**CHAPTER 2**

**SYSTEM DESIGN**

**System Flowchart**

The data is first collected thorough applying threshold and converted as black and white image. Then we train the data using convolutional neural networks. Finally we predict the gestures using the trained data. Steps involved in design and planning of system are -

**Research on algorithm**

CNN was the most suitable algorithm for this project as it is used for object classification.

**Set environment**

Anaconda jupyter notebook was set up and and a separate environment other than base environment was created.

**Install and import required libraries**

Installation of required libraries like keras, CNN and OpenCV. Then they were imported in the project.

**Collect Data**

User defined dataset was collected for each gesture and stored in the dataset folder.

**Splitting into train and test data**

Dataset was split into test and train folder . 750 images was stored in the train folder and 250 in the test folder for each gesture.

**Pre-Processing of data**

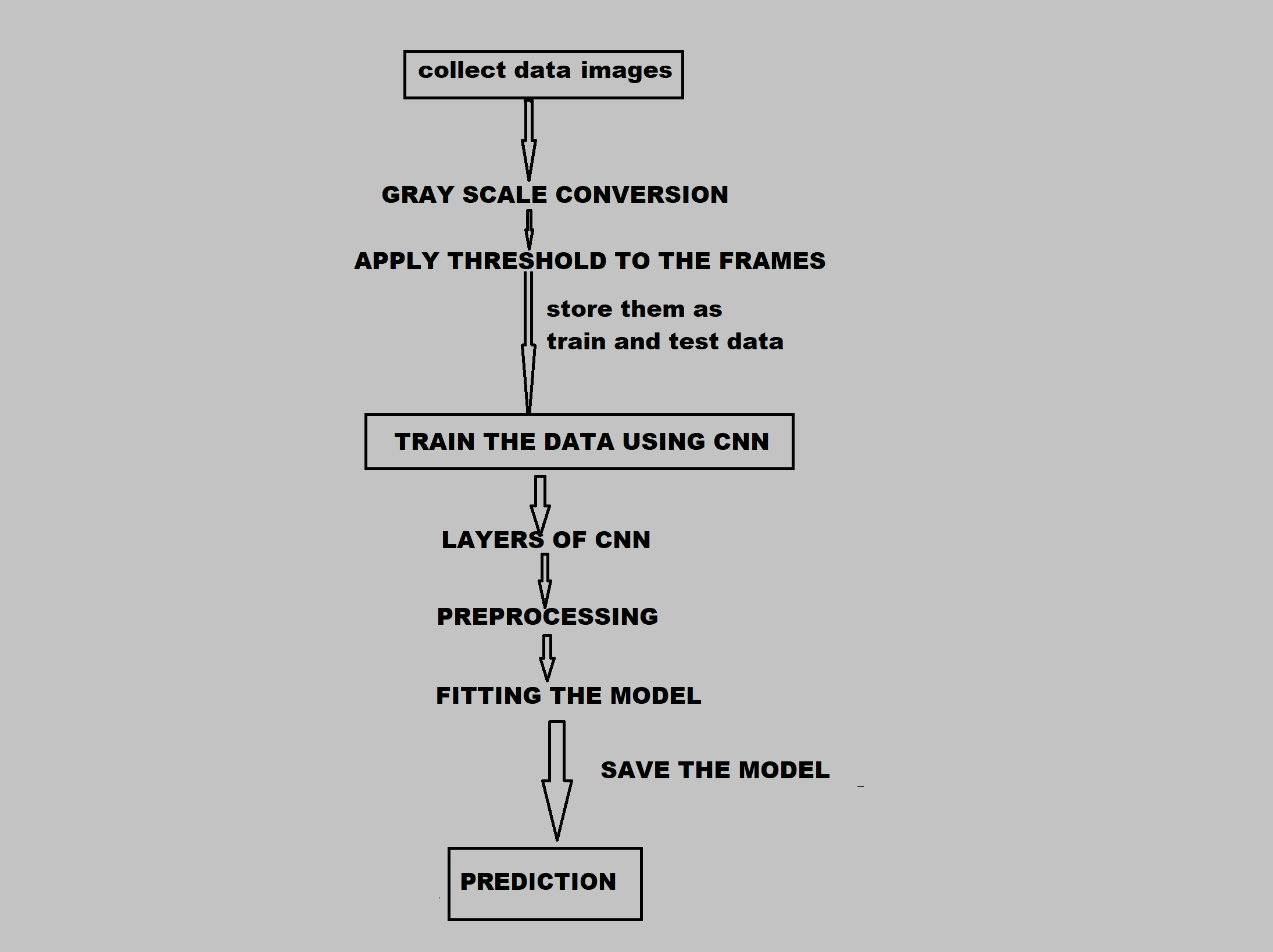
Pre-processing is required for better accuracy. Resizing of data, gray-scale conversion and threshold was applied to assign pixel values.

**Train the model**

Then we created a sequential model and trained our data using CNN. Convolution and max-pooling layer were applied twice for better feature extraction.

**Prediction**

Hand gestures will be captured in live video feed and predict the sign languages in real time.

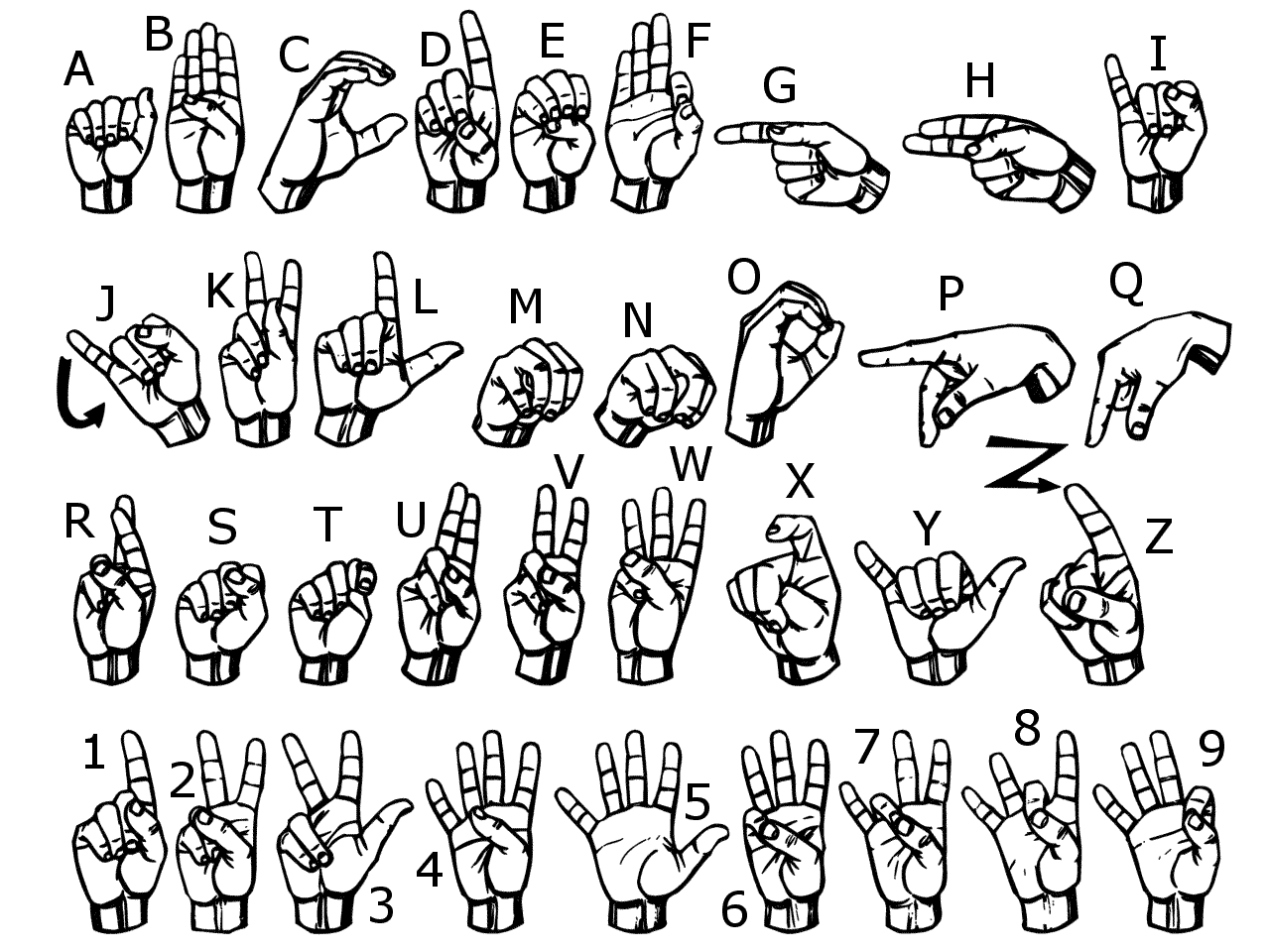
****

**Collecting Dataset**

American Sign Language (ASL) is standard language that serves as the communication means of Deaf communities in the United States. It is a set of 26 hand gestures known as the American manual alphabet, which can be used to spell out words from English.

This is widely popular among deaf and dumb community to communicate using a standard communication language.

We store self created dataset through live video feed and split the data into two parts train and test. The training data for each gesture consist of 750 images and test data contains 250 images.

****

**Training the model using Convolutional Neural Network**

We use CNNs over other NNs as we do not need to flatten the input images to 1D as they are capable of working with image data in 2D. This method helps in retaining the “spatial” properties of images. CNN works best with image like data. If we use ANN then processing of such huge no. of pixels will be computationally expensive.

A Convolutional Neural Network is a Deep Learning algorithm which takes in an input image, assign weights and biases to pixels in the image and extract features from them. This enables it to differentiate one image from the other.

It sequentially builds model layer by layer.

It consists of convolutional layer, max-pooling layer, flattening layer and dense layer.

Activation functions are applied in the layers according to the need of the features.

****