Part-Based Recognition of Handwritten Characters

Seiichi Uchida
Kyushu University, Japan
uchida@ait.kyushu-u.ac.jp

Marcus Liwicki
Kyushu University, Japan / DFKI, Germany
marcus.liwicki@dfki.de

Abstract

In the part-based recognition method proposed in this paper, a handwritten character image is represented by just a set of local parts. Then, each local part of the input pattern is recognized by a nearest-neighbor classifier. Finally, the category of the input pattern is determined by aggregating the local recognition results. This approach is opposed to conventional character recognition approaches which try to benefit from the global structure information as much as possible. Despite a pessimistic expectation, we have reached recognition rates much higher than 90% for a digit recognition task. In this paper we provide a detailed analysis in order to understand the results and find the merits of the local approach.

1. Introduction

The purpose of this paper is to observe and analyze experimental results of *part-based character recognition*, where each handwritten character image is broken up into a set of small local parts, and then recognized by aggregating the individual recognition results of the parts. Each local part is located at a *keypoint*, which is an important point for representing the shape of the target character. As reviewed below, there are only a few trials on part-based character recognition and thus its characteristics and performance are not well studied.

Since part-based character recognition disregards the global structure of handwritten character, some readers may have a pessimistic expectation on its recognition performance. Our experimental results, however, show that this expectation is too pessimistic. In fact, the recognition rate on handwritten digits can exceed 90% and, moreover, can reach 98% with a certain parameter setting. This shows that there is a large potential for part-based character recognition. To assess all merits of this recognition approach, our current focus is to analyze its characteristics and thus not to achieve the highest possible recognition rate.

This paper is organized as follows. In the remaining of this section, the merits of part-based recognition are emphasized and then a brief review of part-based recognition is provided. In Section 2, the methodology of part-based character recognition is described. Section 3 is devoted to observation and analysis of experimental results from various viewpoints.

1.1. Merits of Part-Based Recognition

We can expect that part-based character recognition has the following unique merits.

- Since it does not rely on the global structure, it is possible to recognize characters which lose their global structure by occlusion, decoration, and other degradations. If a line or curve is drawn on a character, for example, it can still be recognized.
- If each local part is represented by any invariant feature (e.g., scale invariance and rotation invariance), it is not necessary to pay big and careful consideration to some preprocessing, such as scaling and slant correction. In other words, we can recognize characters even if they are difficult to be normalized by preprocessing.
- It is equivalent to the most unconstrained version of image distortion model [1], where each local part is perturbed around its original position for representing deformations. Consequently, it is robust to severe deformations.
- It can be applied directly to cursive scripts for recognizing their component characters. This relaxes the difficulty of segmentation.
- It can also be applied to scenery images for detecting characters in the images. It is well-known that character detection in scenery images is one of the most difficult problems in pattern recognition research. This difficulty arises from the segmentation problem, i.e., it is not easy to detect the objects in the image. Again, part-based recognition will relax the difficulty of the segmentation and thus will be a promising strategy of the detection problem.

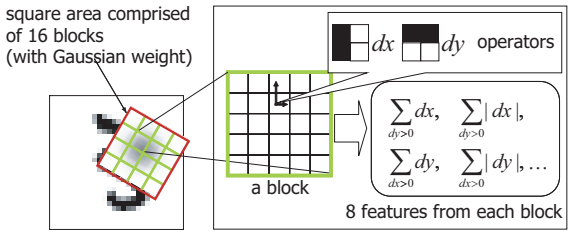

Figure 1. Describing a local part as a 128-dimensional SURF feature vector.

1.2. Related Work

Nowadays, computer vision researchers develop many part-based recognition methods [2] for recognizing visual objects, such as motorbikes, lions, airplanes, etc. In those methods, local parts are detected and described by, for example, scale-invariant feature transform (SIFT) [3] or speeded-up robust features (SURF) [4]. SURF detects keypoints (i.e., location of local parts) as local maxima of approximate Hessian values in scale space. Then, SURF describes each local part as a 64 or 128-dimensional feature vector. The element of the vector is a local directional feature value. It should be noted that the resulting feature vector becomes *rotation and scale invariant*, by adapting the orientation and the size of each local part automatically.

By considering this trend in computer vision, it seems very worthy to try part-based character recognition. Furthermore, part-based character recognition is reasonable to the mechanism of human reading. For example, Schomaker and Segers [5] have pointed out that local parts, such as crossings, line endings, and curvatures, play an important role in human reading. Similarly, Avallone et al. [6] have pointed out that ascenders and descenders are processed first in human reading.

However, only little attention has been paid to part-based recognition of handwritten characters. The feature vectors employed in handwritten character recognition always represent the global structure of characters, explicitly or implicitly [7]. (Even bitmap represents the global structure because the location of each pixel is fixed always.) Character-SIFT [8] seems to have some relation to part-based recognition, but it uses keypoints at every dense regular grid points on the character image; thus, it is not truly part-based. Diem and Sablatnig [9] have proposed a part-based character recognition for historical manuscripts; unfortunately, it was a limit trial just showing rather low recognition accuracy. This paper will show that if we have enough reference keypoints, we can expect far better performance.

2. Part-Based Character Recognition

The part-based character recognition method is organized in a two-step manner, that is, a training step and a recognition step. In the following, those steps are detailed, while assuming a recognition problem of isolated handwritten characters and employing SURF [4] for detecting and describing local parts. (Note that any other method can be used instead of SURF.)

2.1. Training Step

First, keypoints are detected from each training pattern by using the SURF keypoint detector. Since the detector is based on the local maximum of approximate Hessian, keypoints are often located around corners and curves of character stroke. The number of keypoints from one training pattern varies and depends on the shape of the pattern. (In the following experiment, about 60 keypoints were detected in average.)

Second, a square area around each keypoint is described as a 128-dimensional SURF feature vector (*reference vector*), and stored into a database (*dictionary*). As shown in Fig. 1, the reference vector is a kind of directional feature of the square area. The orientation and the size of the square area are determined automatically so that the reference vector becomes rotation and scale invariant. It is important to note that since SURF employs a Gaussian weight on its feature vector, the truly effective area is narrower than the square area.

In parts of our experiments we intentionally fix the orientation at 0° and the scale of the square area to a parameter s, respectively. In this case, the feature vector is neither rotation nor scale invariant.

2.2. Recognition Step

The recognition step is further decomposed into two sub-steps, that is, *feature-level recognition* and *character-level recognition*. At feature-level recognition, each SURF feature vector of the input pattern is recognized by using the Euclidean 1-nearest-neighbor (1NN) rule against all the reference vectors in the dictionary. Consequently, if J feature vectors are extracted from one input pattern, we have J recognition results at this sub-step. Then, character-level recognition is performed for determining the input pattern category by the majority voting of J recognition results. It should be emphasized that the original locations of the feature vectors are totally disregarded.

3. Experiments

3.1. Dataset

As our dataset, 20,000 samples were extracted from the "training" dataset of MNIST handwritten digit

Table 1. Result of local part detection.

		category										
		0	1	2	3	4	5	6	7	8	9	total
#test patterns		1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	10,000
#local parts	total	78,928	38,506	61,546	60,121	57,144	60,071	61,823	54,756	61,867	56,286	591,048
	ave/pattern	78.9	38.5	61.5	60.1	57.1	60.1	61.8	54.8	61.9	56.3	59.1
	max/pattern	123	83	99	101	102	107	112	95	104	94	123
	min/pattern	37	12	32	30	30	31	29	24	32	25	12

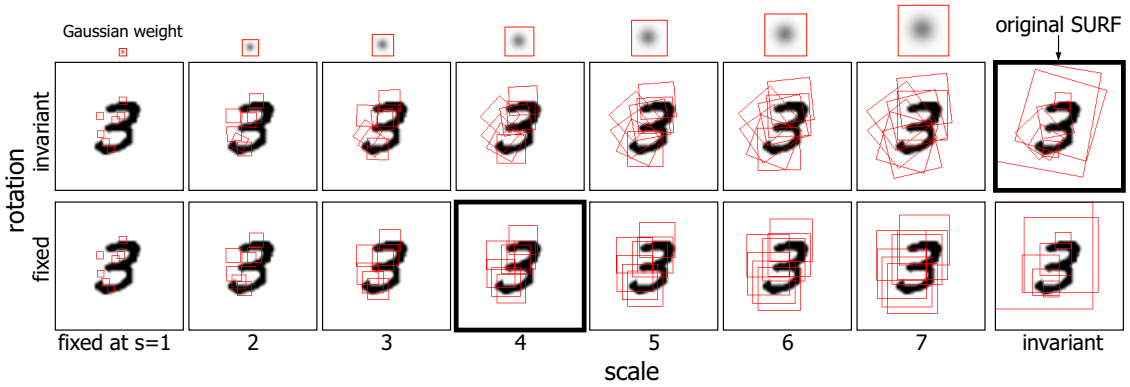

Figure 2. Local parts by SURF. Only 7 local parts (selected randomly) are shown.

database [10]. For each category, the 2,000 samples were divided into 1,000 training patterns and 1,000 test patterns. We used only 1,000 samples for training patterns of a category due to computational limitation, although more training patterns will provide a better recognition rate as shown later.[1]

Each sample (a 28×28 grayscale image) was preprocessed so that enough number of local parts were extracted by SURF. Specifically, it was magnified four times after the addition of 10-pixel surrounding margin. Consequently, the sample became a 192×192 image.

3.2. Training Results

Table 1 shows the statistics of the dictionary. The dictionary was comprised of about 59,000 reference vectors. Since there were 1,000 training patterns, the average number of local parts per training pattern was 59. This number increases on "0" and decreases on "1".

Figure 2 shows examples of detected local parts. The original SURF feature sometimes covers the entire character. Thus, this case should be considered as an exceptional case of the "part-based" recognition.

If we fix the scale by a parameter s, we can realize a "strictly part-based" recognition. For example, when $s = 4$, the size of the square area is always about 1/3 of the character size. Note again, the effective area of the local part becomes narrower if we consider the Gaussian weight, which is also shown in Fig. 2. For example, the effective area becomes about 1/20 of the whole character size at $s = 4$. Hereafter, we consider the condition of $s = 4$ and the fixed rotation as our representative condition of part-based recognition.

[1]The recognition rates of MNIST by up-to-date methods are listed in [10].

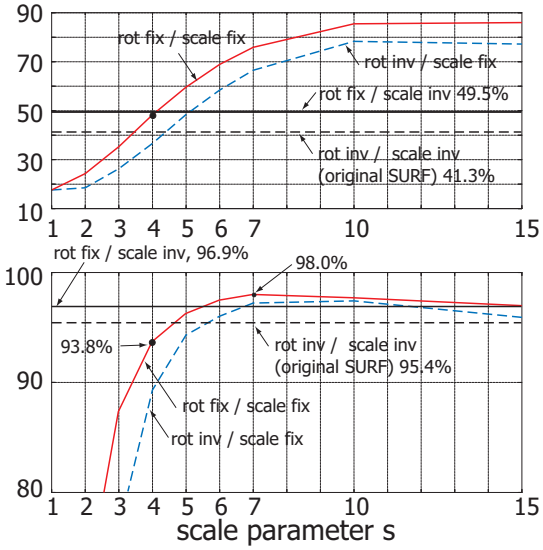

Figure 3. Recognition rates in % feature-level (upper) and character-level (lower).

3.3. Feature-Level Recognition

The feature-level recognition rates, i.e., the rates of the 1NN reference vector belonging to the category of the input pattern, are shown in Fig. 3. It is observed that the feature-level recognition rates are quite low. Note that

Table 2. Confusion matrix (%). (Rot:fix, scale:$s = 4$.)

(a) feature-level recognition rate in %

input \ recognition result	0	1	2	3	4	5	6	7	8	9
0	52.5	3.5	8.0	5.9	2.6	5.9	8.4	4.4	4.1	4.7
1	7.0	48.6	1.8	0.5	**10.0**	0.7	7.6	**16.5**	0.8	6.5
2	9.9	1.2	46.3	9.3	3.7	5.8	5.0	8.1	5.9	4.8
3	8.2	0.4	8.7	47.6	1.4	**14.4**	2.8	5.7	7.4	3.3
4	5.5	8.7	5.0	1.2	47.5	2.1	7.5	7.4	3.0	**12.0**
5	8.9	0.7	6.0	**16.3**	2.6	46.9	5.0	3.6	6.6	3.5
6	**11.3**	4.9	4.9	3.7	5.6	4.3	52.2	2.3	6.9	3.9
7	6.0	**12.3**	8.8	6.1	7.0	3.8	2.6	41.8	2.2	9.2
8	5.8	0.7	6.5	8.1	2.8	5.9	6.7	2.6	52.5	8.4
9	6.6	5.0	5.4	4.1	9.9	3.2	3.9	9.0	8.8	44.3

(b) character-level recognition rate in %

input \ recognition result	0	1	2	3	4	5	6	7	8	9
0	98.0	0.0	0.4	0.5	0.1	0.2	0.5	0.0	0.2	0.1
1	1.0	91.8	0.4	0.0	0.5	0.1	0.9	**4.4**	0.0	0.9
2	1.7	0.1	95.9	0.6	0.0	0.3	0.5	0.8	0.1	0.0
3	1.2	0.1	0.9	95.0	0.0	1.9	0.0	0.5	0.3	0.1
4	0.8	0.6	0.0	0.0	94.1	0.0	0.4	0.3	0.1	**3.7**
5	1.3	0.1	0.1	2.5	0.1	94.7	0.8	0.0	0.2	0.2
6	2.9	0.3	0.1	0.0	0.0	1.0	95.3	0.0	0.4	0.0
7	2.2	**3.8**	2.8	0.3	2.8	0.1	0.0	87.5	0.0	0.5
8	0.7	0.0	0.1	2.1	0.3	0.1	0.3	0.0	96.0	0.4
9	**3.5**	0.4	0.4	0.5	2.3	0.7	0.2	1.0	1.3	89.7

Table 3. Distribution of referred times of each reference vector. (Rot:fix, scale:$s = 4$.)

#cases selected as 1NN of the correct category \ #cases selected as 1NN of an incorrect category	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	17
0	282,184	82,580	28,505	9,920	3,562	1,291	471	160	87	35	15	1	2	0	1	0
1	75,380	26,293	10,230	3,799	1,514	554	217	95	34	16	4	1	2	1	0	0
2	25,357	9,042	3,656	1,412	571	222	110	50	15	4	3	1	0	0	0	1
3	9,019	3,200	1,386	581	227	88	34	24	6	4	3	0	0	1	0	0
4	3,460	1,217	527	243	84	40	14	4	3	1	0	0	0	0	0	0
5	1,294	407	183	70	42	25	7	7	2	0	0	0	0	0	0	0
6	517	156	88	33	15	10	1	0	0	1	0	0	0	0	0	0
7	233	64	36	15	9	1	0	0	0	0	0	0	0	0	0	0
8	84	29	13	6	4	1	4	0	0	0	0	0	0	0	0	0
9	38	16	4	1	2	3	2	0	0	0	0	0	0	0	0	0
10	18	5	2	3	2	0	0	0	0	0	0	0	0	0	0	0
11	5	4	1	1	0	0	0	0	0	0	0	0	0	0	0	0
12	5	3	1	0	0	0	0	0	0	0	0	0	0	0	0	0
13	2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	1	2	1	0	0	0	0	0	0	0	0	0	0	0	0
15	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

only 50% of the feature vectors were correctly recognized in the case of the fixed rotation and the fixed scale ($s = 4$). This fact proves that local parts from different categories often resemble each other.

The difficulty of the feature-level recognition is also illustrated in Fig. 4, which shows the distribution of 1NN distances of each feature vector to the correct category and that to the nearest incorrect category. There is a concentration along the diagonal line and thus the minimum distance to the incorrect category is often close to that to the correct category.

Figure 5 shows examples for correspondences between local parts. Note that the correspondences are reasonable because they are established between similar local parts. It is also observed that local parts from different categories are often very similar and thus distributed with considerable overlaps.

Table 2(a) shows the confusion matrix of feature-level recognition. Major misrecognition pairs (printed in boldface) were "1"↔"7" and "3"↔"5." This is simply because, for example, the lower parts of "3" and "5" are often similar. It is interesting to note that there is no zero entry in the confusion matrix.

Table 3 shows the distribution of referred times of each reference vector, i.e., how often it appeared to be the 1NN. This table indicates that (i) 47.7% reference vectors were never referred to, (ii) 74.5% were referred to at most one time, (iii) 21.4% were always chosen incorrectly, (iv) 19.5% were always chosen correctly, and (v) maximum reference times were 17. The fact (i) indicates that we can halve the dictionary size without loosing important information (and (ii) indicates that 1/4 size might be still useful, respectively). The fact (v) indicates that there is no "notorious" reference vector which causes many misrecognitions.

As shown in Fig. 6, the number of training patterns at each category affects the recognition rates drastically. In the extreme case, that is, if we use only a single training pattern for each category, feature-level recognition rate was degraded to 30%. This also proves that the local parts are distributed with considerable overlaps and thus we need many reference vectors for increasing the probability of finding a 1NN of the correct category. Note that the recognition rate was not saturated with 1,000 training patterns and thus will be improved if we use more training patterns.

Figure 7 visualizes results of feature-level recognition. Color indicates the recognized category. Accord-

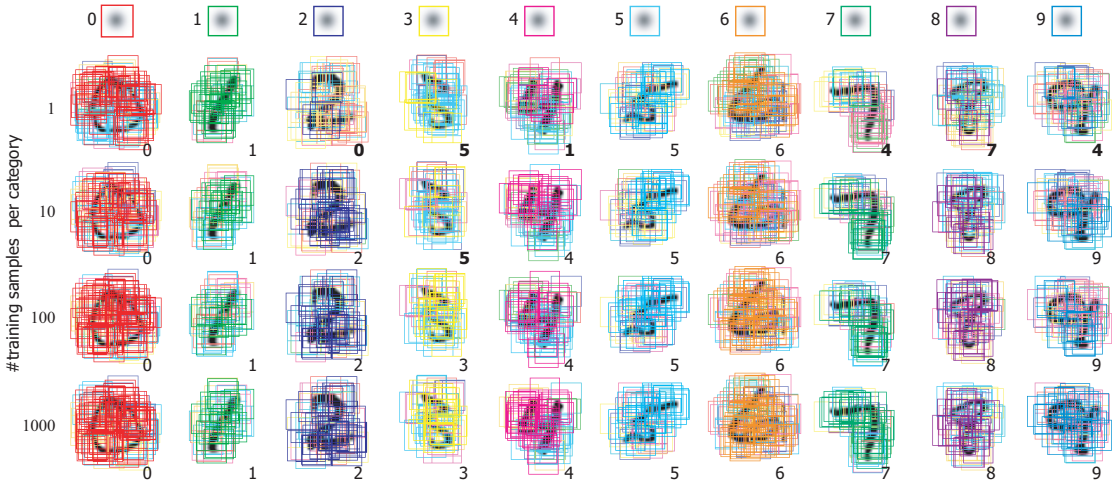

Figure 7. Feature-level recognition result. Better viewed in color. (Rot:fix, scale:$s = 4$.)

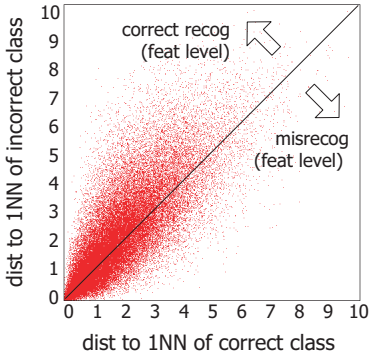

Figure 4. Distribution of 1NN distance. (Rot:fix, scale:$s = 4$.)

Figure 5. Correspence between local parts. (Rot:fix, scale:$s = 4$.)

ing to the increase of the training patterns, a convergence of the colors can be observed; that is, most local parts are recognized correctly. Several parts, such as the lower part of "4", are still misrecognized.

3.4. Character-Level Recognition

In the lower part of Fig. 3 the character-level recognition rates are shown. The original SURF feature achieved 95.4%. When the scale was fixed at a small number, a lower recognition rate was obtained. However, it was still beyond our expectation that we could

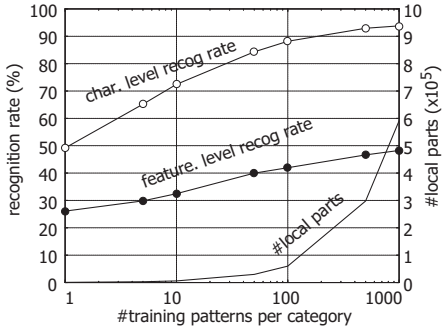

Figure 6. Recognition rate as a function of training set size. (Rot:fix, scale:$s = 4$.)

have 93.8% at $s = 4$ (and the fixed rotation) [2]. Since each local part is about 1/20 of the character size for $s = 4$, this result is very positive about the part-based character recognition method. The highest performance of 98% was achieved at $s = 7$ and the fixed rotation. In this case the effective size is about 1/7 of the character size.

Table 2(b) shows the confusion matrix of character-level recognition. The best rate was 98.0% for "0" and the worst was 87.5% of "7", respectively. The pair "1"↔"7", which was a major misrecognition pair at feature-level, was also a major misrecognition pair at character-level. In contrast, another notorious pair "3"↔"5" was not a major misrecognition pair anymore; this is because their dissimilar upper parts allow a better discrimination. It is also noteworthy that categories with circular strokes (e.g., "2", "6", "9") were misrecognized as "0".

In Fig. 8, all test patterns are projected on a two-dimensional plane according to their v_{cor} and v_{incor} val-

[2] When the 10,000 samples of the official "test" dataset of MNIST were used for final evaluation, the recognition rate of this case was 93.6%. That is, there was no significant difference.

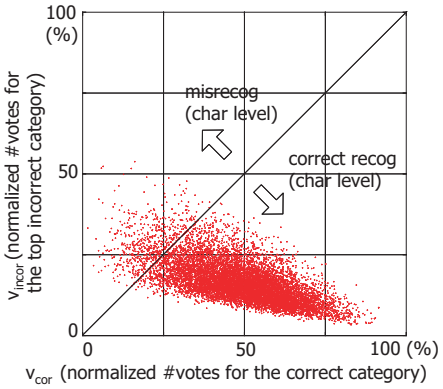

Figure 8. Distribution of v_{cor} and v_{incor}. (Rot:fix, scale:$s = 4$.)

Table 4. Relation between #local parts and character-level recognition accuracy. (Rot:fix; scale:$s = 4$.)

#local parts	#correct	#incorrect	rate(%)
10∼19	6	0	100
20∼29	234	5	98
30∼39	603	49	92
40∼49	1554	155	91
50∼59	2501	195	93
60∼69	2309	123	95
70∼79	1250	54	96
80∼89	613	23	96
90∼99	248	14	95
100∼109	56	1	98
110∼119	5	1	83
120∼129	1	0	100

ues (%). The former is the normalized number of votes to the correct category at the character-level recognition. The latter is the normalized number of votes to the top incorrect category. If $v_{\mathrm{cor}} > v_{\mathrm{incor}}$, a test pattern is correctly recognized at character-level recognition by majority voting. Since the feature-level recognition rate was around 50%, the peak of the distribution of v_{cor} is also around 50%. A more important thing is that the peak of v_{incor} is far lower and around 17%. This implies that the misrecognitions at the feature-level recognition were not converged into a certain incorrect category but scattered into various incorrect categories. According to this fact, character-level recognition achieves recognition rates higher than 90% by majority voting.

Table 4 shows the relation between the number of local parts of the input pattern and character-level recognition accuracy. When we disregard extreme cases (with fewer input patterns), we can observe a trend that input patterns with more local parts have more correct recognition results. This will be because more votes will increase the reliability of the result of majority voting.

4. Conclusion and Future Work

It seems that character recognition researchers have had a common sense that the global structure of each character is very essential for recognition. However, the experimental results presented in this paper may somewhat release the researchers from this common sense. In the proposed method, small local parts (about 1/20 of the character size) are first recognized independently. This step is called feature-level recognition. Then, the recognition results are aggregated by majority voting. This step is called character-level recognition. Although the accuracy of the feature-level recognition is quite low (around 50%), we could achieve 93.8% recognition accuracy by the character-level recognition.

Future work is to apply part-based recognition to some application where its merits listed in Section 1.1 are fully utilized. Furthermore, a more sophisticated method for aggregating the feature-level results will be investigated. For example, we could sum up the distances to the nearest neighbor of each category and finally use the category with the lowest aggregated sum [11]. Or we could use the bag-of-keypoints approach, where each feature vector undergoes a quantization process to be represented as a visual word.

References

[1] D. Keysers, T. Deselaers, C. Gollan, H. Ney, "Deformation Models for Image Recognition," IEEE Trans. PAMI, vol. 29, no. 8, pp. 1422-1435, 2007.

[2] L. Fei-Fei, R. Fergus, A. Torralba, "Recognizing and Learning Object Categories," `http://people.csail.mit.edu/torralba/shortCourseRLOC/index.html`

[3] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," IJCV, vol. 60, no. 2 pp. 91–110, 2004.

[4] H. Bay, T. Tuytelaars, and L. V. Gool, "SURF: Speeded Up Robust Features," Proc. ECCV, 2006.

[5] L. Schomaker and E. Segers, "Finding Features Used in the Human Reading of Cursive Handwriting," IJDAR, vol. 2, no. 1, pp. 13–18, 1999.

[6] L. Avallone, C. De Stefano, C. Gambone, and A. Marcelli, "Visual Processes and Features in Human Reading of Cursive Handwriting," Proc. 12th Conf. Int. Graphonomics Soc., pp. 128–132, 2009.

[7] Ø. D. Trier, A. K. Jain, and T. Taxt, "Feature Extraction Methods for Character Recognition – A Survey," Pattern Recognit., vol. 29, no. 4, pp. 641–662, 1996.

[8] Z. Zhang, L. Jin, K. Ding, and X. Gao, "Character-SIFT: A Novel Feature for Offline Handwritten Chinese Character Recognition," Proc. ICDAR, pp. 763–767, 2009.

[9] M. Diem and R. Sablatnig, "Recognition of Degraded Handwritten Characters Using Local Features," Proc. ICDAR, pp. 221–225, 2009.

[10] `http://yann.lecun.com/exdb/mnist/`

[11] O. Boiman, E. Shechtman, M. Irani, "In Defense of Nearest-Neighbor Based Image Classification," Proc. CVPR, 2008.