Automated Math Expression Solver

Group 18: Teaden Gurung, Faizan Haque, Brian Tom, Gordon Tze

Relevant Topics:

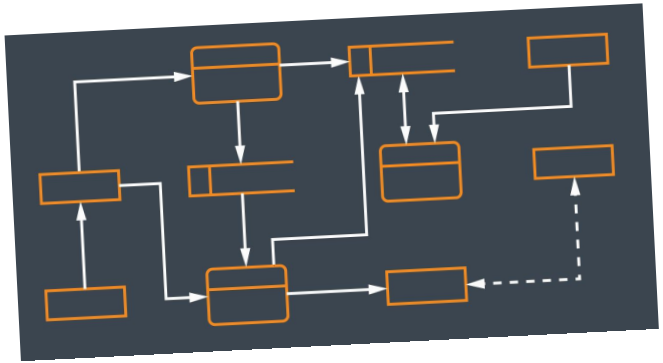
- Image Processing
- Convolutional Neural Networks
- Data Augmentation

A (very) high-level overview

We built a math expression solver

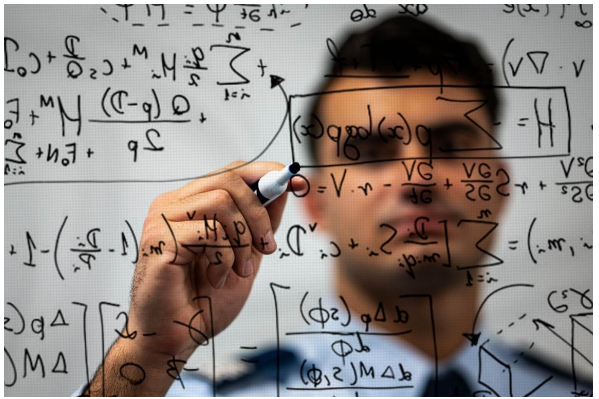
f(**handwritten expressions**) = ??? = **numerical output**

The Inspiration



We wanted an
end-to-end project,
a *full* product!

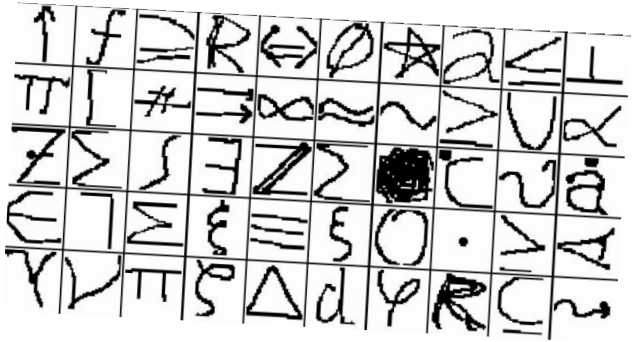
We thought images
were cool



Most of us are CS +
MATH majors/minors!

The *Dataset*

32x32px image
369 symbols
150k samples



“HASYv2” → A Handwritten Symbol Database

- Handwritten Instances of symbols, numbers, letters.
- Labels = {“0”, “A”, “\sqrt”, “\neq”, “\inf”, ...}

We want to use {**0,1,2,3,4,5,6,7,8,9,*,/,+,-,(,),\sqrt**} ⊂ HASYv2

- Goal: To simply perform arithmetic!

*Ok, **how**?*

1. **Image Segmentation**
   - Perform image segmentation of input images into its individual components
2. **Image Classification**
   - Classify the segments into what they actually are!
3. **Expression Parsing**
   - Combine the segments, parse/handle them (dealing with roots, ( ), etc.)
4. **Expression Solving**
   - Solve it!

IMAGE SEGMENTATION

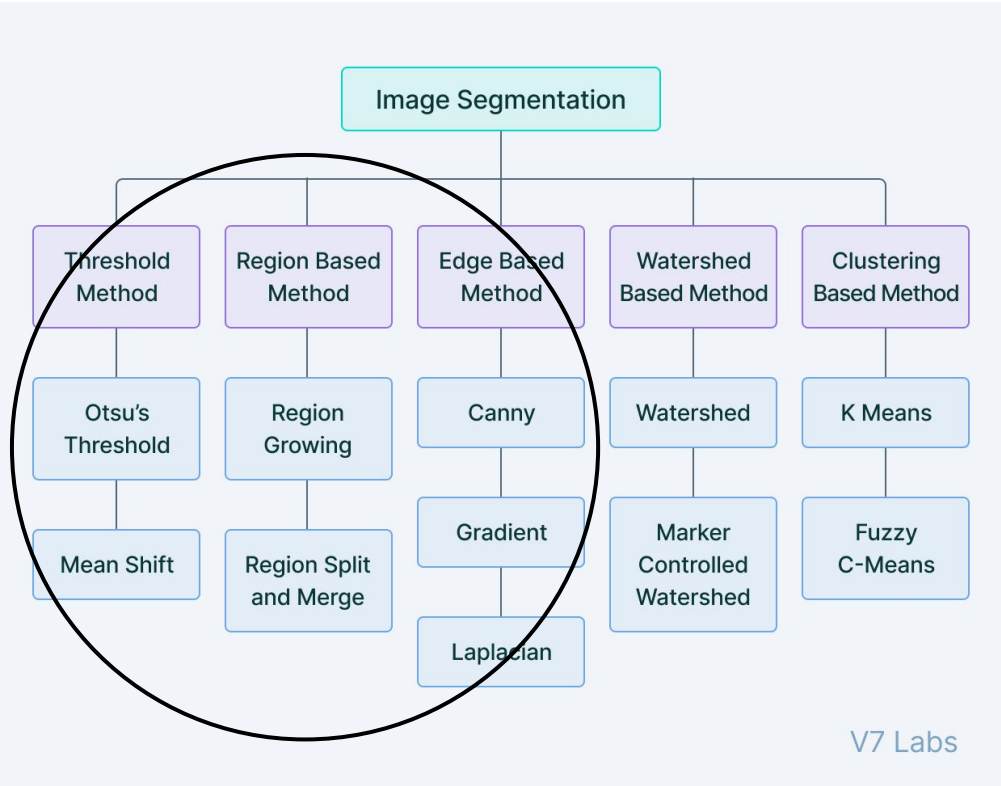
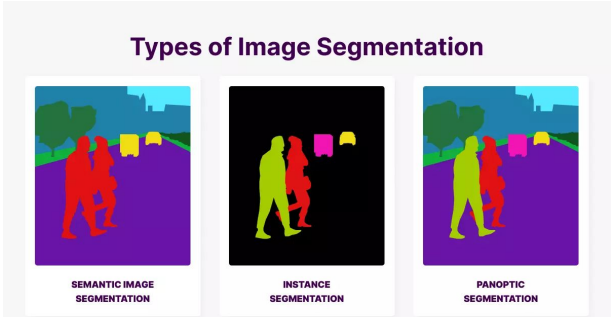
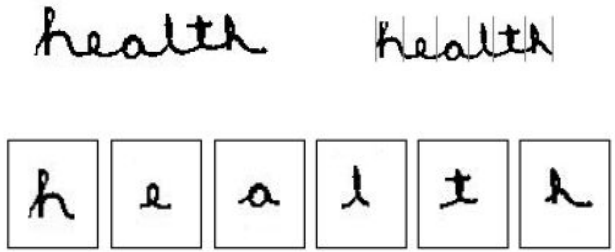


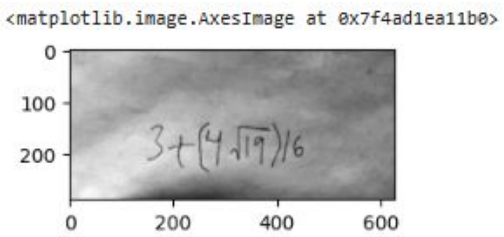
Image Segmentation
Threshold Method
Region Based Method
Edge Based Method
Watershed Based Method
Clustering Based Method
Otsu's Threshold
Region Growing
Canny
Watershed
K Means
Mean Shift
Region Split and Merge
Gradient
Marker Controlled Watershed
Fuzzy C-Means
Laplacian
V7 Labs

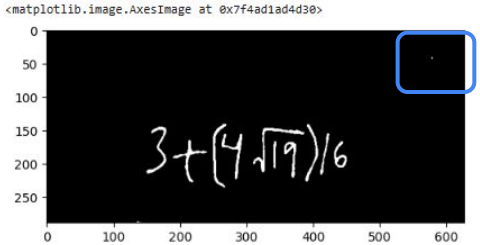

Types of Image Segmentation
SEMANTIC IMAGE SEGMENTATION
INSTANCE SEGMENTATION
PANOPTIC SEGMENTATION

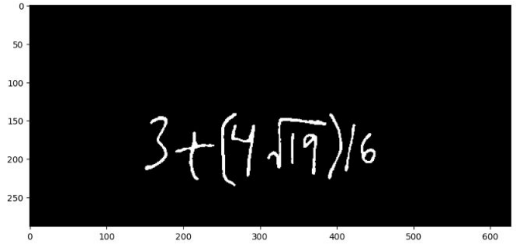

health
health

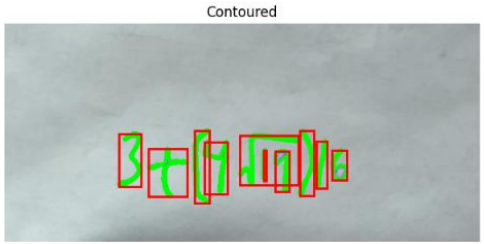
We use:

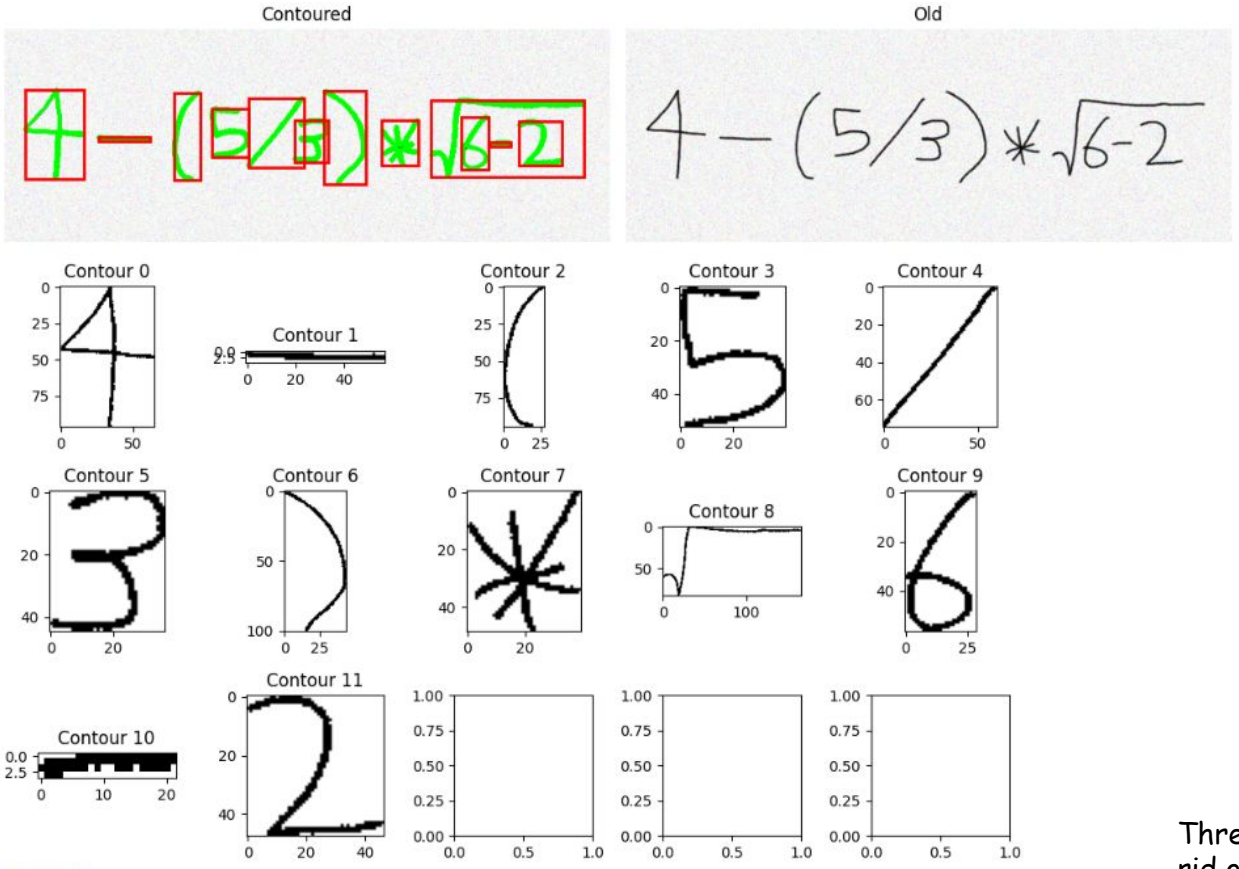
1. Basic image processing (blurring, B/W)
2. Thresholding (Otsu's Method)
3. Connected Components
4. **Contour** (borders) Detection











Bounding boxes
Segment 1: (22, 66, 65, 97)

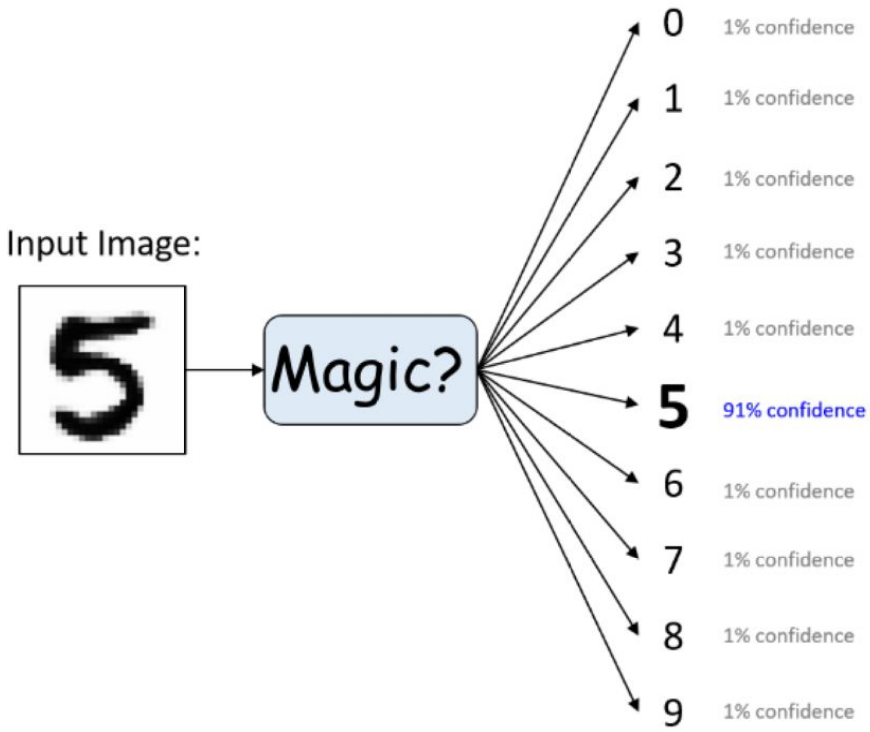
Thresholding is very important to get rid of noise!

IMAGE CLASSIFICATION

We now want to ***classify*** each segment of the expression

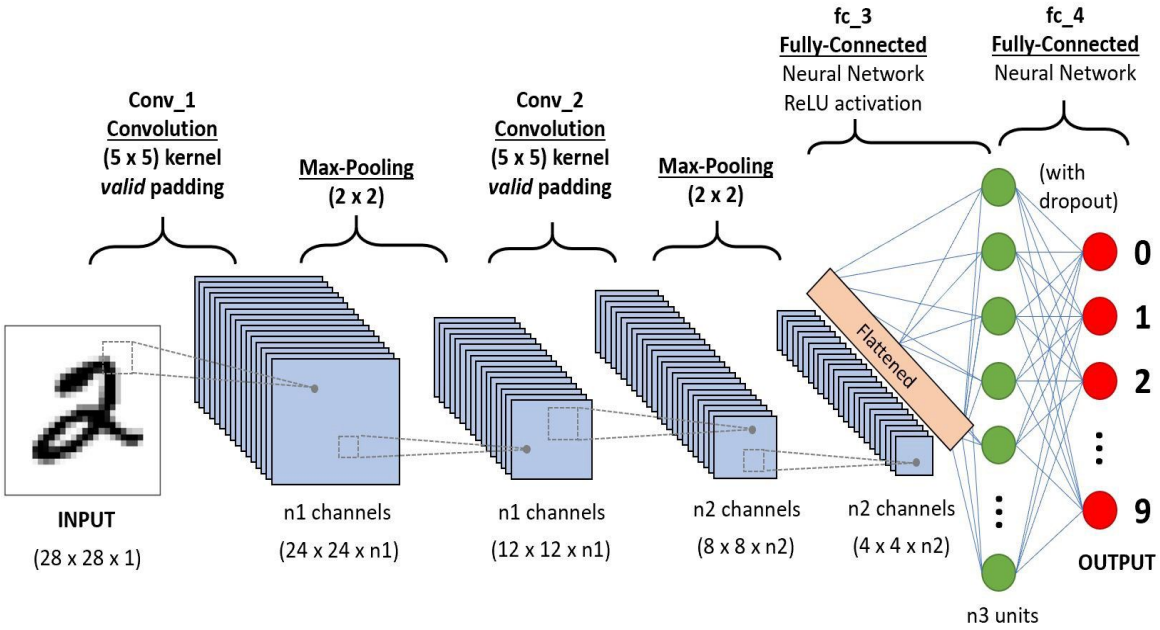
Numerous different methods. But most popular:

Convolutional Neural Networks
(Yann LeCun et al, 1998)



An overview of CNNs

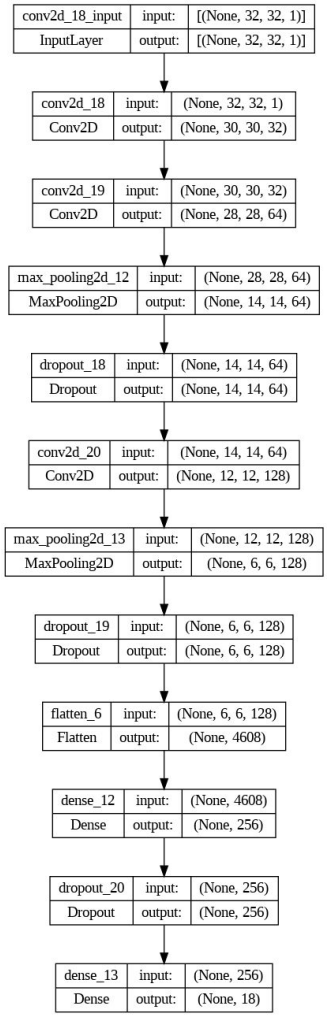
- Input layer
- **Convolutional Layers**
  - Where the core learning happens. Key structures of the images are learned (edges, botches, eventually learning entire shapes)
- **Max Pooling**
  - Reduces dimensions but retains most important information
- Fully connected layers
  - Back to neural network, classification occurs

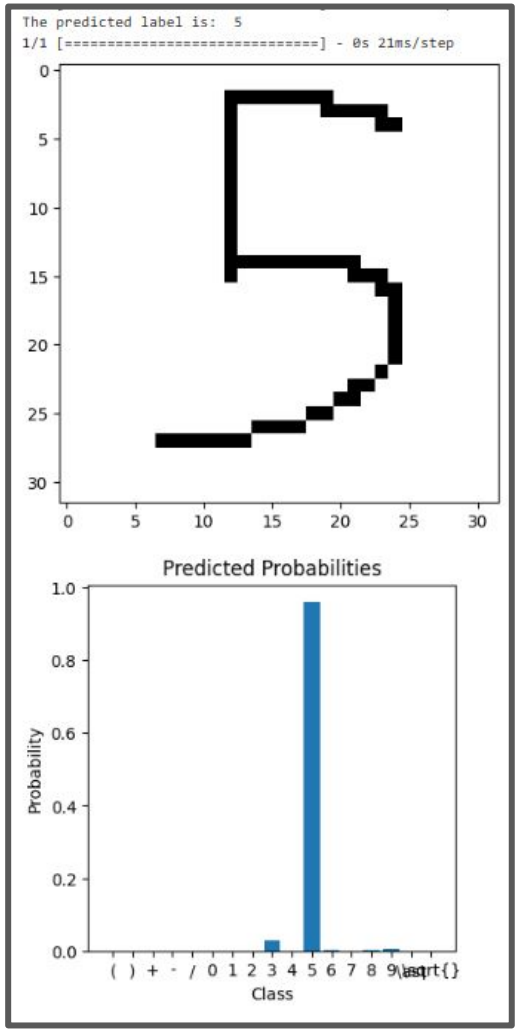


Layer (type)	Output Shape	Param #
conv2d_18 (Conv2D)	(None, 30, 30, 32)	320
conv2d_19 (Conv2D)	(None, 28, 28, 64)	18496
max_pooling2d_12 (MaxPooling2D)	(None, 14, 14, 64)	0
dropout_18 (Dropout)	(None, 14, 14, 64)	0
conv2d_20 (Conv2D)	(None, 12, 12, 128)	73856
max_pooling2d_13 (MaxPooling2D)	(None, 6, 6, 128)	0
dropout_19 (Dropout)	(None, 6, 6, 128)	0
flatten_6 (Flatten)	(None, 4608)	0
dense_12 (Dense)	(None, 256)	1179904
dropout_20 (Dropout)	(None, 256)	0
dense_13 (Dense)	(None, 18)	4626

Total params: 1,277,202
Trainable params: 1,277,202
Non-trainable params: 0

Around 98% accuracy

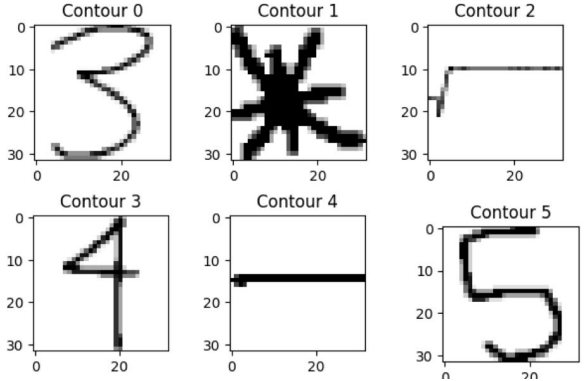




EXPRESSION PARSING AND SOLVING

After classifying: **Parse** it

- Finding square root with bounding boxes
- Handling parentheses
- Converting “\\ast” into “*”




```
array(['3', '\\ast', '\\sqrt{}', '1', '-', '5'], dtype='<U7')
```

```
A square root found at index 2 has bounds: (151 331)
Added current bound of (192 223)
Added current bound of (234 259)
Added current bound of (271 304)
```

After parsing: **solve** it!

- SymPy
  - Symbolic computation library

- We use SymPy library to solve and evaluate the expression


Image Segmentation
Image Classification
Expression Parsing
Expression Solving
9.118
{2,*,5,-,sqrt,7,/,3}
2*5 - sqrt(7)/3

THE DEMO

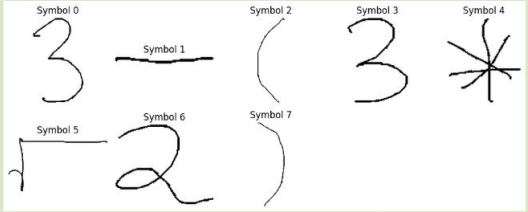
Please look at laptop #3

Group 18 - Expression Solver

Original

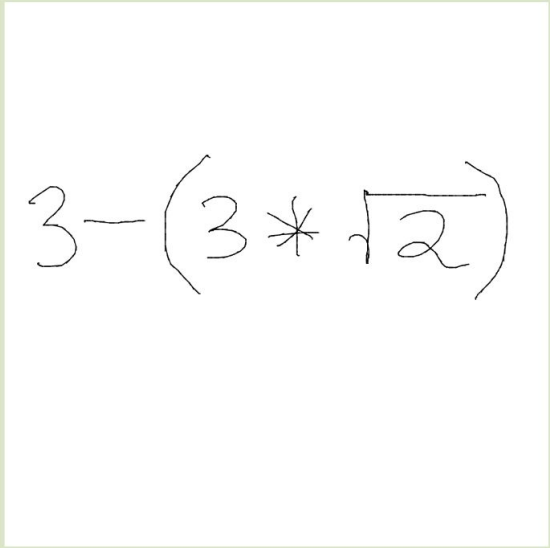
Contours

Terms



Expression

3 - (3 * sqrt(2))



Upload

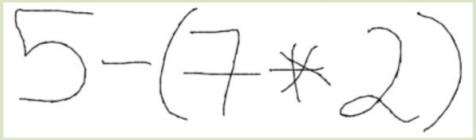
Browse... equation3.jpg

Result

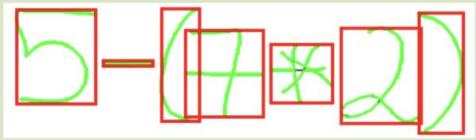
-1.243

Group 18 - Expression Solver

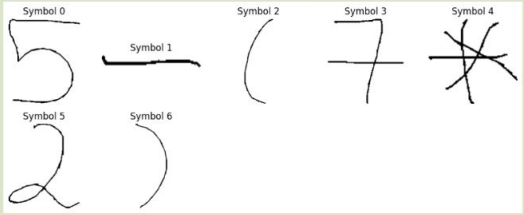
Original



Contours

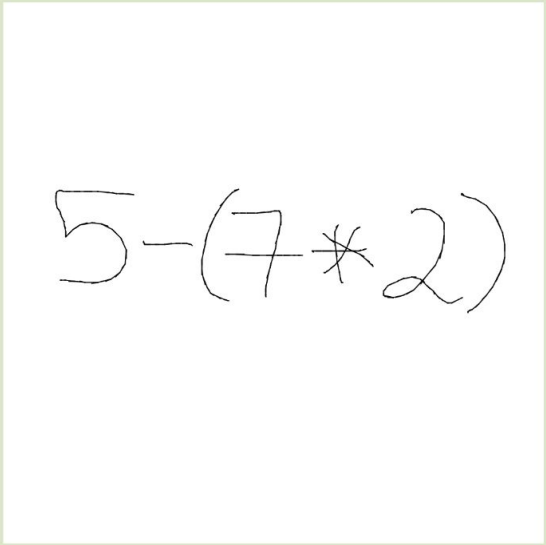


Terms



Expression

5 - (7 * 2)



Upload

Browse... equation2.jpg

Result

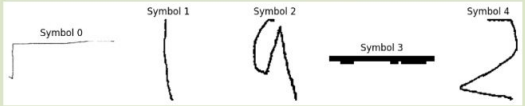
-9

Group 18 - Expression Solver

Original

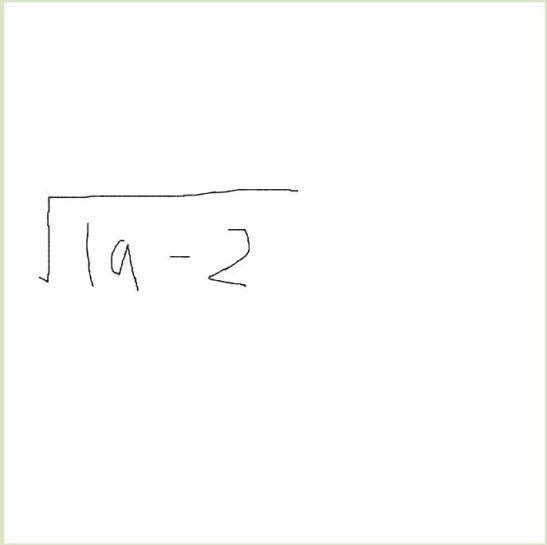
Contours

Terms



Expression

sqrt(1 9 - 2)



Upload

Browse... equation6.jpg

Result

4.123