Word Spotter

Team Members:

- Ritvik Agrawal (2018122005)
- Kirandevraj R (2019701001)
- Vivek Chandela (20171195)

Github Repo:

https://github.com/Kirandevraj/Handwritten-Word-Spotting-with-Corrected-Attributes

Objective

- Find all instances of a given word in a large dataset with multi-writer setting
- Attributes-based approach that leads to a low-dim, fixed length repr. of word images
- Implement particular case of semantic content based image retrieval (CBIR)
- 2 types of queries:
  - Query-by-example (Image)
  - Query-by-string (Text)

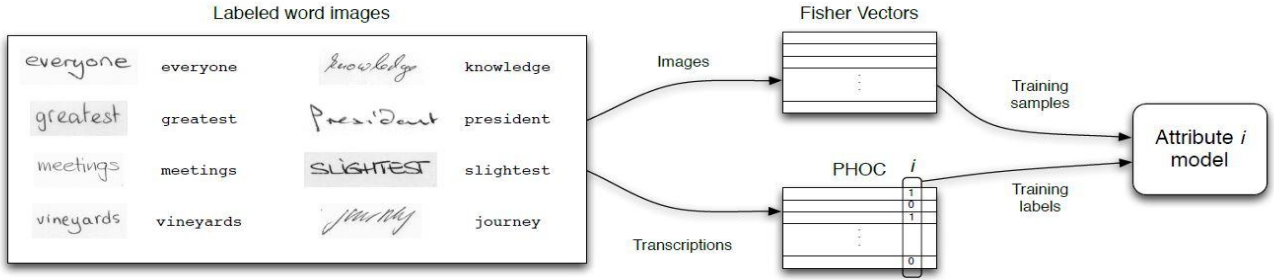

Figure 2. Training process for i-th attribute model. A classifier is trained using the FV representation of the images and the i-th value of the PHOC representation as label.

Challenges

- Very fine-grained classes - difference of one character is a negative result
- Very high intra-class variability - different writers may have completely different writing styles
- Variable-length features are unsuitable due to 3 reasons:
  - OOV (out of vocabulary) not possible so, only a known, limited number of keywords can be used as queries
  - Computing distance between words is very slow at test time
  - Although they are more flexible than fixed-length sequences, they don’t leverage supervised information to learn similarities and differences between different writing styles

Fisher Vector

- A Gaussian Mixture Model (GMM) is used to model the distribution of features (e.g. SIFT) extracted all over the image
- The Fisher Vector (FV) encodes the gradients of the log-likelihood of the features under the GMM, with respect to the GMM parameters.

$$\mathcal{G}^X_{\alpha_k} = \frac{1}{\sqrt{w_k}} \sum_{t=1}^{T} \left(\gamma_t(k) - w_k\right),$$

$$\mathcal{G}^X_{\mu_k} = \frac{1}{\sqrt{w_k}} \sum_{t=1}^{T} \gamma_t(k) \left(\frac{x_t - \mu_k}{\sigma_k}\right),$$

$$\mathcal{G}^X_{\sigma_k} = \frac{1}{\sqrt{w_k}} \sum_{t=1}^{T} \gamma_t(k) \frac{1}{\sqrt{2}} \left[\frac{(x_t - \mu_k)^2}{\sigma_k^2} - 1\right].$$

PHOC

Pyramidal Histogram of Characters:

- This binary histogram encodes whether a particular character appears in the represented word or not.
- E.g ‘listen’ and ‘silent’ have same L1 so we use L2 and so on
- Spatial pyramid representation ensures that the information of the characters order is preserved

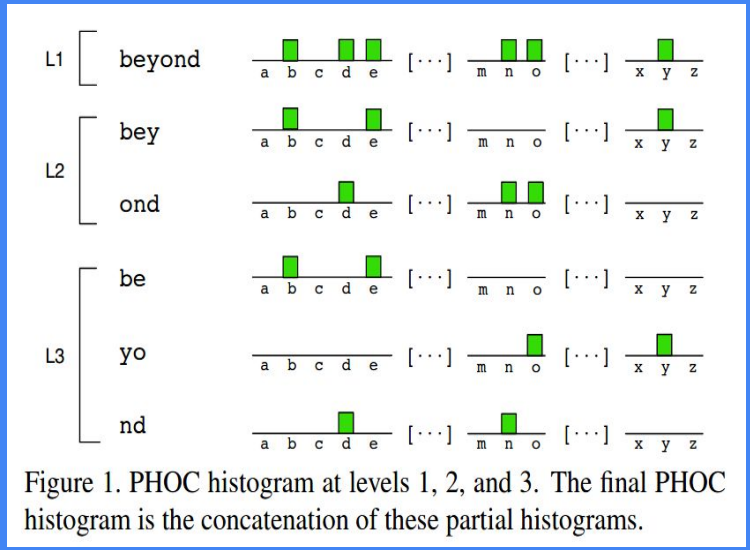

Figure 1. PHOC histogram at levels 1, 2, and 3. The final PHOC histogram is the concatenation of these partial histograms.

Pipeline

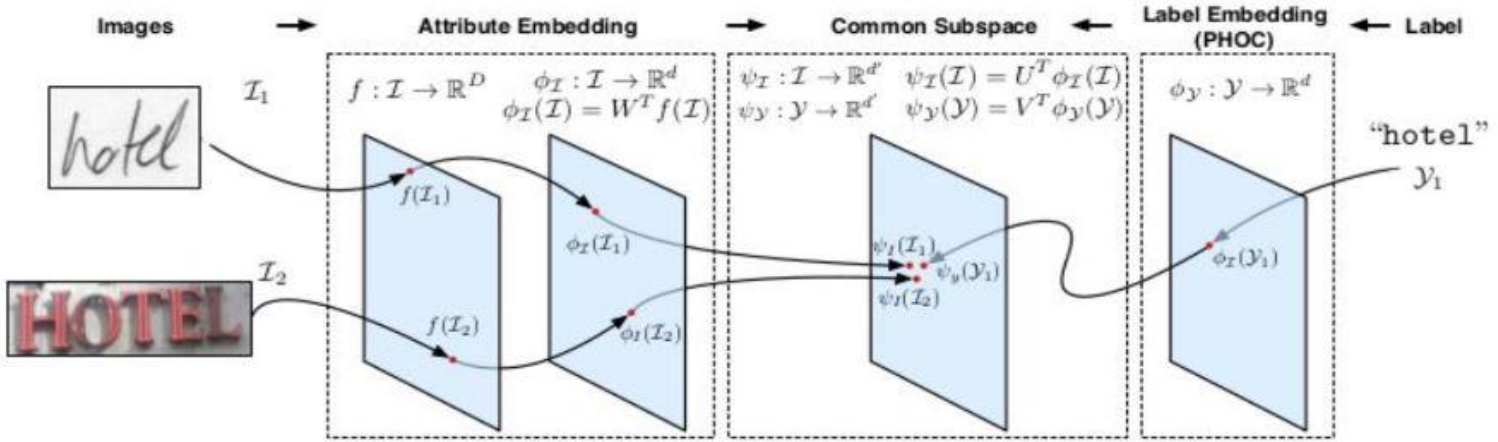

Fig1: Overview of the implemented method. Images are first projected into an attributes space with the embedding function after being encoded into a base feature representation. Then embedded labels and attributes in a learned common subspace

Algorithm

1) SIFT features are densely extracted from the images over a 2x6 spatial grid and reduced to 62 dimensions with PCA
2) Normalized x and y coordinates are appended to the projected SIFT descriptors
3) SIFT features are densely extracted from the image and aggregated into a FV representation using a vocabulary of 16 Gaussians.
4) Predict/train the PHOC attributes using a SVM classifier, given the FV.
5) Since we have the image and string for a word, actual PHOC attributes can be found by using the string input.
6) Using CCA (Canonical Correlation Analysis), get the projections of the predicted scores and the ground truth values.
7) Use cosine similarity to compute the mean average precision.

Canonical Correlation Analysis

- CCA helps in calibration of scores (prevent one attribute to dominate others) for the attributes jointly, since this can better exploit the correlation between different attributes.
- It is used to embed the attribute scores and the binary attributes in a common subspace where they are maximally correlated.
- It is also a dimensionality reduction tool.

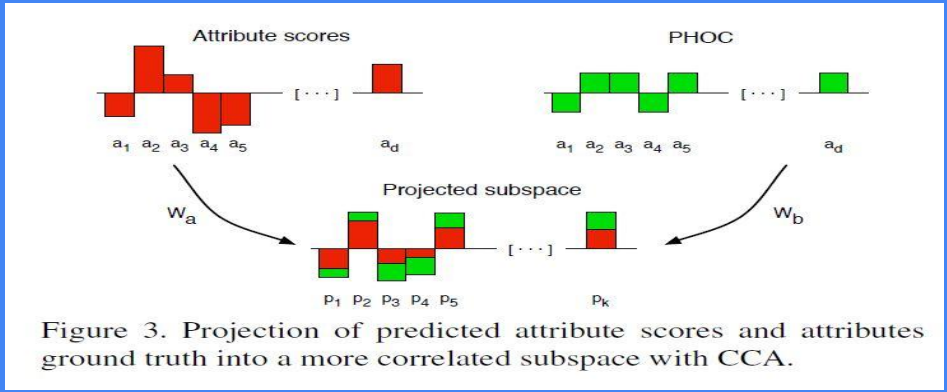

Figure 3. Projection of predicted attribute scores and attributes ground truth into a more correlated subspace with CCA.

Canonical Correlation Analysis

1) For N sample, we have attribute score representations A which is of dimension D x N
2) We have attribute representation B which is of dimension D x N
3) We calculate the covariance matrices C_{aa}, C_{bb} and C_{ab}
4) The goal of CAA is to find a projection of each view that maximizes the correlation between the projected representations.

$$\underset{w_a, w_b}{\operatorname{argmax}} \frac{w_a' C_{ab} w_b}{\sqrt{w_a' C_{aa} w_a} \sqrt{w_b' C_{bb} w_b}}.$$

5) This is solved through a generalized eigenvalue problem.
6) This equation has to be solved only once offline.

Kernel CCA

- We explicitly embed the data using a random Fourier feature (RFF) so that dot-product in the embedded space approx. corresponds to a gaussian kernel $K(x,y) = \exp(-\gamma \|x-y\|^2)$ in the original space.
- Next, we can perform linear CCA on the embedded space.
- So, KCCA helps to make relation between attribute scores and binary attributes linear.

Results

- We compute the mean average precision (map) of each query and report the mean of all the queries.

Figure 4. Sample queries from the IAM dataset and top-5 results using our attributes+KCCA approach. Relevant words to the query are outlined in green.

Table 1. First two main columns: retrieval results on the IAM and GW datasets. Last main column: results on the GW dataset when learning is performed solely on the IAM dataset.

	IAM		GW		GW (adapted)	
	QBE	QBS	QBE	QBS	QBE	QBS
FV	14.81	–	63.21	–	63.21	–
Att.	34.32	32.97	69.34	72.32	57.33	34.78
Att. + Platts	45.46	65.01	85.69	90.33	43.69	42.9
Att. + CCA	49.46	70.42	85.85	87.5	63.78	52.84
Att. + KCCA	54.78	71.81	85.63	87.14	61.33	54.29

Table 2. QBE task: comparison with the state-of-the-art.

IAM		GW	
Baseline FV	14.81	Baseline FV	63.21
Exemplar SVM [1]	15.07	Exemplar SVM [1]	65.84
DTW	12.65	DTW [25]	50.0
Character HMM [7, 9]	15.1 / 36.0	SC-HMM [25]	53.0
Proposed (Platts)	45.46	**Proposed (Platts)**	**85.69**
Proposed (KCCA)	**54.78**	**Proposed (KCCA)**	**85.63**

Thanks