Journey on Image Clustering Based on Color Composition

Achmad Nizar Hidayanto*, Elisabeth Martha Koeanan*

Abstract**—**Image clustering is a process of grouping images based on their similarity. The image clustering usually uses the color component, texture, edge, shape, or mixture of two components, etc. This research aims to explore image clustering using color composition. In order to complete this image clustering, three main components should be considered, which are color space, image representation (feature extraction), and clustering method itself. We aim to explore which composition of these factors will produce the best clustering results by combining various techniques from the three components. The color spaces use RGB, HSV, and L*a*b* method. The image representations use Histogram and Gaussian Mixture Model (GMM), whereas the clustering methods use K-Means and Agglomerative Hierarchical Clustering algorithm. The results of the experiment show that GMM representation is better combined with RGB and L*a*b* color space, whereas Histogram is better combined with HSV. The experiments also show that K-Means is better than Agglomerative Hierarchical for images clustering.

Keywords**—**Image clustering, feature extraction, RGB, HSV, L*a*b*, Gaussian Mixture Model (GMM), histogram, Agglomerative Hierarchical Clustering (AHC), K-Means, Expectation-Maximization (EM).

I. Introduction

INDONESIA is a rich country in cultural heritages. One of them is Batik cloth which has various patterns and colors. As part of the cultural preservation, the need for creating a repository that becomes a reference collection of Batik is increasing. The repository requires functions such as image retrieval that can help users to automatically search particular cloth in the repository. However, retrieving images from a repository is quite time consuming as system should process a large of image data.

In order to improve the efficiency and give better semantic to the image, some researchers such as Chen [1], Liu [2], Guan [3], Kim [4], Park [5], Liu [6], Fakouri [7] apply clustering algorithm for managing images before they can be retrieved. Image clustering is a process of grouping images based on their similarity. By clustering image, the retrieval process does not need to examine images one by one to match with the user query. The system just needs to compare user query with the centroid of the clusters, then returns all images belong to the matched cluster.

This research aims to explore some components of image clustering methods in order to get the best component that will improve the quality of image clustering results. Image clustering based on image content usually uses the color composition, texture, edge, shape, or mixture of two components, etc. This research focuses on image clustering by using color component as previous research result by [8] shows that the use of color composition produces the best result in Batik retrieval.

Three main components regarding the image clustering we considered in this research are color space, image representation (feature extraction), and clustering method. The color spaces use RGB, HSV, and L*a*b* method. The image representations use Histogram and Gaussian Mixture Model (GMM), whereas the clustering methods use K-Means and Agglomerative Hierarchical Clustering algorithm. We expect that through this research we have the best combination of methods of each component that will produce the best clustering results.

Our contributions in this research are twofold:

- First, we compare the image clustering algorithms based on color composition in comprehensive manner by regarding influenced components which are color space, image representation and clustering methods.
- Second, we evaluate the best combination that produces the best image clustering results

In the next section we present a rapid overview of the backgrounds and related works. In section 3, 4 and 5, we present the theoretical foundations of this research which are color space, image representation and clustering methods respectively. In section 6, we give scenario of our experiments and present their result analysis. Finally we summarize our contribution and outline future work in section 7.

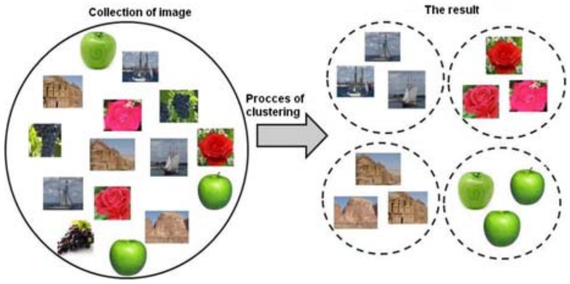


Fig. 1. Ilustration of Image Clustering

II. Background and Related Works

Clustering is a process of classifying to a pattern (data, feature vector) into several groups or clusters based on similarity [9]. Intuitively, members within a cluster have more similar pattern than members of other clusters. While image clustering is a process that divides or classifies a set of images

*Faculty of Computer Science, University of Indonesia. e-mail: nizar@cs.ui.ac.id

into different parts, in which images within a part have similarity each others (homogeneous). Figure 1 shows the illustration of image clustering.

To perform image clustering, we have to go through several steps. The first step that should we do is choosing appropriate representation space. Second, we have to match each image to the selected representation space using appropriate distance measure (similarity measure). At last, the image clustering is then performed either in a supervised process or in an unsupervised process.

In a supervised clustering, images are clustered using human intervention. Huang et al. [10] proposed a method for hierarchical classification of images using supervised learning, provided a training set of images with known class labels is available. Carson et al. [11] used a naive Bayes approach to categorize images. The images are represented by a set of homogeneous regions in feature space of color and texture, based on the “Blobworld" image representation (Carson et al. [12]). Sheikholeslami et al. [13] investigated other approach using a feature-based approach to cluster and retrieve image. Images are clustered on the basis of their similarity to a set of iconic images termed cluster icons. Greenspan et al. [14] introduced a probabilistic and continuous framework for supervised image category modeling and matching.

On other hand, an unsupervised clustering relies on the similarity between the images and the various cluster centers. Chen et al. [15] proposed global image representation using global color, texture and edge histograms. The study on color histograms advantages and disadvantages and its variations can be found in Pass and Zabih [16], Stricker and Dimai [17], and Huang et al. [18]. Barnard et al. [19, 20] and Vailaya et al. [21] suggested a hierarchical model of image clustering which imposes coarse to fine structure on the image collection. The SIMPLIcity system [22] classifies images into graph, textured photograph, or non-textured photograph, and thus narrows down the searching space in images collection. Another algorithm uses the Information Bottleneck (IB) method for unsupervised clustering of image databases. The IB method was introduced by Tishby et al. [23] as a method for solving the problem of unsupervised data clustering and data classification. This method was demonstrated, so far, in the unsupervised classification of discrete data representations for documents [24, 25], galaxies [26] and neural codes [27]. Gordon [28] extends the IB method by using GMM image representation that showed its outstanding performance in clustering images.

The main drawback of supervised clustering is that it requires human intervention. In order to extract the cluster representation, the methods require a-priori knowledge regarding the collection. In contrast the unsupervised clustering algorithms are capable to perform image clustering in fully automatic. Thus, unsupervised clustering provides more flexible way for clustering images. Therefore, in this research we compare the performance of some unsupervised image clustering algorithms.

III. COLOR SPACE

In general, image processing uses RGB color space in image representation, but sometimes we want to use another color space for particular reason. In this research, beside RGB we also explore HSV and L*a*b* color space.

RGB color space represents each pixel by using red (*r*), green (g) and blue (b) coordinate. Each color component is represented by one or more bytes. For example, we represent each color component by using one byte of data, then we have 2^8 = 256 possible values for red, green and blue. Thus, the combination of these color components will produce (256 x 256 x 256) possible values of colors. In general, RGB color space can be represented by $r, g, b \in$ [min, max] where *max* is maximal value of *r*, *g*, *b* and *min* is minimal value of *r*, *g*, *b*.

Another color space is HSV. HSV color space is represented by using *hue* $h \in [0,360]$, *saturation,* and *value*. RGB color space can be transformed to HSV color space by using Equation 1 [29].

$$h = \begin{cases} 0, & \mathit{if} \max = \min \\ (60^0 x \dfrac{g-b}{\max - \min} + 360^0) \;\; \mathrm{mod} \;\; 360^0, & \mathit{if} \max = r \\ 60^0 x \dfrac{b-r}{\max - \min} + 120^0, & \mathit{if} \max = g \\ 60^0 x \dfrac{r-g}{\max - \min} + 240^0, & \mathit{if} \max = b \end{cases}$$

$$s = \begin{cases} 0, & \mathit{if} \max = 0 \\ \dfrac{\max - \min}{\max} = 1 - \dfrac{\min}{\max}, & \mathit{otherwise} \end{cases} \qquad (1)$$

$$v = \max$$

The last color space we used in this research is L*a*b*. L*a*b* color space is XYZ color space with lighting. RGB to XYZ color space transformation can be seen in Equation 2 [28].

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.49 & 0.31 & 0.2 \\ 0.18 & 0.81 & 0.01 \\ 0 & 0.01 & 0.99 \end{bmatrix} \begin{bmatrix} r \\ g \\ b \end{bmatrix} \qquad (2)$$

Then, XYZ to L*a*b* transformation is presented in Equation 3.

$$\begin{aligned} L &= 116 \left(\frac{Y}{Y_n} \right)^{1/3} - 16 \\ a &= 500 \left[\left(\frac{X}{X_n} \right)^{1/3} - \left(\frac{Y}{Y_n} \right)^{1/3} \right] \\ b &= 200 \left[\left(\frac{Y}{Y_n} \right)^{1/3} - \left(\frac{Z}{Z_n} \right)^{1/3} \right] \end{aligned} \qquad (3)$$

IV. IMAGE REPRESENTATION

The image representations we explore in this research using Histogram and Gaussian Mixture Model (GMM). Histogram is color distribution with calculation of pixels of an image. Equation 4 shows histogram formally.

$$h_{A,B,C} = N\, x\, Prob(A{=}a, B{=}b, C{=}c) \qquad (4)$$

In this research, histogram bins the elements of image per channel (red channel, green, and blue for RGB color space; *hue, saturation,* and *value* for HSV; lighting, green-red, blue-yellow for L*a*b*) into 10 equally spaced containers and returns the number of elements in each container. 10 elements of each channel of one color space are concated. So, result for an image with histogram distribution is a vector with 30 elements.

We also experimented on GMM as previous research from Vasconcelos et al [30] showed that GMM is outperform other images representations like color histograms and color correlograms [18]. The foundation of GMM is *Gaussian* function which is shown in Equation 5.

$$f(x) = ae^{-\frac{(x-b)^2}{2c^2}} \tag{5}$$

Suppose $a = \frac{1}{\delta(2\pi)^{1/2}}$, $b=\mu$, and $c=\delta$, where δ and μ are variance and mean respectively, then the *Gaussian* function can be written in Equation 6.

$$f(x) = \frac{1}{\delta(2\pi)^{1/2}} \cdot \exp\{-\frac{(x-\mu)^2}{2\delta^2}\} \tag{6}$$

GMM is mixture model using Gaussian distribution. For example, given X data and number of mixture M, then we can arrange the data using Gaussian distribution from M mixture [28]. Gaussian distribution can be obtained from Equation 7 where $\alpha_j>0$ and $\sum_{j=1}^{k}\alpha_j = 1$, μ_j is *mean*, $\sum_j$ is *covariance* and y is image (matrix).

$$f(y) = \sum_{j=1}^{k}\alpha_j \frac{1}{\sqrt{(2\pi)^d |\Sigma_j|}} \exp\{-\frac{1}{2}(y-\mu_j)^T \sum_j^{-1}(y-\mu_j)\} \tag{7}$$

In reality, GMM representation has similarity with K-means. The difference between both of them is each centroid in GMM uses probability, mean, and variance, whereas K-means uses one parameter as centroid. Gaussian distribution used to determine the distribute of each pixel into centroid. This research uses 10 centroid.

GMM uses Expectation Maximization (EM) algorithm. EM algorithm consist of expectation step and maximization step. Expectation step expects a value that can be seen in Equation 8. Whereas maximization step is to maximize the expectation value (Equation 9) [28]. The followings are the outline of EM algorithm.

1. Expectation step (8)

$$w_{tj} = \frac{\alpha_j f(y_t \mid \mu_j, \Sigma_j)}{\sum_{i=1}^{k}\alpha_i f(y_t \mid \mu_i, \Sigma_i)} \qquad j = 1,\ldots,k \quad , \quad t = 1,\ldots,n \tag{8}$$

2. *Maximization step* (9)

$$\begin{aligned} \hat{\alpha}_j &\leftarrow \frac{1}{n}\sum_{t=1}^{n} w_{tj} \\ \hat{\mu}_j &\leftarrow \frac{\sum_{t=1}^{n} w_{tj} y_t}{\sum_{t=1}^{n} w_{tj}} \\ \hat{\Sigma}_j &\leftarrow \frac{\sum_{t=1}^{n} w_{tj}(y_t - \hat{\mu}_j)(y_t - \hat{\mu}_j)^T}{\sum_{t=1}^{n} w_{tj}} \end{aligned} \tag{9}$$

In this case, we need to initialize the probability, mean, and variance as follow:
- $a_j = M_{1xk}/k$ *(M_{1xk} : vector of ones)*
- $\mu_j = k*max(j)/(k+1)$
- $\sum_j = M_{1xk}* max(j)$

The updating process is repeated until log-likelihood is increased by less than a predefined threshold. In this work, we choose to converge based on the log-likelihood measure and we use 1000 threshold. Log-likelihood is logarithm from sum of posterior probability (W_{tj}) per centroid.

V. CLUSTERING METHOD

The clustering methods we used in this research are K-Means and Agglomerative Hierarchical Clustering [31]. The followings are the steps of Hierarchical Agglomerative Clustering method:

1. choose *k* as number of cluster
2. every data is a cluster *(trivial)*. If there are *N* number of data, and *c* is number of cluster*,* then *c=N*
3. calculate the distance between cluster using one of distance equation (Euclidean distance, Equation 10)
4. search two clusters with minimal distance and cluster it. *c*=*c*-1
5. if *c>k,* back to step 3.

$$d_{rs}^2 = (x_r - x_s)(x_r - x_s)' \tag{10}$$

The K-Means algorithm for *k*-cluster is presented as follow [31]:

1. choose *k* data/pattern randomly for *k* centroid as initial
2. enter all of data to the nearest centroid (using Euclidean distance, Equation 10)
3. calculate new centroid of one cluster using the current member of cluster
4. repeat step 2 to 3 until position of new centroid and old centroid are similar.

VI. EXPERIMENT RESULTS AND ANALYSIS

As we explained in previous section, this research focuses in three main components of image clustering which are color space, image representation, and clustering method. For color spaces, we use RGB, HSV, and L*a*b*. For image representations, we use Histogram and Gaussian Mixture Model (GMM). For the clustering methods, we used K-Means and Agglomerative Hierarchical Clustering. The combination of these components produces 12 experiment scenarios as can be seen in Figure 2. For instance, a scenario may involve the use of RGB color space combined with GMM representation and K-means algorithm.

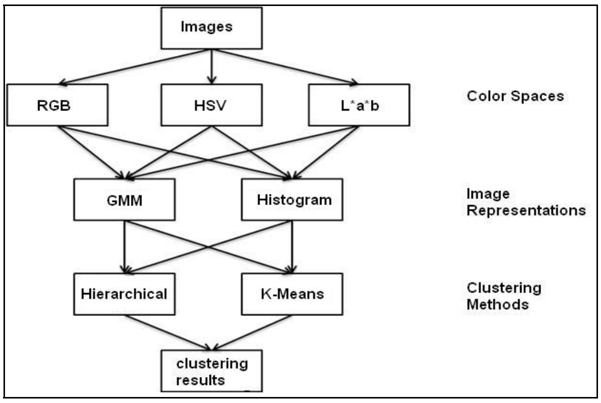


Fig. 2. Scenario of experiments

In the experiments, we use collection of 20 image categories, which are red rose, trumpet, apple, desert, penguin, fire works, horse, sea, grape, sun flower, beach, panda, grass, sunset, jasmine, fish, strawberry, building, tiger, and banana. We vary number of categories in the experiments namely 5, 10, 16, and 20 categories. 5 categories consist of red rose, trumpet, apple, desert, and penguin. 10 categories consist of red rose, trumpet, apple, desert, penguin, fire works, horse, sea, grape, and sun flower. 16 categories consist of red rose, trumpet, apple, desert, penguin, fire works, horse, sea, grape, sun flower, beach, panda, grass, sunset, jasmine, and fish. We select the most distinguished categories for 5 categories in term of color composition. The level of distinction in term of color composition is lesser as the number of categories is increasing. We aim that we can evaluate the effect of color composition in each images to performance of the image clustering. Our hypothesis states that if we vary the color composition in image collection, then it will lessen the performance of image clustering.

We measure the performance of the clustering algorithms by computing their *quality*. The quality is defined as percentage of images which are correctly clustered. If c is correctly clustered images and n is number of images in the collection, then the quality q of the clustering algorithm is $q = (c/a)*100$.

Figures 3 and 4 show the results of the experiments using GMM and histogram respectively. In this experiment we use K-Means clustering algorithm. Figures 5 and 6 show the results of the experiments with scenerio as in Figure 3 and 4 except that we replace K-Means algorithm with Agglomerative Hierarchical Clustering (AHC).

As can be seen in Figure 3, the quality of clustering algorithm is decreasing when we increase number of image category. Figure 3 also shows that GMM representation offers better performance under RGB and L*a*b* color space. The averages of clustering quality using RGB, HSV, L*a*b* color spaces are 55.76%, 37.28%, and 58.16% respectively under GMM representation.

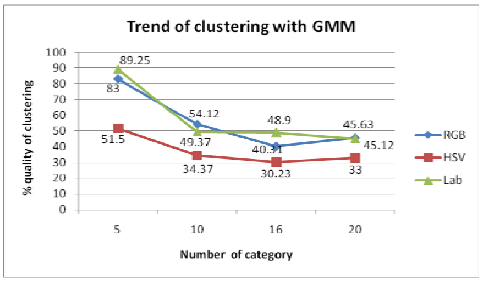


Fig. 3. Clustering results under GMM, K-Means and various color spaces

Figure 4 also exhibits similar trend as in Figure 3. The quality of clustering algorithm is decreasing as we increase number of image category. Figure 4 also shows that Histogram representation will provide better clustering quality if we combine it with HSV color space. The averages of clustering quality using RGB, HSV, L*a*b* color spaces are 50.57%, 57.43%, and 47.54% respectively under Histogram representation.

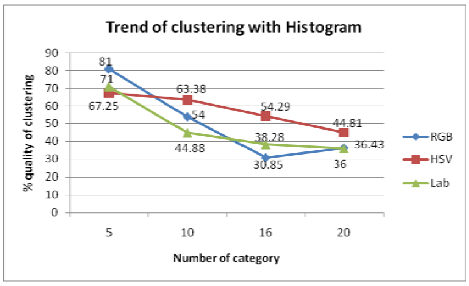


Fig. 4. Clustering results under Histogram, K-Means and various color spaces

Figure 4 shows histogram representation is better combined with HSV color space. Average of every color space with different number of category (5, 10, 16, 20) and histogram representation which RGB, HSV, L*a*b* are 50.57%, 57.43%, and 47.54%.

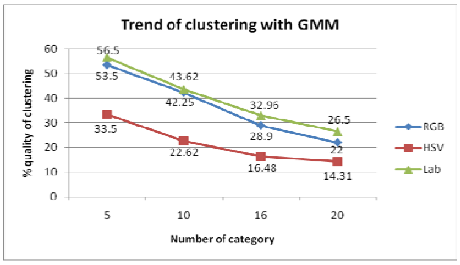


Fig. 5. Clustering results under GMM, AHC and various color spaces

Figure 5 shows image clustering quality with GMM representation and AHC. Figure 5 also shows similar trend with Figure 3 and 4. Quality of clustering is decreasing as the number of image category is increased. Figure 5 also tells us that GMM representation offers better performance by using RGB and L*a*b* color space when combined with AHC. The averages of clustering quality using RGB, HSV, L*a*b* color spaces are 36.66%, 21.72%, and 39.89% respectively under GMM representation.

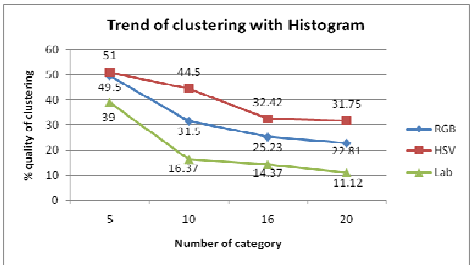


Fig. 6. Clustering results under Histogram, AHC and various color spaces

Figure 6 shows image clustering quality using Histogram representation and AHC. In this scenario, the best performance is offered by HSV color space. The averages of clustering quality using RGB, HSV, L*a*b* color spaces are 32.26%, 39.91%, and 20.21% respectively under Histogram representation.

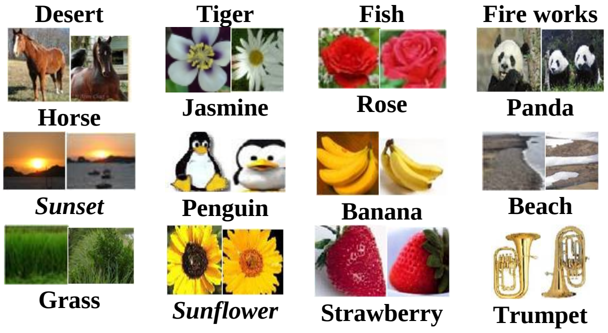
The experiment results in Figure 3, 4, 5 and 6 share similar trend over number of image category and confirm our hypothesis that if image collection shares similar color composition then the quality of clustering algorithm will decrease.

VII. CONCLUSION

This research aims to compare some image clustering algorithms based on color composition. In the image clustering methods, there are three main components that will influence the performance of the algorithm, which are color spaces, image representations and clustering methods. Experiment results show that we will have different performance as we combine methods in each component. Image clustering by combining L*a*b color space, GMM representation and K-Means outperforms other combinations which are showed by the highest average quality score (58.16%). When GMM representation is considered, it is better to combine it with RGB or L*a*b color space. Whereas when the Histogram representation is considered, we better combine it with HSV color space. The experiments also conclude that the K-Means algorithm offers better performance than the Agglomerative Hierarchical clustering. At last, our experiments also confirm that the more clusters share similar color composition then the more difficult for the system to cluster the images correctly.

In the future, we plan to extend our experiment to cluster our Batik clothes. We also extend the experiment by including other aspects such as texture, edge, and so forth.

APPENDIX



REFERENCES

[1] Y. Chen, J.Z. Wang, R. Krovetz, “Content-Based Image Retrieval by Clustering”, Proceedings of the 5th ACM SIGMM international workshop on Multimedia information retrieval, 2003, pp. 193-200.

[2] R. Liu, Y. Wang, T. Baba, Y. Uehara, D. Masumoto and S. Nagata, “SVM-Based Active Feedback in Image Retrieval Using Clustering and Unlabeled Data. LNCS, Computer Analysis of Images and Patterns”, Springer Berlin / Heidelberg, Volume 4673/2007, August 2007, pp. 954-961.

[3] J. Guan, G. Qiu, “Spectral images and features co-clustering with application to content-based image retrieval”, In Proc. of IEEE Workshop on Multimedia Signal Processing, 2005.

[4] D. Kim, “Qcluster: Relevance Feedback Using Adaptive Clustering for Content-Based Image Retrieval”, In Proc. of the ACM SIGMOD Int. Conf. on Management of Data, 2003.

[5] S. Park, K. Seo, D. Jang, ”Fuzzy Art-Based Image Clustering Method for Content-Based Image Retrieval”, International Journal of Information Technology and Decision Making, 06(02), 2007.

[6] Y. Liu, X. Chen, C. Zhang, A. Sprague, “An Interactive Region-Based Image Clustering and Retrieval Platform”, In Proc. of the IEEE International Conference on Multimedia and Expo, 2006, pp. 929-932.

[7] R. Fakouri, B. Zamani, M. Fathy, and B. Minaei, “Region-Based Image Clustering and Retrieval Using Fuzzy Similarity and Relevance Feedback”, In Proc. Of the International Conference on Computer and Electrical Engineering, 2008.

[8] E. Margaretha, H.M. Manurung, “Multimedia Information Processing. Technical report”, Faculty of Computer Science University of Indonesia, 2009.

[9] A. K. Jain, M. N. Murty, P. J. Flynn, “Data Clustering: A Review”, in ACM Computer Survey, 1999, pp 264-323.

[10] J. Huang, S. R. Kumar, and R. Zabith, “An automatic hierarchical image classification scheme”, In ACM Conference on multimedia, England, September 2008, pp. 219-228.

[11] C. Carson, S. Belongie, H. Greenspan, and J. Malik, “Region-based image querying”, In Proc. of the IEEE Workshop on Content-based Access of Image and Video libraries (CVPR'97), 1997, pp. 42-49.

[12] C. Carson, S. Belongie, H. Greenspan, and J. Malik, “Blobworld: Image segmentation using expectation-maximization and its application to image querying”, IEEE Transactions on Pattern Analysis and Machine Intelligence, 24(8):1026-1038, 2002.

[13] G. Sheikholeslami and A. Zhang, “ Approach to clustering large visual databases using wavelet transform”, In Proc. of SPIE conference on visual data exploration and analysis IV, volume 3017, San Jose, California, 1997.

[14] H. Greenspan, J. Goldberger, and L. Ridel, “ A continuous probabilistic framework for image matching”, Journal of Computer Vision and Image Understanding, 84:384-406, 2001.

[15] J. Chen, C.A. Bouman, and J.C. Dalton, “Hierarchical browsing and search of large image databases”, IEEE transactions on Image Processing, 9(3):442-455, March 2000.

[16] G. Pass and R. Zabih, "Comparing images using joint histograms”, Multimedia Systems, 7:234-240, 1999.

[17] M. Stricker and A. Dimai, “Spectral covariance and fuzzy regions for image indexing. Machine Vision and Applications”, 10(2):66-73, 1997.

[18] J. Huang, S. R. Kumar, M. Mitra, W.-J. Zhu, and R. Zabih, ”Image indexing using color correlograms”, In Proc. of the IEEE Comp. Vis. And Patt. Rec., pp. 762-768, 1997.

[19] K. Barnard, P. Duygulu, and D. Forsyth, “Clustering art. In Computer Vision and Pattern Recognition (CVPR 2001)”, Hawaii, December 2001.

[20] K. Barnard and D. Forsyth, “Learning the semantics of words and pictures. In International Conference on Computer Vision”, volume 2, pp. 408-415, 2001.

[21] A. Vailaya, M. A. T. Figueiredo, A. K. Jain, and H.-J. Zhang, “Image Classification for Content-Based Indexing,” *IEEE Trans. Image Processing*, vol. 10, no. 1, pp. 117–130, 2001.

[22] J. Z. Wang, J. Li, and G. Wiederhold, “SIMPLIcity: Semantics-Sensitive Integrated Matching for Picture LIbraries,” *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 23, no. 9, pp. 947–963, 2001.

[23] N. Tishby, F. Pereira, and W. Bialek, “The information bottleneck method”, In Proc. Of the 37-th Annual Allerton Conference on Communication, Control and Computing, pp. 368-377, 1999.

[24] N. Slonim and N. Tishby, :Agglomerative information bottleneck”, In Proc. of Neural Information Processing Systems, pp. 617-623, 1999.

[25] N. Slonim, N. Friedman, and N. Tishby, “Unsupervised document classification using sequential information maximization”, In Proc. of the 25-th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 2002.

[26] N. Slonim, R. Somerville, N. Tishby, and O. Lahav, “Objective classification of galaxy spectra using the information bottleneck method”, 323:270-284, 2001.

[27] E. Schneidman, N. Slonim, N. Tishby, R. R. deRuyter van Steveninck, and W. Bialek, “Analysing neural codes using the information bottleneck method”, In Advances in Neural Information Processing Systems, NIPS, 2001

[28] S. Gordon, “Unsupervised Image Clustering using Probabilistic Continuous Models and Information Theoretic Principle”, Thesis, Universitas Tel-Aviv Israel, 2006.

[29] D. Cardani. Adventures in HSV Space. Available at *http://robotica.itam.mx/espanol/archivos /hsvspace.pdf.* 2006.

[30] N. Vasconcelos and A.Lippman, “Feature representations for image retrieval: Beyond the color histogram”, In Proc. of the Int. Conference on Multimedia and Expo, New York, August 2000.

[31] C. Adi, ”Comparison of Agglomerative Hierarchical Clustering methods for Text Data”, Thesis, Faculty of Computer Science, University of Indonesia.