D depth information is limited to a resolution of 640x480 pixels. The depth data is limited to a range of 11 bits, meaning that the distance from the camera to a point from the field of view is quantized in 2048 steps. The accuracy of the depth camera is less strict, but ideal for the purpose of this project.

A method of having a good amount of control and analysis on the robot itself is achieved by using a computer with a considerable processing power, but it also has to be a small-form factor PC. Currently, most Linux based operating system support various processor architectures (arm, x86 etc.), but because x86 is still the most wide-spread, it is the architecture of choice for this project.

The chosen PC was the Alix3D3. This small-form factor PC consists of a 533 MHz x86-processor with various peripherals that can easily run a modern Linux distribution. Another advantage that this device has is the embedded Wi-Fi antenna that can allow us to connect it to a wireless network easily.

The device on which all the processing will be done is a more powerful PC that will basically generate the panorama.

3.1 The Communication Model

Because the project relies on a complex communication system it needs a well-established communication protocol that will be integrated in all the devices that are implicated in this process.

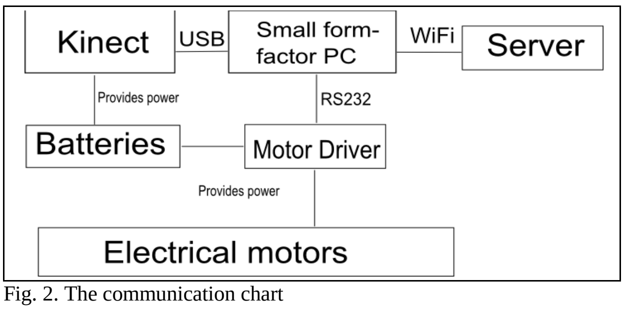

Fig. 2. The communication chart

The main component of the robotic platform is the small form-factor PC. The Alix 3D3 will receive commands from the server and will send commands to the motor driver, which will interpret the signals and pass them on to the motors.

The communication between the Server and the Alix3D3 will be made through a Wi-Fi network. The protocol will be Transmission Control Protocol (TCP) based, because of the low reliability of wireless communication and the large frames of data transmitted from the Kinect to the server through the PC. The TCP protocol provides reliable, ordered delivery of a stream of octets from a program on one computer to another program on another computer. Other applications, which do not require reliable data stream service, may use the User Datagram Protocol (UDP), which provides a datagram service that emphasizes reduced latency over reliability [1].

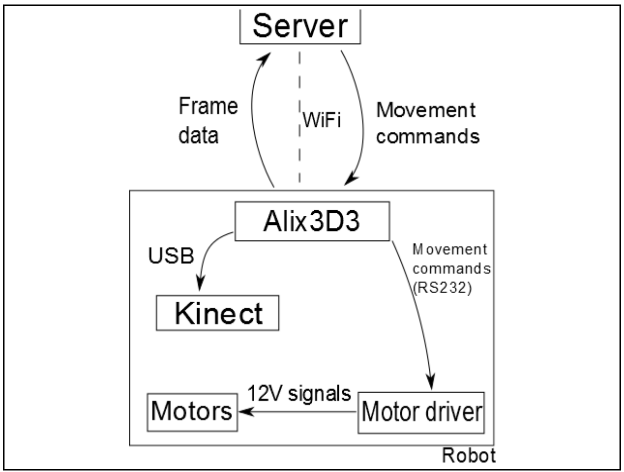

Fig. 3. Detailed communication model

A data frame sent from the Kinect to the server will not be compressed, due to the need of saving processing power. One frame of data will be as large as 640x480 pixels, each pixel having 3 channels of color (Red, Green, and Blue) or 3 bytes. The depth information will not be packed, also due to the need of saving processing speed. As stated before, the Kinect sends depth information by quantizing the distance 11-bits. Each depth pixel will round its storage needs to 16 bits (2 bytes). In conclusion, one pixel of information will have 5 bytes, amounting to 1,536,000 bytes per frame.

All the other movement commands are sent in small packages of 16 bits.

The communication model implemented on the RS232 protocol is based on short bursts of data. This method was chosen because of the reduced processing power of the AtMega168. For example for the Alix3D3 to command the motor driver to move the motors forward, it first needs to receive a command from the server. After the command is received the algorithm on the Alix3D3 will control the driver by sending a simple one byte packet (e.g. 0xAA).

4. SOFTWARE SOLUTION

From the software point of view there are 3 separate programs that are running on the three devices that this project integrates. These programs have the following roles:

1. On the server: visualize the data, 3D panorama stitching, movement-monitoring and 3D analysis of the latest frame of data received.
2. On the Alix3D3: receives movement commands, interprets and sends movement commands to the motor driver, acts as an interface for the Kinect, sends data back to the server.
3. On the motor driver: receives movement commands from the Alix3D3, monitors the electric motors, and commands the electric motors.

4.1 3D stitching process

This algorithm basically consists of overlapping a number of photographs in order to obtain a large resolution panorama. Presently this technique is implemented in various computer programs and embedded devices. The process is split into 3 steps:

1. Image analysis.
2. Image calibration.
3. Final picture reconstruction.

The image analysis step consists of searching the image for defining features. In our case the images are analyzed in groups of two.

The image calibration step consists of reducing the difference between an ideal lens and the camera lens, in order to eliminate distortions, exposure differences, vignette or chromatic aberrations. In the algorithm implemented in this project, this is a critical step in obtaining accurate panoramas, because if the images are distorted and not well calibrated there may be problems in aligning the 3D information.

The final picture reconstruction normally consists of using the features obtained in the first steps and running a feature matching algorithm. The goal of this step is obtaining a panorama that does not expose the margins where the images have been stitched together. The algorithm implemented in this project does this by rotating the data sets in 3D space [4].

4.2 The process running on the devices

Basically the server runs an application that is built on top of a light-weight 3D engine. When the application runs it will go through the following steps:

1. Sends a request for a frame to the Alix3D3.
2. Performs a waiting process in a semaphore.
3. Receives the frame and increments frame count.
4. Analyzes the frame and checks if the robot can move forward.
5. If it cannot move forward, it enters a decision process: it will analyze the entire frame by weighting the 3D points. If the points on the left side of the image are closer than the points on the right side of the image, then it will move right, else it will move left.
6. Send the image analysis results to the Alix3D3.
7. Select the previous image for the image stitching algorithm and start the process.
8. The algorithm will return the Euler angles, panorama size and reference points for the new image, relative to the position of the old image.
9. If the values are valid results, then generate a rotation and translation matrix and reposition the current frame.
10. Save the last valid picture position as a global reference point for the next picture.
11. Go to step 1.

The algorithm running on the Alix3D3 consists of multiple threads that run in parallel. The need for parallelization is given by the fact that there are multiple roles that the device has to fulfill at the same time: listen for movement commands, listen for new frames from the Kinect and send movement commands. When launched the program will do the following:

1. Allocate buffers for the data frames.
2. Initialize the Kinect.
3. Launch the Kinect monitoring thread. This thread has the role on configuring the Kinect at runtime, meaning that it can change the picture resolution or change the angle of the cameras.
4. Launch the RS232 monitoring thread. This thread waits for data from the motor driver. It is used to maintain a keep-alive communication between the two devices.
5. Launch the network monitoring thread. This thread waits for data from the server and interprets it on arrival.

The process that runs on the motor driver consists of a simple loop that monitors the serial communication pipeline and interprets the data received. If it has to move the robot forward it will apply a 12V voltage on the motors.

5. RESULTS

One of the most complex parts of this project is the stitching process. As this is a delicate part of the project, it requires much attention and careful calibration. In Fig. 4. there are 2 overlapped frames that form a 3D panorama.

Fig. 4. 3D image stitching result

At this moment more work is needed on this step of the process as it is not very accurate. This can be seen in the Fig.4. where the two images are overlapped on the right side of the picture. Also, some additional processing on the intermediate data [6][7][8], with the purpose of finding a better alignment may be used to increase the accuracy of the process.

5.1 Limitations

One of the biggest limitations that this project has is the fact that the Kinect is mounted on a platform close to the ground, making the objects more detailed on the bottom and less detailed on the top.

Another disadvantage that this implementation brings is the limited resolution of the images. If the user wants to obtain a high quality image, he is limited to working with depth images of 640x480.

6. CONCLUSION

This project can be developed in the direction of more complex processes, as the current work is merely a small example of what can be done with a robot similar to this. For example an object recognition algorithm based on the shape of the object can be easily implemented due to the fact that the whole framework is already built.

The next step in developing this project is making a more complex object detection algorithm based on predetermined planning of the path that the robot will explore. For example when the scanning algorithm is started, it will scan the room from its starting point and based on the information on the environment from that point, will be able to perform the scanning process more efficient.

7. ACKNOWLEDGMENTS

The work presented in this paper was funded by the Sectorial Operational Programme Human Resources Development 2007-2013 of the Romanian Ministry of Labour, Family and Social Protection through the financial agreement POSDRU/89/1.5/S/62557.

8. REFERENCES

[1] http://en.wikipedia.org/wiki/Transmission_Control_Protocol, (2012). Transmission Control Protocol – Wikipedia, The Free Encyclopedia, *Accessed on: 2012-10-01*

[2] https://www.sparkfun.com/products/11057, (2012). Wild Thumper Controller Board – SparkFun Electronics, *Accessed on: 2012-10-01*

[3] http://www.pololu.com/catalog/product/1561, (2010). Pololu - Dagu Wild Thumper 6WD All-Terrain Chassis, Silver, 75:1, *Accessed on: 2012-10-01*

[4] http://en.wikipedia.org/wiki/Image_stitching, (2012). Image Stitching – Wikipedia, The Free Encyclopedia, *Accessed on: 2012-10-01*

[5] http://en.wikipedia.org/wiki/Wi-Fi, (2012). Wi-Fi – Wikipedia, The Free Encyclopedia, *Accessed on: 2012-10-01*

[6] Boiangiu, C.A., Spataru, A.C., Dvornic, A.I. & Bucur, I. (2008). “Usual Scenarios and Suitable Approaches Used in Automatic Merge of Scanned Images”, *International Journal of Computers*, pp. 340-349, ISSN: 1998-4308

[7] Boiangiu, C.A., Bucur, I. & Spataru, A.C., (2008). “Statistical Approaches Used in Automatic Merge of Scanned Images”. *Annals of DAAAM for 2008 & Proceedings of the 19th International DAAAM Symposium*, Editor B. Katalinic, Published by DAAAM International (Vienna, Austria), pp. 0129–0130, Trnava, Slovakia, October 22-25, 2008, ISBN 978-3-901509-68-1, ISSN 1726-9679

[8] Boiangiu, C.A., Spataru, A.C., Dvornic, A.I. & Bucur, I (2008). “Merge Techniques for Large Multiple-Pass Scanned Images”. *Proceedings of the 1st WSEAS Int. Conf. on Visualisation, Imaging and Simulation (VIS ‘08)*, WSEAS Press, pp. 72 – 76, Bucharest, Romania, November 7-9, 2008, ISBN 978-960-474-022-2, ISSN 1790-2769

[9] Castaneda, V., Mateus, D., Navab, N., Stereo Time-of-Flight, *Computer Aided Medical Procedures (CAMP)* Technische Universitat¨ Munchen (TUM), Germany

[10] Cui, Y., Schuon, S., Chan, D., Thrun, S., Theobald, C., (2010). “3D Shape Scanning with a Time-of-Flight Camera”, *Computer Vision and Pattern Recognition (CVPR)*, San Francisco, pp 1173-1180, ISSN 1063-6919

[11] http://opencv.willowgarage.com/documentation/cpp/feature_detection.html, (2012). Feature detection – OpenCV v2.1 Documentation, *Accessed on: 2012-10-01*