Solving Sudoku using Object Character Recognition

CS 6384 – COMPUTER VISION

PROJECT GROUP NO. 16

BY ADITI KOTABAGI, ADITYA VISWANATHAM, GAURAV MADNANI

What is Sudoku?

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

(a) Sudoku Puzzle

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

(b) Solution

Project is divided into three major steps:

1. Image Processing of the Sudoku puzzle to extract digits from the image.

2. Digit Classification of the extracted digits using Convolutional Neural Networks.

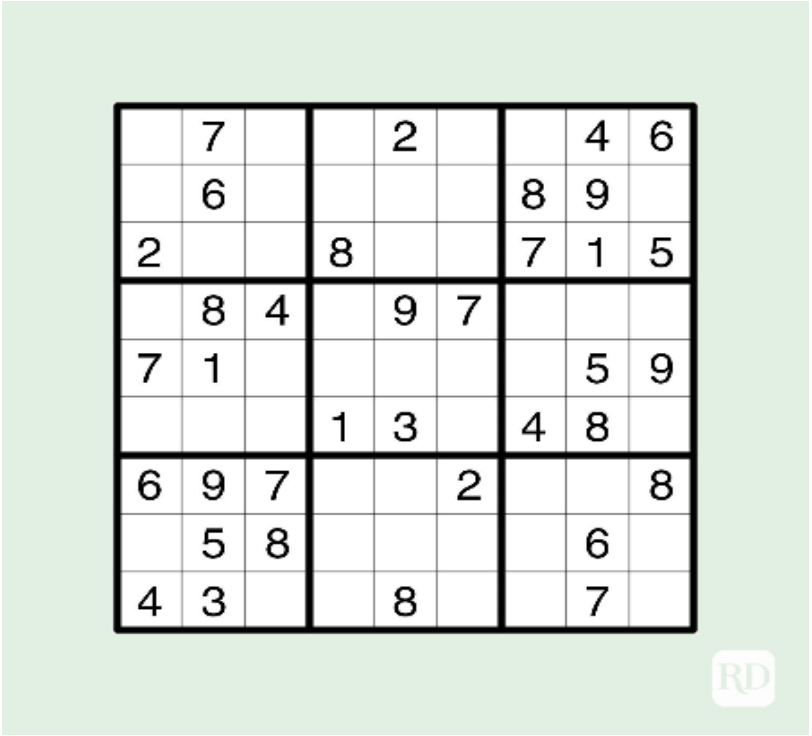
3. Solving the Sudoku puzzle using Backtracking Algorithm.

Image Processing of an image

1. Preprocessing
2. Finding all contours
3. Finding the biggest contour
4. Extracting numbers of a particular grid

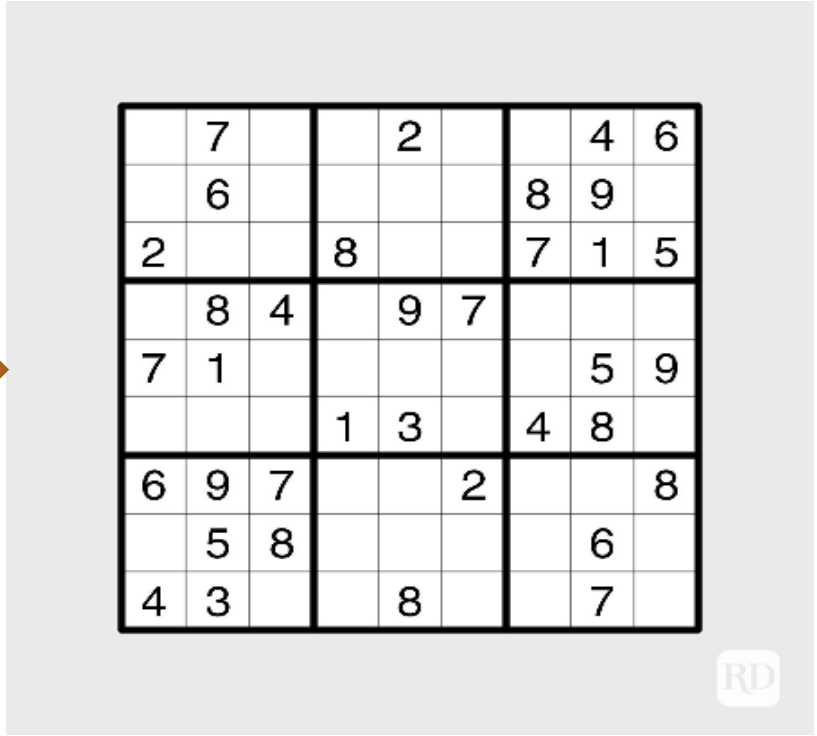
Preprocessing

Original Image

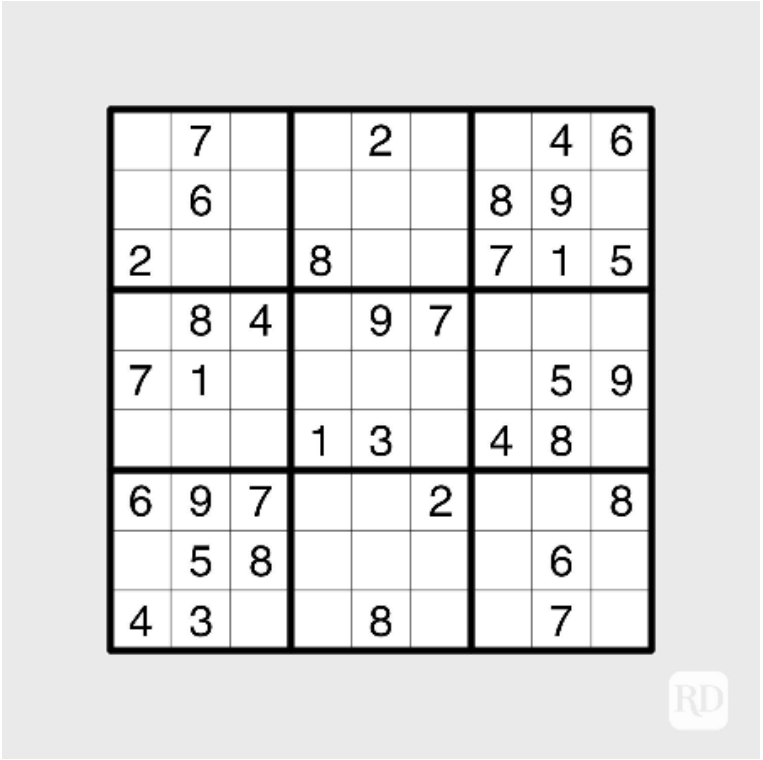


Grayscale the image

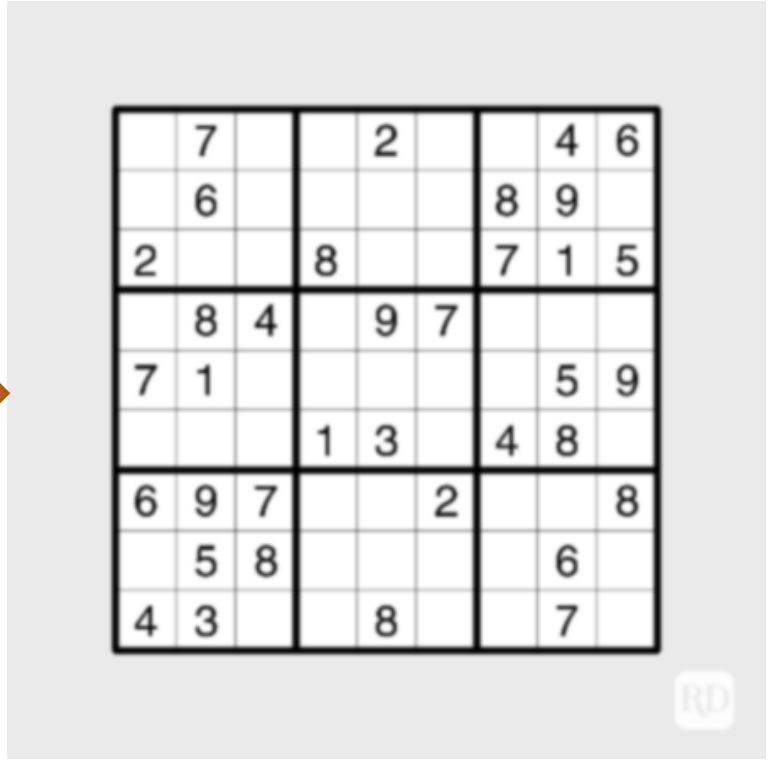
Gray scaled Image



Preprocessing



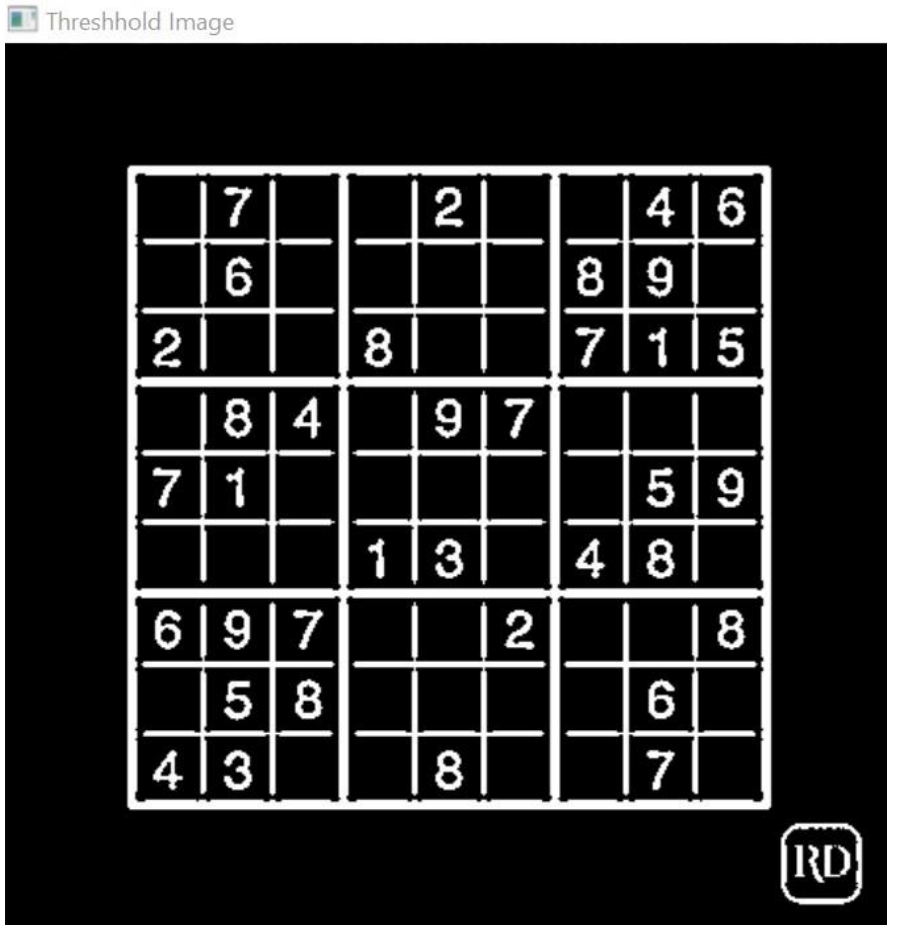
Adding Gaussian Blur



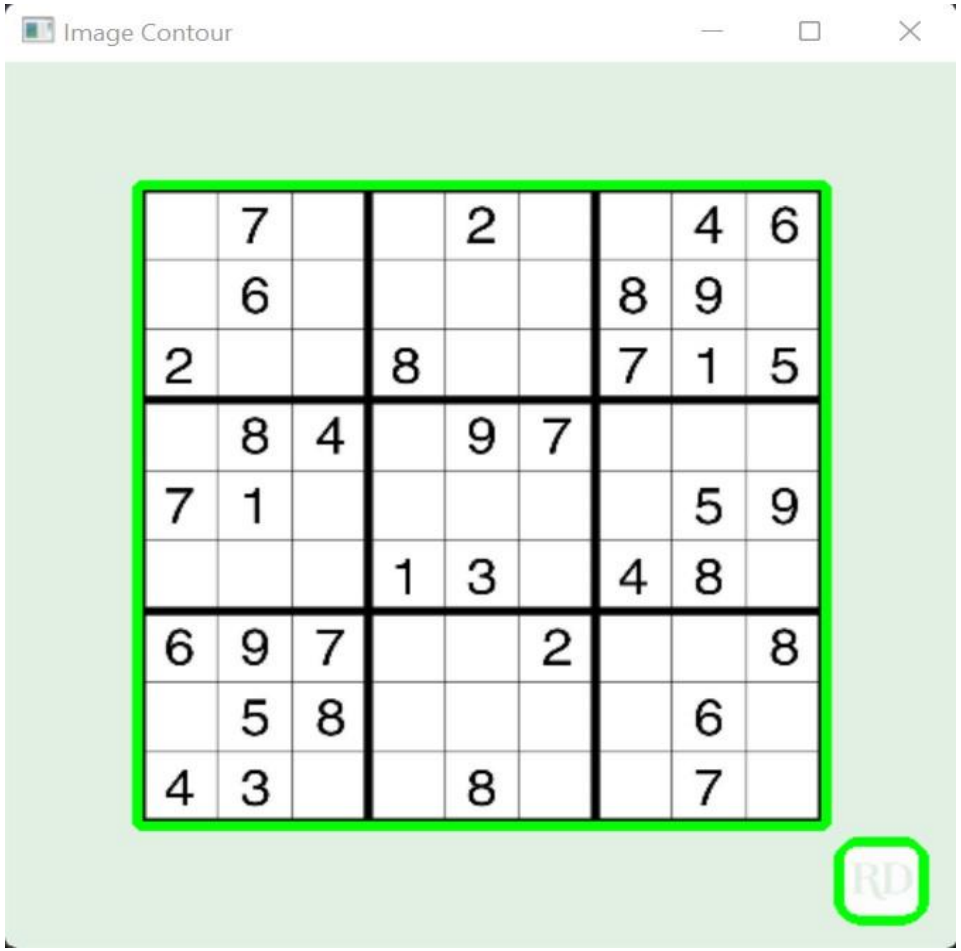
Preprocessing

Thresholding:

The threshold used is the mean value of the image intensity values.

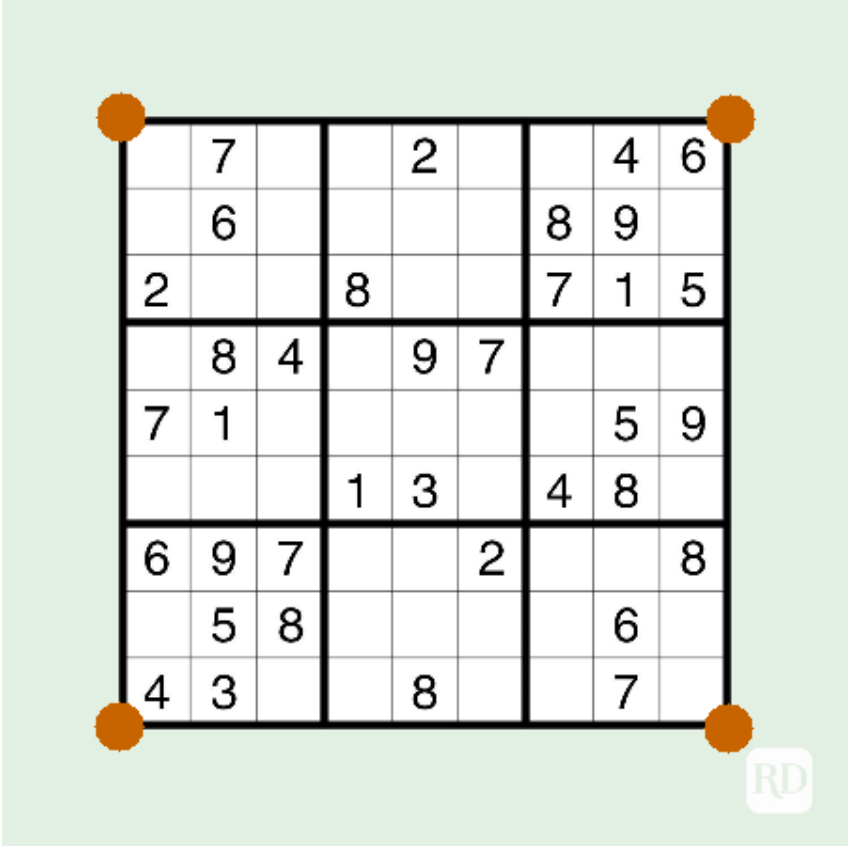


Finding all the contours



Finding biggest contour

Finding the corners of the contours that have maximum area.



Extracting numbers of a grid

The four corners are used to warp the image.

The image now has 9x9 grid that is 81 grids.

The second index out of the 81 indices will be:

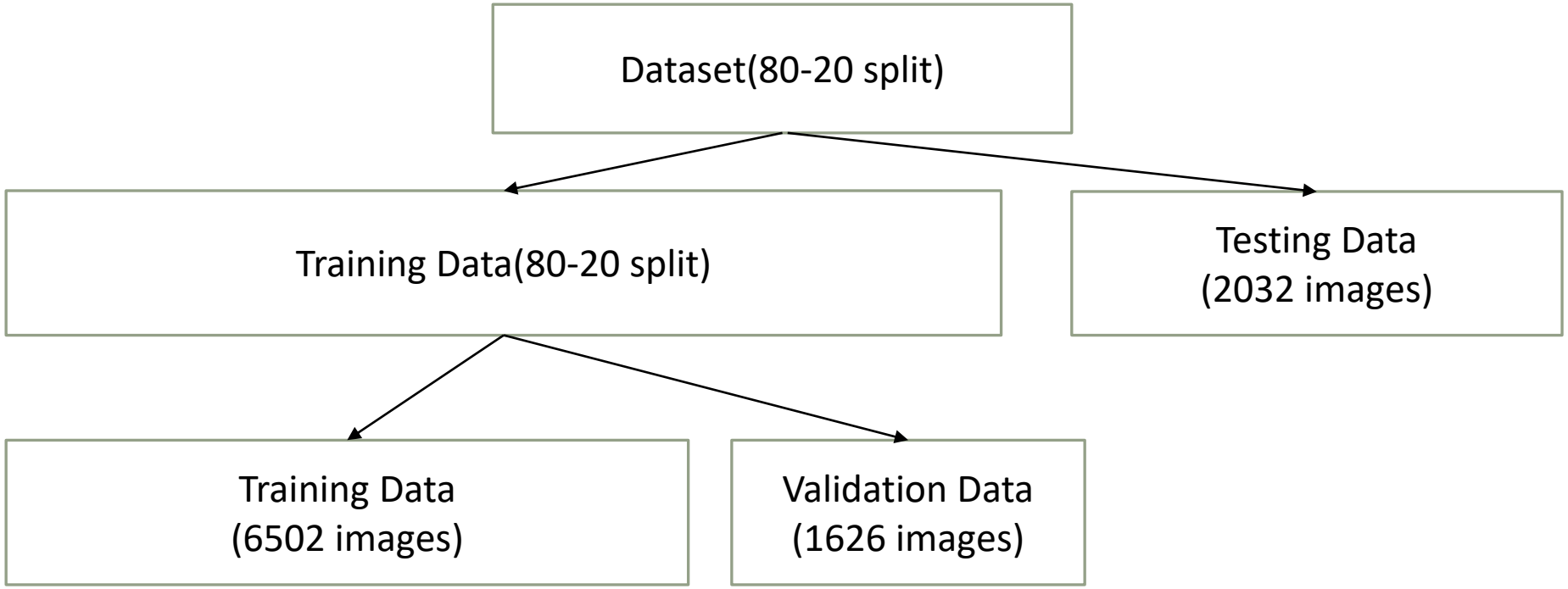
	7			2			4	6
	6					8	9	
2			8			7	1	5
	8	4		9	7			
7	1						5	9
			1	3		4	8	
6	9	7			2			8
	5	8					6	
4	3			8			7	

Digit Classification

1. Importing and Splitting data
2. Preprocess and Image Augmentation
3. CNN Model Definition
4. Accuracy and Loss Graph

Importing and splitting Data

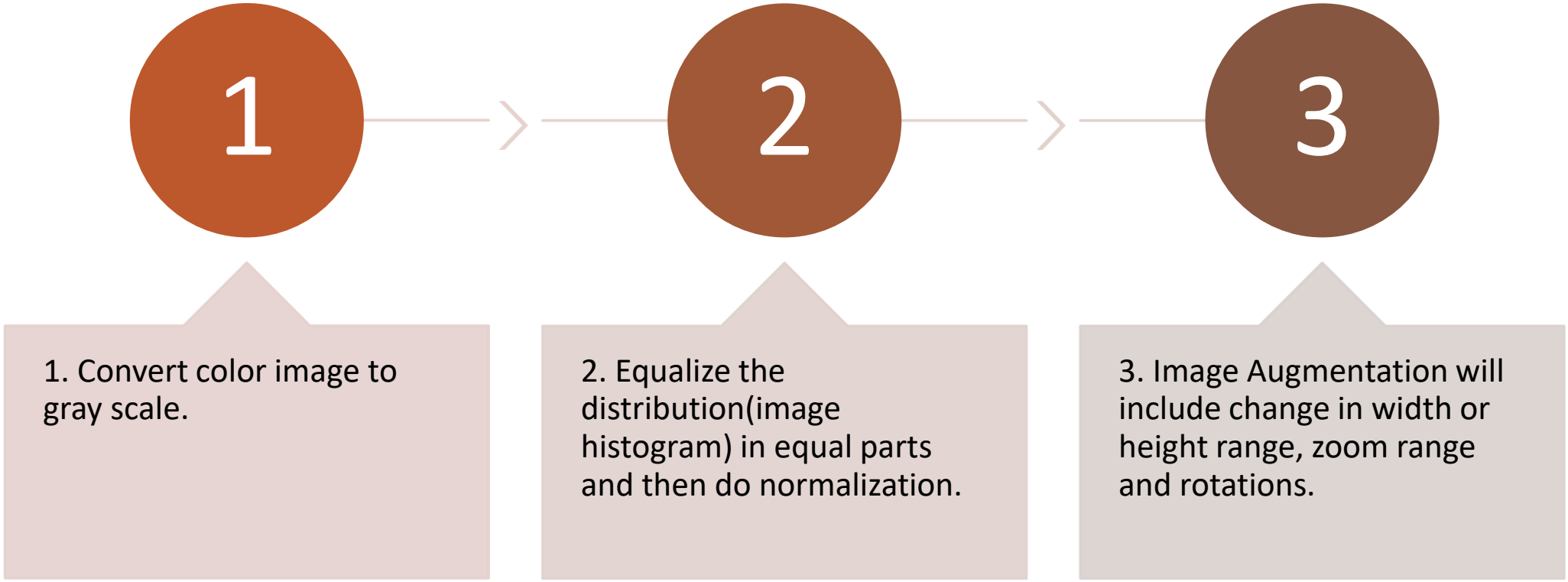
The data has 10 classes from 0 to 9, total of 10160 images



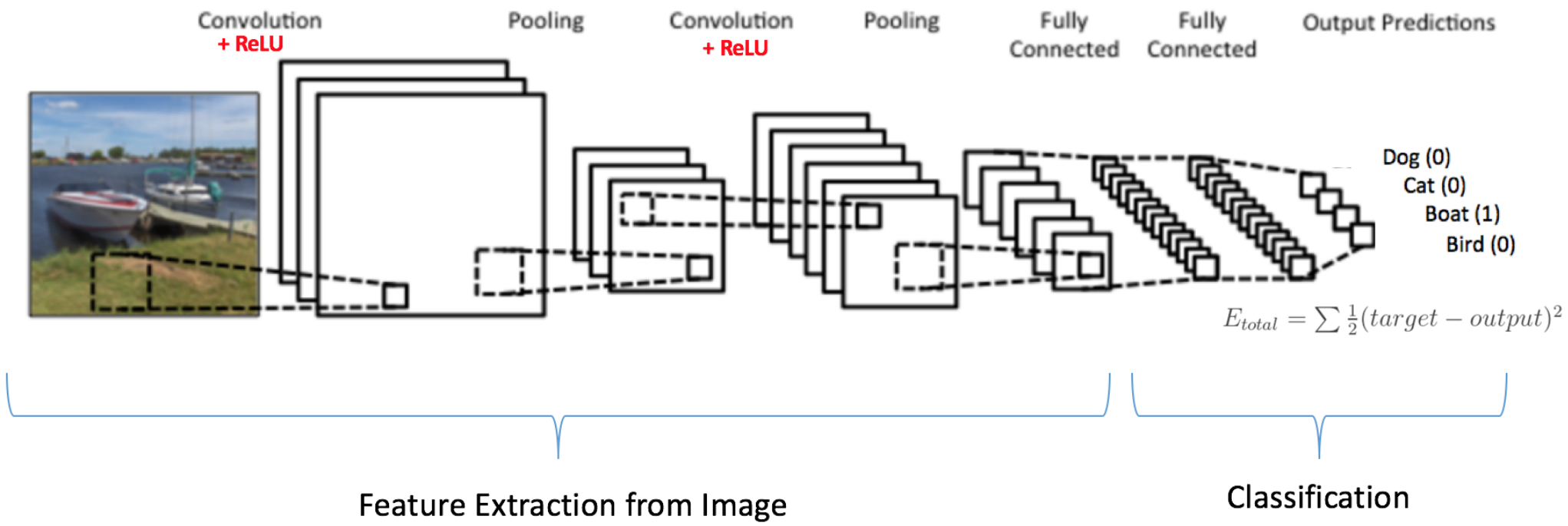
Importing and Splitting Data

```
Number of Images 10160
BeforeSplit train shape (10160, 32, 32, 3)
BeforeSplit label shape (10160,)
After Splitting train (6502, 32, 32, 3) (6502,)
After Splitting validation (1626, 32, 32, 3) (1626,)
After Splitting test (2032, 32, 32, 3) (2032,)
Number of 0 classes in training: 676
Number of 1 classes in training: 643
Number of 2 classes in training: 652
Number of 3 classes in training: 635
Number of 4 classes in training: 673
Number of 5 classes in training: 638
Number of 6 classes in training: 646
Number of 7 classes in training: 655
Number of 8 classes in training: 637
Number of 9 classes in training: 647
```

Preprocess and Image Augmentation



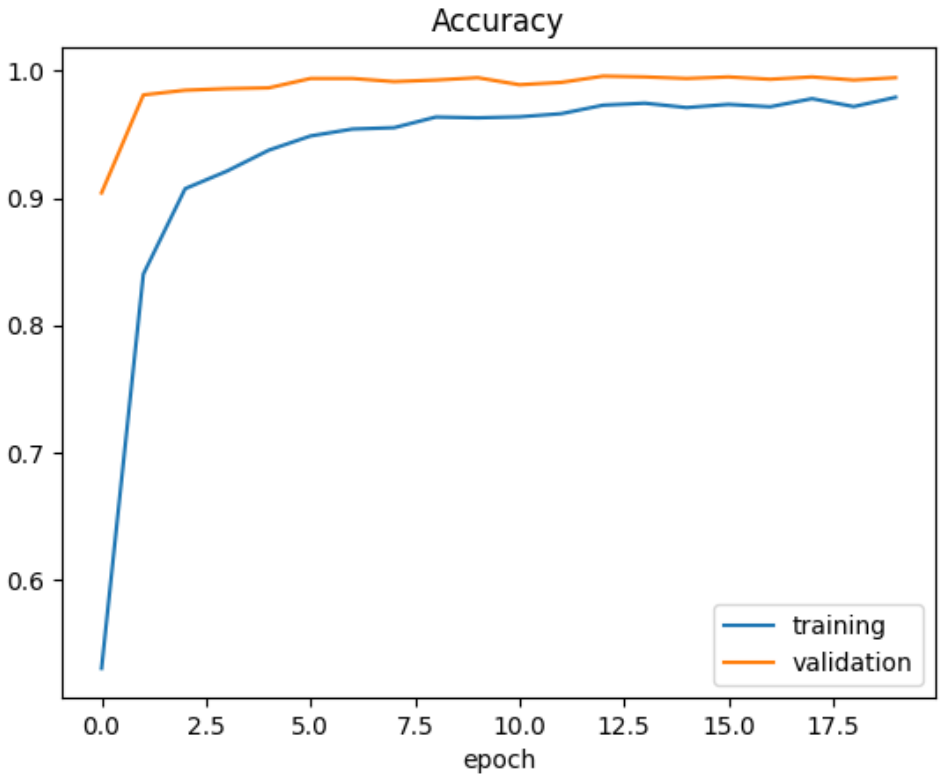
Convolutional neural network

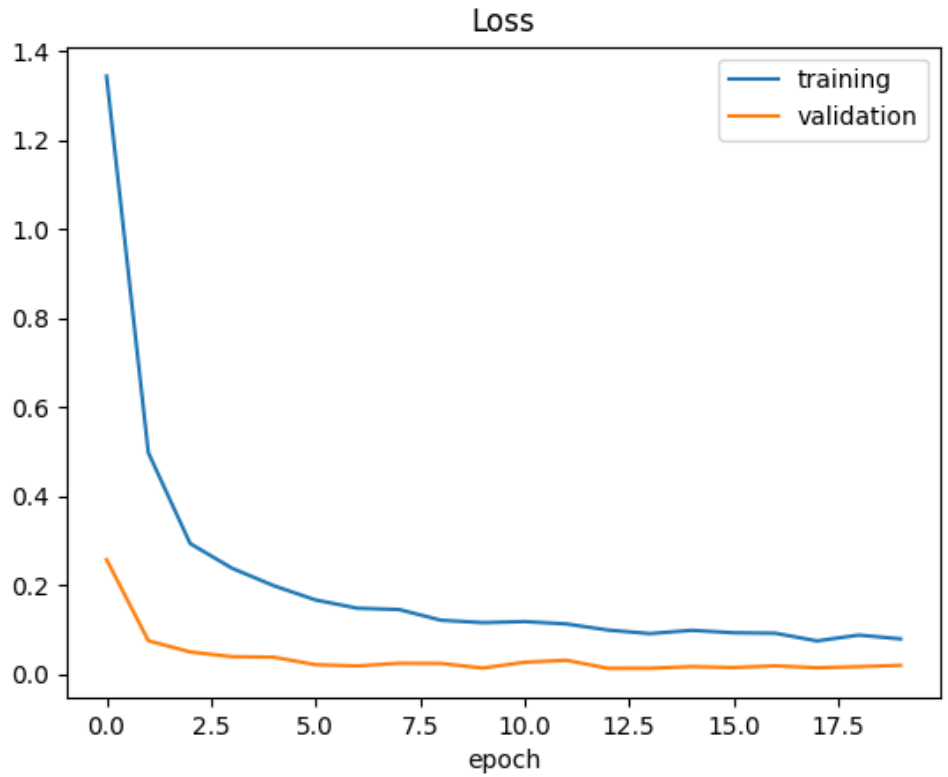


Our CNN Model

```
def mymodel():
    sizeofFilter1=(5,5)
    sizeofFilter2=(3,3)
    poolsize=(2,2)
    numberofnode=500
    numberoffilters=60
    model=Sequential()
    model.add((Conv2D(numberoffilters,sizeofFilter1,input_shape=(32,32,1),activation='relu')))
    model.add((Conv2D(numberoffilters,sizeofFilter1,activation='relu')))
    model.add(MaxPooling2D(pool_size=poolsize))
    model.add((Conv2D(numberoffilters//2, sizeofFilter2, activation='relu')))
    model.add((Conv2D(numberoffilters//2, sizeofFilter2, activation='relu')))
    model.add(MaxPooling2D(pool_size=poolsize))
    model.add(Dropout(0.5))
    model.add(Flatten())
    model.add(Dense(numberofnode,activation='relu'))
    model.add(Dropout(0.5))
    model.add(Dense(numberofclasses,activation='softmax'))
    model.compile(Adam(lr=0.001),loss='categorical_crossentropy',metrics=['accuracy'])
    return model
```

Accuracy and Loss





Solving Sudoku using backtracking algorithm

Algorithm

```
function solveSudoku(board):
  for i in range(9):
    for j in range(9):
      if board[i][j] equals 0:
        for num <- 1 to 9:
          if validNumber(board, num, i, j):
            board[i][j] <- num
            if solveSudoku(board)
              return True
            else
              board[i][j] <- 0
        return False
  return True
```

References

1. https://medium.com/analytics-vidhya/introduction-to-image-processing-part-10-sudoku-solver-4530f6c85d6c

2. https://www.analyticsvidhya.com/blog/2021/07/classification-of-handwritten-digits-using-cnn/

3. https://dev.to/lwolczynski/how-to-solve-sudoku-faster-than-with-backtracking-1nbk