SUDOKU SOLVER BOT

CLUB:ROBOTICS CLUB

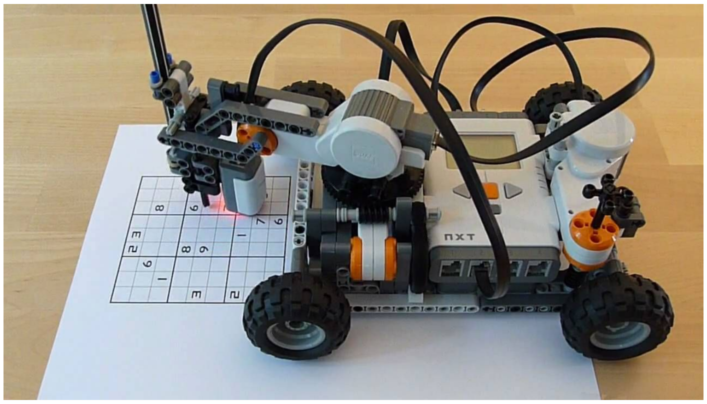
An automated bot which solve and fill sudoku puzzle completely by itself.

Youtube video Link	
Link to drive folder containing codes and all other resources	https://drive.google.com/open?id=0ByDK_y_Ss5KbNVpzZUE1dTBqYjA
Video of our bot writing the digits	https://drive.google.com/file/d/0ByDK_y_Ss5KbcGlJSVMydjF3alE/view?usp=drivesdk
Video of our code solving the puzzle after IP	https://drive.google.com/open?id=0ByDK_y_Ss5KbR2cwUHRyU3E2c2c

INTRODUCTION:

Our project **Sudoku Solver Bot** is an automated bot which solves and fill a sudoku puzzle completely by itself by first taking the picture of the sudoku by a webcam or a phone and then processing the image and then finally solving the puzzle and plotting the solution on the paper.

Our motivation for the project was this LEGO mindstorm sudoku solver bot.



TECHNICAL ASPECTS OF THE PROJECT AND THEORY INVOLVED:

1)IMAGE PROCESSING USING OpenCV:

OpenCV (Open Source Computer Vision) is a library of programming functions mainly aimed at real-time computer vision.OpenCV is written in C++ and its primary interface is in C++.. There are bindings in Python, Java and MATLAB/OCTAVE. The API for these interfaces can be found in the online documentation. The link to the documenation of the OpenCV is : OpenCV DOCUMENTATION we found it very useful and this would provide you with every functions available in OpenCV library.

We used this library to process our image of sudoku and extract the sudoku grid and the finally the digits.

2) K-nearest neighbour Algorithm :

In pattern recognition, the ***k*-Nearest Neighbors algorithm** (or ***k*-NN** for short) is a non-parametric method used for classification and regression.
OpenCV includes a statistical machine learning library that contains K-nearest neighbour algorithm. Using KNN in OpenCV :http://docs.opencv.org/2.4/modules/ml/doc/k_nearest_neighbors.html?highlight=digit .

We used this algorithm for identifying the digits of the sudoku , it's the easiest of the machine learning algorithms though not 100% accurate.

3) **Algorithm for solving sudoku puzzle:**

We used brute force algorithm which guarantees a solution no matter how tough the puzzle is.
The algorithm is written in C++ language and is highly efficient.

4) **Stepper Motors and Servo motors:**

Stepper motors, due to their unique design, can be controlled to a high degree of accuracy without any feedback mechanisms. The shaft of a stepper, mounted with a series of magnets, is controlled by a series of electromagnetic coils that are charged positively and negatively in a specific sequence, precisely moving it forward or backward in small "steps".

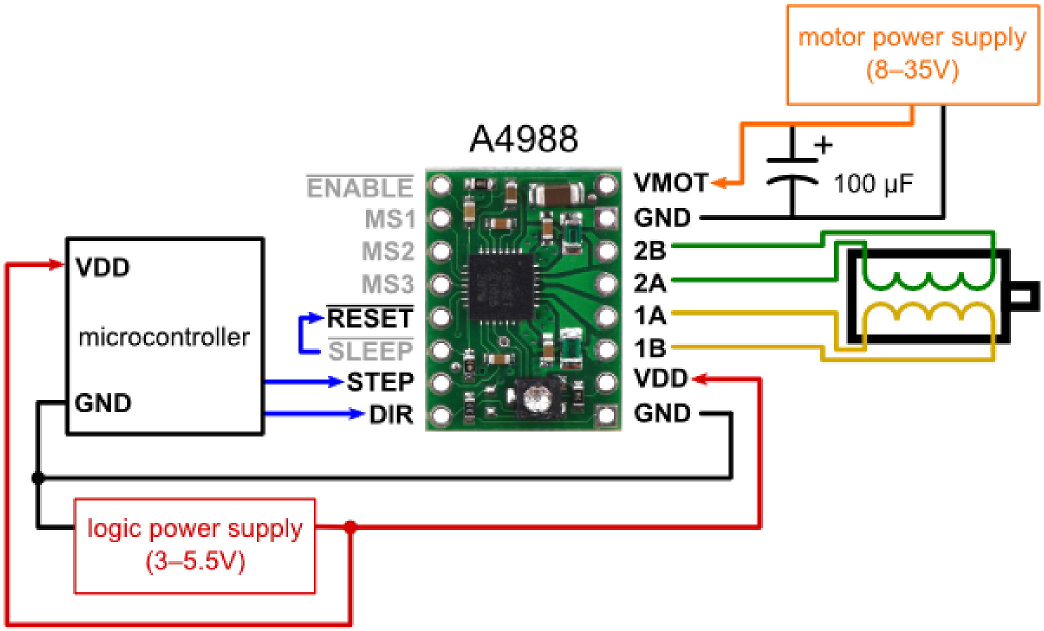
We used the stepper motors to give our bot precise movements in both horizontal and vertical direction All about stepper motors : .https://learn.adafruit.com/all-about-stepper-motors/what-is-a-stepper-motor

A **servomotor** is a rotary actuator or linear actuator that allows for precise control of angular or linear position, velocity and acceleration.Standard servos allow the shaft to be positioned at various angles, usually between 0 and 180 degrees. Continuous rotation servos allow the rotation of the shaft to be set to various speeds.

We used servo motors to lift the pen up and down.

5) Stepper Motor Driver :

We used a4988 stepper motor driver which has easy programmability and easy connections circuit diagram.



Here, 1A 1B 2A 2B goes to the stepper motor. VDD and Ground powers up the driver by supplying 5V.VDD and Ground Pins are connected to 5V and GND pins of Arduino board respectively . Arduino VMOT and ground for motor power supply as mentioned(we provide from 12 V battery)Do not forget to connect capacitors (atleast

47microfarads)between GND and VMOT otherwise there is high risk of damage to the circuit due sudden rise and fall in voltage due to LC oscillation . DIR is given a constant 5V or 0V which indicates the 2 directions in which stepper motor rotates, STEP is given pulses whose number is equal to the number of steps. MS1, MS2 and MS3 is used to control step size.STEP pin controls speed of motor You just have to change the value of delayMicroseconds in arduino code to change the speed of motor .Increasing the value of delay time decreases the speed of motor . According to various logical inputs given to the these 3 pins, the size of step decreases to half , one fourth , one eighth and one sixteenth of the step.**SLEEP** and **RESET** pins should be shorted.

6) Arduino to control Stepper motors and Servo Motor:

We used “Arduino Uno” due to its easy availability.Arduino is an open-source prototyping platform based on easy-to-use hardware and software.
Linkof a video which gives an idea of how to control stepper: controlling steppers
ARDUINO stepper library refrence : https://www.arduino.cc/en/Reference/Stepper
ARDUINO Servo library refrence :https://www.arduino.cc/en/Reference/Servo

Detailed Description

Image Processing via OpenCV :

A picture of sudoku puzzle will be taken by webcam or mobile.The picture should have the sudoku as the largest rectangle/square and image should be straight to produce better results.After taking the sudoku image,we processed the image using OpenCV library to filter out noise,extract the sudoku grid,identifying corners of sudoku and finally to extract the digits and blank spaces.
We used KNN(k nearest neighbour) algorithm for identifying the digits.
We created a training data with sample images of the numbers 1 to 9 and images of the blank grid , the training data can be seen in the resources folder in the link to our drive and then this data was used to identify our numbers , increasing the number of training samples would increase the accuracy of the numbers being correctly identified.
Once the complete sudoku is identified,we assign 0 to blank places and a 9*9 array of numbers is generated.
The link to our code and training samples we used : IP of sudoku image
The link to the our documentation of image processing:
https://docs.google.com/document/d/186TpeHTAbbqYqyyGf81kInH4kyfCQmutOtxEIa9j

4lo/edit?usp=sharing . It contains all the links and the topics we needed to study for our code and also contains the problems we faced.
Video which explains our IP:
https://drive.google.com/open?id=0ByDK_y_Ss5KbR2cwUHRyU3E2c2c

Solving Sudoku using brute force Algorithm:

We used brute force algorithm combined with backtracking recursion to solve the sudoku puzzle.The algorithm checks for every blank space and assign the first possible legal(allowed) number in increasing order of 1-9 in that place in accordance with the rules of sudoku.If it fails at some blank space,it starts from the previous blank space and assigns the second possible legal number and   in this way recursion continues. The link to the code is: Sudoku Solving Algorithm

Input Solutions to arduino:

Once the solution to sudoku is obtained we intend to pass on the solutions to arduino by various serial connection established between arduino and c++ code. It transfers 9x9 array to the arduino which it uses to go to various positions on the paper , check whether a number needs to be written and write the required solution as per the solution obtained.

Characteristics of the Bot:

We made our bot to go front and back and pen go left and write to solve the sudoku. For the bot to go front and back we used 4 stepper motors and for the pen to go sideways we used a lead screw attached to the shaft of stepper motor. On this lead screw we attached a nut to which we attached a wooden block which contained a hole to carry another rod and pen holder.When the lead screw attached to the shaft rotates , we restrict the rotation of the nut (another rod attached parallel to the lead screw) which makes the pen to translate sideways. Servo Motor attached to the pen via a thread makes the pen to go up and down.

Arduino Code used to control the bot motion for writing:

For controlling the stepper motors, we used A4988 stepper motor driver   to give motion to the motors as we intended. To run a stepper motor, all you need is a pulse which is equivalent to a single step in the stepper motor that is 0.9 degree rotation of the shaft of the stepper motor. So we coupled the 4 motors to give front and backward motion and a single stepper motor to give sidewards motion of the pen. For writing numbers we made functions involving various motions of the bot like front(int number of steps), right(int number of steps) and write_digit(int digit) which contained code to write pulses to the motors to move various motors and achieve several tasks.

For controlling the servo motors, we used the inbuilt servo motor library for arduino "Servo.h" which has various inbuilt functions for the the easy control of servo motor like myservo.attach , myservo.detach, myservo.write(angle) which enables easy motion of the servo motor which was used for controlling the pen_up and pen_down motion of the pen.

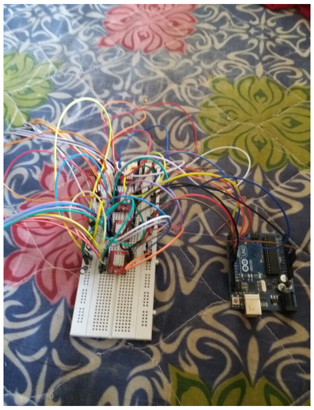
Challenges Faced:

1) Non availability of L clamp to hold the motor to the chassis.
2) Bending of coupling shaft used to connect stepper motor shaft and lead screw causing wobbling of the pen.
3) Limitation in covering the width of the page due to small size of the chassis.
4) For some sudoku images,the KNN algorithm could not identify all digits correctly.
5) Another problem faced was to control the movement of pen and to maintain accurate pen pressure.
6) While doing motor driver connections make sure that all connections are done properly otherwise there is a high chance of damage to the circuit.

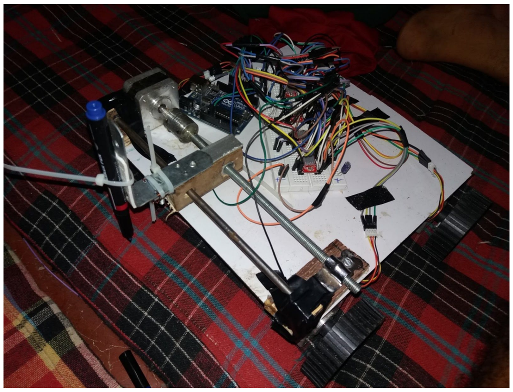
LIMITATIONS AND FUTURE DEVELOPMENTS:

1) Our code for identifying the numbers of sudoku is not 100% correct as we have used K-nearest algorithm which is not a very accurate , it identifies the number up to 95% accuracy so there might be errors sometimes. So, in future we would like to remove these errors and make our code 100% robust.
2) Our writing mechanism was not so efficient due to wobbling present in the lead screw assembly so in future we plan to change our writing mechanism so that pen can fill the digits clearly.

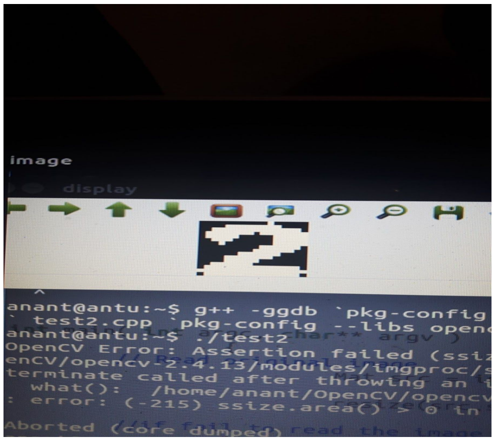
PROJECT PICS



Stepper motor driver circuit
with connections to arduino

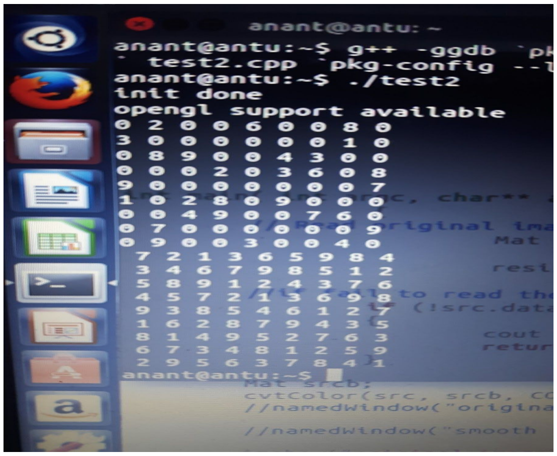


FINAL BOT



```
anant@antu:~$ g++ -ggdb `pkg-config
` test2.cpp `pkg-config --libs openc
anant@antu:~$ ./test2
OpenCV Error: Assertion failed (ssiz
enCV/opencv-2.4.13/modules/imgproc/s
terminate called after throwing an i
  what():  /home/anant/OpenCV/opencv
: error: (-215) ssize.area() > 0 in
Aborted (core dumped)
```

Extracted grids of the sudoku
containing digits



```
anant@antu:~$ g++ -ggdb `pk
` test2.cpp `pkg-config --l
anant@antu:~$ ./test2
init done
opengl support available
0 2 0 0 6 0 0 8 0
3 0 0 0 0 0 0 1 0
0 8 9 0 0 4 3 0 0
0 0 0 2 0 3 6 0 8
9 0 0 0 0 0 0 0 7
1 0 2 8 0 9 0 0 0
0 0 4 9 0 0 7 6 0
0 7 0 0 0 0 0 0 9
0 9 0 0 3 0 0 4 0
 7 2 1 3 6 5 9 8 4
 3 4 6 7 9 8 5 1 2
 5 8 9 1 2 4 3 7 6
 4 5 7 2 1 3 6 9 8
 9 3 8 5 4 6 1 2 7
 1 6 2 8 7 9 4 3 5
 8 1 4 9 5 2 7 6 3
 6 7 3 4 8 1 2 5 9
 2 9 5 6 3 7 8 4 1
anant@antu:~$
```

Output of two arrays : the identified sudoku
using IP and the final solved one.

All our pictures and videos can be seen here
:https://drive.google.com/open?id=0ByDK_y_Ss5KbYWJpQkJiY1d5TFk

COMPONENTS

Here is the list of components we have used to complete this project with the cost mentioned.

- 5 NEMA-17 Stepper motors.3kg-cm torque- ₹250 each **(Servo Electronics, Lamington Road)**
- 1 Servo motor -₹300**(Servo, Electronics Lamington Road)**
- Aluminium chassis-₹60(**Mangaldeep,IIT Market**)
- 4 Tyres -₹25 each (**Mangaldeep,IIT Market**)
- Arduino -Uno board
- Nuts and bolts.**(Mangaldeep,IIT Market)**
- 5 A4988 Stepper motor Driver-₹190/₹200 each (**Visha World, Lamington Road /Mangaldeep ,IIT Market)**
- Coupler.
- Lead screw (6mm diameter)-₹36 (**Mangaldeep,IIT Market)**
- Jumper wires-Male and Female(**Mangaldeep,IIT Market)**
- Bond tite -₹40 (**Mangaldeep,IIT Market).**
- Breadboard.
- PCB-₹60 (**Mangaldeep,IIT Market)**
- Acrylic Chassis-₹80(**Mangaldeep,IIT Market**
- 2 L clamp -₹10 each (**Mangaldeep,IIT Market)**
- Metal Rod
- Sketch pen and Marker
- DST tape-₹30(**Mangaldeep,IIT Market)**
- Sandpaper -₹10(**Mangaldeep,IIT Market)**

TOTAL COST: RS.5444

All our bills can be seen here :https://drive.google.com/folderview?id=0ByDK_y_Ss5KbY0JGZTZCZk9FUIE&usp=sharing

REFERENCES :

We thank **Team STAB and Robotics club** as well as our **Mentor** for helping us in the completion of the project.

Here is the link to document which contains all the link to the resources which helped us in completing our project.
https://drive.google.com/open?id=0ByDK_y_Ss5KbZ3pvd3E5TU84QXM .

LINKS FOR VARIOUS TUTORIALS WHICH WE REFERRED AND ARE USEFUL FOR ANYONE WHO WANT TO KNOW MORE ABOUT THE PROJECT.

1	**http://aishack.in/**
2	**http://aishack.in/tutorials/sudoku-grabber-opencv-detection/**
3	**http://opencvpython.blogspot.in/2012/06/sudoku-solver-part-1.html**