INT J COMPUT COMMUN, ISSN 1841-9836
8(5):722-735, October, 2013.

Fast and Accurate Home Photo Categorization for Handheld Devices using MPEG-7 Descriptors

B. Oh, J. Yu, J. Yang, J. Nang, S. Park

Byonghwa Oh, Jungsoo Yu, Jihoon Yang,
Jongho Nang, Sungyong Park
Department of Computer Science, Sogang University
1 Sinsu-dong, Mapo-gu, Seoul 121-742 Korea
mrfive@sogang.ac.kr, yjs@mlneptune.sogang.ac.kr,
yangjh@sogang.ac.kr, jhnang@sogang.ac.kr, parksy@sogang.ac.kr

Abstract:
Home photo categorization has become an issue for practical use of photos taken with various devices. But it is a difficult task because of the semantic gap between physical images and human perception. Moreover, the object-based learning for overcoming this gap is hard to apply to handheld devices due to its computational overhead. We present an efficient image feature extraction method based on MPEG-7 descriptors and a learning structure constructed with multiple layers of Support Vector Machines for fast and accurate categorization of home photos. Experiments on diverse home photos demonstrate outstanding performance of our approach in terms of the categorization accuracy and the computational overhead.
Keywords: machine learning, feature extraction, image classification, mobile computing, content based retrieval.

1 Introduction

Nowadays, there have been great advances in technology related to computers and cameras. People can easily take pictures anytime and anywhere using handheld devices such as cell/smart phones, digital cameras, game consoles, etc. As a result, the management of such *home photos* (taken by armatures, rather than professionals) has become very important for their practical processing, storage, and use. Unfortunately, as mentioned in [1], home photos vary significantly unlike professional or domain-specific images, and the subjects in them are often misinterpreted. Therefore, browsing, searching, and categorizing such photos are nontrivial tasks.

We consider automatic categorization of home photos in this paper. Manual categorization is not appropriate since the time required for it can be even longer than that for the creation of a photograph. Moreover, people have different criteria for categorizing images, which produces non-uniform, unreliable results. So, it is of interest to develop an accurate, automatic categorization method for home photos.

Many researchers have proposed image categorization methods involving feature extraction and learning structures. Some of the studies include object-based learning and their spatial properties, focusing on the relationships among objects from a regional point of view [2, 3]. However, such method is not appropriate for handheld devices since object segmentation is very time-consuming. Thus, even faster and simpler extraction methods need to be devised instead of applying the region or object-based approaches.

The aim of this paper is to present a fast feature extraction method and an efficient learning structure suitable for accurate categorization of home photos, especially for handheld devices. For the former, we use a set of simple feature extractors in MPEG-7 descriptors [4] and a rapid face detector. For the latter, we present a hierarchical learning structure with Support Vector Machines (SVMs) [5] with the consideration of the meaning of concepts. Our approach is tested with a variety of real-world home photos and deployed on actual handheld devices, which

demonstrated good categorization capability. The rest of the paper is organized as follows: Section 2 defines the image features and introduces our feature extraction methods prior to learning. Then the overall structure of the learning method is described in Section 3. Section 4 explains the data, experimental setup, and the results. Section 5 concludes with a summary and discussion of some directions for future research.

2 Image Features and Their Extraction

In order to classify images by categories, we need to train classifiers taking feature inputs of the image and producing category outputs. The features can take various forms. Usually numeric values are used in training because many common classifiers act on numeric inputs for learning and making predictions [6]. We thus decided to use some of the MPEG-7 visual description methods, and an efficient face detector which detects the regions of frontal upright face objects in an image. The extracted values of all these features are numeric.

2.1 MPEG-7 Visual Descriptors

MPEG-7, an ISO/IEC standard developed by MPEG (Moving Picture Experts Group), provides a rich set of standardized tools to describe multimedia content [4]. It is able to efficiently search and retrieve relevant information that people want to use. There are several parts of standards in MPEG-7, and one of them, MPEG-7 visual, covers the following visual descriptors: Color, Texture, Shape, Motion, Localization, and Face recognition.

Eidenberger asserted that an efficient (general-purpose) descriptor should provide a surjective mapping from media object to points in feature space [7]. He supposed an ideal descriptor should be highly discriminating for any type of media content. He concluded that the best descriptors for combinations are Dominant Color Descriptor (DCD), Color Layout Descriptor (CLD), Edge Histogram Descriptor (EHD), and Texture Browsing Descriptor (TBD).

However, the extraction of texture descriptors such as TBD actually entails a higher time complexity than extracting color descriptors (DCD and CLD). This made us select the three best descriptors, DCD, CLD and EHD. Also, we added Color Structure Descriptor (CSD) to make up the exclusion of TBD and enhance the expressive power. From our previous research, we developed an extraction method for CSD which is much faster than other methods [8]. In the end, we used four MPEG-7 visual descriptors, CSD, DCD, CLD, and EHD. The first three descriptors are color descriptors and the last is a texture descriptor. Color descriptors have the ability to characterize the perceptual color similarity and generally have low complexities of extraction and matching. EHD characterizes the structures in generic images in forms of edge contents and layouts.

Prior to this work, we developed optimized versions of software engines that extract CSD, CLD and DCD [8]. Table 1 shows the computation time of some of the visual descriptors measured on handheld devices. Our engines were much faster than the XM reference software [9] which provides non-optimized extraction methods of visual descriptors. The speed gap will be even greater for images with lower resolutions (e.g., 320×240) instead of 640×480. Additional descriptors such as Homogeneous Texture Descriptor (HTD) and Region Shape Descriptor (RSD) are very time-consuming and therefore they are unsuitable for use in handheld devices.

2.2 Face Descriptor

As the existence of particular objects in the image can aid the categorization process, we adopt the efficient face detector proposed in [10] which is trained with about 100,000 manually

Table 1: MPEG-7 Visual Descriptor Profiling Performances for 640 × 480 Images (in Milliseconds).

Descriptor	**HP iPAQ rx5965 PDA (ARM9 400Mhz)**		**Samsung Omnia 2 Phone (ARM11 800Mhz)**	
	XM Reference	**Optimized**	**XM Reference**	**Optimized**
CSD	4,500	900	2,800	850
CLD	150	50	50	30
DCD	23,000	170	13,000	110
EHD	600	-	550	-
HTD	9,600	-	8,400	-
RSD	61,000	-	50,550	-

cropped upright frontal face images. Then we detect at most 10 faces in an image each of which is represented by its area to define the Face Descriptor (FD).

3 The Process of Categorization

In order to develop the home photo categorization system, it is needed to train a classifier that classifies images under predefined categories. We build two-layered independent classifiers in order to enhance the classification performance in two steps. Figure 1 depicts the overall structure of the proposed categorization system.

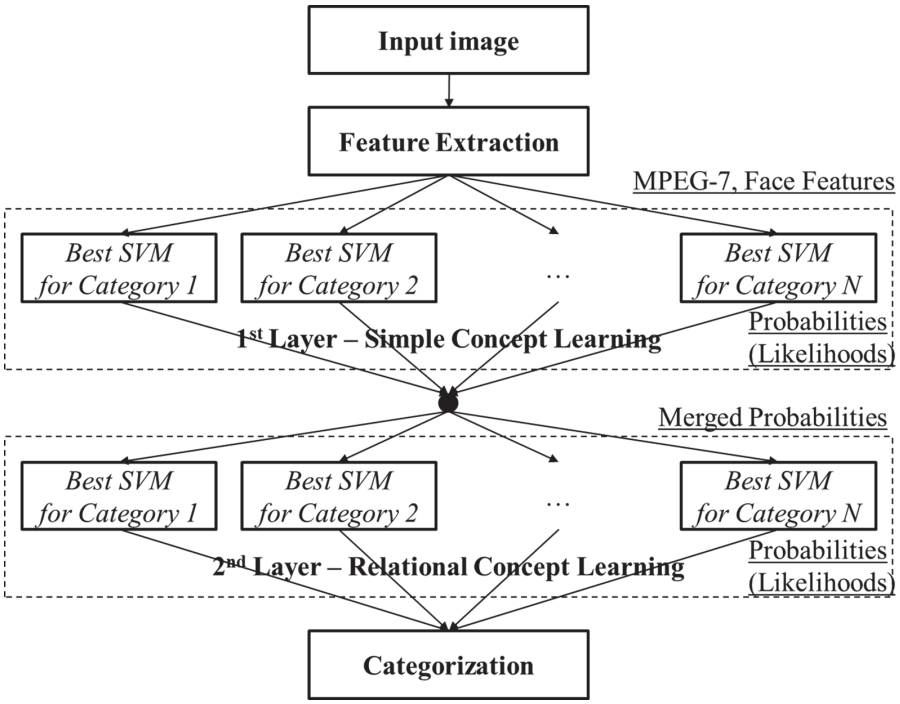

Figure 1: Overall Structure of Home Photo Categorization System.

First, the feature vectors are constructed using the MPEG-7 feature extractors and the face detector as described in Section 2. Then, the classifiers in the 1st layer are trained with the feature vectors to produce the probabilities (or likelihoods) for an image to belong to each category.

Next, the classifiers in the 2nd layer take the probabilities produced in the previous layer as inputs, and compute a new set of probabilities for each category, by considering the constraints immanent in the data. For instance, there may be constraints such as: *"If it is nighttime, then it may not be a landscape"* or *"Most of the photos taken of waterside regions are images of nature, not cities"*, and so on. Then the probability for a landscape will be lowered if the probabilities for nighttime and a landscape are both high, by passing through the 2nd layer. Since these constraints are unknown, the classifiers of the 2nd layer should be trained to reflect such knowledge. In other words, the 2nd layer is in charge of incorporating relational meanings among the categories.

Lastly, the system decides the category the image belongs to. For this, our system simply chooses the category with the greatest probability over all N values for N categories.

In order to follow this process, the best classifiers (in terms of certain performance criteria such as classification accuracy or F1 measure) need to be constructed in both layers.

3.1 Building the First Layer Classifiers

We adopt Support Vector Machines (SVMs) [5] as the base classifiers in our system, with four commonly used kernels (linear, polynomial, RBF, sigmoid). We also applied feature subset selection to obtain the best performance with minimum computational overhead and data acquisition cost. So with the five features of CSD, DCD, CLD, EHD and FD extracted in Section 2, we can construct 31 feature subsets (of different feature combinations).

Now we can find the best classifier for each category by changing kernels and feature subsets with 4×31 experiments for a given training data. This process is repeated to determine the best classifiers for all categories. Figure 2 illustrates the process.

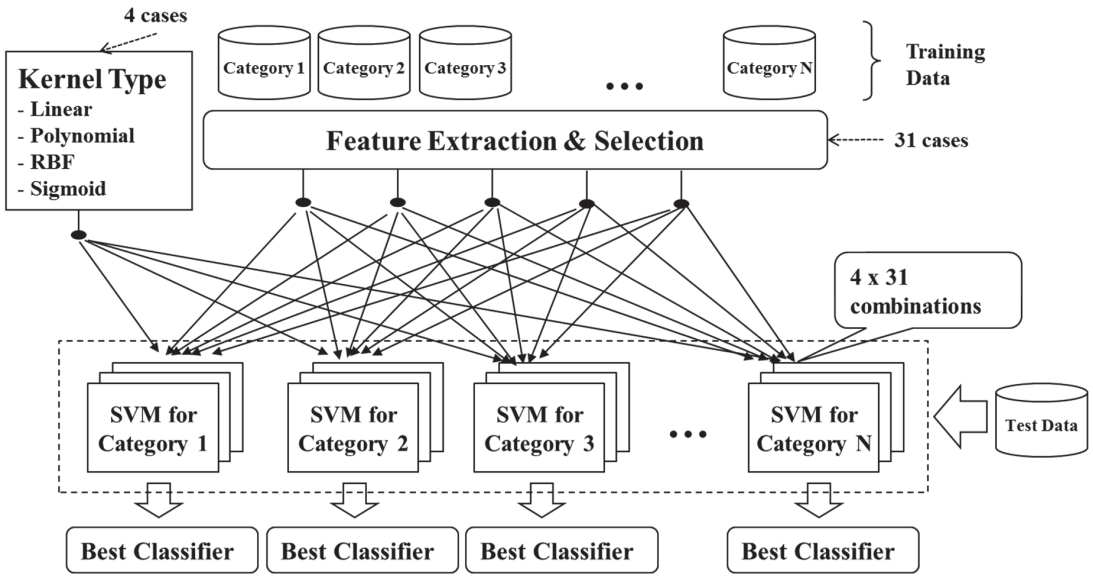

Figure 2: Process of Finding the Best Classifiers in the First Layer.

3.2 Building the Second Layer Classifiers

As a result of the first layer learning, each SVM produces the probability that an image belongs to the category it represents. But these outputs are not guaranteed to be correct since each probability is induced by separate SVMs without considering the correlation (or constraints)

Table 2: Discovered Rules on the Concept *"When"*.

Rule number (Importance)	**Rule**
1 (Strongest)	**IF** Night $<$ 0.6625 **AND** Sunrise/Sunset $<$ 0.4004 **THEN Day**
2	**ELSE IF** Landscape $>$ 0.0024 **AND** Stadium $<$ 0.04265 **AND** Evening $\leq$ 0.5206 **AND** Night $\leq$ 0.9865 **AND** Snow $<$ 0.2154 **AND** Sunrise/Sunset $\leq$ 0.9625 **AND** Waterside $<$ 0.5083 **THEN Day**
3	**ELSE IF** Stadium $<$ 0.2822 **AND** Day $<$ 0.9575 **AND** Evening $>$ 0.2788 **AND** Snow $<$ 0.2154 **AND** Sunrise/Sunset $\geq$ 0.3696 **AND** Waterside $<$ 0.5083 **THEN Evening**
4 (Weakest)	**ELSE Night**

among the categories. For example, if there are more photos that capture nature in daylight than night, then there could be stronger correlation between daylight and nature than night and nature.

In order to check the utility of learning correlations on categories, we conducted a simple experiment. At first we trained the first layer with training data and prepared two datasets different from the training data. Then we produced two datasets by attaching the probabilistic outputs from the first layer to the inputs of the prepared datasets. After that we investigated the relationships between categories by training the rule-based classifier we had developed [11] (The rule-based classifier is based on successive, greedy generation of rules each of which covers most of the data at each time). We could find some relationships between the rules generated. Table 2 displays examples of discovered rules.

As we see from Table 2, there are many (strong and weak) relationships among categories. Generally, slightly dark images (such as the images with the possibility of $0.5 <$ Night < 0.6625) are likely to be classified as night images without considering the correlations. But we can say these images are taken at daytime when they have a weak likelihood of Sunrise/Sunset.

We conclude that if we can build the classifiers reflecting many constraints well, the performance of categorization will improve significantly. In our experiments, the prediction results of these rule-based classifiers were not always good because many relationships had non-linear characteristics. As a remedy for this, we introduce additional layer of SVMs. The SVMs in the 2nd layer take the probabilities from the previous layer as inputs and outputs a new set of probabilities considering such hidden correlations among the categories.

3.3 Outputs of SVM Classifiers

Generally an SVM performs binary classification so it outputs only one of the two predefined numbers (for classes or categories). Also, an SVM does not output likelihoods in real values. So it is not possible to build the classification structure mentioned in this section by using standard SVMs.

Fortunately, there is a way to estimate likelihoods of outputs easily. Yang *et al.* used the confidence values as likelihoods of categorization results [3]. They assumed that the bigger confidence value means the stronger connection with the concept. They defined the confidence value as the distance of the input feature vector from the trained hyperplane of the SVM.

However this approach is not feasible because the meaning of distance varies according to the distribution of sample data. Let us assume that there are two different categories, A and B. For category A, positive samples are far from negative samples. But for category B, the distance between positive and negative samples is small. Then, the same confidence values of A and B

Table 3: Defined Categories and Distributions of Data (Unit: The Number of Samples in Each Category).

Three W's	Category	Brief Description	TD	Q1	Q2	VD
What	Waterside	River, sea or lake	397	103	142	43
	Snow	Snowcapped sites	419	92	101	38
	Self-portrait	Focus on a face (the most part)	374	111	101	35
	Food	Focus on food	400	90	97	40
	People	Many people	422	101	131	44
	Sunrise/Sunset	Sun or a glowing sky	404	94	91	40
	Unknown	No conspicuous object	-	550	440	218
When	Night	Night or in the dark	501	84	109	55
	Evening	Sun or dusk falling	479	93	90	50
	Day	In the bright light	504	964	904	353
Where	Stadium	Park, field or stand	300	109	108	41
	City	Buildings or roads	301	89	152	39
	Landscape	Mountain, river, sea or snowy sites	400	342	346	131
	Unknown	No conspicuous object	-	601	497	247

have different meaning, and it is obvious that the confidence value of B is more important. In addition, in our photo categorization approach, different feature subsets and kernels are used for each category. So the meanings of confidence values on categories can be different even if A and B have the same distribution. Therefore a different way of estimating likelihoods is needed and it must be available to compare the outputs without paying attention to the meanings of likelihoods.

Wu, Lin and Weng developed approaches for obtaining class probabilities in addition to the classification results of binary and multiclass classifiers [12] implemented in the LIBSVM software [13]. Unlike the confidence values of Yang *et al.*, the probabilistic outputs of the Wu's method all have the same ranges of 0 to 1, so the values can be compared with each other. We adopted the method to estimate the probabilities of SVM outputs.

4 Experiments

4.1 Data and Categories

First, we prepared four distinct chunks of home photos by using the feature extraction methods described in Section 2: training data (TD), test data 1 (Q1), test data 2 (Q2) for SVMs in the first layer, and validation data for deciding kernels of SVMs in the second layer (VD). All photos were collected by requests to authors' acquaintances or by downloading from personal blogs.

Home photo can take a variety of forms and be assigned to diverse categories. We focused on the approach of the five W's and one H, which is a formula for getting the full description of a situation. The system was unable to recognize *"who"* in an image. But it could detect whether a person was in it or not. Thus, the person could be regarded as an object of *"what"*. Also, we cannot easily figure out why or how the picture was taken. So we finally considered only three W's which are *"what"*, *"when"* and *"where"* as the highest groupings, and then defined categories (and descriptions or representative objects in the categories) of our interest under such groupings. Table 3 shows the detailed information on the data.

For the training dataset (TD), we gathered representative photos separately for each category.

For example, there are only the objects related to the stadium in stadium images regardless of when they were taken. The images of each category are the inputs of each SVM in the first layer. There are 4,901 photos in total in TD.

There are 1,141, 1,103 and 458 photos in total in Q1, Q2 and VD datasets, respectively. In Q1, Q2 and VD, all pictures belong to three categories: *"what"*, *"when"* and *"where"*. For instance, a photo can belong to *where-unknown*, *when-day*, and *what-food* but cannot belong to *where-city*, *what-snow* and *what-people*. In the case of *"what"*, if there are several objects in an image, the largest and centered object stands for the image. If there is no conspicuous object in the image, it is regarded as *unknown*. Figure 3 shows some sample images corresponding to the categories.

4.2 Experimental Process and Results

Previously we discussed the categorization process of home photos. For the practical applications of this, we need to go through the procedures finding the best classifiers in the first and second layers as explained in Section 3.

We construct two classification models, Model 1 and 2, for the verification of our system. For Model 1, we train SVMs in the first layer with TD, by choosing the feature subset and the kernel for all possible cases. We then discover the best classifier settings for all SVMs by selecting classifiers which yield the best classification results on the test data Q2 for each category. There are 12 categories excluding the unknown category, so 12 SVMs are trained in the first layer. Next, we train SVMs in the second layer with the classification result of Q2 (likelihoods) by the SVMs in the first layer. After that, in the same way as selecting the best classifiers in the first layer, we classify VD and decide kernels for each SVM that yields the best classification result. Similarly, we can construct Model 2 using TD and Q1 as training data.

Finally, we use Q1 as test data for Model 1 and Q2 as test data for Model 2 and evaluate the classification performance. The reason for using four different sets of data (i.e., TD, Q1, Q2, VD) is to derive robust classifiers with good generalization capability by considering data prepared at different times and/or by different people. Figure 4 displays the scheme of the photo categorization system and the learning procedure.

In the studies of image categorization, precision and recall are both important. Precision estimates how well the system removes what users do not want. Recall estimates how well it finds what users want. So we use the F1 measure (1) that combines precision and recall with an equal weight, as a performance criterion in finding the best classifiers.

$$F1 = 2 \cdot precision \cdot recall/(precision + recall). \tag{1}$$

The experimental results (excluding the unknown category) of Model 1 are shown in Table 4 and 5, with the selected parameters of each SVM and the performance on Q1 in terms of precision and recall. The results of Model 2 are shown in Table 6 and 7, with performance on Q2 (*Prec* means precision).

Figure 5(a), 5(b), 6(a) and 6(b) are the visualized results (in precision and recall) of Table 4, 5, 6 and 7, respectively.

We can see a great improvement in precision by using the second layer. The average precision was increased from 0.712 to 0.809 in Model 1 and 0.726 to 0.829 in Model 2. The city and the waterside categories are rather inaccurate, because the objects of a city such as buildings and roads are similar to artificial or indoor objects making it difficult to distinguish them, and the waterside photos have various forms of color, composition, shape, texture, and so on. But these difficulties are overcome by introducing the second layer of SVMs.

Figure 3: Categories of Home Photos and Their Descriptions Including Sample Photographs.

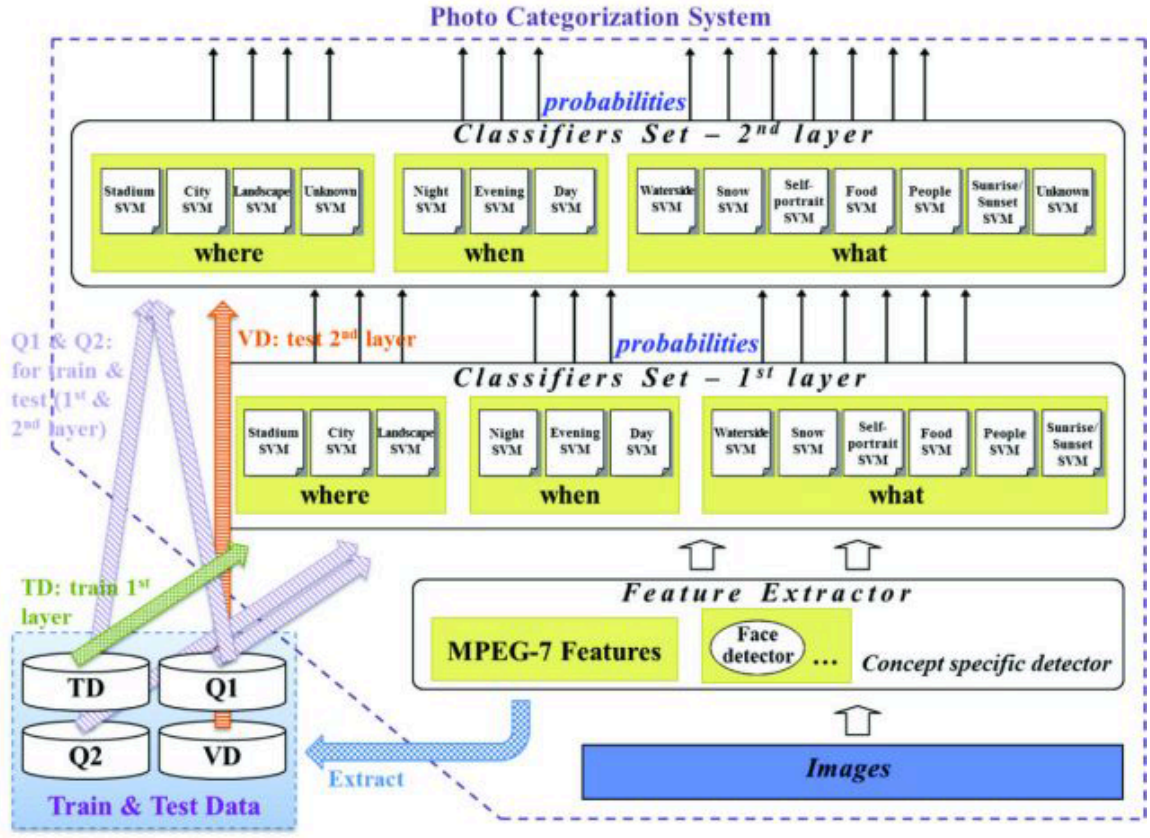

Figure 4: Training Classifiers and Validation with Datasets in Table 3.

Table 4: Selected Settings and Performances of Model 1 (First Layer).

W's	Category	Selected Features	Kernel	Prec	Recall	F1
What	Waterside	All features	RBF	0.446	0.563	0.497
	Snow	All features	Poly	0.586	0.771	0.666
	Self-portrait	DCD, FD	RBF	0.919	0.819	0.866
	Food	DCD, EHD, FD	Poly	0.804	0.733	0.767
	People	FD	RBF	0.969	0.623	0.759
	Sunrise/Sunset	CSD, CLD, EHD	Poly	0.834	0.914	0.873
When	Night	DCD, EHD, FD	Poly	0.650	0.797	0.716
	Evening	DCD, CLD, EHD	Poly	0.750	0.870	0.805
	Day	DCD, CLD, EHD	Poly	0.985	0.845	0.910
Where	Stadium	CSD, EHD, FD	Poly	0.707	0.908	0.795
	City	DCD, CLD, EHD, FD	RBF	0.331	0.674	0.444
	Landscape	CSD, DCD, CLD, EHD	Poly	0.568	0.751	0.647
Average				0.712	0.772	0.729

Yang *et al.* defined categories similar or identical to ours (i.e., terrain (corresponding to landscape), night-scene (night), snowspace (snow), sunset (sunrise/set), and waterside) [3]. Even though the comparison with the results reported in the paper is not perfect, we see our system produces higher precision and lower recall in general, and is significantly better in snow and sunrise/set categories in particular. The comparison result of our approach (the second layer of

Table 5: Selected Settings and Performances of Model 1 (Second Layer).

W's	Category	Kernel	Prec	Recall	F1
What	Waterside	RBF	0.584	0.300	0.397
	Snow	RBF	0.797	0.641	0.710
	Self-portrait	RBF	0.918	0.810	0.861
	Food	RBF	0.905	0.533	0.671
	People	Poly	0.969	0.623	0.759
	Sunrise/Sunset	RBF	0.912	0.882	0.897
When	Night	Poly	0.744	0.761	0.752
	Evening	Poly	0.951	0.838	0.891
	Day	Poly	0.978	0.973	0.975
Where	Stadium	Sigmoid	0.725	0.944	0.820
	City	Poly	0.616	0.595	0.605
	Landscape	RBF	0.606	0.567	0.586
Average			0.809	0.706	0.744

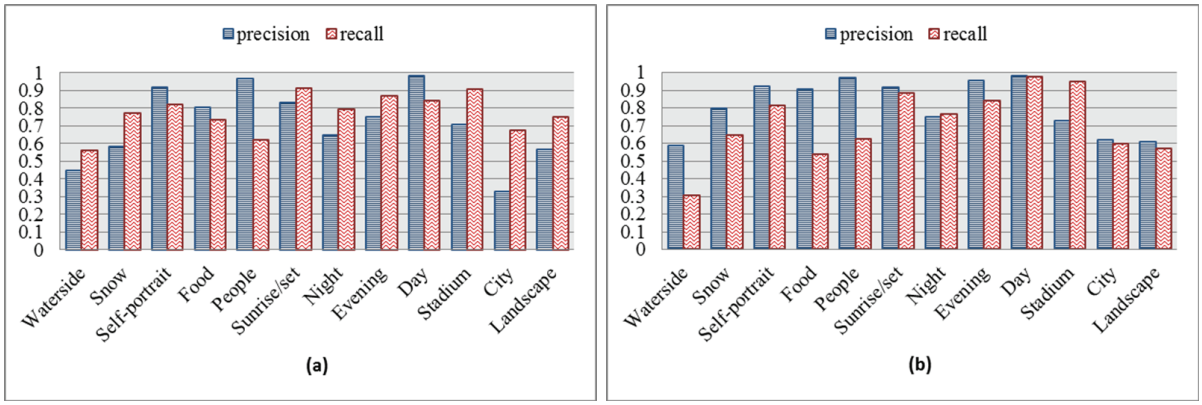

Figure 5: Classification Performance in Model 1 ((a): First Layer, (b): Second Layer).

Table 6: Selected Settings and Performances of Model 2 (First Layer).

W's	Category	Selected Features	Kernel	Prec	Recall	F1
What	Waterside	CSD, DCD, CLD, EHD	Poly	0.494	0.612	0.547
	Snow	All features	Poly	0.758	0.900	0.823
	Self-portrait	CLD, FD	Poly	0.814	0.435	0.567
	Food	DCD, CLD, EHD, FD	RBF	0.733	0.793	0.762
	People	FD	RBF	0.777	0.427	0.567
	Sunrise/Sunset	CSD, DCD, CLD, EHD	Poly	0.802	0.714	0.755
When	Night	DCD	RBF	0.862	0.807	0.834
	Evening	CSD, DCD, CLD, EHD	Poly	0.634	0.733	0.680
	Day	CLD, EHD	Poly	0.980	0.825	0.896
Where	Stadium	CSD, CLD, EHD, FD	Poly	0.701	0.675	0.688
	City	DCD, CLD, EHD, FD	RBF	0.492	0.651	0.560
	Landscape	CSD, DCD, CLD, EHD	Poly	0.659	0.841	0.739
Average				0.726	0.701	0.702

Table 7: Selected Settings and Performances of Model 2 (Second Layer).

W's	Category	Kernel	Prec	Recall	F1
What	Waterside	Poly	0.563	0.373	0.449
	Snow	RBF	0.890	0.801	0.843
	Self-portrait	Poly	0.854	0.405	0.550
	Food	Sigmoid	0.745	0.783	0.763
	People	Sigmoid	0.730	0.351	0.474
	Sunrise/Sunset	Sigmoid	0.869	0.659	0.750
When	Night	RBF	0.926	0.577	0.711
	Evening	Sigmoid	0.904	0.633	0.745
	Day	RBF	0.931	0.991	0.960
Where	Stadium	Poly	0.850	0.472	0.607
	City	Poly	0.905	0.315	0.468
	Landscape	Sigmoid	0.775	0.699	0.735
Average			0.829	0.588	0.671

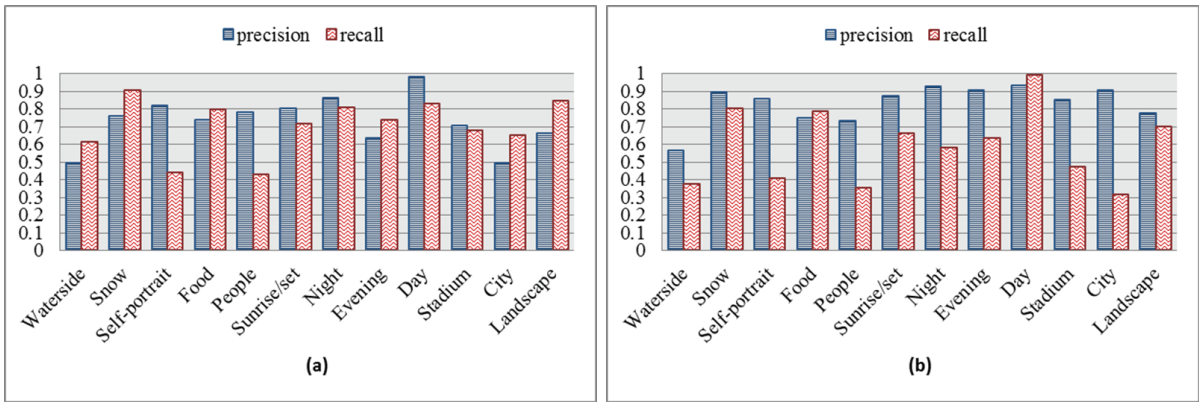

Figure 6: Classification Performance in Model 2 ((a): First Layer, (b): Second Layer).

Model 1) and Yang *et al.*'s method is shown in Figure 7.

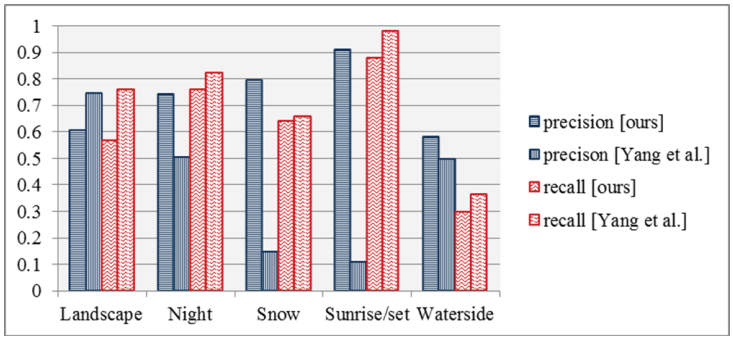

Figure 7: Performance Comparison between the Method of [3] and Our Method.

Moreover, there is no evaluation of computation time in [3]. We surmise their method would be much slower than ours since they use time-consuming features like the Homogeneous Texture Descriptor, and extract such features quite often for every five local regions. In contrast, our algorithm uses fast feature extraction methods and does not extract features repeatedly from an

image. Actually, it takes about less than one second for categorizing a photo on the Samsung Omnia 2 Smartphone (ARM11 800Mhz CPU). So our algorithm is suitable for use on handheld devices.

Figure 8 displays snapshots of sample runs of the home photo categorization and browsing software performed on the actual device. Users can browse photos by selecting *"what"*, *"when"* and *"where"* categories. Also the software supports the auto-categorizing function for photographing by the built-in camera.

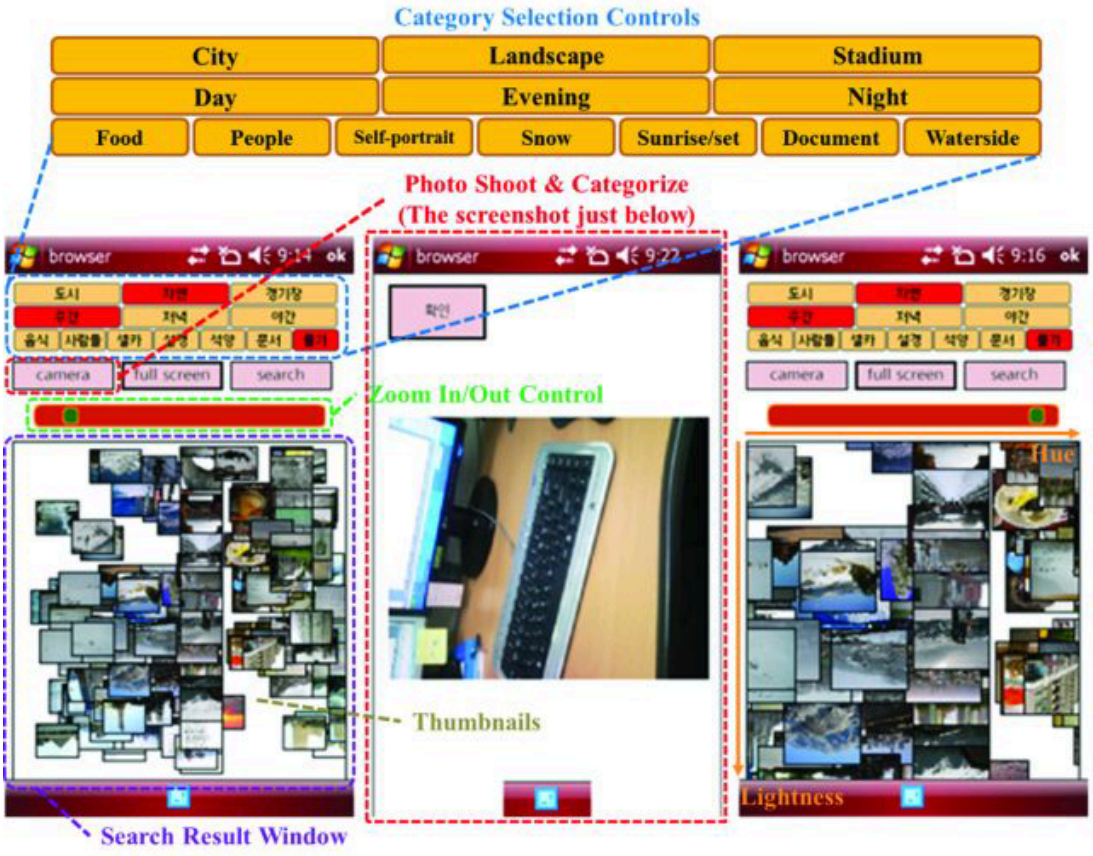

Figure 8: Sample Runs of The Home Photo Categorization and Browsing Software.

5 Conclusion

In this paper, we proposed an efficient home photo categorization method using fast MPEG-7 descriptors and the face extractor, and a two-layered classifiers of SVM. The classifiers in the first layer are trained to assign images into predefined categories, and the ones in the second layer attempt to improve the classification performance by considering the relationships and constraints among the categories. In the way to construct the multi-level classifiers, we also considered feature and kernel selections and obtained the best feature subsets and kernel functions. Our method was compared with one of the home photo categorization methods and verified to produce outstanding performance with less computational overhead, which is a prerequisite for the implementation in real handheld devices.

In spite of the effectiveness of the proposed method, there are several challenging issues. First, as the face feature is the most important factor in distinguishing people and self-portrait photos

from others, we may as well implement new feature extractors specialized in extracting unique features of photos in certain categories (e.g. city, landscape). The extractors should have low computational cost in order to support real-time categorization. By applying the state-of-the-art technology like the fast object detection method [14], we may be able to obtain better results. We are currently developing extracting tools for additional object-based features (e.g., buildings for city category) to enhance our categorization system. As far as face detection, the current algorithm works only with frontal faces, which can be extended to consider rotated objects as proposed in [15]. Also, the recall of the second layer's outputs was not improved with the concept of relational learning, unlike precision, so we need to compensate for this weak point to make our method more powerful. In addition, finding the best parameter settings for SVM is of significance instead of blindly relying on widely used ones.

Acknowledgments

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology to Jihoon Yang, the corresponding author (2013R1A1A2012502), and by the IT R&D program of MSIP/KEIT (10044615: Development of Open-Platform/Social Media Production and Delivery System for Fused Creation, Editing, and Playing of Broadcasting Media Contents on Cloud Environments).

Bibliography

[1] J.H. Lim, J.S.Jin, Unifying local and global content-based similarities for home photo retrieval, *Proceedings of 2004 International Conference on Image Processing*, 4:2371-2374, 2004.

[2] Y. Chen and J.Z. Yang, Image Categorization by Learning and Reasoning with Regions, *The Journal of Machine Learning Research*, 5:913-939, 2004.

[3] S.J. Yang, S.K. Kim, K.S. Seo, Y.M. Ro, J.Y. Kim, Y.S. Seo, Semantic categorization of digital home photo using photographic region templates, *Proceedings of 2005 Information retrieval research in Asia*, 43(2):503-514, 2007.

[4] J.M. Martínez, MPEG-7 Overview, *ISO/IEC JTC1/SC29/WG11N6828*, 2004.

[5] C.J.C. Burges, A Tutorial on Support Vector Machines for Pattern Recognition, *Data Mining and Knowledge Discovery*, 2(2):121-167, 1998.

[6] T. Mitchell, *Machine Learning*, McGraw Hill, 1998.

[7] H. Eidenberger, Statistical analysis of content-based MPEG-7 descriptors for image retrieval, *Multimedia Systems*, 10:84-97, 2004.

[8] J.S. Yu, J.H. Nang, An Optimization Method for Extraction of MPEG-7 Color Structure Descriptor and Dominant Color Descriptor, *Proceedings of Korea Computer Congress 2009*, 36(1A):320-321, 2009.

[9] Institute for Integrated Circuits, Technische Universit Munchen, *MPEG-7 XM Software*, Germany, 2003. Available (Online):

http://standards.iso.org/ittf/PubliclyAvailableStandards/c035364_ISO_IEC_15938-6%28E%29_Reference_Software.zip

[10] B. Fröba, A. Ernst, Face Detection with the Modified Census Transform, *Sixth IEEE International Conference on Automatic Face and Gesture Recognition*, 0:91-96, 2004.

[11] B.H. Oh, J.H. Yang, Discovering Classification Rules using Genetic Algorithm, *Proceedings of Korea Computer Congress 2009*, 36(1C):480-485, 2009.

[12] T.F. Wu, C.J. Lin, R.C. Weng, Probability Estimates for Multi-class Classification by Pairwise Coupling, *Journal of Machine Learning Research*, 5:975-1005, 2003.

[13] C.C. Chang, C.J. Lin, *LIBSVM : a library for support vector machines*, 2001.

[14] M.M. Jlasi, A. Douik, H. Messaoud, Objects Detection by Singular Value Decomposition Technique in Hybrid Color Space: Application to Football Images, *International Journal of Computers Communications & Control*, 5(2):193-204, 2010.

[15] T. Barbu, An Automatic Face Detection System for RGB Images, *International Journal of Computers Communications & Control*, 6(1):21-32, 2011.