DECODING IMAGE SEMANTICS USING COMPOSITE REGION TEMPLATES

John R. Smith and Chung-Sheng Li

IBM T.J. Watson Research Center
30 Saw Mill River Road
Hawthorne, NY 10532
{jrsmith, csli}@watson.ibm.com

ABSTRACT

We present a method for decoding image semantics using composite region templates (CRTs). The CRTs define prototypal spatial arrangements of regions and features in the images. The system classifies unknown images by matching the strings of regions extracted from the images to the templates in a CRT library. We describe the process for generating the CRTs from photographic images by automatically segmenting the images into color regions. We demonstrate that the system performs well in classifying images from ten semantic classes and in searching for images in a large collection.

1. INTRODUCTION

The tremendous proliferation of visual content, i.e., in the World-Wide Web, is increasing the need for more sophisticated methods for automatically analyzing, interpreting and cataloging this imagery [4, 5]. The recent development of content-based query systems has advanced the capabilities for searching for images by color, texture and shape features [1, 3]. However, these systems are limited in their capability for automatically assigning meaningful semantic labels to the images. In particular, effective methods are needed for capturing and labeling the scene information represented by the composition of regions.

In this paper, we present a method for decoding image semantics from the spatial arrangements of image regions using composite region templates (CRTs). The CRTs define prototypal spatial orderings of regions that recur throughout the image collection. Using the CRTs, semantic descriptions of unknown images are generated by matching the arrangements of image regions to the CRTs.

The strength of the method lies in its focus on characterizing the scenes rather than the individual objects within the scenes. The objective is similar to the recent approach in [6], which infers high-level scene properties such as indoor *vs.* outdoor classification by classifying low-level features such as colors and textures. In contrast, the recent body plans approach in [2] focuses on analyzing the main object in a photograph and does not consider other elements of the scene. We consider that all regions in the image, including background, can be useful for characterizing the images.

1.1. Overview

The system generates the CRTs by first segmenting the images into color regions. The segmented images are scanned in a series of top-to-bottom scans to create a set of region strings. The region strings are then consolidated to generate the CRTs. The system can use the CRTs to either classify or search for images. In order to classify the images, the system pools together the CRTs from the training images from each semantic class to construct a CRT library. The system is then able to classify unknown images by decoding their region strings using the entries in the CRT library.

We recognize that, in general, many attributes of the regions such as shape, texture, edges, and motion, can be important in characterizing the regions. Although we use only the color attributes of the regions in this paper, the system, in general, refers to the region attributes as entries in a visual feature library. The visual feature library may contain prototypes of many types of region attributes defined along the dimensions of color, texture, shape and so forth. In practical applications, the visual features may be derived manually, or automatically, such as by sampling the images and clustering the extracted features.

The CRT decoding system utilizes several simplifying assumptions, which we briefly summarize:

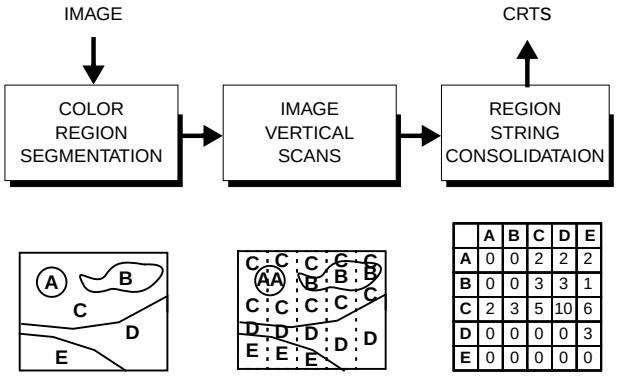
1. Color regions and their relative spatial position characterize the scenes.
2. The relative horizontal position of regions is less important than the relative vertical position.
3. Other meaningful attributes of the regions, such as size and shape, are implicitly captured by the region strings.

We examine the impact of these assumptions by evaluating the system in classifying and searching for images, as follows: in Section 2, we describe the processes for extracting color regions, generating the region strings, and creating the CRTs. In Section 3, we describe the process for classifying unknown images by decoding the region strings using the CRT library. In Section 4, we evaluate the performance of the system in classifying images from ten semantics classes: beaches, buildings, crabs, divers, faces, horses, nature, silhouettes, sunsets, and tigers. We also compare the retrieval effectiveness of the CRT method to methods based on color histograms and texture in retrieving images from a database of 893 color photographs.

2. COMPOSITE REGION TEMPLATES

Humans perceive images by breaking the scenes into surfaces, regions, and objects. The spatial and feature attributes of the regions, and the relationships to other regions are important characteristics of the scenes. Content-based retrieval systems that use global features, such as color histograms, often lose the important spatial information [1, 3]. In order to develop a more powerful representation of image content, we characterize the images by the spatial arrangements of the regions. In a large collection of photographs, there may be recurring regions such as blue skies, oceans, grassy regions, orange horizons, mountains, building facades, and so forth. The objective is to identify and characterize the scenes by the spatial relationships of these regions.

The image analysis process is illustrated in Figure 1. The system first segments the images into color regions. Then, the system extracts five region strings by scanning the segmented images. Finally, the system consolidates the region strings into the set of composite region templates (CRTs). The system uses the CRTs in order to compare, classify and search for images.



	A	B	C	D	E
A	0	0	2	2	2
B	0	0	3	3	1
C	2	3	5	10	6
D	0	0	0	0	3
E	0	0	0	0	0

Figure 1: Overview of the process for generating CRTs from the images.

2.1. Color region segmentation

The system segments the images into homogeneous color regions using the method of color back-projection [3]. The process first palettizes the images using a 166-color quantized HSV color space. Then, the process creates a distance image for each color in the palette that indicates the distance of each pixel to the color. The process filters and thresholds each of these images to segment the image into color regions. We used a color similarity threshold of 0.85× the maximum similarity in HSV color space. We also used a size threshold per extracted region of 0.025× the area of the image. Examples of the color segmented images from the ten semantic classes are illustrated in Figure 2.

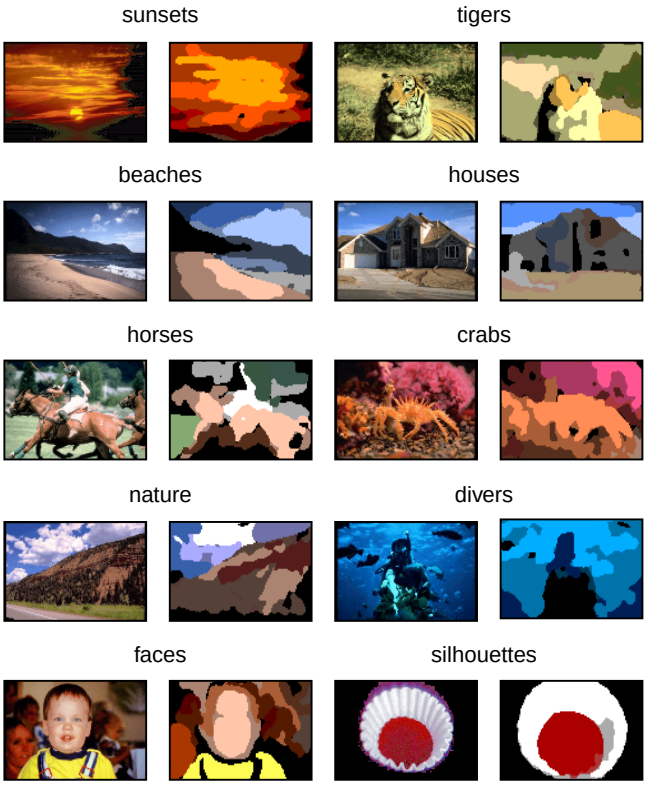


Figure 2: Examples of color region segmentation for images from ten semantic classes: sunsets, beaches, horses, nature, faces, tigers, houses, crabs, divers and silhouettes.

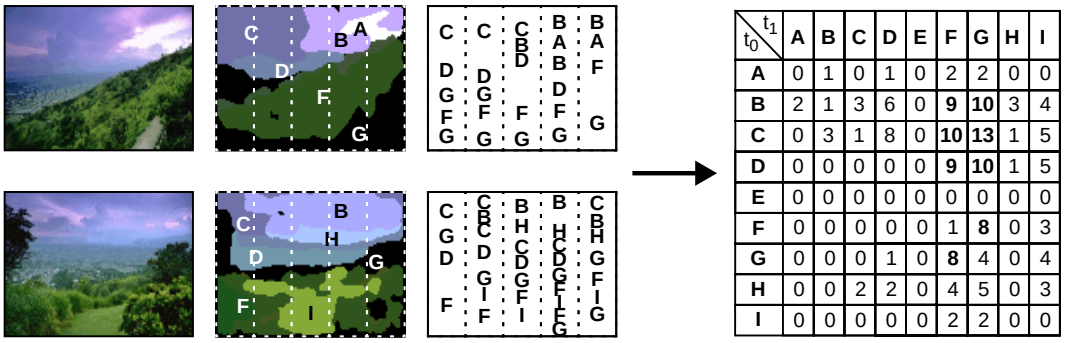
2.2. Region strings

The region strings are generated in a series of five vertical scans of the image that order the segmented regions from top-to-bottom. Note that this process does not use bounding rectangle representations of the segmented regions, rather the color-segmented images are scanned directly. The five vertical scans are equally spaced horizontally.

In general, the attribute of interest of each region is represented by a feature prototype or entry in the visual feature library. We denote the entry by a symbol s_n that indicates the unique identifier or index value of the visual feature. The region strings correspond to a series of the symbols, and are defined as follows:

Definition 1 Region String. *A region string* $\mathbf{S}$ *is a series of symbols* $\mathbf{S} = s_0 s_1 s_2 \ldots s_{N-1}$ *that is generated from the regions of an image, where* s_n *is the symbol value (i.e., color index value) of the* n^{th} *successive region in a top-to-bottom scan.*

In each scan, the symbol value of each consecutive region is concatenated onto the scan's region string. Figure 3 illustrates the region string generation process for two example nature images. In this example, each symbol value (i.e., symbol 'A' in Figure 3(b)) represents the index value of the color of each region in the 166-color HSV color space.



t_0 \ t_1	A	B	C	D	E	F	G	H	I
A	0	1	0	1	0	2	2	0	0
B	2	1	3	6	0	**9**	**10**	3	4
C	0	3	1	8	0	**10**	**13**	1	5
D	0	0	0	0	0	**9**	**10**	1	5
E	0	0	0	0	0	0	0	0	0
F	0	0	0	0	0	1	**8**	0	3
G	0	0	0	1	0	**8**	4	0	4
H	0	0	2	2	0	4	5	0	3
I	0	0	0	0	0	2	2	0	0

(a) (b) (c) (d)

Figure 3: Example of region string extraction, and CRT generation and consolidation processes, (a) two nature images, (b) color segmented regions, (c) extracted region strings, (d) matrix of CRT values.

The top-to-bottom scans capture the vertical positions of the regions, as illustrated in Figure 3(c). The five region strings are not subsequently distinguished by the horizontal positions of the scans.

Notice that in the images in Figure 3(a), the regions corresponding to skies and clouds are found above the regions corresponding to grass and trees. These orderings are reflected in the region strings in Figure 3(c): the symbols 'B,' 'C,' and 'D' (skies and clouds) preceed the symbols 'F,' and 'G' (grass and trees). However, the region strings themselves are difficult to use directly to compare images because of the need to deal with insertions, substitutions and deletions of symbols.

2.3. Region string consolidation

After generating the region strings, the system consolidates them as illustrated in Figure 3(d). Whereas the region strings define a series of symbols, the CRTs define only the relative ordering of symbols, as follows:

Definition 2 CRT. *A composite region template (CRT),* $\mathbf{T}$*, defines a relative ordering of* M *symbols,* $\mathbf{T} = t_0 t_1 \ldots t_{M-1}$.

The system consolidates the region strings extracted from an image or class of images by counting the frequencies of the CRTs in the set of region strings. In general, the CRTs have a length M. However, in this paper we examine the case that $M = 2$. The test for $\mathbf{T} = t_0 t_1$ in region string $\mathbf{S}$ is given by the indicator function $I(\mathbf{T}, \mathbf{S})$, where

$$I(\mathbf{T}, \mathbf{S}) = \sum_{n=0}^{N-1} \sum_{m=n+1}^{N-1} \begin{cases} 1 & \text{if } s_n = t_0 \text{ and } s_m = t_1 \\ 0 & \text{otherwise.} \end{cases} \quad (1)$$

The frequency of each CRT, $\mathbf{T}_i$, in a set of region strings $\{\mathbf{S}_j\}$ is then given by $P(\mathbf{T}_i)$, where

$$P(\mathbf{T}_i) = \sum_j I(\mathbf{T}_i, \mathbf{S}_j). \quad (2)$$

The frequency of each CRT, $\mathbf{T}_i$, in the set of region strings $\{\mathbf{S}_j\}_k$ from semantic class C_k is given by $P(\mathbf{T}_i|C_k)$, where

$$P(\mathbf{T}_i|C_k) = \sum_{\forall_j \mathbf{S}_j \in C_k} I(\mathbf{T}_i, \mathbf{S}_j). \quad (3)$$

The CRTs simplify the problem of comparing images since they use only the relative order of symbols in the regions strings. They are not as sensitive to the insertions, substitutions and deletions of symbols. For example, if symbols are inserted or deleted from the region strings, the CRTs that correspond to the two end-point symbols remain intact. Potentially, many noisy CRTs will be generated within a class of images that correspond to minor differences in the features and positions of regions. However, some of the CRTs will remain dominant for each class of images. This is illustrated for the CRTs in Figure 3(d) that are generated from the nature images, as described next.

The matrix in Figure 3(d) shows the CRT values for the two nature images. The value of the each CRT, $T = t_0 t_1$, where $t_i \in \{A, B, C, D, E, F, G, H, I\}$, is given by the entry in the matrix $[t_0, t_1]$. Although the two nature images have minor differences, and the region strings differ, some of the CRTs, i.e., "BF" "BG", "CF", "CG", "DF" are prevalent in both. Regardless of the minor differences between the images in this class, these CRTs are found to occur with high likelihood.

2.4. CRT library

Given a set of semantic classes of images, the system constructs a CRT library by pooling together the CRTs in each semantic class. The CRT library is defined as follows:

Definition 3 CRT library. *A composite region template (CRT) library is given by the set of tuples:*

$$[\mathbf{T}_i, P(\mathbf{T}_i); P(\mathbf{T}_i|C_0), P(\mathbf{T}_i|C_1), \ldots, P(\mathbf{T}_i|C_{K-1})],$$

where K *is the number of semantic classes, and* $P(\mathbf{T}_i)$ *is given by Eq. 2, and the* $P(\mathbf{T}_i|C_k)$*'s are given by Eq. 3.*

	overall	beach	buildings	crabs	divers	faces	horses	nature	silhouettes	sunsets	tigers
total # images	357	14	56	9	33	55	26	46	41	46	31
# training images	91	7	10	4	10	10	10	10	10	10	10
# test images	266	7	46	5	23	45	16	36	31	36	21
# correctly classified (CRTs)	188	6	30	5	23	19	14	20	20	31	21
# correctly classified (colorhist)	179	5	19	5	23	16	13	24	31	24	19
% correctly classified (CRTs)	**70.7**	**85.7**	**65.2**	100	100	**42.2**	**87.5**	55.6	64.5	**86.1**	**100**
% correctly classified (colorhist)	67.3	71.4	41.3	100	100	35.5	81.3	**66.7**	**100**	66.7	90.5

Table 1: Image semantics classification experiment results using the CRT method compared to color histogram-based classification.

The $P(\mathbf{T}_i|C_k)$'s reflect the frequencies of the CRTs in the training images. For each $\mathbf{T}_i$, the CRT library gives the frequencies in which $\mathbf{T}_i$ is found in each semantic class, $P(\mathbf{T}_i|C_k)$, and in the entire collection, $P(\mathbf{T}_i)$.

3. DECODING IMAGE SEMANTICS

The system uses the region strings and the CRT library for classifying and retrieving the images. In applications involving classification, the system assigns each image to the closest semantic class in the CRT library, as described below. In image retrieval applications, the system retrieves the M most similar images to the query image, where each target image has an entry in the CRT library.

3.1. Image classification

The semantics of an unknown image are decoded from its set of region strings as follows:

1. The systems generates and consolidates the region strings for the unknown image to generate a set of CRTs, $\mathbf{T}'_i$.
2. For each $\mathbf{T}'_i$ from the unknown image, $P(C_k|\mathbf{T}'_i)$ is computed from the entries in the CRT library from:

$$P(C_k|\mathbf{T}'_i) = \frac{P(\mathbf{T}'_i|C_k)}{P(\mathbf{T}'_i)}. \quad (4)$$

3. The system then classifies the unknown image by assigning the image to class l when

$$\forall_{l \neq k}, \ \sum_i P(C_l|\mathbf{T}'_i) > \sum_i P(C_k|\mathbf{T}'_i). \quad (5)$$

That is, class C_l best explains the CRTs represented in the region strings of the unknown image.

3.2. Image retrieval

The image retrieval process is similar to classification using region strings except that each target image forms its own class C_l. The system retrieves the M target images, in ranked order, that have the maximum $P(C_l|\mathbf{T}'_i)$.

4. EVALUATION

We evaluate the region string based semantics decoding method by evaluating its performance in classifying 266 unknown images from ten semantic classes, and retrieving images from a collection of 893 color photographs.

4.1. Experimental setup

The experiments used color images obtained from Expert Software[1]. We used 357 images from the image collection that belong to ten semantic classes. For the image retrieval experiments, we used an additional 536 images that belong outside of these ten semantics classes. We divided the 357 semantic images into non-overlapping training and test sets according to Table 1. We used 91 training images to generate the CRT library and the remaining 266 images for testing the system.

We also performed the classification using 166-bin color histograms defined in HSV color space in order to provide a comparison [3]. Using the histograms, we represented each semantic class by the centroid of the color histograms for the training images in the class.

4.2. Semantic classification results

The classification results are summarized in Table 1. Overall, the semantics decoding system using CRTs provided a classification rate of 0.71. The color histogram performance was slightly lower at 0.67. However, for most of the image classes, the CRT method performed better than color histograms. In particular, the CRT method was significantly better at classifying the images of beaches, buildings, and sunsets. For the silhouette images, the CRTs performed worse than the color histograms. In this case, the dominant property of the silhouette images, a large black background, was not captured well by the CRTs.

The confusion matrix for the semantics classification system is given in Table 2. We can see that with the exception of the faces and nature images, few misclassifications resulted. The faces images were classified correctly only 42.2% of the time, and were misclassified as horses 24.4% of the time. The faces class was the most challenging because the face regions were often only a small part of the image

[1] Expert Software, Inc., 800 Douglas Rd., Coral Gables, FL 33134

and overall, the images had a large variety of backgrounds. The nature images were confused with beaches 19.4% of the time because of the high similarity of the scenes.

Label→	beach	buildings	crabs	divers	faces	horses	nature	silhouettes	sunsets	tigers
beach	**6**	0	0	0	0	0	1	0	0	0
buildings	6	**30**	0	2	0	1	1	0	4	2
crabs	0	0	**5**	0	0	0	0	0	0	0
divers	0	0	0	**23**	0	0	0	0	0	0
faces	5	1	0	0	**19**	**11**	1	0	5	3
horses	0	0	0	0	0	**14**	0	0	0	2
nature	7	3	0	0	0	3	**20**	0	1	2
silhouettes	0	1	0	0	1	1	6	**20**	1	1
sunsets	0	0	0	0	3	1	0	0	**31**	1
tigers	0	1	0	0	0	0	0	0	0	**20**

Table 2: Image classification results confusion matrix for the ten semantic classes.

4.3. Retrieval effectiveness results

We used the full set of 893 images to evaluate the retrieval effectiveness of the CRT method. We analyzed three image queries: sunset images, nature images and diver images. In each of the experiments, the images in the appropriate class were assigned a relevance of one to the query, and the remaining images were assigned a relevance of zero. Each of the relevant images was used in turn to query the database of 893 images. Each query produced a total ordering of the database. We computed the average retrieval effectiveness in terms of precision and recall over the set of queries for each image class.

We performed the queries using CRTs, color histograms and texture. The color histograms used the 166-bin HSV color space [3]. The global texture was defined by a nine-dimensional vector corresponding to the spatial-frequency energy in nine wavelet subbands of the image.

Sunsets. In the sunset image queries (46 queries), the CRT method showed better average retrieval effectiveness than the color histogram and texture methods. For example, in order to obtain half (23) of the sunset images, 61 images needed to be retrieved using CRTs, compared to 118 using color histograms and 333 for texture.

Divers. In the diver image queries (33 queries), the CRT method also showed better average retrieval effectiveness. Using the CRT method, 91% (30) of the diver images were retrieved with a precision of 0.88 (34 retrieved images). Using color histograms, 91% (30) of the diver images were retrieved with a precision of 0.57 (52 retrieved images).

Nature. The nature image queries (46 queries) were most interesting because the CRT method performed better even though color histograms were better at classifying the nature images (see Table 1). Using the CRT method, in the first 20 retrieved images, on average, 8.2 were nature images, compared to 6.1 for color histograms. Figure 4 plots the average retrieval effectiveness in terms of precision *vs.* recall for the nature queries. On average, after the first two returned images, the CRT-method gave higher precision than color histogram and texture-based methods for the same value of recall.

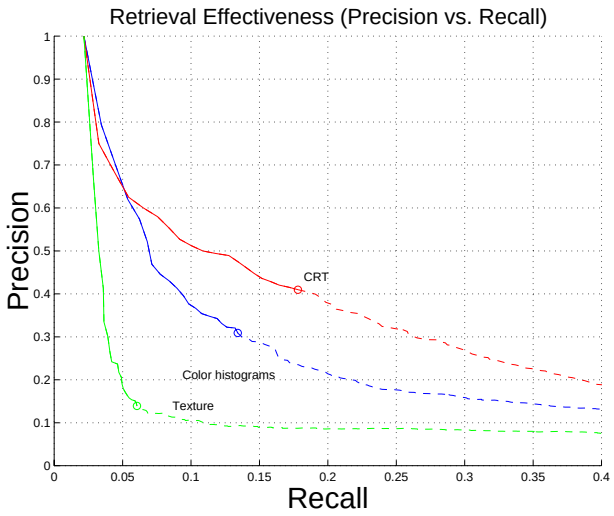


Figure 4: Average retrieval effectiveness of 46 queries for nature images using three methods.

5. SUMMARY

We presented a method for classifying and retrieving images using composite region templates generated from automatically extracted strings of color regions. The system acquires the region strings by scanning the segmented color regions in a series of vertical scans. Images are matched by consolidating the region strings into sets of composite region templates and comparing them. We demonstrated that the system performs well in classifying and retrieving images from ten semantic classes.

6. REFERENCES

[1] M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele, and P. Yanker. Query by image and video content: The QBIC system. *IEEE Computer*, 28(9):23 – 32, September 1995.

[2] D. A. Forsyth, J. Malik, M. M. Fleck, T. Leung, C. Bregler, C. Carson, and H. Greenspan. Finding pictures of objects in large collections of images. In *Proceedings, International Workshop on Object Recognition*. IS&T/SPIE, April 1996.

[3] J. R. Smith and S.-F. Chang. VisualSEEk: a fully automated content-based image query system. In *Proc. ACM Intern. Conf. Multimedia (ACMMM)*, pages 87 – 98, Boston, MA, November 1996.

[4] J. R. Smith and S.-F. Chang. Multi-stage classification of images from features and related text. In *Proc. Fourth DELOS workshop*, Pisa, Italy, August 1997.

[5] J. R. Smith and S.-F. Chang. Visually searching the Web for content. *IEEE Multimedia Mag.*, 4(3):12 – 20, July–September 1997.

[6] M. Szummer and R. W. Picard. Indoor-outdoor image classification. In *IEEE Intl. Workshop on Content-based Access of Image and Video Databases*, Jan 1998.