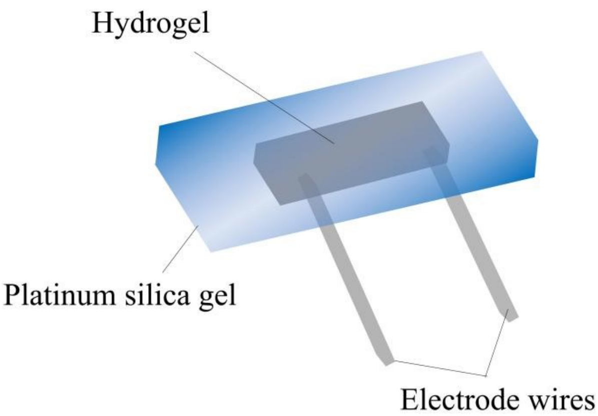

Fig.S1 Schematic diagram of hydrogel strain sensor package

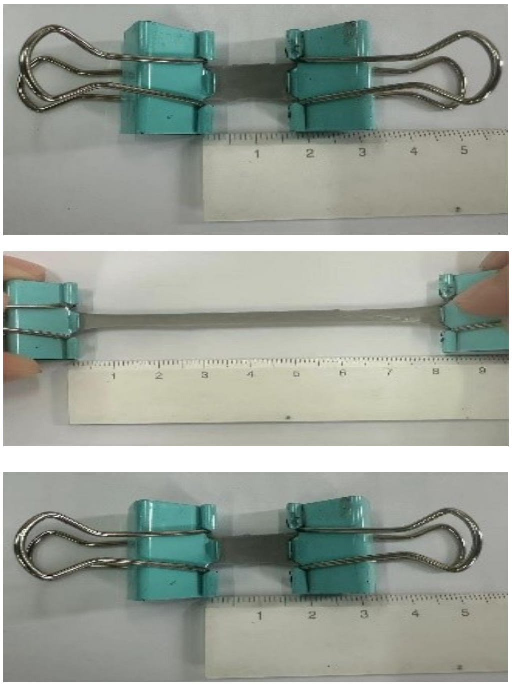

Fig.S2 The self-recovery ability of the hydrogel.

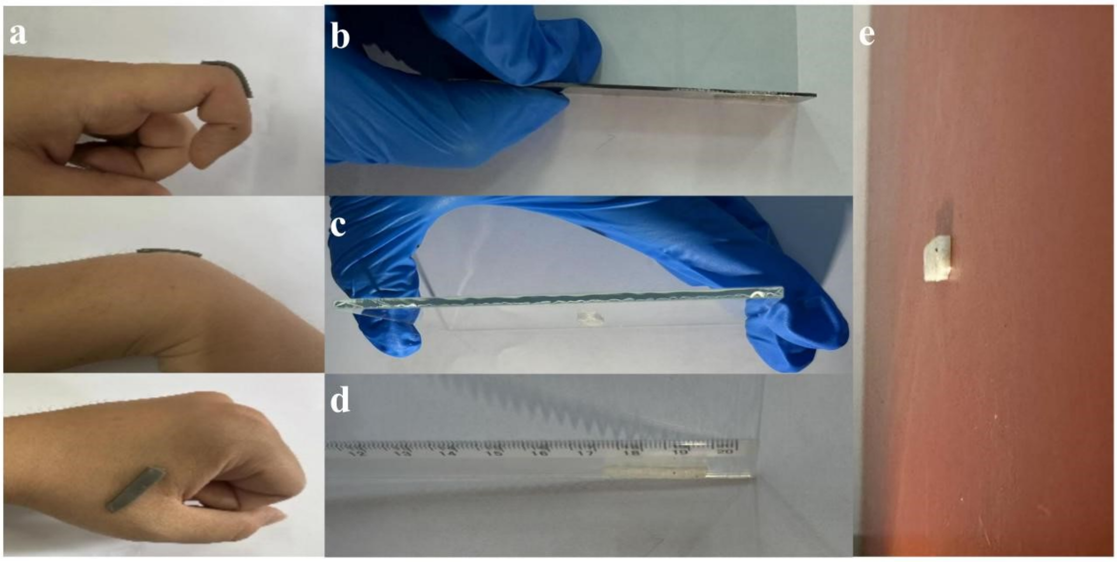

Fig. S3 Adhesion test of hydrogel on different materials ,(a) human body (b) metal (c) glass (d) plastic (e) wood

Fig.S4 Conductive filler excess sample

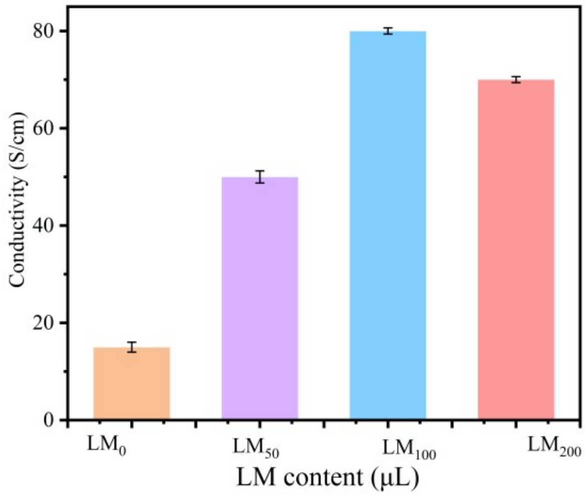

Fig.S5 Comparison of conductivity of samples with different concentrations

Fig.S6 LED small light bulb experiment

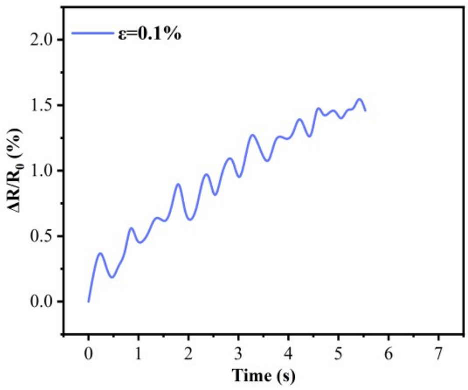

Fig.S7 Hydrogel detection of small strains

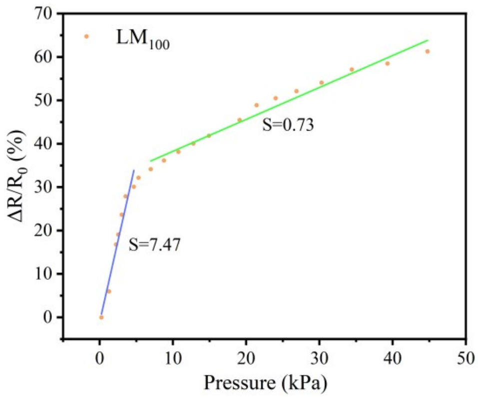

Fig.S8 Hydrogel compression sensitivity

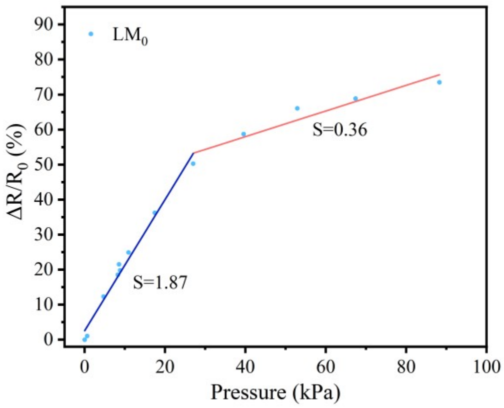

Fig.S9 Hydrogel compression sensitivity without liquid metal addition

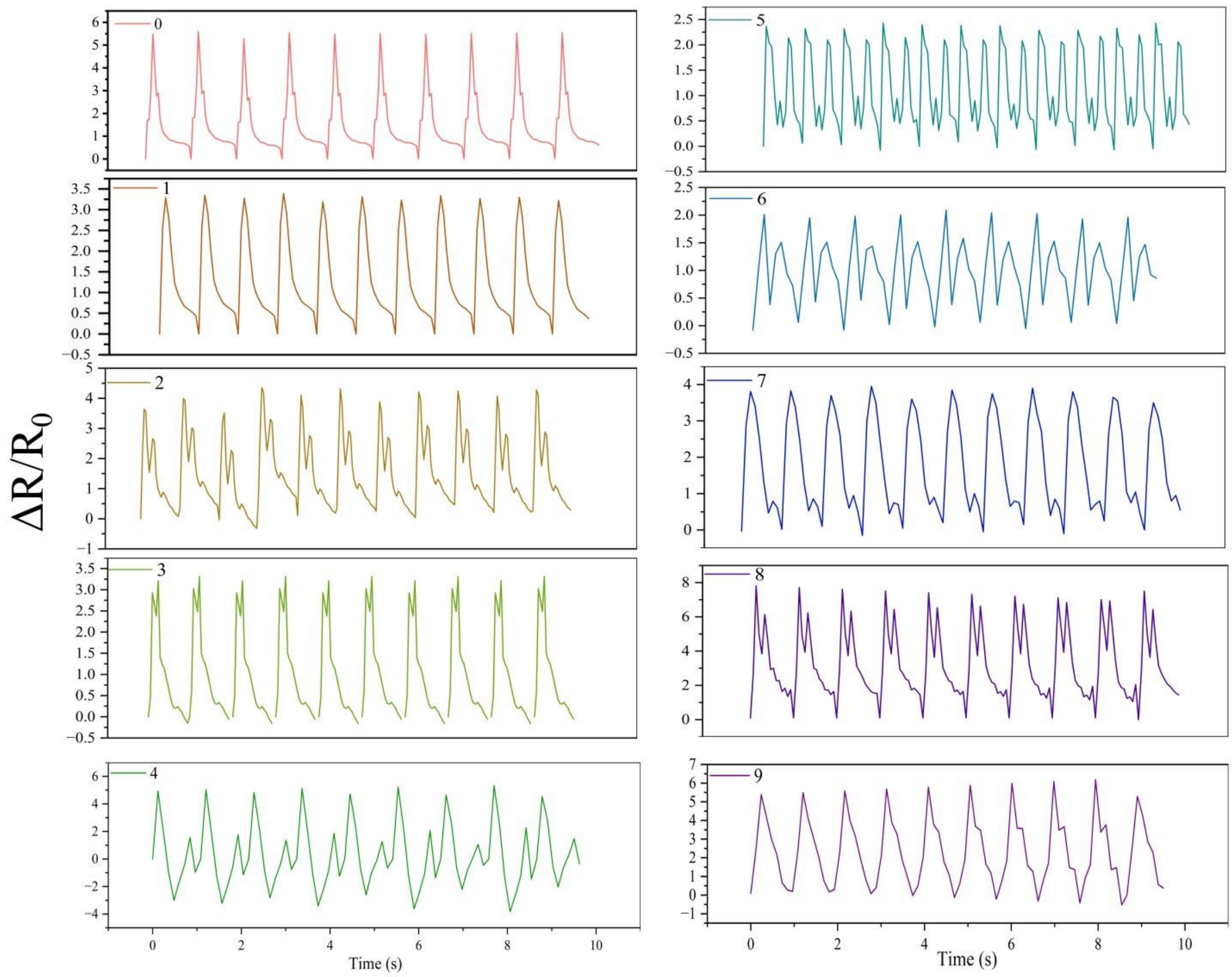

Fig.S10 Compression-Contact Mode Signal

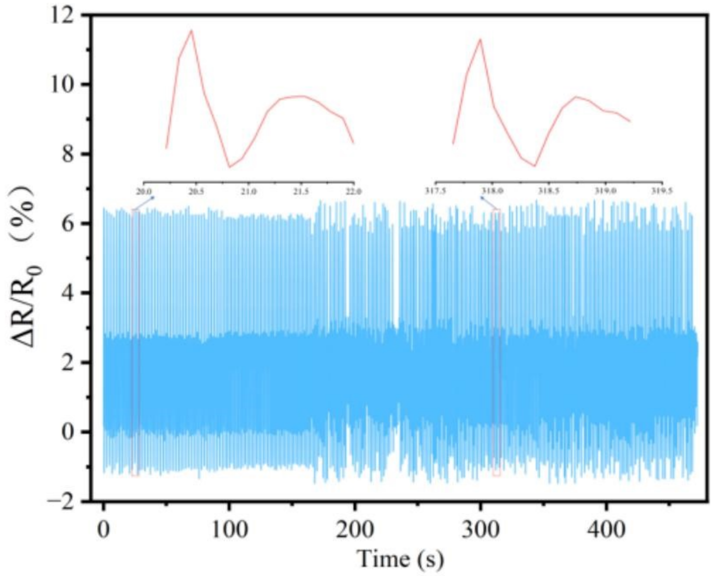

Fig. 11 Letter “a” tested in compression-contact mode cycle 300 times

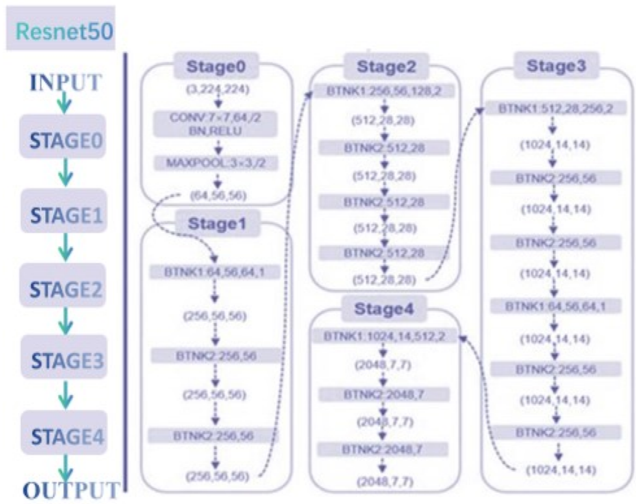

Fig.S12 Schematic diagram of Resnet50 algorithm

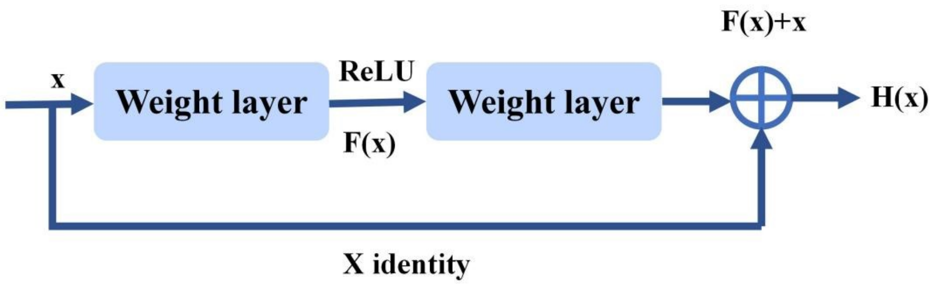

Fig.S13 The building block of residual learning.

As shown in Figure S8,the residual model in the figure passes the input x directly to the output H(x) via a jump connection and adds it to F(x) after activation by two weighting layers and ReLU.

H(x) and adds it to F(x) after two weighting layers and ReLU activation.H(x)represents the output of the current residual unit, and its expression is as follows:

$$H(x)=F(x)+x$$

When the network training reaches a relatively saturated accuracy rate, then next learning process is equivalent to identity mapping learning, that is,H(x)=x, and the subsequent training goal becomes to make the residual F(x) approach 0. As the network deepens, the accuracy rate no longer decreases.

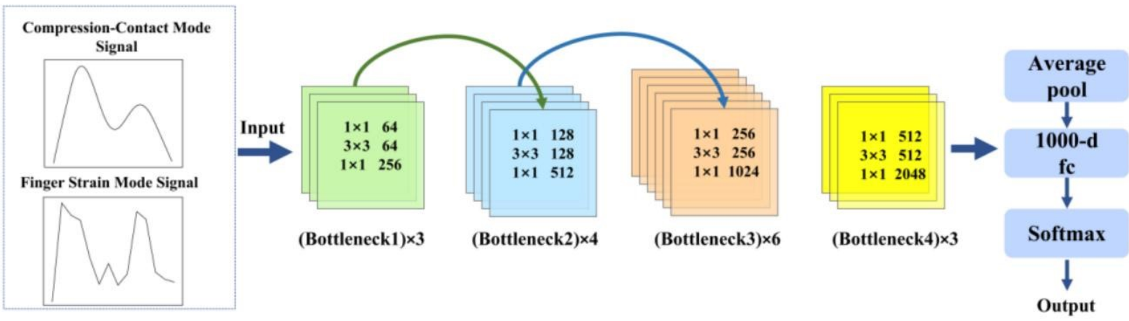

Fig.S14 The principle of the recognition algorithm

Fig.S14 shows the ResNet50 is a pre-trained model whose core principle is to improve the performance of deep neural networks through parameter migration and residual learning. The basic unit of the model is the bottleneck residual block. Each bottleneck residual block consists of three convolutional layers designed to efficiently learn features while reducing computational complexity.

The bottleneck block first uses a 1×1 convolutional layer to reduce the number of input channels, thereby reducing computational cost. Next, a 3×3 convolutional layer is utilized for feature extraction, and the number of channels is restored by another 1×1 convolutional layer, allowing the network to learn deeper features. After the input data is processed by these convolutional layers, the resulting feature mapping is added to the original input data to form the output of residual learning.

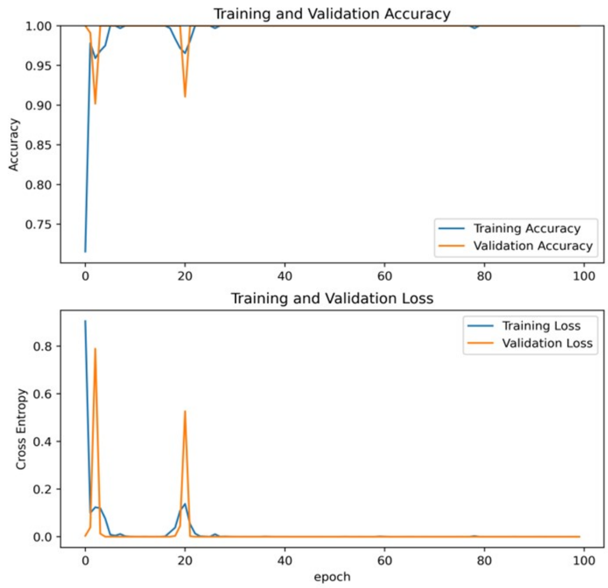

Fig.S15 Compressive-contact mode accuracy and loss in training and validation of 0-9 numbers

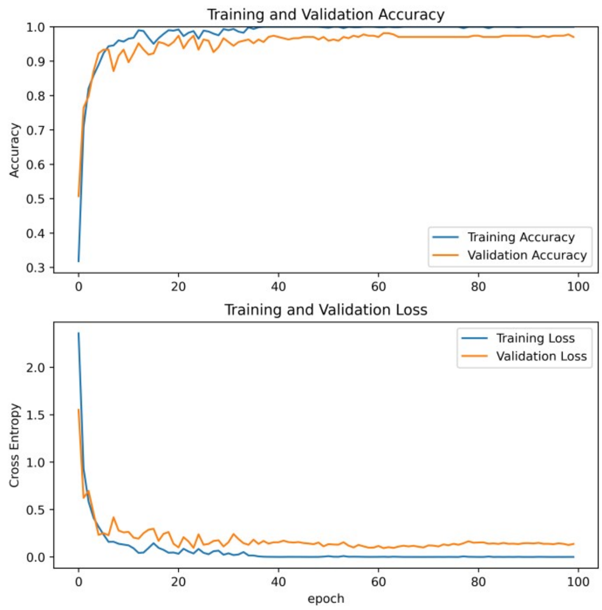

Fig.S16 Compressive-contact mode accuracy and loss in training and verification of a-z letters

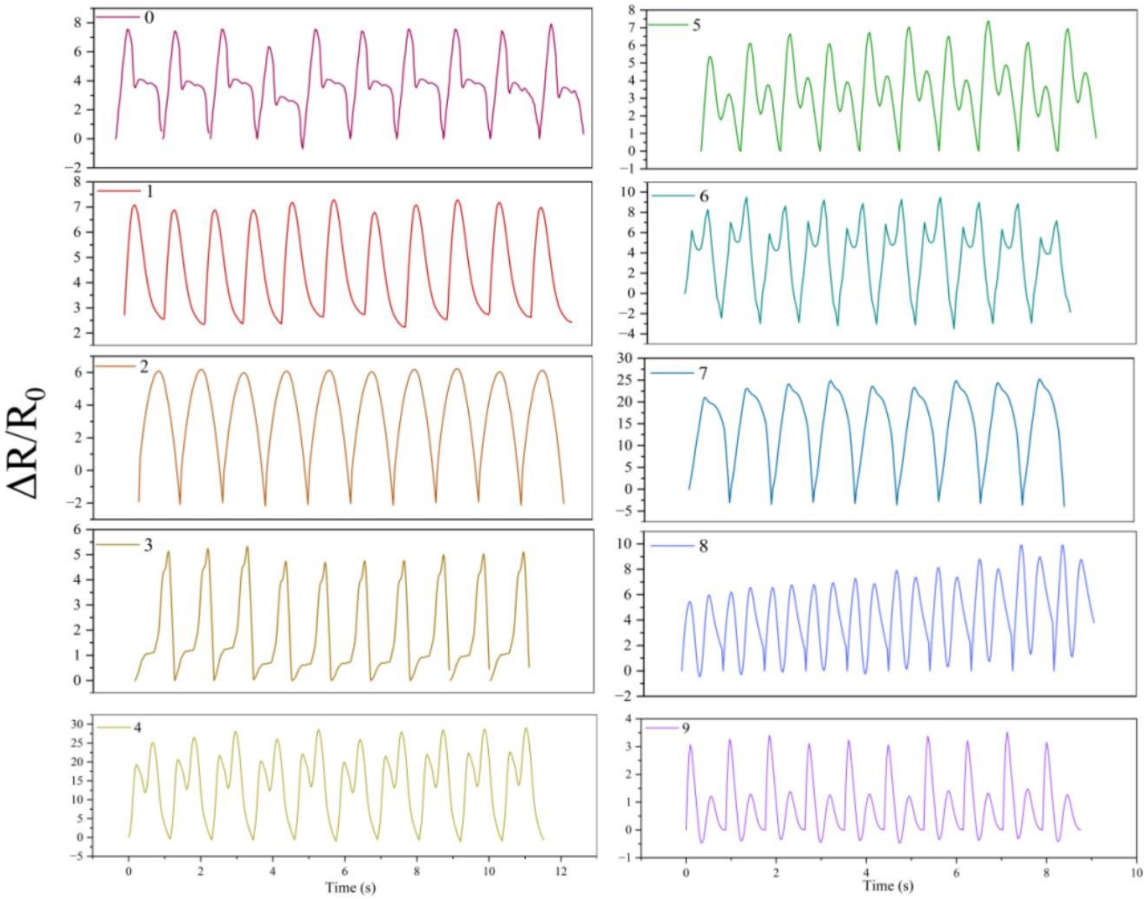

Fig.S17 Finger Strain Mode Signal

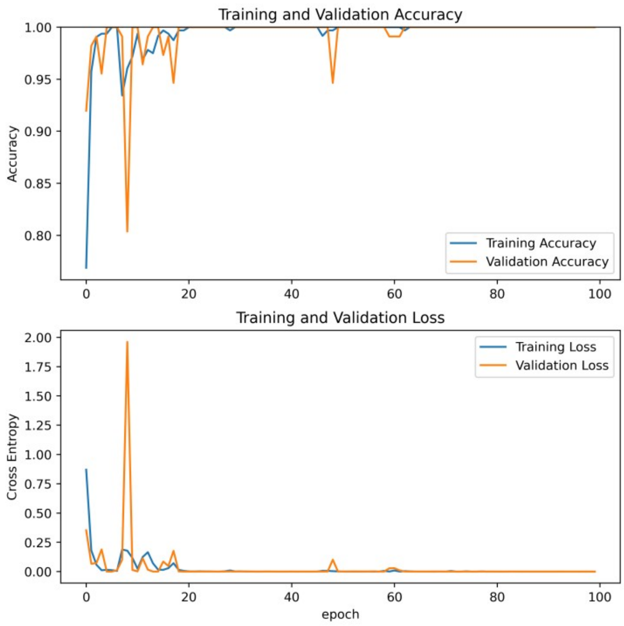

Fig.S18Finger-strain mode accuracy and loss in training and validation of 0-9 numbers

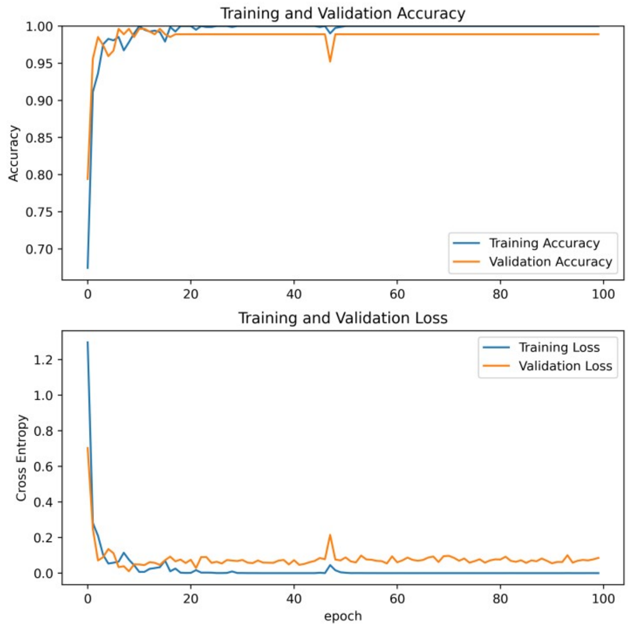

Fig.S19 Finger-strain mode accuracy and loss in training and validation of a-z numbers