Bibliographic report - Dog Robot

Robotic Project - School year 2022 – 2023

Students:

Hugo DURAND

Younes BAZI

Year Manager:

Pascal MASSON

Guillaume DUCARD

Robotics Project 3rd year, University Polytechnic School of Nice Sophia-Antipolis, Robotics Department,930 Rte des Colles, Parc de Sophia Antipolis, 06410 BIOT

SUMMARY

Table of contents

Introduction

The robotics program at the Polytechnic University of Nice Sophia trains engineers capable of mastering the design stages of an autonomous robotic system. An autonomous system is one that can achieve a given set of goals in a changing environment gathering information about the environment and working for an extended period without human control or intervention (*BlackBerry QNX definition*). Autonomous robots can be used to improve the speed and accuracy of routine operations, particularly in warehousing and manufacturing spaces; work side-by-side with humans for added efficiency; and reduce the risk of employee injury in dangerous environments.

The objective is thus to create a robotized system with the JN30D-Nano (NVIDIA) supervised by Mr MASSON and Mr DUCARD, two teacher-researcher. We have some manufacturing constraints. We have a size limit of 20 cm x 40 cm x 30 cm under penalty of devoting an excessive amount of time to the realization of the mechanical parts and for concerns of transport. Finally, we have a limited common university budget. We must choose electronic components and materials that are not too expensive, while meeting the design needs of the robot.

A year ago, as part of the Electronics-Arduino course in the preparatory cycle, I had the opportunity to build a dog robot from scratch. I spent a lot of my free time working on the construction of the body, the gearing for the movement of a leg, and more particularly on the hardware and software. Raphaël ANJOU, my associate for the project, and I, took time to understand the power requirements of the motors, the importance of a good controller card. After two or three tries of cards, the SSC32 servo controller card worked like a charm. We also learned how to work on PlateformIO to build a well-structured algorithm using several classes and functions in C++. We also learned how to work on PlateformIO to build a well-structured algorithm using several classes and functions in C++. This is very useful, especially to be able to configure a specific electronic environment on which to upload the code to the Arduino UNO board. In the end, our robot had difficulty to move because of its heavy weight compared to the strength of the motors, or probably because of the low efficiency of the gears to maintain the initial torque of the servo motors. So, we identified possible improvements we could make to a second version of the dog robot and went on to do so. We rebuilt the body this time out of aluminum instead of a 3D printed plastic body (which was deemed fragile and not suitable for the purpose). In addition, we opted for 1.4x larger legs (more powerful to lift the body), a herringbone gearing (avoiding unintentional degrees of freedom) and using 35kg/cm servo motors instead of 13kg/cm.

Figure 1.1: PolyDog_version1 (fully 3D printed)

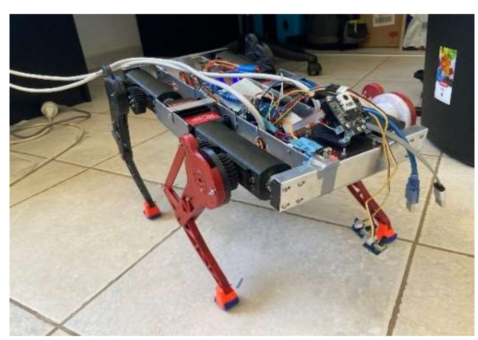


Figure 1.2: PolyDog_version2 (aluminum body)

Problematic and Robot Choice

Our problem is to build an autonomous robot that responds to a mission in the field of agriculture, surveillance, delivery, or cleaning.

Some of the class opted for a robot that plants in arid areas, a robot that cleans boat hulls or a robot that maintains vineyards. Others chose to go for a biomimetic robot, such as a snake or a spider. To do cartography, a pair of people want to build a submarine.

For my part (Hugo), I don't have a particular mission in mind. I want to continue what we started with the dog robot with Raphael. As we can see from people and animals, legged locomotion has the potential to handle rough terrain better than any other locomotion types.

It is a subject of research and competition between large companies such as Boston Dynamics or Unitree Robotics which invest in this kind of project because they see in it an incredible automation of tasks in the future. They would be able to replace the current military fighting dogs, perform boring or dangerous tasks for humans, make package deliveries by being able to climb on pavements or rough terrain that a robot on wheels could not do. A very simple example, SYDNEY ZOO uses Unitree A1 robot dogs for monitoring cheetahs.

However, legged locomotion requires complex leg mechanism, high power density actuators and advanced control methods, which have prevented the widespread use of legged robots for a long time. In addition, our robot dog PolyDog is not conclusive until now and I refuse to move on to another project before I can achieve what I set out to do. These are the reasons why I want to continue with a robot dog by bringing him in a first time even more solid mechanical parts and more powerful electronic components. Then, once the mechanical construction is completed. I would be able to concentrate on the walking of the robot, how it moves with a more uneven terrain.

Since my childhood, I (Younes) have been fascinated by robots, and I was always curious to know how they work, and what are the mechanisms behind their movements. I consider myself lucky, because I integrated robotic major in Polytech Nice-Sophia which allows us to build our robot in the first year.

Humans have always tried to copy nature, because nature is harmonious, as we can see in most of robotic projects (humanoid, cheetah robot, snakebot , robot dog ……..), and that’s the reason why I choose to build a robot dog, because I want to make a robot capable not only to navigate freely without any help from humans , but also capable to do so many tasks, tasks that are dangerous for humans, repetitive tasks that consume a lot of time for men. But because we are just at the beginning of our academic journey, we decided for this project to build a robot dog called Polydog that should walk forward and backward, turn to the left and right , crouch and jump since we consider to make our robot qualified to avoid obstacles. In order to make that possible, we plan to use ultrasonic sensors as well as depth camera, hence, we intend to introduce in our project image processing and Artificial intelligence.

Seeing what Hugo have done last year in this project gives me some hints on the problems that wait for us. The command of legs is more difficult than the command of wheels, since it involves mechanical challenges (leg modelling, the choice of material….) and dynamic challenges (the study of forces), because the major problem in our projects will be the balance of Polydog.

Requirements specification

Reference	Functions	Norm	Level
FP1	Moving in a flat environment	Size	40cmx20cmx10cm (fold up) max so as not to take too long to make the parts and avoid too much weight
		Weight	2,5kg for not buying motors that are too expensive, too much energy input making the robot's autonomy limited
		Degree of freedom	6 degrees of freedom: he will be able to curl up, jump, move forwards, backwards, on his right and left side.
FP2	Autonomy	Depth camera	1 (Infront) seeing 10 meters ahead, Collision avoidance, 3D scanning, volumetric measurement
		Battery	Use of a battery to allow it to move outside and in a space not limited to one cable length.
		Orientation sensor	IMU will provide for us information about our robot's inclinations, which will help us to stabilize more the robot.
FC1	Adhere to a smooth floor	Grip Foot	Ninja flex or other material to form the feet. The aim is to have a good support on the ground to move forward as efficiently as possible.
FC2	Aesthetics	Design	Aesthetics are very important for the general public's view and simply to please (color, form, hide the wires).
FC3	Facilitate easy assembly and disassembly	Design	The different parts must be easy to dismantle to be able to test, change a defective motor or have access to the control board for a code upload if necessary.
FC4	An ergonomic and user-friendly result	Ergonomic	The robot must be easy to operate by any user, without having to know anything about electronics and/or computers science

Building Body

Material

In the first version of Polydog, we built the body only in 3D printing, the esthetic of the robot was very good, but the heavy layers of PETG filament made the dog slow and its movements very unstable.

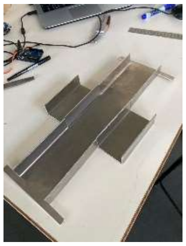


Figure 2.1

We then recreated a new body with Aluminum for the protection cage of the Arduino and SSC32 boards, keeping the legs in 3D printing (the latter weighing practically nothing compared to the body). But we failed in terms of lightness (300 grams heavier). We believe that most of the weight comes from the large support plate as seen in the figure 2.1. A thinner plate might have been the solution, however, it might bend a little when the robot moves forward, as was the case with the wooden board of the very first test.

We learned that in the school's Fablab, we have a machine for machining parts. So, we would like to cut the body parts out of carbon fiber with more precision than with a Dremel on aluminum. Carbon fiber (figure 2.2) is replacing aluminum in an increasing variety of applications and has been doing so for the last few decades. These fibers are known for their exceptional strength and rigidity and are also extremely lightweight. There more, carbon fiber, does not conduct electricity well and can often be used as an insulator to protect against electric shocks.

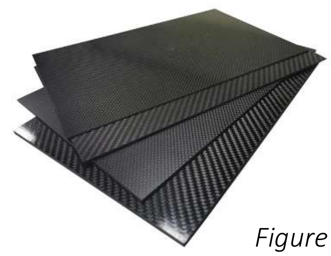


Figure 2.2

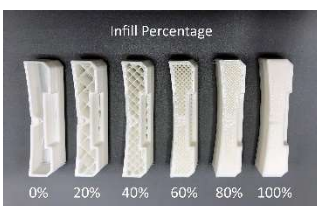


Figure 2.3

The legs made of PETG, or PLA filament are a lit bit fragile mechanically. On the first two versions, we used this type of filament, but with 20% filling. Some of the parts that were subjected to a certain amount of force on the ground broke. Looking at some internet pages, this type of filling is suitable for non-functional parts or light-use parts. A 40% filling should also be tested to overcome this problem.

In addition, on Polydog version 2, Due to the greater force of the servo motors generating more heat, the housings containing two servos have heated up and you can see a melted part on two housings.

Another observation is that when a leg does not move properly, it is sometimes due to excessive friction between two parts of a joint. By dismantling the leg (a very annoying job when it must be put back in the right offset), we sand the surface and put a little surface lubricant. The leg will immediately feel better.

It would therefore be preferable to insert metal joints in the articulation for greater fluidity.

Finally, to ensure a good grip on the ground, a grippy material should be chosen. In the past, we used ninja flex filament for the legs, adding a blue grip to increase the adhesion.

The Stanford robot Doggo’s feet are silicone pieces that they made using a 3D printed, 2-part mold. Ultrarobotics use squash balls for his robots.

Leg Modelling

Moving the knee with a chain, with a gear to keep the motors close to the body.

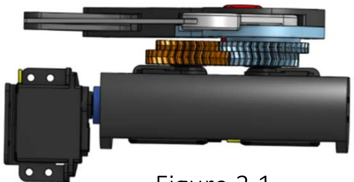


Figure 3.1

When modelling the leg, you have to think about the location of the motors, the structure of the different parts, i.e. the length, the shape, etc. This is essential to optimize the mechanical transmission of the motors to the movement of the leg. I'm going to start with the location of the engines because at the beginning with Raphael, we particularly focused on that.

After seeing some examples of robot, we saw that for an optimization of the movement, to keep a light leg, it is necessary to keep the motors centralized at the level of the robot body.

Then you must think about how you operate the knee movement from a distance. As far as we know, there are two possible solutions, either by means of a belt or chain, or with the parallelogram system. After seeing a video of LokiLeDev on his channel, we opted for the second solution for the first two versions of the robot.

The principle of the parallelogram system is to have a two-piece femur, each connecting the lower leg at two separate points. The hip motor will drive one of the two parts of the femur. As we could not find a way to superimpose the motors at the hip, we used a reduction gear to move the whole leg with a little more torque at the output. The second motor, the one at the knee, moves the upper part of the tibia by tilting or raising it.

Figure 3.2

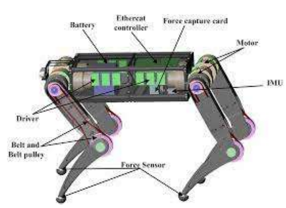


Figure 3.3

The belt system consists of placing a motor in the hip which drives a belt and turns a gear at the knee to control the lower part of the leg. This system also allows for better control of the torque output, as a larger sprocket can be placed at the knee.

Spot mini has servo motor on the knee, same for dizzy wolf, but we can see from the circular shape, he has brushless motor instead of servo. Lakaigo has a strap around the femur and Kangal has the same system as our dear Polydog_V1.

Spot mini

Dizzy wolf

Lakaigo

Kangal

Electronics components

Brush servo motor

Brush servo motors feature simple two wire control, easy installation & operation, and low cost. Brush Servo Motors can operate in extreme environments due to a lack of electronics, usually have replaceable brushes for extended life, and do not require a controller when operating at a fixed velocity.

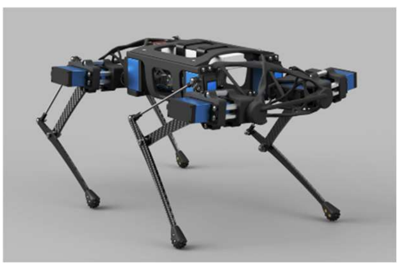


Figure 4.1

This walking Quadruped Robot DIY has twelve Servo, SPT5435LV-180 with a torque of 35kg/cm. This is a video of how this robot move: https://www.youtube.com/watch?v=Y6QYdh4bs70. These servo motors are large torque metal gear digital servo. We used this kind of servo motor for our robot, with the same torque. We have not reached the same level of agility. I think this is due to the fact that it is much lighter, and we did not focus on training the robot on software. Indeed, we just chose angles that described the movement of the legs best to our eyes. In this video, you can get an idea of how it looks like https://www.youtube.com/watch?v=g1pQitycG0I.

We also have found another servomotor: The Smart Servo MS-12A. It has a torque of 12kg/cm and it's fully programmable (via a serial interface) and benefiting from a system of 5 information feedback (position, speed, temperature, current and voltage). This would allow us to better target the problem if some servos are not doing the right action. It have also a smart protection to overcurrent, overvoltage, overheating. However, some of our teachers' personal feedback on this type of product is unfavorable.

Brushless servo motor

Brushless DC (BLDC) technology has been around in one form or another for several decades. Brushless Servo Motors have windings in the stator and permanent magnets attached to the rotor. No brushes are used. Motor rotation is achieved by means of electrical commutation performed by the servo drive. According to Warren Osak, Founder & CEO of two Industrial Automation Distribution companies specializing in Robotics and Factory Automation, Brushless servo motors provide high acceleration, high torque, and no maintenance. Brushless Servo Motors offer the highest torque-to-weight ratio and are commonly used in the highest throughput, precision, and demanding applications. Most of the dog robots that have been created use brushless motors.

They come in all price ranges.

Standford Doggo robot use T-motor 5212 KV340 ($109.90).

Xdog, BE8108 8108 brushless motor ($100).

Dogger from Ultra Robotics, Yet and Ghost Robotics' Minitaur use Tarot TL68P07 6S 380KV 4108 Multi Rotor ($38).

Figure 4.2

There are also gimbal motors. A gimbal motor is a 3-phase brushless motor whose stator is wound with many turns, and as such exhibits much larger resistance and inductance compared to the high-current brushless motors used to provide lift to drones, rc planes etc. Gimbal motors are used in… well… camera gimbals mainly, because they offer smooth motion and require small currents to produce torque compared to the 'regular' brushless motors. This in turn can help minimize the size of the motor driver and associated wires.

How to choose the right brushless motor

To choose the right brushless motor, you must look at several characteristics: weight, size, constant motor velocity (Kv) or rotation per minute (RPM) which is related by the relation V x Kv = RPM, maximum thrust, and motor resistance. It is also possible to choose a motor that runs very, very fast, add a gearbox to it and thus get more torque at the output.

The lower the Kv of a motor, the more torque it will have and therefore the more suitable it is to drive large propellers. But the lower the speed, the lower the RPM (revolutions per minute). In the case of a dog robot, we don't really need a brushless motor that turns very, very fast, the opposite of drones. It is then preferable not to have a lot of KV and to privilege the thrust.

Brushless Motor	KV	Weight	Maximum thrust	Price
Tarot TL68P07 6S 380KV 4108	380 KV	93g	1620g	38euro
Emax rs2205 2300kv	2300KV	30g	1281g	7euro
T-MOTOR MN5212 KV340	340 KV	205g	2100g	109euro
MAD5008 EEE 300KV	240KV	135g	3403 g (6S)	64euro

On the image opposite, I found some brushless motors, the Emax are very economical but require a gearbox which is not necessarily very efficient in output.

For the time being, servomotors are more than adequate. The torque of the servos is high (35kg/cm). The problem of functioning comes from elsewhere since at no load some legs do not perform the adequate movement. This research may be useful later.

Motor Driver / Motor controller

The main difference between a motor controller and a motor driver is that the controller is responsible for controlling the speed, torque, the direction of the motor whereas a motor driver is responsible for providing enough electrical power to the motor as required.

A motor controller might include a manual or automatic means for starting and stopping the motor, selecting forward or reverse rotation, selecting and regulating the speed, regulating, or limiting the torque, and protecting against overloads and electrical faults.

During the two versions of the dog robot, we used different controllers and servo motor drivers. As these are the boards that take care of distributing the juice from the power supply to all the servomotors, it is vital to choose a board that can deliver what each motor needs. For example, when using the Mega Strong 2560, we had this type of problem: https://www.youtube.com/shorts/KD5D6o_LaiM. By changing to a ssc32 servo-controller specialized in controlling many servos with two separate power supply channels V1 and V2, allowing a much better current distribution.

In the case of using brushless, I noticed that several dog robots opt for an Odrive 3.6 motor driver controller. It is said to be a High-Performance Brushless Motor High-Power Dual-Drive Controller FOC BLDC. Sensorless Field Oriented Control (FOC) is one of the methods used to control a BLDC motor's speed and torque. Field oriented control (also known as vector control) is a technique used to generate a 3-phase sinusoidal modulation which then can be controlled in frequency and amplitude. For a better approach of how it works, here is a article about it : https://www.pmdcorp.com/resources/type/articles/get/field-oriented-control-foc-a-deep-dive-article.

The DC power voltage input range of the Odrive: 24V version: 12V to 24V. 56V version: 12V to 56V.

Current: Peak current 120A per motor.

Encoder

Encoders convert motion to an electrical signal that can be read by some type of control device in a motion control system, such as a counter or PLC. The encoder sends a feedback signal that can be used to determine position, count, speed, or direction.

To enable electronic speed control (ESC) to move the motor to a precise position, we need to know the current position of the motor. Therefore, we need an encoder.

There are two types of encoders: incremental rotary encoder and absolute encoder. Incremental rotary encoder outputs the pulse corresponding to the rotation angle only while rotating and is the counting measurement method that adds up the pulse from the measurement beginning point. On the other hand, an absolute position sensor gives information on its position within a given scale or range without the need for a reference point. In other words, it has a static reference point. Absolute encoders, whether optical or magnetic are ideal in situations where speed and position accuracy and fail tolerance are absolute musts.

Figure 5.1

With the use of an absolute rotary position sensor, the absolute position is determined by power-up. It doesn't require a reference mark. In fact, many believe that absolute rotary position sensors offer better performance, better precision, and lower overall costs.

Figure 5.2

The doggo robot used four AS5047P, a high-resolution rotary position sensor for high speed (up to 28krpm) angle measurement over a full 360-degree range. They only used the incremental interface, because at the time they built the robot, there was no absolute encoder support on the ODrive.

When using brushless motors, position sensors are essential for feedback on the angle of rotation. But for a servo motor that is told what angle it should go to, we take a lot of work out of the code.

JN30D-Nano Card

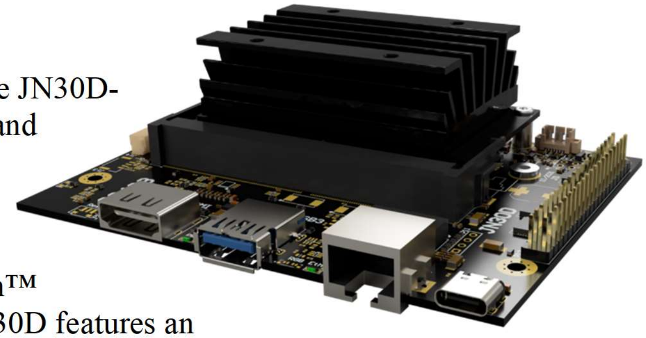


Figure 6.1

We do not have to choose the central brain card. We will use the JN30D-Nano Card for our robot project. I've read up on the Internet and have a small overview of the specifications of this card. It will work in communication with the microcontroller.

Auvidea JN30 Carrier Boards are designed to convert the Jetson™ Nano™ compute module into a super minicomputer. The JN30D features an industrial strength design and is commercially deployable in any volume. These systems let you dive into AI with ease. They allow you to explore AI applications like people detection, face masking and more.

- 128 GB SSD includes: Linux, Jetpack 4.6, SDKs (VisionWorks, DeepStream and more)
- requires Linux host PC and Internet connection to initially flash the system and install the software
- a large variety of add on modules available from Auvidea
- industrial strength design

Power 6V - 19V (5.5/2.5mm connector)

Autonomy

Power supply unit / Battery

Currently, we have a power supply unit that delivers 5V and 50A. Every motor need between 5-8V for functioning normally and the stall current is 2600mA ± 10%-3400mA ± 10%.

In order to choose the battery capacity for our robot, we need in first place sum up all components power , then we multiply this sum with how many hours we want our robot to work (we will choose 1H, because the more hours you want your robot to work, the more the battery will be bigger and heavier), after that we divide it with the tension of the battery we want (battery of 5V , 6V ,12V….).

Equipment	Voltage and Current	Quantity	Power (MAX)
TD-8135MG Digital Servo	4.8 -7.2 Vdc 140 mA-200mA	12	1W
JN30D-Nano Card	6V – 19V 3A	1	18W
BNO055 orientation sensor	5V 50 mA	1	250mW
SparkFun Load Cell Amplifier - HX711	2.7V-5V 1.5mA	4	7,5mW

HC-SR04 Ultrasonic	5V 15Ma	10	75mW
Intel RealSense D455	5V 700Ma	1	3.5W
SSC-32 servo controller	5v 31mA	1	155mW
Arduino uno	6v-20v 20mA per I/O pin	1	4.4W
TOTAL	39 W		

The max power of the embedded system is 39W. for an autonomy of 1h, it takes an energy of 39Wh. Using only a battery of 6V, we will need a load of 6.5 Ah. however, we must always take into account the fact that the state of charge of the battery must always be superior than 20% (that means that after 1 hour of use, the battery must not be completely discharged). This aims to increase the lifetime of the battery. We therefore choose to use a 6V- 10Ah lithium battery for a safety margin.

Batteries that we can use in our robot:

lithium 6V 10Ah LTB06010:
weight: 583g, dimensions: 85 mm x 57 mm x 76 mm,
lifetime: 2000 cycles

lithium Lifepo4 6V 10Ah:
weight: 580g, dimensions: 55mm x70mm x 90mm,
charge time : 6h

NiMH rechargeable Battery 6V 10 Ah:
weight: 810g, dimensions: 165mm x 33mm x 65mm
charge time: 5h-6h

Figure 7.1

Figure 7.2


Figure 7.3

Choice of battery to be postponed when we have a better vision of what our robot needs and when its movements are so important that a battery becomes essential.

Inertial measurement unit (IMU)

An Inertial Measurement Unit (IMU) is a device that can measure and report specific gravity and angular rate of an object to which it is attached. An IMU typically consists of:

- Gyroscopes: providing a measure angular rate
- Accelerometers: providing a measure specific force/acceleration
- Magnetometers (optional): measurement of the magnetic field surrounding the system

In other words, IMU will provide for us information about our robot's inclinations, which will help us to stabilize more the robot.

Adafruit MMA8451:

The Adafruit MMA8451 module detects basic motion, tilt, and orientation with a digital accelerometer. This is an excellent accelerometer to start with. It is designed for use in phones, tablets, smartwatches and more, but works just as well in your Arduino project. High precision and low cost (9, 31 €). It's the most precise of the family.

Figure 8.1

There are other products that detect the orientation of the robot on 9 degrees of freedom, like this one, with more precise calculation systems and algorithms that justify a higher price.

Adafruit 9-DOF BNO055:

Bosch is the first company to get this right by taking a MEMS accelerometer, magnetometer and gyroscope and putting them on a single die with a high-speed ARM Cortex-M0 based processor. They determine the output with a finer degree of accuracy.

Figure 8.2

UM7 Orientation Sensor

- EKF estimation rate: 500 Hz
- ±1° typical static pitch/roll accuracy
- ±3° typical dynamic pitch/roll accuracy
- ±3° typical static yaw accuracy
- ±5° typical dynamic yaw accuracy
- 0.5° angle repeatability
- 0.01° angular resolution

Figure 8.3

However, the last solution has more complex mechanisms and is more precise, but we do not know if this is of much use for our robot. We think, we will therefore choose Adafruit BNO055 module.

Ultrasonic sensor

Our Project is an autonomous robot dog (Polydog), which means, that the robot must be equipped with obstacle detectors in order not to collide with walls and different obstacles.

Among the ultrasonic sensor, we have the HC-SR04 ultrasonic Range Finder, This HC-SR04 Ultrasonic Range Finder is a non-contact distance measurement module which is compatible with the Arduino or another system.

The problem with ultrasonic sensors is that they have a low angle range which is 10-15 degrees. We will need to put various sensors.

Specifications:

- Ultrasonic Frequency: 40k Hz
- Maximal Range: 400 cm
- Minimal Range: 3 cm
- Resolution: 1 cm
- Trigger Pulse Width: 10 µs

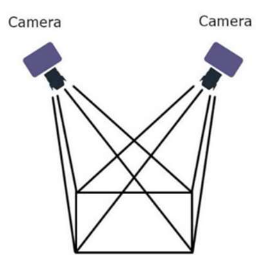


Figure 9.1

Depth Camera

Depth cameras have two sensors, spaced a small distance apart. A Depth camera takes the two images from these two sensors and compares them (figure 6.4). Since the distance between the sensors is known, these comparisons give depth information, in other words, a depth camera gives you 3d data.

But why should we use a depth camera instead of a normal stereo camera?

Depth cameras give you the advantage of additional understanding about a scene. While it’s possible for a computer to understand a 2d image, that requires significant investment and time in training a machine learning network. A depth camera inherently gives some information without the need for training. A depth camera on the other hand, has pixels which have a different numerical value associated with them, that number being the distance from the camera, or “depth.”

LiDAR system uses the light detection technique to calculate depth. It measures the time it takes for each laser pulse to bounce back from an obstacle. This pulsed laser measurement is used to create 3D models (also known as a point cloud) and maps of objects and environments. But lidar systems only provide 3D mappings of the object shape. It has limited capability to interpret roadway information like landmarks and drivable paths. In the other hand, depth cameras create high-definition mapping data by identifying target object shape, appearance, and texture.

Depth cameras are very useful in the field of robotics and autonomous devices, our robot will navigate its way around a space, so we want to automatically detect if something appears directly in front of the robot, to avoid collision.

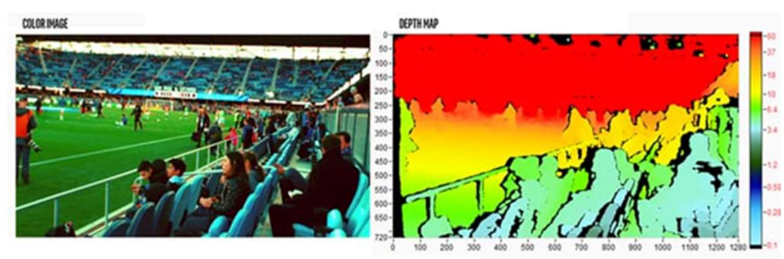


Figure 9.2

Here is a summary table of the cameras we found:

Sensor	Price (approx)	Depth FOV (vertical)	Depth Resolution
Azure Kinect	$400	NFOV: 65° WFOV: 120°	NFOV: 640x576 (very good quality!) WFOV: 512x512
Kinect 2 (Kinect for XBOX One)	$160	60°	512x424 (good quality)
Intel RealSense L515	$350	55°	up to 1024x768
Intel RealSense D455	$240	57°	up to 1280x720 (but noisy!)
First-gen depth sensors	from $30	45°	640x480 (interpolated, actual: 320x240 or even less)
Orbbec Astra (PRO)	$160	45°	640x480 (interpolated, actual: 320x240 or even less)
Intel RealSense D435	$180	65.5° (only 42° for RGB)	up to 1280x720 (but very noisy, inaccurate and with a lot of artifacts like waves and blur!)
ASUS Xtion 2	$270	52° (can be mounted in vertical position for 74°)	640x480 (interpolated: actual is less than 320x240)

We have not yet chosen a particular camera as it is still a long way off before we start ordering one. However, we are looking at the Intel RealSense D455.

Conclusion

To conclude, we have considered different solutions for our robot, in terms of materials for the body, the way to move a leg, the electronic components necessary for movement and autonomy.

We chose a parallelogram system for the movement of the legs with TD-8135 servo motors. We are going to replace the plate of the structure by carbon fiber for more lightness, study again the code, target the problem of movement of the motors, possibly change the defective servomotors.

As our research and new experiments progress, we will draw more precise conclusions on the subject. We will probably realize that some choices were wrong and adjust accordingly.

Summary of equipment order

Type of equipment	Quantity and Name	Already bought
Body structure	4x legged print in PETG	✓
	4x ninja flex foot with rubber coating	✓
	12x Servo Horn	✓
	1x carbon fiber plate	
	4x U aluminum (ball bearing and servomotor support)	✓
Electronics components	12 x ServomotorTD-8135MG	✓
	1x Ssc32 servo controller USB	✓
	1x Arduino uno card	✓
	1x Nvidia JN30D nano card	

	1x Power Supply 5V-50A	✓
	1x Adafruit BNO055	
	1x Intel RealSense D455	
	5x HC-SR04 Ultrasonic	

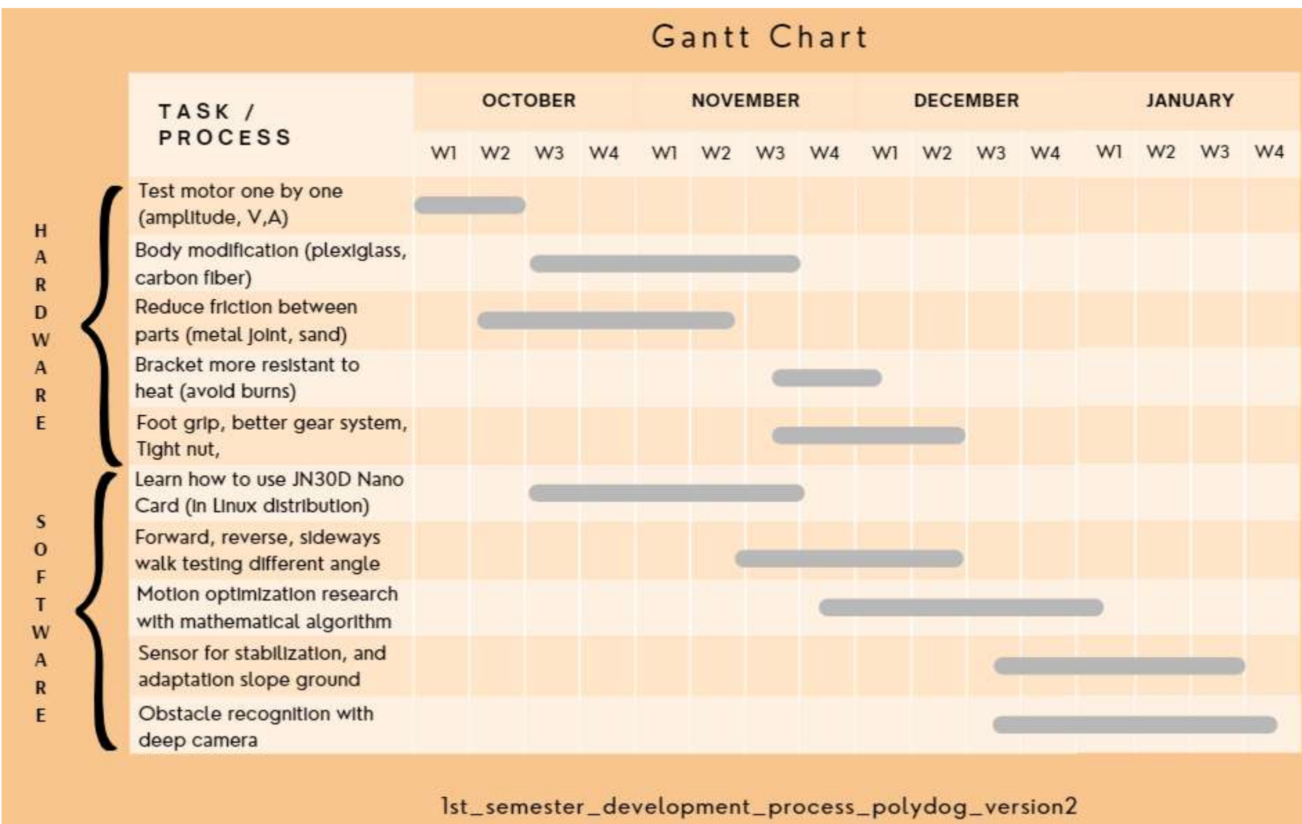
For the moment the last 3 components are not useful, our choices may change, be refined.

Gantt chart

This Gantt chart is still a predictive planning of the building and development process of the robot dog. Most of the time, the difficulty of certain tasks is misjudged, or external parameters corrupt the progress of the project.

Assuming that we start the project from scratch, the first thing we would do is to model the leg with the motors we have chosen, the body and the electronic components in order to get an overview. Then we would start cutting, printing, or machining the parts and building the complete robot.

However, having made a version of Polydog last year, we can build on it to improve it, and save a considerable amount of time, especially by not having to wait for electronic components to arrive by parcel.



On the software level, one must become familiar with the Linux interface to program on the Jn30d Nano card. It will handle the AI and image recognition processing. We will have to connect it with the Arduino board so that it communicates between them, to indicate to the ssc32 servo-controller the commands to the motors.

We wish to deepen the learning of the robot's walking, add a stabilization sensor and obstacle detector (deep camera) to make it more and more autonomous.

Source used

Definition autonomous system:

https://blackberry.qnx.com/en/ultimate-guides/autonomous-systems

https://www2.deloitte.com/content/dam/Deloitte/us/Documents/manufacturing/us-manufacturing-autonomous-robots-supply-chain-innovation.pdf

Example of robot:

Stanford Robot :

https://github.com/Nate711/StanfordDoggoProject

https://docs.google.com/spreadsheets/d/1MQRoZCfsMdJhHQ-ht6YvhzNvye6xDXO8vhWQql2HtlI/edit#gid=726381752

https://www.youtube.com/watch?v=2E82o2pP9Jo

Solo 8 : https://www.youtube.com/watch?v=LPX6YRbefBI

Ghost minitaur : https://www.hackster.io/news/meet-ghost-minitaur-a-quadruped-robot-that-climbs-fences-and-opens-doors-bfec23debdf4

Dogger : https://delft.makerfaire.com/maker/entry/235/

Klange : https://grabcad.com/library/quadruped-robot-w-code-1

A1 Unitree robot: https://www.youtube.com/watch?v=2H3dzZEi-qw&t=46s

Choosing body material:

http://fr.custom-composite.com/info/aluminium-vs-carbon-fiber-comparison-of-materi-25728106.html

http://www.dexcraft.com/articles/carbon-fiber-composites/aluminium-vs-carbon-fiber-comparison-of-materials/

Brush servo motor vs brushless motor:

https://www.linkedin.com/pulse/brush-servo-motor-vs-brushless-which-better-my-application-osak/

https://www.alliedmotion.com/brushless-motors-whats-the-big-difference/

Gimbal motor

https://tinymovr.readthedocs.io/en/latest/hardware/gimbal.html

How to choose the right brushless motor;
https://www.dronement-drone.fr/article/tout-savoir-sur-les-Kv
https://filmora.wondershare.fr/drones/drone-motors.html

What is a driver:
https://www.ourpcb.com/what-is-motor-driver.html
https://core-electronics.com.au/guides/motor-drivers-vs-motor-controllers/

Controlling position and how works bldc:
https://www.youtube.com/watch?v=uOQk8SJso6Q

What encoder to choose:
https://www.omronap.com/service_support/FAQ/FAQ01001/index.asp#:~:text=Incremental%20rotary%20encoder%20outputs%20the,the%20rotation%20angle%20by%20code.

Odrive: https://discourse.odriverobotics.com/t/pid-example-code/9144

Orientation Sensor:

https://www.adafruit.com/product/2472
https://www.pololu.com/product/2764
https://www.melopero.com/fr/shop/sensori/mouvement/Acc%C3%A9l%C3%A9rom%C3%A8tre-triple-axe-adafruit-%C2%B1-248g-14-bits-mma8451/

Tactile pressure sensors :
https://learn.sparkfun.com/tutorials/getting-started-with-load-cells?_ga=2.226078874.1093846988.1664745023-809767648.1664745023

Ultrasonic sensor:
https://www.robotshop.com/en/hc-sr04-ultrasonic-range-finder-tys.html

Depth Camera:
https://www.e-consystems.com/blog/camera/technology/what-is-a-stereo-vision-camera-2/
https://www.intelrealsense.com/beginners-guide-to-depth/#:~:text=Stereo%20Depth,-Stereo%20depth%20cameras&text=For%20a%20stereo%20camera%2C%20all,these%20comparisons%20give%20depth%20information.