An Investigation of Image Segmentation Practices

[1]Jaya Krishna Sunkara, [2]E Sasikala, [3]V. V. Satyanarayana Tallapragada, [2]N Dhanalakshmi
[1]Asst. Prof., [2]Asst. Prof.(GKCE), [3]Asso. Prof.(SVEC)
[1]Department of ECE,
[1]Sree Vidyanikethan Engineering College (AUTONOMOUS), Tirupati, India

Abstract : This paper surveys current trends/techniques in image segmentation. The major themes covered by the paper include: segmentation based on template matching, segmentation based upon Haar wavelet based feature vectors, learning of object templates via examples, segmentation by background modeling, Hidden Markov modeling of images, and reformulation of the segmentation problem as a Bayesian classification problem. The relationship between segmentation techniques and the amount of domain knowledge available is discussed**.**

Index Terms **- Background Modelling, Markov Modelling, Segmentation, Template Matching** [WU1]

I. INTRODUCTION

Image segmentation would seem to be a prerequisite for further semantic analysis and storage or retrieval of image based multimedia data. In this paper we will survey some current techniques in the literature on image segmentation. We will see however, that the segmentation problem can be avoided entirely by a statistical reformulation of the image retrieval problem. A common theme relating the techniques is the amount of domain knowledge required with regard to the content of the images. In this paper we will not discuss color-based histogram, texture based modeling, or shape segmentation techniques as these have been reviewed in class and in the textbook. The interested reader can see [1] for a review/analysis of the MPEG 7 shape descriptors, [2] for a polygon boundary based shape segmentation techniques and [3][4] for applications of region based color histograms/texture models to segmentation and image retrieval. In the following sections we will discuss segmentation based upon template matching, background modeling, space (image) transformation, Hidden Markov modeling and Bayesian error minimization.

II. TEMPLATES

Image segmentation by template matching is based upon the assumptions:

- That the class of objects to be detected all share some invariant information,
- That this information is spatially local (has some bounds),
- That this information can be described by a set of generalized features,
- Object detection is the process of correlating the set of features with an area in the image.

Template based segmentation works best in specific domains in which the content of the images is known, for example in the field of face or people detection. In this domain we know that the user is interested in detecting and forming queries about faces/people not the other objects in the image.

Liu and Wang [5] describe a template based face detection and video segmentation system. The template is designed to capture the frontal view of a person as they look at the camera as their application is detection and tracking of faces in video shots such as newscasts and it can be assumed that the subjects will be facing the camera. The template is not rotation invariant and is primarily designed for speed of computation and invariance to background features and individual differences with regard to facial features (e.g., hairstyle, beards, etc.). The template is 20x26 pixels in width and encompasses from above the eyes (eyebrows) down to the upper lip. Averaging the pixel intensities of sections of images from the Purdue AR face database creates the template. The database contains over 4000 images of 126 people under varying lighting conditions, facial expressions and occlusions (e.g., sunglasses, glasses, neck warmers, etc.) [6]. Liu and Wang selected 132 neutral facial expressions from the database for their training examples [5].

Liu and Wang describe a fast template-matching algorithm utilizing dynamic programming. Let F represent the template, which is a M x N rectangle. Let T represent the search rectangle which varies in size from |F| to 2|F|, that is $M \le I \le 2M$ and $N \le J \le 2N$ (see Figure 1). Liu's and Wang's algorithm iteratively warps a sub-rectangle $\hat{F}$ (with top leftmost pixel on S and bottom rightmost pixel in the shaded area Fs) into a rectangle the same size as F. The dynamic programming technique allows the optimal path for this warping to be found. The cost function is a measure of the similarity between the template and the warped image. To find a face object in the image, a test area rectangle is scanned across the image in steps of size F. Scassellati [7] describes a system for real-time face detection. The basic operator for segmenting the image (finding faces) is a ratio-metric template. A template is a constraint pattern on the intensities of pixels in the image. It is implemented as a 14 x 16 pixel mask and a set of relations describing the ratio of intensities between regions of the template (Figure 2). "This template capitalizes on illumination invariant observations. For example, the eyes tend to be darker than the surrounding face, and the nose is generally brighter than its surround." [7] By comparing the ratio of pixel intensities instead of their absolute values, the technique avoids having to normalize the image (perform histogram equalization) to remove the effects of varying illumination. The template is convolved across the image to generate the average gray scale value of the image in that region.

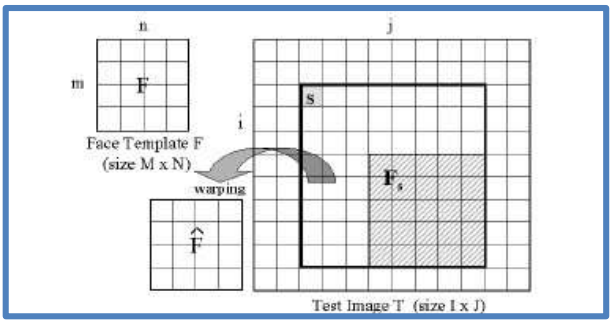

Fig. 1. Liu and Wang's template based search algorithm. T is the test area searched for a face object, F is the face template and F is the portion of the test area warped to the same size as template F. Adapted from [5].

The averages are then used to compute the rations between regions. Only when a subset of the constraint relations, ten of the essential relations (the dark arrows) and eight of the confirming relations (the light arrows) has been satisfied is a face detected in that region. In both template based segmentation algorithms, in order to find faces that are of different sizes than the template, the original image must be resampled (resized) and scanned again for an object match. Liu's and Wang's technique can handle faces twice the size of the template without resampling. Liu and Wang derived their template from the intensity averages of a database of faces; Scassellati did not mention how the values of the ratio-metric relation were determined. It is assumed that a technique similar to Liu's and Wang's could be used. Though not mentioned in the discussion of each technique, templates can be defined for each color map.

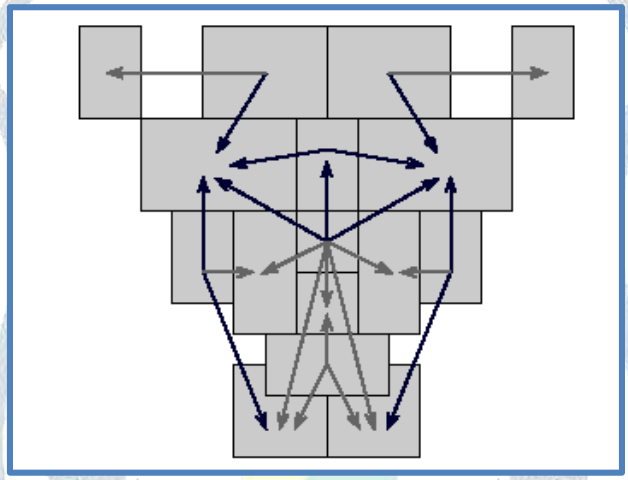

Fig. 2. Ratio-metric template used by Scassellati for real-time face detection. Each block corresponds to a region of the face (eyes, forehead, etc.). The arrows define relations between those regions. Dark arrows are essential relations and light arrows are confirming relations. Only when the ratios between these regions satisfy the constraint relations is a face detected. The figure is from [7].

III. Background Modelling

As mentioned, template segmentation works well when the information about the domain and type of object in that domain is known. What technique will work when nothing is known about the type of objects in the domain? The following background segmentation technique can be used when all that is known about the domain is that the camera is stationary.

Stauffer and Grimson [8] discuss a segmentation scheme that separates the foreground of an image (moving objects) from the background (stationary) objects by modeling the background as a mixture of Gaussians. At any instance in time, the intensity and color distribution of a pixel in an image (its radiance) are the direct result of the lighting and the reflectance properties of the object in that pixels field of view. For each sample frame the probability of observing the current pixels value is determined, if the probability is low, then the background model does not account for this object and it must belong to some new object added to the environment.

Stauffer and Grimson use a weighted sum of Gaussians (Gaussian mixture or GM model) to account for each contribution source for each pixel. When a new pixel sample is obtained, the GM for that pixel is searched for the distribution that best explains its value; where a match is defined as a pixel value within 2.5 standard deviations of a distribution [8]. If none are found, then the k^{th} Gaussian distribution (the one with the lowest weight and hence the one least probably to explain pixel value) is replaced by a new distribution with mean of the current value, an initially high variance and a low weight.

In the case that one of the distributions matches the current value of the pixel, the weight of that distribution is adjusted (by some learning constant α) and all the weights are normalized. $1/\alpha$ defines the time constant of the weight updates and hence the time it will take for new stationary objects added to the environment to become part of the background model.

When a new object is added to the background (moving or stationary) …"it will not in general match one of the existing distributions which will result in either the creation of a new distribution or the increase in the variance of an existing distribution. Also, the variance of a moving object is expected to remain larger than a background pixel until the moving object stops."[8] To determine whether an object is part of the background, or the foreground a threshold is set. If the value of the pixel is described by a

distribution with a weight greater than this threshold then it is part of the background, otherwise it is in the foreground. Pixels that are not accounted for by the background model, are clustered together using a two pass connected components algorithm.

Stauffer's and Grimson's model can segment moving and newly placed objects from backgrounds under conditions of slowly varying lighting, rain and snow. It can handle backgrounds containing standing/moving water, fixed objects affected by the wind (moving leaves, tree branches, flags, etc.) and slowly moving shadows. Due to the value of the learning parameter α and the fact that they mimic real moving objects, the GM model approach will attribute instantaneous changes in lighting (due to flood lights turning on or rapid cloud movement) as objects in the foreground. Stauffer and Grimson have successfully tracked people, cars, mice and fish with a segmentation system based on this technique. Figure 3 show the results of a surveillance system monitoring the traffic outside of their lab. The image in Figure 3 (a) is the input to the system, Figure 3 (b) is the background model (note that the model incorporates the slowly moving building shadow), Figure 3 (c) shows the objects in the foreground and Figure 3 (d) the combined results of background and segmented foreground objects.

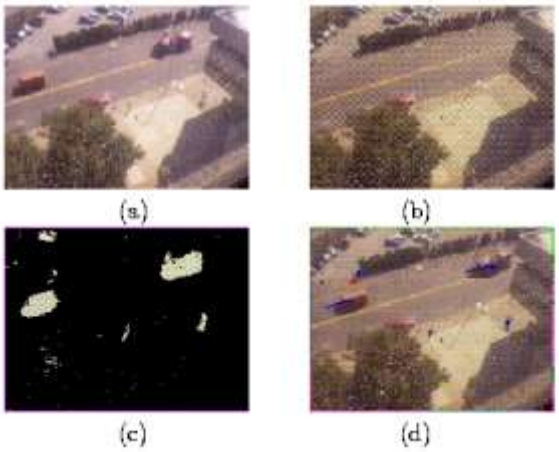

Fig. 3. Results of Stauffer's and Grimson's Gaussian mixture background model segmentation scheme. The image in (a) is the input image, (b) is the representation of the background constructed from the mean of the most probably distributions accounting for the background, (c) is the pixels in the foreground, and (d) is the combined model of background and marked objects in the foreground (marked by blue). Notice that the shadows of the cars/people are considered part of the foreground. This is due to the fact that they move at approximately the same rate as the object itself and thus are not modeled by the background. The images are from [8].

IV. Space Transformations

When attempting to segment an image, instead of performing the analysis in the images domain (intensity values, pixels), one can map the image to a new domain. The new domain should allow the analysis to be performed in an easier fashion. This is the techniques adopted by Poggio and his students at MIT [9-17] . They utilize an over complete 2D Haar wavelet transform to map an image into the spatial domain. The transform they have chosen has the following features that make detecting edges and segmenting objects easier [16]:

- Haar wavelets encode the differences in average intensities between local regions along different orientations, allowing the detection of object boundaries. When performed on the log of pixel intensities encode the ratio of intensities between pixels, gaining the added benefit of being invariant to varying lighting conditions.
- The over complete (quadruple) Haar wavelets transform increases the spatial resolution of the mapping, allowing a one-to-one correspondence between regions in the Haar transform space and the original image. Unlike the normal application of wavelet theory, the goal is not to compress the image, but to aid feature detection.
- The performance of the Haar transform is O(n), where n is the number of pixels. The transformation needs to be done once. All segmentation operations are then performed in the new domain.

To detect objects in this new domain, the Haar coefficients that best describe the features of the object need to be identified. For each class of objects (in the database) the Haar transforms are computed over that object. For example the people detection application [16] computes the Haar transform over an image of 128x64 pixels centered on the image of a person. Three classes of Haar wavelets exist (in the over complete 2D Haar transform): vertical, horizontal and diagonal (Figure 4 (a)), over spatial dimensions of 2x2 (pixels) and greater. The wavelets at the spatial scale thought to encode the important features of the object (for each of these classes) are selected and the coefficients are normalized and averaged. The normalization step is performed to reduce the effects of the lighting variations and averaging identifies the significant wavelets [10][16][18][19].

"Three classes of feature magnitudes will emerge (from the averaged wavelet coefficients): ensemble average values much large than 1 indicate strong intensity difference features that are consistent along all examples, values that are much less than 1 indicate consistent uniform regions, and values that are close to 1 are associated with inconsistent feature, or random patterns" [16][20][21]. Figure 4 (b) depicts the result of determining the significant coefficients for each class of wavelet over spatial dimensions of 16x16 and 32x32 pixels for the people detection application. A gray level is associated with each coefficient, where coefficient greater than 1 are darker. For this particular application, the vertical coefficients capture information about the sides of people; the horizontal coefficients capture information about the heads and the diagonal coefficients captures information about the head, shoulders, hands and feet.

The information derived from the normalization and averaging steps can be used to form a classifier in two ways: similar to [7] and [5] a template can be created from the combination of the vertical, horizontal and diagonal averages, or 2) the identified wavelets can be formed into a feature vector and used by a learning algorithm (neural networks, etc.) to learn the relationships between the coefficients and the target pattern. Object detection in an image with template-based techniques involves convolving the template over the transform space. Regions with the highest value (best match) would then correspond to human shaped objects. To segment the image for objects not the same scale as the template, the image would have to be resized, the Haar wavelets transform recomputed and the new domain searched again.

[9-17] have applied a statistical learning technique for the identification of objects from the set of significant Haar wavelets. In particular they use the Support Vector Machines (SVM) model [22]. A training set of human shapes and background objects are created. As an example, the training set size for the people detection system required 1,848 positive (people) patterns and 11,361 non-people patterns [16]. The SVM is trained over this set until some error threshold is reached.

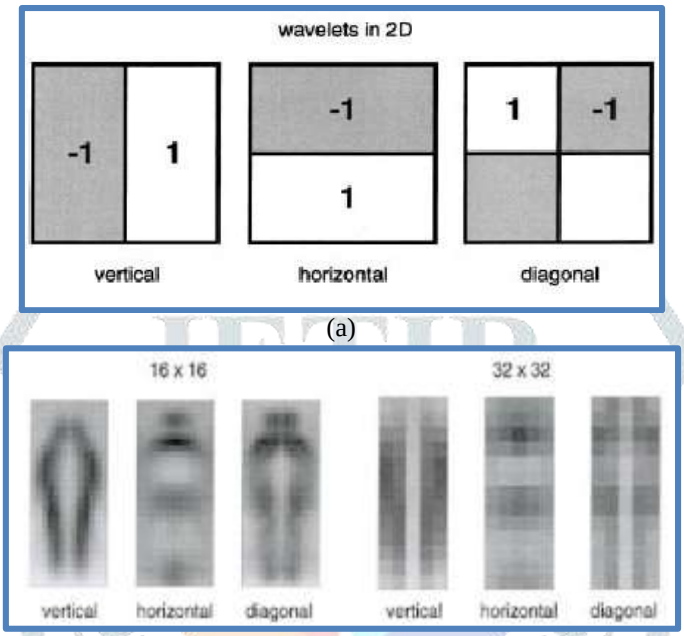

Fig. 4. The three classes of Haar wavelets in the over complete 2D Haar wavelet transform (a). The ensemble averages of the normalized Haar coefficients for the 128x64 training images used to select significant Haar wavelets (b). Each square in the image corresponds to one 16x16 (or 32x32) vertical, horizontal or diagonal wavelet. Haar coefficients with a value of 1 are coded gray, with the coefficients greater than 1 coded towards black. Adapted from [16].

Once the SVM has been trained, similar to the template-matching scheme, the classifiers detection window is scanned across the transformed image space. The output of the classifier indicates the presence of a humanoid object. Multiple scaled objects must be handled by resizing and rescanning the original image. In a comparison between the template based and SVM classifier technique, Oren et al. [10] saw approximately a 20% improvement of SVM classifier in detection rate and the false positive rate dropped from 1:5000 to 1:15,000.

Poggio and his students [9-17] have demonstrated the ability to recognize (detect and classify) people, cars, faces, mouth eyes and noses in static images, in real world image domains (offices, traffic, sidewalks, etc.). Figure 5 is an example of the performance of the people detection systems (using the SVM classifier) in a real world domain.

The above technique is useful for the detection/segmentation of arbitrarily shaped objects in the image. It suffers from poor performance during the scanning stage, on the order of O(n*d) where "n" is the number of pixels in the image and "d" is the size of the template or classifier detection window. The constant can be quite large when multiple scale objects must be handled, as it takes O(n) work to resize the image recompute the Haar wavelet transform. The performance can be improved in video sequences by using motion information to limit areas of the image scanned. In a content-based retrieval application, the complete Haar wavelet coefficients for the segmented region would need to be used for the query, as these coefficients encode specific information about the objects texture and color. Like the previous template based matching techniques, some domain knowledge is required, in this case knowledge about the types of objects of interest in order to train the SVM classifier and to create the object template.

V. Hidden Morkov Model Segmentation

Instead of the previous specific techniques of searching the image for some type of feature and segmenting around that feature, DeMenthon et al. [23] describe a Hidden Markov model technique to segment an image. In their paper they …"focus on the simplest Markov mesh model, a second-order model, i.e. a model in which the label likelihoods depend on the labels of only two past pixel neighbors when the pixels are observed by raster scan."[23] Fig. 6 shows what this looks like, the probability of pixel (u,v) being in state q is conditional to the states immediately to the left and above that pixel.

Associated with each pixel then are an observation vector and a hidden state. The observation vector is the set of parameters (of interest) associated with each pixel, such as color, or the average intensity of the image region centered on that pixel. The hidden state is a label for that pixel. The modeling process determines which label (or state) best describes the value of each pixel. A pixel belongs to a particular state based on the estimate of the probability P(observation of parameter for pixel | state of pixel). The

collections of states (and the pixels that belong to them) segment the image, where each segment is a region with similar observation properties. The number of states, and hence objects/regions in the segmented image is a settable parameter.

DeMenthon et al. develop a dynamic programming based learning algorithm for training the HM model. It runs in time $O(Uvn^3)$, where UxV is the size of the image and n is the desired number of states or segments in the image. "The results (of the training algorithm) are a statistical model of the image, a segmentation of the image, and the evaluation of the probability of observing the image given the model."[23]

Fig. 5. Sample result from the people detection system using the SVM classifier. Adapted from [24].

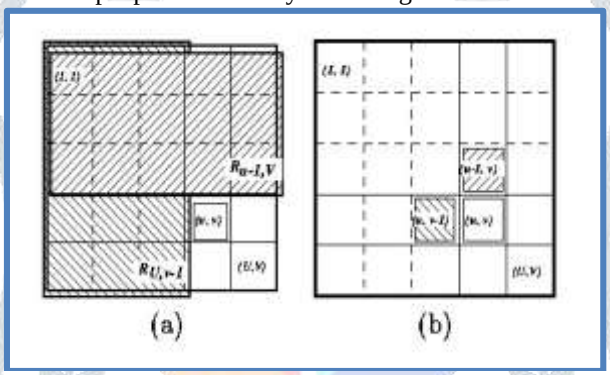

Fig. 6. The conditional probability relationship of a pixel for DeMenthon's et al. second order causal Markov mesh model. (a) The relationship of the probability between pixel (u,v) and the pixels to the east and north of that pixels. The probability of pixel (u,v) being in state q is conditional to the probabilities of the state configurations for pixels in rectangles $R_{U,v-1}$, $R_{u-1,V}$. (b) This can be factored to the probability of pixel (u,v) being in state q is conditional to the the states of immediate neighbor pixels (u,v-1) and (u-1,v). From [23].

Figure 7 show the application of the HM mesh model-training algorithm. The image has been segmented into ten regions based on each pixels color value. As can been seen the algorithm does segment the image, but not necessarily in a manner that preserves semantic meaning. In the case of the face of the girl it is composed of at least three segments, not composed of one segment as human might assign if they were segmenting the image. The domain knowledge required for the application of this technique are the parameters to model (e.g., color space, DCT block, etc.) and an idea of the number of states needed to adequately segment the image. Less knowledge than what is required for the previous techniques, but still some assumptions about the content and type of images need to be made.

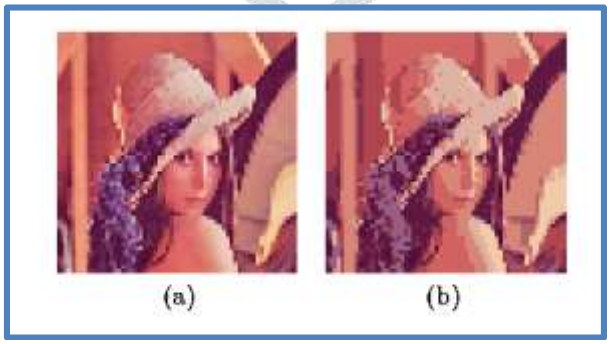

Fig. 7. Results of segmenting a 64x64 color image using DeMenthon's et al. Hidden Markov mesh model. The original image (a) has been segmented in ten regions (b). From [23].

VI. Probabilistic Model and Content Based Retrieval

In the domain of content-based retrieval, we desire to organize and query multimedia data based on the content of that data. Consider the generic image domain, images are complex; they are composed of spatially distinct regions with texture and/or color content. What is the retrieval process for this type of image? The techniques we have surveyed would suggest that the image be broken down into feature vectors describing each image attribute (the color/texture). These feature vectors can be global

descriptions of the image or local to the object(s) comprising the image (including spatial information). The generated feature vectors would then be used to search the database for an image that best matches the query. When the generated feature vectors are compared to those in the database (i.e., the indexes), some metric must be used to compare the similarity of those features. "The standard solution is to evaluate similarity according to each of the attributes and obtain an overall similarity measure by weighting linearly the individual distances" [25] (assuming some type of Euclidian like distance metric). Some of the problems associated with this process are [26][25]:

- It is difficult to find a global descriptor that accounts for (describes) the content of the image (e.g., a color histogram will not necessarily capture the texture information or the spatial organization of an image),
- Use of a global based descriptor may make it impractical to support local region based queries,
- It is difficult to determine objects in the image and to segment around those objects, thus it is not an easy to generate local descriptions,
- What is the best similarity metric to use? What are the domain assumptions behind this metric?
- Weight selection (for the comparison step) has no standard solution and becomes domain and implementation specific.

A. *Retrieval as a Classification Problem*

What is desired is a domain independent technique to describe and to compare image content. That is a technique that works equally well on images with high texture content, spatial discontinuity (many different objects) and with many colors. To address these problem Vasconcelos and Lippman [27-34] propose reformulating the content-based retrieval process as a Bayesian classification problem.

The goal of Bayesian classification is to minimize the error associated with deciding class membership. In minimizing retrieval error we ask the question: "Given a set of features X drawn from a class Y, what is the probability that we will retrieve some set of images g*(X), not belonging to class Y." The optimal solution to this question can be written as:

$g^*(X) = \arg\max_i P(Y=i \mid X)$.

Which reduces to:

$g^*(X) = \arg\max_i P(X \mid Y=i)P(Y=i)$

where $P(X \mid Y=i)$ is the likelihood function or the feature representation for the i^{th} class and $P(Y = i)$ is its prior probability. The advantages of a Bayesian formulation are [26][31][33]:

- Since the feature vector X can be any subset of a given query image, Bayesian retrieval will work for both image based and region based queries,
- Since feature vector X can be a set of independent features (e.g., features derived from text, audio, and image regions), Bayesian formulation allows uniform handling of multi-modal data and provides a solution to the weighting problem,
- The prior probability term allows the incorporation of prior knowledge into the retrieval process, thus providing a mechanism to implement statistical based learning.

B. *Similarity*

Vasconcelos and Lippman have shown [31][32] that the majority of similarity metrics in common use for comparing features are special cases of the Bayesian formulation. In [32] they discuss the different similarity techniques and derive them from the Bayesian classifier. Vasconcelos and Lippman define Maximum Likelihood (ML) similarity criteria to be when the prior probabilities in equation (2) are assumed to be independent and equally likely, that is $P(Y = i) = 1/n$, where n = the number of classes. They find that the ML criterion makes the least assumptions about a domain and provides the most generality when comparing attributes of different modalities. Fig. 8 summarizes the results of their analysis on similarity metrics.

VII. Gaussian Mixture and Embedded Multiresolution Mixture Model

The main goal for using Bayesian classification is to minimize retrieval error. Vasconcelos and Lippman analysis in [31] showed that one of the major factors in minimizing retrieval error is the choice of feature representation, that is how well $P(X|Y = i)$ can be estimated. For example a color histogram representation of an image does not capture local spatial structure such as texture and is inadequate for describing local queries that may be based on this structure, leading to poor retrieval results. The limitation on histograms is not that they can't capture spatial information but that their complexity increases exponentially as the dimension of the feature space increases making them difficult to use [33].

As suggested in [28][29][31][33] a good representation for the probability density $P(X|Y = i)$ is the Gaussian Mixture (GM) model. The Gaussian mixture approximates a density region by a weighted sum of normal distributions. It has the following properties [31]:

- They are able to model arbitrary probability densities,
- Their complexity is quadratic in the dimension of the feature space and hence can handle high dimensional feature spaces.

What this means in practice is that a GM model has the same expressive power as histograms (but without the cost) and can thus be used to represent color content. Second, since it is composed of Gaussian distributions, it can equally well represent the content of images with lots of texture. For complete details the reader is referred to [29][33].

The drawback to the GM model is that its complexity is $O(YRCn^2)$, where Y is the number of classes in the database, R is the number of features in the image, C is the number of components (number of Gaussian terms in the in the approximation of the density function) and n is the dimension of the feature space (the number of pixels in the image for a color image) [33]. Vasconcelos and Lippman propose a modified version of the GM model that does not suffer from the complexity, it is called the Embedded Multiresolution Mixture (EMM) model [33]. Essentially it is a Gaussian mixture model of an image that has been transformed to the spatial domain through use of the Discrete Cosine Transform (DCT). The complexity is $O(C^2n)$, where C is

between 8 to 16. The EMM shares the same capabilities as the GM model in that it can be used to describe both color and texture images compactly.

The DCT is computed in 8x8 or 16x16 sliding blocks over the image (by increments of 2 pixels), then a GM is created for the transformed space. The coefficients in a DCT are organized by spatial energy in the image; the coefficient f_{11} (see Figure 9) captures the DC energy of the image (the intensity) while the remaining coefficients capture the increasing high frequency (edge content and hence texture) information of the image.

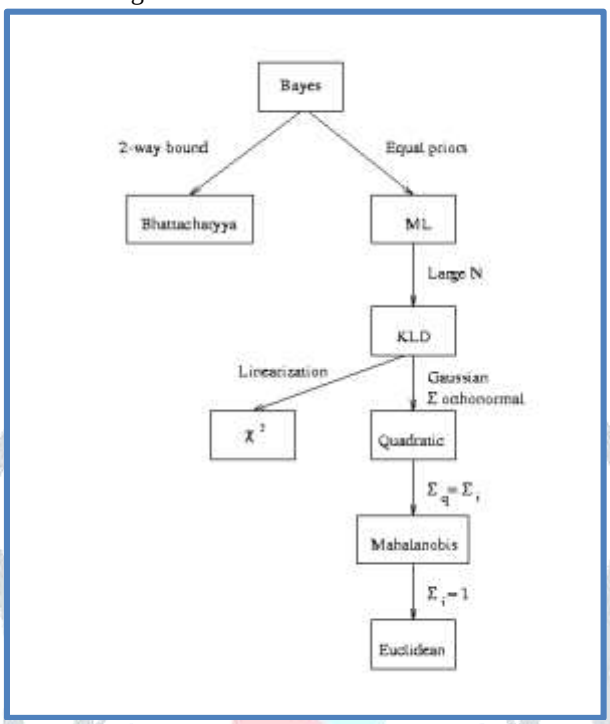

Fig. 8. From [32], the relationship between Bayes similarity criteria and those of commonly used metrics. ML is the Maximum Likelihood criteria, KLD is the Kullback-Leibler divergence.

$$\begin{bmatrix} f_{11} & f_{12} & f_{13} & f_{14} \\ f_{21} & f_{22} & f_{23} & f_{24} \\ f_{31} & f_{32} & f_{33} & f_{34} \\ f_{41} & f_{42} & f_{43} & f_{44} \end{bmatrix}$$

Fig. 9. DCT coefficient matrix. The coefficients are ordered by spatial energy in the original image. Coefficient f_{11} represents the DC (average intensity) energy of the image; higher order coefficients (f_{ij}, i =j ≠ 1) represent the energy in the image due to texture (edge content).

We can think of the DCT as an ordered family of subspaces: coefficient f_{11} for the DC subspace, coefficients $\begin{bmatrix} f_{11} & f_{12} \\ f_{21} & f_{22} \end{bmatrix}$ forming a larger dimension subspace which includes more texture information and so on. Each collection of subspaces (of some dimension from 1 to 8 for the 8x8 DCT) of the transformed image is described by a GM model. This suggests the following algorithm to improve retrieval speed [33]:

1. Take the region in the query and convert to spatial domain using 8x8-sliding DCT.
2. Find the K_1 best matches in the database using only the DC coefficients.
3. Next find the K_2 best matches from the K_1 matches using the first two DCT coefficients.
4. Continue in this manner until the best K_n are found using all of the DCT coefficients.

The complexity of the algorithm is $O(C^2(1 + 2K_1/Y + \ldots + nK_{n-1}/Y)) = O(C^2)$ for $K_i << Y$ as compared to the $O(C^2{}_n)$ for the naïve algorithm. As summarized by [33] "…probabilistic retrieval using embedded mixture feature representation can be seen as an extension of histogram based techniques where, after the best histogram matches are found, subsequent steps are taken to ensure that images whose spatial dependencies resemble most those of the query image are retrieved first. In this way spatial statistics of the query image can be accounted for during retrieval without a significant increase in overall complexity."

VIII. Region Based Retrieval with Relevance Feedback

In [30][34] Vasconcelos and Lippman extend their work and develop a retrieval system that learns (improves its query performance) using relevancy feedback. The approach uses the prior probability term (P(Y = i) in equation (2)) to provide information from the last retrieval to the current retrieval. Vasconcelos and Lippman allow the user to select the retrieved images with the best match and use this information to computer the prior probabilities. Figure 10 shows four screen shots of their system during a retrieval process. In the first shot (top left hand corner) the user has requested to retrieve images containing a clay mug.

The query image is shown on the lower left side while the desired target (not known to the system) is shown on the upper left. The system (using the EMM model and assuming Maximum Likelihood criteria for the first query, P(Y=i) = 1/n) retrieves the best matches to this query, unaware that it is the mug feature that is relevant to the user query. Hence the system retrieves images containing mugs, blue doughnuts, cars, etc. After retrieval the user ranks each image based on the number of relevant objects that the system returned with regard to the context if the users interests. In this case all the images with a brown mug are ranked with a 1 while the remaining images remain unranked.

Fig. 10. Result of relevancy feedback learning for a database containing mixed objects of varying textures and color. The progression of images is from top left to bottom right. The desired target image is shown on the top left of each screen shot and the query image is shown on the bottom left. The numbers above each retrieved image correspond to the relevancy ranking assigned by the user. It corresponds to the number of object in the image that are of interest to the users. The images is taken from [30].

In the second query (upper right screen shot) the user selects one of the retrieved ranked images as the basis for the next query. In this way the system learns what object is of importance to the user. In the third screen shot (lower left) the user has retrieved as a result an image that contains a plastic tape dispenser, but not a clay cup. Since this image is also relevant to their query they rank it as well and use it as the basis for their next query. If the system was not using relevance feedback (learning) it might then just retrieve images containing tape dispensers (and anything similar), but it knows that the user finds both clay cups and tape dispenser of interest. The last screen shot shows the results of this query, images containing both tape dispensers and cups are returned. The desired target is one of the returned images. Issues concerning negative feedback, session learning versus cross session learning (learning of user preferences) and forgetfulness are discussed further in [30][34].

What is the power of a probabilistic approach to image segmentation and retrieval? As we have seen it allows:

- a generic similarity criteria to be used that is not domain specific, little domain knowledge about the type of objects or images is required,
- avoids the problem of image segmentation by using a representational schema that is powerful enough encode global and local features of the image,
- supports region based queries,
- is computationally efficient,
- contain the same expressive power as color histograms and texture based descriptors.

IX. Conclusions

In this paper we have surveyed five techniques for the segmentation of images. The conditions for their application have varied depending on the amount of domain knowledge we have. When no assumptions are to be made about the domain (no knowledge of the image content), then statistical techniques like that of Vasconcelos and Lippman's EMM model would seem to be appropriate. This comes at a price though of not being able to query a database of images based on semantic content, instead some type of query by example with user feedback must be used. This implies that it would be difficult to automate the query process by removing the human from the loop. When domain knowledge is known then techniques like the template segmentation schemes of Liu and Wang, Scassellati and Poggio can be used. The advantages to these schemes are that they segment around objects. Higher level reasoning can them be performed on these objects to assist the database storage or retrieval process. The limitation of course is that we can only segment around objects we know beforehand will exist in the images. But as we have existing databases of specific domain knowledge (e.g., employees, students information) this may not be such a drawback. In addition to the papers surveyed above the reader can find more information regarding techniques for face and people detection in the following papers: Essa [36] and Yang et al. [35] which survey research on seeing people with regard to tracking, finding faces, pose estimation, etc. We have only discussed images segmentation, Wang, Liu and Huang in [37] survey techniques to utilize both audio and visual cues in the segmentation of multimedia data (videos).

References

[1] Latecki, L., Lakamper, R., and Eckhardt, U., "Shape Descriptors for Non-Rigid Shapes with a Single Closed Contour," IEEE Computer Society Confernece on Computer Vision and Patterrn Recognition (CVPR 2000), Held on Hilton Head Island, SC.

[2] Latecki, L. J. and Lakamper, R., "Application of Planar Shape Comparison to Object Retrieval in Image Database," Pattern Recognition, vol. pp. 30.

[3] Duygulu, P., Carkacioglu, A., and Yarman-Vural, F., "Multi-Level Object Description: Color or Texture," IEEE Balkan Conference on Signal Processing, Communications, Circuits, and Systems, Istanbul, Turkey, 2000.

[4] Xu, Y., Duygulu, P., Saber, E., Tekalp, A. M., and Yarman-Vural, F. T., "Object Based Image Retrieval Based on Multi-Level Segmentation," IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP2000), Istanbul, pp. IV-2019-2022, 2000.

[5] Liu, Z. and Wang, Y., "Face Detection and Tracking in Video Using Dynamic Programming," ICIP-2000, Vancouver, Canada.

[6] Martinez, A. M. and Benavente, R., Computer Vision Centre, Purdue University, Jun 1998.

[7] Scassellati, B., "Eye Finding via Face Detection for a Fovea ted, Active Vision System," AAAI 98.

[8] Stauffer, C. and Grimson, W. E. L., "Adaptive Background Mixture Models for Real-Time Tracking," IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 246, 1999.

[9] Oren, M., Papageorgiou, C. P., Sinha, P., Osuna, E., and Poggio, T., "Pedestrian Detection Using Wavelet Templates," Proceedings of Computer Vision and Pattern Recognition, Puerto Rico, pp. pp. 193-199, 1997.

[10] Oren, M., Papageorgiou, C. P., Sinha, P., Osuna, E., and Poggio, T., "A Trainable System for People Detection," Proceedings of Image Understanding Workshop, New Orleans, LA, pp. pp. 207-214, 1997.

[11] Papageorgiou, C. P. and Poggio, T., MIT AI Lab, Massachusetts Institute of Technology, A.I. Memo No. 1673, C.B.C.L. paper No. 180, Oct 1999.

[12] Papageorgiou, C., Evgeniou, T., and Poggio, T., "A Trainable Pedestrian Detection System," Proceedings of Intelligent Vehicles, Stuttgart, Germany, pp. pp. 241-246, 1998.

[13] Papageorgiou, C. and Poggio, T., "A Pattern Classification Approach to Dynamical Object Detection," Proceedings of International Conference on Computer Vision, Kerkyra, Greece, pp. pp. 1223-1228, 1999.

[14] Papageorgiou, C. and Poggio, T., "Trainable Pedestrian Detection," Proceedings of International Conference on Image Processing, Kobe, Japan, 1999.

[15] Papageorgiou, C. and Poggio, T., "A Trainable Object Detection System: Car Detection in Static Images," MIT, 1673, Oct 1999.

[16] Papageorgiou, C. and Poggio, T., "A Trainable System for Object Detection," International journal of computer vision, vol. 38, pp. 15-35, 2000.

[17] Papageorgiou, C. P., Oren, M., and Poggio, T., "A General Framework for Object Detection," Proceedings of International Conference on Computer Vision, Bombay, India, pp. pp. 555-562, 1998.

[18] Jaya Krishna Sunkara, E Navaneethasagari, D Pradeep, E Naga Chaithanya, D Pavani, D V Sai Sudheer, "A New Video Compression Method using DCT/DWT and SPIHT based on Accordion Representation", I.J. Image, Graphics and Signal Processing, pp. 28-34, May 2012

[19] Jaya Krishna Sunkara, Purnima Kuruma, Ravi Sankaraiah Y, "Image Compression Using Hand Designed and Lifting Based Wavelet Transforms", International Journal of Electronics Communications and Computer Technology, Vol. 2 (4), 2012.

[20] Jaya Krishna Sunkara, Sundeep Eswarawaka, Kiranmai Darisi, Santhi Dara, Pushpa Kumar Dasari, Prudhviraj Dara, "Intensity Non-uniformity Correction for Image Segmentation", IOSR Journal of VLSI and Signal Processing, Volume 1, Issue 5, PP 49-57, Jan.-Feb 2013.

[21] Jaya Krishna Sunkara, Uday Kumar Panta, Nagarjuna Pemmasani, Chandra Sekhar Paricherla, Pramadeesa Pattasani, Venkataiah Pattem, "Region Based Active Contour Model for Intensity Non-uniformity Correction for Image Segmentation", International Journal of Engineering Research and Technology (RIP), Volume 6, Number 1, pp. 61-73, 2013.

[22] Evgeniou, T., Pontil, M., and Poggio, T., "Statistical Learning Theory: A Primer," International journal of computer vision, vol. 38, no. 1, pp. 9-13, 2000. 0920-5691.

[23] DeMenthon, D., Stuckelberg, M., and Doermann, D., "Image Distance using Hidden Markov Models," International conference Pattern Recognition (ICPR 2000): Image, Speech and Signal Processing, Barcelona, Spain.

[24] Papageorgiou, C. P., "Object and Pattern Detection in Video Sequences," 1997. S.M. Thesis MIT.

[25] Vasconcelos, N. and Lippman, A., "A Probabilistic Architecture for Content-based Image Retrieval," IEEE Conference on Computer Vision and Pattern Recognition (ICPR 2000), Hilton Head Island, SC, 2000.

[26] Vasconcelos, N. and Lippman, A., "Bayesian Representations and Learning Mechanisms for Content Based Image Retrieval," Proceedings of the SPIE - The International Society for Optical Engineering Storage and Retrieval for Media Databases 2000, San Jose CA, pp. p. 43-54, 2000.

[27] Vasconcelos, N. and Lippman, A., "A Bayesian Framework for Semantic Content Characterization," IEEE Conference Computer Vision and Pattern Recognition (ICPR 1998), Santa Barbara; CA, 1998.

[28] Vasconcelos, N. and Lippman, A., "Learning Mixture Hierarchies," Advances in Neural Information Processing Systems," vol. 11, pp. 606-612, 1999.

[29] Vasconcelos, N. and Lippman, A., "Feature Representations for Image Retrieval: Beyond the Color Histogram," IEEE International Conference on Multimedia and Expo (ICME 2000), New York, NY, 2000.

[30] Vasconcelos, N. and Lippman, A., "Learning Over Multiple Temporal Scales in Image Databases Lecture notes in computer science," Lecture Notes in Computer Science, 0302-9743 2000; pp. 33-47, 2000.

[31] Vasconcelos, N. and Lippman, A., "A Probabilistic Architecture for Content-based Image Retrieval," IEEE Conference on Computer Vision and Pattern Recognition (ICPR 2000), Hilton Head Island, SC, 2000.

[32] Vasconcelos, N. and Lippman, A., "A Unifying View of Image Similarity," IEEE International conference Computer Vision and Image Analysis: Pattern Recognition (ICPR 2000), Barcelona, Spain, 2000.

[33] Vasconcelos, N. M. and Lippman, A. B., "Embedded mixture modeling for efficient probabilistic content-based indexing and retrieval," Proceedings- SPIE Conference on Multimedia Storage and Archiving Systems, Boston; MA, pp. 3527-12, 1998.

[34] Vasconcelos, N. and Lippman, "A. Learning from user feedback in image retrieval systems," Advances in neural information processing systems," Proceedings of the Dec. 1999 conference, eds. Solla, S. A., Müller, K.-R., and Leen, T. K. Cambridge, Mass.: MIT Press, 2000.

[35] Yang, M.-H. , Ahuja, N., and Kriegman, D., "A Survey on Face Detection Techniques," IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI) (2000), vol.S. M. Metev and V. P. Veiko, Laser Assisted Microtechnology, 2nd ed., R. M. Osgood, Jr., Ed. Berlin, Germany: Springer-Verlag, 1998.

[36] Essa, I. A., "Computers Seeing People," AI magazine, vol. 20, no. 2, pp. 69-82, 1999. 0738-4602.

[37] Huang, J., Liu, Z., and Wang, Y., "Joint Video Scene Segmentation and Classification based on Hidden Markov Model," IEEE International Conference on Multimedia and Expo (ICME), New York, NY.