Recognizing Text From CAPTCHA

Team Members
Jaspreet Singh (201202078)
Akash Agrawall (201202061)
Kalpit Thakkar (201201071)

Mentor : Amrisha Vohra

Project Description

- Aim : To develop an algorithm to solve the CAPTCHAs by recognizing the text, using different classifiers.
- Classifiers/Approaches tried :
  1. Template Matching
  2. Support Vector Machines
  3. Decision Trees
  4. k-NN classifier

Steps Involved

- Finding a suitable Dataset for training/testing
- Training + Testing the Classifier
- Preprocessing the input image
- Segmenting the CAPTCHA image
- Obtain the class labels through the classifier

Dataset

- We tried our hands at a wide range of character datasets :
  1. The Chars47K Dataset
  2. Algoval Essex Dataset
  3. EZ-Gimpy Dataset

- Each of the dataset gave different results, though we didn't actually use the 2nd one as it required a lot of preprocessing.

Training the Classifier

- We tried a lot of features for our training phase :
  1. Zernike's Transformation (Translation, Scale and Rotation invariant)
  2. Hu's 7 transforms (Translation and Scale invariant)
  3. Zoning, Pixel density, centroid, etc.
  4. Heirarchical Centroid

Preprocessing and Segmentation

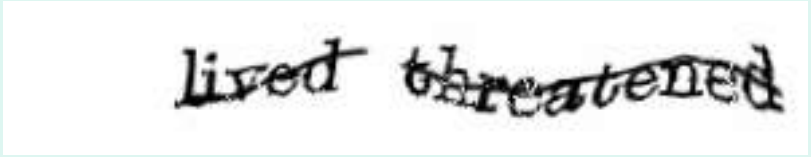
- Binarization by histogram thresholding
- Labelling the blobs and finding the connected components
- Getting the centroid, area and bounding box of the blobs
- Using morphological operations to connect disjoint character segments
- Using bounding box obtained to segment out the character

Conneting disjoint character segments

- Morphological operations served the purpose of solving this problem
- We use a "diamond" structuring element and perform :
  1. Morphological dilation with element size = 2
  2. Morphological erosion with element size = 1
- This helps in making sure that while dilating to join segments, two adjacent characters don't join

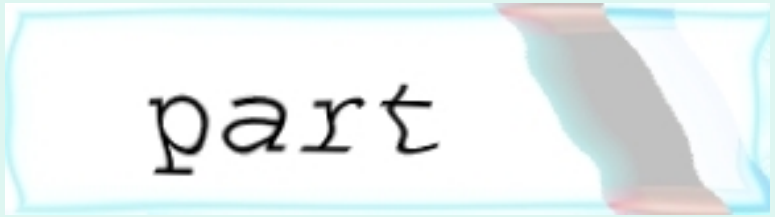
Results on various datasets

Dataset image when
Template Matching was used



Dataset image that we could not
Segment (negative kerneling)

Dataset image for which we
used SVM classifier
(has collapsed letters)



EZ-Gimpy Dataset image for which
we used k-NN and Decision Trees
Best results so far!

Analysis

- Template matching gave highest accuracy as it was used on a simple dataset
- k-NN gave the best results with a high accuracy
- Decision trees also worked well, but not as good as k-NN
- SVM did not work that well, because the Dataset used with it wasn't good

Analysis

Classifier	% Accuracy
Template Matching	100
k-NN	85
Decision Trees	70
SVM	29

Conclusions

- Segmentation of the image into it's component characters is the primary problem for CAPTCHA recognition.
- It is very important to choose a suitable training dataset to ensure good results.
- k-NN classifier gives great results and is easy to train as well, in comparison to other classifiers.

Future Work

- The future scope of this project is to include more robust algorithm to segment the image.
- Neural Networks would work very well with such recognition tasks; it can be extended to use a k-NN on top of a Neural Network.
- Improving the general dataset required for OCR.

References

- Recognizing Objects in Adversarial Clutter : Breaking a Visual CAPTCHA, Greg Mori, Jitendra Malik, CVPR 2003.
- Text-based CAPTCHA Strengths and Weaknesses; Elie Bursztein, Matthieu Martin, and John C. Mitchell, Stanford University.

Thank you!