### Introduction
Spotting words from a possibly large range of vocabulary is challenging. Training a discriminative model requires many positive samples per word. We present a word spotting method for historical document images using convolutional siamese network. This method can spot
1. Words with varying writing styles and backgrounds
2. Out of vocabulary words

### Method
The method works on segmented word images using query-by-example. It is based on training a discriminative model to rank the similarity of two input images regardless of being in or out of vocabulary. On the other hand, previous methods for the same problem are based on using word embedding.

#### Siamese CNN

![Siamese CNN Diagram](image)

**Figure 1.** A siamese network consists of two Convolutional Neural Network (CNN) with same architecture and weights ($W$).

CNN is trained to map pixels into linear space in which $L_2$ distance of the two feature vectors is:
- close if the inputs are same and
- far if the inputs are different

**How does contrastive loss work?**

Contrastive loss pulls the similar points together, pushes the different points apart.

![Contrastive loss](image)

**Figure 2.** Solid circles represent similar, hollow circles represent dissimilar points to the point in the center. Forces acting on the points are shown in blue arrows with length gives the strength of the force.

### Results
George Washington (GW) dataset [1] contains 929 classes with 2372 words. The model was trained using 5 fold cross-validation and tested on completely OOV. It achieves mAP value of 0.49 whereas the work of Rodriguez and Perronin [2] achieves 0.53.

![GW dataset](image)

**Figure 3.** Qualitative results on GW dataset.

<table>
<thead>
<tr>
<th>Book#</th>
<th>mAP</th>
<th>P@1</th>
<th>P@2</th>
<th>P@3</th>
<th>Queries</th>
<th>Labels</th>
<th>OOV</th>
</tr>
</thead>
<tbody>
<tr>
<td>Book1</td>
<td>0.91</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>2100</td>
<td>21</td>
<td>21</td>
</tr>
<tr>
<td>Book2</td>
<td>0.75</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>2100</td>
<td>21</td>
<td>14</td>
</tr>
<tr>
<td>Book3</td>
<td>0.82</td>
<td>1.0</td>
<td>1.0</td>
<td>0.98</td>
<td>2100</td>
<td>21</td>
<td>16</td>
</tr>
<tr>
<td>Book4</td>
<td>0.75</td>
<td>1.0</td>
<td>1.0</td>
<td>0.98</td>
<td>2100</td>
<td>21</td>
<td>15</td>
</tr>
<tr>
<td>Book5</td>
<td>0.80</td>
<td>0.95</td>
<td>0.95</td>
<td>0.95</td>
<td>10500</td>
<td>58</td>
<td>47</td>
</tr>
<tr>
<td>All</td>
<td>0.62</td>
<td>0.95</td>
<td>0.95</td>
<td>0.93</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Table 1.** mAP and P@k values of the model trained on 500 words from book1 and tested on 5 books of VML dataset.

### Conclusion
The method is general and can be applied to historical documents in different languages. Convolutional network makes the method robust to varying writing styles and backgrounds. Discriminative model can be used to spot OOV words. As the number of OOV words increases performance decreases. That's why results on VML are more successful than results on GW dataset.

### References
[1] https://skr.cs.umass.edu/download
[3] https://mmlab.bgu.ac.il/VML