Sparse Coding for Face Image Retrieval
r06946003 湯忠憲

1. Baseline for face image retrieval

Given local binary features for each face image, the baseline algorithm directly compares the **l2 distance** between query feature and database features. The MAP could up to 0.0736. Besides, we can also use other distance measurement. Some experiment results are shown in below table.

Table 1. Performance using different metrics

Distance	Mean Average Precision
l2	0.0736
Cosine similarity	0.0982
Dice similarity	0.0918
Correlation	0.0974
Manhattan distance	0.1245
squared Euclidean distance	0.0736

From the above results, we can find out that Manhattan distance perform well in this case.

2. Sparse coding for face image retrieval

Given the patch features, a dictionary for sparse coding ***D*** can learn by solving the following equation. Since there are 80 patches for each image, we have to learn 80 dictionary to encode each part of image.

$$\min_{D,V} \sum_{i=1}^{n} \left\| x_i - Dv_i \right\|^2 + \lambda \left\| v_i \right\|_1$$

$$\text{subject to } \|v_i\| \leq 1, \forall i$$

Once the dictionaries are learned, the encoded sparse vectorαcan be computed by solving the optimization below:

$$\min_{\alpha \in \mathbb{R}^p} \quad ||\boldsymbol{x} - \boldsymbol{D}\alpha||_2^2 \quad \text{s.t.} \quad ||\alpha||_1 \leq \lambda,$$

In this case, single patch feature (59 dimension) would be mapped to a higher or low p dimension sparse feature. The preliminary result is shown below:

Table 2. Hyper parameter for training dictionary and sparse coding

Parameter	Batch size	Dictionary size	Positive constraint	λ for dictionary learning	λ for Lasso optimization
Value	256	100	True	1	1

Note that λ here denotes lamda1 *SPAMS*[1] document, which may different from that those in paper.

[1] http://spams-devel.gforge.inria.fr/doc/html/doc_spams004.html#sec5

Table 3. Performance using different distance metrics

	Mean Average Precision
Overlap (Binarize vector and compute Cos)	0.1255
Cosine similarity	0.0643
l2 distance	0.0086
Dice similarity	0.1256
Correlation	0.0632
Manhattan distance	0.0258

According to the experiment results above, in contrast to soft assignments, it seems that comparing of hard encoding vectors like Overlap and Dice similarity performance well here.

3. Improving sparse coding for face image retrieval

For improvement, I found out that tuning parameter like batch size of iteration, λ or dictionary size would affect the performance.But most of the tunings could not surpass the above performance, i.e. MAP 0.1255, significantly. Therefore, I tried to consider the identity information. [2] (Chen et al., 2011)[2] used identity constraint on sparse coding since vanilla SC may suffer from low recall rate due to high intra-class variance. The adjusted objective function with normalization is shown below:

$$\min_{\hat{v}_i} \beta \frac{\|x_i - D\hat{v}_i\|^2}{\|x_i\|^2} + (1-\beta)\frac{\|V^P - \hat{v}_i 1^T\|_F^2}{m\|v_i\|^2} + \gamma \frac{\|\hat{v}_i\|_1}{\|v_i\|_1}$$

For more detail please refers to the paper.

Motivated by the above method, I directly computed the *mean* feature vector (sparse coding) for image with same identity. Then assign a weight *beta* (similar to βabove) to balance the mean feature vector and original feature vector. Compute their weighted sum to construct the final feature vector. Note that the query feature vector should not go through this modification. After several experiment, I found out we could just retrieval image base on the intra-class mean feature vector (that 0 weight to original feature vector). The result is shown below:

Table 4. Hyper parameter for training dictionary and sparse coding with identity information

Parameter	Batch size	Dictionary size	Positive constraint	λ for dictionary learning	λ for Lasso optimization
Value	100	100	True	1	1

Table 5. MAP of SC+I with different *beta*

beta weight	0.75	0.5	0.25	0.01
Overlap	0.4033	0.4033	0.4033	0.4033
Cosine	0.1150	0.2345	0.4969	**0.7769**

In Table 5, the trend shows that higher weight for *mean* feature vector (lower for original feature vector) produce better result. The performance could reach 0.77 on MAP. In fact, set *beta* to 0.0 could reach about 0.79.

[2] http://cmlab.csie.ntu.edu.tw/~sirius42/papers/mm11.pdf

4. Attribute

LFW provided some auto detected labels, including gender, age, hair style, and so on. As describe in [3] (Chen et al., 2013)[3], automatically detected human attributes may contain semantic cues that help improving content based face retrieval. The idea could be well illustrated using below figure from paper.

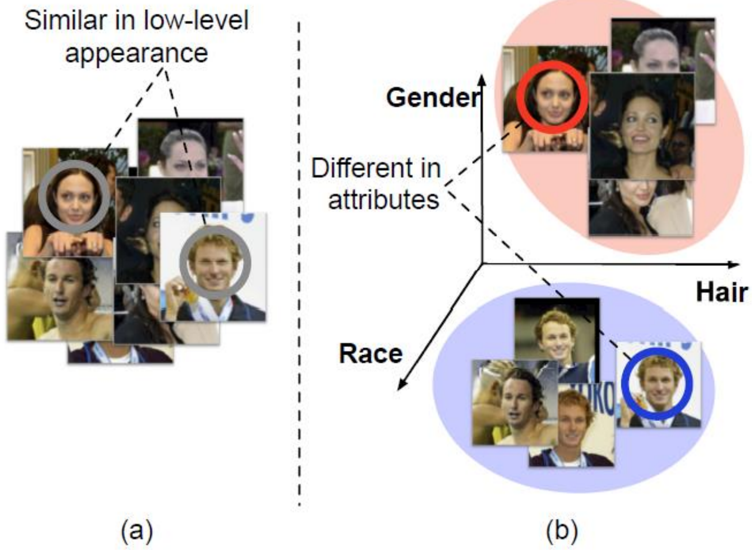

Fig. 1. (a) Because low-level features are lack of semantic meanings, face images of two different people might be close in the traditional low-level feature space. (b) By incorporating high-level human attributes (e.g., gender) into feature representations, we can provide better discriminability for face image retrieval. (Best seen in color)

The adjusted objective function is shown below:

$$\min_{V} \sum_{i=1}^{n} ||x^{(i)} - Dv^{(i)}||_2^2 + \lambda||z^{(i)} \circ v^{(i)}||_1$$

$$z_j^{(i)} = \exp(\frac{d(f_a(i), a_j)}{\sigma}), \quad a_j = \begin{cases} +1, & \text{if } j \geq \lfloor \frac{K}{2} \rfloor \\ -1, & otherwise, \end{cases}$$

Basically, the sparse coding is solved with weighted code-word given specific attribute. Here I try to embed the “Male” attribute to the sparse coding. I assigned the weights *z* with the soft way shown in above formula. First, I create a initial 100 dims vector corresponding to the dictionary size, with -1 in first 50 elements and 1 for last 50. Then we can compute the weights with distance between each {-1, 1} and attribute score. The above problem could also be solved by LARS. When it comes to the result, the method did not give a significant improvement. For example, given the same parameter in Table 2, the MAP is **0.1133** by computing the overlaps of code-word. According to the paper, combine different attribute might improvement the performance. However, the identity feature described in the above part did greatest job I think.

[3] http://cmlab.csie.ntu.edu.tw/~sirius42/papers/mm11.pdf