MINOR PROJECT

(Real Time Sudoku Solver)

SUBMITTED BY:

DEVANSH MARWAHA (18103028)

DEVENDRA SINGH (18103029)

DIVYANSHA SHARMA (18103030)

HARDIK PUNJ (18103039)

SUBMITTED TO:

Dr. AMRITPAL SINGH AND Dr. PRASHANT KUMAR
ASSISTANT PROFESSOR
CSE DEPARTMENT

COMPUTER SCIENCE AND ENGINEERING DEPARTMENT

DR. B.R. AMBEDKAR NATIONAL INSTITUTE OF TECHNOLOGY
JALANDHAR

TABLE OF CONTENTS

ACKNOWLEDGEMENT

It’s rightly said that there are hundreds of people working behind the curtain to make a Play successful. The final result of our project- **“Real-Time Sudoku solver”** required a lot of guidance and help from many people and we are extremely fortunate to have got this throughout the journey of our project. Whatever we are today is only due to such supervision and assistance and we would not forget to thank them from the bottom of our heart.

We are extremely thankful to Professor A L Sangal, Head, Department of Computer Science & Engineering who has provided all his direct and indirect support.

We are thankful to the In-charge, Major Project Final Year, for providing us mentor and all other support.

We are extremely thankful to have got constant encouragement and guidance from all the Faculties of Department of Computer Science & Engineering(CSE) who gave us their time and suggestions and thence helped us in successfully completing our project work. Also, we would like to extend our sincere esteems to all staff in laboratory for their timely support.

DECLARATION

We, hereby declare that our Project titled- **"Real-Time Sudoku Solver"** being submitted by us in the Department Of Computer Science and Engineering is a project work carried by us under the noble supervision of our CSE Department's faculty and the project has not been copied from anywhere and has been made solely by us.

We will be solely responsible if some Plagiarism is found.

Thank You.

Date: 24th May, 2021

1. INTRODUCTION - BUILDING UP THE FOUNDATION

1.1 Problem Statement and its Necessity

Sudoku: The sudoku game is something almost everyone plays either on a daily basis or at least once in a while. The game consists of a 9×9 board with numbers and blanks on it. The goal is to fill the blank spaces with suitable numbers. These numbers can be filled keeping in mind some rules. The rule for filling these empty spaces is that the number should not appear in the same row, same column or in the same n×n grid.

Our motivation to pursue with this project is from a learning perspective & explore areas like :-

I. Augmented reality –

1. It allows the user to see the real world, augmenting it with super-imposed virtual object. It is a technique for adding and supplementing digital content over real world using computers and mobile devices. A common practice is to use the cameras on these devices to record video, take pictures and draw additional content over it and display the result on the screen of the same device; a practical example is to translate signs and labels from one language to another.

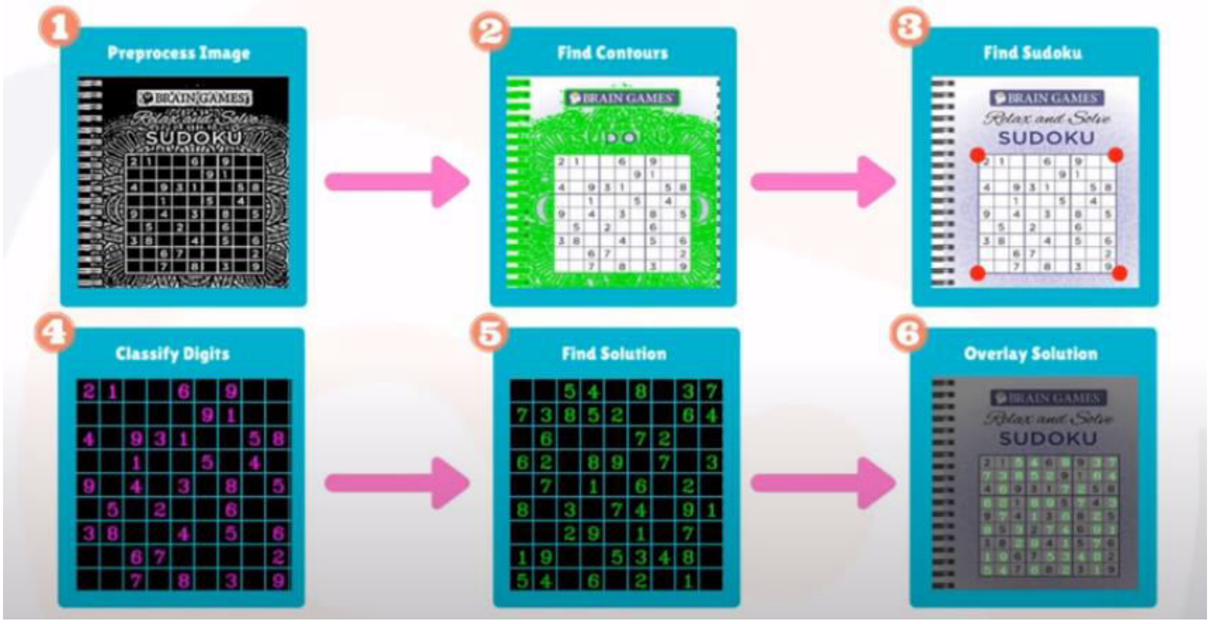
2. Another interesting impact of this technology is to improve accessibility. For example, wearable computers can be used to read affect on people's faces, which is helpful for those with autism spectrum disorder who may struggle with reading social cues. Nowadays this concept is also used in games like Pokemon GO.

Nowadays computers have a lot of processing capability even is common that has an integrated webcam, the need of clear and understandable images brought man-kind to create techniques to improve an image, if necessary reduce noise, blur, alter the brightness, color conversion, etc.

Our work

We have developed a program to detect the Sudoku and print the solution with augmented reality on the live image of Sudoku. We transfer only the locations of digits in the grid and identify if has a number or is empty, if a number is present no action is done but if the grid position is empty, automatically we put a zero.

Afterwards we send these numbers to Sudoku solver. Finally with the solution obtained, we proceed to print on the display the result, thereby obtaining **a program which solves Sudoku with augmented reality.**



1.2 Feasibility- Technical and Non-Technical

Technical:
The prerequisite needed for accomplishing the implementation of this project includes strong hold over-

- Python programming languages
- Open CV
- Computer vision application
- Uses augmented reality

Internet connectivity is required for the system.

Non-Technical:

a. Social: -
Good Pastimes & Solving wits and puzzles, in a way, helps to develop wit and ingenuity. OCR predictions are very spotty for hand written digits and stylized fonts

b. Economic Feasibility: -

This project doesn't require much cost in development.
Only requires cost for database management.

c. Scope: -

- Rewrite the program for Android, Integrate this in an Android app and publish it on play store.

- Parallelize some functions to use multicore processors. PCs have at least dual-core processor. Normally, parallel tasking is used to improve performance. But here we don't need more speed, we need more quality. The idea is that parallel tasks should perform the same operation on the same image frame but with different settings. After all tasks are joined, we will take the task with the best result and discharge others.

2. PROPOSED SOLUTION

2.1 Identifying Stakeholders

The stakeholder can be classified into three categories:-

a. Developers-
The Admin of our Application. The one - who coded the program. They have all the authorization to the Sudoku Solver.

b. Students-
Our End User is the Student. The one- who shall use the facilities being provided by the Application as their pastimes.

c. Researchers-
Those who use it reference & explore more in the field. They will collaborate int the product embedding new functionalities.

2.2 Detailed Solution

Sudoku Solver is the collection of very basic image processing techniques. A very good way to start is the OpenCV library which can be compiled on almost all the platforms.

Every recognition algorithm has these steps:

- Determine features
- Train (learning step)
- Classify (runtime recognition)

Steps we are going to use for this project:

- Build a Sudoku solver algorithm
- Use OpenCV to read blocks of Sudoku from image
- Recognize the number in the block
- Use the algorithm to solve the puzzle
- Fill the result on the image

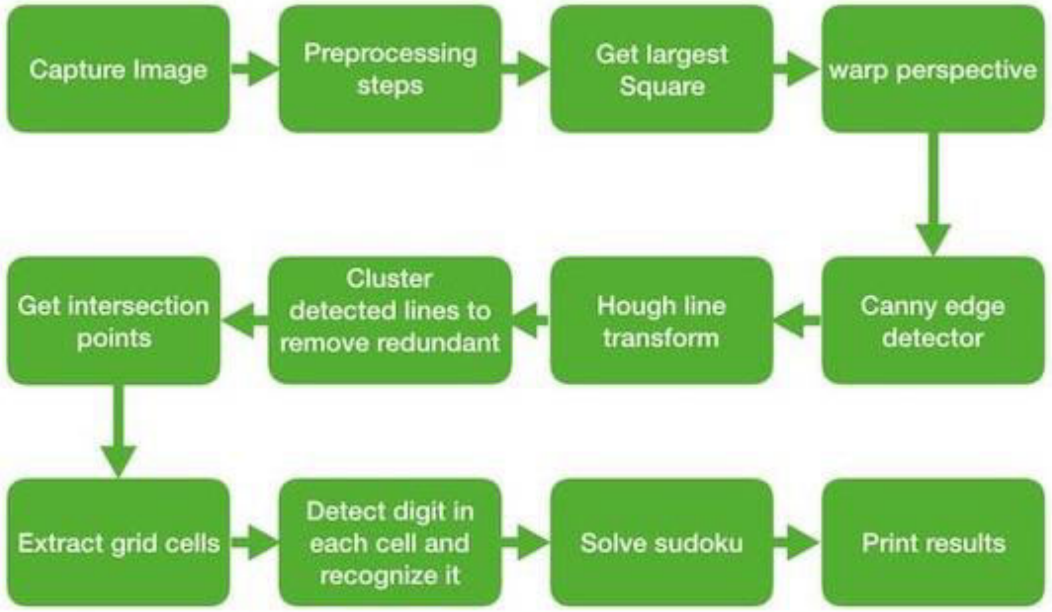


Figure 1: Major Steps of The Algorithm

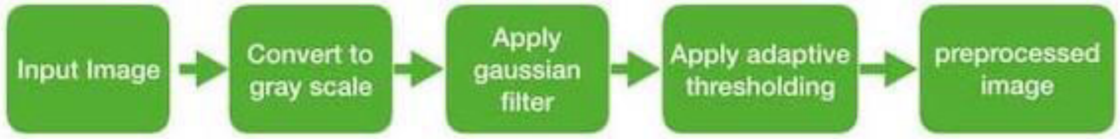
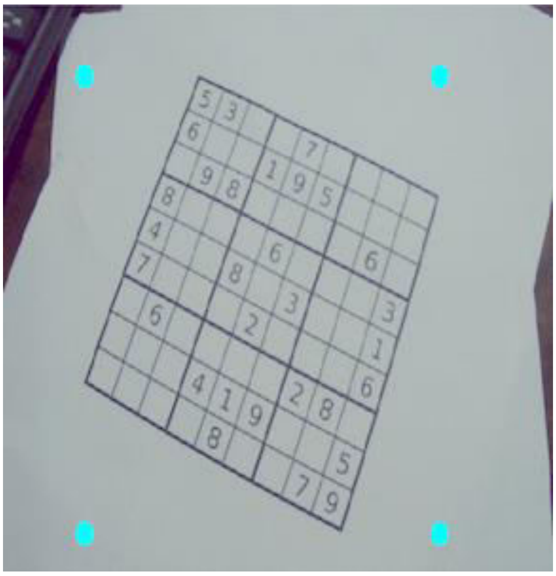


Figure 2: Preprocessing Operations

1. Capture image.

Put the capture object in a AtomicReference in order to be able to stop and start cam streaming in a thread-safe way.

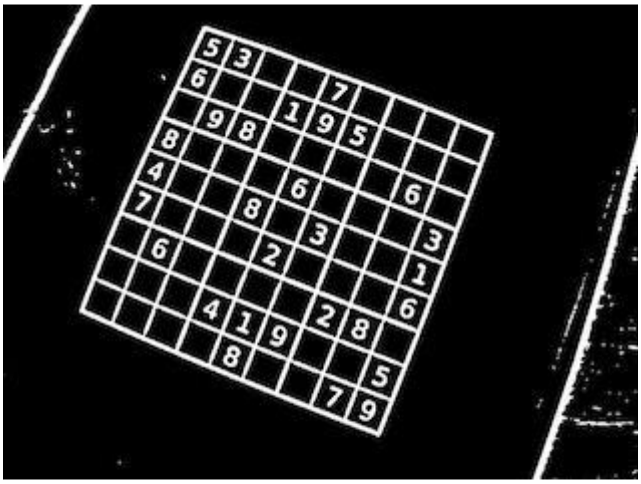


Input Image

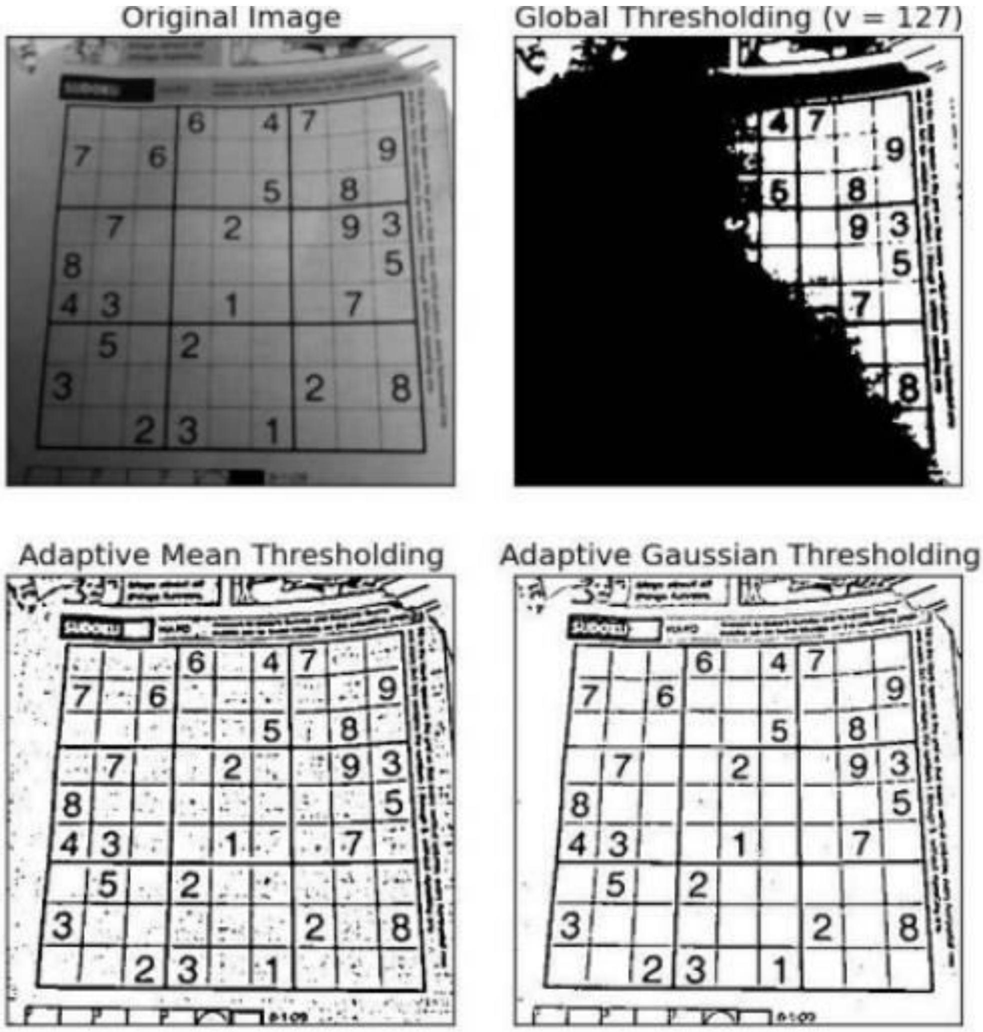
2. Apply pre-processing steps.

These steps include converting to grayscale, applying gaussian filter, and binarizing image via adaptive thresholding.

a. Convert to Grayscale and Resize — Subsequent steps don't rely on color and assuming a consistent size made some of the later processing more simple.



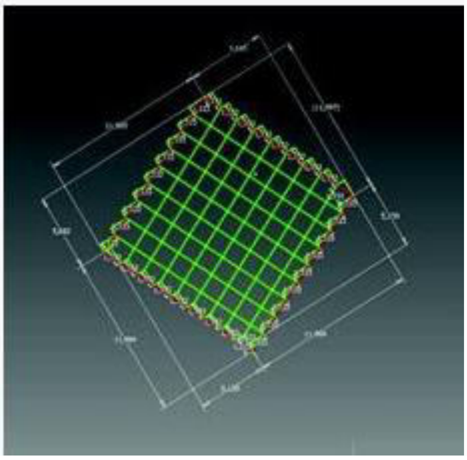
Thresholding is a process used to transform the grayscale image, so that only two colours exist in the image, black and white. The process is commonly used when recognising Sudoku as it is an effective tool to distinguish an object from the background.


Original Image
Global Thresholding (v = 127)
Adaptive Mean Thresholding
Adaptive Gaussian Thresholding

thresholding

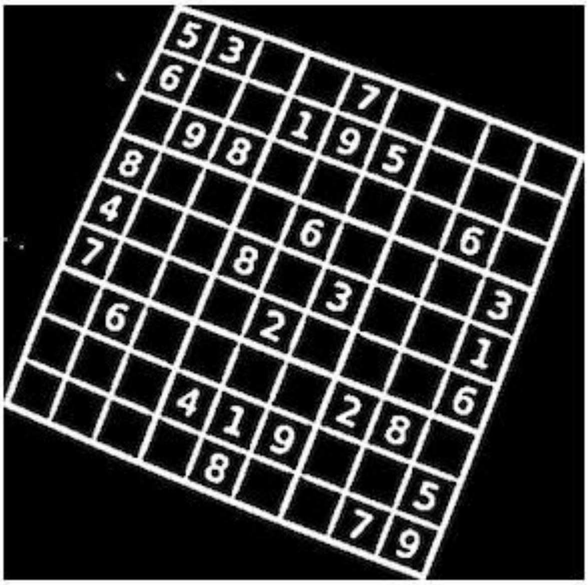
b. Drawing a Grid

We draw the grid for two reasons: identify the correct location of the boxes to determine whether has or not number, if has number the location, it is stored and if it is empty a next step allows us to put a zero for the solution Sudoku mathematical algorithm and the second reason is because allow give an augmented reality. The corners of the rectangle are given by the algorithm, we divide each side into 9 parts, we generate x and y positions of each point of division and after the division, each point is attached to its corresponding position on the side with lines.



c. Extract The largest Blob.

In this step, we extract the largest Blob (Rectangle) assuming that it is the sudoku grid puzzle.

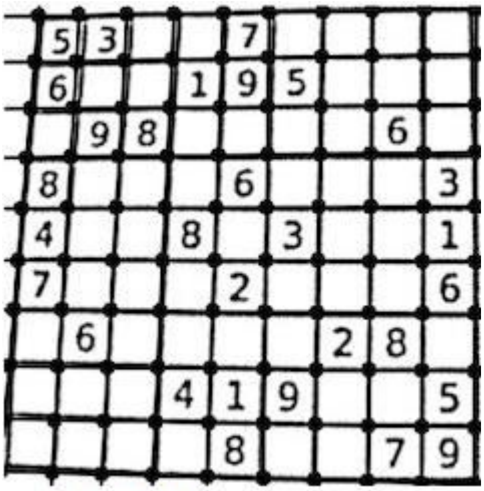


5	3			7				
6			1	9	5			
	9	8					6	
8				6				3
4			8		3			1
7				2				6
	6					2	8	
			4	1	9			5
				8			7	9

Largest rectangle (Rotated) contains sudoku puzzle.

d. Warp perspective and correct skew angle.

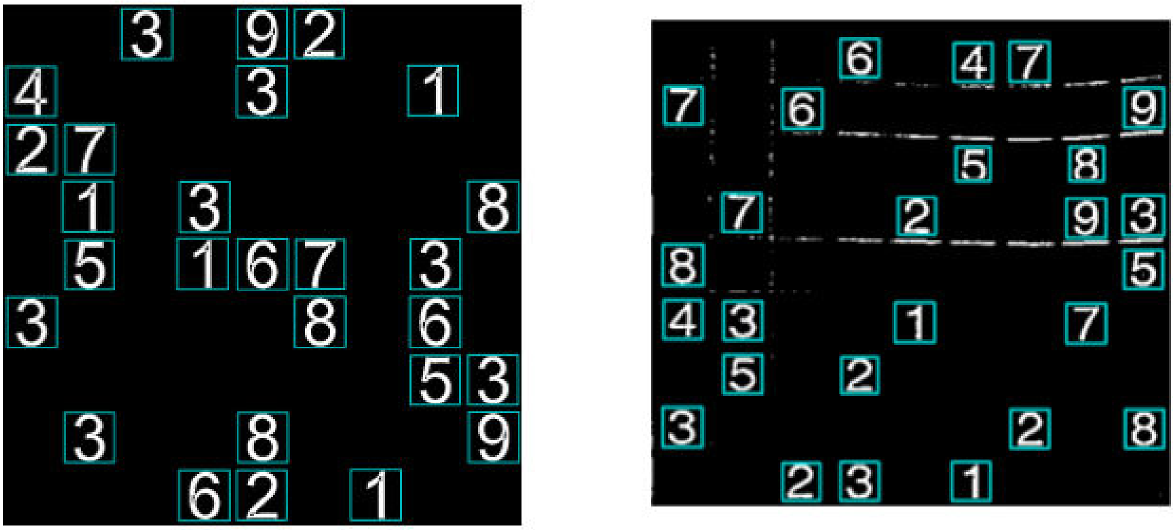
This step includes applying warp perspective using the four corners of the rectangle.



After warping

e. Finding Digits

Finally it's time to detect the digit regions of the image. Once again findContours() is used followed by another aspect-ratio filter. This filter allows us to skip over contours for some of the horizontal artifacts left in the newspaper image on the right.



Detecting Digit Regions of the Image

f. Solve Sudoku puzzle.

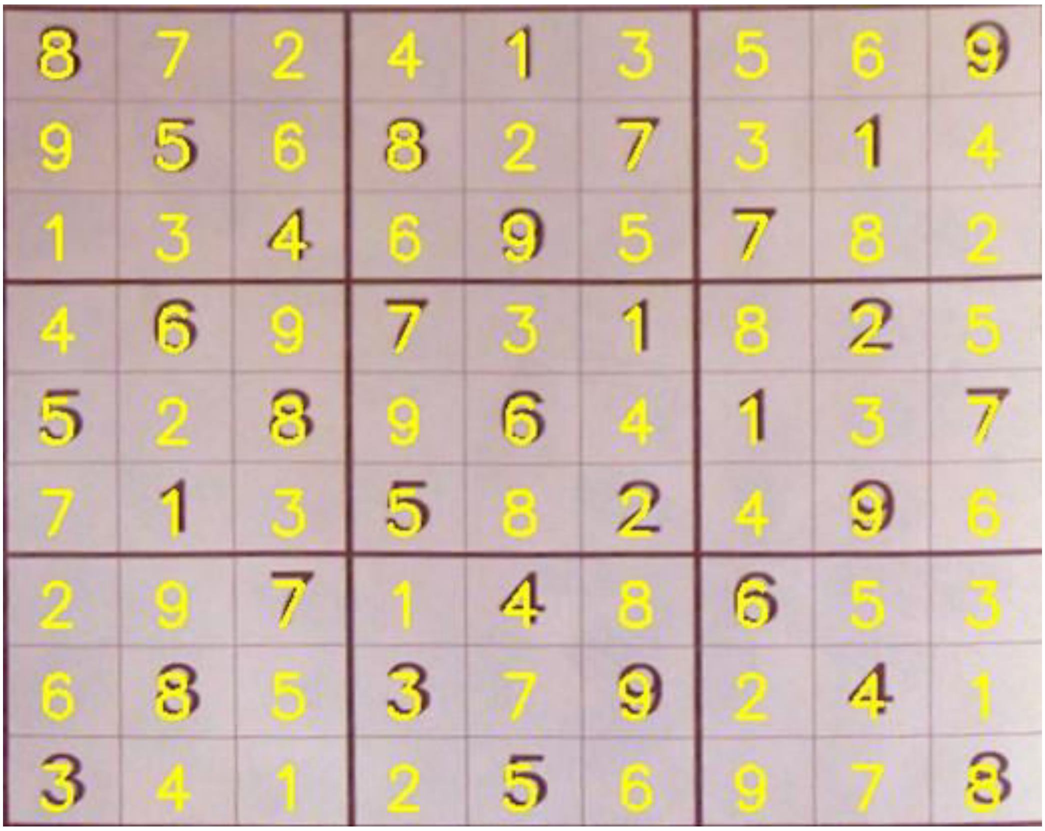
In this step, we form the sudoku matrix from the recognized digits, for the empty cells we just put zero, after that, we send it to SVM.

We have used the in built SVM model of open-cv with a linear kernel & rained the model for 1000 epocs. The model was trained on a labelled training data which included images of digits from 0 to 9 & were labelled by hand. Training accuracy obtained was 100%.

g. Print the result.

In this step, we print the result to the image, the blue digits are the solution, and the red digits are the recognized digits of the sudoku puzzle

.



8	7	2	4	1	3	5	6	9
9	5	6	8	2	7	3	1	4
1	3	4	6	9	5	7	8	2
4	6	9	7	3	1	8	2	5
5	2	8	9	6	4	1	3	7
7	1	3	5	8	2	4	9	6
2	9	7	1	4	8	6	5	3
6	8	5	3	7	9	2	4	1
3	4	1	2	5	6	9	7	8

3. TECHNICAL ANALYSIS

3.1 UML Diagrams

The Unified Modeling Language (UML) is a general-purpose, developmental, modeling language in the field of software engineering that is intended to provide a standard way to visualize the design of a system. UML diagrams helps us to understand the flow of application, how the actual mechanism works and what type of functions are available in the project.

UML diagrams are indispensable with regards to the Software Development Life Cycle and hence they are part of each Developer's life.

UML diagram is basically of two types:

- **Structural Diagrams:** Structure diagrams show the things in the modeled system. In a more technical term, they show different objects in a system. Eg: class based diagrams

- **Behaviour Diagrams:** Behavioral diagrams show what should happen in a system. They describe how the objects interact with each other to create a functioning system. Eg: use case diagrams and sequence diagrams.

In our project, we don't have class based implementations rather we have some functions that help in achieving the end product. So we have made Use case diagrams and Sequence diagrams.

Use Case Diagram

As the most known diagram type of the behavioral UML types, Use case diagrams give a graphic overview of the actors involved in a system, different functions needed by those actors and how these different functions interact.

In our project, we have two actors (user, programs) and between them only our whole application runs. There is only one major function involved in our program i.e. solving Sudoku by presenting a picture of unsolved Sudoku to the camera of the device.

Functions included in our program are as follows:

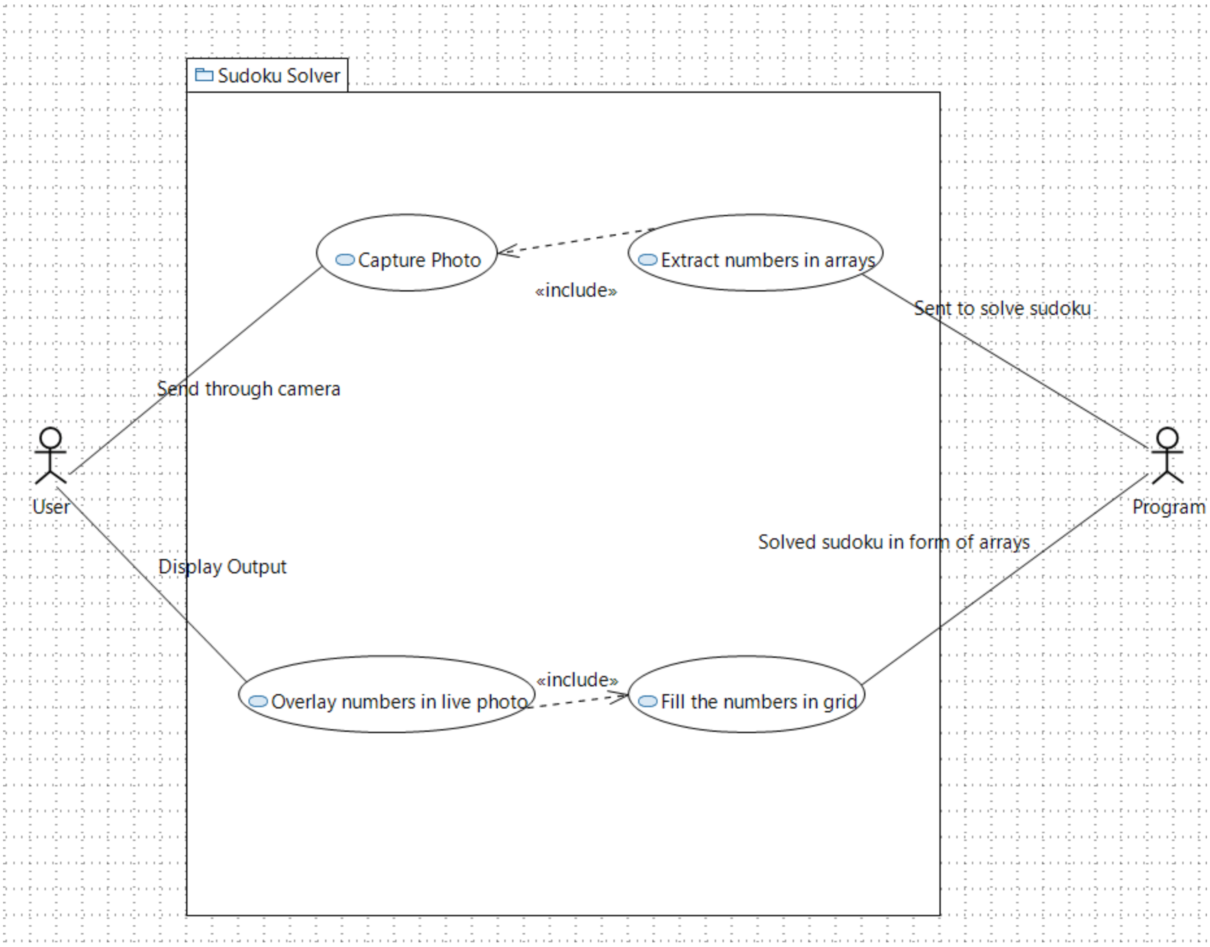
a. Device's camera captures photo

b. Numbers are extracted from the photo captured.

c. Numbers are sent in form of arrays to the program to solve Sudoku.

d. Program sends solved Sudoku in form of arrays

e. arrays are converted into grid and missing numbers are overlayed on the unsolved Sudoku grid.



Sequence Diagram

Sequence diagrams in UML show how objects interact with each other and the order those interactions occur. It's important to note that they show the interactions for a particular scenario. The processes are represented vertically and interactions are shown as arrows.

For our case we have the following sequence diagram:

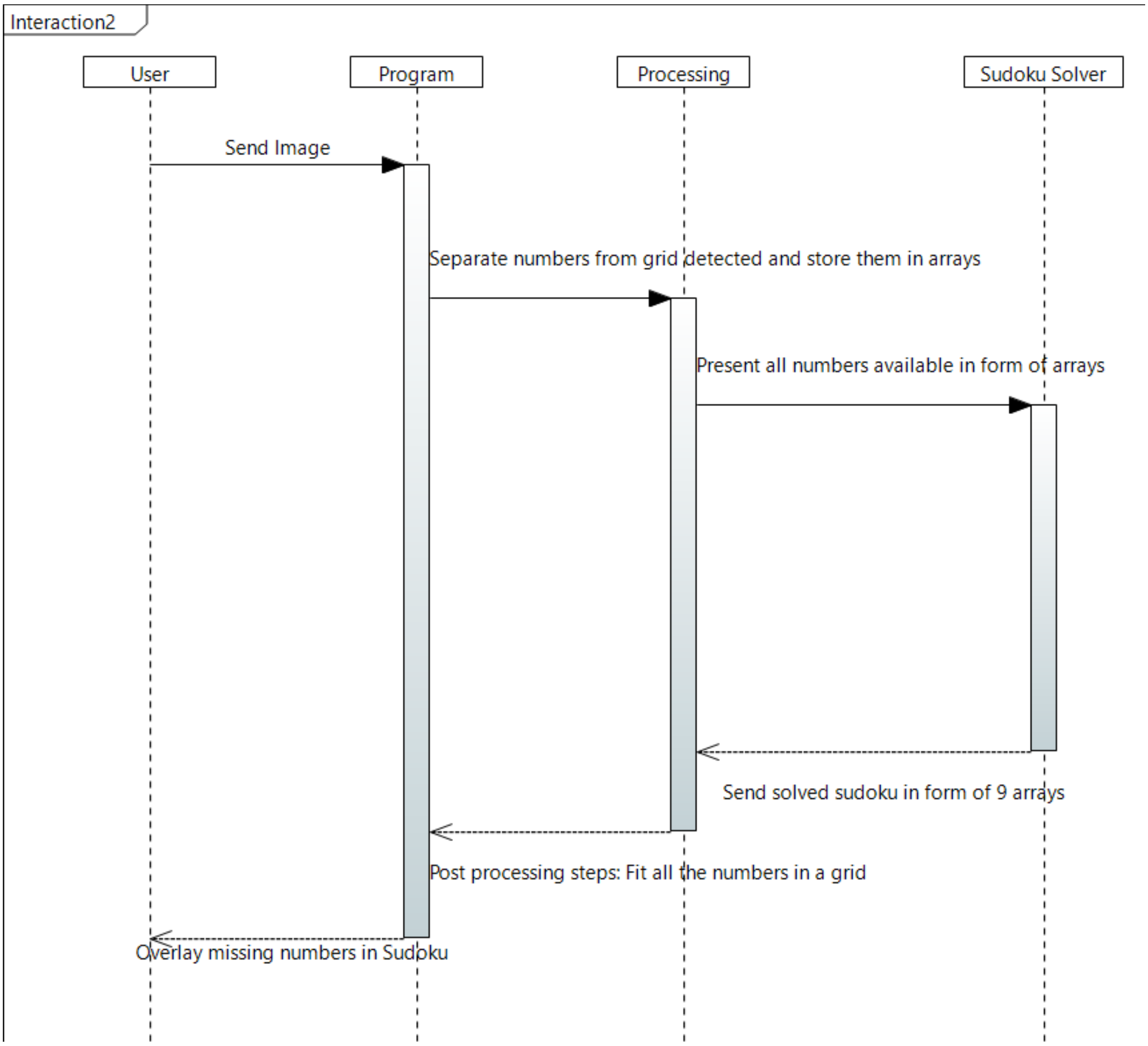
a. First the user interacts with the main program by sending the image of the unsolved Sudoku.

b. Program processes the image by identifying the contours of the grid. Then the numbers identified are separated from the grid and stored in form of arrays. Now these arrays are passed to Sudoku solver program.

c. Sudoku Solver program then solve the Sudoku using backtracking algorithm and send the solution back to processing unit in form of arrays.

d. Now the arrays are converted into a grid 9*9. This grid contains the whole solution of the given Sudoku problem.

e. Finally the missing numbers are overlayed on the live image of unsolved Sudoku.


Interaction2
User
Program
Processing
Sudoku Solver
Send Image
Separate numbers from grid detected and store them in arrays
Present all numbers available in form of arrays
Send solved sudoku in form of 9 arrays
Post processing steps: Fit all the numbers in a grid
Overlay missing numbers in Sudoku

3.2 TECH-STACK Analysis

In order to achieve proper solution, we have used a variety of Tech Stacks. All these technologies have been chosen on the basis of the following few criteria:-

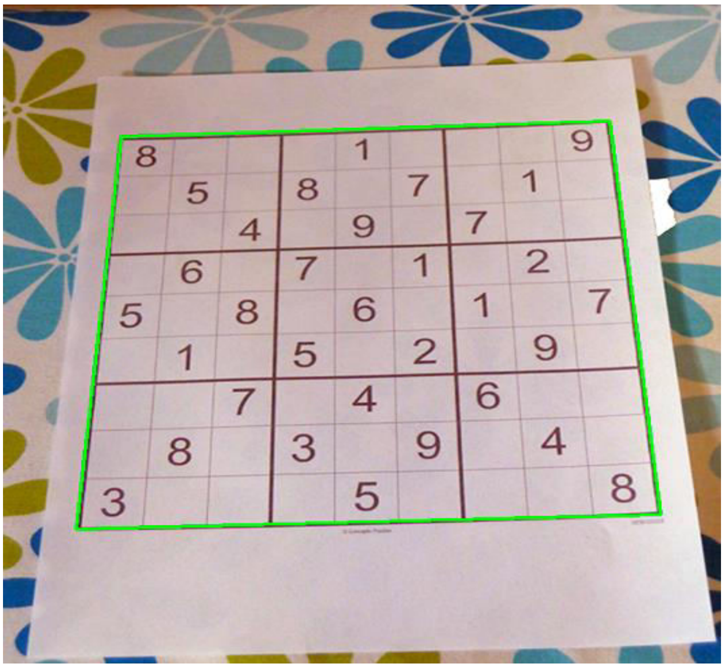
1. Ease of Usage and Ease of Learning
2. Efficiency
3. Accuracy

A. OpenCV

OpenCV 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++, but it still retains a less comprehensive though extensive older C interface. All of the new developments and algorithms appear in the C++ interface.
There are bindings in Python, Java and MATLAB/OCTAVE for using this library.

Use in our project:

OpenCV helps to capture image of unsolved Sudoku from user's device.
When a user starts a program he/she positions the Sudoku in front of camera in such a way that program is able to capture the grid of Sudoku.

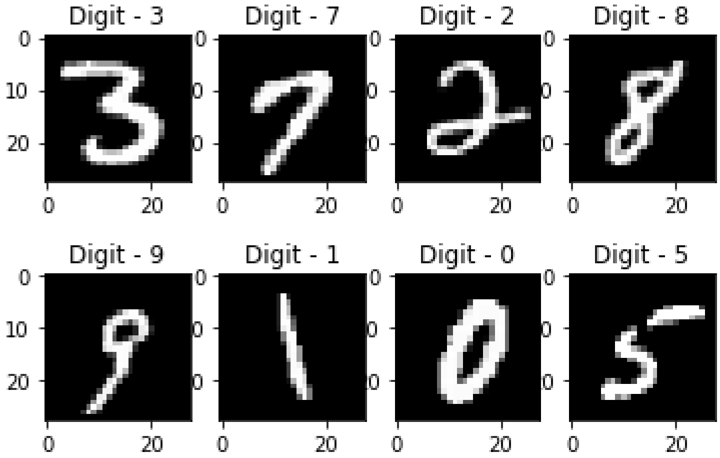


B. Support Vector Machine (SVM)

In machine learning, support-vector machines are supervised learning models with associated learning algorithms that analyze data for classification and regression analysis. After giving a SVM model sets of labeled training data for each category, they're able to categorize new text.

Use in our project:

Main use of SVM in our project is to identify the digits in a Sudoku grid. After the grid is identified the numbers are classified according to the model we trained on SVM. These identified numbers are then sent to Sudoku solver program for final output.



C. Heroku Deployment

Heroku is a cloud platform as a service supporting several programming languages. Heroku is a platform where we can deploy our application freely on the internet. It involves all the facilities like cloud database and assigning domain names to the applications.

Use in our Project:

Heroku will be used to deploy our project on the internet so that any user can use it remotely on his/ her device.

4. RESULTS AND DISCUSSION

4.1 Usage Instructions

This python program is very easy to use. User has to follow these set of instructions:

a. Position the camera in such a way that the grid of Sudoku comes exactly in the visible range of camera.

b. He/she should avoid bringing anything else closer to the camera so that camera can focus on the grid only.

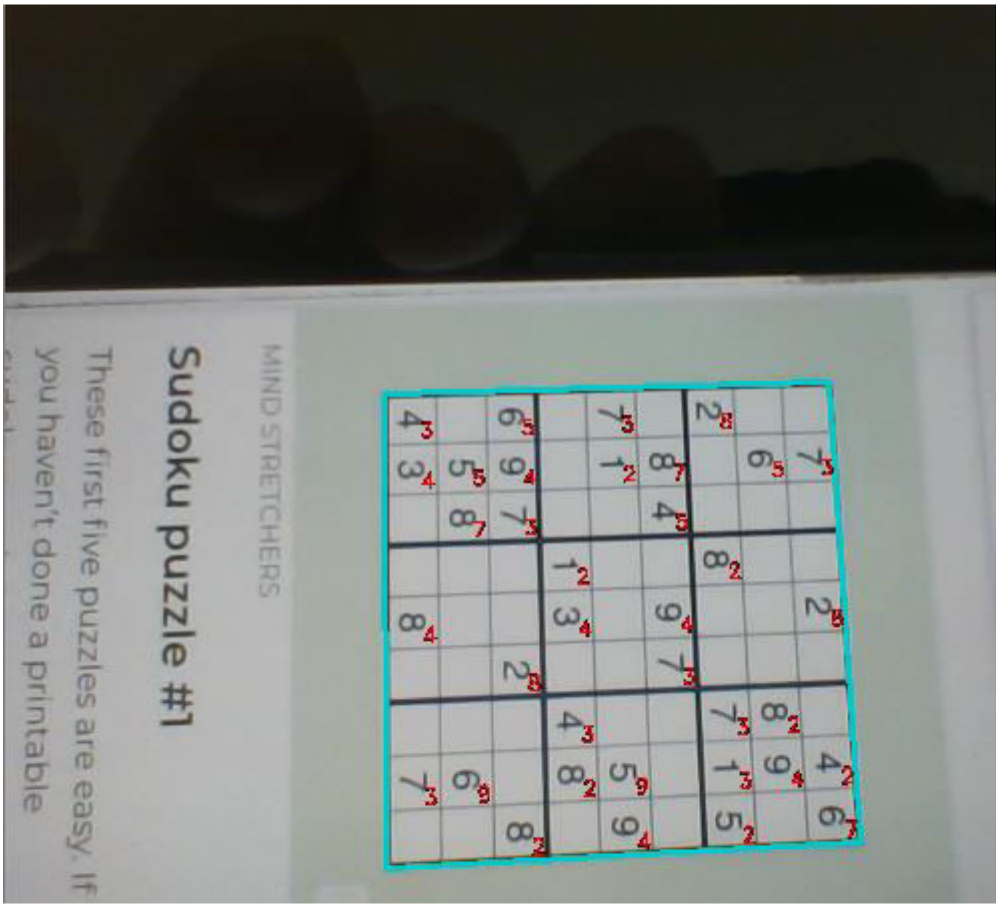
c. He/she should check whether the device camera is clean, if not then he should clean the camera first so that image can be clearly captured.

d. User should keep the unsolved Sudoku still in front of camera to avoid blurry image.

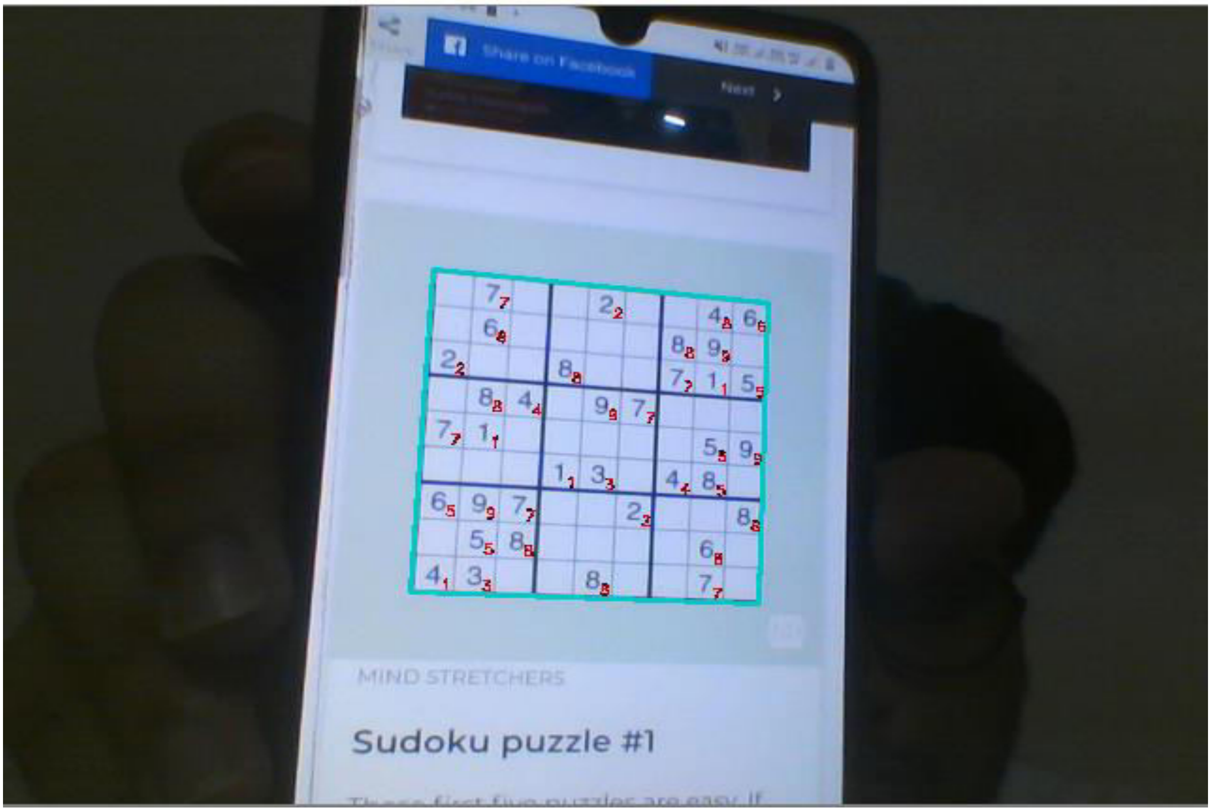
e. User must keep Sudoku properly aligned in front of camera.

f. User must keep the hand holding Sudoku steady till the missing numbers have been overlayed.

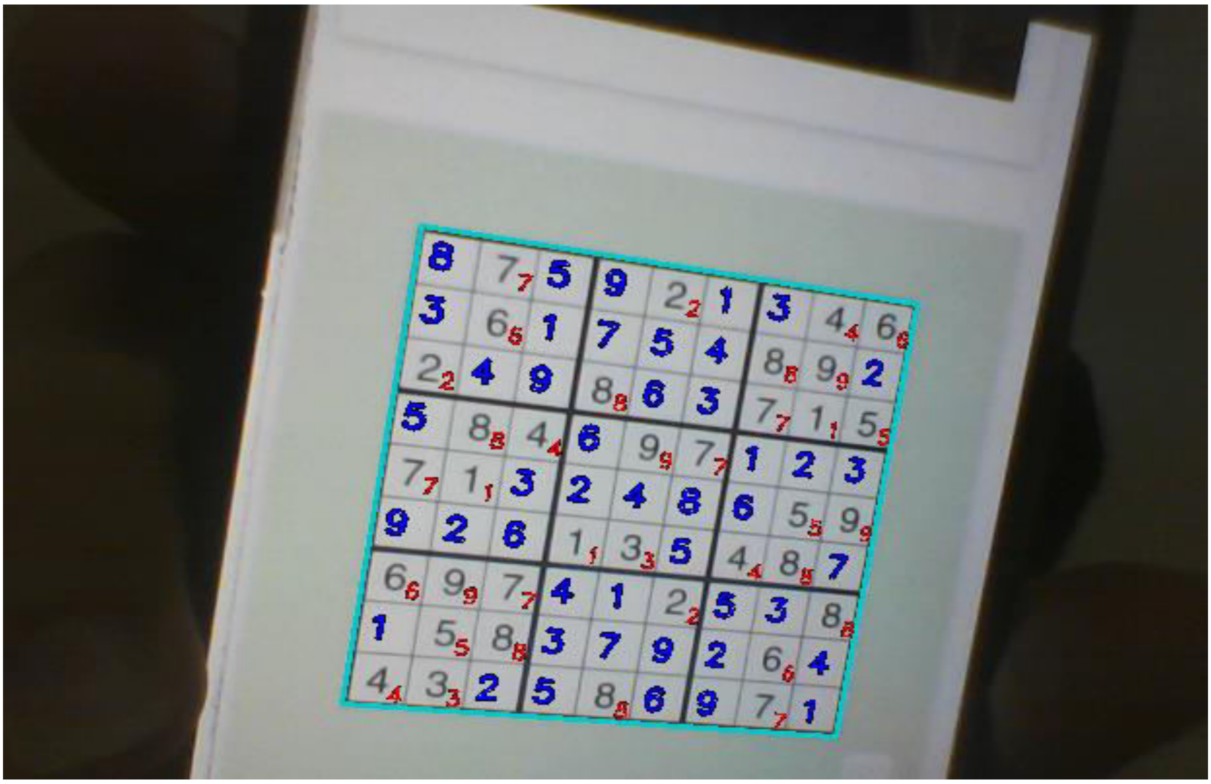
Wrong way:


MIND STRETCHERS
Sudoku puzzle #1
These first five puzzles are easy. If
you haven't done a printable

Right way:



Result:



4.2 Deployment status and testing phase

This program will be deployed on internet using heroku application in the coming future. Testing will also be done after deployment.

5. CONCLUSION

The role of computers in puzzle world is now becoming more and more important because computers changed not only puzzle creators, but also puzzle solvers. Almost all puzzles are exclusively for recreation.

Some of the limitations are-

• The solver is by no means fullproof, it still has trouble with some images and either will fail to parse them or parse them incorrectly leading to failure to solve them (because it may have parsed as an invalid puzzle).

• The execution time of the algorithm is influenced by the stage of numerical resolution, which in turn depends on how well they have been read the numbers present in the Sudoku, if all stages are well the solving is shown in real time

• It also uses backtracking algorithm that runs in exponential time complexity.
• Characters misread also generate errors in the solution of Sudoku.

The goal of course was to ramp up on some new technologies and the project has been valuable from that perspective.

This project is further used for other Augmented Reality related operations which can have a major impact on society.