

Texture Image Classification using Extended 2D HLAC Features

Motofumi T. Suzuki¹

¹ *The Open University of Japan, Japan, motofumi@ouj.ac.jp*

Abstract: HLAC (Higher Order Local Autocorrelation) features are popular image descriptors that have been used for various image-processing applications since the 1980s. Examples of the application of the HLAC features include KANSEI retrievals and subjective retrievals of 2D image databases. In this paper, standard HLAC masks are extended for computing a massive number of features. Typical HLAC features are computed by applying 25 masks to a binary image, whereas our Ext-HLAC features are computed by applying 16,241,567 masks. Since there are a high number of mask combinations, we have developed Ext-HLAC mask generation software programs. Ext-HLAC masks were tested by using 2D benchmark image database sets. For each image, the pattern features were extracted by applying Ext-HLAC masks, and the pattern features were analyzed by a k-NN based approach. Our preliminary experiments show high classification rates for certain image databases.

Keywords: HLAC, Ext-HLAC, pattern feature, k-NN, image classification

1. HLAC MASKS

HLAC (Higher Order Local Autocorrelation) features (Otsu and Kurita, 1998) are popular pattern descriptors that have been used for many image-processing applications since the 1980s. Examples of the application of the HLAC features include KANSEI retrievals and subjective retrievals of 2D image databases (Kato, Kurita, and Shimogaki, 1989) (Kato, 1992) (Kurita and Otsu, 1993). HLAC masks are used for extracting HLAC features from images. An HLAC mask consists of 3x3 (9) cells, and a set of standard HLAC masks for analyzing binary images consists of 25 masks as shown in Figure 1. Figure 2 shows the processes for extracting HLAC features. During the first process, a source image is converted into a binary image. HLAC masks are applied to the binary images, and an HLAC image is obtained as output. HLAC features are

computed by counting the number of black pixels and white pixels, and the number of pixels is normalized by the image sizes. Each HLAC mask produces one feature; thus if there are 25 masks, 25 HLAC features are computed. The computations of HLAC features involve simple multiplications and summations of pixel values; thus, the computations are efficient compared to other image feature extraction techniques that require slow complex functions.

order	# of masks	mask patterns (25)
0	1	
1	4	
2	20	

Figure 1: HLAC (3x3) (n=0, 1, 2)

Once the image features are extracted from the image, these features are analyzed by pattern recognition related techniques, and image features are used for various applications, including object recognition, face detections, image retrievals, image segmentations, and image classifications. There are several pieces of research that are related to the extension of the HLAC masks. Toyoda and Hasegawa (2005) introduced extended HLAC (3x3). In their research, the order of the autocorrelation functions are used up to the 8th order, and the number of HLAC masks is increased to 223, whereas standard HLAC uses 25 masks. Since the extended HLAC (3x3) mask can describe more pattern features for images, the research improved texture classification rates. Additionally, Toyoda and Hasegawa (2005) introduced large-size mask patterns for extracting multi resolution feature extractions. Suzuki, Yaginuma, Yamada and Shimizu (2006) introduced 3D HLAC (3x3x3). During research, two-dimensional HLAC masks are extended to three-dimensional HLAC masks that can handle not only 2D image data, but also 3D data, including movie data, 3D polygonal data, and 3D volumetric data. An example of extended HLAC masks (3x3) is shown in Figure 3, and an example of 3D HLAC masks (3x3x3) is shown in Figure 4

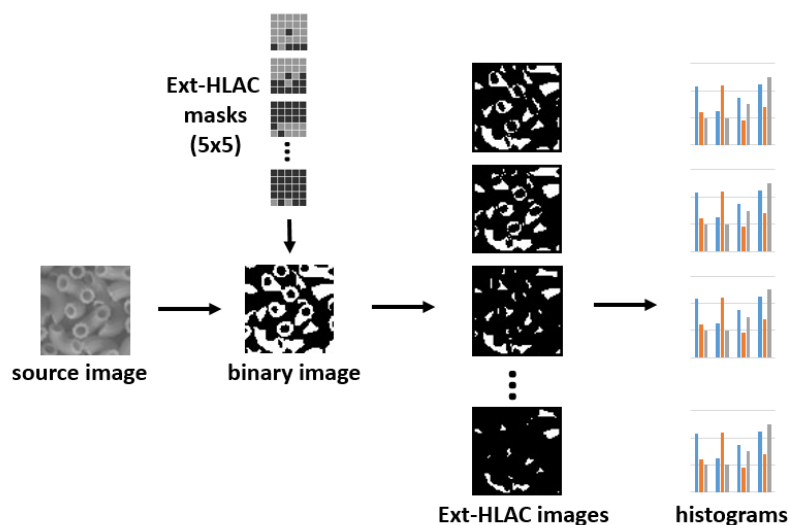


Figure 2: Processes for extracting HLAC features


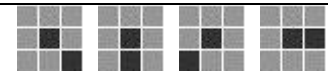
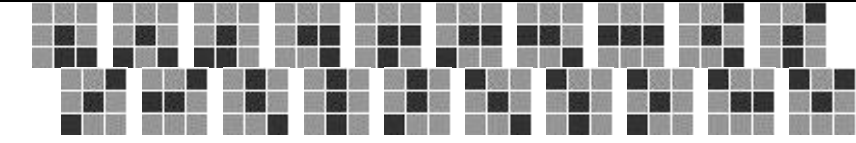
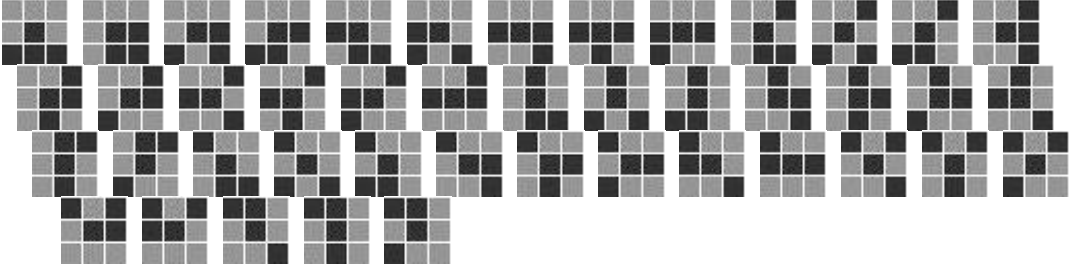
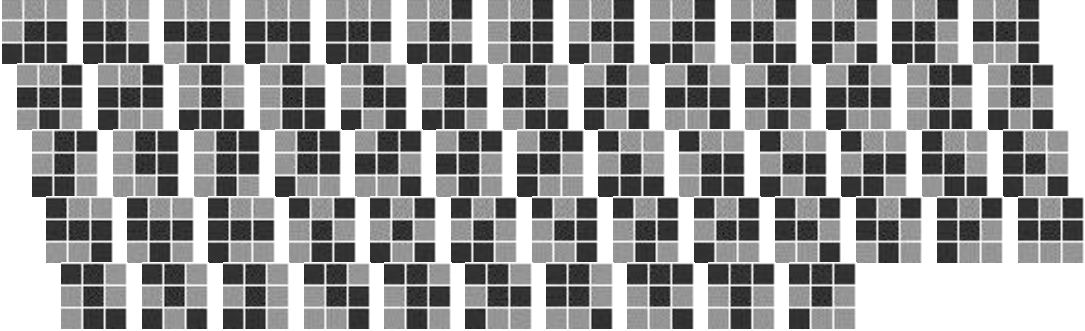
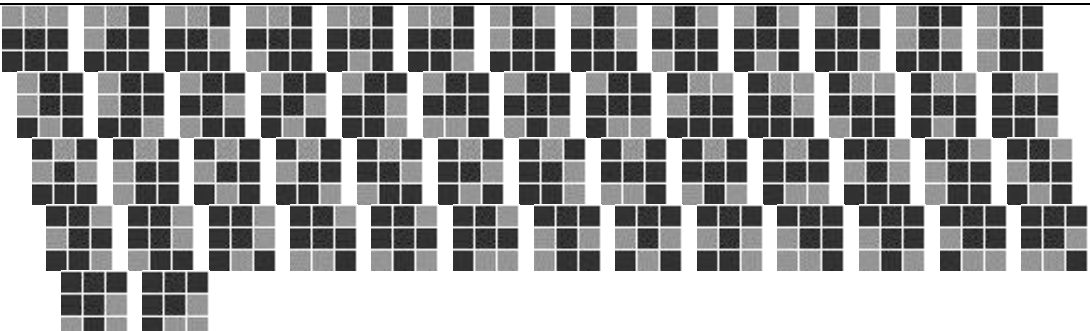
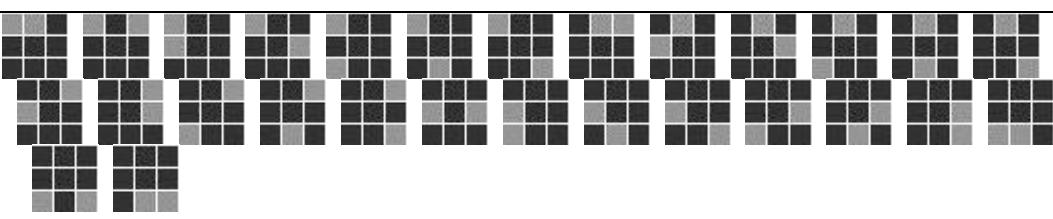


n	#	mask patterns (223)
0	1	
1	4	
2	20	
3	45	
4	62	
5	54	
6	28	
7	8	
8	1	

Figure 3: Extended HLAC (3x3) (n=0,1,2,3,4,5,6,7,8) Toyoda-Hasegawa model

2. EXT-HLAC MASK (5X5)

A standard HLAC mask consists of 3x3 (9) cells. In this research, the number of cells was increased so that an Ext-HLAC (i.e. Extended HLAC) mask would consist of 5x5 (25) cells. These minor cell-size extensions cause the Ext-HLAC to have numerous combinations of the masks. For instance, there are 223 masks for a 3x3 case, but there are 16,241,567 masks for a 5x5 case. Therefore, a software program is necessary for computing these combinations of masks. The program lists all the combinations of mask patterns (16,777,216), and each mask is compared if there are redundant masks in terms of shift invariance. The redundant masks are eliminated from the list, and finally the program determines 16,241,567 masks. These processes are time and memory consuming tasks. The program requires about 16 minutes to complete the processes using a standard desktop computer (Intel Core i7-3930K 3.20 GHz), and over 2.5GB of dynamic memory were used during the execution of the program.

In the program, two-dimensional Ext-HLAC masks are represented as 5x5 arrays, and they are converted into a one-dimensional array with a length of 25. They are treated as a set of binary numbers with a 25-bit width, and each binary number of combinations is represented by 0 and 1. Since these binary numbers have a 25-bit width, there are 2 to the 25th power (33,554,432) of combinations. Since the Ext-HLAC feature concerns the relationship between the center cell and its surrounding 24 neighboring cells, there are in fact 2 to the 24th power (16,777,216) of combinations; however, our program uses a list of 2 to the 25th power of combinations for efficient hashing processes. The program is required to compare these binary numbers with each other, and eliminate redundant binary numbers. The comparisons of binary numbers involve an exhaustive, time-consuming search for a list. For instance, when a simple “linear search” was used for this task, it took over 7.5 days to perform comparisons, whereas a “binary search” took only 30 minutes. In our program, a hashing technique was used for this task to further speed up the process. Since the binary numbers are easily converted to decimal numbers, the program uses these decimal numbers as hash keys for accessing the list. This makes our hash-table based program run quickly, as compared to a linear search based techniques, and comparison of the list using the hashing technique was performed in less than 16 minutes. However, use of the hashing technique requires allocation of a huge amount of dynamic memory during the execution of the program, and over 2.5GB of memory were used in this case.

Figure 5 shows a partial set of Ext-HLAC (5x5) generated from the program. The set of Ext-HLAC can be categorized based on the order of autocorrelation functions. As shown in Figure 5, there are more black cells for the higher order of the autocorrelation functions. Although the limited number of masks is shown in Figure 5, an actual set of the Ext-HLAC contains a high number of masks (16,241,567).

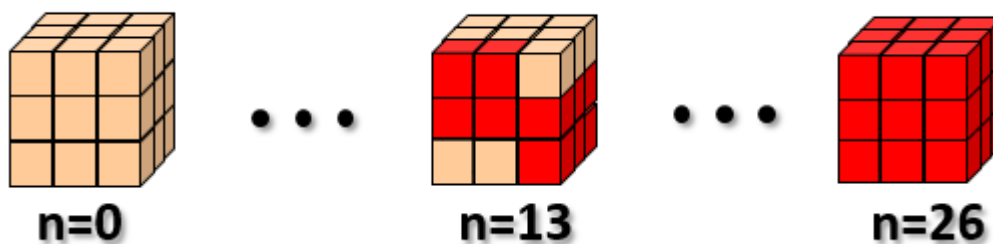


Figure 4: Example of 3D HLAC masks (3x3x3)

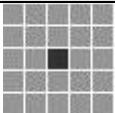

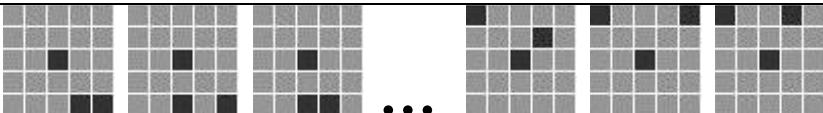
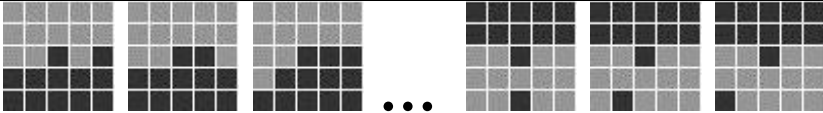

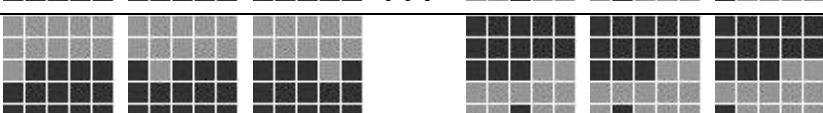
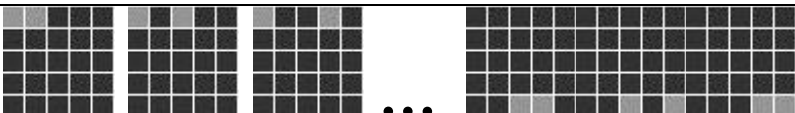

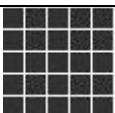
n	#	mask patterns (16,241,567)
0	1	
1	12	
2	180	
3	1449	...
4	8182	...
5	34662	...
6	114804	...
7	306024	...
8	669571	...
9	1218966	...
10	1863932	...
11	2408859	
12	2640680	
13	2459078	
14	1944132	...
15	1301385	...
16	733839	...
17	345798	...
18	134560	...
19	42502	...
20	10626	...
21	2024	...
22	276	
23	24	
24	1	

Figure 5: Ext-HLAC (5x5) (n=0,1,2, ..., 22, 23, 24) (limited number of masks is shown in Figure)

The sample set of the Ext-HLAC mask data is downloadable from the following web address (<http://goo.gl/87FgUr>). These Ext-HLAC mask data are represented in decimal number to reduce the file sizes. Since each Ext-HLAC mask consists of a set of cells, these cells can be represented in a sequence of binary numbers. Each sequence of binary numbers (25bits) is converted into a single decimal number as shown in Figure 6.

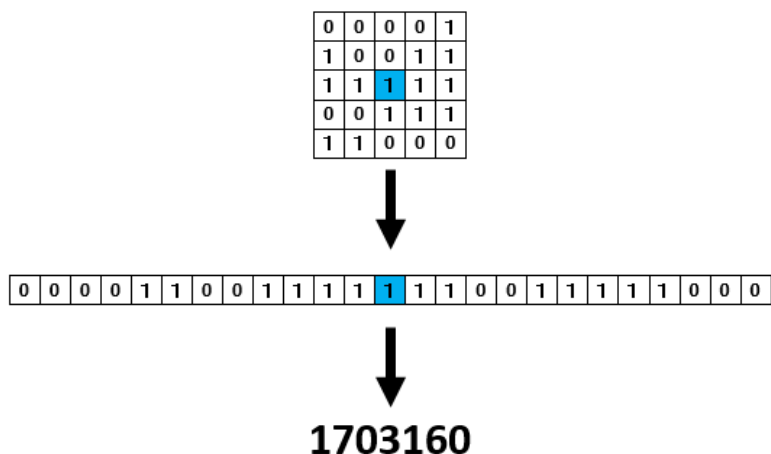


Figure 6: Decimal number representation of Ext-HLAC masks

Ext-HLAC masks shown in Figure 5 are applied for a binary image. Original source images must be converted before the Ext-HLAC feature extraction processes. In the experiment, the P-tile-threshold method was used for binary image conversion. Ext-HLAC masks were applied to each image, and each mask produces one output image (Ext-HLAC image) as shown in Figure 7. Since each input image is a binary image, each Ext-HLAC image is also outputted as a binary image. Ext-HLAC features are computed from the Ext-HLAC images, and the computation involves counting the number of pixels in the Ext-HLAC images.

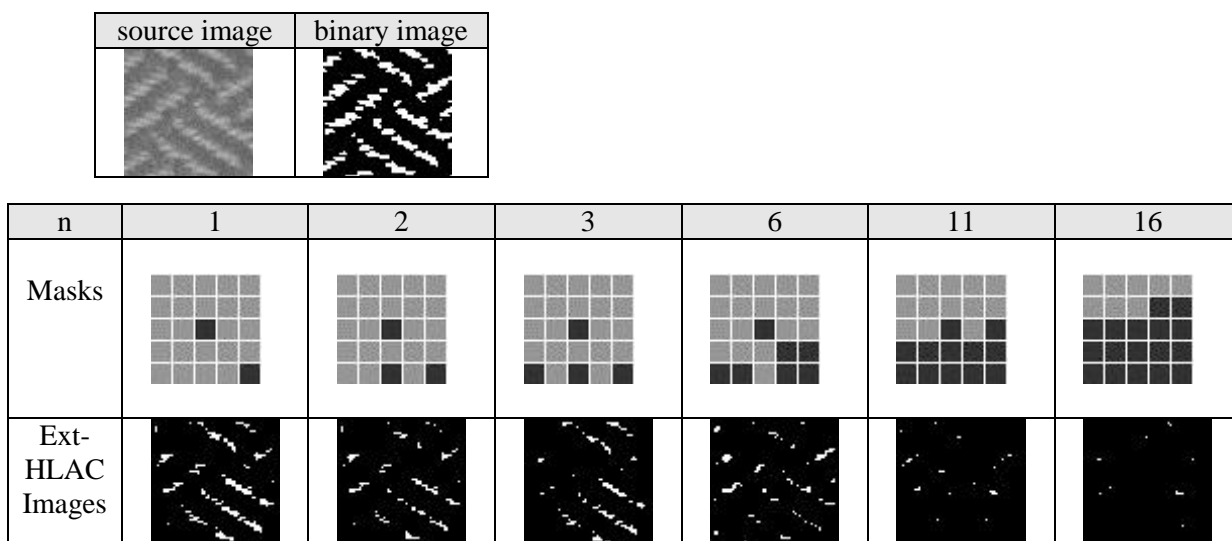


Figure 7: Ext-HLAC masks (5x5) and Ext-HLAC images

3. EXPERIMENTS

3.1. Experimental data sets

Portions of OUTEX (Ojala et al., 2001) texture benchmark data sets were used to test Ext-HLAC masks. Ext-HLAC features were extracted for each image, and features were compared by classification methods. In the experiment, simple k-NN (k Nearest Neighbor) techniques were used for the benchmark image data.

3.2. Classifications of two types of images

The portion of OUTEX data sets (“barley_rice_011” and “canvas_001”) were chosen for our classification experiments, and each data set contains 20 images. An example of these 2 data sets is shown in Figure 8.

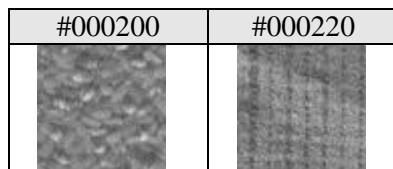


Figure 8: Image data (“barley_rice_011” and “canvas_001”)

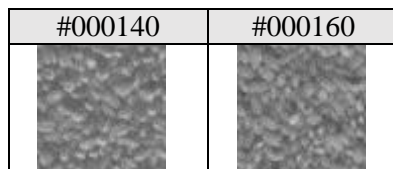


Figure 9: Image data (“barley_rice_008” and “barley_rice_009”)

In the experiment, the OUTEX data set was converted to PGM (Portable Grey Map) format. All image sizes are adjusted to 128x128 with color depth 16 (65536 colors). In the experiments, Ext-HLAC masks are tested, while groups of Ext-HLAC masks are used to efficiently classify test image data sets. Two experiments were conducted (A) if relatively dissimilar image data sets were classified, such as images “barley_rice_011” and “canvas_001,” and another experiment was conducted (B) of relatively similar image data sets, such as “barley_rice_008” and “barley_rice_009”. Samples images are shown in Figure 9.

Ext-HLAC features were categorized into 25 groups based on the order of the autocorrelation functions. For example, there are 180 masks related to the order of the autocorrelation functions with $n=2$. The number of masks increases from $n=0$ through $n=12$, and reaches the maximum of 2,640,680 masks. Since it is difficult to use so many masks for computing Ext-HLAC features, a small number of masks were randomly selected. In the experiment, 25 masks were selected for $n=2$ through $n=22$.

Table 1 shows experimental results of (A) and Table 2 shows experimental results of (B). In these tables, the number of Ext-HLAC masks, the number of features used, and classification rates are shown in each row. The k-NN approach with a value of k=3 was used, and a total of 40 images were examined. Cross-validation (10 folds) was used for experiments where data was split into 10 folds, trained on 9 folds 10 times, and tested on the remaining one. For Experiment (A), as shown in Table 1, image classification rates are high at n=4 and n=6 ...12 where n represents the order of the autocorrelation function. In Experiment (B), image classification rates are high at n=3 and n=4 as shown in Table 2.

Table 1: Dissimilar image data sets (Ext-HLAC (5x5))

N	0	1	2	3	4	5	6	7	8	9	10	11	12
# of features	1	12	25	25	25	25	25	25	25	25	25	25	25
Classification rates (%)	62.5	75.0	87.5	85.0	95.0	92.5	95.0	95.0	95.0	95.0	95.0	95.0	95.0
N	13	14	15	16	17	18	19	20	21	22	23	24	
# of features	25	25	25	25	25	25	25	25	25	25	24	1	
Classification rates (%)	92.5	92.5	92.5	92.5	90.0	90.0	87.5	90.0	82.5	80.0	77.5	65.0	

Table 2: Similar image data sets (Ext-HLAC (5x5))

N	0	1	2	3	4	5	6	7	8	9	10	11	12
# of features	1	12	25	25	25	25	25	25	25	25	25	25	25
Classification rates (%)	97.5	95.5	97.5	100.	100.	97.5	97.5	97.5	97.5	95.0	95.0	95.0	95.0
N	13	14	15	16	17	18	19	20	21	22	23	24	
# of features	25	25	25	25	25	25	25	25	25	25	24	1	
Classification rates (%)	95.0	95.0	95.0	95.0	95.0	95.0	95.0	95.0	92.5	95.0	95.0	95.0	

3.3. Classifications of 5 types of images

In this experiment, classifications of 5 types of images were conducted as shown in Figure 10. Experimental data conditions were similar to previous subsections (3.2), except more data sets were used. The portion of OUTEX data sets (“wall_paper_002” through “wall_paper_006”) were chosen for our classification experiments, and each data set contains 20 images. Thus, there were a total of 100 images for classification tests. As shown in Table 3, classification rates were slightly low compared to previous experiments (3.2).

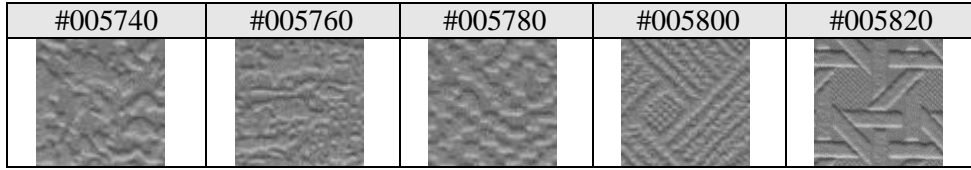


Figure 10: Image data (wall_paper_002, 003, 004, 005 and 006)

3.4. Comparison between standard HLAC (3x3) and Ext-HLAC (5x5)

In this experiment, traditional standard HLAC (3x3) and Ext-HLAC (5x5) were compared using an image data set as described in the previous subsection (3.3). A total of 25 HLAC (3x3) features were used for classifications. In this case, HLAC features with order of $n=0$, $n=1$ and $n=2$ were computed and these features were used as one group of features. For the Ext-HLAC (5x5), 25 features were selected randomly for the comparison purpose. As shown in Table 3, our proposed Ext-HLAC (5x5) features show slightly high classification rates at $n=4$ and $n=5$, as compared to standard HLAC (3x3) features ($k=3$, 45.0%; $k=1$, 51.0%). Although the use of the Ext-HLAC feature requires selection of adequate Ext-HLAC masks from the high number of mask sets, certain Ext-HLAC features can classify images better than traditional HLAC features.

In this section, simple classification techniques based on the k-NN approach were used for experiments. In a KANSEI classification or a user-oriented subjective classification technique, similarity evaluations of user data are analyzed and applied to classification techniques so that classification results match user preferences. We will investigate these KANSEI-based approaches in future works, and these approaches will involve an analysis of a high number of Ext-HLAC features.

Table 3: Five types of images (Ext-HLAC (5x5))

N	0	1	2	3	4	5	6	7	8	9	10	11	12
# of features	1	12	25	25	25	25	25	25	25	25	25	25	25
Classification rates (%) $k=3$	22.0	47.0	65.0	63.0	66.0	67.0	60.0	59.0	59.0	51.0	55.0	54.0	51.0
Classification rates (%) $k=1$	21.0	53.0	71.0	67.0	72.0	78.0	73.0	72.0	65.0	55.0	62.0	57.0	49.0
N	13	14	15	16	17	18	19	20	21	22	23	24	
# of features	25	25	25	25	25	25	25	25	25	25	24	1	
Classification rates (%) $k=3$	41.0	37.0	38.0	31.0	31.0	26.0	28.0	28.0	20.0	22.0	18.0	17.0	
Classification rates (%) $k=1$	41.0	32.0	36.0	29.0	29.0	30.0	23.0	29.0	28.0	27.0	28.0	14.0	

4. CONCLUSIONS

In this paper, standard HLAC (Higher Order Local Autocorrelation) masks (3x3) were extended to Ext-HLAC (Extended Higher Order Local Autocorrelation) masks (5x5). This simple cell size extension greatly increases the number of generated mask patterns. For instance, standard HLAC (3x3) uses 25 masks, extended HLAC (3x3) uses 223 masks, and proposed Ext-HLAC (5x5) uses 16,241,567 masks. Our software program can generate Ext-HLAC masks using typical desktop computers (Intel Core i7-3930K 3.20 GHz) in about 16 minutes. The program-generated Ext-HLAC masks can be used for image processing related applications. In our experiments, Ext-HLAC (5x5) features are extracted from each texture image from a benchmark database. The Ext-HLAC features are grouped into 25 categories based on the order of the autocorrelation functions. Image pattern classification experiments are conducted for comparing each category of Ext-HLAC. Our preliminary experiments show that a certain set of the Ext-HLAC masks can classify images efficiently. Since the proposed Ext-HLAC (5x5) masks can compute various pattern features more efficiently than classical HLAC masks, the Ext-HLAC (5x5) can be applied to KANSEI retrievals or KANSEI classifications.

ACKNOWLEDGMENTS

This research was partially supported by grants from the KAKENHI (C-24500211).

REFERENCES

- Kato, T., Kurita, T., & Shimogaki, H. (1989). Multimedia interaction with image database systems,' Proceedings of Advanced Database Systems Symposium 1989, (pp.271-278)
- Kato, T. (1992). Database architecture for content-based image retrieval, Proc. SPIE 1662. Image Storage and Retrieval Systems, doi:10.1117/12.58497
- Kurita, T., & Otsu, N. (1993). Texture classification by higher order local autocorrelation features, Proc. of Asian Conference on Computer Vision (ACCV'93), (pp.175-178)
- Ojala, T., Maenpaa, T., Pietikainen, M., Viertola, J., Kyllonen, J., & Huovinen, S., (2001). Outex - new framework for empirical evaluation of texture analysis algorithms, International Conference on Pattern Recognition, (pp.701 - 706)
- Otsu, N., & Kurita, T. (1998). A new scheme for practical flexible and intelligent vision systems, Proceedings of the IAPR Workshop on Computer Vision, (pp.431-435)
- Suzuki, M.T., Yaginuma, Y., Yamada, T., & Shimizu, Y., (2006). Shape Descriptors based on Extended 3D Higher Order Local Autocorrelation Masks, 2006 IEEE Mountain Workshop on Adaptive and Learning Systems (SMCals 2006), (pp138-143)
- Toyoda, T., & Hasegawa O. (2005). Texture Classification Using Extended Higher Order Local Autocorrelation Features, The 4th international workshop on texture analysis and synthesis, (pp.131-136)

BIOGRAPHY

Motofumi T. Suzuki received his BS degree in Computer Science from Utah State University, USA, in 1994, and the his MS and Ph.D. degrees in Computer Science from the University of Tsukuba, Japan, in 1997 and 2000, respectively. He was with the National Institute of Multimedia Education, Japan from 2000 to 2009. He is currently an Associate Professor of the Open University of Japan, Japan, and he is teaching "Data Structures" and "Programming Languages" through TV broadcast lecture courses.