Handwritten English Alphabet Recognition Using Bigram Cost
Chengshu (Eric) Li

Introduction & Motivation

Handwritten character recognition has been one of the most challenging and fascinating areas in the field of image processing. It has a wide variety of applications:
- receipt/invoice recognition
- business card information extraction
- books cannning
- assistive technology for blind

My approach is to use both image recognition and bigram cost between English alphabets to achieve high performance.

Feature Extraction

- Raw pixels: used as my baseline
- Blackness threshold: an approximation of the original matrix
- Blackness percentage: another approximation
- Zoning: put a 3 by 3 grid on top of the original image. Use aspect ratio to classify each grid to six different types

Test Result

<table>
<thead>
<tr>
<th>Model</th>
<th>Error Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Softmax</td>
<td>0.1</td>
</tr>
<tr>
<td>Softmax with bigram cost</td>
<td>0.08</td>
</tr>
<tr>
<td>SVM</td>
<td>0.07</td>
</tr>
<tr>
<td>SVM with bigram cost</td>
<td>0.06</td>
</tr>
<tr>
<td>Naive Bayes</td>
<td>0.05</td>
</tr>
<tr>
<td>Naive Bayes with bigram cost</td>
<td>0.04</td>
</tr>
<tr>
<td>Neural Network</td>
<td>0.03</td>
</tr>
<tr>
<td>Neural Network with bigram cost</td>
<td>0.03</td>
</tr>
</tbody>
</table>

NIST Database 19 (800,000+ hand-written samples)
I used 19240 samples (370 samples for each of the 52 upper and lower case English alphabet)

Data & Preprocessing

Preprocessing:
- Crop out the central part where the character lies
- Resize it to a standardized size (e.g. 128×128 pixels)

Conclusion & Future Work

Conclusion:
- Test error decreases when training data increases
- Convolutional neural network performs significantly better than other models
- Bigram cost helps improve accuracy

Future Work:
- Find better features to feed in SVM and Naïve Bayes
- Improve performance on ‘bottleneck’
- Extend bigram cost method to words, not just characters