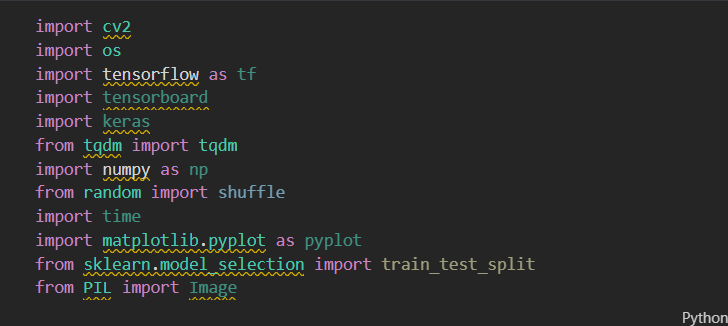
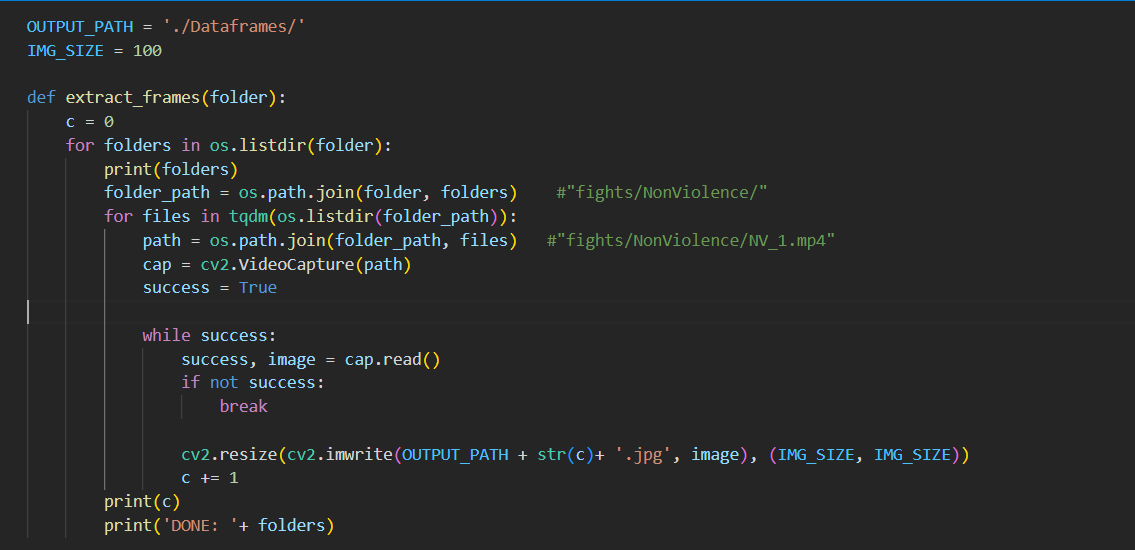
**Presentation**

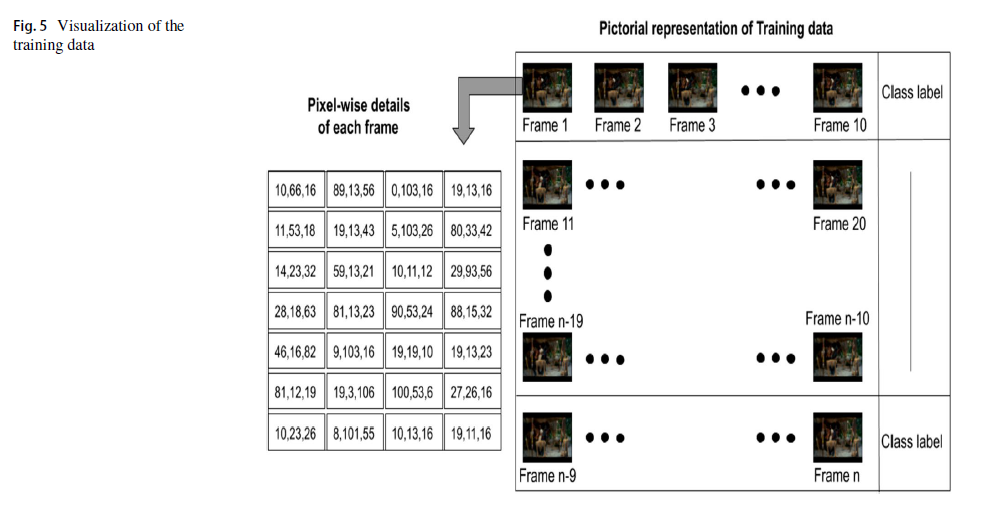
**first importing libraries like ( cv2 , os , tensorflow , …)**

****

**Then we extracted frames from videos and read images and**

**and resize images.**

****

****

**Data Preprocessing**

Frames have been extracted from the videos. The extracted

frames are reshaped to 100 × 100 pixels (denoted as *x* × *y* ).

The training data are a Numpy3 array, with each of its row

representing a sequence or pattern in video. A sequence

might include a degree of movement and actions, whether

a movement of the arm is a punch or a handshake, etc. The

minimum number of frames required to extract a sequence

is 2. However, we have used 10 consecutive frames (denoted

as *n*) to extract the temporal features (that is, time-related

features). The total number of samples (denoted by *N*) is

the number of such sequences present in the dataset (( total

number of frames )/(number of frames to be considered in a

sequence)). For a simple implementation, numpy allows an

arbitrary value of −1 to be used. Hence, a structure containing

a sequence of 10 consecutive frames with their respective

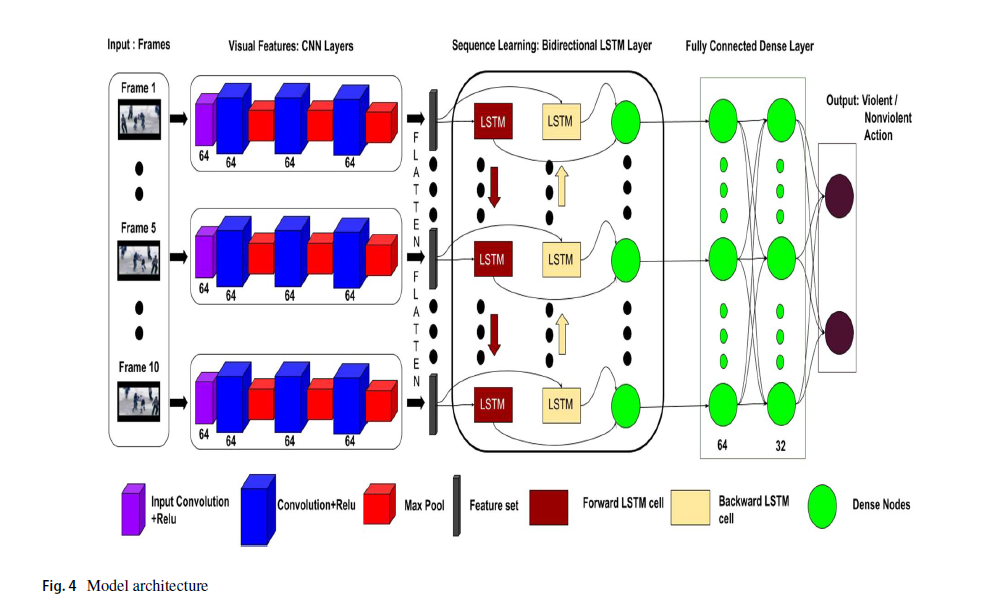
class labels is prepared. The shape of the training data is

(-1, N, x, y, c)4. Here, *c* represents the number of channels in

each frame. The pictorial representation of the training data

is shown in Fig.5.

**#Model creation**

****

**We made a sequential Consists of layers (CNN , LSTM , Dense)**