

## 1.Environment

[https://drive.google.com/file/d/1YsqlO861z4FoSEZE3sRUhrPRWG\\_itYk/view?usp=share\\_link](https://drive.google.com/file/d/1YsqlO861z4FoSEZE3sRUhrPRWG_itYk/view?usp=share_link)

## 2. Implement details

### 1. model architecture:

- In task 1, I choose resnet18 as my deep learning model , since it is a suitable model for the image recognition. I use pretrained resnet18 to run the train, and also change the output layer to 10 to fit the ten possible number.
- In task 2, I also choose resnet18 as my deep learning model , but I change the output layer to 72 to fit the ten possible number plus 26 character for two image .
- In task 3, I also choose resnet18 as my deep learning model , but I change the output layer to 144 to fit the ten possible number plus 26 character for four image .
- In task 2 and task 3, I use one hot label to represent English character , by making a len of 32 list and make the mapping value to one if the key is found. Meanwhile, I also choose MultiLabelSoftMarginLoss to compute loss between multilabel.
- In all the three dataset , I shrink the value of the img, so that the value is good for the model to train .

### 3.model weight place

model 1:

[https://drive.google.com/file/d/1BiCzUrEwphratxQtllC6lgLFjmAjDMHH/view?usp=share\\_link](https://drive.google.com/file/d/1BiCzUrEwphratxQtllC6lgLFjmAjDMHH/view?usp=share_link)












model 2:


[https://drive.google.com/file/d/1EGRsZJ54ddOTJ9\\_CNIFNBb25FeNBVq1l/view?usp=share\\_link](https://drive.google.com/file/d/1EGRsZJ54ddOTJ9_CNIFNBb25FeNBVq1l/view?usp=share_link)

model 3:

[https://drive.google.com/file/d/1FYssG\\_GBFADS2sr5GoX9T4-WH1FoW9HG/view?usp=share\\_link](https://drive.google.com/file/d/1FYssG_GBFADS2sr5GoX9T4-WH1FoW9HG/view?usp=share_link)

#### 4.screenshot score

#	Team	Members	Score	Entries	Last	Code
1	0810756		0.99980	1	4d	
2	109550171		0.99940	10	11h	
3	109550182		0.99920	5	13h	
4	109550136		0.99900	6	4h	
5	109550018		0.99880	9	4d	
6	109550100		0.99880	4	1d	
7	109550099		0.99860	9	5d	
8	109704011		0.99820	30	2d	
9	109550158		0.99800	16	3h	
10	109550134		0.99800	2	8h	
11	<b>109550164</b>		0.99780	9	3h	



Your Best Entry!  
Your most recent submission scored 0.99780, which is an improvement of your previous score of 0.99760. Great job!

Tweet this