Abstract

With the explosive advancement in imaging technologies and specially with proliferation of the world wide web, image retrieval has attracted the increasing interests of researches in the field of digital libraries, image processing and database system. Research in human perception of image content suggests that content-based image retrieval (CBIR) can follow a sequence of steps. The typical steps of CBIR system are: image query formation, image feature extraction, similarity measurement, indexing and retrieval, and user interaction. The correct choice and set up for each step will result in a well, efficient and suitable CBIR system.

This work concentrates on one important and crucial step of the whole CBIR system: feature extraction (or feature formation). The image features used are all characterized as low – level features. These include: image luminance histogram, low – passed luminance histogram, luminance pyramid, color pyramid, and combined feature.

The main contributions are: simplicity (i.e. easy to implement the feature extraction phase), suitability (i.e. provide acceptable retrieving results), efficiency, and economy. The CBIR with the presented feature extraction variants are tested on a selected database of a set of thirteen image classes. In general, the results indicate that the choice of image feature can greatly affect the performance of CBIR system. Experimental results showed that image features that utilize achromatic and chromatic information of the image can provide about 75% accurate results, while those depend on only intensity information can give accurate results in about 25% - 75%. Moreover, the combination of two features can give in better results.

List of Contents

List of Figures

List of Tables

List of Abbreviations

Symbol	Name
ANMRR	Average Normalized Modified Retrieval Rank
AVR	Average Rank
Cb	Color Difference for Blue
CBIR	Content-Based Image Retrieval
CIE TCI	Commission International de I'Eclairage Technical Committee Industrial
CIECAM97	Commission International de I'Eclairage Interim Color Appearance Model 1997
Cr	Color Difference for Red
IR	Information Retrieval
KL	Karhunen – Loeve
L	Lightness
MPEG7	
MPEG7	Moving Picture Experts Group
MRR	Modified Retrieval Rank
NMRR	Normalized Modified Retrieval Rank
PCA	principle component analysis
QBE	Query – By – Example
RGB	Red, Green, Blue
TIR	Text Information Retrieval
WWW	World Wide Web
Y	Luminance or Luma

الأسم :خولة عمر فرحان الدباغ
عنوان السكن :بغداد / المنصور
رقم الهاتف :٥٤١٤٣٤٦
تاريخ المناقشة :١٣/ ٣ / ٢٠٠٨
اسم الرسالة :امضاء الصور لأسترجاع الصور بالأعتماد على أساس المحتوى
أسماء المناقشين :د٠ليث عبد العزيز العاني(رئيس اللجنة) ,د٠بشرى قاسم العبودي (عضواً), د٠هيثم عبد اللطيف العاني(عضواً), د٠براء علي عطية (المشرف)

CERTIFICATION

I certify that this thesis was prepared under my supervision at the Department of Computer Science / College of Science / Al – Nahrain University, as partial fulfillment of the requirements for the degree of **Master of Science in Computer Science**.

Signature:

Name: Dr. Bara'a Ali Attea

Title: Assist. Prof.

Date: 10 / 12 / 2007

In view of the available recommendations, I forward this thesis for debate by the examination committee

Signature:

Name: Dr. Taha S. Bashaga

Title: Head of Department of Computer Science

Al – Nahrain University

Date: 10 / 12 / 2007

5.1 Conclusions

The main contribution of this thesis is to use low-level image features for indexing images in CBIR system. The concentrations on designing low-level image features are: easy to implement image signature, economic (requires relatively less memory storage as signature vectors are small in size), fast computation of image signature, and ability of providing acceptable results. The implementation of image signature with these aims in mined draws the following conclusions:

1. On overall and for small database size, the performance of CBIR based on the implemented low-level image signature variation is accepted.

2. Luminance based image signatures can provide CBIR with acceptable results. This is due to that luminance vision is able to detected sharp edges and the fine details of patterns and textures in the image.

3. Image signatures that are extracted from the three image components (i.e. both achromatic and chromatic information) perform better than those achromatic – based image signatures.

4. Subsampling – based image signatures are more better than histogram – based image signature. This comes from the fact that subsampling methods preserve spatial image layouts while histogram – based methods point out occurrence or distribution of intensities within the image.

5. Color subsampling based signatures are more powerful than luminance subsampling based ones. The reason behind this fact is that the first method of subsampling preserves spatial color distribution within the image while the latter preserves only spatial distribution of intensities within the image.

6. Combining two (but more quantized) features in one image signature behaves, at least, as good as the better of the two combined features.

5.2 Future Work

The signature extraction methodologies presented in this thesis still produce some grossly mis – retrieved images, and they are never wholly successful. Thus, there is a room for improvement if the following directions of future work are investigated:-

1. After images indexing via their signatures, signature matching is required. Hence, the key problem is deciding which signature searching technique is the fastest one to break down the efficiency problem when dealing with very large database.

2. Combining these easy-to-implement, economic low-level signatures with more complex mid – or high – level image features to get more better results.

3. Deciding which color model that can give the best similarity metric between two image signatures belonging to two semantically ″very relevant″and ″very irrelevant ″ images.

4. Deciding which image filter that can preserve the most fine and sharp details of a large variety of images, so as to keep this useful information within.

5. One can build a CBIR system with two – stages image signature calculations. The first signature (e.g. combined signature) can be used to retrieve K images from the whole database. Thus, it should give acceptable results but at the same time it should be just to calculate and easy to implement. In the second stage, one can use more complex and powerful image signature to filter out the only relevant images from those K retrieved images.

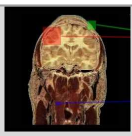
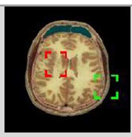
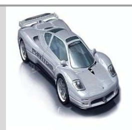
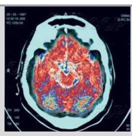
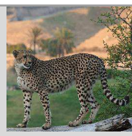
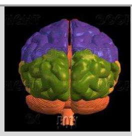
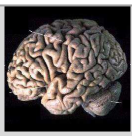
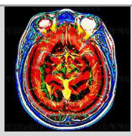
4.1 Introduction

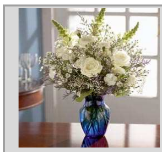
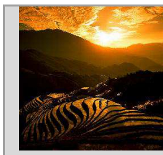
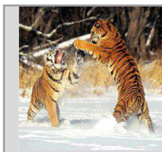
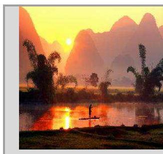
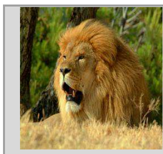
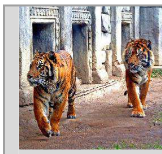
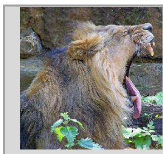
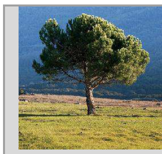
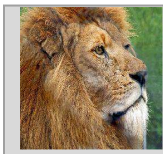
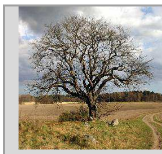
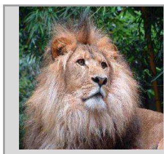
This chapter is intended to test the application of the signature (or feature) generation variants explained in the previous chapter for content-based image retrieval process. The system software was completely written in Matlab (R2007a) Version 1.5.2.

4.2 Image Database and Results

In order to assess the five types of image signature, they are used with the same database of natural images. Although the image database used in the experiments is relatively small in size, but it contains a diverse collection of semantically distinct image classes. This database has 13 semantically distinct classes (including lions, flowers, faces, air planes, etc.) with a total of 63 images, as shown in table (4.1). The images were taken from different image resources available for most CBIR work. Some of these resources are: source image courtesy © Ian Brittan – FreeFoto.com, natural image database of ENSEA University, Vision Human database, courtesy of Scott Chumbley, Corel database and Landsat satellite images.

Table (4.1): Image Database.

Class 1	Class 2	Class 3	Class 4	Class 5	Class 6	Class 7
 Airplane01	 Brain01	 Bug01	 Car01	 Cheetah01	 Cloud01	 Face01
 Airplane02	 Brain02	 Bug02	 Car02	 Cheetah02	 Cloud02	 Face02
	 Brain03	 Bug03	 Car03	 Cheetah03	 Cloud03	 Face03
	 Brain04	 Bug04	 Car04	 Cheetah04	 Cloud04	
	 Brain05		 Car05	 Cheetah05	 Cloud05	
	 Brain06		 Car06	 Cheetah06		
	 Brain07			 Cheetah07		
	 Brain08					
	 Brain09					

Class 8	Class 9	Class 10	Class 11	Class 12	Class 13
 Flower01	 Land01	 Lion01	 Sunset01	 Tiger01	 Tree01
 Flower02	 Land02	 Lion02	 Sunset02	 Tiger02	 Tree02
 Flower03		 Lion03	 Sunset03	 Tiger03	 Tree03
		 Lion04	 Sunset04	 Tiger04	 Tree04
		 Lion05			 Tree05
		 Lion06			 Tree06
		 Lion07			
		 Lion08			

During experiments, the query images were also taken from the database but, of course, each image submitted in a query was temporarily deleted from the database while the query was processed. The results present a total of $Q=13$ different queries using the five image signatures in several variations. In the 1^{st}, 2^{nd}, 3^{rd}, and 4^{th} signature variations the total number of queries was set to $Q=4$. For each query, q, each ground truth image $k \in [1,...,N(q)]$ is assigned a rank value k if it is in the first $K = \min[4N(q), 2M]$ where $M = \max(N(q_1), N(q_2),..., N(q_{13})) = 8$. Table 4.2 illustrates the selected queries and their corresponding number of ground truth images.

Table 4.2: Queries and their number of ground truth images.

Query no.	Query	N(q)	K
1	Airplane01	1	4
2	Brain01	8	16
3	Bug02	3	12
4	Car02	5	16
5	Cheetah02	6	16
6	Cloud01	4	16
7	Face03	2	8
8	Flower03	2	8
9	Land01	1	4
10	Lion02	7	16
11	Sunset02	3	12
12	Tiger02	3	12
13	Tree01	5	16

All results below present only the first ten retrieved results. Following illustrates some of the results obtained throughout experiments. All images shaded with gray color reflect the correct retrieving result within the first 10 images. Additionally, there is an entry at the tail of each result denoting the case of satisfactory completion of retrieving, thus its value can be either successful or unsuccessful. Successful retrieving can be satisfied only if all $N(q)$ images are retrieved within the first K images, otherwise it is unsuccessful retrieving.

4.2.1 Luminance Histogram

Table (4.3a): Results of CBIR using luminance histogram image signature in YIQ color space.

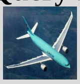
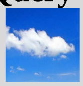
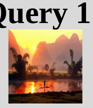
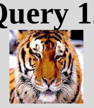
Query 1  **Airplane01**		**Query 8**  **Flower03**		**Query 9**  **Land01**		**Query 10**  **Lion02**	
K = 4	**N(q) = 1**	**K = 8**	**N(q) = 2**	**K = 4**	**N(q) = 1**	**K = 16**	**N(q) = 7**
Result	**Distance**	**Result**	**Distance**	**Result**	**Distance**	**Result**	**Distance**
Bug02	277.45	Flower02	268.64	Cloud04	257.54	Lion04	123.21
Sunset03	363.32	Tiger02	298.25	Land02	258.21	Lion03	139.93
Cloud05	409.07	Flower01	308.18	Lion04	267.31	Tree06	155.21
Cheetah06	456.53	Tree05	313.89	Cheetah06	286.17	Lion06	162.15
Land 01	461.23	Car05	317.42	Lion03	304.70	Cloud04	173.34
Airplane02	471.90	Tiger01	321.23	Tree06	311.31	Cheetah06	188.89
Bug03	479.81	Car01	338.03	Lion02	327.93	Cheetah04	200.20
Tree06	484.45	Face03	354.34	Cheetah05	343.96	Tree03	205.26
Lion03	487.43	Tiger04	356.09	Lion06	345.04	Lion08	207.84
Cloud04	487.53	Tiger03	361.26	Lion05	360.31	Lion05	209.40
Unsuccessful		Successful		Successful		Successful	

Table (4.3b): Results of CBIR using luminance histogram image signature in ℓαβ color space.

Query 2 Brain01		Query 5 Cheetah02		Query 7 Face03		Query 8 Flower03	
K = 16	**N(q) = 8**	**K = 16**	**N(q) = 6**	**K = 8**	**N(q) = 2**	**K = 8**	**N(q) = 2**
Result	**Distance**	**Result**	**Distance**	**Result**	**Distance**	**Result**	**Distance**
Brain05	269.15	Cheetah01	263.26	Sunset04	150.32	Flower02	224.18
Brain03	301.93	Cheetah07	263.31	Face01	173.76	Car01	257.53
Brain07	319.87	Cheetah03	301.40	Lion03	174.09	Tree5	264.60
Brain09	327.23	Lion05	308.26	Lion08	188.95	Car05	280.43
Brain06	336.39	Tree02	309.14	Tree06	191.81	Flower01	280.54
Brain08	350.89	Lion07	319.60	Sunset03	198.01	Tiger02	294.93
Brain02	364.68	Lion01	336.45	Bug03	215.67	Tiger03	296.79
Car05	490.87	Cheetah06	338.31	Tree03	218.10	Tiger01	307.87
Sunset01	492.45	Lion04	381.79	Lion06	224.65	Tiger04	324.85
Face02	500.09	Cheetah04	381.96	Lion02	242.12	Car06	352.32
Unsuccessful		Successful		Unsuccessful		Successful	

Table (4.3c): Results of CBIR using luminance histogram image signature in CIECAM97 color space.

Query 4  **Car02**		**Query 6**  **Cloud01**		**Query 11**  **Sunset02**		**Query 12**  **Tiger02**	
K = 16	**N(q) = 5**	**K = 16**	**N(q) = 4**	**K = 12**	**N(q) = 3**	**K = 12**	**N(q) = 3**
Result	**Distance**	**Result**	**Distance**	**Result**	**Distance**	**Result**	**Distance**
Car01	71.93	Tree04	147.37	Sunset04	67.34	Tiger01	67.26
Tiger03	72.93	Lion08	153.23	Lion08	77.75	Cheetah04	76.04
Tree01	75.15	Tree03	176.10	Tree04	108.15	Car04	82.89
Tree05	82.57	Tiger04	176.68	Car06	121.64	Face02	83.26
Flower01	87.29	Sunset02	179.67	Face01	122.79	Tiger04	84.56
Car03	91.14	Tiger01	179.70	Tiger01	138.34	Lion01	88.51
Flower02	92.07	Sunset04	179.78	Tiger04	138.84	Tree05	90.48
Flower03	92.85	Car04	182.37	Lion02	146.87	Cheetah07	94.42
Land02	100.57	Cloud02	189.43	Car05	146.92	Face03	99.28
Tiger02	104.29	Lion03	189.59	Lion03	150.31	Car05	101.00
Unsuccessful		Unsuccessful		Unsuccessful		Successful	

4.2.2 Low-passed Luminance Histogram

Table (4.4a): Results of CBIR using low – pass luminance histogram image signature in YIQ color space.

Query 3 **Bug02**		**Query 9** **Land01**		**Query 10** **Lion02**		**Query 13** **Tree01**	
K = 12	**N(q) = 3**	**K = 4**	**N(q) = 1**	**K = 16**	**N(q) = 7**	**K = 16**	**N(q) = 5**
Result	**Distance**	**Result**	**Distance**	**Result**	**Distance**	**Result**	**Distance**
Cloud05	216.01	Cloud04	243.50	Lion04	140.37	Tree05	119.37
Airplane01	273.98	Lion04	270.21	Lion03	145.07	Cheetah05	135.37
Airplane02	305.67	Land02	273.12	Cloud04	146.92	Tiger03	148.31
Bug03	310.39	Lion03	285.89	Lion06	183.26	Tree03	172.53
Sunset03	310.90	Cheetah06	294.28	Tree06	186.43	Lion07	188.32
Bug04	334.90	Lion02	333.57	Tree03	189.03	Tiger04	191.50
Lion03	374.42	Cheetah04	335.96	Lion08	193.75	Face02	194.78
Cheetah06	394.51	Cloud02	356.28	Cheetah04	196.18	Lion06	197.20
Lion02	397.45	Tree06	357.98	Cheetah06	196.93	Face03	202.79
Sunset04	407.04	Tree03	362.06	Lion07	205.34	Cheetah01	203.95
Unsuccessful		Successful		Successful		Unsuccessful	

Table (4.4b): Results of CBIR using low – pass luminance histogram image signature in ℓαβ color space.

Query 6 Cloud01		Query 7 Face03		Query 11 Sunset02		Query 12 Tiger02	
K = 16	**N(q) = 4**	**K = 8**	**N(q) = 2**	**K = 12**	**N(q) = 3**	**K = 12**	**N(q) = 3**
Result	**Distance**	**Result**	**Distance**	**Result**	**Distance**	**Result**	**Distance**
Sunset01	294.87	Sunset04	150.28	Face01	278.53	Car05	116.31
Cloud03	315.71	Face01	171.51	Face03	280.67	Tiger04	176.37
Cloud05	404.34	Lion03	185.82	Car05	296.79	Tree05	220.46
Cloud02	409.48	Lion08	186.32	Car06	304.73	Flower01	28312
Cloud04	428.67	Sunset03	190.32	Lion08	305.73	Car06	221.18
Brain04	430.23	Bug03	215.34	Tiger02	328.37	Tiger01	233.82
Airplane02	451.67	Tiger04	228.54	Tree04	330.75	Lion08	235.60
Brain09	507.28	Cheetah05	247.56	Sunset04	348.75	Land02	237.17
Sunset02	538.10	Lion02	255.09	Cloud04	341.53	Face02	245.95
Brain02	543.67	Tree03	266.43	Face02	344.79	Sunset04	250.04
Successful		Unsuccessful		Unsuccessful		Successful	

Table (4.4c): Results of CBIR using low – pass luminance histogram image signature in CIECAM97 color space.

Query 1 Airplane01		**Query 2 Brain01**		**Query 5 Cheetah02**		**Query 8 Flower03**	
K = 4	**N(q) = 1**	**K = 16**	**N(q) = 8**	**K = 16**	**N(q) = 6**	**K = 8**	**N(q) = 2**
Result	**Distance**	**Result**	**Distance**	**Result**	**Distance**	**Result**	**Distance**
Bug02	265.78	Brain07	256.45	Tree05	27.14	Flower02	36.29
Airplane02	302.04	Brain06	275.35	Lion05	43.28	Land02	45.35
Sunset03	339.48	Brain05	295.67	Bug04	93.62	Cheetah01	50.81
Bug03	347.53	Brain03	303.20	Tiger03	101.28	Tree01	55.98
Tree03	367.00	Brain08	352.45	Cheetah04	102.07	Car01	62.03
Tiger04	382.14	Brain09	394.00	Land01	400.89	Flower01	67.20
Cheetah06	384.09	Brain02	417.39	Tree02	108.10	Cloud04	67.79
Cheetah04	403.14	Brain04	428.48	Lion03	113.62	Lion07	72.76
Cheetah05	421.57	Sunset01	431.10	Lion02	125.82	Car02	93.20
Car05	424.31	Car05	499.48	Cheetah06	126.14	Face03	98.56
Successful		Successful		Successful		Unsuccessful	

4.2.3 Luminance pyramid

Table (4.5a): Results of CBIR using luminance pyramid image signature of 8*8 pixels in YIQ color space.

Query 3 Bug02		Query 9 Land01		Query 10 Lion02		Query 13 Tree01	
K = 12	N(q) = 3	K = 4	N(q) = 1	K = 16	N(q) = 7	K = 16	N(q) = 5
Result	**Distance**	**Result**	**Distance**	**Result**	**Distance**	**Result**	**Distance**
Bug04	2.8642	Cheetah07	4.0221	Bug03	3.5652	Cheetah01	3.6756
Airplane01	3.2270	Lion05	4.5262	Bug04	3.8201	Cheetah04	3.7940
Cheetah03	3.6123	Lion04	4.7132	Cheetah03	3.8448	Lion06	4.2250
Airplane02	3.7844	Cheetah05	4.9141	Tree02	4.1427	Car05	4.3265
Cloud05	3.8796	Lion03	4.9748	Tree06	4.2962	Tree05	4.3919
Brain09	4.0663	Land02	4.9754	Lion01	4.4082	Tree03	4.4857
Bug03	4.2503	Lion07	5.2807	Lion08	4.5497	Lion01	4.5356
Bug01	4.2636	Cheetah03	5.3048	Cheetah06	4.6110	Lion05	4.5437
Cheetah02	4.4983	Tree04	5.4128	Airplane01	4.6634	Tree02	4.7040
Tree02	4.6644	Lion06	5.4988	Lion06	4.7762	Tree06	4.7238
Successful		Unsuccessful		Unsuccessful		Unsuccessful	

Table (4.5b): Results of CBIR using luminance pyramid image signature of 8*8 pixels in ℓαβ color space.

Query 4 Car02		Query 5 Cheetah02		Query 6 Cloud01		Query 8 Flower03	
K = 16	**N(q) = 5**	**K = 16**	**N(q) = 6**	**K = 16**	**N(q) = 4**	**K = 8**	**N(q) = 2**
Result	**Distance**	**Result**	**Distance**	**Result**	**Distance**	**Result**	**Distance**
Car01	1.9358	Cheetah03	1.0629	Cloud02	4.6271	Flower02	1.8121
Flower02	2.2945	Cheetah01	1.0938	Cloud03	4.9523	Flower01	2.0985
Car03	2.4433	Tree02	1.3651	Cloud05	6.0967	Car03	2.3067
Flower01	2.5090	Lion05	1.5059	Bug01	7.0229	Cheetah01	2.3627
Tree05	2.7731	Lion06	1.5430	Brain04	7.2754	Car01	2.4044
Cheetah01	2.8749	Cheetah07	1.6111	Airplane01	7.8393	Tiger03	2.4192
Car04	2.8907	Lion04	1.6255	Brain08	7.8435	Lion04	2.6684
Flower03	2.9352	Tree03	1.6258	Brain07	7.9403	Tree05	2.7314
Tiger03	3.0220	Cheetah04	1.6689	Bug02	8.0963	Tree01	2.7605
Tree01	3.0705	Cheetah06	1.6989	Brain06	8.1443	Land02	2.8056
Unsuccessful		Successful		Successful		Successful	

Table (4.5c): Results of CBIR using luminance pyramid image signature of 8*8 pixels in CIECAM97 color space.

Query 1 Airplane01		Query 2 Brain01		Query 5 Cheetah02		Query 11 Sunset02	
K = 4	**N(q) = 1**	**K = 16**	**N(q) = 8**	**K = 16**	**N(q) = 6**	**K = 12**	**N(q) = 3**
Result	**Distance**	**Result**	**Distance**	**Result**	**Distance**	**Result**	**Distance**
Airplane02	4.3168	Brain08	3.7202	Car03	1.8052	Airplane01	4.8715
Tiger04	4.4351	Brain03	4.0334	Cheetah01	1.8856	Lion02	4.9145
Bug03	4.4761	Brain06	4.5020	Car02	1.9874	Airplane02	4.9761
Sunset02	4.8715	Brain02	4.9311	Flower02	2.0111	Sunset04	5.0068
Tree02	5.0311	Brain09	5.5926	Cheetah03	2.1133	Tiger04	5.6576
Car05	5.0336	Brain04	5.9548	Tree02	2.1456	Cheetah04	5.6986
Lion02	5.1036	Brain05	7.0169	Lion01	2.2169	Cheetah02	5.7560
Cheetah05	5.1281	Brain07	7.3204	Tree05	2.2920	Car02	5.7656
Bug02	5.1441	Airplane02	10.8965	Car01	2.2991	Tree02	5.7685
Car02	5.1728	Sunset01	11.0510	Flower01	2.3010	Bug04	5.8177
Successful		Successful		Unsuccessful		Unsuccessful	

4.2.4 Color Pyramid

Table (4.6): Results of CBIR using color pyramid of 8*16 pixels.

Query 1 Airplane01		**Query 3 Bug02**		**Query 6 Cloud01**		**Query 13 Tree01**	
K = 4	**N(q) = 1**	**K = 12**	**N(q) = 3**	**K = 16**	**N(q) = 4**	**K = 16**	**N(q) = 5**
Result	**Distance**	**Result**	**Distance**	**Result**	**Distance**	**Result**	**Distance**
Airplane02	32.9372	Bug04	25.5768	Cloud02	30.5957	Tree02	43.4071
Cheetah02	34.3608	Cheetah03	33.1975	Cloud03	40.4727	Tree05	56.5787
Cheetah03	38.1805	Bug01	36.0708	Cloud04	58.2638	Tree04	59.3139
Cloud05	40.1712	Bug03	37.5542	Land02	62.2912	Airplane02	59.3846
Cheetah06	42.7717	Cheetah02	38.2595	Land01	67.6112	Tree03	59.6205
Lion04	43.1281	Brain06	41.8342	Cloud05	68.6486	Cheetah04	60.4064
Land01	46.5424	Lion03	42.0345	Airplane02	70.7966	Lion01	60.6877
Bug04	46.7499	Brain08	42.4020	Tree02	70.8955	Cheetah02	62.9621
Cheetah07	46.8414	Lion02	43.4599	Tree01	73.8020	Cheetah06	65.5223
Lion02	48.3874	Brain01	44.8557	Airplane01	75.8539	Tiger03	65.7923
Successful		Successful		Successful		Unsuccessful	

Table (4.7): Results of CBIR using YIQ pyramid of 8*16 pixels.

Query 2 Brain01		**Query 5 Cheetah02**		**Query 8 Flower03**		**Query 9 Land01**	
K = 16	**N(q) = 8**	**K = 16**	**N(q) = 6**	**K = 8**	**N(q) = 2**	**K = 4**	**N(q) = 1**
Result	**Distance**	**Result**	**Distance**	**Result**	**Distance**	**Result**	**Distance**
Brain08	11.5924	Cheetah03	6.1901	Flower02	19.5182	Cloud03	17.8369
Brain03	12.0063	Bug04	9.8058	Tree05	23.4253	Cheetah07	20.0813
Brain06	12.7444	Lion01	12.6164	Flower01	24.7460	Land02	20.3677
Brain02	12.8236	Tree02	12.8691	Tiger03	24.8622	Lion05	20.8059
Brain05	14.7497	Cloud05	13.1111	Cheetah01	25.3249	Cloud05	20.9381
Brain09	19.6702	Cheetah07	13.3862	Car01	25.4375	Cheetah05	21.1187
Brain07	19.8368	Bug02	13.5060	Car03	26.2777	Lion04	21.1479
Brain04	20.4649	Lion04	13.6073	Cheetah05	27.6826	Cheetah03	22.2581
Bug02	22.3403	Cheetah04	14.2146	Car04	28.3075	Lion03	22.4491
Bug01	23.1082	Cheetah06	14.4802	Tiger02	28.4574	Cheetah02	22.5499
Successful		Unsuccessful		Successful		Successful	

4.2.5 Combined Signature

Table (4.8a): Results of CBIR using combined signature by combining luminance histogram of 64 elements in YIQ color space with luminance pyramid of 8*8 pixels in YIQ color space.

Query 4 Car02		**Query 8** Flower03		**Query 10** Lion02		**Query 12** Tiger02	
K = 16	**N(q) = 5**	**K = 8**	**N(q) = 2**	**K = 16**	**N(q) = 7**	**K = 12**	**N(q) = 3**
Result	**Distance**	**Result**	**Distance**	**Result**	**Distance**	**Result**	**Distance**
Car03	18087	Flower02	29799	Lion04	15871	Car05	17250
Tiger02	55838	Tiger02	35408	Tree06	19345	Tiger04	18195
Car05	56003	Tree05	37038	Lion03	19387	Face03	24043
Car04	58157	Flower01	37506	Lion06	20864	Tree05	24491
Tiger04	59151	Tiger01	38634	Cloud04	21083	Face02	28618
Tiger01	61013	Car05	39337	Cheetah06	23965	Tiger03	29740
Tiger03	61953	Car01	41364	Lion08	24599	Face01	29742
Car06	63441	Face03	43147	Tree03	26978	Flower01	30190
Tree05	63814	Tiger04	44142	Lion05	27007	Sunset04	30530
Face02	1.93e+003	Tiger03	44506	Cheetah04	27662	Tiger01	31814
Successful		Successful		Successful		Successful	

Table (4.8b): Results of CBIR using combined signature by combining luminance histogram of 64 elements in YIQ color space with color pyramid of 8*8 pixels.

Query 2		**Query 6**		**Query 7**		**Query 9**	
Brain01		**Cloud01**		**Face03**		**Land01**	
K = 16	**N(q) = 8**	**K = 16**	**N(q) = 4**	**K = 8**	**N(q) = 2**	**K = 4**	**N(q) = 1**
Result	**Distance**	**Result**	**Distance**	**Result**	**Distance**	**Result**	**Distance**
Brain07	29837	Cloud02	24854	Tiger04	17348	Cloud04	29255
Brain05	32180	Car06	34660	Lion08	19843	Land02	32327
Brain03	33501	Cloud03	42953	Tree05	20100	Lion04	33244
Brain09	35521	Lion02	46996	Tree03	20427	Cheetah06	34415
Brain04	35707	Cloud04	48141	Sunset04	21798	Tree06	37186
Brain02	37135	Lion08	48539	Cheetah05	22197	Lion03	38673
Sunset01	37436	Tiger04	48621	Face01	22289	Lion02	40433
Brain08	39129	Sunset04	48796	Tiger02	23254	Cheetah05	43512
Brain06	39419	Land01	49417	Face02	23737	Lion06	43523
Car05	57374	Lion03	50049	Lion06	25429	Cloud02	44765
Successful		Successful		Unsuccessful		Successful	

Table (4.8c): Results of CBIR using combined signature by combining luminance pyramid of 8*8 pixels in YIQ color space with color pyramid of 8*8 pixels.

Query 2 Brain01		**Query 3 Bug02**		**Query 4 Car02**		**Query 5 Cheetah02**	
K = 16	**N(q) = 8**	**K = 12**	**N(q) = 3**	**K = 16**	**N(q) = 5**	**K = 16**	**N(q) = 6**
Result	**Distance**	**Result**	**Distance**	**Result**	**Distance**	**Result**	**Distance**
Brain08	288.83	Bug04	377.28	Car01	591.47	Cheetah03	243.50
Brain06	410.96	Airplane01	437.14	Flower02	713.35	Tree02	307.80
Brain03	444.86	Cheetah03	477.80	Car03	744.25	Lion06	319.88
Brain02	455.94	Airplane02	512.03	Flower01	843.15	Lion01	326.41
Brain05	467.91	Cloud05	523.68	Flower03	880.38	Lion05	361.60
Brain09	551.68	Brain09	545.71	Tree05	910.94	Cheetah07	394.19
Airplane02	588.00	Bug03	560.60	Car04	943.27	Bug03	408.38
Brain04	622.54	Bug01	567.99	Tiger03	1.04e+003	Lion04	408.95
Brain07	657.40	Cheetah02	594.35	Cheetah01	1.05e+003	Cheetah04	426.76
Bug02	730.98	Tree02	621.36	Tiger04	1.06e+003	Cheetah06	427.62
Successful		Successful		Successful		Successful	

4.3 General Evaluation

This section gives an overall evaluation for the applicability of the five different image signatures used throughout the experiments. Table (4.9) presents the performance evaluation of the overall results given in section 4.2. Table (4.10) presents the overall accuracy of the implemented signatures, where accuracy is defined, here, as the percentage of successful retrieving results.

Table (4.9): results of performance evaluation.

Signature type	Table	Query no.	AVR	MRR	NMRR	ANMRR
Luminance Histogram	**(4.3a)**	1	5	4	1	0.25
		8	1.5	0	0	
		9	1	0	0	
		10	4	0	0	
	(4.3b)	2	5.625	1.125	0.09	0.139
		5	3.5	0	0	
		7	5	3.5	0.466	
		8	1.5	0	0	
	(4.3c)	4	10.8	7.8	0.557	1.605
		6	9.25	6.75	0.465	
		11	9	7	0.583	
		12	2	0	0	
Low – Pass Luminance Histogram	**(4.4a)**	3	5.333	3.333	0.277	0.208
		9	1	0	0	
		10	4	0	0	
		13	10.8	7.8	0.557	
	(4.4b)	6	2.5	0	0	0.275
		7	5	3.5	0.466	
		11	9	7	0.636	
		12	2	0	0	
	(4.4c)	1	1	0	0	0.033
		2	4.5	0	0	
		5	5.333	1.833	0.135	
		8	1.5	0	0	

Signature type	Table	Query no.	AVR	MRR	NMRR	ANMRR
Luminance Pyramid	(4.5a)	3	2	0	0	0.336
		9	5	4	1	
		10	6.285	2.285	0.175	
		13	5.4	2.4	0.171	
	(4.5b)	4	8	5	0.357	0.089
		5	3.5	0	0	
		6	2.5	0	0	
		8	1.5	0	0	
	(4.5c)	1	1	0	0	0.256
		2	4.5	0	0	
		5	9.5	6	0.444	
		11	9	7	0.583	
Color Pyramid	(4.6)	1	1	0	0	0.042
		3	2	0	0	
		6	2.5	0	0	
		13	5.4	2.4	0.171	
YIQ Pyramid	(4.7)	2	4.5	0	0	0.070
		5	7.333	3.833	0.283	
		8	1.5	0	0	
		9	1	0	0	
Combined Signature	(4.8a)	4	4.5	0	0	0
		8	2	0	0	
		10	3	0	0	
		12	3.5	0	0	
	(4.8b)	2	4.5	0	0	0.116
		6	2.5	0	0	
		7	5	3.5	0.466	
		9	1	0	0	
	(4.8c)	2	3	0	0	0
		3	1.5	0	0	
		4	4	0	0	
		5	2	0	0	

Table (4.10): The accuracy of the implemented signatures.

Signature type	Signature variation	Accuracy
Luminance Histogram	YIQ	75%
	ℓαβ	50%
	CIECAM97	25%
Low – Pass Luminance Histogram	YIQ	50%
	ℓαβ	50%
	CIECAM97	75%
Luminance Pyramid	YIQ	25%
	ℓαβ	75%
	CIECAM97	50%
Color Pyramid	RGB	75%
	YIQ	75%
Combined Signature	Luminance Pyramid + Luminance Histogram	100%
	Color Pyramid + Luminance Histogram	75%
	Color Pyramid + Luminance Pyramid	100%

Luminance based image signatures (i.e. 1^{st}, 2^{nd}, and 3^{rd} variations) can have tolerable information for CBIR (see tables 4.3 – 4.5). The accuracy of results ranges from 25% to 75%. This is due to that luminance vision is capable of detecting sharp edges and the fine details of patterns and textures in the image. On the other hand, image signatures that are extracted from the three image components (i.e. achromatic and chromatic information) support, on overall, more power to the CBIR than those achromatic – based image signatures (see tables 4.6 and 4.7). The accuracy of results is 75%.

While histogram – based methods, (results presented in tables 4.3 and 4.4), depend on only occurrence or distribution of intensities within the image, subsampling – based image signatures, (results presented in tables 4.5 – 4.7), are signatures that preserve spatial image layouts which in turn provide better results than histogram – based image signature (60% of accuracy for subsampling – based CBIR vs. 54% of accuracy for histogram – based CBIR).

Moreover, Color subsampling - based signatures (results given in tables 4.6 and 4.7) are dominant over luminance subsampling – based ones (They give accuracy of about 75% vs. 25% to 75% for luminance subsampling). The reason behind this fact is that the color method of subsampling preserves spatial color distribution within the image while luminance preserves only spatial distribution of intensities within the image.

Finally, combining two (but more quantized) features in one image signature behaves, at least, as good as the better of the two combined features. For example, comparing results in tables 4.6 with results presented in tables (4.8c).

1.1 Information Retrieval

Information retrieval (IR) is finding material of an unstructured nature that satisfies an information need from within large collections (usually stored on computers) [Man07].

Now, hundreds of millions of people engage in information retrieval every day when they use a web search engine or search their email. Information retrieval is fast becoming the dominant form of information access, overtaking traditional database-style searching.

IR can cover different kinds of data and information problems. The term “unstructured data” refers to data which does not have clear, semantically overt, easy – for – a computer structure. It is the opposite of structured data, the canonical example of which is a relational database, of the sort companies usually use to maintain product inventories and personnel records. In reality, almost no data are truly “unstructured”. This is definitely true of all text data if you count the latent linguistic structure of human languages. But even accepting that the intended notion of structure is overt structure, most text has structure, such as headings and paragraphs and footnotes, which is commonly represented in documents by explicit markup (such as the coding underlying web pages). IR is also used to facilitate “semi-structured” search such as finding a document where the title contains Java and the body contains threading.

The field of information retrieval also covers supporting users in browsing or filtering document collections or further processing a set of retrieved documents. Given a set of documents, clustering is the task of coming up with a good grouping of the documents based on their contents. It is similar to arranging books on a bookshelf according to their topic.

Information retrieval systems can also be distinguished by the scale at which they operate, and it is useful to distinguish three prominent scales. In *web search*, the system has to provide search over billions of documents stored on millions of computers. Distinctive issues need to gather documents for indexing, being able to build systems that work efficiently at this enormous scale, and handling particular aspects of the web, such as the exploitation of hypertext and not being fooled by site providers manipulating page content in an attempt to boost their search engine rankings, given the commercial importance of web. At the other extreme is *personal information retrieval*. In the last few years, consumer operating systems have integrated information retrieval (such as Apple's Mac OS X Spotlight or Windows Vista's Instant Search). Email programs usually not only provide search but also text classification: they at least provide a spam (junk mail) filter, and commonly also provide either manual or automatic means for classifying mail so that it can be placed directly into particular folders. Distinctive issues here include handling the broad range of document types on a typical personal computer, and making the search system maintenance free and sufficiently lightweight in terms of startup, processing, and disks-pace usage that it can run on one machine without annoying its owner. In between is the space of *enterprise, institutional, and domain-specific search*, where retrieval might be provided for collections such as a corporation's internal documents, a database of patents, or research articles on biochemistry. In this case, the documents will typically be stored on centralized file systems and one or a handful of dedicated machines will provide search over the collection [Man07].

1.2 CBIR Motivations

When it comes to express ideas and conveying information, mankind has always preferred concrete visual means (images, painting) to more abstract counterparts (written text). This is clear to the vision and understanding from the ancient times of the visual methods compared to the written methods. Moreover, the recent development of technological digital data handling has further strengthened human dependence on visual modes of communication. This can, for instance, be witnessed in the explosively growing amount of digital image data, especially with the proliferation of the World Wide Web (WWW). This rapid production has, as a result, generated a huge repository of visual information in large area of applications such as medical image database, criminal suspect tracking, travel image gallery, scientific, educational, industrial, and personal or family picture collection [Hua98].

The information stored in the visual repository is visually useless if it is unorganized. Retrieving a particular image from a huge unorganized image database is similar to searching for book from a huge library without the aid of catalogs. In other words, indexing an image database is analogous to cataloging a library. Thus, the importance of the ability to search and retrieve images from an image collection can not be overemphasized and is usually denoted by CBIR problem [Lon03].

The well text indexing techniques that were designed for text are neither suitable nor sufficient for image data. This can be traced back to the following facts [Hua98]. A text document is one – dimensional (an array of words), whereas a digital image is two dimensional (a video is three dimensional because of time component), and hence, the size of image data is usually larger than text. Moreover, the most significant difference between text and image data is that words are in some sense semantic objects, while the image data need to be processed and interpreted to extract the perceptual meaning (by which image data is likely to be explored, navigated and retrieved), which is yet to be achieved task in computer vision and image understanding.

Traditionally, manual text annotations approach was used for image data indexing and retrieving. The annotations can be used by some text information retrieval (TIR) systems to search images indirectly [Sal89, Reg06]. However, there are several inherent difficulties and problems with this approach. *First*, since image data contains very rich information, it is very difficult to capture the content of an image using only a few keywords, as well as the tedious work, labour intensive, language dependent, and vocabulary controlled involved in such an annotation process itself. *Second*, manual annotation process is quite subjective, ambiguous, and incomplete. *For example*, if a query refers to image content that was not initially annotated, or if the user uses different words to describe the same image content, the text retrieval system will then fail. Moreover, in some cases it is rather difficult to characterize certain important real world concepts, entities, and attributes by means of text only. Example of such concepts is the shape of single object and the various spatial constraints among multiple objects in an image. These have created great demands for automatic and effective techniques for content – based image (video) retrieval systems that can compute descriptors for symbolizing various properties of images.

1.3 CBIR Challenges

Most CBIR systems adopt the following two steps approach to search image databases [Hua98, Qiu03].

First, *indexing* each image in a database; where a feature vector capturing certain essential properties of the image is computed and stored in a *feature base*.

Second, *searching* feature base; where a query image is given by the user, its feature vector is computed, compared to the feature vectors in the feature base, and images most similar to the query image according to a *heuristic similarity measure*, are retrieved from the image database. By such a way the system is called Query – By – Example (QBE).

In a typical QBE system, a user poses a query by providing an existing image, and the system ranks the target images in the database according to the query image.

Content – based video browsing tools also provide users with similar capabilities. Here, a user provides an interesting query frame and the system retrieves other similar frames from a video sequence. *For example*, a FBI agent might want to locate a criminal from a video clip by supplying a mug shot of that criminal.

The indexing of an image database is often referred as *feature extraction*. Mathematically, a feature is an n–dimensional vector, with its components computed by some image analysis. Feature representation scheme can be either low – level, intermediate – level, or high – level [Qiu03]. Low – level deals with pixel level features, high – level deals with abstract concepts and intermediate – level deals with something in between. Whilst low – level vision is fairly well studied, mid and high – level concepts are very difficult to grasp, certainly extremely difficult to represent using computer bits. Examples of low – level visual cues are color, texture, shape, and spatial information. Regions or blobs generated as result of image segmentation are examples of middle – level features, whilst objects, semantic categories or types of event depicted in images are examples of high – level features. Then, the n components of a feature may be derived from one visual cue or from composite cues, such as combination of color and texture.

The gap between high – level features (human perception) and low – level features limits the query scheme to be QBE which is not natural to human interaction with the image retrieval system. High – level features facilitate a more natural user interaction with the image retrieval system. *For example*, user's queries are typically semantic (e.g.,"show me an image of the sky") and not on a low – level (e.g.,"show me a predominantly a blue and white image"). In other words, users typically expect content – based image retrieval system to analyse their queries at the same level of semantics that a human would do while performing analysis of the content of the image. However, high – level features are almost, impossible to generate or extract without human interaction (i.e. manually generated features) [Reg06].

Beside image features, there are other issues and challenges need to be addressed for content – based image retrieval and video browsing tasks. These are [Qiu03]:

- *Perceptual similarity:* Perceptual similarity determines the effectiveness of the feature for the purpose of retrieval. A good feature $f(I)$ for an image I should be designed to have certain qualities such that a distance function $|f(I)-f(\bar{I})|$ should be large if and only if images I and $\bar{I}$ are not similar.

- *Efficiency:* where the computation of $f(I)$ should be fast enough.

- *Economy:* the size of image feature $f(I)$ should be small. This not only affects the efficiency of retrieval, but also affects the design of indexing data structure, such as multi – dimensional indexing scheme.

1.4 Literature Survey

M. Swain and D. Ballard [Swa91]: in 1991, M. Swain and D. Ballard used color histogram method for image indexing. They showed that color histogram has a number of advantages including: easy and simple to implement, insensitive to scale, rotation and translation, very successful for small size databases. However, color histogram based image retrieval systems encounter poor performance for large database. They demonstrate that this weakness comes from the fact that color histogram method preserve only color information and does not include any information regarding the spatial positions of the color. Hence, any two images with very different spatial color layouts can have the same color histogram. This case evident in large database, where the chance of (visually) different images having similar color histograms increase.

To overcome the problem encountered by color histogram of the previous method; different authors proposed different approaches. Two of such approaches are those of K. Hirata and T. Kato (1992) [Hir92] and W. Hsu et. al. (1995) [Hsu95]. They divided the image into sub images and describe each sub image features with a separate color histogram. These methods suffer from expensive computation, storage overhead, and can not accommodate translation and rotation of color regions.

Another line of work is that of J. Huang in 1998 [Hua98]. He proposed a new image features called correlogram for image indexing. Color correlogram represents the spatial correlation of colors in an image. It can be represented as a table indexed by color pairs and distance, where the K^{th} entry for $< i, j >$ specifies the probability of finding a pixel of color i at distance K away from another pixel of color j. By providing an efficient algorithm for computing the correlogram table, the resulted CBIR gives high performance for tackling various problems in image retrieval and video browsing.

Then in 1999, C. Carson et. al. [Car99] used image segmentation approach for image indexing. This technique (and all techniques based on image segmentation) can identify region more accurately than color histogram but the difficulty associated with an accurate image segmentation process makes it complicated both in terms of image features extraction and matching.

In 2003, G. Qiu and K. Lam [Qiu03]: suggested a color indexing method based on human vision theories and digital signal analysis. Their argument is that image patches of different spatial frequencies would have different perceptual as well as physical significances. The input image is passed through a bank of filters that decompose the image into a number of images; each K^{th} layer image corresponds to the K^{th} filter of the filter bank. By filter bank, pixels of the input image are classified into different layers. Then, each image layer is indexed using a quantized color histogram. The color histogram of the final image feature is obtained by concatenating the features of all sub –

layers. Their method significantly enhances the performance and power of color indexing schemes used in CBIR but at the same time retains its simplicity and elegance.

Also in 2003, L. F. M. Vieira, et. al. [Vie03] proposed several ways to browse a color image from a database of collection of color images according to a greyscale query image. Their main image feature used is image Luminance. Luminance information is extracted in their work using ℓαβ color model, and is passed through number of processing (including histogram computation, convolution with low and high pass filters) to obtain the final image feature. They keep simplicity in feature implementation so as to provide an acceptable level of accuracy in image retrieval.

1.5 Work Contributions

The work in this thesis is concentrating on implementing a group of image features that can facilitate the process of content – based image retrieval system. The main objectives in the feature extraction and image indexing implementation are:

- *Design simplicity;* where the implementation of image feature extraction and indexing phase is made as simple as possible.
- *Efficiency;* where the computation of image feature is made fast.
- *Economy;* where the size of image feature vector is made as small as possible.

- *Accuracy;* where the retrieving results of the CBIR using the implemented group of image features are suitable and acceptable according to the query image.

1.6 Thesis Layout

The rest of this thesis is organized as follows: -

- *Chapter 2;* presents the fundamental theories of content – based image retrieval system, their development, and components that can characterize a typical CBIR technique including query formation, feature extraction, similarity metric, indexing and retrieval, and user interaction.
- *Chapter 3;* presents five types of image features, all of them are considered to be low – level image features. Also, it discusses how to extract these features from an image, feature length, and similarity or distance measurements used to compute relation between two distinct features.
- *Chapter 4;* presents how to evaluate a particular CBIR system using the image features mentioned in chapter 3. The results on a given database of a set of image classes are tabulated to show the accuracy range for a give feature.
- Finally, *chapter 5* concludes the work and points out some possible future ramifications.

3.1 Introduction

In the previous chapter, the whole components that constitute a typical content – based image retrieval systems are present. One important step in this retrieval system is the image feature vector specification. This chapter presents a number of image feature (called image signature) variations. The variations are based on either luminance, processed luminance, or color information. The following sections present these variations together with the similarity metric used between images.

3.2 Signature 1: Quantized Luminance Histogram

In this variation, the luminance (i.e. intensity) information is used as a relevant similarity criterion for content – based Image retrieval. The formation of this signature follows the following sequence: -

1. Extract luminance information from the image by decorrelating achromatic channel from the chromatic channels. This can be established by converting the RGB color space of the image into a decorrelated color space (e.g., YIQ, $\ell\alpha\beta$, and CIECAM97 presented in the previous chapter).

2. The values of intensity extracted in step 1 can be real numbers with small value range. Hence, to overcome this problem a linear normalization of the luminance channel is required before

computing histogram. A quantization to n levels is performed using the formula: -

$$\overline{L}(x, y) = \text{int}\left(\frac{L(x, y) - \min}{\max - \min} * (n-1) \right) \tag{3.1}$$

Where:

$L(P(x, y))$: represents the luminance value of pixel P at coordinate (x, y).

n : number of quantization levels (Integer value).

$\max$: maximum value of luminance channel before quantization.

$\min$: minimum value of luminance channel before quantization.

3. After post – conversion quantization, the luminance channel will admit 128 possible values. Then, the histogram can be evaluated as a vector of 128 pixel counts. Formally speaking, the luminance histogram of an $m*n$ image I is a discrete function that maps each value K in the image′s intensity range to the fraction of pixels in image I that have intensity K, thus:

$$h_I = \left|\lambda_I(K)\right| \tag{3.2}$$

where:

$\lambda_I(K) = \{I(x, y) = K\}$

$I(x, y)$ = intensity of a pixel at coordinate (x, y) in image I.

Figure 3.1 depicts an example of luminance histogram applied for a given color image.

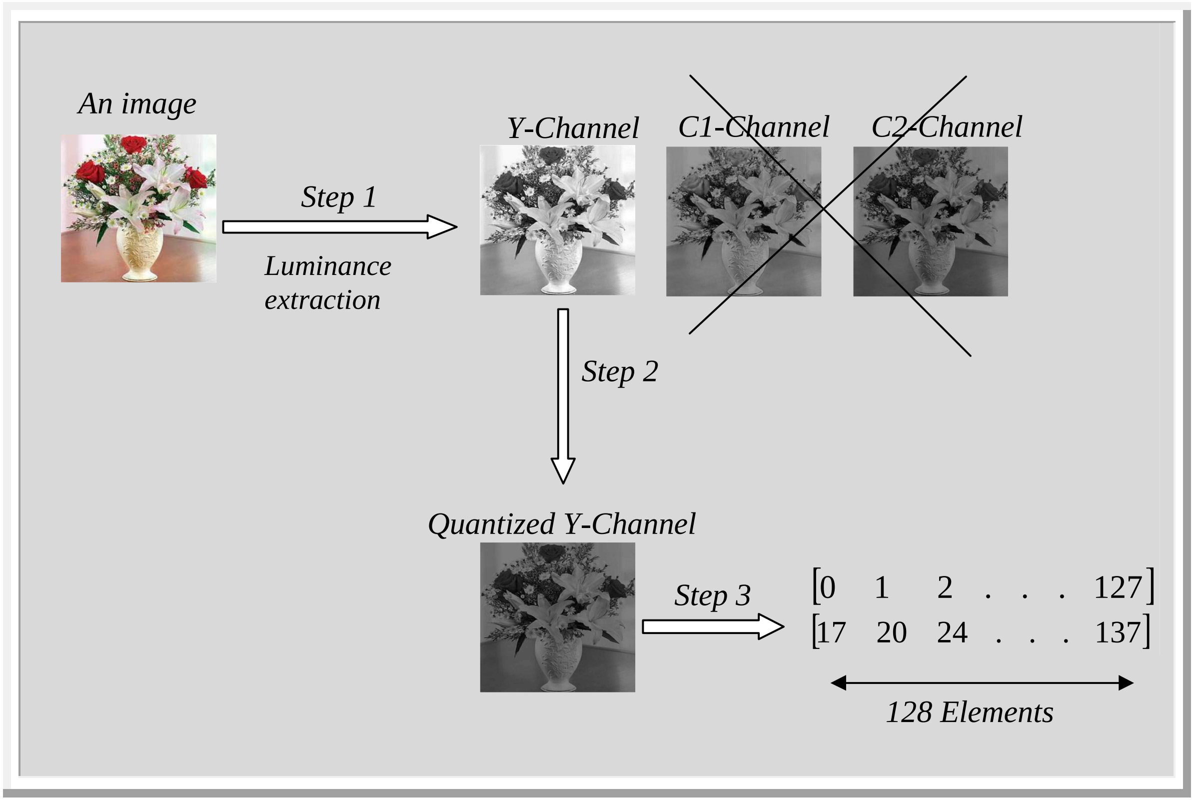


Figure 3.1: Luminance histogram.

In this variant, the similarity metric used between two feature vectors $f1(I)$ and $f2(J)$ of images I and J respectively can be performed as:

$$D(f1(I), f2(J)) = \sum_{i=1}^{\max i} \left| f1_i(I) - f2_i(J) \right| \tag{3.3}$$

Where $\max i$ equals to the feature vector length, e.g., 128.

3.3 Signature 2: Low – Passed Luminance Histogram

The second variation that compresses an image into a signature is that of convolving the pre – processed (i.e. quantized linearly normalized) luminance channel with a low – pass filter.

The low – pass or more commonly referred to as smoothing filters are used for blurring and for noise reduction. Blurring is used in preprocessing steps, such as removal of noise and small details from an image prior to (large) object extraction, and bridging of small gaps in lines or curves [Gon01]. Blurring can be applied by using a low – pass filter to the input image in order to allow the low spatial frequencies in the image to pass through, while attenuating the high spatial frequencies of the noise components. Unfortunately it is impossible to retain all the image detail in such smoothed image and hence some degradation will occur (i.e. edges in the original image become less well defined or identified). Thus all smoothing filters will seek a compromise in removing as much noise as possible while still preserving the detailed edge information. The filtering operation can be implemented by convolving the entire image with a square mask (of size, e.g. 3×3 or 5×5) which operate by replacing each pixel value P by the average of its neighbours [Awc95].

A revision and improvement of square mask was published by Nagao and Matsuyama. These authors divide the 5×5 neighbourhood of a pixel into nine regions. The nine regions are four pentagonal and hexagonal regions as shown in figure 3.2.

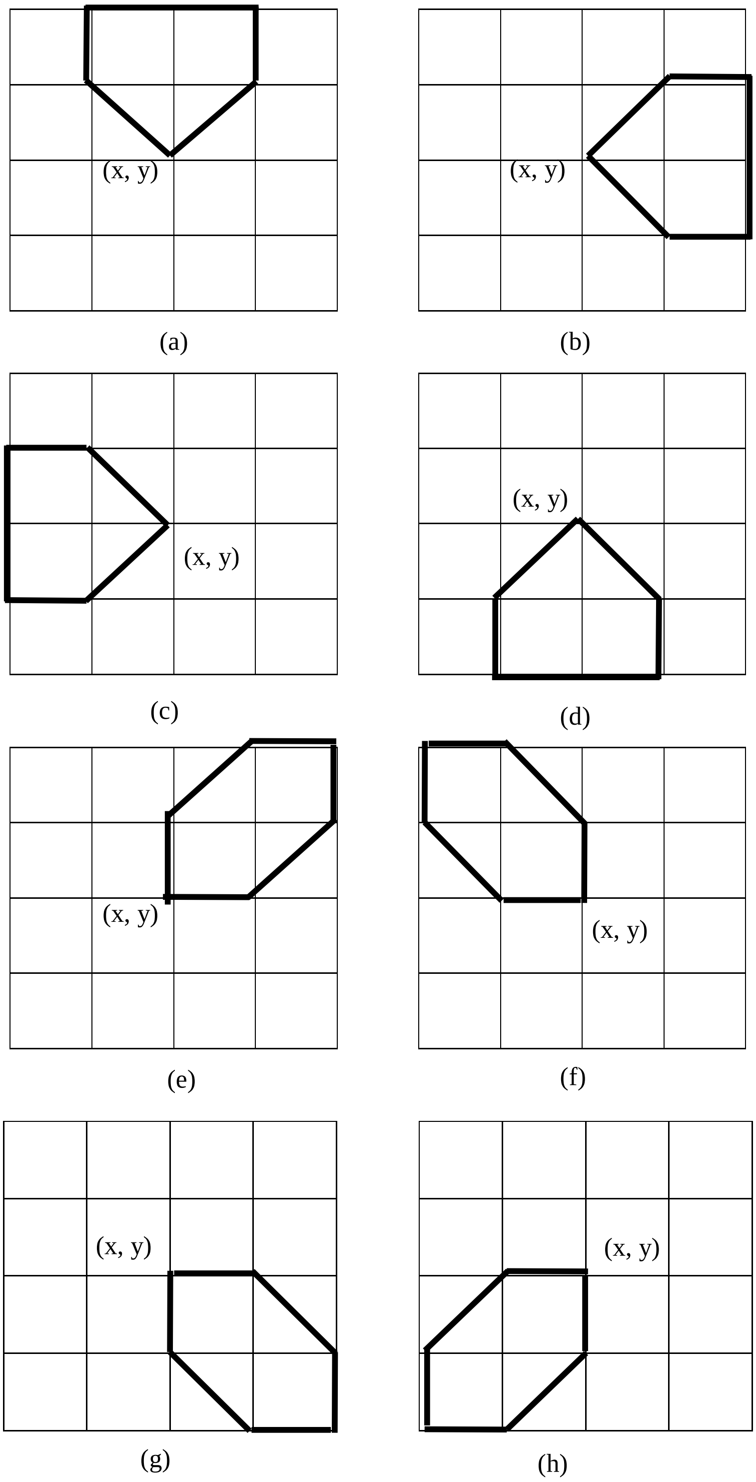


Figure 3.2: (a) Upper Pentagonal mask. (b) Right Pentagonal mask. (c) Left Pentagonal mask. (d) Lower Pentagonal mask. (e) Upper-Right Hexagonal mask. (f) Upper-Left Hexagonal mask. (g) Lower-Right Hexagonal mask. (h) Lower-Left Hexagonal mask.

Another type of low – pass filters is Gaussian filter. An example of Gaussian filter is that of Burtand Aldenson [Bur83] with coefficients (0.05, 0.25, 0.4, 0.25, and 0.05) applied separable to the rows and columns of an image.

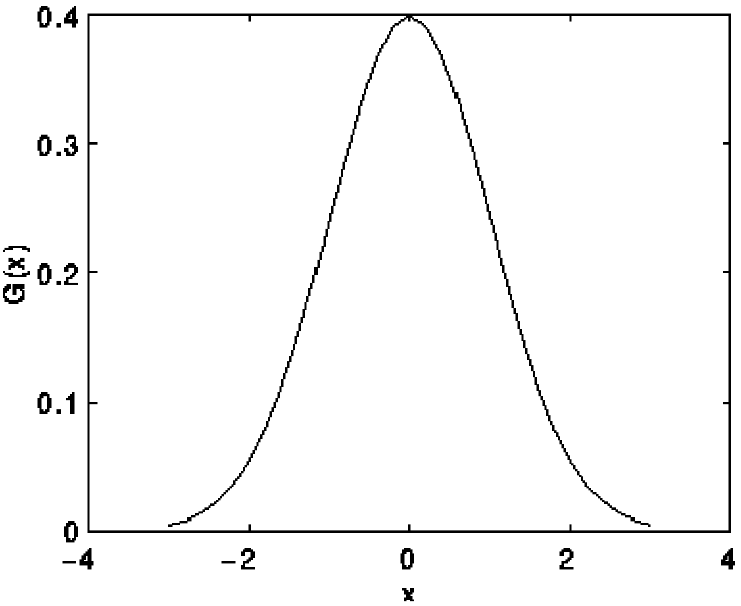


Figure 3.3: 1-D Gaussian distribution with mean 0 and $\sigma=1$

The process of generating low – passed luminance histogram is illustrated in figure 3.4. Although the generation of image signature that depends on either pure or low – passed luminance histogram is simple, but it faces a particular drawback. The histograms are invariant to the positions and orientations of the various objects within an image.

Here also, the similarity measure in equation (3.3) is used for computing relation between two image feature vectors.

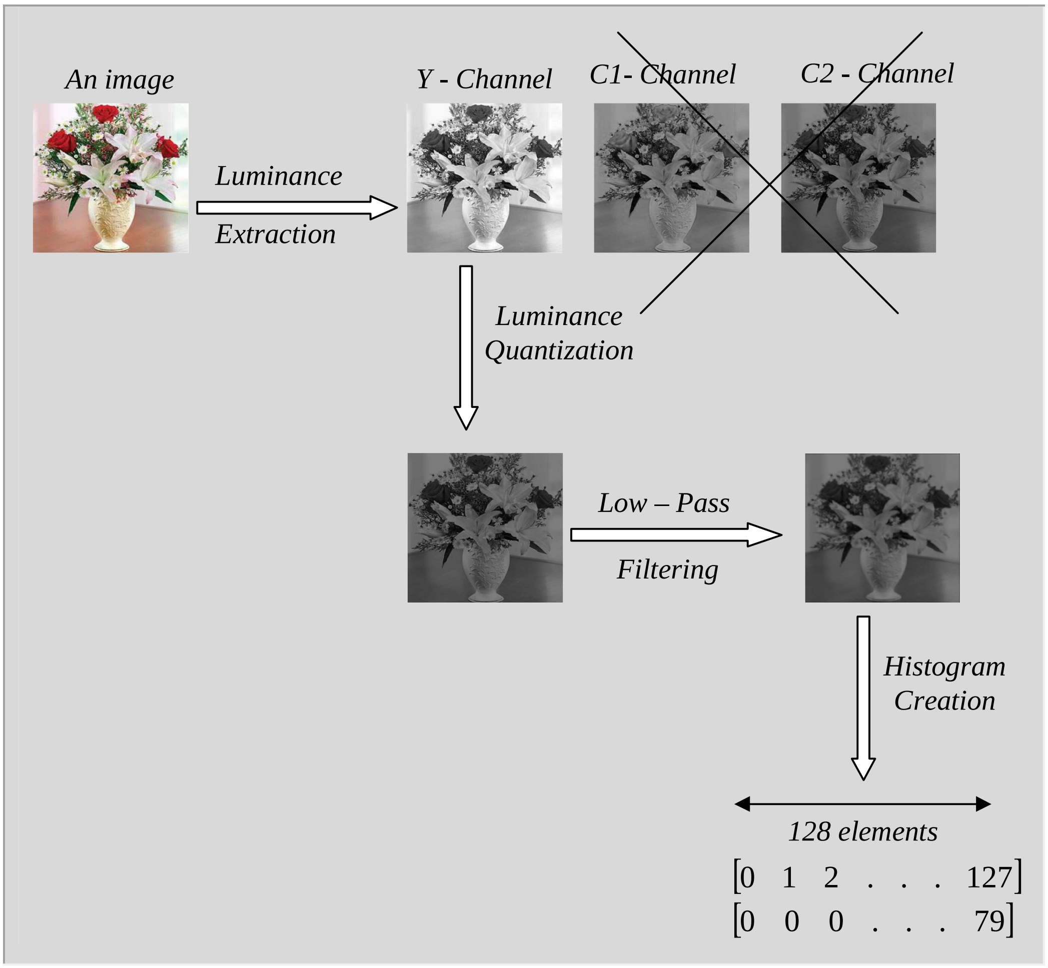


Figure 3.4: Generating low – passed luminance histogram.

1.3 Signature 3: Luminance Pyramid

Luminance pyramid or plain subsampling is another way of compressing an image into its signature. In this technique, the properly pre – processed quantized luminance image is subsampled down into a regular grid of regions of the same size, and the mean intensity within each such region is computed. The process of subsampling is continued on the resulted subsampled image for a given number of times so as to create a multi – level of subsampled images. The last level can correspond to an image of, e.g., 128 or 64 regions (i.e. 128 or 64 pixel). By this sequence of subsampling, an image pyramid is constructed in which the base layer corresponds the whole image with all its fine details, and the apex of the pyramid corresponds to the image signature that has a reasonable approximations to the intensity organization for the image at the pyramid base. Figure 3.5 depicts an example of image pyramid starting at the base with 256×256 intensity images and ending with the 8×16 apex image.

Unlike the previous two variations, in this type of image feature, the similarity metric is performed on values of intensity themselves rather than on their histograms. Thus, for two features vectors f_1 and f_2 belong to image I and J respectively, the similarity metric is: -

$$D(f1(I), f2(J)) = \sum_{x=1}^{n}\sum_{y=1}^{m}\left|L_I(x,y) - L_J(x,y)\right| \quad (3.4)$$

Where

n: width of the signature,

m: hight of the signature,

$L_I(x,y)$: luminance value of a pixel at coordinates (x,y) in the feature vector I,

$L_J(x,y)$: luminance value of a pixel at coordinates (x,y) in the feature vector J, and

$n \times m$ = length of the signature.

Figure 3.5: Image luminance pyramid.

Figure 3.6 illustrates the sequence of generating image signature based on luminance pyramid.

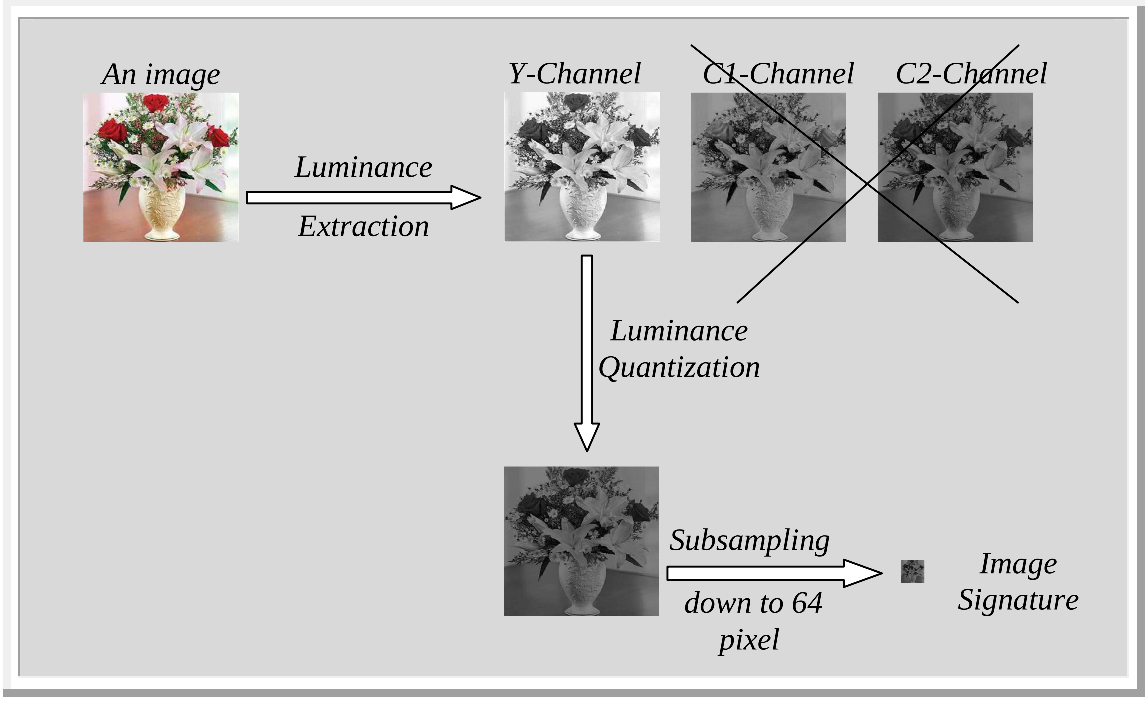


Figure 3.6: Image signature based on luminance pyramid.

As can be shown from the figure, the image signature preserves information related to the spatial distribution of intensities within the image which may be important to disambiguate two semantically different regions that have similar colors but at the expanse of obliterating some or all clues about the existence or not of fine texture in the image.

1.4 Signature 4 : Color Pyramid

Like luminance pyramid, an image color pyramid can be constructed by subsampling the image into several layers but the exception is that subsampling is performed on the three image channels rather than on only the luminance channel. In RGB color space, the channels Red, Green, and Blue are image components, whilst in any decorrelated color space, one channel is for luminance information and the other two channels are for chromatic information. The second difference is that; during subsampling and when mean value of each channel is computed, the pixel with closest color components is selected to be the subsampled pixel in the next finest layer of the pyramid. The closest pixel is that pixel with minimum Euclidean distance from the average color in case of RGB color model. For YIQ or YUV color model, the weighted Euclidean distance presented below is used [Tas98]:

$$D(P,\overline{P}) = \sqrt{\frac{10}{16}(Y-\overline{Y})^2 + \frac{3}{16}(C_1-\overline{C}_1)^2 + \frac{3}{16}(C_2-\overline{C}_2)^2} \tag{3.5}$$

Where:

P : is a colored pixel.

$\overline{P}$: is a colored pixel neighboured to pixel P.

Y : Luminance value of pixel of pixel P.

$\overline{Y}$: Luminance value of pixel of pixel $\overline{P}$.

$\mathrm{C1}$: 1^{st} chromatic component (either I or U) of pixel P.

$\overline{\mathrm{C}}1$: 1^{st} chromatic component (either I or U) of pixel $\overline{P}$.

$\mathrm{C2}$: 2^{nd} chromatic component (either Q or V) of pixel P.

$\overline{\mathrm{C}}2$: 2^{nd} chromatic component (either Q or V) of pixel $\overline{P}$.

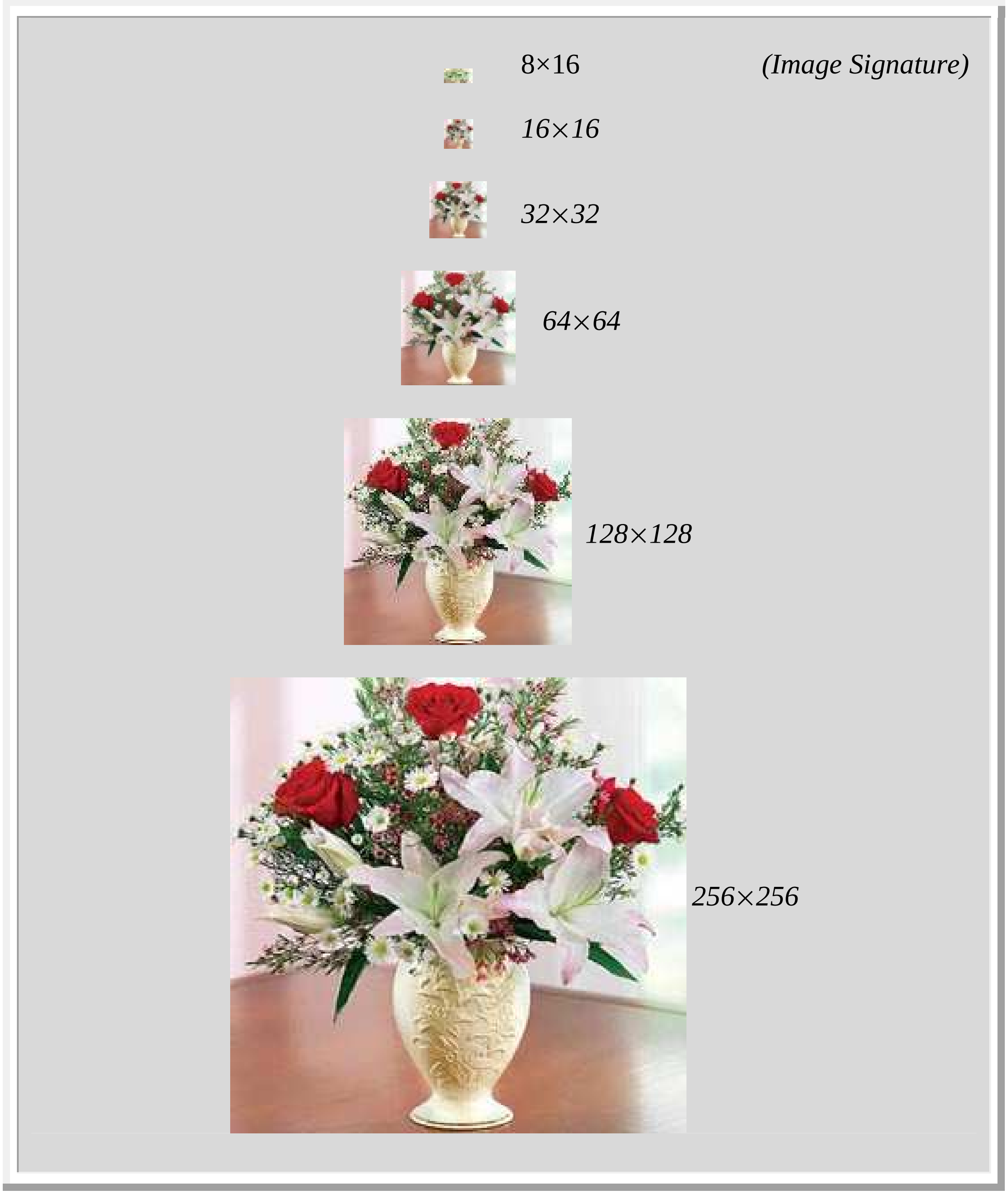


Figure 3.7: Image color pyramid.

1.5 Signature 5: Combined Signature

In this variant, we considered the use of image signatures that are computed from two basic signatures (i.e. those signatures discussed in the previous sections). *For example*, the signature that combines luminance with subsampling starts by computing the histogram of the luminance (or its corresponding low – passed luminance) and computing pyramid of either luminance or color image. However, the computation of histogram and pyramid are considered in this research to a more coarse quantization of only 64 values on each histogram and pyramid computations. Thus, the size of the signature does not to be increased. The final signature is, thus, obtained by a concatenation of the quantized luminance histogram and the subsampled image. Figure 3.8 depicts the steps of this signature variation. Also, figure 3.9 presents the application of this signature when luminance pyramid and color pyramid are used together to generate the combined signature.

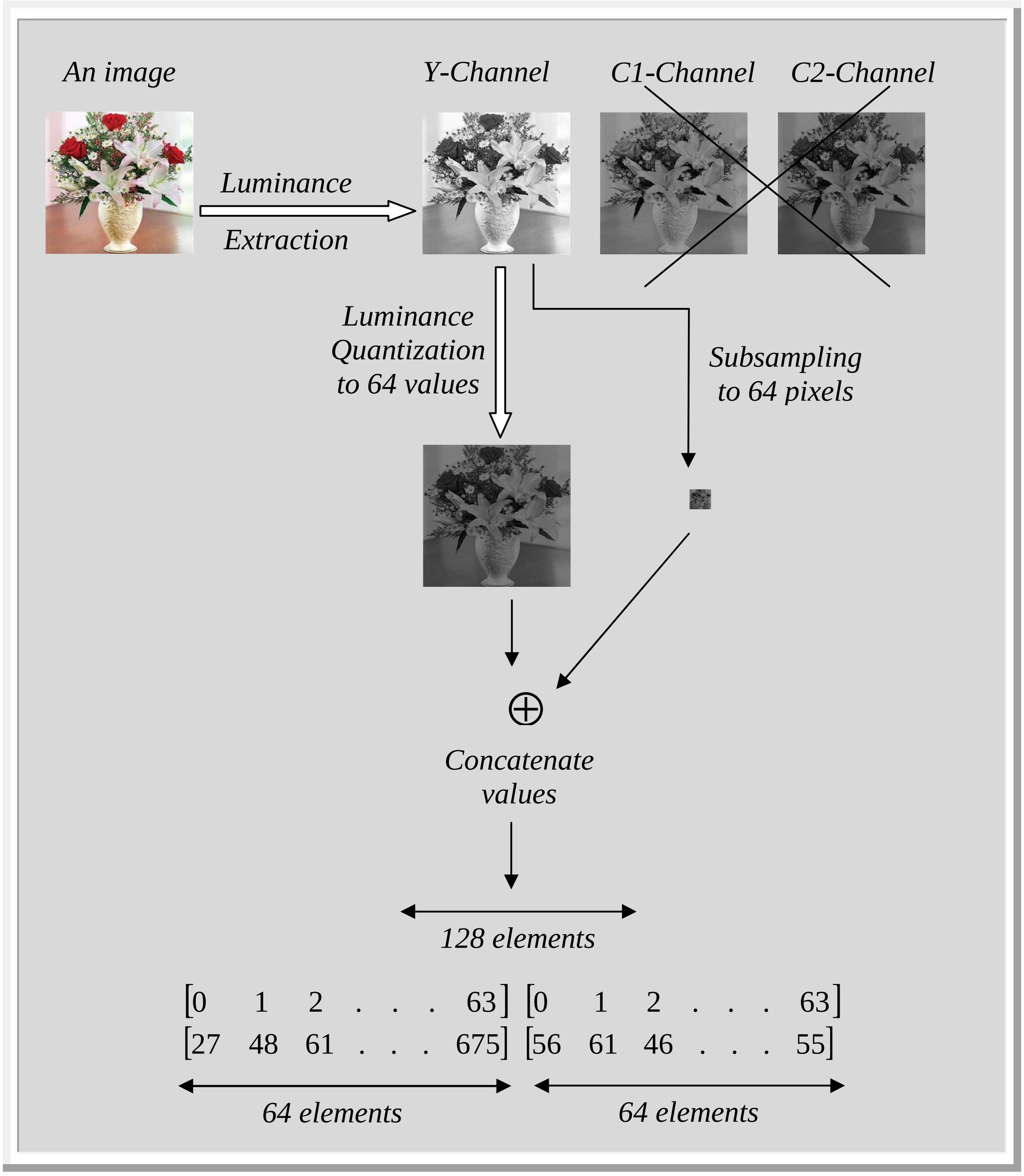


Figure 3.8: A combined signature of 64 elements from the luminance histogram and 64 elements from luminance pyramid.

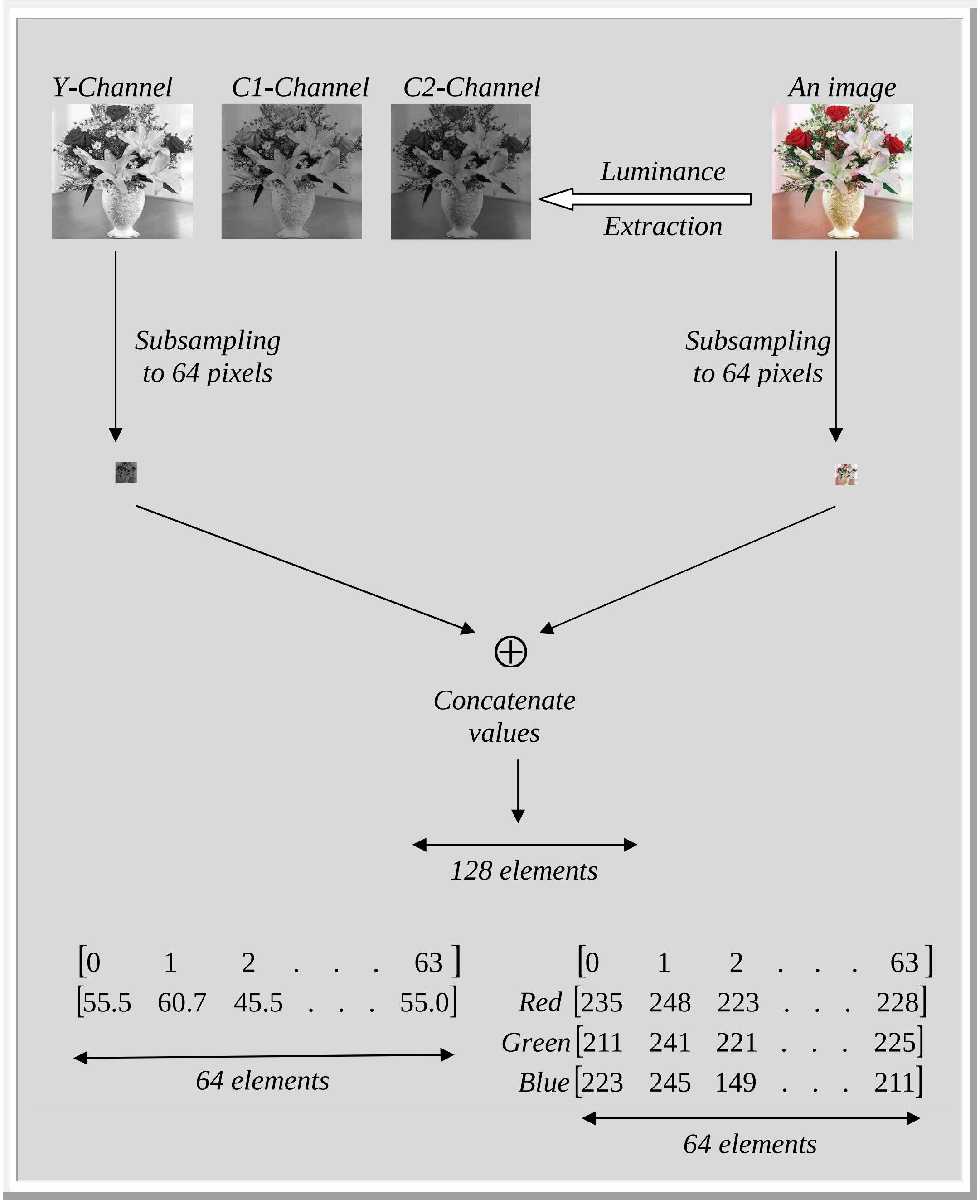


Figure 3.9: A combined signature generated from luminance pyramid of 8*8 pixels with color pyramid of 8*8 pixels.

2.1 Introduction

The advances in the internet and digital image sensor technologies resulted in a huge volume of digital image repository produced by the scientific, educational, medical, industrial, etc. application. The difficulties encountered by text – based image retrieval systems, thus, became more severe, and the need for efficient and new directions in image database management systems that are based on visual – rather than text – based retrieving techniques became urgent [Reg06].

This chapter looks at the fundamental of content – based image retrieval techniques. This includes the development of content – based image retrieval techniques, visual content features, indexing schemes, similarity/distance measurements, and system performance evaluations.

2.2 CBIR Framework

Content – based image retrieval uses the visual contents of an image such as *color, shape, texture,* and *spatial layout* to represent and index the image. In typical content – based image retrieval systems (Figure 2.1) [Lon03], the following general steps are followed: -

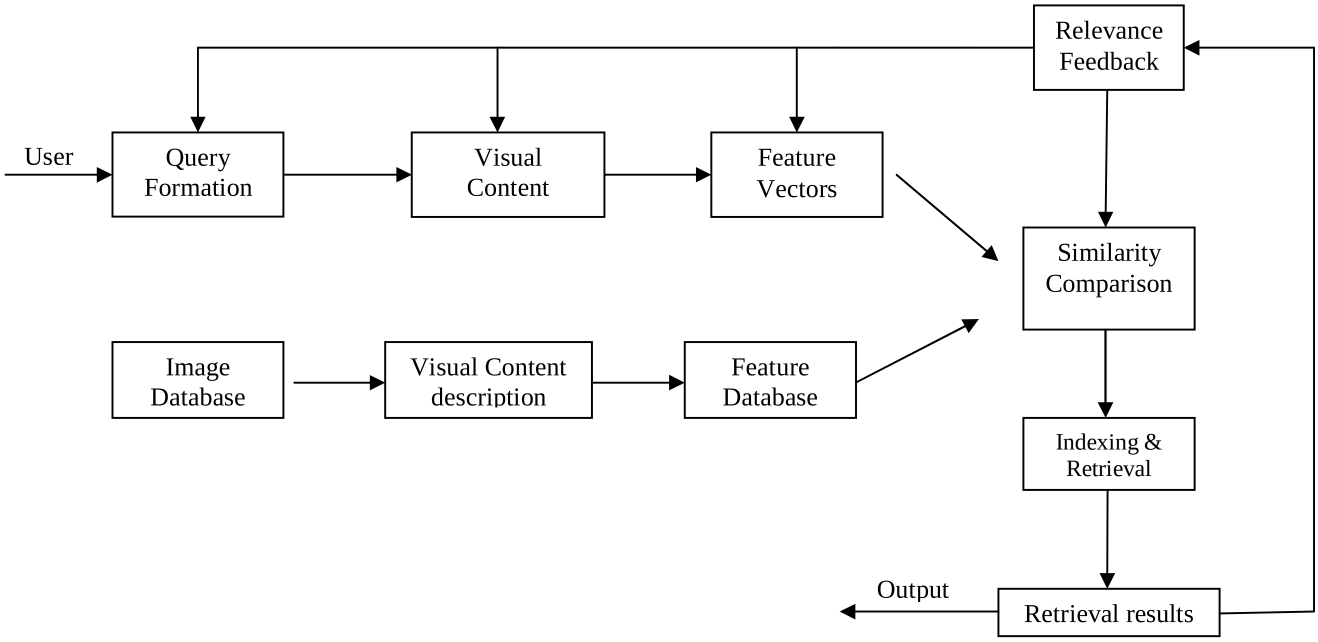


Figure 2.1: Diagram for content – based image retrieval system

- *Feature extraction;* the visual content of the images in the database are extracted and described by multi – dimensional feature vectors. Generally speaking, image content may include both visual and semantic content. Visual content can be very general or domain specific. *General visual content* include color, texture, shape, and spatial relationship. *Domain specific visual content*, like human faces, is application dependent and may involve domain knowledge. *Semantic content* is obtained either by textual annotation or by complex inference procedures based on visual content.

- *Feature database formation;* the feature vectors of the images in the database form feature database.

- *Feature extractions from the query image; to retrieve Images, users provide* the retrieval system with example images or sketched figures. The system then changes these examples into its internal representation of feature vectors.

- *Similarity measurement;* the similarities/distances between the feature vectors of the query example or sketch and those of the images in the database are then calculated and retrieval is performed with the aid of an indexing scheme.

- *Feedback;* recent retrieval systems have incorporated feedback to modify the retrieval process in order to generate perceptually and semantically more meaningful retrieval results.

Formally speaking the image retrieval problem is then: let D be an image database and Q be the query image. Obtain a permutation of the images in D based on Q, i.e. assign rank $(I) \in [|D|]$ for each $I \in D$, using some notion of similarity to Q. This problem is usually solved by sorting the images $I \in D$ according to $|f(I) - f(Q)|$, where $f(.)$ is a function computing feature vectors of images and $|\ .\ |f$ is some distance metric defined on feature vectors.

One important and critical step of the whole CBIR system depicted in figure 2.1 is the feature formation or feature extraction used to index images. The remaining sections of this chapter review image features that are considered to be general visual image descriptions.

2.3 Color and Color Models

Color US (colour, internationally) is a sensation produced in the brain in response to the incidence of light on the retina of the eye. The sensation of color is caused by differing qualities of the light emitted by light sources or reflected by objects [Lev97].

A graphical image is nothing more than a collection of organized colors intended to communicate some information. In the case of a scene or abstract image, the intent may primarily be to influence one's aesthetic sense. On the other hand, a quarterly revenues graph might be intended to influence a stockholder's blood pressure. Ultimately, however, both boil – down to the same thing a collection of colors [Lus93].

Color is very important component of graphical imagery. Many of the tasks associated with the manipulation and display of graphics files involve color – related operations [Lus93, Dav00]. In many engineering application, qualitative and quantitative characterization of color is essential. Color is an expression of spectral brightness of targets and as such is the basis of identifying objects or estimating their attributes. With the advent of computer aided digital techniques, the facility to produce a wide range of colors has dramatically improved. Color may be used on visual systems for aesthetic purpose, for formatting or for coding. Human eye is capable of distinguishing more colors than gray shades.

There are several models used to describe the tri – stimulus color scheme, some are: RGB, YIQ, YUV, $\ell\alpha\beta$, and CIECAM97 models. Each model has certain advantages over the others. Converting between the different models is generally done by a relatively simple mapping. Following subsections highlight in some details these color models.

2.3.1 RGB Model

In the RGB model, each color appears in its primary spectral components of red, green, and blue. This model is based on a Cartesian coordinate system. The color subspace of interest is the cube shown in figure 2.3 [Dav00], in which RGB values are at three corners. Cyan, magenta, and yellow are at three other corners. Black is at the origin, and white is at the corner farthest from the origin. In this model, the gray scale extends from black to white along the line joining these two points, and colors are points on or inside the cube, defined by vectors extending from the origin. For convenience, the assumption is that all color values have been normalized so that the cube shown in the figure is the unit cube that is, all values of R, G, and B are assumed to be in the range (0, 1) [Gon01].

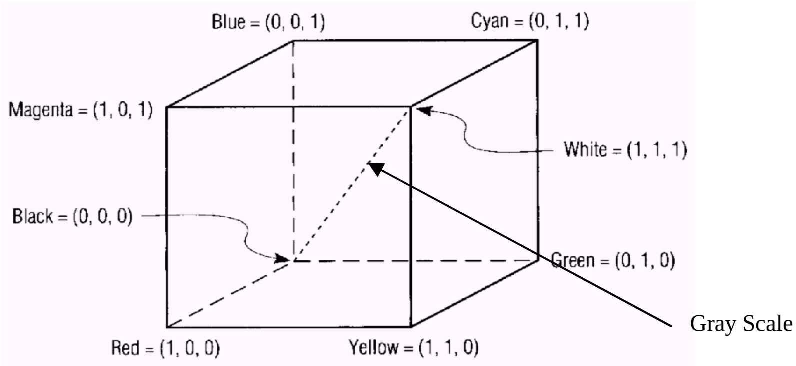


Figure 2.2: RGB color cube. Points along the main diagonal have gray values from black the origin to white at point (1, 1, 1).

2.3.2 YUV and YIQ Color Spaces

The YUV is a format that was first developed for color television in order to be compatible with the black and whites and is widely used throughout Europe. Y stands for luminance (or Luma), U (Cb) is the color difference for blue and V (Cr) is the color difference for red. In black and white televisions only the Y component is shown. The RGB to YUV conversion is defined as [Pon04]:

$$\begin{bmatrix} Y \\ U \\ V \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.147 & -0.289 & 0.436 \\ 0.615 & -0.515 & -0.300 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{2.1}$$

YUV uses a matrixes combination of Red, Green and Blue to reduce the amount of information in the signal. The Y channel describes Luma (slightly different than Luminance), the range of value between light and dark. Luma is the signal seen by black and white television. The U (Cb) and V (Cr) channels subtract the Luminance values from Red (U) and Blue (V) to reduce the color information. These values can then be reassembled to determine the mix of Red, Green and Blue. Some deeper research into YUV reveals two reasons why Blue always looks so crummy when extracted from video images. The U channel ranges from Red to Yellow, the V channel ranges from Blue to Yellow. Because Yellow is Red and Green, Red is essentially sent three times, Green twice and Blue only once. Reconstruction the Luminance component reveals another reason Blue suffers, the Blue channel is only 11% of Luminance [Mal03]. The transformation from YUV back to RGB model is [Bou94]:

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1.0 & 0.0 & 1.140 \\ 1.0 & -0.394 & -0.581 \\ 1.0 & 2.028 & 0.0 \end{bmatrix} \begin{bmatrix} Y \\ U \\ V \end{bmatrix} \tag{2.2}$$

The YIQ model is used in commercial color TV broadcasting. Basically, YIQ is a recording of RGB for transmission efficiency and for maintaining compatibility with monochrome TV standards. In fact, the Y component of the YIQ system provides all the video information required by a monochrome television set. The RGB to YIQ conversion is defined as [Gon01]:

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.144 \\ 0.596 & -0.275 & -0.321 \\ 0.212 & -0.528 & 0.311 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{2.3}$$

Converting from YIQ space to RGB space with the inverse matrix transformation [Hea94]:

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1.000 & 0.956 & 0.620 \\ 1.000 & -0.272 & -0.647 \\ 1.000 & -1.108 & 1.705 \end{bmatrix} \begin{bmatrix} Y \\ I \\ Q \end{bmatrix} \tag{2.4}$$

In the YIQ color space, where the Y coordinate represents the luminance, and I and Q coordinates represent the chrominance components I and Q, respectively. The I and Q components are used to jointly represent saturation and hue. It is worth noting that the luminance component Y contains a large component of the visual content, whereas the two chrominance components, I and Q, contain less perceptual information. Thus, due to the human visual system's lower sensitivity to color information, it is possible to sub sample the chrominance information and then integrate it back into the overall color image without any loss of perceptual quality.

2.3.3 ℓαβ and CIECAM97 Color Spaces

$\ell\alpha\beta$ color space was proposed by Ruderman et. al. [Pan04, Fil04]. It was developed to minimize correlation between the three coordinate axes of the color space for many natural scenes. This space is based on data – driven human perception research that assumes the human visual system is ideally suited for processing natural scenes. The color space provides three decorrelated, principal channels corresponding to an achromatic luminance channel ℓ and two chromatic channels α and β, which roughly correspond to yellow – blue and red – green opponent channels. Small changes in one channel impose minimal effect on values of other two. Following is the conversion from RGB to $\ell\alpha\beta$ and vice versa [Pan04, Fil04]:

The image can be converted from RGB to LMS space using the following conversion [Pan et. al. 04]:

$$\begin{bmatrix} L \\ M \\ S \end{bmatrix} = \begin{bmatrix} 0.3811 & 0.5783 & 0.0402 \\ 0.1967 & 0.7244 & 0.0782 \\ 0.0241 & 0.1288 & 0.8444 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{2.5}$$

Then convert the data to logarithmic space:

$$\begin{aligned} \mathrm{L} &= Log \quad L \\ \mathrm{M} &= Log \quad M \\ \mathrm{S} &= Log \quad S \end{aligned} \tag{2.6}$$

Then transform from LMS to $\ell\alpha\beta$ follows:

$$\begin{bmatrix} \ell \\ \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{3}} & 0 & 0 \\ 0 & \frac{1}{\sqrt{6}} & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & -2 \\ 1 & -1 & 0 \end{bmatrix} \begin{bmatrix} \mathrm{L} \\ \mathrm{M} \\ \mathrm{S} \end{bmatrix} \tag{2.7}$$

The result can be transferred back with the inversed operation from $\ell\alpha\beta$ to RGB:

$$\begin{bmatrix} \mathrm{L} \\ \mathrm{M} \\ \mathrm{S} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & -2 \\ 1 & -1 & 0 \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{3}} & 0 & 0 \\ 0 & \frac{1}{\sqrt{6}} & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} \ell \\ \alpha \\ \beta \end{bmatrix} \tag{2.8}$$

$$\begin{aligned} L &= 10^{\mathrm{L}} \\ M &= 10^{\mathrm{M}} \\ S &= 10^{\mathrm{S}} \end{aligned} \tag{2.9}$$

Finally, convert from LMS to RGB:

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 4.4679 & -3.5873 & 0.1193 \\ -1.2186 & 2.3809 & -0.1624 \\ 0.0497 & -0.2439 & 1.2045 \end{bmatrix} \begin{bmatrix} L \\ M \\ S \end{bmatrix} \tag{2.10}$$

CIECAM97 is one of many color appearance models. It was developed as an international standard color appearance model that borrows the best ideas from many different color appearance models. Previous to the development of CIECAM97s, only CIELAB stands for (Commission International de I'Eclairage) developed by Richard Hunter in 1942 that defines colors along two polar axes for color (a and b) and a third for lightness (L)) was available as a color appearance [DiC98]. CIE TCI – 34 proposed the CIE 1997 Interim Color Appearance Model (Simple Version) (CIECAM97s) to answer the need of the imaging industry for a single color appearance model. The goal was to develop a model that could be used for device – independent color imaging applications and to promote uniformity throughout the industry [Hat02]. The CIECAM97s model is closely relates to the $\ell\alpha\beta$ color space. The transformation from RGB to LMS and then to CIECAM97s is [Rei01]:

$$\begin{bmatrix} L \\ M \\ S \end{bmatrix} = \begin{bmatrix} 0.3811 & 0.5783 & 0.0402 \\ 0.1967 & 0.7244 & 0.0782 \\ 0.0241 & 0.1288 & 0.8444 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{2.11}$$

$$\begin{bmatrix} A \\ C1 \\ C2 \end{bmatrix} = \begin{bmatrix} 2.00 & 1.00 & 0.05 \\ 1.00 & -1.09 & 0.09 \\ 0.11 & 0.11 & -0.22 \end{bmatrix} \begin{bmatrix} L \\ M \\ S \end{bmatrix} \quad (2.12)$$

Where A is the achromatic channel and $C1$, and $C2$ are the chromatic channels. The two chromatic channels $C1$, and $C2$ resemble the chromatic channels α and β in $\ell\alpha\beta$ space, while the chromatic channel is different. Another difference is that CIECAM97s operates in linear space while $\ell\alpha\beta$ is defined in log space [Rei01]. The inverse operation to convert from CIECAM97s to RGB color space is:

$$\begin{bmatrix} L \\ M \\ S \end{bmatrix} = \begin{bmatrix} 0.3279 & 0.3216 & 0.2061 \\ 0.3279 & -0.6353 & -0.1854 \\ 0.3279 & -0.1569 & -4.5351 \end{bmatrix} \begin{bmatrix} A \\ C1 \\ C2 \end{bmatrix} \quad (2.13)$$

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 4.4679 & -3.5873 & 0.1193 \\ -1.2186 & 2.3809 & -0.1624 \\ 0.0497 & -0.2439 & 1.2045 \end{bmatrix} \begin{bmatrix} L \\ M \\ S \end{bmatrix} \quad (2.14)$$

2.4 Texture

Texture, as observed in wood, grain, stone, cloth, grass, etc. is an important surface property measure to describe region smoothness, coarseness, regularity, reflectivity, and granularity. Although texture is readily detected by the human visual system, no formal definition of texture exists. Analysis of texture begins by identifying basic texture elements (i.e. texels) which repeat with some degree of predictability. Each texel consists of a group of pixels which can have random, periodic, or partially periodic distributions. On the whole, man – made materials feature regular, periodic textures, whilst naturally occurring textures are random. In general, texture analysis can be divided into three principle approaches: statistical, structural, and spectral. Statistical techniques are used primarily for naturally occurring textures, having a random nature yield characterizations of textures as smooth, course, grainy, and so on. Structural techniques are well suited to man – made textures. They deal with the arrangement of image primitives, such as the description of texture based on regularly spaced parallel lines. Spectral techniques are used for partially periodic patterns so as to detect global periodicity in an image [Awc95].

2.5 Edge/Shape

The shape of objects / regions is yet another low – level image feature for content – based image retrieval. Edge detection, and sometimes segmentation are used to extract shape features and segment image into regions or objects. Since robust and accurate image segmentation is difficult to achieve, the use of shape features for image retrieval has been limited to special applications where objects or region are readily available. The state – of – art methods for shape description can be categorized into either *boundary – based* (rectilinear shape [Jag91], polygonal approximation [Ark91], finite element models [Scl95], and Fourier – based shape descriptions [Arb90, Kau95]) or *region – based* methods (statistical moments [Yan94]). A good shape representation for an object should be invariant to translation, rotation and scaling. Shape features based on some shape descriptors have limited discriminating capability. One problem with shape descriptors is that it is hard to find a good perceptual measurement of similar shapes. *For example*, similar moments do not guarantee similar shapes. Shortly speaking, shape is an important visual cue but alone does not provide good image retrieval performance.

2.6 Spatial Relationship

Regions or objects with similar color and texture properties but have different semantic meaning can be further distinguished by imposing spatial constraints. *For example*, regions of blue sky and ocean may have similar color histogram, but their spatial locations in images are different. Thus, the spatial location of regions (or objects) or the spatial relationship between multiple regions (or objects) in an image can be useful for searching images.

The most widely used spatial relationship is the 2D representation proposed by chang et. al. [Cha87] where the spatial relationships in an image is divided into two sets of spatial operators. One defines local spatial relationships and the other defines the global spatial relationships. This and subsequent methods can facilitate three types of query. The first type is for finding all images that contain objects O_1, $O_2 \ldots O_n$. The second type is for finding all images containing images that have certain relationship between each other, irrelevant to the distance between them. The third type is for finding all images that have certain distance relationship between objects.

2.7 Similarity / Distance Measures

Instead of exact matching, content – based image retrieval calculates visual similarities between a query image and images in database. Accordingly, the retrieval result is not a single image but a list of images ranked by their similarities with query image. The distance function used to compare features of images also plays an important role. An ideal distance function D and the feature $f(I)$ would satisfy the perceptual similarity:

$D(f(I_1), f(I_2))$ is small $\Leftrightarrow$ I_1 and I_2 are perceptual similar. In most cases, features are treated as points in high – dimension space. Therefore, it is naturally to define distance functions in terms of Euclidean norms. The L_1 norm and L_2 norm are commonly used when comparing two feature vectors. These norms can be specified by following Minkowski – From Distance [Hau98]: -

Minkowski – Form Distance: - If each dimension of image feature vector is independent of each other and is of equal importance, the ***Minkowski – form distance L_p*** is appropriate for calculating the distance between two images. This distance is defined as [Hua98]:

$$D(I,J) = \sum \left| f(I) - f(J) \right|^p \qquad (2.15)$$

When $p = 1, 2$, $D(I,J)$ is the L_1, L_2 (also called Euclidean distance), distance respectively. Minkowski – form distance is the most widely used metric for image retrieval.

2.8 Indexing Scheme

One important issue in content – based image retrieval systems is effective indexing and fast searching of images based on visual features. However, feature vectors of images tend to have high dimensionality, and thus they are not well suited to conventional indexing structures. Thus, to set up an efficient indexing scheme, dimension reduction should be used.

One commonly used dimension reduction techniques is the principle component analysis (PCA) that linearly maps input data to coordinate space such that the axes are aligned to reflect the maximum variations in the data. PCA is commonly used in microarray research as a cluster analysis tool. It is designed to capture the variance in a dataset in terms of *principle components*. In effect, one is trying to reduce the dimensionality of the data to summarize the most important (i.e. defining) parts whilst simultaneously filtering out noise. Principle Components is a set of variables that define a projection that encapsulates the maximum amount of variation in a dataset and is orthogonal (and therefore uncorrelated) to the previous principle component of the same dataset. Additionally Karhunen – Loeve (KL) transform can be used for dimension reduction. This method has the ability to locate the most important sub – space but at the expense of destroying the important feature properties that identify pattern similarity.

After dimension reduction, the multi – dimensional data are indexed using number of methods, e.g., R – tree, linear quad – trees, K

– d – B trees and grid files [Lon03]. Most of these indexing schemes can provide reasonable performance for a small number of dimensions (up to 20), but explore exponentially with the increasing of the dimensionality and eventually reduce to sequential searching.

2.9 user interaction

The formation and modification of queries in content – based image retrieval systems can only be obtained by system interaction with the user. User interaction typically consists of two parts.

1. *Query formulation part*; where kind of images a user wishes to retrieve from the database is specified. Query can be formed in either:

   - *Category browsing:* to browse through the database according to the category of the image, which classifies images according to their semantic or usual content.

   - *Query by concept:* to retrieve images according to the conceptual description associated with each image in the database.

   - *Query by sketch:* to draw a sketch (using graphical editing tool) from which images with similar visual features will be extracted from the database.

- *Query by example:* to provide an example image from which images with similar characteristics will be retrieved from the database.

2. *Results presentation part;* where an interactive relevance feedback is used as an active learning process for improving the effectiveness of information systems. The main idea is to use positive and negative examples from the user to improve system performance. For a given query, the system, first, retrieve a list of ranked images according to predefined similarity metrics. Then, the user marks the retrieved images as relevant (positive example) to the query or not relevant (negative examples). Then, the system will refine the retrieval results based on the feedback and present a new list of images to the user. Shortly speaking, in relevance feedback, one can incorporate positive and negative examples to refine the query and/or to adjust the similarity measure.

2.10 Performance Evaluation

In 1999, MPEG7 recommend a retrieval performance evaluation measure called, the *average normalized modified retrieval rank* [Lon03]. Let number of ground truth images for a given query q be denoted as $N(q)$ and the maximum number of ground truth images for all Q queries, i.e. max $(N(q_1), N(q_2) \ldots,\ N(q_Q))$, as M. Then, for aed given query q, each ground truth image K is assigned a rank value rank (k) that is equivalent to its rank in the ground truth images if it is in the first K query results where $K = \min[4N(q), 2M]$. Otherwise, the image k is assigned a rank value $K+1$. The average rank $AVR(q)$ for query q is computed as:

$$AVR(q) = \sum_{K=1}^{N(q)} \frac{rank\ (K)}{N(q)} \tag{2.16}$$

The modified retrieval rank $MRR(q)$ is computed as:

$$MRR(q) = AVR(q) - 0.5 - 0.5 * N(q) \tag{2.17}$$

$MRR(q)$ takes value 0 when all the ground truth images are within the first K retrieval results.

Then, the normalized modified retrieval rank $NMRR(q)$, which ranges from 0 to 1, is computed as:

$$NMRR(q) = \frac{MRR(q)}{K + 0.5 - 0.5 * N(q)} \tag{2.18}$$

Finally, the average normalized modified retrieval rank $ANMRR$ over all Q queries is computed as:

$$ANMRR = \frac{1}{Q} \sum_{q=1}^{Q} NMRR(q) \tag{2.19}$$

Republic of IRAQ
Ministry of Higher Education
and Scientific Research
AL-Nahrain University
College of science

Image Signature
For
Content-Based Image Retrieval

A Thesis
Submitted to the College of Science, Al-Nahrain University as a Partial Fulfilment in the Requirements for the Degree of Master of Science in Computer Science

By
Khawla Omer Farhan Al-Dabagh
(B.Sc. 2004)

Supervised By
Dr. Bara'a Ali Attea

November 2007

Acknowledgement

I would like to express with great interest my sincere thanks and gratitude to Assist. Prof. Dr. Bara'a Ali Attea for the continuous supervision and her interest, encouragement and guidance through out the course of this work.

I would like to thank Dr. Taha S. Bashaga the head of Computer Department / College of Science for his encouragement through out the studying years.

I also would like to thank the entire stiff in Computer Department / College of Science for their help.

I can not forget the assistance given by all my friends.

Dedicated to …

To the woman who gives us her compassion, consume her life and her youth for our happiness and our raising, and illuminate the way for us… My Dear Mother.

***

To the person who supports me, protect me, and give me confidence and happiness… My Dear Husband.

***

To the girl who standby me in sorrow and in joy, in good days and in bad days… My Dear Sister.

***

To the sun that shines my life, to the hope in my life, and whom I live my life for him… My Little Baby.

***

To all my family

Khawla

CHAPTER ONE

INTRODUCTION

CHAPTER TWO

FUNDAMENTALS OF CONTENT-BASED IMAGE RETRIEVAL

CHAPTER THREE

IMAGE SIGNATURE GENERATION

CHAPTER FOUR

EXPEREMENTAL EVALUATIONS

CHAPTER FIVE

CONCLUSIONS AND FUTURE WORK

REFERENCES

جمهورية العراق
وزارةالتعليم العالي والبحث العلمي
جامعةالنهرين/ كليةالعلوم

امضاء الصور لاسترجاع الصور بالاعتماد على أساس المحتوى

رسالة
مقدمة الى كلية العلوم – جامعة النهرين كجزء من متطلبات
نيل درجة الماجستير في علوم الحاسبات

من قبل
خولة عمر فرحان الدباغ
(بكلوريوس علوم الحاسبات -جامعة النهرين ٢٠٠٤)

بأشراف
أ٠ م٠ د٠ براء علي عطية

كانون الأول ٢٠٠٧

الخلاصة

أدى التقدم السريع لتكنلوجيا التصوير، خاصةً بعد التطور الحاصل في الشبكة المعلوماتية، لأن يصبح موضوع استرجاع الصور عامل جذب واهتمام الباحثين في مجال المكتبات الرقمية ومعالجة الصور وأنظمة قواعد البيانات لما يتمتع به من خواص تميزه عن المعلومات الرقمية.

ان تقنيات الأسترجاع التي صممت للنص غير مناسبة وغير كافية لبيانات الصور، وهذا يعود الى صعوبة الوصول الى بيانات صور يمكن استكشافها والعمل عليها واسترجاعها بواسطة ادراك التشابه، بالأضافة الى كون تقنيات الأسترجاع التي صممت للنص محددة التطبيقات. كما ان تقنيات استرجاع الصور المبنية على اساس المحتوى تأخذ مساحة مهمة في الحسابات المتعددة الوسائط.

ان استعادة الصور المبنية على اساس المحتوى يمكن أن يتم بسلسلة من الخطوات: تشكيل صورة الطلب , استخراج معالم الصورة ,قياس التشابه ,الفهرسة والأسترجاع ,تفاعل المستخدم .

ان الأختيار الصحيح لكل خطوة يمكن أن يعطي نتائج جيدة وكفوءة ومناسبة للنظام .

ان هذا البحث يركز على خطوة مهمة في النظام وهي استخراج المعالم أو تشكيل المعالم ، كما أن معالم الصورة المستخدمة في هذا البحث كلها تتميز بكونها معالم بسيطة أو معالم ذات مستوى واطئ ، وهذه المعالم تتضمن: الرسم البياني لنصوع الصورة ,الرسم البياني للنصوع ذو المرور الواطئ ,هرم النصوع ,هرم اللون ,المعالم المدمجة .

ان ما يقدمه هذا البحث هو الوصول الى طريقة سهلة وبسيطة ومنظمة لاستخلاص المعالم وتوفير استعادة نتائج مقبولة بكفاءة عالية.

تم اختبار النظام على قاعدة بيانات مكونة من ١٣ صنف من أصناف الصور وقد دلت النتائج على ان اختيار معالم الصورة يمكن أن يؤثر بشكل كبير في فعالية وأداء النظام ، كما ان المعالم التي تعتمد على المعلومات اللونية والمعلومات غير اللونية يوفر نتائج موثوقة أكثر من تلك التي تعتمد على كثافة المعلومات فقط، بالاظافة الى ذلك فان اتحاد معلمين يمكن أن يعطي نتائج أفضل.

بسم الله الرحمن الرحيم

"قَالَ رَبِّ اشرَح لِي صَدرى (٢٥) وَيَسِّر لِى أمرى (٢٦) وَاحلُل عُقدَةً مِن لِّسَانِى (٢٧) يَفقَهُوا قَولى (٢٨)"

صدق الله العلي العظيم

سورة طه (الآيات ٢٥-٢٨)

Abstract

With the explosive advancement in imaging technologies and specially with proliferation of the world wide web, image retrieval has attracted the increasing interests of researches in the field of digital libraries, image processing and database system. Research in human perception of image content suggests that content-based image retrieval (CBIR) can follow a sequence of steps. The typical steps of CBIR system are: image query formation, image feature extraction, similarity measurement, indexing and retrieval, and user interaction. The correct choice and set up for each step will result in a well, efficient and suitable CBIR system.

This work concentrates on one important and crucial step of the whole CBIR system: feature extraction (or feature formation). The image features used are all characterized as low – level features. These include: image luminance histogram, low – passed luminance histogram, luminance pyramid, color pyramid, and combined feature.

The main contributions are: simplicity (i.e. easy to implement the feature extraction phase), suitability (i.e. provide acceptable retrieving results), efficiency, and economy. The CBIR with the presented feature extraction variants are tested on a selected database of a set of thirteen image classes. In general, the results indicate that the choice of image feature can greatly affect the performance of CBIR system. Experimental results showed that image features that utilize achromatic and chromatic information of the image can provide about 75% accurate results, while those depend on only intensity information can give accurate results in about 25% - 75%. Moreover, the combination of two features can give in better results.

الخلاصة

أدى التقدم السريع لتكنلوجيا التصوير، خاصةً بعد التطور الحاصل في الشبكة المعلوماتية، لأن يصبح موضوع استرجاع الصور عامل جذب واهتمام الباحثين في مجال المكتبات الرقمية ومعالجة الصور وأنظمة قواعد البيانات لما يتمتع به من خواص تميزه عن المعلومات الرقمية.

ان تقنيات الأسترجاع التي صممت للنص غير مناسبة وغير كافية لبيانات الصور، وهذا يعود الى صعوبة الوصول الى بيانات صور يمكن استكشافها والعمل عليها واسترجاعها بواسطة ادراك التشابه، بالأضافة الى كون تقنيات الأسترجاع التي صممت للنص محددة التطبيقات. كما ان تقنيات استرجاع الصور المبنية على اساس المحتوى تأخذ مساحة مهمة في الحسابات المتعددة الوسائط.

ان استعادة الصور المبنية على اساس المحتوى يمكن أن يتم بسلسلة من الخطوات: تشكيل صورة الطلب , استخراج معالم الصورة ,قياس التشابه ,الفهرسة والأسترجاع ,تفاعل المستخدم .

ان الأختيار الصحيح لكل خطوة يمكن أن يعطي نتائج جيدة وكفوءة ومناسبة للنظام .

ان هذا البحث يركز على خطوة مهمة في النظام وهي استخراج المعالم أو تشكيل المعالم ، كما أن معالم الصورة المستخدمة في هذا البحث كلها تتميز بكونها معالم بسيطة أو معالم ذات مستوى واطئ ، وهذه المعالم تتضمن: الرسم البياني لنصوع الصورة ,الرسم البياني للنصوع ذو المرور الواطئ ,هرم النصوع ,هرم اللون ,المعالم المدمجة .

ان ما يقدمه هذا البحث هو الوصول الى طريقة سهلة وبسيطة ومنظمة لاستخلاص المعالم وتوفير استعادة نتائج مقبولة بكفاءة عالية.

تم اختبار النظام على قاعدة بيانات مكونة من ١٣ صنف من أصناف الصور وقد دلت النتائج على ان اختيار معالم الصورة يمكن أن يؤثر بشكل كبير في فعالية وأداء النظام ، كما ان المعالم التي تعتمد على المعلومات اللونية والمعلومات غير اللونية يوفر نتائج موثوقة أكثر من تلك التي تعتمد على كثافة المعلومات فقط، بالاظافة الى ذلك فان اتحاد معلمين يمكن أن يعطي نتائج أفضل.

Certification of the Examination Committee

We chairman and members of the examination committee certify that we have studied this thesis **"Image Signature for Content – Based Image Retrieval"** presented by the student **"Khawla Omer Farhan"** and examined here in its content and that we have found its worthy to be accepted for the Degree of Master of Science in Computer Science.

Signature:
Name: Dr. Laith A. AL-Ani
Title: Assist. Prof.
Date: 13 / 4 / 2008
(Chairman)

Signature:
Name: Dr. Bushra K. AL-Abudi
Title: Assist. Prof.
Date: 13 / 4 / 2008
(Member)

Signature:
Name: Dr. Haithem A. AL-Ani
Title: Lecturer
Date: 13 / 4 / 2008
(Member)

Signature:
Name: Dr. Bara'a A. Attea
Title: Assist. Prof.
Date: 13 / 4 / 2008
(Supervisor)

Approved by the Deen of the College of Science, AL-Nahrain University

Signature:
Name: Dr. Laith A. AL-Ani
Title: Assist. Prof.
Date: 13 / 4 / 2008
(Dean of College of Science)

[Arb et. al. 90] K. Arbter, W. E. Snyder, H. Burkhardt, and G. Hirzinger, *"Application of Affine – invariant Fourier Descriptors to Recognition of 3D Objects"*, IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 12, p.p. 640 – 647, 1990.

[Ark et. al. 91] E. M. Arkin, L. P. Chew, D. P. Huttenlocher, K. Kedem, and J. S. B. Mitchell, *"An Efficiently Computable Matric for Comparing Polygonal Shapes"*, IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 13, no. 3, p.p. 209 – 226, 1991.

[Awc Tho 95] G. J. Awcock, and R. Thomas, *"Applied Image Processing"*, ACMILLAN Press Ltd., 1995.

[Bou 94] P. Bourke, *"Converting between RGB and CMY, YIQ, YUV"*, 1994. Available at [URL: stronomy.swin.edu.au/~pbourke/colour/convert.html].

[Bur Ade 83] P. J. Burt and E. H. Adelson, *"The Laplacian Pyramid as a Compact Image Code"*, IEEE Trans. Communication, vol. Com – 13, p.p. 532 – 540, 1983.

[Car et. al. 99] C. Carson, M. Thomas, S. Belongie, J. M. Hellerstain and T. Malik, *"Blob world: A System for Region – based Image Indexing and Retrieval"*, in Proc. Int. Conf. Visual Information Systems, pp. 509 – 516, 1999.

[Cha et. al. 87] S. K. Chang, Q. Y. Shi, and C. Y. Yan, *“Iconic Indexing by 2D Strings”*, IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 9, no. 3, p.p. 413 – 428, May 1987.

[Dav 00] K. Davies, "*Digital Color Model Puts Color in a New Light* ", 2000. [URL: http://www.ColorCube.com/articles/models/model,htm].

[DiC Sab 98] J. M. DiCarlo, and J. C. Sabataitis, *"CIECAM97 Color Appearance Model"*, 1998. Available at [URL: ise. stanford. edu / class / psych221/ projects/ 98 / ciecam / Description. html].

[Fil et. al. 04] L. Filipe, M. Vieira., R. D. Erickson, R. Nascimento, A. F. Fernando Jr, R. L. Carceroni, A. A. and Araujo, *"Automatically Choosing Source Color Images for Coloring Grayscale Images"*, IEEE Computer Society, Pages 3 – 4, 2004.

[Gon Woo 01] R. C. Gonzalez, and R. E. Woods, "*Digital Image Processing*", Second Edition – Wesley Publishing, 2001.

[Hat 02] E. F. Hattenberger, *"Graphical User Interface for the CIECAM97s and Revised CIECAM97s Color Appearance Model"*, 2002.

[Hir Kat 92] K. Hirata and T. Kata, *“Rough Sketch – based Image Information Retrieval”*, NEC Res. Develop., vol. 34, pp. 263 – 273, 1992.

[Hsu et. al. 95] W. Hsu, T. S. Chua, and H. K. Pung, *"An Integrated Color – spatial Approach to Content – based Image Retrieval"*, in Proc. 1995 ACM Multimedia Conf., pp. 305 – 313, 1995.

[Hua 98] J. Huang, "*Color – Spatial Image Indexing and Applications"*, Ph.D dissertation, Cornell University, 1998.

[Jag 91] H. V. Jagadish, *"A Retrieval Technique for Similar Shapes"*, Proc. of Int. Conf. on Management of Data, SIGMOID `91, p.p. 208 – 217, May 1991.

[Kau et. al. 95] H. Kauppinen, T. Seppnäen, and M. Pietikäinen, *"An Experimental Comparison of Autoregressive and Fourier – based Descriptors in 2D Shape Classification"*, IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 17, no. 2, p.p. 201 – 207, 1995.

[Lev 97] H. Levkowitz, "*Color Theory and Modeling for Computer Graphics, Visualization, and Multimedia Applications*", Lowell, Massachusetts, USA, 1997.

[Lon et. al. 03] F. Long, H. Zhang, and D. D. Feng, *"Fundamentals of Content – based Image Retrieval"*, p.p. 476, 2003.

[Lus 93] L. Marv, "*Bitmapped Graphics Programming in C++*", 1993.

[Mal 03] J. Maller, *"RGB and YUV Color"* 2003. Available at [URL: http://www.Joemaller.com/fcp/fxscipt-yuv-color.shtml].

[Man et. al. 07] C. D. Manning, P. Raghavan, and H. Schütze, *“An Introduction to Information Retrieval”*, Cambridge University Press, Cambridge, London, 2007.

[Pan et. al. 04] Z. Pan, Z. Dong, and M. Zhang, *"A new Algorithm for Adding Color to Video or Animation Clips"*, Journal of WSCG Pages 2 – 3, 2004.

[Pon et. al. 04] D. Pongas, F. Gunter, A. Uignard, and A. Billard, *"Mobile cameras for the eyes of Robota"*, 2004. Available at [URL: http://www.humanoids.epfl.Ch/research/hardware/report/ Pongas.short.pdf].

[Qiu Lam 03] G. Qiu and K. Lam, *“Frequency Layered Color Indexing for Content – based Image Retrieval”*, IEEE Trans. on Image Processing, vol. 12, no. 1, pp.102 – 113, Jan. 2003.

[Reg et. al. 06] M. Rege, M. Dong, and F. Fotouhi, *“Building a User – centered Semantic Hierarchy in Image Databases”*, Multimedia Systems, Springer – Verlag, vol. 12, no. 45, pp. 325 – 338, March 2006.

[Rie et. al. 01] E. Reinhard, M. Ashikhmin, B. Gooch, and P. Shirley, *"Color Transfer between Images"*, IEEE Computer Graphics and Applications, Pages 21(5): 34 – 41, 2001.

[Sal 89] *“Text Information Retrieval”*, 1989.

[Sel Pen 95] S. Sclaroff, and A. Pentland, *“Modal Matching for Correspondence and Recognition”*, IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 17, no. 6, p.p. 545 – 561, Jun 1995.

[Swa Bal 91] M. J. Swain and D. H. Ballard, *“Color Indexing”*, International Journal of Computer Vision, vol. 7, no. 1, pp.11 – 32, Jan. 1991.

[Tas et. al. 98] T. Tansdizen, L. Akarun, and C. Erosy, *“Color Quantization With Generic Algorithms”*, Signal Processing Image Communication, vol. 12, no. 1, p.p. 49 – 57, 1998.

[Vie et. al. 03] L. F. M. Vieira, R. D. Vilela, and E. R. do Nascimento, *“Automatically Choosing Source Color Images for Coloring Grayscale Images”*, IEEE Computer Society, 2003.

[Yan Alg 94] L. Yang, and F. Algregtsen, *“Fast Computation of Invariant Geometric Moments: A New Method Giving Correct Results”*, Proc. IEEE Int. Conf. on Image Processing, 1994.