Layered Self-Organizing Map for Image Classification in Unrestricted Domains

Christian O'Connell[1], Andrea Kutics[2], and Akihiko Nakagawa[2,3]

[1] University of Essex, Colchester, United Kingdom
cdocon@essex.ac.uk
[2] International Christian University (ICU), Tokyo, Japan
matz@icu.ac.jp
[3] University of Electro-Communications (UEC), Tokyo, Japan
ranaka@ppp.bekkoame.ne.jp

Abstract. The inherent difficultly in unrestricted image domain classification is due to the many different features exhibited by images. Efforts made toward classification of abstract features tend to focus on a single attribute. Without a method of unifying descriptors, it becomes very difficult to perform multi-feature analysis. Extending the concept of the Self-Organizing Feature Map to include multiple competitive layers, it has been possible to create a new type of Artificial Neural Network capable of analyzing image and signal datasets with multiple feature descriptors concurrently in a powerful yet computationally light manner. Compared to standard CBIR retrieval approach, a marked increase in the precision of clustering of 13 points has been achieved, along with a reduction in computation time.

Keywords: self-organizing map, image classification, features, image retrieval.

1 Introduction

The applications of image classification are vast, encompassing many fields from simple image searches to complex object detection. While great strides have been made to derive meaning from large datasets into a manner akin to a human observer, the complexity of the challenge means such a solution remains elusive. Fundamental differences between machine interpretation and the subjective nature of human perception – the so called "semantic gap" – have made it an extremely difficult task to which no clear solution for consistent meaningful results has arisen. Attempts to bridge this gap have led to numerous methods to derive meaning from image data being developed, utilizing most prominently the fields of artificial intelligence, data mining, cognitive psychology and learning from context. Progress toward achieving human-like reasoningwith image datasets are often inflexible, working only within predefined domains. Efforts such as the MPEG-7 multimedia content description standard [1], which provides a number of 'feature descriptors' to describe features such as color distribution and texture as a vector, have achieved a level of consistency

A. Petrosino (Ed.): ICIAP 2013, Part I, LNCS 8156, pp. 310–319, 2013.

for comparison, but little continuity between descriptors makes multi-descriptor analysis difficult. The non-linear statistical processing capabilities of Artificial Neural Networks (ANNs) have also generated interest in the field of image processing. By utilizing their architecture, it would be possible to unify their dynamicity with the accuracy offered by feature descriptors to improve image classification.This paper proposes a new architecture that is a derivation of the Self-Organizing Map (SOM) ANN [2] which employs an arbitrary number of competitive layers to perform unsupervised multi-descriptor analysis of datasets.

2 Related Work

Unsupervised clustering methods like SOM are frequently used for image classification, and the SOM has proved to be especially convenient for this task due to its 2D mapping capabilities making the resulting clusters easy to visualize. In SOMs, the neuron's weight often acts as a representation of the color domain such as RGB in vector format, and is altered during the training period to more closely match randomly selected pixels from the input image, "mapping" the SOM's topology to form a 2D representation of a higher dimensional vector space [3]. As in image classification the input data is represented as a composite high dimensional feature vector with variable component numbers in various individual features. Different types of SOM architectures are proposed, for example, satellite images can use different input layers according to the satellite specialization directive, and a GRID solution is proposed to be able to store satellite image codebook vectors and also different layers corresponding to soil and water in a weather forecast application [4]. The field of bioinformatics uses a Multi-SOM algorithm [5] for data mining large high-dimensional datasets with a number of small SOMs. Different data structures have been analyzed by SOMs using multiple kernels [6]. Another proposed text/image classification uses different font-types for grouping and presenting them to prototypes [7] and finally clustering them to three main clusters. SOMs can be considered as a solution to automatically estimate and linearly merge the weights of individual features with different distance measures for content-based image retrieval (CBIR) [8][9]. The problem can be traced back to [10] where the authors use a tree-structured type of SOM. Clustering is carried out on this basis and the contribution of each feature for the cluster structure is used for the selection of the proper weights. Although all of the above-mentioned contributions obtain relatively good results on restricted image domains, the problem of accurately classifying images of non-restricted domains and having specific, high dimensional and independent features, that cannot be combined linearly, remains unsolved.

3 Layered Self-Organizing Map

Analyzing more abstract image features such as an MPEG-7 feature can be performed effectively using an SOM to reduce the higher dimensional descriptors into a two dimensional network of relationships. Descriptor values are taken from an image

Algorithm 1. Layered SOM algorithm

```
for t = 1 to λ
  select X ∈ L_0
  N ≔ ι(X)
  while N ≠ {∅}
    BMU ≔ N_i | min_{∀i∈N}(‖X_F − N_i‖)
    for each v ∈ ω[BMU, Ω(t)]
      v(t + 1) = v(t) + α(t)[X_F − v(t)]
    end
    N ≔ ι(BMU)
  end
end
```

L_n – Layer n
$\iota(X)$ – Set of neurons interlinked to X
X_F – Appropriate descriptor associated with X for distance calculation
$\omega(X, Y)$ – Set of neurons in the neighborhood of X with size Y
$\alpha(t)$ – Learning rate at t

dataset and fed into the SOM as input which maps itself to the feature space. Clustering is discerned through analysis of the topographical and topological structure of the map. The primary limitation encountered is the maximum of one feature that can be analyzed at a time, significantly reducing the effectiveness of the map. A way of extending the principle of SOMs to enable concurrent processing of multiple vector weights needed to be found, however descriptors can have different numerical representations of data components with varying probability distributions (number of dimensions, range etc.), thus are unable to be compared directly. Initial attempts to adapt the original SOM algorithm to concurrently analyze multiple descriptors linearly proved unstable. Each neuron held all weights and used a calculated weighted measure to obtain the feature distance, however intrinsic differences between descriptor components based on their statistical distribution made it difficult to obtain fair comparison results, whilst quantization and scaling lead to an unacceptable loss of precision.This motivated the development the Layered SOM (LSOM), a new extended approach to the original SOM which delegates the task of analyzing each descriptor to individual competitive layers, interlinked sequentially to form a feed-forward network. This retained the simplicity and effectiveness of each neuron classifying a single weight, with each layer mapping a single descriptor, whilst maintaining a level of independence between descriptors so they need never be compared directly to one another; instead their topological position forming the basis for comparison creating an implicit correlation between descriptors.

3.1 Feature Extraction

Before any classification can occur, the image features used in the LSOM need to be calculated via the appropriate methods for every image in the dataset. The features can be expressed in any format which allows comparison based on a scalar output i.e. distance; a vector or matrix representation being the most common. Feature extraction in this paper is performed mostly by using the methods defined by the MPEG-7 standard. The Homogeneous Texture Descriptor (HTD) uses directional 2D Gabor filters to measure the statistical properties of the partitioned domain over a set of scales.

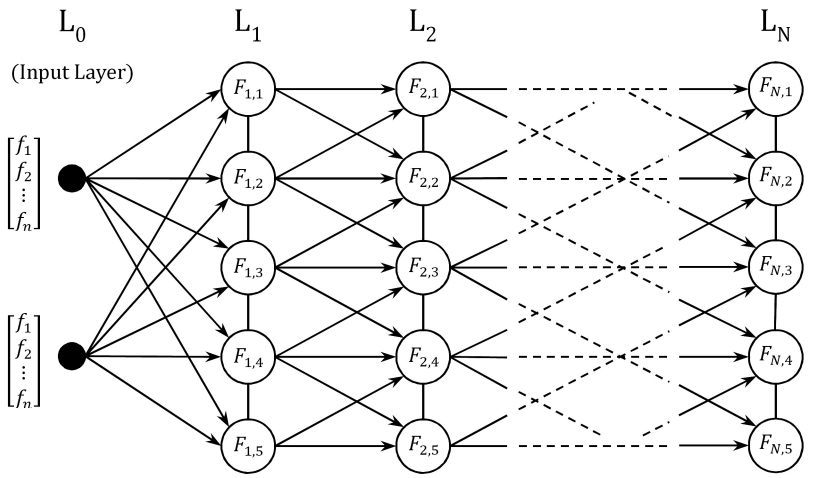

Fig. 1. LSOM Architecture

Its power comes from how the Gabor filter's operation closely models that of the human visual cortex, producing data in line with human understanding. The Edge Histogram Descriptor (EHD) is also used as a method to extract texture-like properties in a highly computationally efficient way by measuring direction of detected edges. Its functionality has been improved by incorporating Canny edge detection to create a new version more adept at texture detection through improved detection of relevant edges. Color analysis is handled by the Color Layout Descriptor (CLD), which segments an image into a spatially equal 8×8 grid, and the average of a cell's YCbCr color space is calculated. The final descriptor, although not a part of the MPEG-7 standard is the Color Correlogram [11], which documents the spatial relationship between pixels in an image. The Autocorrelogram is a variation which takes the main diagonal. Calculating the features for each image is the most computationally intensive part of the process; however is necessary only once as descriptors can be reused for repeated calculations. Each input to the LSOM is a collection of features to be analyzed by the LSOM. This collection of inputs is referred to as Layer 0 (L_0).

3.2 Network Initialization

- **Layer ordering.** Competitive layers in the LSOM are ordered in a feed-forward topology from L1 to LN, with neurons on each layer classifying a single descriptor. Descriptors positioned earlier in the map provide an approximate initialization of clusters for the next layer, thus descriptors which should have a higher influence on the results should be positioned later in the map.
- **Layer topology.** Each layer is identical in the number of neurons and topology, differing only in the type of descriptor held as the weight. More efficient topologies include hexagonal and square structures; the latter is used in this paper as it is effective for visualization, clearly showing the propagation of change evenly

throughout the network. Neurons are interlinked to neurons on adjacent layers using binary connections to create sets of neurons from which the training algorithm (Algorithm 1) selects the best matching units (BMUs). Inputs (Layer 0) are interlinked to every neuron on Layer 1 as with the regular SOM. For all other layers, neurons on Ln are interlinked to only a subset neurons on Ln+1. This is usually the neuron with the same coordinates and a 1-radius neighborhood (Figure 1). The effect is to apply an approximate regional sort based upon prior descriptors, which is then refined by subsequent layers.

- **Weights.** Neurons on L1 are the only ones which do not receive any form of real time initialization. At its simplest they can be initialized on a purely random basis. Better results can often be achieved by applying statistical analysis to the input domain (L0) to give an approximate mapping, which are then propagated throughout the layers during the training period. This paper calculates the initial weights using the properties obtained from the Gaussian type distribution of input features. Taking the central neuron of Layer 1 and assigning it the mean value of the appropriate input dataset descriptor, and using the standard deviation as the rate of change to set neurons in each neighborhood radius. Alternative methods include using Hidden Markov Models, although they are more computationally intensive.

3.3 Neighborhood Function

When a neuron is excited, its topological neighborhood, as defined by the neighborhood function, is also affected. Choice of implementation is critical to ensuring a correct propagation of inputs throughout the network, and best results were obtained using a retracting radius proportional to the learning restraint:

$$\Omega(t) = \max\left(1, \left[\max\left(w, h\right) \cdot \alpha(t)\right]\right) \tag{1}$$

Giving the number of concentric radii to include in the neighborhood, where w and h represent the width and height of the map respectively, α is the learning rate and t is the current epoch. The retracting radius ensures there is no variation with how the neighborhood is calculated between descriptors, increasing the systems modularity and adaptivity. This is as opposed to more feature dependent methods such as a Gaussian neighborhood, which requires a large sigma value to be effective as it has to describe the input dataset as a whole. The learning rate uses a logarithmic equation:

$$\alpha(t) = \left(0.9\left(\frac{\lambda}{100}\right)\right) \Big/ \left(\left(\frac{\lambda}{100}\right) + t\right) \tag{2}$$

where t is the current epoch and λ is the maximum number of epochs the training period will run for. This starts high at 0.9 to quickly move neurons into position, but quickly dampens the map's ability to learn, entering the so called 'tuning phase'. If the map is to be updated with new inputs for an indeterminate period of time, as t tends toward infinity, the learning rate tends towards a minimum value above zero, effectively extending the tuning phase for all inputs past a certain epoch.

3.4 Algorithm

The LSOM algorithm (Algorithm 1) is a derived from the original SOM algorithm. The number of epochs varies according to the number of inputs and size of the map. However we found approximately 10,000 iterations gives good results for a map with 2,000 inputs and neurons on each layer. An input is taken at random from the input dataset and the BMU on L_1 by feature distance and its neighborhood are moved towards it using the learning restraint as a multiplier. The BMU for the next layer is selected from the neurons interlinked to the BMU on the previous layer.

4 Clustering and Visualization

4.1 Clustering

Clustering on the LSOM is performed by applying an adapted Watershed transformation [12] to the Unified Distance Matrix (U-Matrix) – a matrix of scalar values representing the average distance of each neuron from its immediate neighborhood. Clustering is performed on the N th layer of the LSOM as it represents the most ordered point, rendering explicit consideration of prior layers unnecessary. Clustering the trained map in a way that is semantically useful to humans is arguably the most difficult area to achieve a high level of accuracy since the algorithm will cluster based solely on differences between the descriptors leading to a high risk of under clustering. In an attempt to negate this, 'flooding' points, local U-Matrix minima that form the basis for each cluster are reduced prior to the application of the transformation by merging through differential smoothing of the U-Matrix. A reaction-diffusion model can be used to achieve nonlinear smoothing; this merges small clusters together whilst retaining the boundaries of larger ones. The transformation is applied to the unsmoothed matrix, with the minima equivalent to the remaining post-smoothed minima selected as the flooding points, ignoring all others. Neurons are incrementally added to neighboring clusters as if water were 'flooding' the structure from the selected minima. Where two clusters meet, the dividing neuron, known as a boundary neuron, is left unclustered until the algorithm has terminated where they are then sequentially assigned to the same cluster as their closest clustered topographical neighbor, starting with the smallest feature distance.

4.2 Relationship Graph

Discovered relationships by the LSOM can be used to create a graph detailing similar images for visualization and image retrieval in $O(1)$ time. The graph structure is identical to Layer N of the LSOM. Each node is assigned the image(s) closest to the equivalent neuron on the output layer by feature distance. It is often to be expected that a single image may be assigned to more than one node, however a large number of duplicates may be a sign of undertraining. A more useful relationship graph can be constructed by ranking images to each node according to their distance,

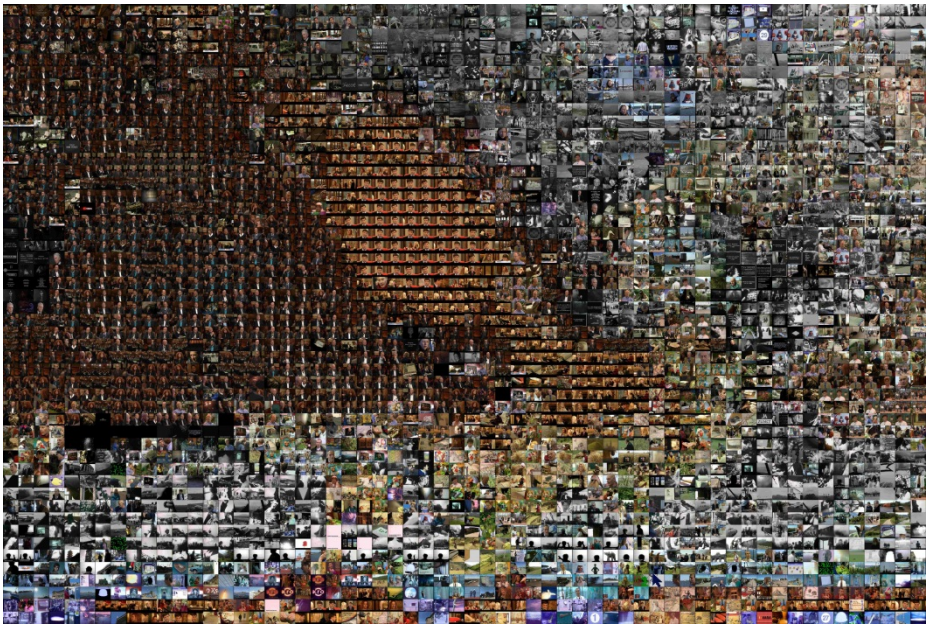

Fig. 1. Example visualization of relationship graph from LSOM with Homogenous Texture Descriptor (HTD) - Color Layout Descriptor (CLD) for 4,000 key frame images

then assigning the images according to their rank once only until all nodes have an associated image. An example visualization of 4,000 key frames is shown in Figure 2.

5 Experimental Results

Experimentation was performed using LSOMs with various layers and orderings, with each processing three image datasets (1,500 images from the Corel Gallery, 26,892 key frame images, and 2,000 color textures). The neurons in the resulting LSOMs were then clustered using the method described in section 4.1 and compared against the results from a CBIR search using the same descriptors combined using ranking. Comparison was made on the basis of how well a ground truth clustered using a rate of recall and precision metrics. The recall rate is given by the formula:

$$R_k = N/k \tag{3}$$

where N is the number of images in the ground truth in the main cluster(s), and k is the total number of images in the ground truth. The precision rate is given by:

$$P = (N + S)/L \tag{4}$$

where S is the number of similar images in the main cluster(s) of the ground-truth, but themselves are not part of the ground truth, and L is the number of images in the labeled area, including irrelevant images.

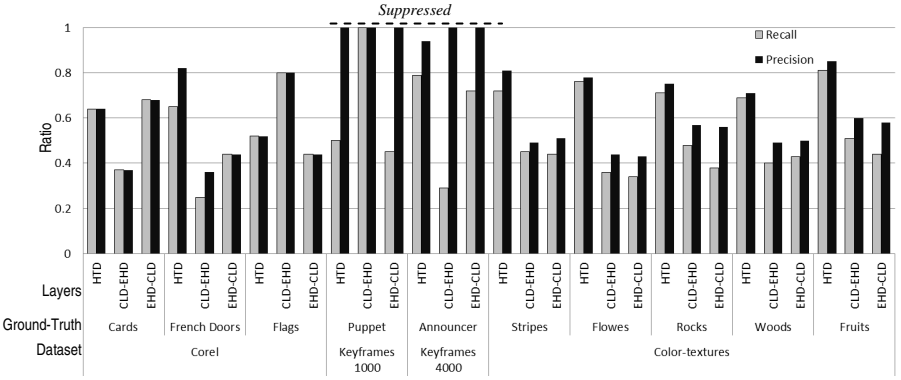

Fig. 2. Experimental results

Table 1. Comparison of Methods by Precision and Recall

Methods	LSOM	CBIR
Recall	56	55
Precision	67 (86*)	73

*Unsuppressed result Ave. [%]

5.1 Corel Dataset

A set of 15 clusters were selected from the Corel Gallery dataset, with three selected as ground-truths to measure retrieval rate and precision: 'cards,' 'French doors' and 'flags.' The 1,500 images were sorted using a 50×30 LSOM. Expectedly, each ground-truth was categorized best by a different LSOM. Since the clusters were selected to represent a broad range of clusters with little overlap, it is unsurprising that the different features required different descriptor combinations to categorize their most prominent features. Further to this, the Corel Gallery also exhibited near identical precision and recall rates, as the clusters were too dissimilar to result in significant entropy.

5.2 Key Frames Dataset

The key frames dataset employed in [13] was formed from a group of television broadcast stills. As a test of the LSOM scalability, the results were run twice, once with a 1,000 images in a 40×25 LSOM, and again with a 4,000 image 80×50 map. While small clusters in the smaller map clustered very well, in one instance at 100% with the Color Layout-Canny EHD arrangement, they fared significantly less well in the larger LSOM. However this was compensated by retention of a high precision rate, for example for 'puppets', a 355% precision rate was obtained and suppressed to 100%, as seen in Figure 3, suggesting that the LSOM finds similarity amongst images on a macro level. Observation returned the notion that although many images were not connected in a way which satisfied the clustering condition, they were to be found in the same region.

5.3 Color Textures

Color textures consisted of 2,000 textured images selected from the Corel Gallery dataset and were classified with a 50x40 map. Unexpectedly, images containing approximately regular or directional patterns were well clustered. However, random textures such as clouds were often spread to different clusters. This effect is due to the Homogeneous Texture Descriptor being unsuitable for handling random patterns. On occasion, directional patterns like check or brick were also misclustered, suggesting the spatial frequency or directionality alterations would benefit from additional features such as steerable filters [14].

6 Conclusions

This paper proposes the Layered SOM (LSOM), a new form of ANN derived from the Self Organizing Feature Map for classifying images of arbitrary image domains. From the results of the conducted experiments on each domain containing thousands of images as well as from empirical observations, the findings can be summarized as follows:

On the basis of the achieved results we clearly attained a higher level of meaningful clustering and classification of the images. In many of the instances where falls in recall rates were observed, an increase in precision was also witnessed. Images in the ground truths may not have conformed to the clustering condition yet were still placed in the same region of the map, mixed with other images exhibiting similar properties, suggesting the LSOM forms clusters based on overriding themes rather than specifics. This is further supported during the scalability test where small ground truths clustered very well – in one case perfectly – for the smaller LSOM, yet represented a much smaller percentage of the larger map's area, and were scattered accordingly. Based on this the LSOM would lend itself well to data labeling. Compared to CBIR retrieval (Table 1), the LSOM exhibited only slightly better recall rates as the measurement was unable to account for when images gravitated towards similar clusters elsewhere in the map. While the same is true for similar images being clustered elsewhere, perhaps with greater accuracy, the LSOM precision still exhibits significantly better precision than CBIR supporting the notion that the LSOM clusters themes rather than specific features. The ordering of the layers and selection of descriptors is crucial to the success of the LSOM; and these choices depending very much on the type of data being analyzed. In several instances it was observed that while the measured ground truths did not cluster well, others clustered with a much greater degree of accuracy; this was particularly apparent with the Corel Gallery dataset, where a number of unrelated classes were categorizes together. The LSOM values in Table 1 have been impacted due to averaging including non-optimal layer arrangements. Automatic feature selection and layer ordering would make for interesting future work. The power of the LSOM lies in its flexibility to categorize large quantities of data implicitly in its topological structure, allowing new data to be incrementally added expending relatively few computational resources, as not all images need to be considered during the training period in order to be classified. In addition, the LSOM need only

be trained once, yet can be queried multiple times, using it's topology to find similar data points, as opposed to distance retrieval where new distances need to be calculated or compared between every image in the dataset per query. As a framework, the LSOM approach can be applied to any general case where data features are compared using a distance function, with initial experimentation in the domain of one-dimensional signals exhibiting favorable results.

References

1. Majunath, B.S., Salembier, P.H., Sikora, T.: Introduction to MPEG-7. Wiley (2002)
2. Kohonen, T.: Self-organizing map, 3rd edn. Springer (2000)
3. Kirk, J.S., Chang, D.-J.: Zurada. J.M.: A self-organizing map with dynamic architecture for efficient color quantization. In: Proceedings of International Joint Conference on Neural Networks (IJCNN), vol. 3, pp. 2128–2132 (2001)
4. Arias, S.: Gomez. H.: Satellite Image Classification by Self-Organized Map on GRID Computing Infrastructures. In: Proceedings of the Second EELA-2 (2009)
5. Lu, S., Segall, R.S.: Multi-SOM: an Algorithm for High-Dimensional, Small Size Datasets. Journal of Systemics, Cybernetics and Informatics 11(2), 41–46 (2013)
6. Olteanu, M., Villa-Vialaneix, N., Cierco-Ayrolles, C.: Multiple Kernel Self-Organizing Maps. In: European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (2013)
7. Martín-Merino, M., Muñoz, A.: Extending the SOM Algorithm to Visualize Word Relationships. In: Famili, A.F., Kok, J.N., Peña, J.M., Siebes, A., Feelders, A. (eds.) IDA 2005. LNCS, vol. 3646, pp. 228–238. Springer, Heidelberg (2005)
8. Breiteneder, C., Merkl, D., Eidenberger, H.: Merging Image Features by Self-organizing Maps in Coats of Arms Retrieval. In: Proceedings of European Conference on Electronic Imaging and the Visual Arts, Berlin, Germany (1999)
9. Rahman, M.: Image Search in a Visual Concept Feature Space with SOM-Based Clustering and Modified Inverted Indexing. In: Application and Novel algorithm Design, pp. 173–188. Intech. (2011)
10. Oja, E., Laaksonen, J., Koskela, M., Brandt, S.: Self-Organizing Maps for Content-Based Image Database Retrieval. In: Kohonen Maps, pp. 349–362. Elsevier (1999)
11. Huang, J., Ravi Kumar, S., Mitra, M., Zhu, W.-J., Zabih, R.: Image Indexing Using Color Correlograms. In: Proceedings of IEEE Computer Vision and Pattern Recognition, pp. 762–768 (1997)
12. Vincent, L., Soille, P.: Watersheds in Digital Spaces: An Efficient Algorithm Based on Immersion Simulations. IEEE Transactions on Pattern Analysis and Machine Intelligence 3(6), 583–598 (1991)
13. Naito, M., Hoashi, K., Matsumoto, K., Shishibori, M., Kita, K., Kutics, A., Nakagawa, A., Sugaya, F., Nakajima, Y.: High-Level Feature Extraction Experiments for TRECVID 2007. In: TRECVID 2007 (2007)
14. Kondo, I., Kutics, A., Tanaka, H., Sakano, H.: A new texture descriptor using steerable filters, In: Technical Report of IEICE, Vol. 104, No. 573 (PRMU2004), pp. 13-18 (2004)